



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ & ΣΥΣΤΗΜΑΤΩΝ  
ΠΛΗΡΟΦΟΡΙΚΗΣ

**Επιλογή Δικτύου Πρόσβασης Ελεγχόμενη από Κινητά  
Τερματικά, σε Ασύρματα Περιβάλλοντα Πέραν της Τρίτης  
Γενιάς**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ευγενία Φ. Αδαμοπούλου

Κωνσταντίνος Π. Δεμέστιχας

**Επιβλέπων :** Μιχάλης Ε. Θεολόγου  
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2005





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ & ΣΥΣΤΗΜΑΤΩΝ  
ΠΛΗΡΟΦΟΡΙΚΗΣ

**Επιλογή Δικτύου Πρόσβασης Ελεγχόμενη από Κινητά  
Τερματικά, σε Ασύρματα Περιβάλλοντα Πέραν της Τρίτης  
Γενιάς**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

Ευγενία Φ. Αδαμοπούλου

Κωνσταντίνος Π. Δεμέστιχας

**Επιβλέπων :** Μιχάλης Ε. Θεολόγου  
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 3<sup>η</sup> Ιουνίου 2005.

.....  
Μ. Θεολόγου  
Καθηγητής Ε.Μ.Π.

.....  
Ε. Συκάς  
Καθηγητής Ε.Μ.Π.

.....  
Μ. Αναγνώστου  
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2005

.....  
Ευγενία Φ. Αδαμοπούλου  
Κωνσταντίνος Π. Δεμέστιχας

Διπλωματούχοι Ηλεκτρολόγοι Μηχανικοί και Μηχανικοί Υπολογιστών Ε.Μ.Π.

Copyright © Ευγενία Φ. Αδαμοπούλου, 2005  
Copyright © Κωνσταντίνος Π. Δεμέστιχας, 2005  
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

## Περίληψη

Η επόμενη γενιά συστημάτων κινητών τηλεπικοινωνιών αναμένεται να αποτελείται από ετερογενή δίκτυα, τα οποία, χρησιμοποιώντας διαφορετικές τεχνολογίες πρόσβασης, θα είναι ικανά να υποστηρίζουν ασύρματα τερματικά, καθένα από τα οποία θα μπορεί να λειτουργεί εναλλακτικά στις διάφορες τεχνολογίες πρόσβασης που είναι διαθέσιμες στο σύστημα. Σκοπός της παρούσας εργασίας είναι η ανάπτυξη μιας αρχιτεκτονικής διαχείρισης για τις σύνθετες αυτές δομές πρόσβασης η οποία θα περιλαμβάνει ένα καινοτόμο Σύστημα Διαχείρισης Τερματικού (ΣΔΤ), ενσωματωμένο στο κινητό τερματικό και ικανό να επιλέγει, δυναμικά και ανεξάρτητα, με διαφανή τρόπο, το κατάλληλο δίκτυο πρόσβασης από το οποίο θα εξυπηρετηθούν ικανοποιητικά οι υπηρεσίες, όσον αφορά το κόστος και το επίπεδο ποιότητας (QoS). Η καινοτομία του στηρίζεται στη λειτουργικότητα της προτεινόμενης δομής, η οποία υποστηρίζει την ελεγχόμενη και εκκινούμενη από το τερματικό επιλογή δικτύου πρόσβασης, επεκτείνοντας, συνεπώς, την ευφυΐα από το δίκτυο στο τερματικό. Προκειμένου να υλοποιηθεί το παραπάνω ΣΔΤ, αναπτύχθηκε μια εφαρμογή, βασισμένη σε Java, η οποία αποτελείται από δυο βασικά τμήματα. Το πρώτο τμήμα είναι μια γραφική διεπαφή χρήστη (GUI), μέσω της οποίας ο χρήστης θα μπορεί να καθορίζει, να αποθηκεύει και να τροποποιεί δυναμικά τις προτιμήσεις του. Η γραφική αυτή διεπαφή δίνει τη δυνατότητα στο χρήστη να προσδίδει διαφορετική βαρύτητα σε παραμέτρους που μπορεί να επηρεάζουν τη διαδικασία επιλογής δικτύου πρόσβασης, όπως είναι η «ποιότητα», ο «πάροχος» και το «κόστος». Το έτερο σημαντικό τμήμα του ΣΔΤ είναι η λειτουργία «Ευφυούς Επιλογής Δικτύου» (ΕΕΔ). Η τελευταία είναι υπεύθυνη για τον ανεξάρτητο καθορισμό της βέλτιστης τεχνολογίας και σημείου πρόσβασης από τα οποία θα λαμβάνονται οι υπηρεσίες που ο χρήστης εκκινεί ή ήδη χρησιμοποιεί, με τον καλύτερο δυνατό τρόπο. Η προαναφερθείσα διαδικασία βελτιστοποίησης λαμβάνει υπόψη της μετρήσεις από τα σημεία πρόσβασης, που αντανακλούν την κατάσταση και τη διαθεσιμότητα πόρων του δικτύου, τις προτιμήσεις του χρήστη, καθώς και τις προδιαγραφές των υπηρεσιών. Δύο τυπικά σενάρια έχουν προσομοιωθεί με σκοπό την επίδειξη της λειτουργικότητας του αλγορίθμου «Ευφυούς Επιλογής Δικτύου».

## Λέξεις Κλειδιά

κινητό τερματικό, επιλογή δικτύου πρόσβασης, σημείο πρόσβασης, τεχνολογία πρόσβασης, διαδικασία βελτιστοποίησης, προτιμήσεις χρήστη, γραφική διεπαφή χρήστη

## **Abstract**

The next generation of mobile systems is expected to comprise heterogeneous networks consisting of diverse radio segments, able to host multimode wireless terminals, each of them capable of alternatively operating in the diverse radio segments available in the infrastructure. The purpose of this paper is the development of a management architecture for composite radio infrastructures that incorporates an innovative Terminal Management System (TMS), located at the mobile terminal and capable of dynamically and independently selecting the appropriate access network through which services can be obtained efficiently in terms of cost and QoS, in a transparent manner. Its innovation lies on the functionality of the proposed scheme that supports terminal-initiated and terminal-controlled access network selection, thus expanding intelligence from the network to the terminal. In order to implement the above TMS, a Java-based application has been developed, consisting of two basic components. The first component is a graphical user interface (GUI) through which the user will be able to specify, store and dynamically edit his preferences. This GUI will allow the user to give different priorities to parameters that may influence the access network selection process, such as 'quality', 'operator', and 'cost'. The other main component of the TMS is the Intelligent Interface Selection (IIS) function. The latter is responsible for independently determining the optimal local interface and attachment point through which the applications that the user is requesting or is already receiving can be obtained as efficiently as possible. The aforementioned optimization process takes into account link-layer measurements from the access points that reflect the network's status and resource availability, user preferences and service requirements. Two typical scenarios have been simulated in order to demonstrate the access network selection functionality of the IIS algorithm.

## **Key Words**

mobile terminal, access network selection, access point, access technology, optimization process, user preferences, graphical user interface

## Περιεχόμενα

Περιεχόμενα.....	7
1. Εισαγωγή.....	11
1.1 Σημερινά και Μελλοντικά Συστήματα Ασύρματης Επικοινωνίας.....	11
1.1.1 Συστήματα προ της 4 <sup>ης</sup> Γενιάς.....	11
1.1.2 Ιστορική Εξέλιξη.....	11
1.1.3 Συστήματα 4 <sup>ης</sup> Γενιάς.....	12
1.1.4 «Always Best Connected» (ABC).....	13
1.1.5 Η «υπηρεσία» ABC.....	13
1.1.6 Λειτουργικές Ενότητες μιας ABC Λύσης.....	15
1.1.7 Περιγραφή Ασύρματων Τεχνολογιών Πρόσβασης.....	19
1.1.7.1 Εισαγωγή – Εξέλιξη Ασύρματων Συστημάτων Επικοινωνιών.....	19
1.1.7.2 Κυβελωτά Συστήματα 1 <sup>ης</sup> Γενιάς.....	19
1.1.7.3 Μετάβαση στα Ψηφιακά Κυβελωτά Συστήματα.....	20
1.1.7.4 Κυβελωτά Συστήματα 2 <sup>ης</sup> Γενιάς.....	22
1.1.7.5 Κυβελωτά Συστήματα 3 <sup>ης</sup> Γενιάς.....	23
1.1.7.6 Ασύρματα Τοπικά Δίκτυα Επικοινωνιών (WLANs).....	25
1.1.7.7 Ασύρματα Προσωπικά Δίκτυα Επικοινωνιών (WPANs).....	26
1.1.7.8 Δίκτυα Τηλεοπτικών Μεταδόσεων.....	26
1.2 Δικτυακές Υποδομές και Λειτουργίες.....	28
1.2.1 Εισαγωγή.....	28
1.2.2 Τα Επιχειρηματικά Μοντέλα.....	28
1.2.3 Δίκτυα Επόμενης Γενιάς.....	31
1.2.4 Εισαγωγή στο Μοντέλο Αρχιτεκτονικής για Υπηρεσίες και Δίκτυα 4 <sup>ης</sup> Γενιάς.....	31
1.2.5 Υπηρεσίες Τελικού Χρήστη – Ταυτότητα Τελικού Χρήστη.....	32
1.2.6 Μονάδα Συγκέντρωσης Υπηρεσιών.....	33
1.2.7 Η Ενοποίηση των Δικτύων.....	34
1.2.8 Ανάγκη για Συνεργασία.....	35
1.2.9 Σύνοψη.....	35
1.3 Τερματικές Υποδομές και Λειτουργίες.....	36
1.3.1 Εισαγωγή – Δυνατότητες Σύγχρονων Τερματικών.....	36
1.3.2 Η Επιλογή Δικτύου Πρόσβασης ως Ρόλος του Τερματικού.....	37
1.4 Αντικείμενο της Διπλωτικής.....	39
1.4.1 Εισαγωγή.....	39
1.4.2 Ευφυής Επιλογή Δικτύου Πρόσβασης.....	40
1.4.3 Δυναμική Προσαρμογή Προφίλ Χρήστη.....	41
1.4.4 Διάρθρωση της Εργασίας.....	41
2. Αρχιτεκτονική Διαχείρισης Τερματικού.....	43
2.1 Επισκόπηση Αρχιτεκτονικής.....	43
2.1.1 Η Συνεργασία Δικτύου και Τερματικών.....	43
2.1.2 Βασικές Συνιστώσες της Αρχιτεκτονικής Διαχείρισης Τερματικού.....	43
2.1.3 Η Υλοποίηση του Συστήματος Διαχείρισης Τερματικού.....	45
2.2 Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου.....	45
2.2.1 Ρόλος της Συνιστώσας.....	45
2.2.2 Περιγραφή της Συνιστώσας – Πληροφορίες των Μετρήσεων.....	46
2.2.3 Πρωτόκολλα Ανακάλυψης Δυνατοτήτων Γειτονικών Access Routers.....	47
2.3 Συνιστώσα Προτιμήσεων Χρήστη.....	51
2.4 Συνιστώσα Διαχείρισης Κινητικότητας.....	53

2.5	Ροή Πληροφορίας .....	55
3.	Συνιστώσα Προτιμήσεων Χρήστη .....	57
3.1	Παρουσίαση της Γραφικής Διεπαφής .....	57
3.1.1	Εισαγωγή .....	57
3.1.2	Παρουσίαση Διαθέσιμων Υπηρεσιών.....	57
3.1.3	Κεντρικό Μενού .....	65
3.2	Διαμόρφωση και Αποθήκευση Προφίλ Χρήστη.....	68
4.	Ευφυής Επιλογή Δικτύου Πρόσβασης.....	75
4.1	Περιγραφή του Προβλήματος Ευφυούς Επιλογής Δικτύου Πρόσβασης	75
4.1.1	Σκοπός του IIS module .....	75
4.1.2	Εισαγωγή στο Πρόβλημα .....	75
4.1.3	Γενική Περιγραφή του Προβλήματος και Σχηματική Αναπαράσταση... 75	75
4.1.4	Είδη Διαπομπής.....	77
4.1.5	Στοιχεία Εισόδου.....	78
4.1.5.1	Πρώτο στοιχείο εισόδου – Το προφίλ χρήστη.....	78
4.1.5.2	Δεύτερο στοιχείο εισόδου – Οι συνθήκες δικτύου.....	82
4.1.5.3	Τρίτο στοιχείο εισόδου – Η υπηρεσία .....	83
4.1.6	Τράπεζα Πληροφοριών.....	83
4.1.7	Στοιχεία Εξόδου .....	92
4.1.8	Περιορισμοί.....	93
4.1.9	Περιπτώσεις Εκκίνησης της Διαδικασίας Βελτιστοποίησης .....	93
4.2	Λύση Προβλήματος .....	99
4.2.1	Εισαγωγή .....	99
4.2.2	Μαθηματική Διατύπωση .....	100
4.2.3	Βήματα Επίλυσης του Προβλήματος.....	101
4.2.4	Ολοκληρωμένη Μορφή της Τράπεζας Πληροφοριών .....	113
5.	Ειδικά Θέματα Υλοποίησης.....	115
5.1	Εισαγωγή .....	115
5.2	Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου.....	116
5.2.1	Γενική Περιγραφή.....	116
5.2.2	Αρχείο Μετρήσεων .....	117
5.2.3	Πίνακες Μετρήσεων .....	119
5.2.4	Μέθοδος <i>measure()</i> .....	120
5.2.5	Μέθοδοι Ελέγχου Εμφάνισης ή Εξαφάνισης Σημείων Πρόσβασης ....	120
5.2.6	Συμπληρωματικά Στοιχεία .....	120
5.3	Συνιστώσα Διαχείρισης Κινητικότητας.....	120
5.3.1	Γενική Περιγραφή.....	120
5.3.2	Μέθοδος <i>executeIIS()</i> .....	121
5.3.3	Λίστα Τρεχουσών Υπηρεσιών.....	122
5.4	Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης.....	122
5.5	Γραφική Διεπαφή .....	123
5.5.1	Γενική Περιγραφή – Ο Ρόλος της Κλάσης <i>MenuFrame</i> .....	123
5.5.2	Συμπληρωματικά Στοιχεία .....	123
5.6	Συμπληρωματικά Σχήματα .....	126
5.7	Υλοποίηση Υπηρεσιών .....	128
5.8	Οντότητα Διαχείρισης του Δικτύου .....	128
6	Εκτέλεση Προσομοιώσεων.....	129
6.1	Εισαγωγή .....	129
6.2	Σενάριο 1 – «Μέσα σε ένα πανεπιστημιακό campus».....	130



6.2.1	Γενική Περιγραφή.....	130
6.2.2	Εξέλιξη Σεναρίου .....	130
<b>6.3</b>	<b>Σενάριο 2 – «Από το σπίτι στη δουλειά».....</b>	<b>176</b>
6.3.1	Γενική Περιγραφή.....	176
6.3.2	Εξέλιξη Σεναρίου .....	176
<b>6.4</b>	<b>Σύνοψη .....</b>	<b>214</b>
<b>7.</b>	<b>Συμπεράσματα και Επεκτάσεις .....</b>	<b>215</b>
<b>7.1</b>	<b>Ανακεφαλαίωση.....</b>	<b>215</b>
<b>7.2</b>	<b>Συμπεράσματα Λειτουργίας του Αλγορίθμου.....</b>	<b>217</b>
<b>7.3</b>	<b>Επεκτάσεις της Εργασίας.....</b>	<b>220</b>
7.3.1	Εισαγωγή .....	220
7.3.2	Ενσωμάτωση της Τεχνολογίας Πρόσβασης ως Παράγοντα του Προφίλ Χρήστη	220
7.3.3	Τροποποίηση της Μεθόδου Κανονικοποίησης του Κόστους .....	221
7.3.4	Το Σύνολο των Τρεχουσών Υπηρεσιών ως Είσοδος του Αλγορίθμου .....	222
7.3.5	Αυτόματος Καθορισμός Συντελεστών Βαρύτητας.....	223
<b>Βιβλιογραφία.....</b>		<b>225</b>
<b>ΠΑΡΑΡΤΗΜΑ .....</b>		<b>227</b>



# 1. Εισαγωγή

## 1.1 Σημερινά και Μελλοντικά Συστήματα Ασύρματης Επικοινωνίας

### 1.1.1 Συστήματα προ της 4<sup>ης</sup> Γενιάς

Στον τομέα των Τηλεπικοινωνιών, έχει αναπτυχθεί, τα τελευταία χρόνια, ένα πλήθος νέων τεχνολογιών πρόσβασης, τόσο ενσύρματων όσο και ασύρματων. Καθώς τα κυψελωτά συστήματα 2<sup>ης</sup> Γενιάς εξελίσσονται σε συστήματα 3<sup>ης</sup> Γενιάς, επεκτείνοντας διαρκώς την κάλυψή τους, παρατηρείται ταυτόχρονα μια ευρεία διάδοση των ασύρματων τοπικών δικτύων, ώστε να καλυφθεί η ανάγκη παροχής ευρυζωνικής πρόσβασης στο Internet, σε σημεία με μεγάλη κίνηση, όπως αεροδρόμια, ξενοδοχεία, συνεδριακά κέντρα και πανεπιστήμια. Την ίδια στιγμή, συστήματα σταθερής πρόσβασης, διασυνδεδεμένα με ασύρματα τοπικά δίκτυα, κάνουν την εμφάνισή τους, τόσο σε οικιακά όσο και σε εταιρικά περιβάλλοντα. Αξιοσημείωτο είναι ότι όλες αυτές οι νέες τεχνολογίες, και οι αντίστοιχες υποδομές, που ραγδαίως αναπτύσσονται, είναι ετερογενείς μεταξύ τους.

### 1.1.2 Ιστορική Εξέλιξη

Κατά τη διάρκεια της δεκαετίας του 1980, παράλληλα με την ευρεία διάδοση των κυψελωτών συστημάτων 2<sup>ης</sup> Γενιάς (2G), όπως το GSM (Global System for Mobile Communications), αναπτύχθηκε και η ιδέα «always connected» («διαρκούς σύνδεσης»). Σύμφωνα με αυτήν και χάρη στις συμφωνίες περιαγωγής μεταξύ των διαφόρων παρόχων, ένας GSM χρήστης μπορεί να μένει πάντοτε συνδεδεμένος, σχεδόν σε κάθε γωνιά του πλανήτη, και παράλληλα να είναι προσβάσιμος μέσω του ίδιου τηλεφωνικού αριθμού, λαμβάνοντας ένα και μόνο λογαριασμό.

Τα τελευταία χρόνια, έχει αναπτυχθεί η υπηρεσία GPRS (General Packet Radio Service), προκειμένου να παρέχει IP (Internet Protocol) συνδεσιμότητα στους χρήστες GSM. Αυτό σημαίνει, δηλαδή, ότι οι χρήστες μπορούν, συνδεδεμένοι σε ένα σταθερό δίκτυο IP (IP – based fixed network), να έχουν πρόσβαση στο Internet. Παράλληλα, τα κυψελωτά συστήματα 3<sup>ης</sup> Γενιάς (3G), όπως το UMTS (Universal Mobile Telecommunications System) και το cdma2000 (code-division multiple access), έχουν διαμορφωθεί κατάλληλα ώστε να παρέχουν πρόσβαση στο Internet, μέσω του κινητού τερματικού. Συγχρόνως με την εξέλιξη των κυψελωτών συστημάτων, ένα πλήθος άλλων τεχνολογιών πρόσβασης έχει εμφανιστεί και έχει γίνει μέρος της καθημερινής μας ζωής. Ασύρματα τοπικά δίκτυα (Wireless Local Area Networks – WLANs), όπως το IEEE 802.11b, έχουν εγκατασταθεί σε γραφεία, αεροδρόμια, ξενοδοχεία, συνεδριακά κέντρα και πανεπιστήμια, παρέχοντας ταχύτατη πρόσβαση στο Internet. Η τεχνολογία WLAN χρησιμοποιείται, επίσης, ευρέως σε οικιακό περιβάλλον, προκειμένου να διαμοιραστεί, με ασύρματο τρόπο, μια σύνδεση στο Internet, μέσω DSL γραμμής (Digital Subscriber Line) ή καλωδιακού μόντεμ (cable modem), σε περισσότερους από έναν προσωπικούς υπολογιστές.

Μία ακόμη τεχνολογία που αναμένεται να γνωρίσει μεγάλη εξάπλωση είναι η τεχνολογία DVB (Digital Video Broadcasting). Χαρακτηρίζεται από μεγάλες τυπικές ταχύτητες μετάδοσης και μπορεί να καλύψει γεωγραφικές περιοχές μεγάλου εύρους. Μάλιστα, οι υπηρεσίες που μπορούν να προσφερθούν μέσω της τεχνολογίας αυτής δεν περιορίζονται μόνο στη μετάδοση τηλεόρασης και βίντεο.

Αναλυτικότερη περιγραφή των κυριότερων ασύρματων τεχνολογιών πρόσβασης θα δοθεί σε επόμενη παράγραφο (1.1.7).

### *1.1.3 Συστήματα 4<sup>ης</sup> Γενιάς*

Η συνύπαρξη τόσων ετερογενών τεχνολογιών πρόσβασης καθιστά εμφανή την ανάγκη ανάπτυξης μιας πλατφόρμας η οποία θα ενοποιεί, διαφανώς, τις παραπάνω τεχνολογίες. Η πλατφόρμα αυτή θα παρέχει τη βασική λειτουργικότητα διαχείρισης δικτύου, καθώς επίσης και μία αφαιρετική εικόνα των υποκείμενων δικτύων. Με αυτόν τον τρόπο, τόσο στην πλευρά των υπηρεσιών όσο και στην πλευρά του χρήστη, θα δημιουργείται η ψευδαίσθηση ενός ενιαίου εικονικού δικτύου, που θα λειτουργεί ανεξάρτητα από τον τύπο των υποστηριζόμενων τεχνολογιών.

Εδώ ακριβώς έγκειται και ο στόχος των αποκαλούμενων συστημάτων «4<sup>ης</sup> Γενιάς» ή, εν συντομία, συστημάτων 4G ή B3G (Beyond 3G). Στόχος των συστημάτων 4G, λοιπόν, είναι όχι μόνο η χρήση όλων των παραπάνω τεχνολογιών για την παροχή υψηλότερων ρυθμών δεδομένων, αλλά κυρίως η διαφανής ενοποίησή τους. Ένας επιπλέον στόχος των συστημάτων αυτών είναι και η αξιοποίηση ενός συνόλου πληροφοριών που είναι γνωστό ως «context information» και διευκολύνει την ανάπτυξη νέων υπηρεσιών.

Στα συστήματα 4G, οι χρήστες θα έχουν τη δυνατότητα να μεταβαίνουν ανάμεσα σε διαφορετικά δίκτυα πρόσβασης, που θα μπορούν να είναι ενσύρματα ή ασύρματα, δημόσια ή ιδιωτικά. Προσδοκείται ότι οι υπηρεσίες θα παρέχονται ανεξάρτητα από τον τύπο του χρησιμοποιούμενου δικτύου και θα βασίζονται σε «συνοδευτικές» πληροφορίες (context information) που σχετίζονται με την επικοινωνία του χρήστη. Ως τέτοιου είδους πληροφορίες μπορούν να θεωρηθούν η γεωγραφική θέση του χρήστη, οι προτιμήσεις του όσον αφορά τον τρόπο σύνδεσης, καθώς και άλλα χαρακτηριστικά της επικοινωνίας, όπως το διαθέσιμο εύρος ζώνης (bandwidth) και τα χαρακτηριστικά του τερματικού. Γενικά, πρόκειται για πληροφορίες που μπορούν να αντληθούν από μια σύνοδο επικοινωνίας, χωρίς ωστόσο να σχετίζονται ή να αντλούνται από το πραγματικό περιεχόμενο της επικοινωνίας.

Η παρούσα εργασία ασχολείται με τα συστήματα 4G, παρουσιάζοντας τους στόχους και τα πλεονεκτήματά τους. Ιδιαίτερη βαρύτητα αποδίδεται στη θεώρηση των συστημάτων αυτών από την πλευρά του τερματικού, περιγράφοντας εις βάθος το ρόλο και την αρχιτεκτονική του συστήματος διαχείρισης των 4G τερματικών και προχωρώντας στην υλοποίηση του συστήματος αυτού.

#### 1.1.4 «Always Best Connected» (ABC)

Σε ένα περιβάλλον πολλαπλών τεχνολογιών πρόσβασης, η ιδέα «always connected», η οποία αναφέρθηκε στην παράγραφο 1.1.2, εξελίσσεται στην «always best connected» («διαρκώς βέλτιστη σύνδεση») ή, εν συντομία, «ABC». Σύμφωνα με την ABC απαίτηση, ο χρήστης δεν αρκεί απλά να είναι διαρκώς συνδεδεμένος (always connected), αλλά πρέπει να είναι συνδεδεμένος και με τον καλύτερο δυνατό τρόπο, συνδυάζοντας, λόγου χάρη, την παγκόσμια κάλυψη των κυψελωτών συστημάτων με το μεγάλο εύρος ζώνης των ασύρματων τοπικών δικτύων.

Ο καθορισμός, όμως, του «καλύτερου δυνατού τρόπου» εξαρτάται από πολλές διαφορετικές παραμέτρους, όπως είναι οι προτιμήσεις του χρήστη, οι δυνατότητες του τερματικού του, οι απαιτήσεις των εφαρμογών, η ασφάλεια, οι πολιτικές των παρόχων, οι διαθέσιμοι πόροι και η κάλυψη του δικτύου. Ανάλογα με το είδος των εφαρμογών που τρέχει και τις προτιμήσεις του, ένας χρήστης μπορεί να είναι συνδεδεμένος μέσω μιας μόνο τεχνολογίας πρόσβασης κάθε στιγμή ή και μέσω περισσότερων τεχνολογιών συγχρόνως. Η ιδέα του ABC περιλαμβάνει όλους τους τύπους τεχνολογιών, σταθερών και ασυρμάτων, είτε εφαρμόζονται ήδη είτε βρίσκονται σε στάδιο ανάπτυξης.

#### 1.1.5 Η «υπηρεσία» ABC

Στην πράξη, το ABC μπορεί να αποκτηθεί από το χρήστη σαν υπηρεσία. Ορίζεται, δηλαδή, η υπηρεσία ABC ως η υπηρεσία που εξασφαλίζει στο χρήστη, ανά πάσα στιγμή και σε κάθε σημείο, IP συνδεσιμότητα με έναν κόμβο στο Internet, μέσω του δικτύου ή των δικτύων πρόσβασης που καλύπτουν καλύτερα τις ανάγκες του, τη δεδομένη στιγμή. Αυτές οι ανάγκες μπορούν να καταγραφούν, για παράδειγμα, από τις εφαρμογές που τρέχει ο χρήστης ή μέσα στη συμφωνία που έχει συναφθεί μεταξύ του χρήστη και του παρόχου της υπηρεσίας ABC. Προκειμένου να παραχωρηθεί στο χρήστη συνδεσιμότητα πάνω από διαφορετικά δίκτυα πρόσβασης, η υπηρεσία ABC είναι βασισμένη σε μια υποδομή που αναλαμβάνει την ταυτοποίηση, τον έλεγχο πρόσβασης και τη χρέωση (υποδομή AAA – authentication, authorization, accounting).

Μία από τις σημαντικότερες πτυχές του ABC είναι η «εμπειρία» του χρήστη (user experience), τι πιστεύει, δηλαδή, ότι ο χρήστης κερδίζει από την υπηρεσία αυτή και με ποιον τρόπο πρέπει να του παρέχεται. Οι παράγοντες που διαμορφώνουν αυτήν την «εμπειρία» είναι, επιγραμματικά, οι εξής:

- Ροή της Πληροφορίας (Information Delivery)
- Υποστήριξη Κινητικότητας (Mobility Support)
- Αλληλεπίδραση με το Χρήστη (User Interaction)
- Συνδρομή στην ABC Υπηρεσία (Subscription)

Παρακάτω θα αναφερθούμε, εν συντομία, σε καθέναν από αυτούς:

- Ροή της Πληροφορίας (Information Delivery):

Ένα σημαντικό μέρος της εμπειρίας του χρήστη σχετίζεται με τη συμπεριφορά των εφαρμογών. Σε ένα περιβάλλον που οι συνθήκες αλλάζουν συνεχώς, οι εφαρμογές πρέπει να προσαρμόζονται στις αλλαγές αυτές, έτσι ώστε η ροή της πληροφορίας, όπως αυτή βιώνεται από το χρήστη, να είναι διαφανής ως προς την αλλαγή του δικτύου πρόσβασης και, γενικά, ως προς την αλλαγή του περιβάλλοντος επικοινωνίας. Και αυτό, γιατί ο χρήστης δεν ενδιαφέρεται μέσω ποιας τεχνολογίας ή μέσω ποιας υποδομής του παρέχεται μια υπηρεσία. Αυτό που ενδιαφέρει το χρήστη είναι να λαμβάνει, οποιαδήποτε στιγμή, την επιθυμητή υπηρεσία με τον καλύτερο δυνατό τρόπο.

Οι τρόποι για να εξασφαλιστεί αυτή η διαφανής ροή της πληροφορίας είναι δύο: Πρώτον, οι εφαρμογές είναι δυνατόν, είτε από μόνες τους είτε λόγω των ρυθμίσεων του χρήστη, να απαιτήσουν ένα συγκεκριμένο επίπεδο ποιότητας (QoS level), προκειμένου να παρέχουν βέλτιστη απόδοση. Με αυτόν τον τρόπο, οι εφαρμογές και οι ρυθμίσεις του χρήστη επηρεάζουν τη διαδικασία επιλογής του τρόπου σύνδεσης. Δεύτερον, οι εφαρμογές είναι δυνατό να προσαρμοστούν στις δυνατότητες της συσκευής και στα ιδιαίτερα χαρακτηριστικά των τεχνολογιών, βελτιστοποιώντας την παρουσίαση και την παράδοση της πληροφορίας

- Υποστήριξη Κινητικότητας (Mobility Support):

Στην απλούστερή της μορφή, η υπηρεσία ABC παρέχει στο χρήστη πρόσβαση σε υπηρεσίες, μέσω διαφόρων τεχνολογιών πρόσβασης, χωρίς, όμως, υποστήριξη κινητικότητας, χωρίς, δηλαδή, να μπορεί ο χρήστης να μεταβεί σε άλλη τεχνολογία πρόσβασης, κατά τη διάρκεια λήψης μιας υπηρεσίας. Ωστόσο, σύμφωνα με μια βελτιωμένη προσέγγιση, η ABC υπηρεσία θα πρέπει να επιτρέπει στο τερματικό να μεταβαίνει, διαφανώς, μεταξύ των διαφόρων τεχνολογιών πρόσβασης, ενώ παράλληλα διατηρεί τις συνδέσεις που έχει ανοίξει για τη λήψη των υπηρεσιών. Πέραν τούτου, η ABC υπηρεσία μπορεί να συμπεριλάβει τη δυνατότητα (χρήσιμη για ορισμένες υπηρεσίες) αναγνώρισης του δικτύου στο οποίο είναι συνδεδεμένος ο χρήστης.

- Αλληλεπίδραση με το Χρήστη (User Interaction):

Με την εγγραφή του στην ABC υπηρεσία, ο χρήστης δημιουργεί ένα προσωπικό «προφίλ», μέσω του οποίου καθορίζει τις προτιμήσεις του, όσον αφορά το κόστος και την ποιότητα των υπηρεσιών. Προφανώς, πρέπει να του δίδεται η δυνατότητα αλλαγής αυτού του «προφίλ», οποτεδήποτε αλλάζουν οι προτιμήσεις του. Μεγάλο βάρος πρέπει να δοθεί στο να είναι εύχρηστος ο τρόπος επεξεργασίας του προφίλ. Και αυτό, γιατί ο μέσος χρήστης σπάνια είναι διατεθειμένος να αφιερώσει πολύ χρόνο για να κατανοήσει και να πειραματιστεί με τις ρυθμίσεις μιας εφαρμογής, ακόμα και αν αυτό είναι προς όφελός του. Με το να διατηρεί, όμως, τις προεπιλεγμένες ρυθμίσεις (default settings), στερείται, ουσιαστικά, τη δυνατότητα καθορισμού του τρόπου σύνδεσης ανάλογα με τις ανάγκες του. Επίσης, πέραν της δυνατότητας καθορισμού των προτιμήσεών του, ο χρήστης πρέπει να έχει και μια εποπτεία των συνδέσεων που βρίσκονται σε εξέλιξη, ανά πάσα στιγμή. Έτσι, θα

αισθάνεται περισσότερο «ασφαλής» ότι το τερματικό βρίσκεται υπό τον έλεγχο του και ότι πληρώνει για ό,τι ακριβώς έχει ζητήσει.

- Συνδρομή στην ABC Υπηρεσία (Subscription):

Αφού, όπως είδαμε, η ABC δυνατότητα προσφέρεται στο χρήστη ως μια υπηρεσία, θα πρέπει ο χρήστης να είναι συνδρομητής σε αυτήν, προκειμένου να μπορεί να την αξιοποιήσει. Το θέμα αυτό θα παρουσιαστεί αναλυτικότερα σε επόμενη παράγραφο (1.2).

### 1.1.6 Λειτουργικές Ενότητες μιας ABC Λύσης

Σε μια ολοκληρωμένη ABC λύση συναντούμε τις εξής λειτουργικές ενότητες:

- Ανίχνευση Δικτύων Πρόσβασης (Access Discovery)
- Επιλογή Δικτύων Πρόσβασης (Access Selection)
- AAA Υποστήριξη (Authentication, Authorization, Accounting Support)
- Διαχείριση Κινητικότητας (Mobility Management)
- Διαχείριση Προφίλ (Profile Handling)
- Προσαρμογή Υπηρεσιών (Content Adaptation)

Κάποιες από τις παραπάνω ενότητες πρέπει να ενσωματωθούν στο ABC τερματικό, ενώ οι υπόλοιπες πρέπει να υλοποιηθούν στην πλευρά του δικτύου. Παρακάτω περιγράφεται λεπτομερώς κάθε λειτουργική ενότητα και παρατίθενται σημαντικά ζητήματα προς διερεύνηση, προκειμένου να προσφερθεί μια πλήρης ABC λύση.

- Ανίχνευση Δικτύων Πρόσβασης (Access Discovery):

Όταν ένα ABC τερματικό ενεργοποιείται, εκτελεί μια διαδικασία ανίχνευσης δικτύων, έτσι ώστε να εντοπίσει τα διαθέσιμα σημεία πρόσβασης. Αφού επιλεγεί ένα σημείο πρόσβασης, το ABC τερματικό χρειάζεται να πραγματοποιεί έναν περιοδικό έλεγχο, προκειμένου να είναι σε θέση να διαπιστώσει τυχόν «εμφάνιση» ή «εξαφάνιση» σημείων πρόσβασης.

Σημεία κλειδιά που πρέπει να ληφθούν υπόψη, κατά τη διαδικασία ανίχνευσης, είναι τα ακόλουθα:

- Ο καθορισμός ενός συνόλου παραμέτρων που να περιγράφει, επακριβώς, τα σημεία πρόσβασης. Το σύνολο αυτό θα πρέπει να περιλαμβάνει τον τύπο της τεχνολογίας πρόσβασης, τον πάροχο δικτύου, τα επίπεδα ποιότητας υπηρεσίας, το τρέχον κόστος για ένα χρήστη που επιθυμεί να συνδεθεί πάνω από το συγκεκριμένο δίκτυο, κ.λπ.
- Η ακριβής μορφή με την οποία θα συλλέγονται, από το κινητό τερματικό, περιγραφικά χαρακτηριστικά των δικτύων, όπως, για παράδειγμα, υπό τη μορφή μετρήσεων, πληροφοριών από τους παρόχους, κ.λπ.

▪ Επιλογή Δικτύων Πρόσβασης (Access Selection):

Η λειτουργία αυτή αναφέρεται στη διαδικασία με την οποία λαμβάνεται η απόφαση για το ποιο θα είναι το δίκτυο πρόσβασης, και ειδικότερα το σημείο πρόσβασης, με το οποίο θα συνδέεται το τερματικό κάθε στιγμή. Μπορούμε να διαχωρίσουμε τρία σημαντικά στοιχεία στην παραπάνω διαδικασία:

- Επιλογή προερχόμενη από το τερματικό
- Επιλογή προερχόμενη από το δίκτυο
- Συμμετοχή του χρήστη

Κατά τη διαδικασία επιλογής δικτύου πρόσβασης, ένα πλήθος διαφόρων παραμέτρων πρέπει να ληφθούν υπόψη. Μία από αυτές είναι το ABC προφίλ του χρήστη, το οποίο αντικατοπτρίζει τις προσωπικές επιλογές του χρήστη, όσον αφορά τον τρόπο σύνδεσης. Μια δεύτερη παράμετρος είναι τα χαρακτηριστικά του δικτύου (π.χ. διαθέσιμο εύρος ζώνης, κόστος και πάροχος). Άλλες παράμετροι, τέλος, είναι οι δυνατότητες του τερματικού και οι απαιτήσεις των υπηρεσιών.

Όταν το τερματικό τίθεται σε λειτουργία, η αρχική επιλογή δικτύου πρόσβασης πρέπει να προέρχεται από το τερματικό, αφού δεν υπάρχει καμία ενεργή σύνδεση, ώστε να μπορεί η επιλογή να προέλθει από το δίκτυο. Για το σκοπό αυτό, το τερματικό χρειάζεται να έχει αποθηκευμένο ένα «προφίλ» (όπως αυτό που περιγράψαμε παραπάνω) ή έστω μια προεπιλεγμένη ρύθμιση επιλογής δικτύου πρόσβασης. Το ίδιο ισχύει, επίσης, και για την περίπτωση απώλειας της τρέχουσας σύνδεσης. Προκειμένου να αποκατασταθεί η σύνδεση, το τερματικό πρέπει να επιλέξει μόνο του ένα νέο τρόπο πρόσβασης, χωρίς παρέμβαση του δικτύου.

Ένα πλεονέκτημα της προερχόμενης από το δίκτυο επιλογής δικτύου πρόσβασης είναι ότι, με αυτόν τον τρόπο, ο πάροχος της υπηρεσίας ABC μπορεί να προσφέρει πληροφορίες σχετικά με τα ιδιαίτερα χαρακτηριστικά του δικτύου, τις οποίες το τερματικό δύσκολα μπορεί να αποκτήσει από μόνο του. Επιπλέον, αναθέτοντας τη διαδικασία επιλογής στο δίκτυο, μπορεί να γίνει εξισορρόπηση του φορτίου του δικτύου και μείωση της κίνησης στους ραδιοδιαύλους, εάν η διαδικασία επιλογής προϋποθέτει επανειλημμένες αναζητήσεις στις βάσεις δεδομένων του δικτύου. Τέλος, η προερχόμενη από το δίκτυο επιλογή τρόπου πρόσβασης καθιστά δυνατή τη μεγιστοποίηση της συνολικής απόδοσης του συστήματος του δικτύου.

Σημεία κλειδιά που πρέπει να διερευνηθούν περισσότερο, σχετικά με την ABC επιλογή δικτύου πρόσβασης, είναι:

- Πώς μπορεί να γίνει η εφαρμογή της από το δίκτυο προερχόμενης επιλογής πρόσβασης, η οποία επιτυχώς χρησιμοποιείται στα συστήματα 2G, 2.5G και 3G, ώστε να επιτραπεί στους παρόχους δικτύου να διαχειριστούν το φορτίο των δικτύων τους.
- Πώς ο χρήστης θα ενημερώνεται με έναν κατανοητό τρόπο για τις τρέχουσες ABC επιλογές του και θα καθορίζει τις προτιμήσεις του.



- AAA Υποστήριξη (Authentication, Authorization, Accounting Support):

Ένα σημαντικό συστατικό της ιδέας ABC είναι η AAA υποστήριξη. Σκοπός της είναι η επαλήθευση της ταυτότητας του χρήστη (authentication), ο έλεγχος δικαιώματος πρόσβασης του χρήστη στην αιτούμενη υπηρεσία (authorization), καθώς και η συλλογή και επεξεργασία δεδομένων που αφορούν τη χρέωση του χρήστη για τις παρεχόμενες υπηρεσίες (accounting). Η παροχή AAA υποστήριξης προϋποθέτει την ύπαρξη συμφωνιών μεταξύ των εμπλεκόμενων μονάδων (χρήστης, πάροχοι δικτύου, πάροχοι υπηρεσιών, κ.λπ.). Για την επίτευξη λειτουργιών AAA, είναι απαραίτητη η υιοθέτηση κάποιου κοινού AAA πρωτοκόλλου, που θα καθορίζει τους κανόνες βάσει των οποίων θα γίνεται η ανταλλαγή των μηνυμάτων μεταξύ των εμπλεκόμενων μονάδων. Ένα τέτοιο πρωτόκολλο είναι το «Diameter», το οποίο βρίσκεται σε φάση προτυποποίησης από την IETF (Internet Engineering Task Force). Σημείο κλειδί για την ορθή λειτουργία της AAA υποστήριξης είναι η παροχή εγγυήσεων ασφαλείας.

- Διαχείριση Κινητικότητας (Mobility Management):

Όσον αφορά τη διαχείριση κινητικότητας, δύο είναι τα ζητήματα στα οποία πρέπει να δοθεί ιδιαίτερη βαρύτητα:

- Συνέχιση Συνόδου (Session Continuity): Ο όρος αυτός αφορά τη συνέχιση λήψης μιας υπηρεσίας όταν το ABC τερματικό μεταβαίνει σε άλλο δίκτυο ή σε άλλη τεχνολογία πρόσβασης. Μια πρόταση επίλυσης του παραπάνω ζητήματος είναι το «Mobile IP». Πρόκειται για μια λύση, σε επίπεδο IP, η οποία στοχεύει στο να κάνει διαφανείς στα ανώτερα στρώματα τις μετακινήσεις που λαμβάνουν χώρα στο στρώμα IP.
- Πρόσβαση στο Χρήστη (Reachability): Ο όρος αυτός αναφέρεται στη δυνατότητα πρόσβασης στο χρήστη ανεξάρτητα από το δίκτυο πρόσβασης στο οποίο βρίσκεται. Μια λύση σε αυτό το πρόβλημα παρέχεται από το πρωτόκολλο «SIP» (Session Initiation Protocol).

Σημείο κλειδί είναι η παρουσίαση μιας συνδυασμένης λύσης διαχείρισης επιπέδων ποιότητας και διαχείρισης κινητικότητας.

- Διαχείριση Προφίλ (Profile Handling):

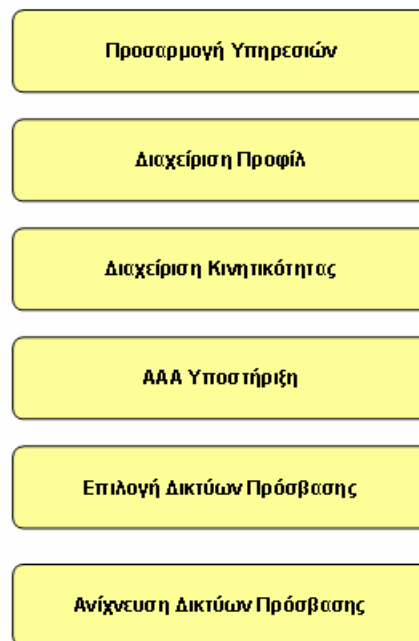
Όταν ένας χρήστης γίνεται συνδρομητής στην ABC υπηρεσία, δημιουργείται ένα προφίλ χρήστη το οποίο περιλαμβάνει τις προσωπικές του προτιμήσεις, αναφορικά με την επιλογή του τρόπου πρόσβασης και της ποιότητας των παρεχόμενων υπηρεσιών. Το προφίλ αυτό αποθηκεύεται από τον πάροχο της υπηρεσίας ABC και μπορεί να τροποποιηθεί, είτε απευθείας από το χρήστη είτε μέσω του παρόχου. Ο πάροχος, επίσης, διατηρεί ένα προφίλ με πληροφορίες που αφορούν το χρήστη, όπως, για παράδειγμα, τα δικαιώματα της συνδρομής του (authorization), τα στοιχεία ταυτοποίησης του (authentication) και λογιστικά δεδομένα (accounting).

▪ Προσαρμογή Υπηρεσιών (Content Adaptation):

Όπως αναφέρθηκε παραπάνω, είναι απαραίτητο οι υπηρεσίες να προσαρμόζονται στις τρέχουσες συνθήκες του δικτύου. Υπάρχουν πολλοί εναλλακτικοί τρόποι προσέγγισης αυτής της προσαρμογής. Ένας τρόπος επίλυσης είναι η υπηρεσία να ανιχνεύει αλλαγές στα χαρακτηριστικά του δικτύου και να προσαρμόζει το περιεχόμενο της κατάλληλα. Μια άλλη προσέγγιση είναι το τερματικό να παρέχει τις πληροφορίες για την κατάσταση του δικτύου στην υπηρεσία. Μια τρίτη λύση είναι το δίκτυο πρόσβασης να αναλάβει να παρέχει πληροφορίες τόσο στους παρόχους των υπηρεσιών όσο και στο κινητό τερματικό.

Σημαντικό ζήτημα είναι η παροχή υποστήριξης για δυναμικό καθορισμό ή επανακαθορισμό κάθε συνόδου επικοινωνίας, με βάση τις εξειδικευμένες ανάγκες της συνόδου και σε συνάρτηση με ένα πλήθος εξωτερικών παραγόντων, όπως είναι η κινητικότητα, τα επίπεδα ποιότητας της υπηρεσίας και τα ιδιαίτερα χαρακτηριστικά του δικτύου πρόσβασης.

Στο σχήμα που ακολουθεί παρουσιάζονται συνοπτικά οι λειτουργικές ενότητες που συγκροτούν την υπηρεσία ABC.



Οι λειτουργικές αυτές ενότητες, καθώς και τα σχετικά τεχνικά ζητήματα, θα πρέπει να κατανεμηθούν μεταξύ των εμπλεκόμενων μονάδων. Αυτό είναι ένα σημαντικό θέμα, που πρέπει να ληφθεί υπόψη κατά την ανάπτυξη λύσεων για κάθε λειτουργική ενότητα. Για παράδειγμα, μια απλή ανίχνευση δικτύων πρόσβασης μπορεί να εκτελεστεί αποκλειστικά από το τερματικό, ενώ μια πιο σύνθετη λύση απαιτεί πληροφορίες από το δίκτυο πρόσβασης και, ενδεχομένως, από τον πάροχο της ABC υπηρεσίας. Ο τρόπος με τον οποίο γίνεται η κατανομή επηρεάζει, επίσης, και τις σχέσεις μεταξύ των ABC μονάδων. Χαρακτηριστικό παράδειγμα αποτελεί η λειτουργία της AAA υποστήριξης, όπου δεν αρκεί μόνο η ανάπτυξη της αντίστοιχης τεχνολογίας, αλλά απαιτείται, επιπλέον, η σύναψη εταιρικών συμφωνιών ανάμεσα στις ABC μονάδες.

### *1.1.7 Περιγραφή Ασύρματων Τεχνολογιών Πρόσβασης*

#### *1.1.7.1 Εισαγωγή – Εξέλιξη Ασύρματων Συστημάτων Επικοινωνιών*

Κλείνοντας το τμήμα 1.1, θα κάνουμε μια ιστορική αναδρομή στα κυριότερα ασύρματα συστήματα επικοινωνιών που υιοθετήθηκαν κατά καιρούς. Παράλληλα, θα δοθούν μερικά βασικά τεχνικά χαρακτηριστικά των σπουδαιότερων τεχνολογιών ασύρματης πρόσβασης.

Αρχικά, τα πρώτα συστήματα επικοινωνιών χρησιμοποιήθηκαν από αστυνομικά τμήματα σε πόλεις των ΗΠΑ για λόγους δημόσιας ασφάλειας. Το 1934, εγκαταστάθηκαν σε 252 αστυνομικά τμήματα τα πρώτα αναλογικά συστήματα κινητών επικοινωνιών, που χρησιμοποιούσαν αναλογική διαμόρφωση πλάτους (AM). Περίπου 5000 συσκευές είχαν εγκατασταθεί σε αυτοκίνητα της αστυνομίας για την επικοινωνία με τα τμήματα. Με την εισαγωγή από τον Edwin Armstrong, το 1935, της διαμόρφωσης συχνότητας (FM), όλα τα συστήματα κινητών επικοινωνιών υιοθέτησαν την FM διαμόρφωση. Το 1946, εγκαταστάθηκαν για πρώτη φορά συστήματα κινητών επικοινωνιών σε 25 πόλεις των ΗΠΑ. Κάθε σύστημα χρησιμοποιούσε έναν πομπό σε υψηλό πύργο για να καλύπτει αποστάσεις μέχρι 50Km. Το εύρος ζώνης για τη φωνή ήταν 120KHz, σε half-duplex επικοινωνία. Στη δεκαετία του 1950, η εξέλιξη της τεχνολογίας επέτρεψε τη μείωση του εύρους στα 60KHz και στη δεκαετία του 1960 στα 30KHz.

Με την πάροδο των χρόνων, η ανάγκη για εξυπηρέτηση περισσότερων χρηστών έγινε ασφυκτικά μεγάλη. Στη δεκαετία του 1960, η AT&T Bell Labs, καθώς και άλλες εταιρίες τηλεπικοινωνιών, ανέπτυξαν τις βασικές αρχές των κυψελωτών συστημάτων. Η βασική ιδέα ήταν ο χωρισμός μιας περιοχής κάλυψης σε μικρές κυψέλες, κάθε μια από τις οποίες επαναχρησιμοποιεί διαύλους, ώστε να αυξηθεί η χωρητικότητα των συστημάτων. Παρόλο που η AT&T πρότεινε τα κυψελωτά συστήματα στην FCC το 1968, η τεχνολογία δεν ήταν ακόμη έτοιμη να τα υποστηρίξει μέχρι τα τέλη της δεκαετίας του 1970. Το 1976, το WARC (World Administrative Radio Conference) ενέκρινε την απόδοση συχνοτήτων για κυψελωτά συστήματα κινητών επικοινωνιών στην περιοχή των 800/900MHz, θέτοντας τη βάση για την ανάπτυξη εμπορικών συστημάτων.

#### *1.1.7.2 Κυψελωτά Συστήματα 1<sup>ης</sup> Γενιάς*

Το πρώτο παγκοσμίως κυψελωτό σύστημα που λειτούργησε ήταν στην Ιαπωνία, το 1979, από την Nippon Telephone and Telegraph (NTT). Το σύστημα χρησιμοποιούσε 600 FM duplex διαύλους με εύρος 25KHz στα 925 – 940 / 870 – 885MHz. Το 1981, η Ericsson ανέπτυξε το πρώτο ευρωπαϊκό κυψελωτό σύστημα, το NMT450, στη ζώνη 450 – 470MHz, το οποίο μετεξελίχθηκε στο NMT900, στη ζώνη 890 – 915 / 917 – 950MHz, το 1986. Το 1983, αναπτύχθηκε το πρώτο δοκιμαστικό κυψελωτό σύστημα στην Αμερική, το AMPS από την AT&T, στο Σικάγο, στη ζώνη 824 – 849 / 869 – 894MHz, με εύρος διαύλου τα 30KHz.

Τα NTT, NMT450, NMT900, AMPS, ETACS, JTACS, C-450 αποτελούν κυψελωτά συστήματα 1<sup>ης</sup> Γενιάς, με βασικά χαρακτηριστικά την αναλογική διαμόρφωση FM

και την τεχνική FDD. Η φασματική πυκνότητα ισχύος του διαμορφωμένου σήματος στα συστήματα FDD πρέπει να ελέγχεται προσεκτικά, ώστε η ακτινοβολούμενη ισχύς σε γειτονικούς διαύλους να είναι 60 – 80dB χαμηλότερα από την επιθυμητή. Όλα τα προαναφερθέντα συστήματα χρησιμοποιούσαν μια απόσταση 45MHz μεταξύ των συχνοτήτων εκπομπής και λήψης, τέτοια ώστε να είναι εφικτή η υλοποίηση του διπλέκτη με επαρκή απομόνωση των δύο συχνοτήτων.

Στον Πίνακα 1.1, απεικονίζονται τα βασικά χαρακτηριστικά των κυψελωτών συστημάτων 1<sup>ης</sup> Γενιάς.

Προδιαγραφή	Ζώνη Συχνοτήτων Αντίστροφη/Ευθεία Ζεύξη (MHz)	Απόσταση Φερόντων (KHz)	Αριθμός Διαύλων	Περιοχή Ανάπτυξης
NTT	925-940/870-885	25	600	Ιαπωνία
NMT450	453-457,5/463-467,5	25	180	Σουηδία
NMT900	890-915/935-960	12,5	1999	Σουηδία
AMPS	824-849/869-894	30	832	ΗΠΑ
ETACS	872-905/917-950	25	1240	Αγγλία
JTACS	915-925/860-870	25	400	Ιαπωνία
C-450	450-455,74/460-465,74	10	573	Γερμανία

Πίνακας 1.1 – Χαρακτηριστικά Κυψελωτών Συστημάτων 1<sup>ης</sup> Γενιάς

### 1.1.7.3 Μετάβαση στα Ψηφιακά Κυψελωτά Συστήματα

Οι σημερινές ψηφιακές επικοινωνίες έχουν τις ρίζες τους στην επαναστατική μελέτη του Nyquist, το 1924, ο οποίος καθόρισε το μέγιστο ρυθμό σηματοδοσίας που μπορεί να μεταδοθεί από τηλεγραφικό δίαυλο συγκεκριμένου εύρους ζώνης, χωρίς διασυμβολική παρεμβολή. Στη συνέχεια, ο Hartley, το 1928, ασχολήθηκε με το μέγιστο πλήθος δεδομένων που μπορούν να μεταδοθούν αξιόπιστα από ένα δίαυλο περιορισμένου εύρους ζώνης και με τη χρήση σημάτων πλάτους σε πολλαπλές στάθμες. Το 1948, ο Shannon έθεσε τις μαθηματικές αρχές της θεωρίας της πληροφορίας και τα βασικά όρια στα ψηφιακά συστήματα επικοινωνιών. Σημαντική συνεισφορά υπήρξε και εκείνη του Kotelnikov, το 1947, που ανέλυσε διάφορα ψηφιακά συστήματα επικοινωνιών, με βάση τη γεωμετρική προσέγγιση, καθώς και η επέκταση της εργασίας του από τους Wozencraft και Jacobs, το 1965. Στους θεμελιωτές των ψηφιακών συστημάτων συγκαταλέγεται και ο Hamming, που το 1950 δημοσίευσε τις εργασίες του πάνω στην αναγνώριση και διόρθωση σφαλμάτων.

Ενώ οι ψηφιακές τεχνικές γρήγορα υιοθετήθηκαν στα ενσύρματα συστήματα επικοινωνιών, έπρεπε να προηγηθεί η ραγδαία εξέλιξη στην τεχνολογία της μικροηλεκτρονικής, μέχρι να βρουν εφαρμογή στα ασύρματα συστήματα. Τα πλεονεκτήματα των ψηφιακών συστημάτων έναντι των αναλογικών είναι αρκετά και περιλαμβάνουν:

- την αυξημένη ανοσία στο θόρυβο

- τις περισσότερο αποδοτικές τεχνικές μετάδοσης και την καλύτερη ποιότητα υπηρεσιών, κάνοντας χρήση κωδικών διόρθωσης σφαλμάτων, φασματικά αποδοτικών τεχνικών διαμόρφωσης, αποδοτική κωδικοποίηση πληροφορίας και κωδικοποίηση διαύλου
- τη δυνατότητα εφαρμογής τεχνικών κρυπτογράφησης για την ασφάλεια της μετάδοσης
- την ευλυγισία στην ανάπτυξη και επέκταση των δικτύων
- τη χαμηλότερη κατανάλωση ισχύος
- επιτρέπουν την εφαρμογή διαφορετικών επιπέδων ποιότητας υπηρεσίας, παρέχοντας ταυτόχρονα υπηρεσίες φωνής και δεδομένων
- δίνουν τη δυνατότητα επεξεργασίας του σήματος
- είναι εύκολα υλοποιήσιμα σε VLSI χαμηλού κόστους
- παρέχουν τη δυνατότητα για επιπλέον τεχνικές πολλαπλής πρόσβασης (TDMA / CDMA / SDMA) και duplexing (TDD)
- επιτρέπουν την υλοποίηση software δεκτών

Υπάρχουν, βέβαια, και κάποια μειονεκτήματα, έναντι των αναλογικών συστημάτων, όπως:

- ανάγκη για ισοστάθμιση, ώστε να αποφεύγονται διασυμβολικές παρεμβολές
- κόστος αντικατάστασης υπάρχοντων αναλογικών συστημάτων, όπου αυτά υπάρχουν, π.χ. στις ΗΠΑ
- ποιότητα φωνής συχνά υποβαθμισμένη
- ανάγκη για συγχρονισμό
- αυξημένη πολυπλοκότητα αλγορίθμων, διαδικασιών ελέγχου και πρωτοκόλλων

Στις αρχές του 1980, είχε γίνει εμφανής η αδυναμία των αναλογικών κυνελωτών συστημάτων να αντεπεξέλθουν στις απαιτήσεις των επερχόμενων χρόνων, λόγω των τεσσάρων βασικών περιορισμών που τα χαρακτήριζαν:

1. εξαιρετικά περιορισμένο φάσμα προς απόδοση και, συνεπώς, χαμηλή χωρητικότητα συστημάτων
2. η αντίληψη των χρηστών ότι ήταν περιορισμένης χρησιμότητας, λόγω της χαμηλής ποιότητας υπηρεσιών, αλλά και του αριθμού των προσφερόμενων υπηρεσιών

3. αδυναμία να μειώσουν το κόστος των τερματικών και της υποδομής των δικτύων
4. ασυμβατότητα μεταξύ των διάφορων αναλογικών συστημάτων

Ήταν, συνεπώς, αναμενόμενη η στροφή της βιομηχανίας στις ψηφιακές τεχνικές, που αποδείχτηκαν ιδιαίτερα αποτελεσματικές και στα ενσύρματα συστήματα επικοινωνιών. Τα 2<sup>ης</sup> Γενιάς κυψελωτά δίκτυα βασίζονται όλα σε ψηφιακές, εκμεταλλευόμενα τα εμφανή και πολλαπλά πλεονεκτήματα.

#### *1.1.7.4 Κυψελωτά Συστήματα 2<sup>ης</sup> Γενιάς*

Στα κυψελωτά συστήματα 2<sup>ης</sup> Γενιάς συγκαταλέγονται σχεδόν όλα τα υπάρχοντα και εν λειτουργία συστήματα. Στηρίζονται στις τεχνικές TDMA ή DS-CDMA και για τις ΗΠΑ είναι τα IS-54, IS-136 και IS-95, για δε την Ευρώπη το GSM και για την Ιαπωνία το PDC.

- *GSM (900/1800):*

Το σύστημα GSM σχεδιάστηκε και υλοποιήθηκε εξ ολοκλήρου στην Ευρώπη και είναι το πλέον πετυχημένο κυψελωτό σύστημα παγκοσμίως. Η σχεδιάσή του ξεκίνησε το 1982 ως πανευρωπαϊκή προδιαγραφή και λειτούργησε το 1992 ως το πρώτο ψηφιακό κυψελωτό σύστημα. Στηρίζεται σε TDMA τεχνική, με 200KHz απόσταση φερόντων, αλλά συνδυάζει και την FDMA τεχνική με FDD. Κάθε φέρον έχει 8 διαύλους, με διάρκεια χρονοσχιμής τα 0,577msec, ενώ χρησιμοποιεί την τεχνική ψηφιακής διαμόρφωσης GMSK, με τελικό ρυθμό μετάδοσης τα 270,8Kbps. Οι συχνότητες λειτουργίας είναι οι 890 - 915MHz / 935 - 960MHz. Υποστηρίζει υπηρεσίες φωνής (13Kbps) και δεδομένων μέχρι 9,6Kbps. Το GSM είναι το πιο διαδεδομένο σύστημα παγκοσμίως και υλοποιήθηκε από περισσότερους από 560 παρόχους σε πάνω από 170 χώρες, προσφέροντας υπηρεσίες σε 1,25 δισεκατομμύρια χρήστες (Δεκέμβριος 2004).

Μια τροποποίηση του GSM900 είναι το GSM1800 ή DCS1800, το οποίο λειτουργεί στις ζώνες συχνοτήτων 1710 - 1785MHz / 1805 - 1880MHz. Το σύστημα έχει βελτιστοποιηθεί για τερματικές συσκευές χειρός με μέγιστη ισχύ το 1Watt ή τα 250mWatt και υποστηρίζει την επικάλυψη macrocells και picocells.

- *IS-54, IS-136 και IS-95:*

Στις ΗΠΑ, αναπτύχθηκαν δύο διαφορετικά συστήματα 2<sup>ης</sup> Γενιάς, το IS-54 με τεχνική TDMA, και το IS-95 με τεχνική CDMA. Και τα δύο συστήματα σχεδιάστηκαν ώστε να είναι συμβατά με το σύστημα AMPS. Η IS-54 προδιαγραφή, που πολλές φορές καλείται και USDC (US Digital Cellular), καθορίζει τερματικά διπλού τρόπου λειτουργίας, συμβατά με το AMPS, αλλά και ικανά για ψηφιακή σηματοδότηση βασισμένη στο TDMA (3 χρονοσχιμμές ανά φέρον), με απόσταση φερόντων τα 30KHz, διαμόρφωση π/4 – DQPSK, με τελικό ρυθμό μετάδοσης τα 48,6Kbps. Οι συχνότητες λειτουργίας είναι ίδιες με εκείνες του AMPS (824 –

894MHz). Το σύστημα υποστηρίζει δύο τύπους διαύλων σηματοδότησης ελέγχου, ένα για το ψηφιακό, με ρυθμό 48,6Kbps, και ένα για τους διαύλους του AMPS, με ρυθμό 10Kbps. Εξέλιξη του IS-54 είναι το IS-136, που παρέχει δυνατότητα αποστολής μικρών μηνυμάτων και υποστηρίζει κλειστές ομάδες χρηστών. Τα τεμαχικά του IS-136 δεν είναι συμβατά με εκείνα του IS-54, γιατί το IS-136 έχει διαύλους σηματοδότησης ελέγχου μόνο στα 48,6Kbps.

Το Μάρτιο του 1992, υιοθετήθηκε ένα επιπλέον σύστημα, το IS-95, που βασίζεται στην τεχνική CDMA και προτάθηκε από την Qualcomm. Χρησιμοποιεί διασπορά φάσματος Direct Sequence και παρουσιάζει ασυμμετρία ζεύξης, κάνοντας χρήση διαφορετικών τεχνικών για την ευθεία και την αντίστροφη ζεύξη. Σε κάθε κινητό σταθμό σε μια κυψέλη, αποδίδεται ένας διαφορετικός κώδικας, παρέχοντας έτσι πλήρη ορθογωνιότητα μεταξύ των χρηστών. Οι συχνότητες λειτουργίας είναι ίδιες με το AMPS και το IS-54, αλλά έχουν αποδοθεί και επιπλέον συχνότητες στην περιοχή (1,8 – 2GHz). Η επιτυχία του IS-95, αλλά και η πολλά υποσχόμενη τεχνική της διασποράς φάσματος που χρησιμοποιεί, το κατέστησαν οδηγό και βάση εκκίνησης για τα συστήματα 3<sup>ης</sup> Γενιάς.

- Personal Digital Cellular (PDC):

Το PDC αναπτύχθηκε στην Ιαπωνία από το 1989 και στηρίζεται στις αρχές του IS-54, με TDMA τεχνική πολλαπλής πρόσβασης (3 χρονοσχισμές ανά φέρον), απόσταση φερόντων τα 25KHz, διαμόρφωση  $\pi/4$  – DQPSK, με τελικό ρυθμό μετάδοσης τα 42Kbps. Ένα σημαντικό χαρακτηριστικό του PDC είναι η διαφορισμότητα στην κεραία του κινητού σταθμού. Οι συχνότητες λειτουργίας είναι τα 810 – 1501MHz.

Στον Πίνακα 1.2, παρουσιάζονται τα χαρακτηριστικά των κυψελωτών συστημάτων 2<sup>ης</sup> Γενιάς.

Σύστημα	Ζώνη Συχνοτήτων Αντίστροφη/Ευθεία Ζεύξη (MHz)	Απόσταση Φερόντων (KHz)	Διαμόρφωση	Τεχνική Πολλαπλής Πρόσβασης
GSM	890-915/935-960	200	GMSK	TDMA/ FDMA/FDD
IS-54	824-849/869-894	30	$\pi/4$ - DQPSK	TDMA/ FDMA/FDD
IS-95	824-849/869-894 1800-2000	1250	QPSK/BPSK	CDMA
PDC	810-830/940-960 1429-1453/1477-1501	25	$\pi/4$ - DQPSK	TDMA/ FDMA/FDD

Πίνακας 1.2 – Χαρακτηριστικά Κυψελωτών Συστημάτων 2<sup>ης</sup> Γενιάς

#### 1.1.7.5 Κυψελωτά Συστήματα 3<sup>ης</sup> Γενιάς

Η μετάβαση από τα κυψελωτά συστήματα 1<sup>ης</sup> Γενιάς σε εκείνα της 2<sup>ης</sup> Γενιάς σηματοδεύτηκε από την εισαγωγή των ψηφιακών τεχνικών, που έδωσαν τη δυνατότητα

παροχής υπηρεσιών φωνής σε μεγάλους πληθυσμούς και μεγάλες γεωγραφικές εκτάσεις, λόγω της πολύ καλής ποιότητας της φωνητικής υπηρεσίας. Η δυνατότητα, όμως, υποστήριξης υπηρεσιών δεδομένων είναι περιορισμένη στα 2<sup>ης</sup> Γενιάς συστήματα και η ανάγκη για παροχή υπηρεσιών με υψηλούς ρυθμούς μετάδοσης, ώστε να μεταδίδονται εικόνες υψηλής ποιότητας και βίντεο πραγματικού χρόνου ή να παρέχεται πρόσβαση στο Internet με υψηλές ταχύτητες, οδήγησε στη σχεδίαση των συστημάτων 3<sup>ης</sup> Γενιάς. Το βασικό χαρακτηριστικό τους είναι η υποστήριξη εφαρμογών πολυμέσων και η δυνατότητα πρόσβασης σε πληροφορίες και υπηρεσίες από δημόσια ή ιδιωτικά δίκτυα, με υψηλούς ρυθμούς μετάδοσης. Οι βασικές απαιτήσεις που τίθενται από τα 3<sup>ης</sup> Γενιάς συστήματα είναι οι εξής:

- Ρυθμοί μετάδοσης μέχρι και 2Mbps
- Μεταβαλλόμενος ρυθμός μετάδοσης για δυνατότητα προσφοράς εύρους ζώνης κατά απαίτηση
- Πολυπλεξία υπηρεσιών, με διαφορετικές απαιτήσεις ως προς την ποιότητα (π.χ. φωνή, video, πακέτα δεδομένων), στην ίδια σύνδεση
- Μεταβαλλόμενες απαιτήσεις ως προς την καθυστέρηση, ανάλογα με την ευαισθησία της υπηρεσίας ως προς την καθυστέρηση
- Μεταβαλλόμενες απαιτήσεις ποιότητας, ανάλογα με την υπηρεσία. από 10% FER (Frame Error Rate) ως  $10^{-6}$  BER (Bit Error Rate)
- Συνύπαρξη 2<sup>ης</sup> και 3<sup>ης</sup> Γενιάς συστημάτων και υποστήριξη διαπομπής μεταξύ των συστημάτων, για βελτίωση της κάλυψης και εξισορρόπηση του τηλεπικοινωνιακού φορτίου
- Υποστήριξη ασύμμετρης τηλεπικοινωνιακής κίνησης στην ευθεία και την αντίστροφη ζεύξη (π.χ. όπως στο web browsing, όπου η κίνηση στην ευθεία ζεύξη είναι πολύ μεγαλύτερη από την αντίστροφη)
- Μεγάλη φασματική απόδοση
- Συνύπαρξη FDD και TDD συστημάτων

Οι ραδιοεπαφές που έχουν αναπτυχθεί για τα συστήματα 3<sup>ης</sup> Γενιάς, που από την ITU καλούνται IMT-2000, είναι το WCDMA, ή UTRA, το multicarrier-CDMA, ή cdma2000, και το EDGE (Enhanced Data Rates for GSM Evolution). Οι συχνότητες που αποδόθηκαν από το WARC'92 για τα 3<sup>ης</sup> Γενιάς συστήματα είναι διαφορετικές για κάθε περιοχή. Αυτό οφείλεται στις ζώνες συχνοτήτων που είχαν καταλάβει τα συστήματα 2<sup>ης</sup> Γενιάς σε κάθε περιοχή. Ιδιαίτερα για την Ευρώπη, οι συχνότητες που αποδόθηκαν είναι  $2 \times 60\text{MHz}$  (1920-1980MHz uplink, 2110-2170MHz downlink) για WCDMA FDD συστήματα, 25MHz (1900-1920MHz και 2020-2025MHz) για TDD συστήματα με υποχρέωση έκδοσης άδειας, και 10MHz (2010-2020MHz) για TDD συστήματα χωρίς αδειοδότηση (SPA, Self Provided Applications).



#### 1.1.7.6 Ασύρματα Τοπικά Δίκτυα Επικοινωνιών (WLANs)

Τα ασύρματα τοπικά δίκτυα παρέχουν υπηρεσίες δεδομένων υψηλού ρυθμού μετάδοσης σε μια μικρή περιοχή, με κινητικότητα των χρηστών. Παγκοσμίως, τα συστήματα αυτά λειτουργούν στην ISM (Industrial Scientific & Medical) ζώνη συχνοτήτων, γιατί δεν απαιτείται άδεια λειτουργίας. Παραδείγματα ISM ζωνών είναι η περιοχή των 2,4GHz, των 5,2GHz, και των 17GHz, με κάποιες παραλλαγές ανά περιοχή. Βέβαια, το πλεονέκτημα της μη απαιτούμενης αδειοδότησης μετατρέπεται εύκολα σε μειονέκτημα, αν αναλογιστούμε ότι στην ίδια ζώνη συχνοτήτων λειτουργούν παρόμοια συστήματα, αυξάνοντας τις υπάρχουσες παρεμβολές. Για το λόγο αυτό, τα WLAN συστήματα πρέπει να εκπέμπουν με περιορισμένη ισχύ ανά μονάδα εύρους ζώνης και για να το επιτύχουν χρησιμοποιούν τεχνικές διασποράς φάσματος. Οι προδιαγραφές που υφίστανται προέρχονται από δύο οργανισμούς, την IEEE και το Ευρωπαϊκό Ινστιτούτο Τηλεπικοινωνιακών Προδιαγραφών, το λεγόμενο ETSI.

Η πρώτη προδιαγραφή ευρείας αποδοχής ήταν η IEEE 802.11, η οποία προβλέπει την εφαρμογή δύο εναλλακτικών τεχνολογιών, την Direct Sequence διασπορά φάσματος (DSSS) και την Frequency Hopped διασπορά (FHSS). Για την DSSS τεχνολογία, το εύρος του διαύλου είναι 20MHz και ο ρυθμός μετάδοσης κυμαίνεται από 1 έως 2Mbps. Οι συχνότητες λειτουργίας είναι στην περιοχή των 2,4GHz και η προδιαγραφή είναι προσανατολισμένη σε δίκτυα Ethernet. Το Νοέμβριο του 1999, έγινε δεκτή μια τροποποίηση της προδιαγραφής και προέκυψε το IEEE 802.11b, το οποίο υποστηρίζει ρυθμούς μετάδοσης μέχρι 11Mbps και λειτουργεί στη περιοχή των 2,4GHz. Παράλληλα, αναπτύχθηκε και μια ακόμη προδιαγραφή από την IEEE, η IEEE 802.11a, με ρυθμούς μετάδοσης μέχρι 54Mbps, αλλά στην περιοχή των 5,2GHz, ενώ πιο πρόσφατα υιοθετήθηκε και η προδιαγραφή IEEE 802.11g, η οποία είναι συμβατή με την IEEE 802.11b και μπορεί να προσφέρει ταχύτητες μέχρι 54Mbps.

Μια άλλη οικογένεια προδιαγραφών είναι αυτή των HIPERLAN, η οποία ξεκίνησε το 1991 στους κόλπους του ETSI και οδήγησε το 1997 στην πρώτη προδιαγραφή με υψηλό ρυθμό μετάδοσης, το HIPERLAN Type 1, που υποστήριζε μέχρι 23,5Mbps. Κατά την ανάπτυξη της προδιαγραφής, δόθηκε μεγάλο βάρος σε εφαρμογές τοπικών δικτύων (LANs), και για το λόγο αυτό θεωρείται ένα μοντέρνο ασύρματο Ethernet. Λόγω του υπερβολικού φορτίου στην περιοχή των 2,4GHz, το HIPERLAN Type 1 σχεδιάστηκε για την περιοχή των 5,2GHz. Τον Απρίλιο του 1997, ο ETSI αποφάσισε ότι οι ανάγκες της αγοράς επέβαλαν την επέκταση της προδιαγραφής, ώστε να περιλάβει εφαρμογές ασύρματου ATM (Wireless ATM). Δημιούργησε, λοιπόν, το έργο ETSI BRAN (Broadband Radio Access Networks), από το οποίο έχουν προκύψει άλλες τρεις προτάσεις, το HIPERLAN Type 2, το HIPERLAN Type 3, που αργότερα μετονομάστηκε σε HiperACCESS, και το HIPERLAN Type 4, που αργότερα ονομάστηκε HiperLINK.

Στον Πίνακα 1.3, εμφανίζονται οι τέσσερις τύποι HIPERLAN με τα κυριότερα χαρακτηριστικά τους. Ιδιαίτερα το HIPERLAN Type 2 είναι μια προδιαγραφή με έμφαση στην κυψελωτή αρχιτεκτονική και τη διασυνεργασία με άλλα τηλεπικοινωνιακά δίκτυα.

Τύπος	Ζώνη Συχνοτήτων	Εύρος Κάλυψης	Ρυθμός Μετάδοσης	Περιγραφή
HIPERLAN 1	5,2GHz	50 - 100m	23,5Mbps	Wireless LAN
HIPERLAN 2	5,2GHz	50 - 100m	54Mbps	Wireless IP – ATM
HiperACCESS	υπό εξέταση	5Km	25Mbps	Wireless IP – ATM
HiperLINK	17GHz	150m	155Mbps	Wireless

**Πίνακας 1.3 – Χαρακτηριστικά HIPERLAN Συστημάτων**

#### 1.1.7.7 Ασύρματα Προσωπικά Δίκτυα Επικοινωνιών (WPANs)

Τα ασύρματα προσωπικά δίκτυα επικοινωνιών (WPANs, Wireless Personal Area Networks) στοχεύουν στην ασύρματη δικτύωση φορητών υπολογιστών, κινητών υπολογιστικών μονάδων, όπως τα PDAs (Personal Digital Assistants), περιφερειακών, κυψελωτών τηλεφώνων και άλλων ηλεκτρονικών συσκευών, που βρίσκονται σε μικρές αποστάσεις μεταξύ τους. Χαρακτηριστικό παράδειγμα προδιαγραφής είναι το Bluetooth, που είναι μια ραδιοτεχνολογία που παρέχει ασύρματη σύνδεση μεταξύ ασύρματων συσκευών σε αποστάσεις μέχρι 10 μέτρα περίπου. Το σύστημα λειτουργεί στην ISM ζώνη συχνοτήτων (2,4GHz), χρησιμοποιώντας τεχνική μεταπήδησης συχνότητας για την αποφυγή παρεμβολών, και υποστηρίζει ρυθμούς μετάδοσης των 700Kbps, για μεταφορά δεδομένων, και μέχρι 3 τηλεφωνικές συνδέσεις των 64Kbps. Οι συσκευές που χρησιμοποιούν τεχνολογία Bluetooth, αποτελούν ένα ad hoc δίκτυο, που καλείται και piconet, δηλαδή ένα δυναμικά δημιουργούμενο και επαναπροσδιοριζόμενο σύμφωνα με τις τρέχουσες απαιτήσεις δίκτυο.

#### 1.1.7.8 Δίκτυα Τηλεοπτικών Μεταδόσεων

Τα δίκτυα τηλεοπτικών μεταδόσεων (Broadcast Television Networks) είχαν αρχικά σχεδιαστεί προκειμένου να υποστηρίξουν τη διάδοση της αναλογικής τηλεόρασης (όπως και του ραδιοφώνου) σε μεγάλες γεωγραφικές περιοχές. Στην εποχή της ψηφιακής τηλεόρασης, το μέσο μετάδοσης είναι ένα καλωδιακό δίκτυο διανομής για την περίπτωση των πόλεων, ενώ για ευρύτερες περιοχές συνήθως χρησιμοποιείται είτε ένα δορυφορικό δίκτυο είτε ένα επίγειο ασύρματο δίκτυο. Πάνω από τα δίκτυα αυτά προσφέρονται όχι μόνο τηλεοπτικές ψηφιακές υπηρεσίες, αλλά και μια ποικιλία άλλων υπηρεσιών, ιδιαίτερα στον τομέα της ψυχαγωγίας (π.χ. δικτυακά παιχνίδια) και του ηλεκτρονικού εμπορίου (αγορές μέσω Internet).

Προκειμένου η επικοινωνία με το χρήστη να είναι αμφίδρομη και ο τελευταίος να έχει τη δυνατότητα της αλληλεπίδρασης, υπάρχει και αντίστροφη ζεύξη, η οποία υλοποιείται με τη χρήση ενός καναλιού χαμηλού ρυθμού μετάδοσης. Η δυνατότητα της αλληλεπίδρασης είναι πολύ σημαντική, αφού, χάρη σε αυτήν, η εμπειρία του χρήστη επεκτείνεται πέρα από την απλή παρακολούθηση τηλεοπτικών

προγραμμάτων και του προσφέρονται πολλές νέες δυνατότητες, όπως, για παράδειγμα, είναι η υπηρεσία μετάδοσης βίντεο «κατ' απαίτηση» (video on demand ή movie on demand) ή η δυνατότητα εναλλαγής της γωνίας θέασης μιας εκπομπής.

Σε οικιακό περιβάλλον, για να έχει κάποιος πρόσβαση στην υπηρεσία ψηφιακής τηλεόρασης, γίνεται χρήση μιας συσκευής η οποία συνδέεται με την τηλεόραση και καλείται set-top box (STB). Στην περίπτωση ενός καλωδιακού δικτύου διανομής, το STB συνδέεται με το δίκτυο αυτό και παρέχει όχι μόνο έλεγχο των τηλεοπτικών καναλιών που λαμβάνονται αλλά και πρόσβαση σε πολλές άλλες υπηρεσίες. Για παράδειγμα, όταν ένα καλωδιακό μόντεμ (cable modem) είναι ενσωματωμένο στο STB, προσφέρει στο χρήστη τόσο ένα χαμηλού ρυθμού μετάδοσης όσο και ένα υψηλού ρυθμού μετάδοσης κανάλι. Συνήθως, το πρώτο χρησιμοποιείται για να συνδέει το συνδρομητή με ένα PSTN δίκτυο, ενώ το δεύτερο για να παρέχει σύνδεση στο Internet. Κατά συνέπεια, πέρα από το να προσφέρει ραδιοτηλεοπτικές εκπομπές, ένα καλωδιακό δίκτυο διανομής παρέχει, επίσης, πρόσβαση σε μια πλειάδα υπηρεσιών που είναι διαθέσιμες σε ένα PSTN δίκτυο ή στο Internet. Ομοίως, σε ένα ασύρματο επίγειο ή δορυφορικό δίκτυο, όταν ένα PSTN μόντεμ υψηλής ταχύτητας είναι ενσωματωμένο στο STB, προσφέρει στο χρήστη ένα κανάλι «αλληλεπίδρασης», με αποτέλεσμα να διευρύνεται το φάσμα των διαθέσιμων υπηρεσιών.

Η ασύρματη πρόσβαση στις υπηρεσίες ψηφιακής τηλεόρασης είναι εκείνη που ενδιαφέρει περισσότερο και αναμένεται να γνωρίσει σημαντική άνθηση. Η τεχνολογία που υποστηρίζει τη μετάδοση ψηφιακής τηλεόρασης είναι η τεχνολογία DVB (Digital Video Broadcasting), και διακρίνεται στην DVB-T (Digital Video Broadcasting – Terrestrial), για την περίπτωση της επίγειας ασύρματης εκπομπής, και στην DVB-S (Digital Video Broadcasting – Satellites), για την περίπτωση της δορυφορικής εκπομπής (το αντίστοιχο σύστημα στη Βόρεια Αμερική είναι το DBS – Digital Broadcast Satellites).

Στον Πίνακα 1.4, συνοψίζονται τα τυπικά εύρη ζώνης που απαιτούνται για τη μετάδοση τηλεόρασης και βίντεο, ανάλογα με την επιθυμητή ποιότητα. Όπως παρατηρούμε, ο απαιτούμενος ρυθμός μετάδοσης ξεκινά από τα 1,5Mbps για βίντεο ποιότητας VCR και φτάνει ακόμη και τα 80Mbps για τηλεόραση υψηλής ευκρίνειας (high-definition).

Ποιότητα	Εύρος Ζώνης	Πρωτόκολλο
VCR-Quality Video	1,5 Mbps	MPEG-1
Broadcast-Quality Video	4 / 6 / 8 Mbps	MPEG-2, Main
Studio-Quality Television	9 / 15 / 18 Mbps	MPEG-2, Main
High-Definition Television	60 / 80 Mbps	MPEG-2, High

Πίνακας 1.4 – Τυπικά Εύρη Ζώνης για την Εκπομπή Τηλεόρασης και Βίντεο

## 1.2 Δικτυακές Υποδομές και Λειτουργίες

### 1.2.1 Εισαγωγή

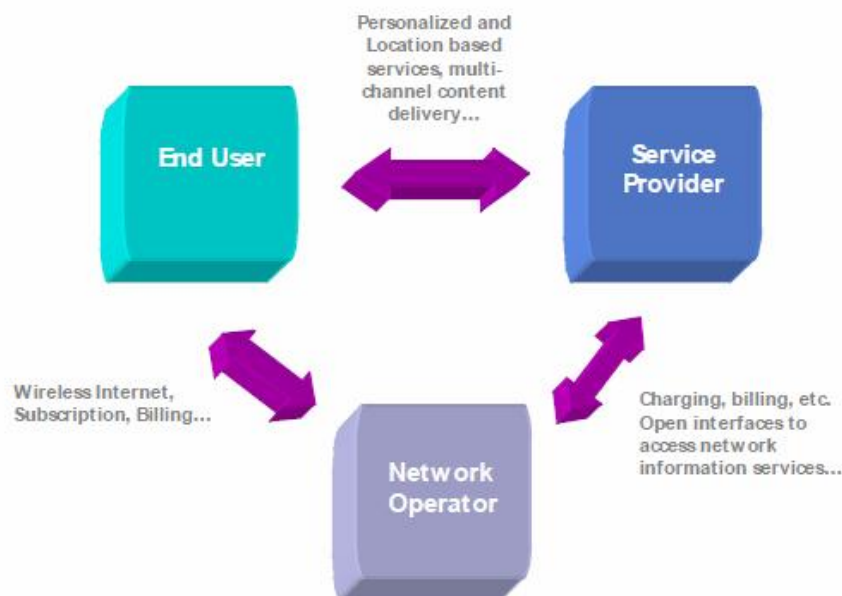
Στο τμήμα αυτό, θα παρουσιαστεί ένα μοντέλο αρχιτεκτονικής για τα συστήματα 4G. Κύριος σκοπός αυτής της αρχιτεκτονικής είναι να παρέχει στο χρήστη υπηρεσίες πάνω από ετερογενή δίκτυα, οι οποίες βασίζονται στην ανοικτή και ενοποιημένη δομή των πολλαπλών δικτύων πρόσβασης. Βασική απαίτηση είναι και η κάλυψη της ανάγκης για «προσωποποίηση» των υπηρεσιών, με παράλληλη προστασία της ασφάλειας, τόσο του χρήστη όσο και των παρόχων υπηρεσιών. Όλα αυτά προϋποθέτουν τη διάθεση περισσότερων πληροφοριών στις νέες υπηρεσίες. Καθώς οι πληροφορίες αυτές διακινούνται σε πολλές υπηρεσίες, πρέπει να υλοποιηθούν μηχανισμοί ελέγχου πρόσβασης, προκειμένου να αποφευχθεί μη εγκεκριμένη χρήση.

Η επίτευξη των παραπάνω απαιτεί ένα σύνολο κοινής λειτουργικότητας, το οποίο δε θα επικεντρώνεται ούτε μόνο στην υπηρεσία ούτε μόνο στο δίκτυο. Παρακάτω θα παρουσιαστεί μια νέα έννοια, η έννοια της «Υποστήριξης Υπηρεσίας» («Service Support»), που θα ενθυλακώνει όλη αυτήν τη λειτουργικότητα σε ενοποιημένη μορφή.

Προτού προχωρήσουμε σε περισσότερες λεπτομέρειες για την αρχιτεκτονική των συστημάτων 4G και για την Υποστήριξη Υπηρεσίας, θα αναφερθούμε στα επιχειρηματικά μοντέλα που αναπτύσσονται για τα 4G δίκτυα.

### 1.2.2 Τα Επιχειρηματικά Μοντέλα

Ένα τυπικό επιχειρηματικό μοντέλο Τηλεπικοινωνιών αποτελείται, συνήθως, από τρεις μονάδες: τον (τελικό) χρήστη, τον πάροχο υπηρεσίας και τον πάροχο δικτύου, όπως απεικονίζεται στο επόμενο σχήμα.

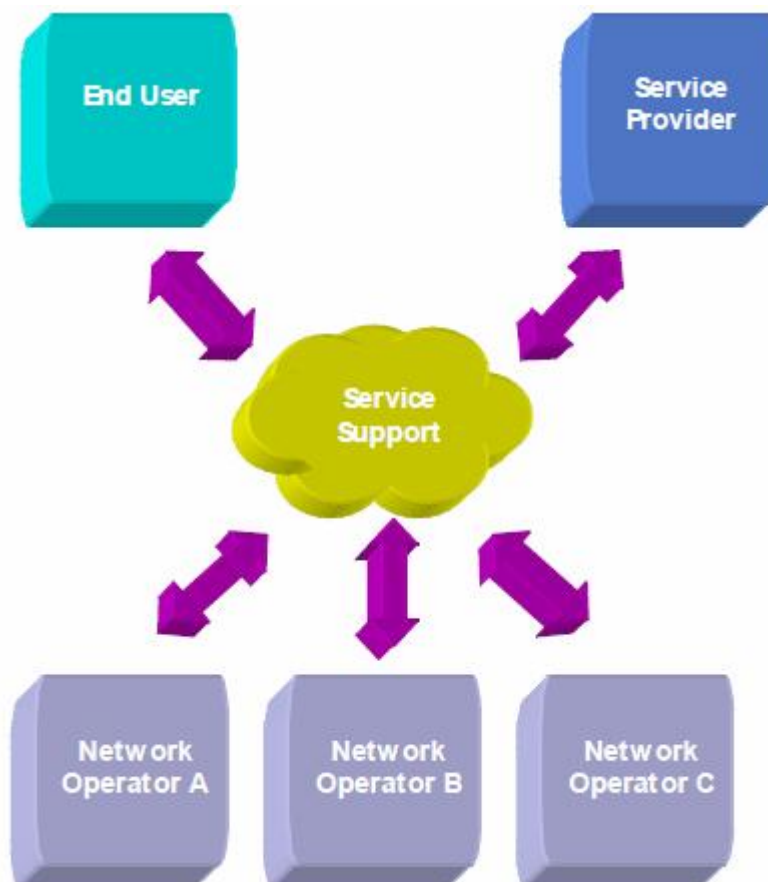


Στο μοντέλο αυτό, ο χρήστης έχει μια εμπορική συμφωνία (π.χ. μια συνδρομή) με:

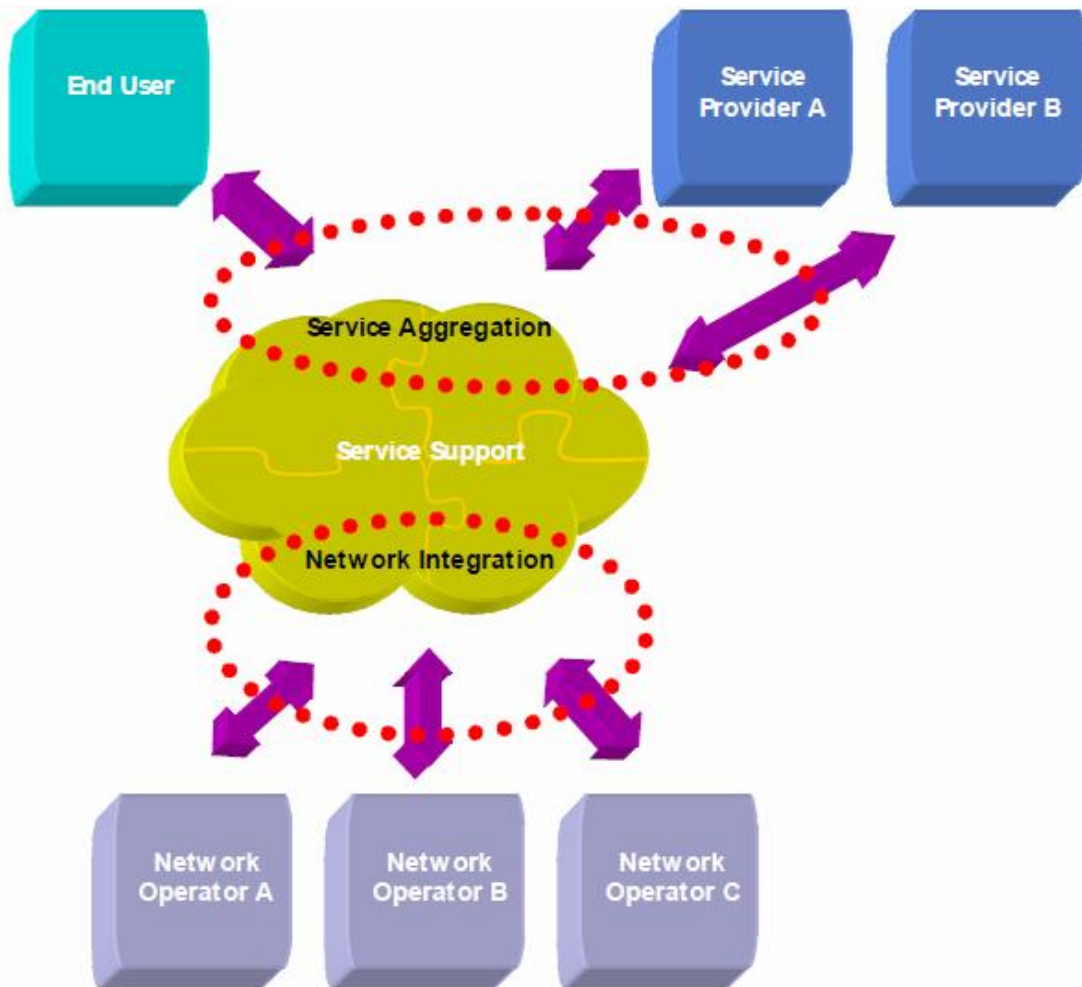
- Τον πάροχο δικτύου, προκειμένου να αποκτά συνδεσιμότητα με τα διάφορα δίκτυα (π.χ. GSM, UMTS, WLAN, DVB).
- Τον πάροχο υπηρεσιών, προκειμένου να αποκτά πρόσβαση σε υπηρεσίες και εφαρμογές (π.χ. e-mail, video streaming, κ.λπ.).

Το μοντέλο αυτό υποθέτει ότι υπάρχουν εμπορικές συμφωνίες μεταξύ του τελικού χρήστη, του πάροχου υπηρεσίας και του πάροχου δικτύου.

Στον κόσμο των συστημάτων 4G, ο αριθμός των δικτύων πρόσβασης και κατ' επέκταση των παρόχων δικτύου αυξάνεται ραγδαία. Ειδικά η κατασκευή των επονομαζόμενων «hotspots», περιοχών, δηλαδή, με κάλυψη ασύρματου τοπικού δικτύου, συνεισφέρει στην προαναφερθείσα ραγδαία αύξηση. Τόσο, όμως, ο χρήστης όσο και ο πάροχος υπηρεσιών δεν ενδιαφέρονται για τα ιδιαίτερα χαρακτηριστικά κάθε τεχνολογίας πρόσβασης. Για παράδειγμα, ο χρήστης θέλει να λαμβάνει υπηρεσίες που προσφέρονται από τους παρόχους υπηρεσιών, ανεξάρτητα από τον τύπο του δικτύου στο οποίο είναι συνδεδεμένος. Αυτό, λοιπόν, απαιτεί μια καινούρια προσέγγιση, με επίκεντρο την υπηρεσία, όπου ο τελικός χρήστης, ο πάροχος υπηρεσιών και ο πάροχος δικτύου συνδέονται δια μέσου ενός στρώματος Υποστήριξης Υπηρεσίας σε ένα νέο επιχειρηματικό μοντέλο, που διαμορφώνεται πλέον ως εξής:



Δύο βασικές ευθύνες επωμίζεται η Υποστήριξη Υπηρεσίας: τη «συγκέντρωση υπηρεσιών» (service aggregation – SA) και την «ενοποίηση δικτύων» (network integration – NI). Η πρώτη επιτρέπει την αδιάλειπτη παροχή υπηρεσιών στους τελικούς χρήστες, ενώ η δεύτερη εξασφαλίζει ότι αυτό πραγματοποιείται ανεξάρτητα από το δίκτυο στο οποίο οι χρήστες είναι συνδεδεμένοι. Το όφελος του χρήστη είναι ότι μπορεί να έχει πρόσβαση στις υπηρεσίες στις οποίες είναι συνδρομητής, σε κάθε μέρος και κάθε στιγμή. Πρέπει να σημειωθεί ότι ο ρόλος της Υποστήριξης Υπηρεσίας υλοποιείται από πολλές συνεργαζόμενες μονάδες, παρουσιάζεται δε στο σχήμα που ακολουθεί.



Μία σημαντική δυνατότητα που παρέχει το μοντέλο αυτό είναι η ελάττωση της γραφειοκρατικής διαδικασίας ταυτοποίησης του χρήστη, αφού η ταυτότητα του χρήστη, μετά την πρώτη του σύνδεση, μπορεί να φυλάσσεται από τη μονάδα Υποστήριξης Υπηρεσίας και να γίνεται χρήση της όποτε ο χρήστης μεταβαίνει σε διαφορετικό δίκτυο πρόσβασης. Έτσι, ο χρήστης δε θα χρειάζεται να παρέχει τα στοιχεία του για ταυτοποίηση παραπάνω από μία φορά.

### 1.2.3 Δίκτυα Επόμενης Γενιάς

Όπως αναφέρθηκε προηγουμένως, ο αριθμός των δικτύων πρόσβασης σε δημόσιους, επαγγελματικούς και οικιακούς χώρους αυξάνεται με αλματώδη ρυθμό. Προκειμένου οι χρήστες και οι υπηρεσίες να αποφύγουν την απευθείας επαφή με τόσες πολλές ετερογενείς τεχνολογίες, στρατηγικές ταυτοποίησης και ιδιαιτερότητες των δικτύων, οι πληροφορίες αυτές πρέπει να τους αποκρύπτονται.

Από την άλλη πλευρά, όμως, τα ιδιαίτερα χαρακτηριστικά ενός συγκεκριμένου δικτύου πρόσβασης έχουν μεγάλη σημασία. Οι τελικοί χρήστες, για παράδειγμα, θέλουν να μπορούν να χρησιμοποιούν ένα συγκεκριμένο δίκτυο, βασισμένοι σε κριτήρια επιλογής, όπως η ταχύτητα και το κόστος. Οι πάροχοι των υπηρεσιών, επίσης, χρειάζεται να γνωρίζουν ειδικά χαρακτηριστικά του δικτύου, προκειμένου να βελτιώσουν τις υπηρεσίες τους, ενώ μπορεί, παράλληλα, να έχουν κάποιες ειδικές απαιτήσεις, όπως εγγυήσεις για ελάχιστη καθυστέρηση.

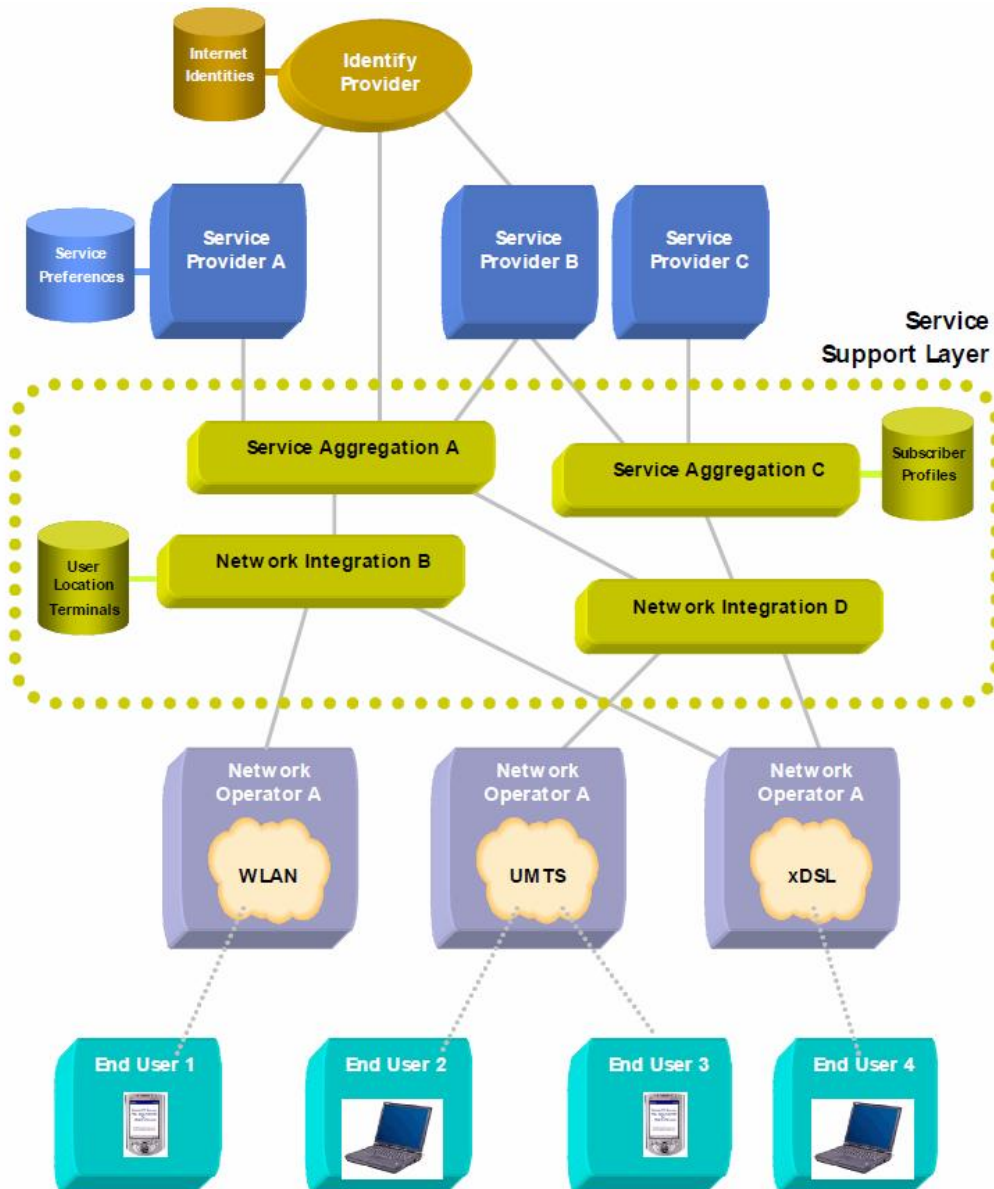
Για τον τελικό χρήστη, αυτό σημαίνει ότι στην πλειοψηφία των περιπτώσεων δε θα αντιλαμβάνεται ότι οι υπηρεσίες του λαμβάνονται ταυτόχρονα από διαφορετικά δίκτυα πρόσβασης. Η ιδέα αυτή περιλαμβάνει την έννοια της διαφανούς πρόσβασης. Μια άλλη, όμως, πτυχή είναι η ανάγκη του χρήστη να ελέγχει τη χρήση των διαθέσιμων δικτύων, ειδικά όταν αυτή η χρήση συνεπάγεται και κάποιο κόστος. Η ανάγκη αυτή περιλαμβάνει μια σύνθετη διαδικασία λήψης απόφασης, η οποία μπορεί να διαμορφώνεται από πολιτικές διαχείρισης, αλλά και από τις προτιμήσεις του χρήστη.

### 1.2.4 Εισαγωγή στο Μοντέλο Αρχιτεκτονικής για Υπηρεσίες και Δίκτυα 4<sup>ης</sup> Γενιάς

Προχωρούμε, τώρα, σε μια πιο λεπτομερή παρουσίαση του μοντέλου αρχιτεκτονικής για συστήματα 4<sup>ης</sup> Γενιάς. Το μοντέλο αυτό, όπως φαίνεται και στο παρακάτω σχήμα, συμπεριλαμβάνει τόσο επιχειρηματικούς ρόλους όσο και αφηρημένα τμήματα λειτουργικότητας.

Όπως τονίζει το σχήμα αυτό, κεντρικό ρόλο σε ένα σύστημα 4G διαδραματίζει το στρώμα Υποστήριξης Υπηρεσίας, ο ρόλος του οποίου έχει ήδη παρουσιαστεί συνοπτικά στις παραγράφους 1.2.1 και 1.2.2. Στις επόμενες παραγράφους, θα αναλυθούν εκτενέστερα οι δύο βασικές λειτουργίες που ενσωματώνει το στρώμα αυτό, η συγκέντρωση υπηρεσιών και η ενοποίηση δικτύων.

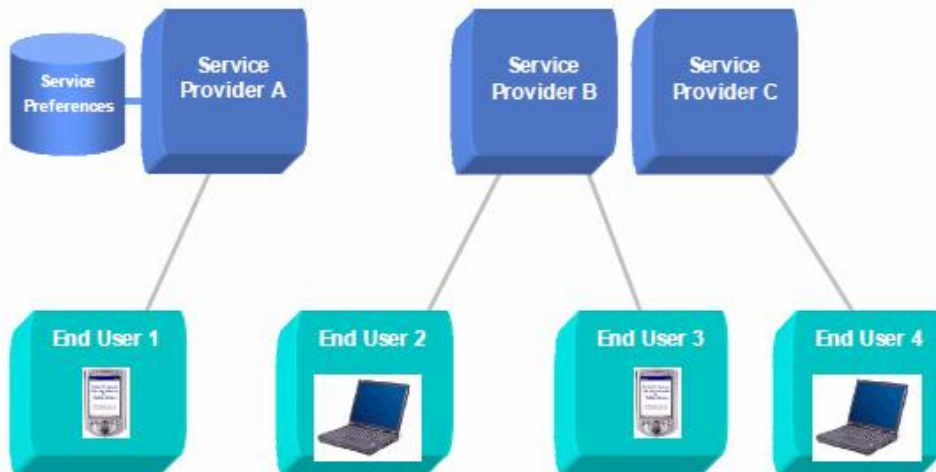




### 1.2.5 Υπηρεσίες Τελικού Χρήστη – Ταυτότητα Τελικού Χρήστη

Στα παραπάνω, έχει θεωρηθεί ότι οι υπηρεσίες παρέχονται στο χρήστη μέσω μιας τρίτης πλευράς, των παρόχων υπηρεσιών. Οι πάροχοι αυτοί έχουν υπογράψει μια εμπορική συμφωνία με μία ή περισσότερες μονάδες συγκέντρωσης υπηρεσιών (service aggregators). Με αυτόν τον τρόπο, οι πάροχοι υπηρεσιών μπορούν όχι μόνο να επωφεληθούν από μια ευρεία πελατειακή βάση, αλλά επίσης να προσαρμόσουν τις υπηρεσίες τους, βασισμένοι σε πληροφορίες από το δίκτυο, οι οποίες τους παρέχονται μέσω των μονάδων συγκέντρωσης υπηρεσιών. Επιπλέον, η υπηρεσία μπορεί να προσαρμοστεί στο μεταβαλλόμενο περιβάλλον και στις μεταβαλλόμενες προτιμήσεις του χρήστη (για παράδειγμα, ο χρήστης μπορεί να μεταβεί σε ένα διαφορετικό δίκτυο, με μεγαλύτερο bandwidth). Το πώς ο χρήστης αντιλαμβάνεται την παροχή των διαφόρων υπηρεσιών απεικονίζεται στο σχήμα που ακολουθεί.





Είναι εμφανές ότι η έννοια της «ταυτότητας χρήστη» διαδραματίζει ένα σημαντικό ρόλο στην ενοποιημένη παράδοση των υπηρεσιών. Η έννοια αυτή εκτείνεται από την ταυτοποίηση και τον έλεγχο πρόσβασης στην υπηρεσία μέχρι τη χρέωση και το λογαριασμό του χρήστη. Στο πλαίσιο των δικτύων και υπηρεσιών επόμενης γενιάς, η έννοια της ταυτότητας του χρήστη εμπεριέχει, επίσης, και το προφίλ των προτιμήσεων. Η χρήση των πληροφοριών ταυτοποίησης με έναν ενοποιημένο τρόπο θα ενισχύσει την εδραίωση ενός τρόπου διαχείρισης που θα επικεντρώνεται γύρω από τις προτιμήσεις του χρήστη.

#### 1.2.6 Μονάδα Συγκέντρωσης Υπηρεσιών

Ο ρόλος μιας μονάδας συγκέντρωσης υπηρεσιών είναι διττός: Από τη μια πλευρά, δρα σαν ενδιάμεσος μεταξύ ενός τελικού χρήστη – συνδρομητή και των προσφερόμενων υπηρεσιών. Από την άλλη πλευρά, για τον πάροχο υπηρεσιών, η μονάδα συγκέντρωσης δρα σα μεσολαβητής μεταξύ αυτού και των πολλαπλών διαφορετικών παρόχων δικτύου, έτσι ώστε οι υπηρεσίες να μη χρειάζεται να ασχολούνται με θέματα όπως ο προσδιορισμός του δικτύου στο οποίο βρίσκεται ο χρήστης μία δεδομένη στιγμή. Σε αυτό το μοντέλο, οι χρήστες αποκτούν μία και μόνο συνδρομή, με τη μονάδα συγκέντρωσης υπηρεσιών. Η μονάδα διαχειρίζεται αυτές τις συνδρομές και, επίσης, αναλαμβάνει την έκδοση του λογαριασμού, βασιζόμενη σε πληροφορίες χρεώσεων που λαμβάνει από τις υπηρεσίες, τους παρόχους δικτύου, αλλά και την ίδια τη μονάδα. Με αυτόν τον τρόπο, δεν παρέχει απλά τη διευκόλυνση της μιας και μοναδικής διαδικασίας ταυτοποίησης του χρήστη αλλά και, γενικότερα, ένα μοναδικό σημείο επαφής.

Στα συμβατικά δίκτυα τηλεπικοινωνιών, η ταυτοποίηση του τερματικού (τηλεφωνικός αριθμός ή ταυτότητα) συχνά εξυπηρετούσε την ταυτοποίηση του χρήστη. Σε ετερογενή δίκτυα, όμως, όπου οι χρήστες μπορεί να έχουν πρόσβαση σε πολλαπλά δίκτυα και, πιθανώς, να έχουν πολλά διαφορετικά είδη τερματικών, κάτι τέτοιο δεν είναι πλέον εφικτό. Συνεπώς, υπάρχει η ανάγκη για ένα υψηλότερου επιπέδου αναγνωριστικό (ταυτότητα), που θα καθορίζει τις συνοδευτικές πληροφορίες μιας συνδρομής. Οι πληροφορίες αυτές είναι ένα προσωπικό περιβάλλον, χάρη στο οποίο ένας συνδρομητής επιτρέπεται να έχει πρόσβαση σε

υπηρεσίες, μέσω διαφορετικών δικτύων πρόσβασης και τερματικών. Ένα τέτοιο περιβάλλον ανήκει σε ένα συγκεκριμένο χρήστη, παρόλα αυτά ένας χρήστης μπορεί να έχει πολλά τέτοια περιβάλλοντα (π.χ. ένα που θα του παραχωρηθεί από τον εργοδότη του και ένα προσωπικό), για τα οποία και θα λάβει τελικά έναν ξεχωριστό λογαριασμό. Τέτοια αναγνωριστικά μπορούν να δίνονται από την ίδια τη μονάδα συγκέντρωσης υπηρεσιών ή από τρίτους παρόχους ταυτοτήτων. Η παροχή ταυτοτήτων και, ενδεχομένως, σχετικών προφίλ πληροφοριών είναι μια υπηρεσία όπως όλες οι άλλες, ενώ τεχνικές υλοποίησής της εφαρμόζονται ήδη στο χώρο του Internet. Πρόκειται για τεχνικές τύπου Web, που στηρίζονται, δηλαδή, στο πρωτόκολλο HTTP.

Αρκετά σημαντικά ζητήματα πρέπει να εξεταστούν σε αυτόν τον τομέα, όπως είναι η συλλογή των υπαρχουσών πληροφοριών από τις εμπλεκόμενες μονάδες (ταυτότητες, προφίλ) και η υλοποίηση της ιδέας της μιας και μοναδικής διαδικασίας ταυτοποίησης.

### *1.2.7 Η Ενοποίηση των Δικτύων*

Η ενοποίηση δικτύων επιτρέπει στους χρήστες να μεταβαίνουν, διαφανώς, μεταξύ ετερογενών δικτύων και των αντίστοιχων διαχειριστικών περιοχών. Το σημείο κλειδί της λειτουργίας αυτής είναι η διαφάνεια.

Η ενοποίηση δικτύων παρέχει τη βασική λειτουργία της διαχείρισης του «ενοποιημένου» δικτύου, καθώς και μια διαφορετική εικόνα των υποκείμενων δικτύων. Παρέχεται, με αυτόν τον τρόπο, τόσο στις ανώτερες υπηρεσίες όσο και στους χρήστες, η ψευδαίσθηση ενός και μόνο εικονικού δικτύου. Για παράδειγμα, παρέχεται διαφανής και ενοποιημένη ταυτοποίηση, έτσι ώστε οι χρήστες να μη χρειάζεται να κάνουν «login», κάθε φορά που αλλάζουν δίκτυο πρόσβασης. Όσον αφορά τις υπηρεσίες, η ενοποίηση δικτύων αναλαμβάνει το θέμα της συνδεσιμότητας, δηλαδή της δυνατότητας πρόσβασης στο τερματικό κάθε τελικού χρήστη. Η διαδικασία αυτή εμπλέκει θέματα διευθυνσιοδότησης δικτύου και τεχνολογίες όπως το Mobile IP.

Παράλληλα, η ενοποίηση δικτύων μεταφέρει συγκεκριμένες πληροφορίες και λειτουργίες από το υποκείμενο δίκτυο στο ανώτερο στρώμα, π.χ. πληροφορίες για την τοποθεσία του χρήστη, μέσω ανοικτών και προκαθορισμένων διεπαφών.

Τα διάφορα δίκτυα, σε ένα σύστημα 4G, δεν προσφέρουν απλά συνδεσιμότητα, αλλά, επιπλέον, παρέχουν και συγκεκριμένες υπηρεσίες δικτύου. Η λειτουργία ενοποίησης δικτύων είναι υπεύθυνη και για την ομαδοποίηση και ενοποίηση αυτών των υπηρεσιών. Για παράδειγμα, σε ένα UMTS δίκτυο η γεωγραφική θέση του χρήστη μπορεί να ληφθεί χρησιμοποιώντας «OSA/Parlay», αλλά ένα δημόσιο ασύρματο τοπικό δίκτυο μπορεί να παρέχει μια διαφορετική διεπαφή, προκειμένου να αποκτηθεί αυτή η πληροφορία.

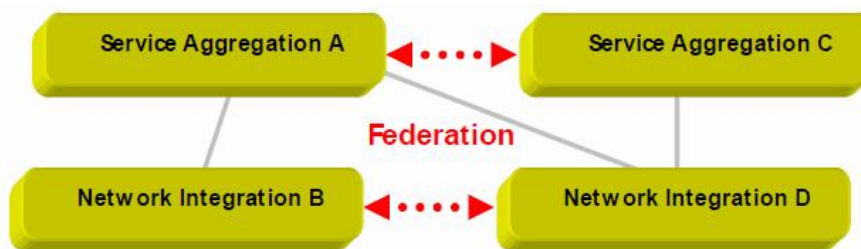
Το υπο-επίπεδο ενοποίησης δικτύων του στρώματος της Υποστήριξης Υπηρεσίας, συνήθως, περιλαμβάνει γενική λειτουργικότητα για όλα τα υποκείμενα δίκτυα. Η διαχείριση κινητικότητας και η μορφή των διεπαφών για συγκεκριμένα δίκτυα πρόσβασης αποτελούν σημαντικά ζητήματα προς διερεύνηση.

### 1.2.8 Ανάγκη για Συνεργασία

Προβλέπεται ότι στο μέλλον οι διάφοροι εμπλεκόμενοι παράγοντες θα εξειδικευθούν σε διαφορετικές λειτουργικές περιοχές. Για παράδειγμα, θα υπάρχουν πάροχοι ταυτοτήτων, πάροχοι προφίλ, πάροχοι υπηρεσιών, κ.λπ.

Κάθε μία από αυτές τις λειτουργικές περιοχές θα απαιτεί επικοινωνία με άλλους παρόχους, είτε εντός είτε εκτός της ίδιας περιοχής. Κάτι τέτοιο θα είναι ωφέλιμο για και τον τελικό χρήστη και τον πάροχο υπηρεσιών. Οι χρήστες θα μπορούν να διαλέγουν τον πάροχο ταυτότητας της προτίμησής τους. Επίσης, θα υπάρχει διαλλειτουργικότητα μεταξύ των διαφόρων λειτουργικών μονάδων. Επομένως, η επικοινωνία και η συνεργασία είναι στοιχεία εκ των ουκ άνευ. Η πολλαπλότητα των παρόχων είναι πολύ χρήσιμη, εξαιτίας των μηχανισμών της ανοικτής αγοράς (ο ανταγωνισμός οδηγεί σε χαμηλότερες τιμές, σε καλύτερη ποιότητα, κ.λπ.). Ενώ, όμως, έχει διαπιστωθεί η ανάγκη για επικοινωνία μεταξύ των εμπλεκόμενων παραγόντων, αυτήν τη στιγμή δεν υπάρχει κάποια συμφωνία σχετικά με το πώς πρέπει να γίνεται. Ένας ενδιαφέρων τομέας έρευνας πάνω σε αυτό το θέμα είναι ο καθορισμός των ειδικών λειτουργιών που θα υλοποιούν αυτήν τη συνεργασία.

Συχνά αναφερόμαστε στο μηχανισμό επικοινωνίας και συνεργασίας που, όπως τονίσαμε, είναι απαραίτητο να υπάρχει με τον όρο «Ένωση» ή «Ομοσπονδία» (Federation), όπως εικονίζεται και στο σχήμα που ακολουθεί.



### 1.2.9 Σύνοψη

Στην παράγραφο 1.2, αναφερθήκαμε στις δικτυακές υποδομές και λειτουργίες ενός συστήματος 4G. Από την πλευρά του δικτύου, λοιπόν, το καινούριο στοιχείο στην αρχιτεκτονική ενός τέτοιου συστήματος είναι η ενσωμάτωση ενός ενδιάμεσου στρώματος που το αποκαλούμε Υποστήριξη Υπηρεσίας. Το στρώμα αυτό αναλαμβάνει την επιτέλεση αφενός μεν της λειτουργίας ενοποίησης δικτύων, αφετέρου δε της λειτουργίας συγκέντρωσης υπηρεσιών.

Χάρη στην πρώτη, επιτυγχάνεται η συνένωση των διαφόρων ετερογενών δικτύων σε ένα ενιαίο εικονικό δίκτυο, παρέχοντας προς την πλευρά των υπηρεσιών σαφώς καθορισμένες διεπαφές ανάκτησης πληροφοριών, αντί πολλαπλών ετερογενών διεπαφών.

Χάρη στη δεύτερη, ο χρήστης μπορεί να αποκτήσει πρόσβαση σε πολλές διαφορετικές υπηρεσίες, ποικίλων παρόχων, χρησιμοποιώντας μία και μόνο

ταυτότητα και λαμβάνοντας ένα μοναδικό λογαριασμό, αρκεί να γραφτεί συνδρομητής σε μία μονάδα συγκέντρωσης υπηρεσιών. Εκτός αυτού, οι πάροχοι υπηρεσιών μπορούν να βελτιώσουν τις προσφερόμενες υπηρεσίες, προσαρμόζοντάς τις στις ανάγκες και προτιμήσεις του χρήστη και στις συνθήκες του δικτύου, χάρη στις πληροφορίες με τις οποίες τροφοδοτούνται από τις μονάδες συγκέντρωσης υπηρεσιών με τις οποίες συνεργάζονται. Έτσι, λοιπόν, η συγκέντρωση υπηρεσιών ωφελεί τόσο το χρήστη όσο και τους παρόχους υπηρεσιών.

Με άλλα λόγια, μπορούμε να θεωρήσουμε τη μονάδα συγκέντρωσης υπηρεσιών ως μια οντότητα (οργανισμό) που έχει συνάψει συμφωνίες με διάφορους παρόχους υπηρεσιών. Έτσι, ένας συνδρομητής της μονάδας συγκέντρωσης μπορεί να χρησιμοποιήσει όσες από τις προσφερόμενες υπηρεσίες επιθυμεί και, μάλιστα, χωρίς να χρειάζεται πολλαπλή υποβολή των στοιχείων του για ταυτοποίηση. Παράλληλα, στην ταυτότητα που χρησιμοποιεί υπάρχουν ενσωματωμένες και πληροφορίες που αφορούν τις προτιμήσεις του (προφίλ), τις οποίες και μπορεί, ανά πάσα στιγμή, να επεξεργασθεί. Κάτι αντίστοιχο στο χώρο του Internet θα ήταν να μπορεί ο χρήστης να χρησιμοποιεί διάφορες υπηρεσίες, π.χ. e-mail, επίσκεψη σε ιστοσελίδες με συνδρομητικό περιεχόμενο, e-banking, κ.λπ., χρησιμοποιώντας για όλες την ίδια κοινή ταυτότητα και υποβάλλοντάς την μία και μοναδική φορά (κατά την πρώτη του απόπειρα σύνδεσης), εκτός αυτού δε οι εν λόγω υπηρεσίες να παρείχαν περιεχόμενο προσαρμοσμένο στις προτιμήσεις του.

### **1.3 Τερματικές Υποδομές και Λειτουργίες**

#### *1.3.1 Εισαγωγή – Δυνατότητες Σύγχρονων Τερματικών*

Στην προηγούμενη παράγραφο, εξετάσαμε τις απαραίτητες υποδομές και λειτουργίες που πρέπει να υλοποιηθούν στην πλευρά του δικτύου, προκειμένου να μπορεί να λειτουργήσει επιτυχώς ένα σύστημα 4G. Θα περάσουμε, τώρα, στην πλευρά του τερματικού, εξετάζοντας το ρόλο και τις λειτουργίες που θα πρέπει να αναλάβει σε ένα περιβάλλον 4G.

Τα σύγχρονα κινητά τερματικά είναι μικρά, αλλά ολοένα και πιο ισχυρά, ανοικτά υπολογιστικά συστήματα, τα οποία μπορούν να τρέξουν υπηρεσίες διαφόρων τύπων. Είναι εφοδιασμένα με ένα λειτουργικό σύστημα, όπως Windows CE, Windows XP Tablet PC, Symbian, Linux, κ.λπ., καθώς και με μία στοίβα IP (IP stack), η οποία επιτρέπει την επικοινωνία με άλλους χρήστες και υπηρεσίες, μέσω Internet. Επίσης, το κινητό τερματικό μπορεί να είναι εφοδιασμένο με πολλές διεπαφές δικτύου, έτσι ώστε να μπορεί να συνδέεται χρησιμοποιώντας διαφορετικούς τύπους δικτύων πρόσβασης. Οι αυξημένες αυτές δυνατότητες του τερματικού (υπολογιστική ισχύς, μνήμη, κ.λπ.) έχουν εισαγάγει την ιδέα επέκτασης της ευφυΐας από το δίκτυο στο τερματικό.

Η ιδέα αυτή δεν είναι πρωτόγνωρη. Οι τηλεφωνικές συσκευές, για παράδειγμα, έχουν βελτιωθεί με την προσθήκη μνήμης, που επιτρέπει την αποθήκευση συχνά καλούμενων αριθμών, την κλήση του τελευταίου αριθμού (redial), κ.λπ. Πρόσφατα, και άλλες λειτουργίες έχουν προστεθεί στα κινητά τερματικά, όπως είναι η ενσωμάτωση προσωπικού τηλεφωνητή, web browser και διακομιστή e-mail.

Η εξέλιξη αυτή, όμως, δεν υπονοεί απαραίτητως ότι η ευφυΐα θα μετακινηθεί από το δίκτυο στο τερματικό. Αντίθετα, κάποιες υπηρεσίες μπορούν να υλοποιηθούν μόνο με το συνδυασμό της ευφυΐας τερματικού και δικτύου. Ένα απλό παράδειγμα της περίπτωσης αυτής είναι η υπηρεσία αναγνώρισης κλήσης, η οποία προϋποθέτει αφενός τη μεταφορά από το δίκτυο του καλούντος αριθμού, μέσω σηματοδοσίας, και αφετέρου τον εξοπλισμό του τερματικού με κάποιου είδους λογική και με μια μικρή οθόνη, προκειμένου να μεταφράσει αυτήν την πληροφορία και να την παρουσιάσει στο χρήστη. Με άλλα λόγια, η ευφυΐα των τερματικών δεν αντικαθιστά αλλά είναι συμπληρωματική εκείνης του δικτύου.

### *1.3.2 Η Επιλογή Δικτύου Πρόσβασης ως Ρόλος του Τερματικού*

Στα πλαίσια αυτής της επέκτασης της ευφυΐας από το δίκτυο στην πλευρά του τερματικού, ολοένα και περισσότερο σύνθετες εφαρμογές και καθήκοντα ανατίθενται στο τερματικό. Σε ένα 4G περιβάλλον, ένα από τα σημαντικότερα ζητήματα είναι η επιλογή δικτύου πρόσβασης.

Μέχρι τώρα, στα συστήματα 2<sup>ης</sup> και 3<sup>ης</sup> Γενιάς οι χρήστες σπάνια ενδιαφέρονται για την επιλογή δικτύων που ανήκουν σε διαφορετικούς παρόχους, και χρησιμοποιούν την προκαθορισμένη «αυτόματη επιλογή» του τερματικού τους. Συνεπώς, οι χρήστες συχνά καταλήγουν (π.χ. σε κατάσταση περιαγωγής) σε ένα ακριβό GSM ή UMTS δίκτυο, εξαιτίας αυτής της αυτόματης επιλογής. Οι πάροχοι των δικτύων μπορούν, έμμεσα ή άμεσα, να καθορίσουν τις προεπιλεγμένες ρυθμίσεις των τερματικών σχετικά με την επιλογή δικτύου και τις παραμέτρους μιας διαπομπής, «χειραγωγώντας» κατά κάποιον τρόπο τους συνδρομητές τους.

Ωστόσο, σε ένα αμιγώς ετερογενές περιβάλλον, με πλήθος εναλλακτικών επιλογών, όπως αυτό της 4<sup>ης</sup> Γενιάς, η επιλογή δικτύου πρόσβασης είναι προτιμότερο να μην παραμείνει αποκλειστικά στην πλευρά του δικτύου, αλλά να μεταβεί κυρίως στην πλευρά του τερματικού. Με τις ολοένα αυξανόμενες δυνατότητες των σύγχρονων τερματικών, κάτι τέτοιο είναι απόλυτα εφικτό.

Τα πλεονεκτήματα αυτής της μετάβασης είναι πολλά και εστιάζονται τόσο σε τεχνικούς όσο και σε εμπορικούς λόγους. Οι λόγοι αυτοί θα αναλυθούν αμέσως πιο κάτω.

- Σε ένα ετερογενές περιβάλλον, με πολλά υποστηριζόμενα δίκτυα πρόσβασης, οι πληροφορίες για τα δίκτυα μεταβάλλονται με γρήγορο ρυθμό. Τα κινητά τερματικά, όντας υπεύθυνα για το συνεχή έλεγχο εμφάνισης και εξαφάνισης δικτύων, έχουν άμεση πρόσβαση στα χαρακτηριστικά κάθε δικτύου, όπως είναι η ισχύς σήματος, το διαθέσιμο εύρος ζώνης, κ.λπ.
- Τα κινητά τερματικά γνωρίζουν, φυσικά, ακριβώς πόσα και ποια δικτυακά interfaces και άρα ποιες τεχνολογίες πρόσβασης μπορούν να αξιοποιήσουν και σε ποια δίκτυα μπορούν να συνδεθούν.
- Επιπρόσθετα, ένα τερματικό γνωρίζει πόσες και ποιες υπηρεσίες ήδη τρέχουν ή μπορούν να τρέξουν σε αυτό.

- Ο χρήστης είναι, προφανώς, σε άμεση επαφή με το τερματικό και άρα οι προτιμήσεις του μπορούν να αποθηκευτούν και να γνωστοποιηθούν πιο εύκολα σε αυτό.
- Ο χρήστης αποκτά αυτονομία έναντι του δικτύου και δε «χειραγωγείται», αλλά αποφασίζει ο ίδιος για αυτό που πιστεύει ότι είναι συμφέρον.

Ας δούμε, όμως, πώς θα αντιμετώπιζαν οι πάροχοι δικτύου μια τέτοια μετάβαση:

- Από τη μια πλευρά, η επιλογή του δικτύου πρόσβασης από το τερματικό θα τους απάλλασσε από ένα σημαντικό υπολογιστικό φορτίο, που θα αφορούσε την ενημέρωση για την τρέχουσα κατάσταση των δικτύων, τις προτιμήσεις του χρήστη και τις δυνατότητες του τερματικού. Επίσης, για τις περιπτώσεις μετάβασης σε δίκτυο διαφορετικού παρόχου, περιορίζει την ανάγκη ύπαρξης ειδικών συμφωνιών μεταξύ των διαφορετικών παρόχων δικτύου ή ύπαρξης μιας κοινής διαχειριστικής αρχής που να αναλαμβάνει, με ακέραιο τρόπο, την επιβολή τέτοιου είδους μεταβάσεων (διαπομπών).
- Από την άλλη πλευρά, οι πάροχοι θα επιθυμούσαν να διατηρήσουν τον έλεγχο της διαδικασίας επιλογής, προκειμένου να μην έχουν διαρροές από την πελατειακή τους βάση. Μία ανοικτή αγορά, όπως αυτή ενός 4G συστήματος, θα επέφερε συνθήκες ανταγωνισμού που, αν και προς το συμφέρον του χρήστη, θα ήταν ελάχιστα επιθυμητές από τους παρόχους. Επιπλέον, οι επιλογές των τερματικών ίσως να μην εξισορροπούν κατά το βέλτιστο δυνατό τρόπο ή τουλάχιστον κατά τον τρόπο που επιθυμούν αυτοί το φορτίο του δικτύου τους.

Παρόλο που η ανάθεση της επιλογής δικτύου πρόσβασης στην πλευρά του τερματικού παρέχει ένα σημαντικό βαθμό ελευθερίας, εντούτοις οι πάροχοι δικτύου θα εξακολουθήσουν να θέλουν να έχουν κάποιον έλεγχο επί των τερματικών. Μπορεί η εποχή του πλήρους ελέγχου, την οποία διανύουμε, να λήξει με την είσοδο των συστημάτων 4G, ωστόσο κάποια μορφή ελέγχου από την πλευρά του δικτύου θα συνεχίσει να υπάρχει. Ο έλεγχος αυτός θα επιτευχθεί, πιθανότατα, με δύο τρόπους:

- Πρώτον, μέσω των προεπιλεγμένων ρυθμίσεων των τερματικών. Ένα σημαντικό ποσοστό των χρηστών θα συνεχίσει να επαφίεται στις προεγκατεστημένες ρυθμίσεις των συσκευών που χρησιμοποιεί. Εξάλλου, οι ρυθμίσεις αυτές παρέχουν ένα γρήγορο και σίγουρο τρόπο για να κάνουν τις συσκευές τους να δουλεύουν, έστω και αν δεν καλύπτουν απόλυτα τις ανάγκες τους.
- Δεύτερον, μέσω μιας διαδικασίας έγκρισης της επιλογής του τερματικού. Με άλλα λόγια, η επιλογή του τερματικού θα υποβάλλεται πρώτα για έγκριση στον πάροχο δικτύου που έχει επιλεγεί, προτού εγκατασταθεί η αντίστοιχη σύνδεση. Σε περίπτωση απόρριψης, η δεύτερη καλύτερη επιλογή θα προωθείται για έγκριση.

Η ανάθεση της επιλογής δικτύου πρόσβασης στο τερματικό εγείρει, όπως διαφαίνεται και από τα παραπάνω, ορισμένα σημαντικά ζητήματα και απαιτήσεις:

- Η διαδικασία της επιλογής πρέπει να είναι όσο το δυνατό πιο διαφανής προς το χρήστη, χωρίς, δηλαδή, να χρειάζεται τη διαρκή προσοχή και παρέμβασή του. Έτσι, ο χρήστης θα μπορεί να απολαμβάνει ανενόχλητος τις υπηρεσίες που επιθυμεί.

- Ταυτόχρονα, η διαδικασία αυτή πρέπει να καταλήγει σε ένα αποτέλεσμα που να είναι πάντα προς το συμφέρον του χρήστη, ικανοποιώντας στο ακέραιο τις ανάγκες και προτιμήσεις του.
- Οι προτιμήσεις του χρήστη πρέπει να γνωστοποιηθούν στο κινητό τερματικό. Ο χρήστης θα πρέπει να μπορεί να καθορίζει τις προτιμήσεις του μέσω ενός εύχρηστου τρόπου, που να μην του καταναλώνει πολύ χρόνο. Διαφορετικά, υπάρχει ο «κίνδυνος» η πλειοψηφία των χρηστών να αφήνουν τις προεπιλεγμένες από τους παρόχους ρυθμίσεις και μόνο μια μικρή μειοψηφία να εκμεταλλεύεται πλήρως τις δυνατότητες της συσκευής του. Προφανώς, ο καθορισμός ενός πολύπλοκου συνόλου κανόνων που να ρυθμίζουν τις μεταβάσεις (διαπομπές) δεν είναι και ο καλύτερος τρόπος για να επιτευχθεί κάτι τέτοιο, αφού είναι και σύνθετος και χρονοβόρος. Αντίθετα, η χρησιμοποίηση μιας εύχρηστης και χωρίς ασάφειες γραφικής διεπαφής (Graphical User Interface – GUI) είναι η ενδεδειγμένη λύση.

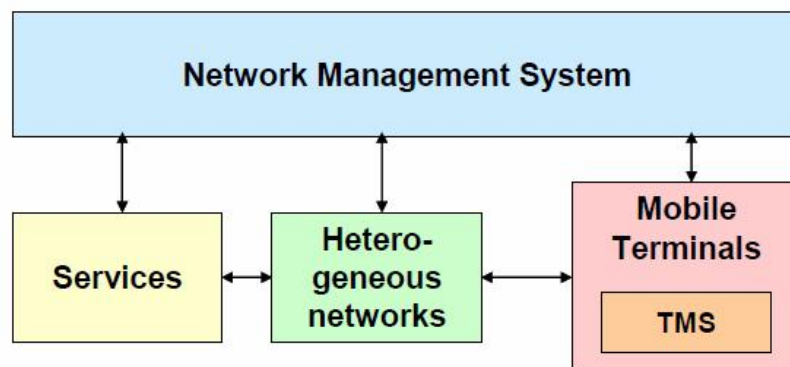
## 1.4 Αντικείμενο της Διπλωτικής

### 1.4.1 Εισαγωγή

Στην παρούσα εργασία, θα προχωρήσουμε στην υλοποίηση μιας διαδικασίας επιλογής δικτύου πρόσβασης, από την πλευρά του τερματικού, που να ικανοποιεί όλες τις παραπάνω απαιτήσεις. Θα αναφερόμαστε στη λειτουργία που θα υλοποιήσουμε με τον όρο «ευφυής επιλογή δικτύου (ή σημείου) πρόσβασης». Βασικό της χαρακτηριστικό είναι ότι η επιλογή εξαρτάται όχι μόνο από τα χαρακτηριστικά του δικτύου αλλά και τις προτιμήσεις του χρήστη.

Η ευφυής επιλογή δικτύου πρόσβασης αποτελεί συστατικό του συστήματος διαχείρισης (Terminal Management System – TMS) ενός τερματικού 4<sup>ης</sup> Γενιάς. Στηρίζεται πάνω στην αρχή ABC, που περιγράφηκε εκτενώς στις παραγράφους 1.1.4 έως 1.1.6.

Σε μία σύνθετη δικτυακή υποδομή, τα βασικά συστατικά που συναντώνται είναι αυτά που εικονίζονται στο παρακάτω σχήμα.



Σε ένα 4G σύστημα, αναβαθμισμένο ρόλο διαδραματίζει το σύστημα διαχείρισης τερματικού (TMS), το οποίο θα είναι καινοτόμο, με την έννοια ότι θα ενσωματώνει τη λειτουργία ευφυούς επιλογής δικτύου πρόσβασης.

Στην υλοποίηση που προτείνεται, δύο είναι οι βασικές συνιστώσες του συστήματος διαχείρισης τερματικού στις οποίες πρέπει να αποδοθεί ιδιαίτερη βαρύτητα: η διαδικασία μέσω της οποίας φτάνει το τερματικό στην επιλογή του κατάλληλου δικτύου πρόσβασης και η διεπαφή μέσω της οποίας δίνεται στο χρήστη η δυνατότητα να εισάγει τις προτιμήσεις του. Οι δύο αυτές βασικές συνιστώσες παρουσιάζονται περιληπτικά στις παραγράφους που ακολουθούν.

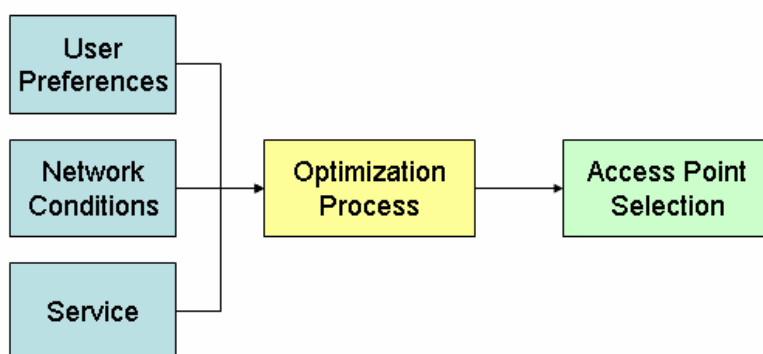
#### 1.4.2 Ευφυής Επιλογή Δικτύου Πρόσβασης

Κεντρικός σκοπός της παρούσας εργασίας είναι η ανάπτυξη ενός αλγορίθμου ο οποίος θα καταλήγει στην επιλογή του κατάλληλου δικτύου και σημείου πρόσβασης μέσω του οποίου θα εξυπηρετηθεί μια υπηρεσία. Με τον όρο «κατάλληλο», εννοούμε ότι πρέπει να καταλήγει στην πιο συμφέρουσα για το χρήστη λύση, λαμβάνοντας υπόψη τις προτιμήσεις του.

Ο αλγόριθμος αυτός λαμβάνει ως είσοδο τις τρέχουσες συνθήκες του δικτύου, όπως αυτές τις αντιλαμβάνεται το τερματικό, βάσει των μετρήσεων του, τις προτιμήσεις του χρήστη, τις οποίες ο τελευταίος έχει φροντίσει να εισάγει, και το είδος της υπηρεσίας για την οποία πρέπει να βρει τη βέλτιστη επιλογή ανάθεσης.

Ως έξοδο ο αλγόριθμος αυτός δεν παράγει μόνο τη βέλτιστη επιλογή όσον αφορά το δίκτυο και το σημείο πρόσβασης, αλλά και τη βέλτιστη επιλογή όσον αφορά το επίπεδο ποιότητας στο οποίο πρέπει να διατεθεί μία συγκεκριμένη υπηρεσία (η υπηρεσία εισόδου).

Ο συγκεκριμένος αλγόριθμος θα πρέπει να ενεργοποιείται και να εκτελείται, όποτε ο χρήστης εκκινεί μια νέα υπηρεσία ή τροποποιεί τις προτιμήσεις του, αλλά και όποτε συμβεί σημαντική μείωση της ισχύς του λαμβανόμενου σήματος από ένα χρησιμοποιούμενο σημείο πρόσβασης ή εμφανιστεί ένα νέο σημείο πρόσβασης.





### *1.4.3 Δυναμική Προσαρμογή Προφίλ Χρήστη*

Η παροχή στο χρήστη της δυνατότητας εισαγωγής των προτιμήσεών του υλοποιείται μέσω μιας εύχρηστης γραφικής διεπαφής. Η διεπαφή αυτή επιτρέπει, όπως θα δούμε, τον καθορισμό τόσο κάποιων γενικών ρυθμίσεων όσο και ρυθμίσεων ανά υπηρεσία. Βασικό της στοιχείο είναι ότι ο χρήστης μπορεί να έχει, σε μικρό χρόνο, μια εποπτική εικόνα των τρεχουσών ρυθμίσεών του και, επιπλέον, ότι μπορεί να τις τροποποιήσει ανά πάσα στιγμή, εύκολα και γρήγορα. Οι ρυθμίσεις του αυτές αποθηκεύονται στο τερματικό και δε χάνονται, ακόμα και αν ο χρήστης κλείσει συσκευή.

### *1.4.4 Διάρθρωση της Εργασίας*

Στο κεφάλαιο 1 της παρούσας εργασίας, δόθηκε, αρχικά, μια περιγραφή και ιστορική εξέλιξη των σημαντικότερων συστημάτων ασύρματων τηλεπικοινωνιών. Παρουσιάστηκε, λεπτομερώς, η ιδέα ABC και, εν συνεχεία, η απαραίτητη δικτυακή υποδομή που θα την υποστηρίξει. Κατόπιν, εξετάστηκε η δυνατότητα του τερματικού να αναλάβει τη διαδικασία επιλογής δικτύου πρόσβασης και έγινε μία εισαγωγή στη λειτουργία του αλγορίθμου που αποτελεί τον πυρήνα της εργασίας.

Στο κεφάλαιο 2, θα παρουσιαστεί με περισσότερες λεπτομέρειες η αρχιτεκτονική διαχείρισης ενός 4G τερματικού, στην οποία και θα ενσωματώνεται η λειτουργία της ευφυούς επιλογής δικτύου πρόσβασης. Θα δοθεί έμφαση στη δομή της αρχιτεκτονικής, με ταυτόχρονη περιγραφή των σημαντικότερων συνιστωσών που την απαρτίζουν.

Στο κεφάλαιο 3, θα προβούμε σε μια αναλυτική περιγραφή της συνιστώσας προτιμήσεων χρήστη, της γραφικής διεπαφής, δηλαδή, μέσω της οποίας ο χρήστης θα μπορεί να καθορίζει με εύκολο και αποτελεσματικό τρόπο τις προτιμήσεις του σχετικά με τον τρόπο σύνδεσης.

Στο κεφάλαιο 4, θα προχωρήσουμε στην αναλυτική παρουσίαση της προτεινόμενης υλοποίησης για τη διαδικασία επιλογής δικτύου και σημείου πρόσβασης από την πλευρά του τερματικού.

Στο κεφάλαιο 5, θα παρουσιαστούν ορισμένα ειδικά θέματα και τεχνικές λεπτομέρειες του υλοποιηθέντος συστήματος διαχείρισης τερματικού.

Στο κεφάλαιο 6, θα παρουσιαστούν τα αποτελέσματα εκτέλεσης του προτεινόμενου αλγορίθμου, στα πλαίσια αντιπροσωπευτικών σεναρίων.

Στο κεφάλαιο 7, θα γίνει η ανακεφαλαίωση και η εξαγωγή των τελικών συμπερασμάτων.

Τέλος, στο παράρτημα, στο τέλος της εργασίας, μπορεί να βρεθεί ο πηγαίος κώδικας και το documentation του συστήματος που υλοποιήθηκε.



## 2. Αρχιτεκτονική Διαχείρισης Τερματικού

### 2.1 Επισκόπηση Αρχιτεκτονικής

#### 2.1.1 Η Συνεργασία Δικτύου και Τερματικών

Όπως έχει γίνει, ίσως, φανερό από το προηγούμενο κεφάλαιο, η εύρυθμη λειτουργία ενός 4G συστήματος μπορεί να επιτευχθεί μόνο με τη συνδυασμένη συνεισφορά και των δύο πλευρών: του δικτύου, από τη μια, και του τερματικού, από την άλλη.

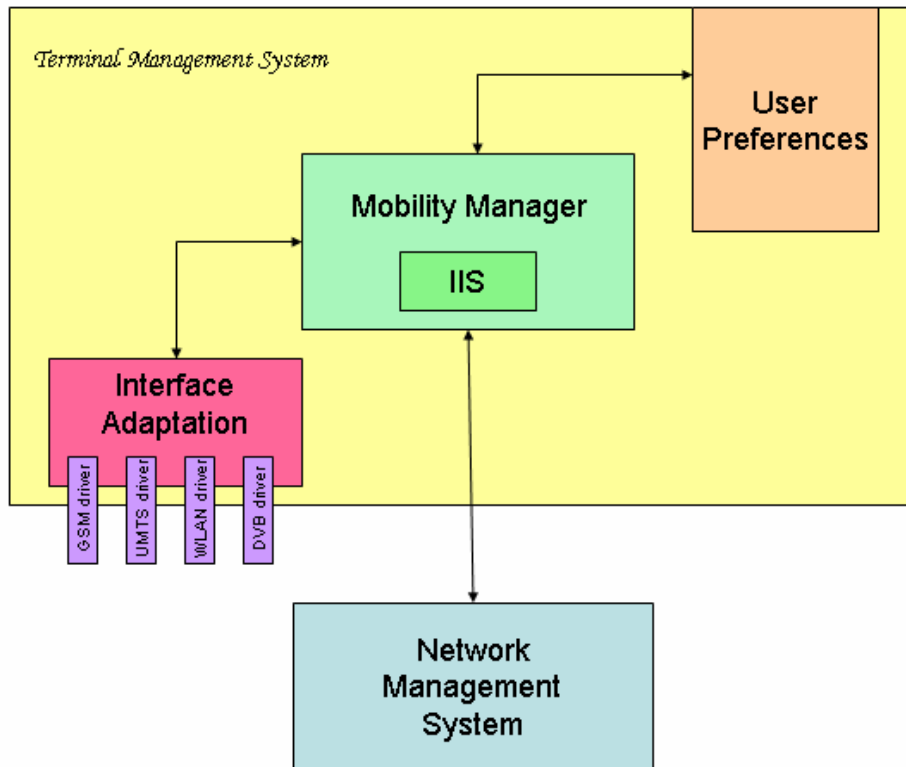
Από την πλευρά του δικτύου, η οντότητα διαχείρισης (βλ. και παράγραφο 1.2, ειδικά τη λειτουργία «Ενοποίησης Δικτύων») θα εκτελεί εργασίες όπως η εξισορρόπηση φορτίου μεταξύ των διαφόρων ραδιοδιαύλων, αλλά και ο γενικός συντονισμός. Ο κύριος στόχος της θα είναι η βέλτιστη διαχείριση των δικτυακών πόρων. Ειδικότερα, μια οντότητα διαχείρισης δικτύου θα πρέπει να έχει τη δυνατότητα της παρακολούθησης και ανάλυσης της απόδοσης και των προσφερόμενων επιπέδων ποιότητας των διαφόρων στοιχείων του δικτύου, της δυναμικής εύρεσης και επιβολής της βέλτιστης κατανομής της κίνησης, αλλά και της συνεργασίας με τα κινητά τερματικών των χρηστών.

Από την πλευρά του τερματικού, το σύστημα διαχείρισης θα αλληλεπιδρά με την προαναφερθείσα οντότητα διαχείρισης δικτύου, με σκοπό τη βέλτιστη επιλογή δικτύου πρόσβασης. Αυτή η συνεργασία εξασφαλίζει ότι οι μετρήσεις και υπολογισμοί του δικτύου και του τερματικού, όσον αφορά τις συνθήκες και τα προσφερόμενα επίπεδα ποιότητας στο δίκτυο, συνδυάζονται εποικοδομητικά, προκειμένου να ληφθεί η καλύτερη δυνατή απόφαση για το σημείο πρόσβασης μέσω του οποίου πρέπει να ληφθεί μια υπηρεσία. Συνεπώς, τόσο το δίκτυο όσο και τα τερματικά θα συνεισφέρουν «ευφυΐα» για την επίτευξη της βέλτιστης λειτουργίας του 4G συστήματος.

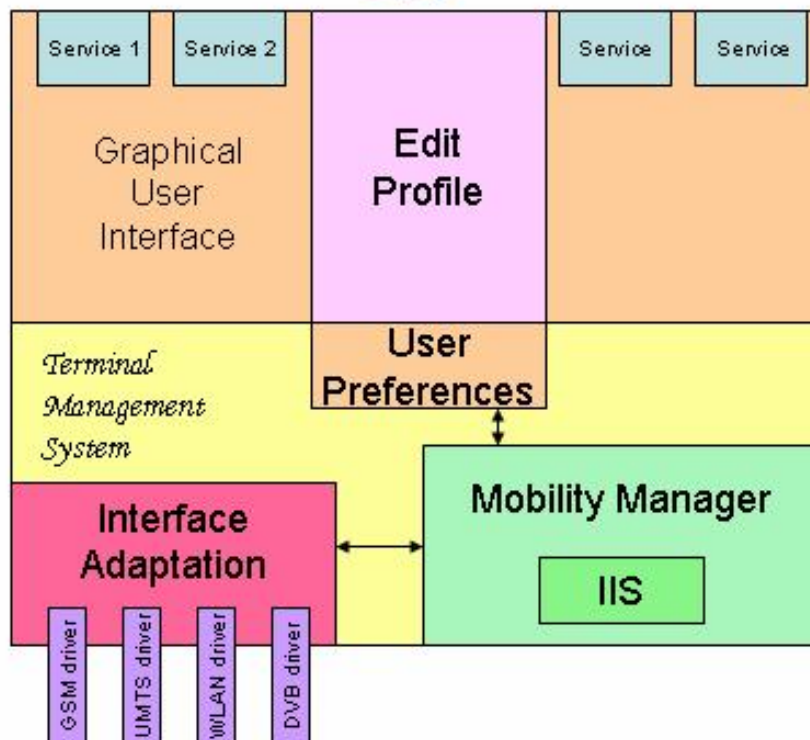
#### 2.1.2 Βασικές Συνιστώσες της Αρχιτεκτονικής Διαχείρισης Τερματικού

Στο επόμενο σχήμα, εικονίζεται η αρχιτεκτονική διαχείρισης ενός 4G τερματικού. Από το σχήμα αυτό, ξεχωρίζουμε τα βασικά συστατικά στοιχεία (modules) που απαρτίζουν ένα Σύστημα Διαχείρισης Τερματικού (Terminal Management System – TMS):

- Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου (Interface Adaptation module – IA)
- Συνιστώσα Προτιμήσεων Χρήστη (User Preferences module – UP)
- Συνιστώσα Διαχείρισης Κινητικότητας (Mobility Manager module – MM), μαζί με τη Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης (Intelligent Interface Selection sub-module – IIS)



Το επόμενο σχήμα είναι παρόμοιο με το προηγούμενο, με τη διαφορά ότι τονίζεται η δυνατότητα του χρήστη, μέσω της γραφικής διεπαφής του τερματικού (GUI), να διαχειριστεί όχι μόνο τις υπηρεσίες αλλά και το προφίλ του.



### 2.1.3 Η Υλοποίηση του Συστήματος Διαχείρισης Τερματικού

Η παρούσα εργασία συμπεριλαμβάνει μια υλοποίηση σε Java του συστήματος διαχείρισης ενός τερματικού. Υλοποιούνται όλα τα modules των παραπάνω σχημάτων, ιδιαίτερη βαρύτητα, όμως, δίνεται στη Συνιστώσα Προτιμήσεων Χρήστη και στη Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης, οι οποίες και θα περιγραφούν πιο αναλυτικά από τις υπόλοιπες συνιστώσες.

Σκοπός είναι η υλοποίηση και παρουσίαση ενός συστήματος που να δίνει την αίσθηση ενός ολοκληρωμένου οικοδομήματος, ικανού να τρέξει ολοκληρωμένες προσομοιώσεις. Όπως θα δούμε, ακόμη, η γραφική διεπαφή που σχεδιάστηκε για το σύστημα παραπέμπει σε αυτήν ενός σύγχρονου κινητού τερματικού, ώστε η υλοποίηση να μη στερείται ρεαλισμού.

Η χρήση της Java ως γλώσσας υλοποίησης του συστήματος είναι η πλέον κατάλληλη. Η Java παρέχει ένα ελκυστικό περιβάλλον για την κατασκευή κινητών και καταναμημένων εφαρμογών. Προσφέρει μεταφερσιμότητα, μέσω της λεγόμενης Εικονικής Μηχανής Java (Java Virtual Machine – JVM), η οποία εγκαθίσταται πάνω από τις πραγματικές αρχιτεκτονικές. Με αυτόν τον τρόπο, οι εφαρμογές Java καθίστανται ανεξάρτητες της πλατφόρμας του συστήματος στο οποίο τρέχουν, αφού η JVM φροντίζει για τα θέματα συμβατότητας και μεταφερσιμότητας. Κατά συνέπεια, η επιλογή της Java ως γλώσσας υλοποίησης του συστήματος διαχείρισης ενός τερματικού είναι μια φυσική επιλογή, αφού έτσι η υλοποίηση του συστήματος καθίσταται ανεξάρτητη τόσο του είδους του τερματικού όσο και του λειτουργικού συστήματος που χρησιμοποιεί.

## 2.2 Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου

### 2.2.1 Ρόλος της Συνιστώσας

Ο ρόλος της Συνιστώσας Προσαρμογής στους Οδηγούς Δικτύου (Interface Adaptation module – IA) είναι να παρέχει ένα επίπεδο αφαίρεσης προς τη μεριά του TMS, πράγμα που σημαίνει ότι αποκρύπτει λεπτομέρειες αναφορικά με τις κάρτες δικτύου του τερματικού, παρέχοντας μια απλή και σαφώς καθορισμένη διεπαφή προς το υπόλοιπο TMS. Ταυτόχρονα, προσφέρει λειτουργικότητα Επιπέδου 2 (L2) στο κινητό τερματικό.

Πιο συγκεκριμένα, το IA module είναι υπεύθυνο για την αναγνώριση των διαφόρων καρτών δικτύου που είναι ενσωματωμένες στο κινητό τερματικό, την παρακολούθηση της κατάστασής τους και τη συγκέντρωση των μετρήσεων από κάθε μία κάρτα – διεπαφή.

### 2.2.2 Περιγραφή της Συνιστώσας – Πληροφορίες των Μετρήσεων

Η λειτουργία της συγκέντρωσης των μετρήσεων από τις κάρτες δικτύου είναι και η σημαντικότερη. Οι μετρήσεις αυτές (L2 measurements) προέρχονται από τους οδηγούς (drivers) των καρτών δικτύου με τις οποίες είναι εξοπλισμένο το τερματικό και αντανακλούν τις συνθήκες του δικτυακού περιβάλλοντος στο οποίο βρίσκεται. Το IA module συγκεντρώνει τις μετρήσεις και τις μετασχηματίζει σε έναν πίνακα πληροφοριών.

Έτσι, διευκολύνεται η άμεση πρόσβαση στις συγκεκριμένες πληροφορίες και η ευκολότερη επεξεργασία τους από το Mobility Manager module. Κατ' αυτόν τον τρόπο, το IA module παρέχει προς την πλευρά του Mobility Manager module (και, ειδικότερα, του IIS sub-module) μια εύχρηστη και απλή διεπαφή, αποκρύπτοντας τον τρόπο με τον οποίο οι μετρήσεις έχουν ληφθεί και οργανωθεί.

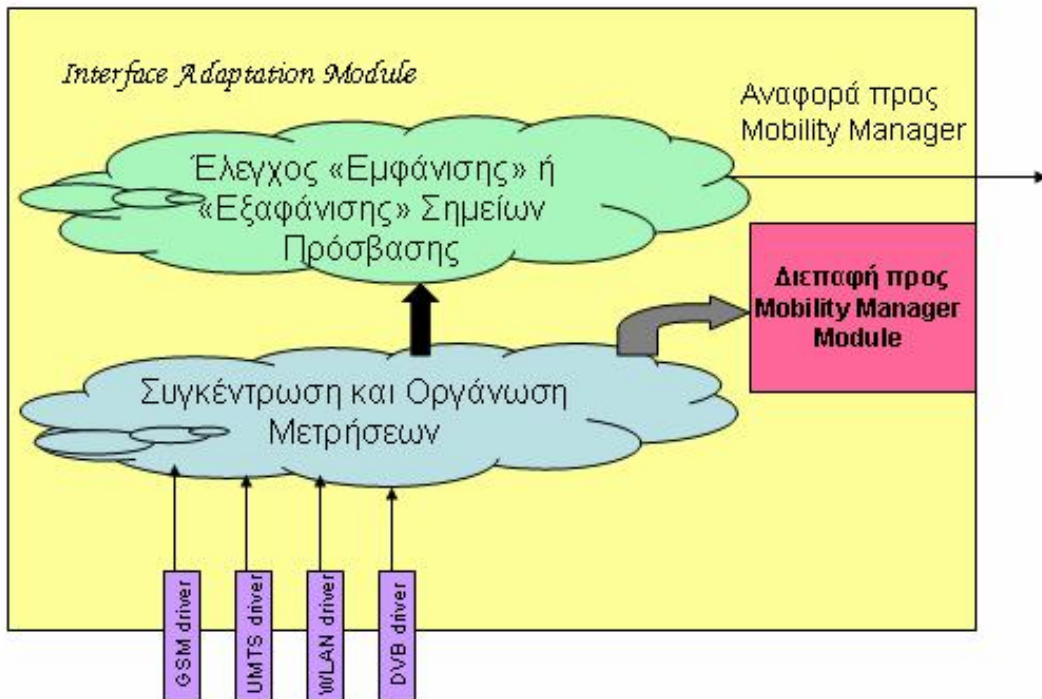
Για κάθε σημείο πρόσβασης που βρίσκεται στην εμβέλεια του κινητού τερματικού, είναι διαθέσιμες, λοιπόν, χάρη στο IA module, οι εξής πληροφορίες:

- IP διεύθυνση (IP Address) του σημείου πρόσβασης.
- Πάροχος δικτύου (Provider) στον οποίο ανήκει το σημείο πρόσβασης.
- Τύπος τεχνολογίας (Access Technology) του σημείου πρόσβασης.
- Ποιότητα του σήματος (Signal Strength) που λαμβάνεται από το σημείο πρόσβασης, σε μια αφηρημένη μορφή (π.χ. κλίμακα από 1 έως 5).
- Διαθέσιμο εύρος ζώνης (Available Bandwidth) στο σημείο πρόσβασης (σε Kbps).

Το σύνολο των παραπάνω πληροφοριών χρησιμεύει ως είσοδος στον αλγόριθμο της ευφυούς επιλογής δικτύου πρόσβασης (IIS module).

Αξίζει να σημειωθεί ότι, εκτός από τη συγκέντρωση των μετρήσεων από τους οδηγούς των καρτών δικτύου και την οργάνωσή τους, σημαντικό καθήκον του IA module είναι και η ειδοποίηση του Mobility Manager module για την «εμφάνιση» ή «εξαφάνιση» (σημαντική υποβάθμιση) ενός σημείου πρόσβασης. Ένα τέτοιο γεγονός είναι σημαντικό για το σύστημα διαχείρισης του τερματικού, διότι οδηγεί, όπως θα δούμε και στο κεφάλαιο 4, στην ενεργοποίηση του αλγορίθμου ευφυούς επιλογής δικτύου πρόσβασης.

Σύμφωνα με τα παραπάνω, μια εποπτική εικόνα του IA module είναι αυτή που ακολουθεί.



### 2.2.3 Πρωτόκολλα Ανακάλυψης Δυνατοτήτων Γειτονικών Access Routers

Η περιγραφή που ακολουθεί εξετάζει τις λειτουργίες ενός συγκεκριμένου CARD πρωτοκόλλου, ενός πρωτοκόλλου, δηλαδή, που ανήκει στην οικογένεια πρωτοκόλλων ανίχνευσης υποψηφίων σημείων πρόσβασης. Σκοπός της περιγραφής αυτής δεν είναι η εκτενής παρουσίαση του συγκεκριμένου πρωτοκόλλου, αλλά η αναφορά στους βασικούς μηχανισμούς που διέπουν τη λειτουργία της προαναφερθείσας οικογένειας πρωτοκόλλων. Τονίζεται, λοιπόν, ότι στην αρχιτεκτονική του συστήματος διαχείρισης τερματικού το οποίο προτείνεται μπορεί να χρησιμοποιηθεί, εναλλακτικά, οποιοδήποτε πρωτόκολλο το οποίο είναι σε θέση να επιτελέσει παρόμοια λειτουργία.

- Εισαγωγή:

Προκειμένου να γίνει εφικτή η διαφανής διαπομπή, σε επίπεδο IP, ενός κινητού τερματικού από ένα σημείο πρόσβασης σε ένα άλλο, το τερματικό απαιτείται να ανακαλύψει τα χαρακτηριστικά και τις δυνατότητες των υποψηφίων σημείων πρόσβασης, στοιχεία απαραίτητα για την πραγματοποίηση της διαπομπής. Η διαδικασία αυτή της ανακάλυψης έχει δύο σκέλη: πρώτον, την αναγνώριση των IP διευθύνσεων των υποψηφίων σημείων πρόσβασης και, δεύτερον, τον καθορισμό των δυνατοτήτων κάθε τέτοιου σημείου. Η διαδικασία αυτή αναφέρεται ως «Candidate Access Router Discovery» (CARD). Τη στιγμή της διαπομπής, το σημείο πρόσβασης, οι δυνατότητες του οποίου ταιριάζουν καλύτερα με τις προτιμήσεις του κινητού τερματικού, γίνεται ο προορισμός της διαδικασίας διαπομπής. Το πρωτόκολλο που θα περιγραφεί στη συνέχεια επιτρέπει σε ένα κινητό τερματικό να εκτελέσει την αναζήτηση υποψηφίων σημείων πρόσβασης (CARD).

- Λειτουργίες του CARD Πρωτοκόλλου:

Το CARD πρωτόκολλο υλοποιεί τις ακόλουθες λειτουργίες:

1. Αντίστροφη Μετάφραση Διευθύνσεων (Reverse Address Translation):

Αν ένα κινητό τερματικό λαμβάνει L2 μετρήσεις από νέα σημεία πρόσβασης, πριν τη λήψη απόφασης για IP επιπέδου διαπομπή, είναι απαραίτητη η λειτουργία ενός μηχανισμού Αντίστροφης Μετάφρασης Διευθύνσεων. Η λειτουργία αυτή του CARD πρωτοκόλλου επιτρέπει στο κινητό τερματικό την αντιστοίχιση του L2 αναγνωριστικού (πιθανόν MAC διεύθυνση) του υποψήφιου σημείου πρόσβασης με την IP διεύθυνσή του. Προκειμένου να ανακτήσει την IP διεύθυνση, το κινητό τερματικό στέλνει το L2 αναγνωριστικό στο τρέχον σημείο πρόσβασης και αυτό, με τη σειρά του, ενημερώνει το κινητό τερματικό για την IP διεύθυνση του υποψήφιου σημείου πρόσβασης. Σε περιπτώσεις όπου το κινητό τερματικό μπορεί να αποκτήσει IP συνδεσιμότητα με τα υποψήφια σημεία πρόσβασης πριν τη λήψη της απόφασης διαπομπής, η λειτουργία αυτή καθίσταται ιδιαίτερα απλή, αφού το κινητό τερματικό μπορεί να ζητήσει ενημέρωση από τα ίδια τα σημεία πρόσβασης για τις δυνατότητες τους.

2. Ανακάλυψη των Δυνατοτήτων των Υποψηφίων Σημείων Πρόσβασης:

Οι πληροφορίες για τις δυνατότητες των υποψηφίων σημείων πρόσβασης (π.χ. διαθέσιμο bandwidth) βοηθούν, σε μεγάλο βαθμό, το κινητό τερματικό στη λήψη μιας βέλτιστης απόφασης διαπομπής. Οι πληροφορίες αυτές δίνονται ως είσοδο σε έναν αλγόριθμο βελτιστοποίησης, ο οποίος και καταλήγει στην τελική απόφαση διαπομπής. Κάποια από τα ιδιαίτερα χαρακτηριστικά κάθε σημείου πρόσβασης μπορεί να παραμένουν σταθερά, ενώ άλλα μπορεί να μεταβάλλονται με την πάροδο του χρόνου.

Υπάρχουν δύο τρόποι προσέγγισης που μπορεί να ακολουθεί το κινητό τερματικό, προκειμένου να ανακτήσει τη διεύθυνση και τις παραμέτρους των υποψηφίων σημείων πρόσβασης. Σύμφωνα με τον πρώτο τρόπο, το τερματικό στέλνει μια απλή αίτηση στο τρέχον σημείο πρόσβασης, το οποίο με τη σειρά του το ενημερώνει για τη διεύθυνση και τα χαρακτηριστικά του υπό εξέταση σημείου πρόσβασης. Κατά το δεύτερο τρόπο, το τρέχον σημείο πρόσβασης, περιοδικά, ενημερώνει όλα τα κινητά τερματικά, που είναι συνδεδεμένα με αυτό, για τις διευθύνσεις και τις παραμέτρους των σημείων πρόσβασης, μέσω downlink καναλιών, χωρίς να έχει λάβει εκ των προτέρων μια ερώτηση για ένα σημείο πρόσβασης.

3. Συνολική Λειτουργία του CARD Πρωτοκόλλου:

Το CARD πρωτόκολλο χρησιμοποιείται προκειμένου τα κινητά τερματικά να αντιστοιχούν το L2 αναγνωριστικό καθενός υποψηφίου, προς διαπομπή, σημείου πρόσβασης σε IP διεύθυνση, καθώς και να ανακαλύπτουν τις δυνατότητες των σημείων αυτών (π.χ. διαθέσιμο bandwidth). Επιπλέον, το πρωτόκολλο επιτρέπει στα σημεία πρόσβασης να συμπληρώνουν και να διατηρούν πίνακες με τις δυνατότητες



των υπολοίπων σημείων πρόσβασης. Για το σκοπό αυτό, το CARD πρωτόκολλο χρησιμοποιεί ένα μηχανισμό μηνυμάτων CARD Request και CARD Reply ανάμεσα σε ένα τερματικό και το τρέχον σημείο πρόσβασής του, καθώς και ανάμεσα στο τρέχον σημείο πρόσβασης και άλλα σημεία πρόσβασης, αντίστοιχα.

Για να επιτραπεί σε ένα κινητό τερματικό η ανάκτηση της IP διεύθυνσης και των χαρακτηριστικών ενός σημείου πρόσβασης, τα μηνύματα CARD Request και CARD Reply που ανταλλάσσονται μεταξύ ενός κινητού τερματικού και του τρέχοντος σημείου πρόσβασής του περιέχουν τα L2 αναγνωριστικά ενός ή περισσότερων σημείων πρόσβασης και τις IP διευθύνσεις τους, αντίστοιχα. Προαιρετικά, τα μηνύματα CARD Reply μπορούν να περιέχουν, επίσης, πληροφορίες για τις δυνατότητες του σημείου πρόσβασης. Οι δυνατότητες αυτές καθορίζονται ως μια λίστα παραμέτρων με τη μορφή «όνομα-τιμή», η οποία μετατρέπεται σε κατάλληλη μορφή και εμπερικλείεται στο CARD Reply μήνυμα

Πληροφορίες για τα διαθέσιμα σημεία πρόσβασης και τις αντίστοιχες δυνατότητές τους μπορεί να χρησιμοποιηθούν από το κινητό τερματικό κατά τη διαδικασία επιλογής του προορισμού της διαπομπής. Το τρέχον σημείο πρόσβασης εκκινεί ανταλλαγή πληροφοριών δυνατοτήτων με ένα υποψηφίο σημείο πρόσβασης, είτε όταν λαμβάνει ένα μήνυμα CARD Request από ένα κινητό τερματικό, το οποίο περιέχει πιθανόν αναγνωριστικά νέων σημείων πρόσβασης, είτε όταν διαπιστώνει ότι κάποιος από τους πίνακες πληροφοριών με τις δυνατότητες των σημείων πρόσβασης που διατηρεί περιέχουν παρωχημένες πληροφορίες. Μετά την ολοκλήρωση της ανταλλαγής πληροφοριών μεταξύ του τρέχοντος σημείου και των υπολοίπων σημείων πρόσβασης, το πρώτο οφείλει να ενημερώσει το κινητό τερματικό για τις αιτούμενες δυνατότητες μέσω ενός CARD Reply μηνύματος, που περιέχει όλες τις απαραίτητες πληροφορίες. Το τρέχον σημείο πρόσβασης μπορεί, επίσης, να στέλνει περιοδικά, με δική του πρωτοβουλία, CARD Reply μηνύματα σε όλα τα συνδεδεμένα κινητά τερματικά. Η συμπεριφορά αυτή των σημείων πρόσβασης καθορίζεται από πολιτικές των παρόχων δικτύων και οφείλει να ελέγχεται κεντρικά.

Τα μηνύματα CARD Reply που αποστέλλονται με πρωτοβουλία του τρέχοντος σημείου πρόσβασης πρέπει να στέλνονται σε όλα, ανεξαιρέτως, τα κινητά τερματικά και να περιέχουν αναγνωριστικά που θα τα διακρίνουν από τα μηνύματα CARD Reply που αποστέλλονται κατόπιν αιτήσεως του τερματικού.

Το πρωτόκολλο CARD επιτρέπει, επίσης, στο κινητό τερματικό να δηλώνει, προαιρετικά, στο τρέχον σημείο πρόσβασης τις προτιμήσεις του, όσον αφορά τις δυνατότητες του σημείου πρόσβασης για τις οποίες επιθυμεί να ενημερωθεί, εμπερικλείοντας την Παράμετρο Προτιμήσεων στο CARD Request μήνυμα. Το τρέχον σημείο πρόσβασης μπορεί να χρησιμοποιήσει την πληροφορία αυτή, προκειμένου να εκτελέσει ένα προαιρετικό φιλτράρισμα, για λόγους βελτιστοποίησης, και να επιστρέψει μόνο εκείνα τα χαρακτηριστικά τα οποία ενδιαφέρουν το αντίστοιχο τερματικό.

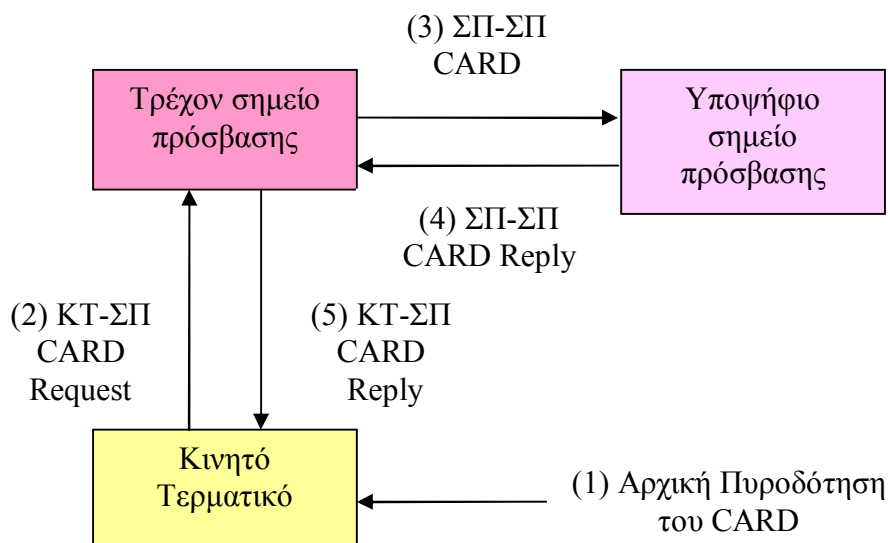
Επιπρόσθετα, το κινητό τερματικό μπορεί να παρέχει στο τρέχον σημείο πρόσβασης μια λίστα δυνατοτήτων με τη μορφή «όνομα-τιμή», δηλώνοντας όχι μόνο τις παραμέτρους, όπως απαιτούνται για το φιλτράρισμα, αλλά και συγκεκριμένες τιμές για κάθε μία παράμετρο. Με αυτόν τον τρόπο, δίνεται η δυνατότητα στο τρέχον σημείο πρόσβασης να κάνει ένα φιλτράρισμα των υποψηφίων σημείων πρόσβασης

και να παρέχει στο τερματικό τις διευθύνσεις και τα χαρακτηριστικά μόνο όσων σημείων πρόσβασης ικανοποιούν τις προτιμήσεις του.

Παραδείγματος χάριν, χρησιμοποιώντας την προαιρετική Παράμετρο Προτιμήσεων του CARD Request μηνύματος, ένα κινητό τερματικό δηλώνει στο τρέχον σημείο πρόσβασης του ότι ενδιαφέρεται μόνο για τις συγκεκριμένες παραμέτρους δυνατοτήτων της IEEE 802.11 διεπαφής, αφού αυτή είναι η μοναδική διεπαφή που το τερματικό υποστηρίζει. Συνεπώς, το τρέχον σημείο πρόσβασης επιστρέφει στο τερματικό μόνο τις δυνατότητες των IEEE 802.11 σημείων πρόσβασης. Παρόμοια, χρησιμοποιώντας την προαιρετική Παράμετρο Απαιτήσεων του CARD Request μηνύματος, ένα κινητό τερματικό δηλώνει στο τρέχον σημείο πρόσβασης ότι ενδιαφέρεται μόνο για σημεία πρόσβασης που μπορεί να ικανοποιήσουν ένα συγκεκριμένο περιορισμό επιπέδου ποιότητας (QoS). Στην περίπτωση αυτή, το κινητό τερματικό στέλνει το αντίστοιχο επίπεδο ποιότητας στο τρέχον σημείο πρόσβασης, χρησιμοποιώντας την προαιρετική παράμετρο προτιμήσεων του CARD Request μηνύματος. Ο QoS περιορισμός διατυπώνεται ως ένας ζεύγος «όνομα-τιμή» και ενσωματώνεται στην Παράμετρο Απαιτήσεων, η οποία, με τη σειρά της, προστίθεται στο αρχικό CARD Request μήνυμα. Βασιζόμενο στη ληφθείσα λίστα χαρακτηριστικών της Παραμέτρου Προτιμήσεων ή σε μια λίστα ζευγών «όνομα-τιμή» στην Παράμετρο Απαιτήσεων του μηνύματος, το τρέχον σημείο πρόσβασης μπορεί να χρησιμοποιήσει αυτά τα στοιχεία για να διαμορφώσει κατάλληλα το περιεχόμενο του CARD Reply μηνύματος που θα σταλεί πίσω στο τερματικό.

Εναλλακτικά, στην περίπτωση που δε πραγματοποιείται κάποια βελτιστοποίηση από τη μεριά του τρέχοντος σημείου πρόσβασης, τότε αυτό μπορεί απλά να αγνοήσει τις Παραμέτρους Προτιμήσεων και Απαιτήσεων που περιέχονται στο μήνυμα CARD Request.

Η λειτουργία του CARD πρωτοκόλλου, όπως περιγράφηκε παραπάνω, διακρίνει τα μηνύματα που ανταλλάσσονται μεταξύ κινητού τερματικού και σημείου πρόσβασης από αυτά που ανταλλάσσονται μεταξύ δυο διαφορετικών σημείων πρόσβασης. Για το λόγο αυτό τα μηνύματα αυτά, από εδώ και στο εξής, θα αναφέρονται ως ΚΤ-ΣΠ και ΣΠ-ΣΠ μηνύματα, αντίστοιχα.



Το παραπάνω σχήμα περιγράφει τη λειτουργία του Request/Reply-based πρωτοκόλλου, όταν αυτή εκκινείται από την πλευρά του τερματικού. Μετά τη λήψη των L2 αναγνωριστικών των σημείων πρόσβασης ή την πυροδότηση του πρωτοκόλλου (1), το κινητό τερματικό μεταδίδει ένα ή περισσότερα L2 αναγνωριστικά στο τρέχον σημείο πρόσβασής του, χρησιμοποιώντας ένα KT-ΣΠ CARD Request μήνυμα (2). Στην περίπτωση που το τερματικό επιθυμεί το σημείο πρόσβασής του να πραγματοποιήσει και ανάκτηση δυνατοτήτων των άλλων σημείων, πέρα από Αντίστροφη Μετάφραση Διευθύνσεων, αυτό πρέπει να δηλωθεί στο KT-ΣΠ CARD Request μήνυμα, μέσω κατάλληλου δείκτη. Χωρίς την ύπαρξη αυτού του δείκτη, το σημείο πρόσβασης που λαμβάνει το CARD Request μήνυμα θα εκτελέσει μόνο Αντίστροφη Μετάφραση Διευθύνσεων. Το τρέχον σημείο πρόσβασης του τερματικού αντιστοιχεί το L2 αναγνωριστικό με μια IP διεύθυνση, ή, σε περίπτωση που το τερματικό δεν έχει στείλει κάποιο τέτοιο αναγνωριστικό, μεταβιβάζει ολόκληρο τον πίνακα αντιστοίχισης L2 αναγνωριστικών – IP διευθύνσεων που διατηρεί τοπικά. Σε περίπτωση που μία ή περισσότερες εγγραφές δυνατοτήτων είναι παρωχημένες στον πίνακα υποψηφίων σημείων πρόσβασης του τρέχοντος σημείου, το τελευταίο επικοινωνεί απευθείας με ένα υποψήφιο σημείο πρόσβασης και ανακτά από αυτό τις απαιτούμενες πληροφορίες μέσω μια χειραψίας (3) ΣΠ-ΣΠ CARD Request - (4) ΣΠ-ΣΠ CARD Reply μηνυμάτων. Το τρέχον σημείο πρόσβασης, τότε, ανανεώνει τον τοπικό πίνακα πληροφοριών του και, στη συνέχεια, μεταβιβάζει στο κινητό τερματικό την IP διεύθυνση και τις δυνατότητες του υποψηφίου σημείου πρόσβασης, εφόσον αυτές έχουν ζητηθεί, μέσω ενός (4) KT-ΣΠ CARD Reply μηνύματος.

### 2.3 Συνιστώσα Προτιμήσεων Χρήστη

Η Συνιστώσα Προτιμήσεων Χρήστη (User Preferences module – UP) είναι υπεύθυνη για την πρόσβαση, αποθήκευση και επεξεργασία του προφίλ του χρήστη του τερματικού. Μια γραφική διεπαφή (GUI), μέσω της οποίας ο χρήστης είναι σε θέση να καθορίσει τις προτιμήσεις του, επιτρέπει στο χρήστη να δίνει διαφορετική προτεραιότητα σε παραμέτρους που επηρεάζουν τη διαδικασία επιλογής δικτύου πρόσβασης. Αυτή η απόδοση προτεραιοτήτων ισοδυναμεί, όπως θα δούμε, με τον καθορισμό των τιμών των συντελεστών που αντιστοιχούν στις παραμέτρους «ποιότητα», «κόστος» και «πάροχος» μιας πιθανής σύνδεσης.

Έστω, δηλαδή, ότι επιθυμούμε να εξυπηρετήσουμε μια υπηρεσία S που ζητά ο χρήστης εγκαθιστώντας μια σύνδεση με ένα σημείο πρόσβασης P, εντός της εμβέλειας του τερματικού του χρήστη. Η παράμετρος «ποιότητα» αντικατοπτρίζει την ποιότητα της σύνδεσης που θα εγκατασταθεί, σε συνδυασμό με το επίπεδο ποιότητας στο οποίο θα διατεθεί η υπηρεσία S, η παράμετρος «κόστος» αντιστοιχεί στο κόστος μιας τέτοιας σύνδεσης (δηλαδή, είναι ένα μέτρο για το πόσο θα χρεωθεί ο χρήστης), ενώ η παράμετρος «πάροχος» φανερώνει αν ο πάροχος στον οποίο ανήκει το σημείο πρόσβασης P είναι ο προτιμώμενος από το χρήστη πάροχος ή όχι.

Συνεπώς, οι συντελεστές των τριών αυτών παραμέτρων αναπαριστούν το «βάρος» που θα έχει καθεμιά τους κατά τη διαδικασία λήψης απόφασης από τον αλγόριθμο ευφυούς επιλογής δικτύου πρόσβασης. Για παράδειγμα, έστω ότι ο χρήστης έχει

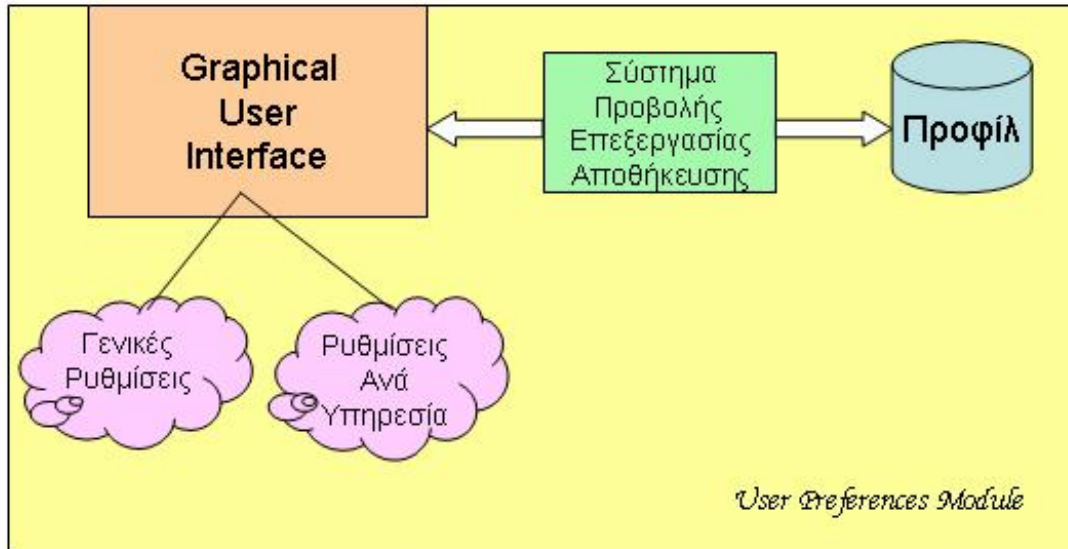
επιλέξει, μια δεδομένη στιγμή, την παράμετρο «ποιότητα» ως πρώτη (πιο σημαντική), το «κόστος» ως δεύτερη και τον «πάροχο» ως τρίτη (λιγότερο σημαντική). Τότε ο συντελεστής που αντιστοιχεί στην «ποιότητα» πρέπει να έχει τη μεγαλύτερη τιμή, αμέσως μετά να ακολουθεί ο συντελεστής του «κόστους», ενώ τη μικρότερη τιμή πρέπει να έχει ο συντελεστής του «παρόχου».

Εκτός, όμως, από τον καθορισμό των παραπάνω γενικών προτιμήσεων, ο χρήστης μπορεί να προβεί και σε μερικές περισσότερο ειδικές, ανά υπηρεσία, ρυθμίσεις. Πιο συγκεκριμένα, η ίδια γραφική διεπαφή τού παρέχει τη δυνατότητα να καθορίσει, για κάθε μία από τις υπηρεσίες που υποστηρίζει το κινητό του, το μέγιστο επίπεδο ποιότητας στο οποίο επιθυμεί να διατίθεται. Με αυτόν τον τρόπο, δηλώνει ποιο είναι το μέγιστο επίπεδο ποιότητας που τον ενδιαφέρει σε κάθε υπηρεσία και, επομένως, ποιο είναι το μέγιστο κόστος που είναι διατεθειμένος να επωμιστεί.

Η ύπαρξη αυτής της δυνατότητας προσφέρει μερικά πολύ σημαντικά πλεονεκτήματα: Πρώτον, ο χρήστης είναι σε θέση, μέσω των συγκεκριμένων ρυθμίσεων, να καλύψει καλύτερα τις ανάγκες του, ακόμα και αν οι προτιμήσεις του αλλάζουν συχνά με την πάροδο του χρόνου. Δεύτερον, προσφέρεται ένας σαφής τρόπος περιορισμού του κόστους που επωμίζεται ο χρήστης για τη χρήση των υπηρεσιών που υποστηρίζει το τερματικό του, κάτι ιδιαίτερα ελκυστικό για όλους, λίγο ή πολύ, τους χρήστες. Ο χρήστης μπορεί να επιλέξει εκείνο το επίπεδο ποιότητας που θα του προσφέρει την καλύτερη ισορροπία μεταξύ κόστους και επαρκούς κάλυψης των αναγκών του. Από εκεί και πέρα, ο αλγόριθμος ευφυούς επιλογής δικτύου πρόσβασης θα αναλάβει να βρει την καλύτερη για αυτόν λύση. Τρίτον, και ίσως λιγότερο προφανές, οι χρήστες, αλλά και οι πάροχοι υπηρεσιών, απαλλάσσονται από την ανάγκη να συμπεριλαμβάνονται στα συμβόλαιά τους πληροφορίες σχετικές με το μέγιστο επιθυμητό επίπεδο ποιότητας ανά υπηρεσία. Αυτού του είδους οι πληροφορίες μεταφέρονται στην πλευρά του τερματικού και, έτσι, είναι πολύ ευκολότερο να τροποποιηθούν, οποιαδήποτε στιγμή.

Η γραφική διεπαφή για την οποία έχει γίνει λόγος θα πρέπει να είναι απλή και εύχρηστη, σαφής και κατανοητή με την πρώτη ματιά, έτσι ώστε να προτρέπει το χρήστη να την χρησιμοποιήσει, αντί να τον αποτρέπει. Πρόκειται για μια από τις πλέον βασικές απαιτήσεις: Ο χρήστης θα πρέπει να καταναλώνει όσο το δυνατό λιγότερο χρόνο για τον καθορισμό των επιθυμητών ρυθμίσεων.

Πέρα, όμως, από την εν λόγω γραφική διεπαφή, το UP module περιλαμβάνει μια μικρή βάση δεδομένων (ή κάποιου είδους αρχείο) στην οποία εμπεριέχεται το προφίλ του χρήστη, καθώς και ένα σύστημα δια μέσου του οποίου καθίσταται δυνατή η προβολή, επεξεργασία και αποθήκευση του προφίλ, με χρήση του GUI. Όλα τα παραπάνω εικονίζονται στο σχήμα που ακολουθεί.



Περισσότερες πληροφορίες για το UP module μπορούν να βρεθούν στο κεφάλαιο 3, που ασχολείται εκτενέστερα με την περιγραφή της Συνιστώσας Προτιμήσεων Χρήστη.

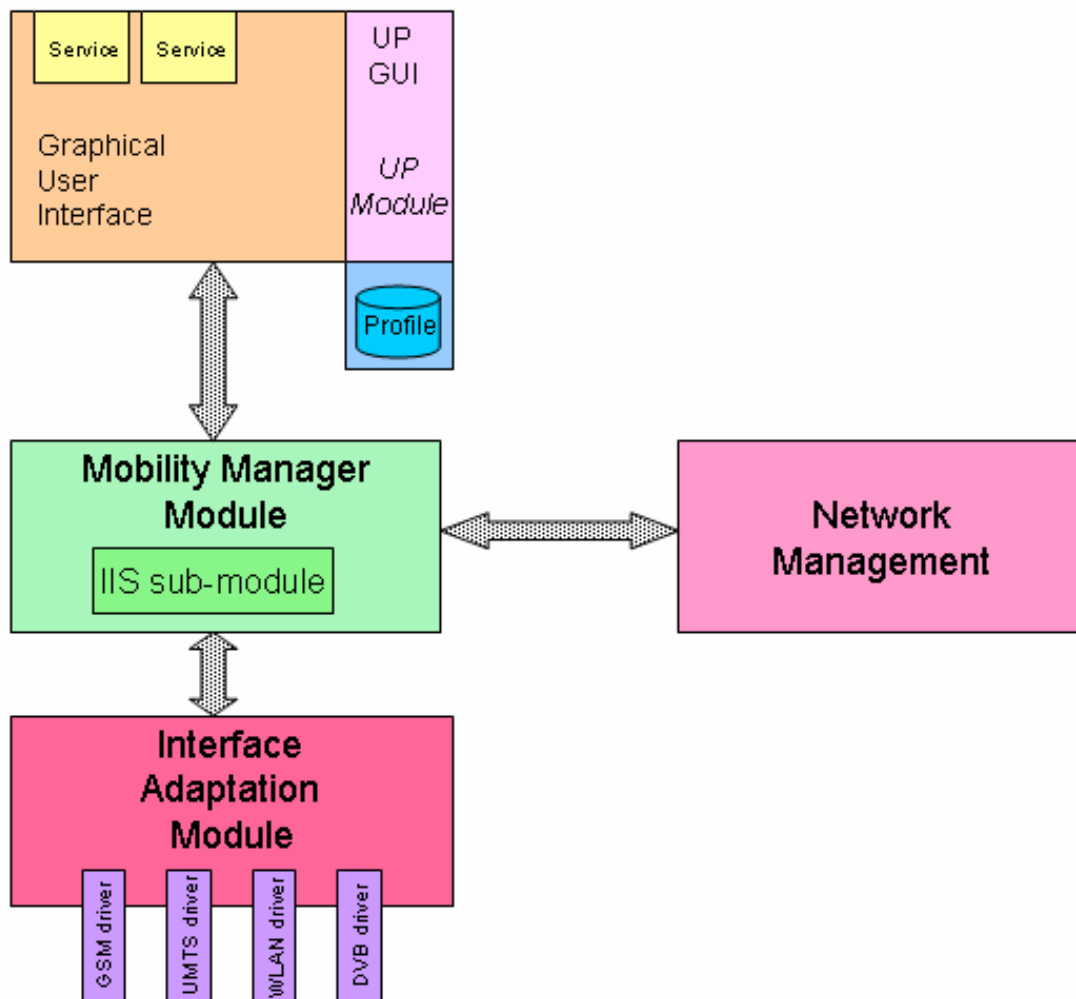
#### 2.4 Συνιστώσα Διαχείρισης Κινητικότητας

Η Συνιστώσα Διαχείρισης Κινητικότητας (Mobility Manager module – MM) είναι υπεύθυνη για τη διαχείριση όλων των γεγονότων που αφορούν μια διαπομπή. Αυτό ισχύει τόσο για την περίπτωση που η διαπομπή επιβάλλεται από το τερματικό όσο για εκείνη που επιβάλλεται από το δίκτυο. Στην πρώτη περίπτωση, και μόλις ειδοποιηθεί από το IA module (παράγραφος 2.2.2) περί σημαντικής υποβάθμισης ενός σημείου πρόσβασης που χρησιμοποιείται ή περί εμφάνισης κάποιου καινούριου, ο Mobility Manager θα εκκινήσει μια αλυσίδα γεγονότων, η οποία θα ξεκινά με την ενεργοποίηση της διαδικασίας ανακάλυψης υποψήφιων σημείων πρόσβασης και θα λήγει με την ενεργοποίηση του σημαντικότερου συστατικού του, του IIS sub-module. Το ίδιο, φυσικά, θα συμβεί και στην περίπτωση αίτησης μιας καινούριας υπηρεσίας από την πλευρά του χρήστη ή στην περίπτωση τροποποίησης των προτιμήσεών του.

Ένα πρωτόκολλο σαν το CARD της IETF (παράγραφος 2.2.3) μπορεί να χρησιμοποιηθεί για την υποστήριξη της διαπομπής του κινητού τερματικού μέσω των εξής δύο λειτουργιών: Πρώτον, τη μετάφραση του L2 αναγνωριστικού των υποψήφιων σημείων πρόσβασης (access points) στην IP διεύθυνση του συσχετιζόμενου υποψήφιου δρομολογητή πρόσβασης (access router). Δεύτερον, την ανακάλυψη των δυνατοτήτων καθενός υποψήφιου δρομολογητή πρόσβασης. Η λίστα των υποψήφιων δρομολογητών πρόσβασης και οι παράμετροι που αντιπροσωπεύουν τις δυνατότητές τους, σε συνδυασμό με ένα πλήθος άλλων παραμέτρων, συμπεριλαμβανομένων των μετρήσεων που έχουν συλλεχθεί από το IA module, των προτιμήσεων του χρήστη και των πληροφοριών που αφορούν τις αιτούμενες υπηρεσίες, χρησιμεύουν ως είσοδοι για το IIS module, που αποτελεί και το σημαντικότερο τμήμα του Mobility Manager.

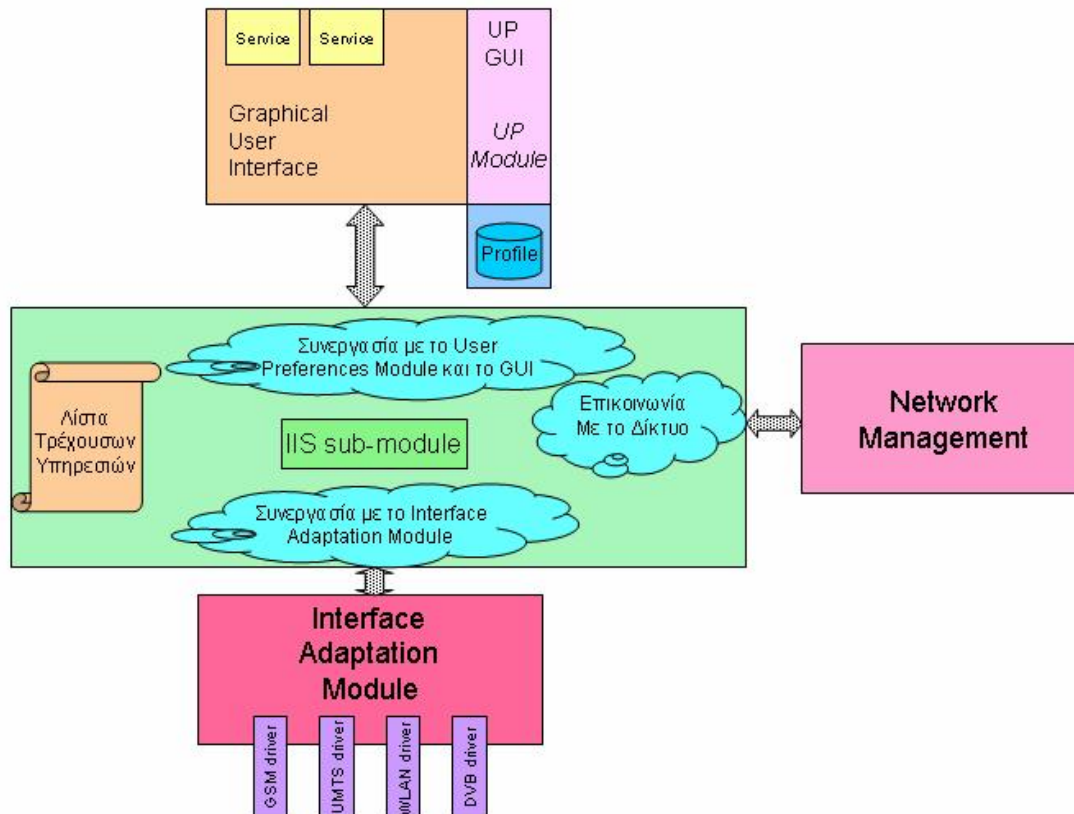
Λαμβάνοντας υπόψη όλες τις παραπάνω εισόδους, ο αλγόριθμος ευφυούς επιλογής δικτύου πρόσβασης ή, πιο σύντομα, αλγόριθμος του IIS εντοπίζει τη βέλτιστη λύση για την εξυπηρέτηση της αιτούμενης υπηρεσίας. Η λύση αυτή συνίσταται τόσο στην επιλογή του καταλληλότερου σημείου πρόσβασης όσο και στην επιλογή του καταλληλότερου (βάσει των συνθηκών που επικρατούν στο δίκτυο και των προτιμήσεων του χρήστη) επιπέδου ποιότητας για την υπηρεσία. Στο κεφάλαιο 4, θα γίνει αναλυτική παρουσίαση της διαδικασίας ευφυούς επιλογής δικτύου πρόσβασης.

Ο Mobility Manager είναι, ίσως, το σπουδαιότερο κομμάτι του συστήματος διαχείρισης ενός τερματικού, όχι μόνο γιατί περιλαμβάνει το IIS sub-module, αλλά και επειδή αποτελεί, ουσιαστικά, το συνδετικό κρίκο ανάμεσα στη Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου, τη Συνιστώσα Προτιμήσεων Χρήστη, τη Γραφική Διεπαφή Χρήστη αλλά και την Οντότητα Διαχείρισης του Δικτύου, όπως δείχνει και το επόμενο σχήμα.



Η επικοινωνία με την Οντότητα Διαχείρισης Δικτύου, που φαίνεται στο παραπάνω σχήμα, γίνεται προαιρετικά, προκειμένου να εγκριθεί η απόφαση στην οποία έφτασε ο αλγόριθμος του IIS. Σε μια γενικότερη προσέγγιση, μια διατεταγμένη (κατά φθίνουσα σειρά προτίμησης) λίστα αποφάσεων / επιλογών θα μπορούσε να υποβάλλεται στο σύστημα διαχείρισης του δικτύου, προκειμένου το τελευταίο να επιλέξει με τη σειρά του την καταλληλότερη, χρησιμοποιώντας τα δικά του κριτήρια.

Εκτός, λοιπόν, από τη λειτουργία της ευφυούς επιλογής δικτύου πρόσβασης (IIS sub-module), ο Mobility Manager ενσωματώνει και τις λειτουργίες της επικοινωνίας / συνεργασίας με το IA module, της επικοινωνίας / συνεργασίας με το UP module και το GUI, της επικοινωνίας με το Δίκτυο (για την έγκριση της απόφασής του ως προς το σημείο πρόσβασης που επέλεξε) αλλά και της διατήρησης μιας λίστας με τις τρέχουσες υπηρεσίες. Τη λίστα αυτή φροντίζει να την ενημερώνει κατάλληλα όποτε ξεκινά ή τερματίζεται μια υπηρεσία στο τερματικό. Τα παραπάνω συνοψίζονται στο σχήμα που ακολουθεί.

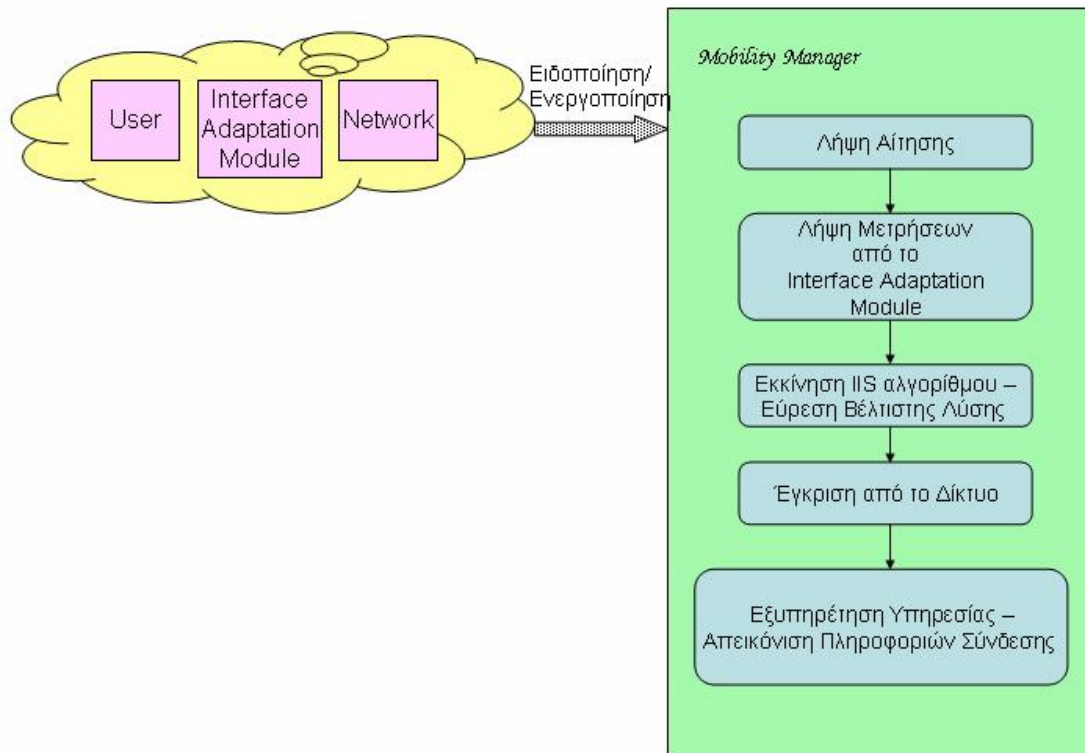


## 2.5 Ροή Πληροφορίας

Όπως έχουμε εξηγήσει, ο Mobility Manager μπορεί να ειδοποιηθεί να «αναλάβει δράση» είτε από το IA module (εμφάνιση καινούριου σημείου πρόσβασης ή εξαφάνιση, δηλαδή έντονη υποβάθμιση, ενός χρησιμοποιούμενου σημείου πρόσβασης), είτε από το χρήστη (μέσω τροποποίησης του προφίλ του από το UP module ή μέσω αίτησης για καινούρια υπηρεσία από το GUI), είτε, τέλος, από το δίκτυο (για λόγους που σχετίζονται με τη διαχείριση των πόρων του ή με τη διαχείριση της κινητικότητας των χρηστών). Οι δύο πρώτες περιπτώσεις, στις οποίες η «ειδοποίηση» προέρχεται από την πλευρά του τερματικού, είναι εκείνες που μας ενδιαφέρουν περισσότερο.

Η ακολουθία ενεργειών (ροή πληροφορίας) που λαμβάνουν χώρα, ύστερα από την ενεργοποίηση του Mobility Manager, παρουσιάζεται στο σχήμα που ακολουθεί.





Όπως παρατηρούμε, η γενική αλληλουχία ενεργειών έχει ως εξής: Αφού ληφθεί μια αίτηση (από το IA module, το χρήστη ή το δίκτυο), ο Mobility Manager επικοινωνεί με το IA module για τη λήψη των πλέον πρόσφατων μετρήσεων από τους οδηγούς των καρτών δικτύου. Στη συνέχεια, ενεργοποιεί το IIS sub-module, το οποίο τρέχει τον αλγόριθμο ευφυούς επιλογής δικτύου πρόσβασης. Η λύση στην οποία καταλήγει υποβάλλεται έπειτα, προαιρετικά, για έγκριση από το σύστημα διαχείρισης του δικτύου. Όταν συμβεί και αυτό, εγκαθίσταται η σύνδεση με το κατάλληλο σημείο πρόσβασης, προκειμένου να εξυπηρετηθεί η αιτούμενη υπηρεσία. Ταυτόχρονα, γίνεται απεικόνιση των πληροφοριών σύνδεσης (δηλαδή, πληροφοριών σχετικά με την απόφαση που λήφθηκε από το IIS και εγκρίθηκε από το δίκτυο) στο GUI του τερματικού.

Αναλυτική παρουσίαση του συγκεκριμένου θέματος θα γίνει στο κεφάλαιο 4 και, συγκεκριμένα, στην παράγραφο 4.1.9.



### 3. Συνιστώσα Προτιμήσεων Χρήστη

#### 3.1 Παρουσίαση της Γραφικής Διεπαφής

##### 3.1.1 Εισαγωγή

Όπως έχει ήδη αναφερθεί, κεντρικό ρόλο στη λειτουργία του συστήματος διαχείρισης του τερματικού κατέχουν οι προτιμήσεις του χρήστη, οι οποίες επηρεάζουν σε σημαντικό βαθμό την τελική επιλογή του κατάλληλου access point. Οι προτιμήσεις αυτές καταγράφονται μέσω μιας γραφικής διεπαφής (Graphical User Interface – GUI), η οποία δίνει τη δυνατότητα στο χρήστη να καθορίσει, μέσω ενός εύχρηστου περιβάλλοντος, τις προτιμήσεις του, όσον αφορά τις κρίσιμες παραμέτρους που συντελούν στη λήψη της τελικής απόφασης.

Στο σχεδιασμό που ακολουθήθηκε, η γραφική διεπαφή που αφορά τον καθορισμό των παραπάνω παραμέτρων (προφίλ του χρήστη) εντάσσεται στο πλαίσιο μιας ευρύτερης γραφικής διεπαφής, η οποία προσομοιώνει τη λειτουργία και το “look and feel” ενός κινητού τερματικού. Αλληλεπιδρώντας ο χρήστης με αυτήν τη γραφική διεπαφή, είναι σε θέση να εκκινήσει ή να τερματίσει μια υπηρεσία, καθώς επίσης και να ενημερωθεί για το σύνολο των τρεχουσών υπηρεσιών και τις λεπτομέρειες της εκάστοτε σύνδεσης.

##### 3.1.2 Παρουσίαση Διαθέσιμων Υπηρεσιών

Προτού, όμως, προχωρήσουμε στην παρουσίαση αυτής της γραφικής διεπαφής, θα σταθούμε στην παρουσίαση των υπηρεσιών που συμπεριλήφθηκαν στην υλοποίηση του συστήματος.

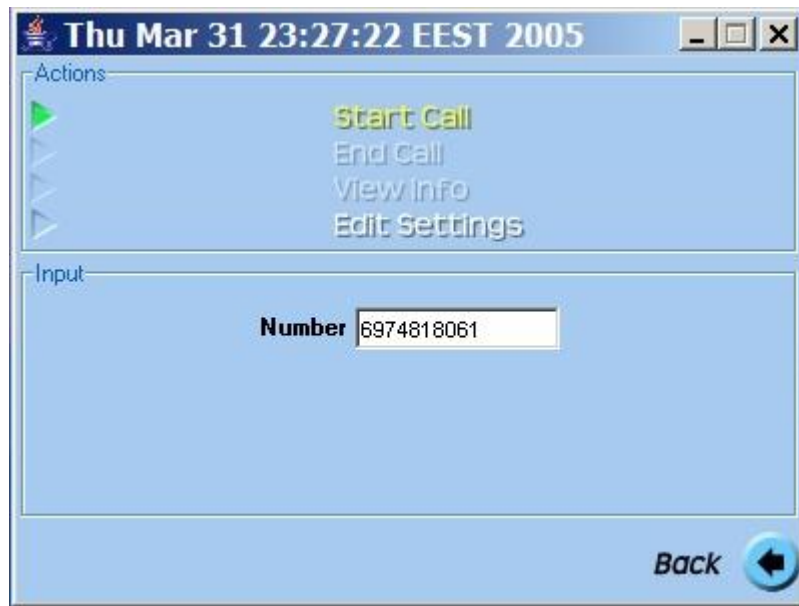
Στο σύνολο τους, οι διαθέσιμες υπηρεσίες στην εφαρμογή είναι 7. Ακολουθεί αναλυτική περιγραφή καθεμιάς:



##### 1. Voice Call (Φωνητική Κλήση):

Μέσω της υπηρεσίας αυτής, ο χρήστης έχει τη δυνατότητα να πραγματοποιήσει συνομιλία με κάποιον άλλο χρήστη που είναι συνδρομητής στην ίδια υπηρεσία.

Η γραφική διεπαφή δίνει τη δυνατότητα στο χρήστη να εισάγει τον αριθμό του συνδρομητή με τον οποίο επιθυμεί να συνομιλήσει και, κατόπιν, να εκκινήσει τη διαδικασία της κλήσης.



Οποιαδήποτε στιγμή, ο χρήστης μπορεί να τερματίσει την κλήση που είναι σε εξέλιξη, να δει πληροφορίες σχετικές με την τρέχουσα σύνδεση ή να τροποποιήσει τις προτιμήσεις που αφορούν τη συγκεκριμένη υπηρεσία.

Η υπηρεσία Voice Call παρέχεται σε 5 διαθέσιμα επίπεδα ποιότητας (QoS levels), καθένα από τα οποία αντιστοιχεί σε ένα διαφορετικό bitrate, σύμφωνα με τις δυνατότητες της κάθε υποστηριζόμενης τεχνολογίας (Πίνακας 3.1).

Voice Call	
QoS level	Bitrate (Kbps)
QoS-1	4
QoS-2	8
QoS-3	16
QoS-4	20
QoS-5	25

Πίνακας 3.1

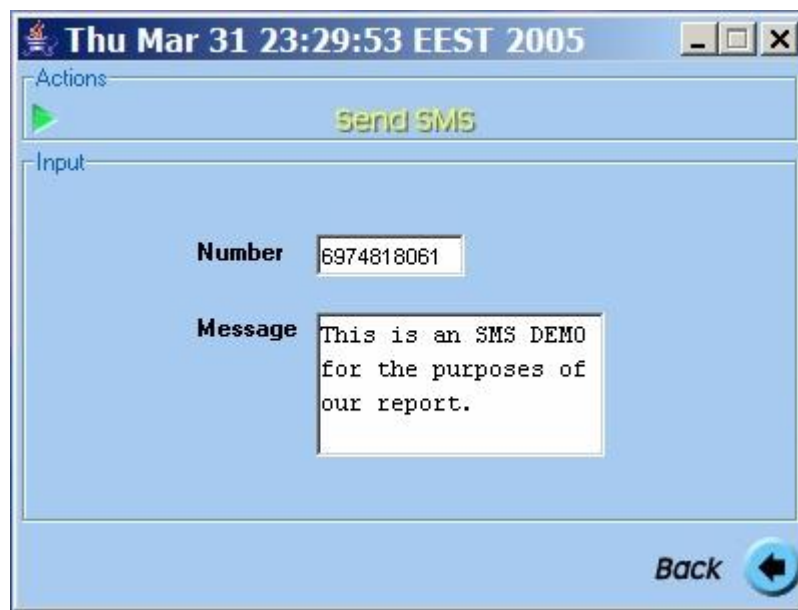
Γενικότερα, για την υπηρεσία φωνητικής κλήσης, έχει αναπτυχθεί μια μεγάλη ποικιλία κωδικοποιητών φωνής, που χρησιμοποιούν εξαιρετικά μικρούς ρυθμούς μετάδοσης, της τάξεως των 2.4Kbps, έως αρκετά μεγαλύτερους, της τάξεως ακόμα και των 64Kbps. Σε GSM συστήματα διατίθενται, συνήθως, μέχρι 13Kbps για υπηρεσίες φωνής, αλλά μπορούν να υποστηριχθούν και μεγαλύτερες ταχύτητες. Μεγάλη ποικιλία ρυθμών μετάδοσης συναντάται και σε VoIP (Voice over IP) συστήματα. Τα παραπάνω στοιχεία μάς οδηγούν στη θεώρηση των 5 επιπέδων ποιότητας που παρουσιάζονται στον Πίνακα 3.1, ξεκινώντας από ένα αρκετά χαμηλό bitrate (4Kbps) και καταλήγοντας σε ένα αρκετά υψηλότερο (25Kbps), καλύπτοντας έτσι ένα αρκετά ευρύ φάσμα αναγκών.



## 2. Short Message Service – SMS (Υπηρεσία Σύντομων Μηνυμάτων):

Μέσω της υπηρεσίας αυτής, ο χρήστης έχει τη δυνατότητα να αποστείλει ένα σύντομο μήνυμα κειμένου (μέχρι 160 χαρακτήρες).

Μέσω της γραφικής διεπαφής, ο χρήστης επιλέγει τον αριθμό του παραλήπτη και εισάγει το κείμενο που επιθυμεί να αποστείλει.



Καθώς ο όγκος των μεταφερόμενων δεδομένων σε αυτήν την υπηρεσία είναι εξαιρετικά μικρός, δεν έχει νόημα η ύπαρξη περισσότερων του ενός επιπέδων ποιότητας. Ενδεικτικά παραθέτουμε το απαιτούμενο bitrate για την αποστολή ενός SMS (Πίνακας 3.2).

SMS	
QoS level	Bitrate (Kbps)
QoS-1	1

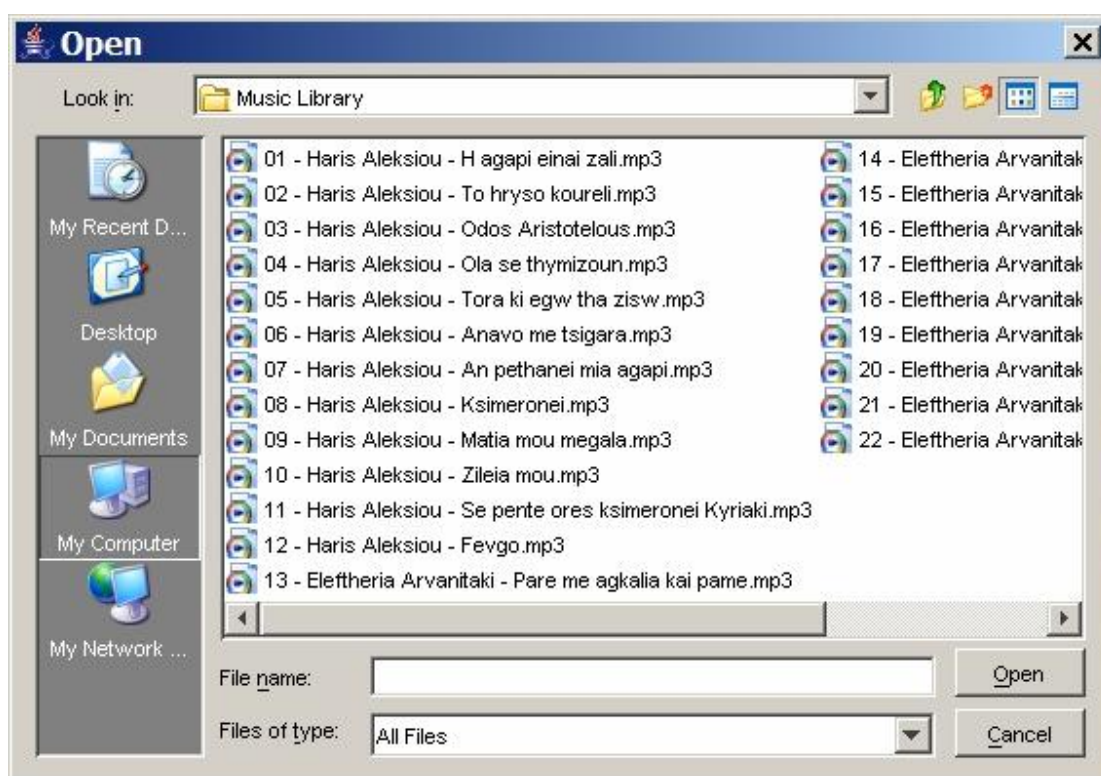
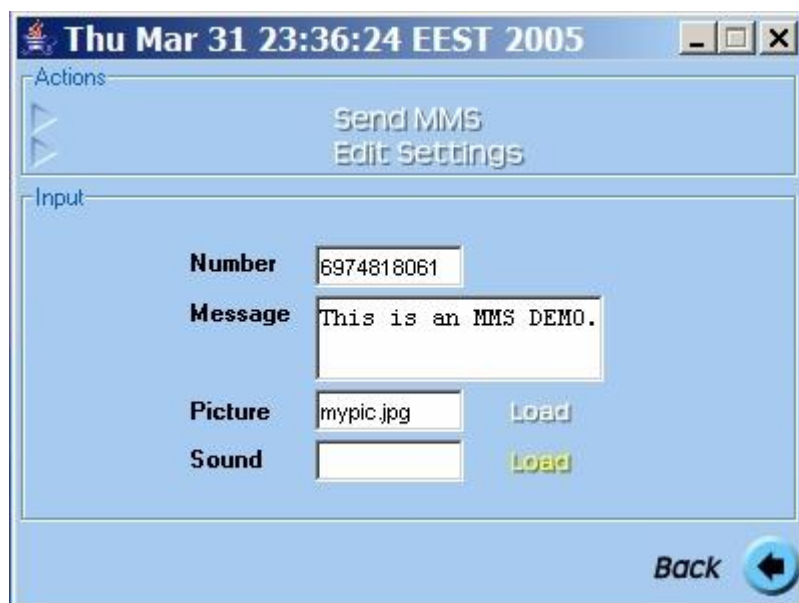
Πίνακας 3.2



## 3. Multimedia Message Service – MMS (Υπηρεσία Μηνυμάτων Πολυμέσων):

Η υπηρεσία αυτή είναι παρόμοια με την υπηρεσία σύντομων μηνυμάτων. Παρέχει αυτόματη και άμεση παράδοση προσωπικών μηνυμάτων πολυμέσων, ανάμεσα σε δύο χρήστες. Εκτός από το οικείο περιεχόμενο κειμένου των γραπτών μηνυμάτων, τα μηνύματα πολυμέσων είναι δυνατό να περιέχουν εικόνες, γραφικά, φωνή και κλιπ ήχου. Ένα μήνυμα MMS, δηλαδή, είναι μια παρουσίαση πολυμέσων σε μία και μόνο καταχώρηση. Δεν είναι ένα αρχείο κειμένου με συνημμένα.

Η γραφική διεπαφή παρέχει στο χρήστη ακριβώς τη δυνατότητα να επιλέξει και να εισάγει τα στοιχεία πολυμέσων (εικόνα, ήχος) που επιθυμεί να συνοδεύσουν το κείμενό του, καθώς επίσης και τον αριθμό του παραλήπτη. Επίσης, του δίνει η δυνατότητα να τροποποιήσει την επιθυμητή ποιότητα του μηνύματος.



Η ύπαρξη του multimedia περιεχομένου, αλλά ταυτόχρονα και ο μικρός σχετικά όγκος ενός MMS μηνύματος, μας οδηγούν στη θεώρηση 2 επιπέδων ποιότητας, τα οποία παρατίθενται αμέσως μετά (Πίνακας 3.3).

MMS	
QoS level	Bitrate (Kbps)
QoS-1	8
QoS-2	32

Πίνακας 3.3

Και τα δύο επίπεδα ποιότητας (8Kbps και 32Kbps) που θεωρήθηκαν είναι εντός των δυνατοτήτων ενός GSM/GPRS συστήματος. Το ίδιο ισχύει, φυσικά, και για τους υπόλοιπους τύπους συστημάτων που μας απασχολούν.



#### 4. Email (Υπηρεσία Μηνυμάτων Ηλεκτρονικού Ταχυδρομείου):

Πρόκειται για την πιο διαδεδομένη υπηρεσία, αυτήν τη στιγμή, στο Internet. Αφορά την αποστολή σε μία ηλεκτρονική διεύθυνση ενός ηλεκτρονικού μηνύματος που μπορεί να περιλαμβάνει μορφοποιημένο κείμενο, εικόνες, αρχεία ήχου, αλλά και οποιοδήποτε άλλο τύπο συνημμένου αρχείου. Τα μηνύματα ακολουθούν, συνήθως, το πρότυπο MIME (Multipurpose Internet Mail Extensions).

Η γραφική διεπαφή δίνει τη δυνατότητα εισαγωγής του κειμένου, επιλογής των αρχείων που επιθυμεί ο χρήστης να επισυναφθούν και, τέλος, καθορισμού της ηλεκτρονικής διεύθυνσης του παραλήπτη.



Για την υπηρεσία Email, θεωρήθηκε μόνο ένα επίπεδο ποιότητας (Πίνακας 3.4), καθώς το ηλεκτρονικό μήνυμα πρέπει να αποσταλεί αυτούσιο, χωρίς περιθώρια υποβάθμισης ποιότητας.

EMAIL	
QoS level	Bitrate (Kbps)
QoS-1	32

Πίνακας 3.4

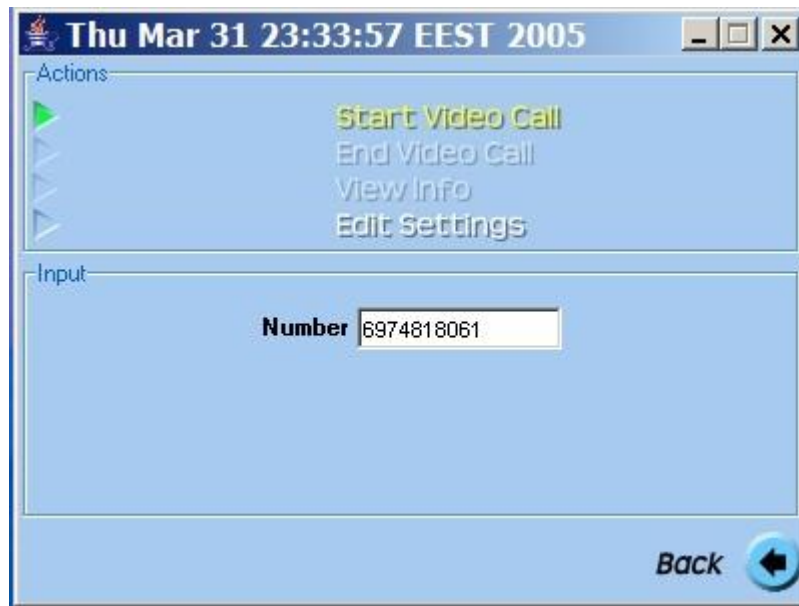


### 5. Video Call (Βιντεοκλήση):

Η υπηρεσία αυτή συνδυάζει τη φωνή με την οπτική επικοινωνία. Δίνει τη δυνατότητα στους δύο συνομιλητές να βλέπουν ο ένας τον άλλο στην οθόνη του τερματικού τους, κατά τη διάρκεια της συνομιλίας.

Η γραφική διεπαφή δίνει τη δυνατότητα στο χρήστη, πέραν του καθορισμού του αριθμού του συνομιλητή, να εκκινήσει μια βιντεοκλήση, να τερματίσει τη βιντεοκλήση που είναι σε εξέλιξη, να δει πληροφορίες σχετικές με την τρέχουσα σύνδεση ή να τροποποιήσει τις προτιμήσεις που αφορούν τη συγκεκριμένη υπηρεσία.





Οι αυξημένες απαιτήσεις σε bitrate της συγκεκριμένης υπηρεσίας, αλλά και οι ανοχές των χρηστών σε μείωση της ποιότητας, μας οδηγούν στη θεώρηση 5 επιπέδων ποιότητας, με αντίστοιχους ρυθμούς, σύμφωνα με τη δυνατότητα κάθε τεχνολογίας πρόσβασης (Πίνακας 3.5).

Σημειώνεται ότι η θεωρητικά μέγιστη παρεχόμενη ταχύτητα ενός GSM/GPRS συστήματος είναι τα 171.2Kbps (άρα μπορούν να ικανοποιηθούν τα τρία πρώτα επίπεδα ποιότητας), με ταυτόχρονη χρήση και των οκτώ χρονοσχισμών, ενώ οι δυνατότητες των 3G και WLAN συστημάτων είναι σημαντικά υψηλότερες (2Mbps και 11Mbps ή και παραπάνω, αντίστοιχα) και καθιστούν τα συστήματα αυτά ικανά να υποστηρίξουν όλα τα δυνατά επίπεδα ποιότητας που έχουν θεωρηθεί.

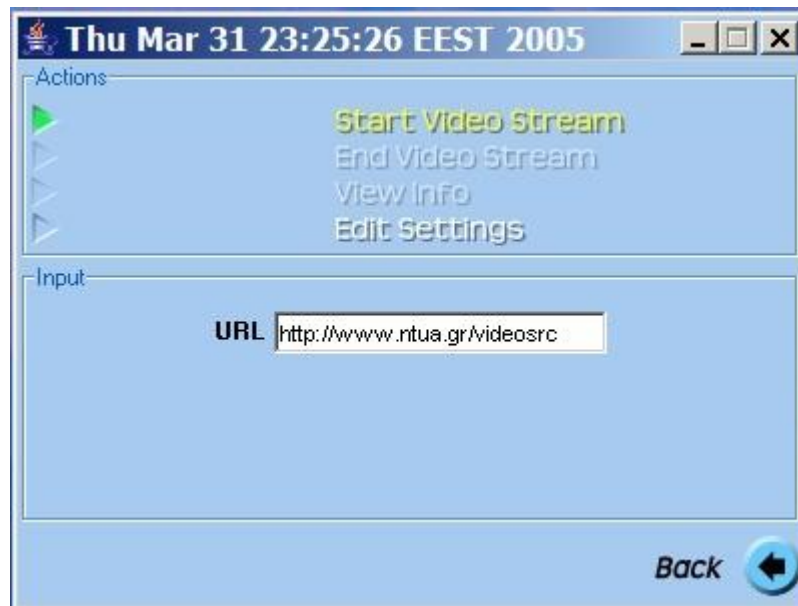
Video Call	
QoS level	Bitrate (Kbps)
QoS-1	32
QoS-2	64
QoS-3	128
QoS-4	256
QoS-5	384

Πίνακας 3.5



6. Video Stream (Υπηρεσία Λήψης και Ταυτόχρονης Αναπαραγωγής Βίντεο):

Η υπηρεσία αυτή προσφέρει στους συνδρομητές τη δυνατότητα να λαμβάνουν βίντεο και να το αναπαράγουν σε πραγματικό χρόνο, που σημαίνει ότι ο χρήστης μπορεί να παρακολουθεί το βίντεο στην οθόνη του τερματικού του, κατά τη διάρκεια της λήψης του. Η λήψη του βίντεο αυτού μπορεί να γίνει από κάποιον άλλο χρήστη ή από κάποιο server. Η γραφική διεπαφή επιτρέπει στο χρήστη να καθορίζει το URL της πηγής του βίντεο.



Ανά πάσα στιγμή, ο χρήστης μπορεί να τερματίσει τη λήψη του βίντεο, να δει πληροφορίες σχετικές με την τρέχουσα σύνδεση ή να τροποποιήσει τις προτιμήσεις που αφορούν τη συγκεκριμένη υπηρεσία.

Το βίντεο είναι μια υπηρεσία πολυμέσων που, από τη φύση της, μπορεί να διατεθεί σε πολλά επίπεδα ποιότητας, ανάλογα με τον επιθυμητό βαθμό συμπίεσης, το διαθέσιμο εύρος ζώνης και την ανοχή των χρηστών ως προς τη μείωση της ποιότητας. Θεωρήθηκαν, συνολικά, 5 επίπεδα ποιότητας, τα οποία συνοψίζονται στον ακόλουθο πίνακα (Πίνακας 3.6).

Σε ένα GSM/GPRS σύστημα (δυνατότητες μέχρι 171.2Kbps), θα μπορούσαν να ικανοποιηθούν τα δύο πρώτα επίπεδα ποιότητας, αλλά όχι τα επόμενα. Από την άλλη, ένα UMTS (τυπικές δυνατότητες μέχρι 2Mbps) ή ένα WLAN (τυπικές δυνατότητες μέχρι 11 ή 54Mbps) σύστημα θα μπορούσε να ικανοποιήσει όλα τα δυνατά επίπεδα ποιότητας που έχουν θεωρηθεί.

Video Stream	
QoS level	Bitrate (Kbps)
QoS-1	64
QoS-2	128
QoS-3	512
QoS-4	1024
QoS-5	2000

Πίνακας 3.6



### 7. Web Browsing (Περιήγηση στον Παγκόσμιο Ιστό):

Πρόκειται σίγουρα για μια από τις πιο δημοφιλείς υπηρεσίες του Internet, χάρη στην οποία καθίσταται εξαιρετικά εύκολη η αναζήτηση και ανάκτηση πληροφοριών. Το Web είναι, ουσιαστικά, ένα σύνολο υπολογιστών/διανομέων (servers) που χρησιμοποιεί hypertext links (συνδέσμους υπερκειμένου), για την προσπέλαση HTML εγγράφων. Οι χρήστες



έχουν, εκτός των άλλων, τη δυνατότητα να αναζητήσουν πληροφορίες, να περιηγηθούν μεταξύ των αναρίθμητων δικτυακών τόπων, να προσπελάσουν multimedia περιεχόμενο, αλλά και να «κατεβάσουν» στο τερματικό τους αρχεία οποιουδήποτε τύπου.

Η γραφική διεπαφή επιτρέπει στο χρήστη να καθορίζει το URL της ιστοσελίδας που θέλει να επισκεφτεί ή του αρχείου που θέλει να «κατεβάσει». Επιπλέον, επιτρέπει, εκτός από την εκκίνηση της εν λόγω υπηρεσίας, τον τερματισμό της τρέχουσας λήψης δεδομένων, την παρουσίαση πληροφοριών σχετικά με τη σύνδεση που είναι σε εξέλιξη και την τροποποίηση των παραμέτρων που σχετίζονται με τις προτιμήσεις του χρήστη.

Ανάλογα με το bandwidth που επιθυμεί να δεσμεύσει ο χρήστης και, κατά συνέπεια, ανάλογα με την ταχύτητα σύνδεσης που θεωρεί ότι ικανοποιεί τις ανάγκες του, μπορεί να διαλέξει ανάμεσα στα εξής 5 επίπεδα ποιότητας (Πίνακας 3.7):

Web Browsing	
QoS level	Bitrate (Kbps)
QoS-1	32
QoS-2	64
QoS-3	128
QoS-4	384
QoS-5	1024

Πίνακας 3.7

Αναφέρουμε, ενδεικτικά, για τα bitrates του παραπάνω πίνακα ότι: Η ταχύτητα 32Kbps είναι μια τυπική ταχύτητα που προσφέρεται από μια απλή PSTN dial-up σύνδεση. Ομοίως, η ταχύτητα 64Kbps προσφέρεται τυπικά από μια ISDN-64 σύνδεση, η ταχύτητα 128Kbps από μια ISDN-128 σύνδεση, η ταχύτητα 384Kbps από μια DSL-384 σύνδεση, ενώ η ταχύτητα 1024Kbps από μια σύνδεση τύπου DSL-1024 ή T1. Οι τρεις πρώτες μπορούν να ικανοποιηθούν από ένα GSM/GPRS σύστημα, ενώ οι υπόλοιπες τεχνολογίες που μας έχουν απασχολήσει μπορούν να προσφέρουν όλα τα δυνατά επίπεδα ποιότητας που θεωρήσαμε.

### 3.1.3 Κεντρικό Μενού

Κάθε μία από τις υπηρεσίες που παρουσιάστηκαν στα παραπάνω, μπορεί να επιλεγεί από το χρήστη μέσω του κεντρικού μενού της εφαρμογής, το οποίο περιλαμβάνει και ένα διαφορετικό εικονίδιο για κάθε υπηρεσία. Ο χρήστης μπορεί να πλοηγηθεί εύκολα σε αυτό το μενού, χρησιμοποιώντας τα βελάκια ή το ποντίκι, και να επιλέξει την επιθυμητή υπηρεσία. Κάθε επιλεγμένο εικονίδιο γίνεται αυτόματα highlighted.



Στην προηγούμενη παράγραφο, έχουμε ήδη γνωρίσει τη λειτουργία των 7 από τα 9 κουμπιά/εικονίδια του μενού αυτού. Εν συνεχεία, θα αναφερθούμε περιληπτικά και στη λειτουργικότητα των άλλων 2.

Μέσω του κεντρικού μενού, λοιπόν, πέραν της επιλογής μιας υπηρεσίας, ο χρήστης έχει επιπλέον τη δυνατότητα να δει συνοπτικά πληροφορίες σύνδεσης για κάθε μία από τις τρέχουσες υπηρεσίες. Αυτό μπορεί να το κάνει εύκολα, επιλέγοντας το τελευταίο από τα εικονίδια του μενού, όπως φαίνεται και στο screenshot που ακολουθεί.



Το παράθυρο που ανοίγει πληροφορεί το χρήστη για το ποιες υπηρεσίες τρέχουν, όπως, επίσης, και για το πού έχει ανατεθεί η εξυπηρέτηση καθεμιάς υπηρεσίας, όπως δείχνει και το ακόλουθο screenshot.

Fri Apr 01 11:46:31 EEST 2005

Running Services

Service	Technology	Provider	Quality
Voice Call	WLAN	QTelecom	3
Web Browsing	WLAN	QTelecom	3
Video Stream	GSM	Cosmote	2

Back

Όπως είναι, επίσης, φανερό, κεντρικό ρόλο στο εν λόγω μενού κατέχει το κουμπί που οδηγεί το χρήστη στο μενού επιλογής και τροποποίησης των προφίλ, δηλαδή των σετ ρυθμίσεων που αντανακλούν τις προτιμήσεις του χρήστη, όσον αφορά τον τρόπο σύνδεσης και παροχής των υπηρεσιών. Το κουμπί αυτό βρίσκεται στο κέντρο του μενού για ευκολότερη και ταχύτερη πρόσβαση.

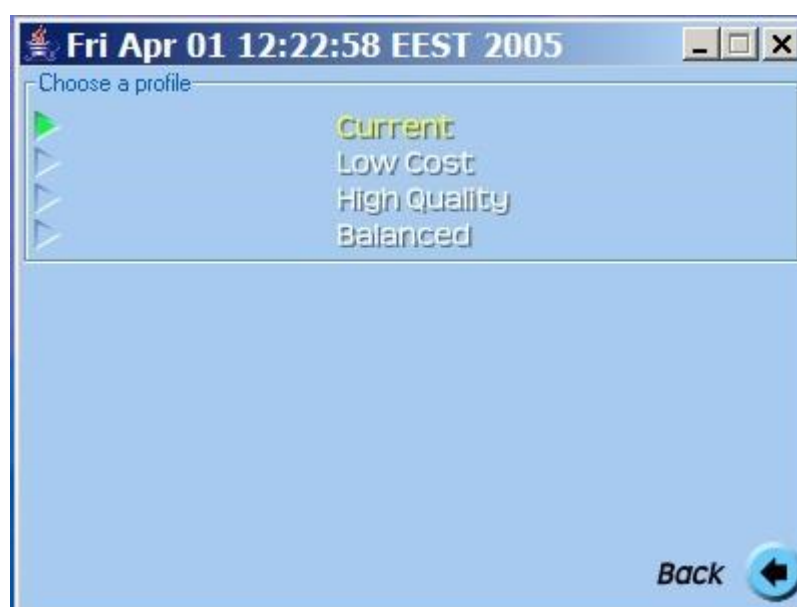


Στο ρόλο και τη λειτουργία αυτών των προφίλ θα αναφερθούμε στην επόμενη παράγραφο.

### 3.2 Διαμόρφωση και Αποθήκευση Προφίλ Χρήστη

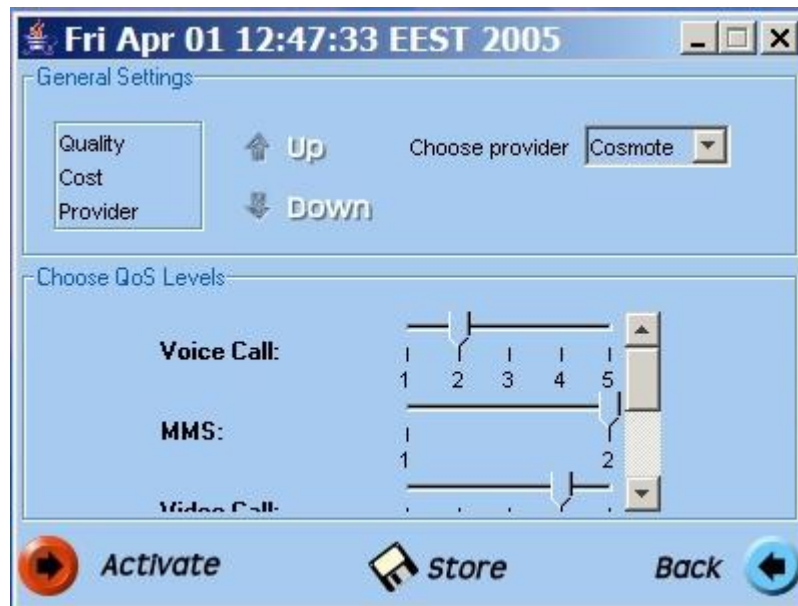
Η ύπαρξη και λειτουργία των προφίλ χρήστη ικανοποιεί ακριβώς την ανάγκη, που είχε τονιστεί και παραπάνω, απόδοσης, δηλαδή, κεντρικού χαρακτήρα στις προτιμήσεις του χρήστη, κατά τη διαδικασία λήψης της τελικής απόφασης για τον τρόπο σύνδεσης. Ας δούμε, τώρα, το πώς ακριβώς ο χρήστης καθορίζει τις παραμέτρους εκείνες μέσω των οποίων εκφράζονται οι προτιμήσεις του.

Αρχικά, μόλις ο χρήστης επιλέξει από το κεντρικό μενού το εικονίδιο «Επεξεργασία Προφίλ», οδηγείται σε μία νέα οθόνη, όπου καλείται να επιλέξει ένα από τα διαθέσιμα προφίλ, ένα από τα διαθέσιμα, δηλαδή, σεντ ρυθμίσεων, έτσι ώστε είτε να το ενεργοποιήσει, είτε να το τροποποιήσει, είτε και τα δύο. Στην παρακάτω εικόνα, βλέπουμε ότι ο χρήστης παροτρύνεται να επιλέξει ένα από 4 διαθέσιμα προφίλ («Current», «Low Cost», «High Quality», «Balanced» προφίλ).



Ο λόγος ύπαρξης πολλών προφίλ εξυπηρετεί ακριβώς τη διευκόλυνση του χρήστη, καθώς έτσι μπορεί να διατηρεί διαφορετικά σεντ ρυθμίσεων και να τα «φορτώνει» ταχύτερα στην εφαρμογή, κατά βούληση, χωρίς να αναγκάζεται να τροποποιεί κάθε φορά ένα και μοναδικό προφίλ. Ειδικά, το προφίλ με όνομα «Current» (τρέχον προφίλ) έχει και μια ξεχωριστή σημασία, καθώς περιέχει το σεντ των τρεχουσών ρυθμίσεων. Βέβαια, οποιοδήποτε και από τα υπόλοιπα προφίλ μπορεί να γίνει το τρέχον.

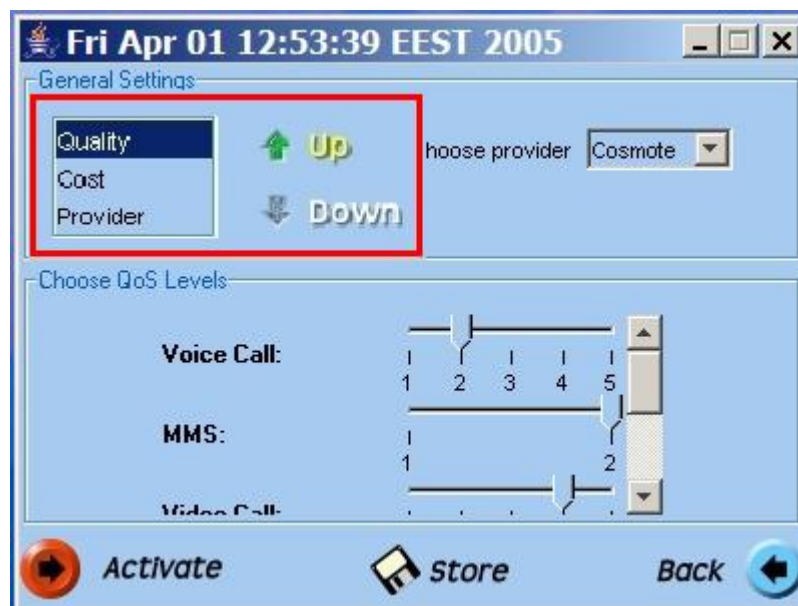
Προχωρούμε τώρα στον πυρήνα της γραφικής διεπαφής, που δεν είναι άλλος από το πάνελ, μέσα από το οποίο ο χρήστης δηλώνει τις προτιμήσεις του, καθορίζοντας τις αντίστοιχες παραμέτρους. Ας δούμε, αρχικά, ένα screenshot του πάνελ, προκειμένου να προχωρήσουμε στη σταδιακή επεξήγηση καθενός από τα συστατικά του.



Όπως βλέπουμε, το πάνελ αυτό χωρίζεται σε δύο τμήματα. Το τμήμα των γενικών ρυθμίσεων («*General Settings*») και το τμήμα καθορισμού της ποιότητας καθεμιάς υπηρεσίας χωριστά («*Choose QoS Levels*»). Θα προχωρήσουμε στη μελέτη καθενός τμήματος χωριστά.

- Τμήμα Γενικών Ρυθμίσεων:

Στα αριστερά αυτού του τμήματος, ο χρήστης καλείται να ιεραρχήσει τη βαρύτητα που έχουν γι' αυτόν τρεις εξαιρετικά σημαντικές, όπως θα δούμε και στη συνέχεια, παράμετροι: η ποιότητα (*Quality*), το κόστος (*Cost*) και ο πάροχος (*Provider*).



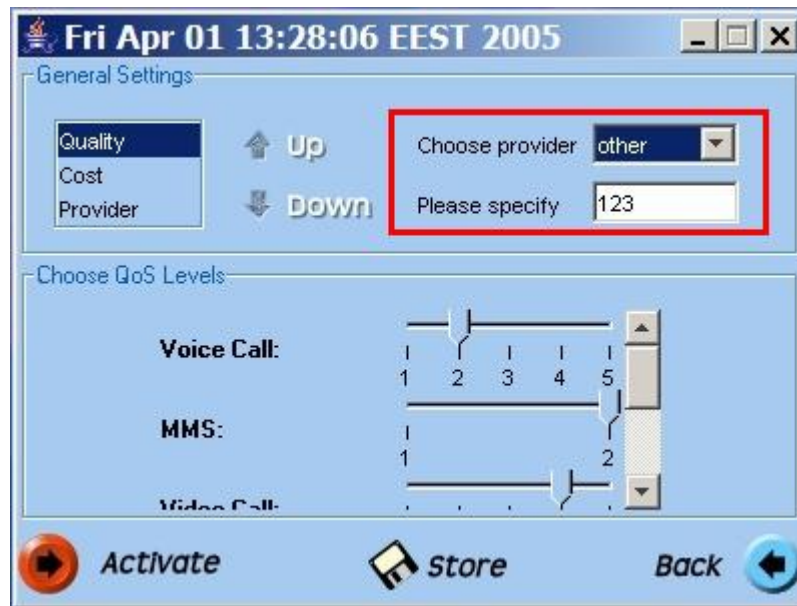
Ταξινομώντας τους τρεις αυτούς παράγοντες, με τη βοήθεια των κουμπιών «Up» και «Down», τους αποδίδει και την αντίστοιχη βαρύτητα, τη μαθηματική ερμηνεία της οποίας θα εξετάσουμε αναλυτικά στο επόμενο κεφάλαιο (κεφάλαιο 4). Έτσι, για παράδειγμα, στην παραπάνω περίπτωση, ο πιο σημαντικός παράγοντας είναι για το χρήστη η ποιότητα, έπεται το κόστος και, τέλος, ο πάροχος.

Η ύπαρξη του παρόχου ως μιας εκ των τριών παραμέτρων γίνεται, προκειμένου να δοθεί στο χρήστη η δυνατότητα να δηλώσει την προτίμησή του, όσον αφορά τη λήψη των υπηρεσιών από ένα συγκεκριμένο πάροχο. Τον πάροχο αυτόν τον δηλώνει στο δεξί τμήμα των γενικών ρυθμίσεων, επιλέγοντας το όνομά του από το drop-down box. Αξίζει να σημειωθεί ότι η προτίμηση που αφορά στον πάροχο λαμβάνεται υπόψη (έχοντας «θετική» συνεισφορά) μόνο για εκείνα τα υποψήφια σημεία πρόσβασης, εντός της εμβέλειας του κινητού, που ανήκουν στον προτιμώμενο πάροχο. Διαφορετικά, δηλαδή για τα υπόλοιπα σημεία πρόσβασης, δε διαδραματίζει κάποιο ρόλο.



Πέραν από τους προτεινόμενους παρόχους, στο χρήστη παρέχονται και δύο επιπλέον επιλογές. Η μία είναι η επιλογή «none», μέσω της οποίας δηλώνει ότι δεν τον ενδιαφέρει κάποιος συγκεκριμένος πάροχος, ενώ η άλλη είναι η επιλογή «other», μέσω της οποίας μπορεί να δηλώσει ως προτιμώμενο έναν πάροχο, ο οποίος δε συμπεριλαμβάνεται στη λίστα με τους προτεινόμενους παρόχους. Στην περίπτωση αυτή, εμφανίζεται ένα επιπλέον πεδίο κειμένου, στο οποίο ο χρήστης παροτρύνεται να εισάγει τον κωδικό αριθμό του παρόχου που επιθυμεί (μοναδικός για κάθε πάροχο). Έτσι, λοιπόν, δίνεται η δυνατότητα, εάν ο προτιμώμενος από το χρήστη πάροχος δε συμπεριλαμβάνεται στη λίστα παρόχων της συσκευής, να δηλωθεί δυναμικά από το χρήστη. Φυσικά, στην περίπτωση αυτή, απαραίτητο είναι ο χρήστης να έχει πληροφορηθεί προηγουμένως τον κωδικό αριθμό του παρόχου που τον ενδιαφέρει.





- Τμήμα Καθορισμού Ποιότητας Υπηρεσιών:

Στο τμήμα αυτό, ο χρήστης καλείται να καθορίσει το επίπεδο ποιότητας που επιθυμεί για κάθε μία από τις υπηρεσίες που εξετάσαμε παραπάνω, υπό την προϋπόθεση, φυσικά, να διαθέτουν περισσότερα από ένα επίπεδα ποιότητας. Πρέπει να επισημανθεί ότι το επίπεδο αυτό που δηλώνει ο χρήστης είναι το μέγιστο επίπεδο ποιότητας που επιθυμεί να λαμβάνει. Η ανάγκη καθορισμού της μέγιστης ποιότητας προκύπτει προκειμένου ο χρήστης να εκφράσει, αλλά και να ελέγξει, το μέγιστο κόστος το οποίο προτίθεται να πληρώσει για κάθε υπηρεσία, και το οποίο είναι ανάλογο του αντίστοιχου επιπέδου ποιότητας.

Για παράδειγμα, στο screenshot που ακολουθεί το επιλεγμένο μέγιστο επίπεδο ποιότητας για την υπηρεσία Video Call είναι το επίπεδο 4. Αυτό σημαίνει ότι αποκλείεται η παροχή της υπηρεσίας αυτής σε υψηλότερο επίπεδο, χωρίς ωστόσο να αποκλείεται η παροχή της προαναφερθείσας υπηρεσίας σε χαμηλότερο επίπεδο, αν κάτι τέτοιο επιβληθεί από τις συνθήκες του δικτύου τη δεδομένη στιγμή ή από τη διαδικασία βελτιστοποίησης που ακολουθεί η εφαρμογή μας, προκειμένου να λάβει την τελική απόφαση. Ως εκ τούτου, ο χρήστης μπορεί να λαμβάνει την υπηρεσία Video Call σε επίπεδο ποιότητας 1, 2, 3 ή 4, ανάλογα με το τι είναι βέλτιστο εκείνη τη στιγμή.



Αξίζει, ίσως, να αποσαφηνισθεί ότι, στο τμήμα αυτό, ο χρήστης επιλέγει το μέγιστο και όχι το ελάχιστο επιθυμητό επίπεδο ποιότητας ανά υπηρεσία. Αυτό είναι λογικό, αν σκεφτούμε ότι σε ένα συμβόλαιο θα καθοριζόταν το μέγιστο και όχι το ελάχιστο επίπεδο ποιότητας που θα μπορούσε να λάβει ο χρήστης / συνδρομητής. Εξάλλου, το ελάχιστο επίπεδο ποιότητας καθορίζεται όχι από τις προτιμήσεις του χρήστη, αλλά από τις συνθήκες του δικτύου τη δεδομένη στιγμή της αίτησης χρήσης της υπηρεσίας. Αντίθετα, ο καθορισμός του μέγιστου επιπέδου ποιότητας είναι ιδιαίτερα χρήσιμος, αφού διευκολύνει το χρήστη να θέσει ένα όριο (ένα «πλαφόν») στο κόστος που είναι διατεθειμένος να επωμιστεί.

- Αποθήκευση ή/και Ενεργοποίηση:

Έχοντας αναλύσει με λεπτομέρεια τα δύο βασικά τμήματα του πάνελ, το μόνο που απομένει είναι να εξετάσουμε τις ενέργειες στις οποίες μπορεί να προβεί ο χρήστης, αφού ολοκληρώσει τη διαμόρφωση του προφίλ του.

Το τελικό, λοιπόν, βήμα είναι η αποθήκευση τυχόν αλλαγών στις οποίες προέβη ο χρήστης. Αυτό μπορεί να γίνει από την εύχρηστη γραμμή κουμπιών που έχει τοποθετηθεί στο κάτω μέρος του πάνελ. Συγκεκριμένα, με επιλογή του κουμπιού «Store», ο χρήστης αποθηκεύει τις επιλογές του στο προφίλ το οποίο έχει επιλέξει.





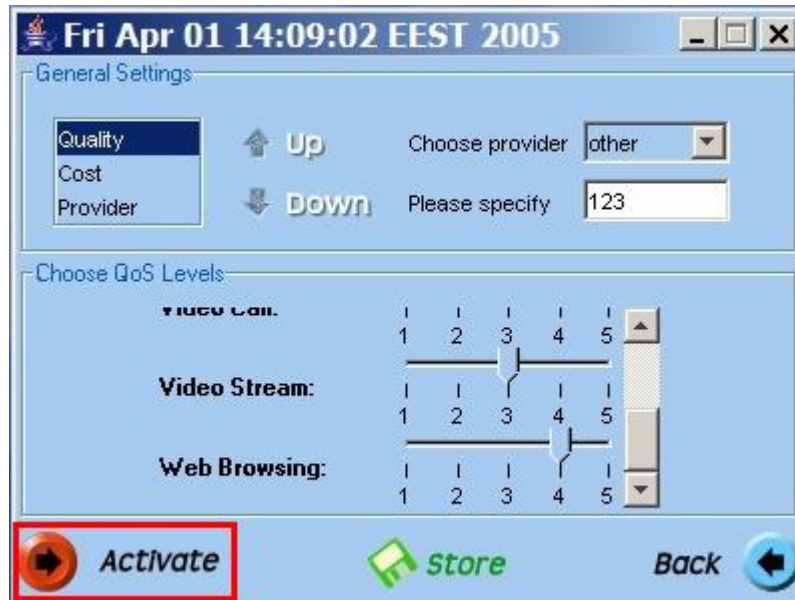
Σε αυτό το σημείο, αξίζει να σταθούμε στον τρόπο με τον οποίο κωδικοποιούνται αυτές οι προτιμήσεις, προκειμένου να αποθηκευτούν. Η αποθήκευση γίνεται σε ένα properties file, σε ένα αρχείο, δηλαδή, όπου κάθε καταχώριση είναι ένα ζεύγος της μορφής:

*name = value*

Για παράδειγμα, για την περίπτωση του παραπάνω screenshot, το αντίστοιχο properties file θα έχει ως εξής:

```
#The 3 main factors
#factor1 is the most important one, factor3 is the least important
factor1=Quality
factor2=Cost
factor3=Provider
#Specify the provider
providerName=other
providerCode=123
#QoS Settings
#Max QoS for Voice Call Service = 3
1=3
#Max QoS for SMS Service = 1 - Not Editable
2=1
#Max QoS for MMS Service = 2
3=2
#Max QoS for Email Service = 1 - Not Editable
4=1
#Max QoS for Video Call Service = 4
5=4
#Max QoS for Voice Stream Service = 3
6=3
#Max QoS for Web Browsing Service = 4
7=4
```

Τέλος, αν ο χρήστης επιθυμεί να χρησιμοποιήσει το προφίλ που διαμόρφωσε ως τρέχον προφίλ, το μόνο που έχει να κάνει είναι να πατήσει το κουμπί «*Activate*», οπότε και ενεργοποιούνται οι τρέχουσες ρυθμίσεις. Οι διαδικασίες βελτιστοποίησης που εκκινούνται με το πάτημα αυτού του κουμπιού θα εξεταστούν σε άλλη παράγραφο.



## 4. Ευφυής Επιλογή Δικτύου Πρόσβασης

### 4.1 Περιγραφή του Προβλήματος Ευφυούς Επιλογής Δικτύου Πρόσβασης

#### 4.1.1 Σκοπός του IIS module

Το βασικότερο συστατικό της αρχιτεκτονικής διαχείρισης ενός τερματικού είναι το IIS module, σκοπός του οποίου υπενθυμίζουμε ότι είναι η ευφυής επιλογή δικτύου πρόσβασης, με άλλα λόγια η επιλογή, μέσα από μια διαδικασία βελτιστοποίησης, της τεχνολογίας και του σημείου πρόσβασης μέσω των οποίων θα παρέχεται μια υπηρεσία. Το IIS module, δηλαδή, είναι επιφορτισμένο με το χειρισμό γεγονότων που πηγάζουν είτε από το ίδιο το τερματικό είτε από το δίκτυο και αφορούν την αίτηση για μια νέα υπηρεσία ή το αίτημα για επανεξέταση του τρόπου σύνδεσης μιας τρέχουσας υπηρεσίας.

#### 4.1.2 Εισαγωγή στο Πρόβλημα

Πριν επεκταθούμε περαιτέρω στην περιγραφή της λειτουργικότητας του IIS module, είναι σημαντικό να διασαφηνίσουμε τη δικαιοδοσία αφενός του τερματικού και αφετέρου του δικτύου ως προς τη λήψη και επιβολή μιας απόφασης διαπομπής (handover). Η πλήρης προσέγγιση της διαδικασίας διαπομπής ακολουθεί μια πολιτική που απαρτίζεται από δύο βασικές αρχές:

- Πρώτον, το τερματικό έχει τη δυνατότητα να μεταβαίνει σε διαφορετικούς παρόχους δικτύου, ανεξάρτητα από την επιθυμία της διαχειριστικής οντότητας δικτύου. Το κέντρο λήψης μιας τέτοιας απόφασης είναι το τερματικό και η εν λόγω απόφαση υπερτερεί οποιασδήποτε άλλης.
- Δεύτερον, ο πάροχος δικτύου έχει τη δυνατότητα να μεταφέρει τη σύνδεση σε άλλο σημείο πρόσβασης, εντός της δικής του διαχειριστικής περιοχής (network domain), θεωρώντας πως ο χρήστης εμπιστεύεται ότι το δίκτυο θα ακολουθήσει κάποιους συμφωνημένους όρους.

Προχωρούμε, αμέσως μετά, στην περιγραφή του προβλήματος της ευφυούς επιλογής δικτύου πρόσβασης.

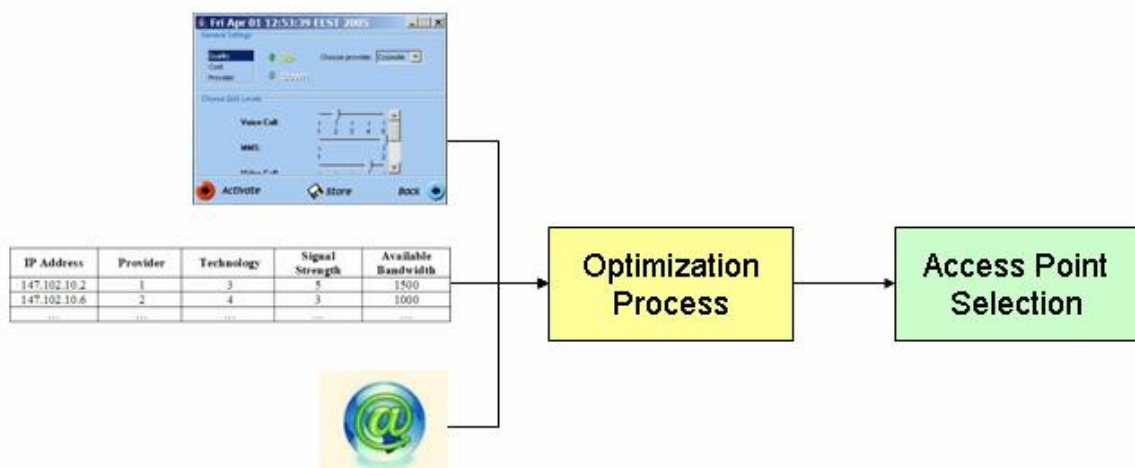
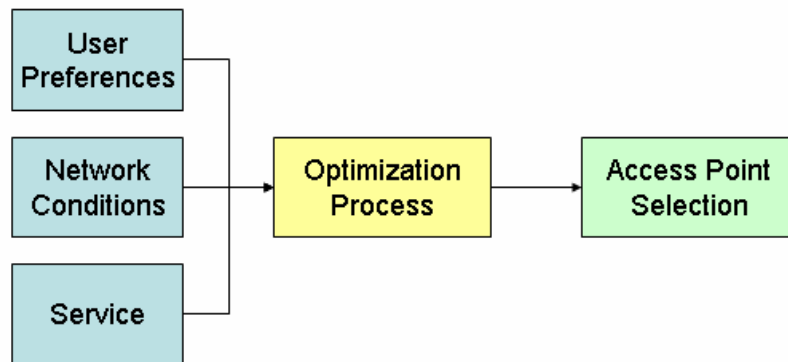
#### 4.1.3 Γενική Περιγραφή του Προβλήματος και Σχηματική Αναπαράσταση

Ένα σημαντικό χαρακτηριστικό της αρχιτεκτονικής διαχείρισης τερματικού που περιγράφηκε στα προηγούμενα είναι η δυνατότητα του χρήστη να τροποποιεί δυναμικά το προφίλ του, δηλαδή το σετ ρυθμίσεων που χρησιμοποιείται ως είσοδος στον αλγόριθμο βελτιστοποίησης του IIS. Η γραφική διεπαφή, που έχει παρουσιαστεί λεπτομερώς στο προηγούμενο κεφάλαιο, μέσω της οποίας ο χρήστης καθορίζει τις προτιμήσεις του, επιτρέπει να δίνεται διαφορετική βαρύτητα στις παραμέτρους που επηρεάζουν τη λήψη της τελικής απόφασης σχετικά με την επιλογή του τρόπου σύνδεσης. Οι παράμετροι αυτοί, όπως είδαμε, περιλαμβάνουν το κόστος, τον πάροχο

και το επίπεδο ποιότητας που προσφέρεται από καθένα από τα υποψήφια access points τα οποία ο αλγόριθμος του IIS θεωρεί ως πιθανούς προορισμούς διαπομπής. Το προφίλ αυτό του χρήστη, όμως, αποτελεί μία μόνο από τις εισόδους του αλγορίθμου του IIS.

Το πρόβλημα της βελτιστοποίησης που αντιμετωπίζεται από το IIS module βασίζεται, επίσης, στις εξής εισόδους: Πρώτον, σε μία ομάδα L2 μετρήσεων, οι οποίες αντανakλούν την ποιότητα του σήματος από καθένα από τα διαθέσιμα access points, όπως αυτή λαμβάνεται από το κινητό τερματικό. Δεύτερον, την υπηρεσία την οποία αφορά αυτή η διαδικασία διαπομπής, καθώς και τα επίπεδα ποιότητας στα οποία μπορεί, γενικά, να διατεθεί.

Η απόφαση του αλγορίθμου του IIS, σχετικά με το κατάλληλο access point μέσω του οποίου η υπηρεσία αυτή μπορεί να ληφθεί πιο αποτελεσματικά, πρέπει να βελτιστοποιεί μια αντικειμενική συνάρτηση, η οποία σχετίζεται με τα βάρη που αποδίδονται σε καθένα από τα διαφορετικά κριτήρια επιλογής. Η διαδικασία αυτή βελτιστοποίησης παρουσιάζεται συνοπτικά στα δύο παρακάτω σχήματα.



#### 4.1.4 Είδη Διαπομπής

Ας εξετάσουμε, εν συντομία, στο σημείο αυτό, τα είδη διαπομπής που μπορούν να προκύψουν σε ένα τηλεπικοινωνιακό σύστημα 4<sup>ης</sup> γενιάς και τα πλεονεκτήματα της αντιμετώπισής του από το IIS.

- Intra-domain, intra-technology handover:

Το είδος αυτό αναφέρεται στη διαπομπή που μπορεί να συμβεί ανάμεσα σε σημεία πρόσβασης που ανήκουν στην ίδια τεχνολογία και στον ίδιο διοικητικό τομέα (administrative domain). Όταν ένας πάροχος διαθέτει και διαχειρίζεται ένα μεγάλο δίκτυο της ίδιας τεχνολογίας, στην ίδια περιοχή, το τερματικό μπορεί να κληθεί να πραγματοποιήσει μια intra-technology διαπομπή, σε περίπτωση που ανιχνευτεί σημαντική υποβάθμιση του σήματος. Αυτή η εντολή για διαπομπή θα προέλθει από το IIS module, το οποίο είναι σε θέση να ανιχνεύει, κάθε στιγμή, τέτοιες διακυμάνσεις σήματος, μέσω του interface του προς τη μεριά του Interface Adaptation module. Για να διασφαλισθεί ότι οι αποφάσεις αυτές, που λαμβάνονται τοπικά στο κινητό τερματικό από το IIS module, δεν είναι αντίθετες με τις διαχειριστικές προτεραιότητες του δικτύου, το IIS, εναλλακτικά, μπορεί να παρέχει μια ταξινομημένη λίστα των προτεινόμενων access points, αντί μιας απευθείας εντολής. Η λίστα αυτή δεν είναι ανάγκη να προωθείται ολόκληρη, αλλά μόνο στοιχείο προς στοιχείο, μέχρις ότου η οντότητα διαχείρισης του δικτύου κάνει αποδεκτή την πρόταση του τερματικού. Η προσέγγιση αυτή, με τα προφανή πλεονεκτήματά της ως προς την καλύτερη διαχείριση, θα ακολουθηθεί και στην υλοποίησή μας.

- Intra-domain, inter-technology handover:

Ο όρος αυτός περιγράφει τη διαπομπή μεταξύ διαφορετικών τεχνολογιών, αλλά εντός του ίδιου διοικητικού τομέα. Όταν, δηλαδή, ένας πάροχος διαχειρίζεται περισσότερες από μία τεχνολογίες, στην ίδια περιοχή, το τερματικό καλείται και πάλι, σε περιπτώσεις σημαντικής υποβάθμισης σήματος, να πραγματοποιήσει μια διαπομπή, μεταβαίνοντας πιθανώς στη χρήση άλλης τεχνολογίας. Όπως και στην προηγούμενη περίπτωση, το IIS module είναι αυτό που θα δώσει την οδηγία για μετάβαση στην κατάλληλη τεχνολογία και στο κατάλληλο access point, προαιρετικά συμβουλευόμενο και την οντότητα διαχείρισης του δικτύου για έγκριση της απόφασής του. Στην περίπτωση της intra-domain, inter-technology διαπομπής, γίνεται ακόμη πιο εμφανής η ανάγκη για αυτονομία του κινητού τερματικού στη λήψη της απόφασης, καθώς, αν αυτή η απόφαση προερχόταν από το δίκτυο, θα έπρεπε να λαμβάνει υπόψη της τις τεχνολογίες πρόσβασης που υποστηρίζονται από το κινητό τερματικό. Το πρόβλημα αυτό λύνεται εύκολα με την ανάθεση της λήψης της απόφασης στη μεριά του τερματικού, καθώς αυτό προτείνει μόνο σημεία πρόσβασης των τεχνολογιών που το ίδιο υποστηρίζει.

- Inter-domain, intra-technology handover:

Η διαπομπή αυτού του είδους πραγματοποιείται ανάμεσα σε διαφορετικούς παρόχους, που υποστηρίζουν όμως την ίδια τεχνολογία. Και αυτός ο τύπος διαπομπής

αντιμετωπίζεται από το IIS, αφού οι προτιμήσεις του χρήστη, όσον αφορά τους παρόχους του δικτύου, έχουν εισαχθεί και είναι γνωστές στο κινητό τερματικό, μέσω της γραφικής διεπαφής χρήστη (Graphical User Interface – GUI). Αυτή η προσέγγιση είναι προτιμότερη από το να θεωρούμε ότι η οντότητα διαχείρισης του δικτύου επιβάλλει διαπομπές ανάμεσα σε διαφορετικούς παρόχους, γιατί σε αυτήν την περίπτωση η συγκεκριμένη οντότητα θα έπρεπε να ανήκει όχι σε ένα συγκεκριμένο πάροχο δικτύου, αλλά σε μία ουδέτερη διαχειριστική αρχή, η οποία θα όφειλε να εγγυηθεί αντικειμενικότητα. Και αυτό, γιατί σπάνια ένας πάροχος θα επέτρεπε τη μετάβαση των χρηστών του σε άλλον πάροχο.

- Inter-domain, inter-technology handover:

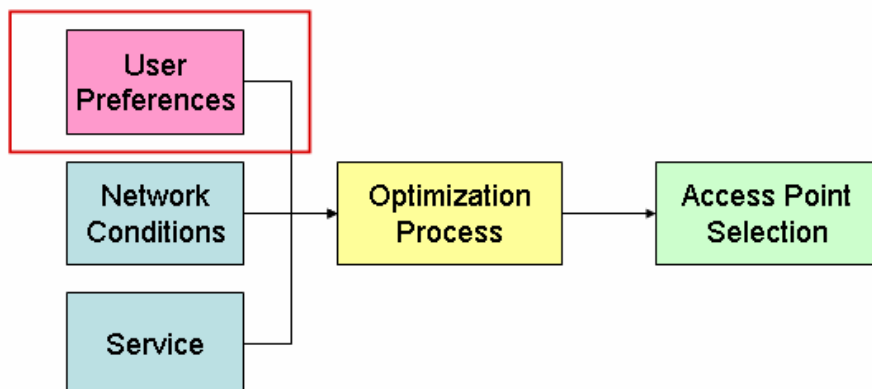
Ο όρος αυτός αναφέρεται στη διαπομπή ανάμεσα σε διαφορετικές τεχνολογίες και διαφορετικούς διαχειριστικούς τομείς. Αυτό το είδος διαπομπής αντιμετωπίζεται επίσης από το IIS. Η υιοθέτηση μιας τέτοιας προσέγγισης είναι και πάλι καλύτερη συγκρινόμενη με μια δικτυο-κεντρική, αφού επιλύει τόσο τα προβλήματα που εμφανίζονται σε μια intra-domain, inter-technology διαπομπή, όσο και αυτά που προκύπτουν σε μια inter-domain, intra-technology διαπομπή.

Αξίζει να τονίσουμε ότι για κάθε είδος διαπομπής η αντιμετώπιση του IIS module είναι η ίδια, η υλοποίησή του, δηλαδή, είναι ανεξάρτητη από το είδος της διαπομπής που επίκειται να συμβεί.

#### 4.1.5 Στοιχεία Εισόδου

Επανερχόμενοι στην περιγραφή του προβλήματός μας, θα εξετάσουμε με περισσότερη λεπτομέρεια τα στοιχεία εισόδου του αλγορίθμου του IIS. Σε γενικές γραμμές, τα στοιχεία αυτά τα έχουμε ήδη σκιαγραφήσει στην παράγραφο 4.1.3 και έχουμε δει ότι πρόκειται για το προφίλ του χρήστη, τις συνθήκες του δικτύου και την υπηρεσία που μας ενδιαφέρει.

##### 4.1.5.1 Πρώτο στοιχείο εισόδου – Το προφίλ χρήστη



Στο κεφάλαιο 3, έχουμε αναφερθεί διεξοδικά στη μορφή του προφίλ του χρήστη ενός τερματικού, στον τρόπο διαμόρφωσής του, μέσω της γραφικής διεπαφής, και στον τρόπο αναπαράστασης και αποθήκευσής του στο εσωτερικό της συσκευής. Στο ίδιο κεφάλαιο, είδαμε ότι οι παράμετροι ενός τέτοιου προφίλ χωρίζονται σε δύο κατηγορίες.

➤ Κατηγορία 1<sup>η</sup> – Γενικές ρυθμίσεις:

Στην πρώτη, ο χρήστης καλείται να ταξινομήσει τους τρεις βασικούς παράγοντες μιας σύνδεσης: την «ποιότητα», το «κόστος» και τον «πάροχο».

✓ Η «ποιότητα»:

Η «ποιότητα» αντικατοπτρίζει, κατά κύριο λόγο, το εύρος ζώνης (bandwidth) που θα αφιερωθεί για την παροχή της υπηρεσίας. Πέρα, όμως, από το εύρος ζώνης, σημαντικό ρόλο διαδραματίζει, όπως είναι φυσικό, και η ποιότητα (ισχύς) του λαμβανόμενου σήματος. Ο τρόπος με τον οποίο η παράμετρος αυτή θα ληφθεί υπόψη κατά τη διαδικασία βελτιστοποίησης θα εξεταστεί παρακάτω.

✓ Το «κόστος»:

Ο παράγοντας «κόστος» αντιστοιχεί στο κόστος χρήσης της υπηρεσίας και, γενικά, εξαρτάται από τον τύπο της υπηρεσίας, το επίπεδο ποιότητας στο οποίο λαμβάνεται, το είδος της τεχνολογίας και τον πάροχο από τα οποία διατίθεται. Στο σημείο αυτό, ανακύπτει ένα σημαντικό πρόβλημα: Με ποιον τρόπο πρέπει να προσδιορισθεί το κόστος, ώστε να ανταποκρίνεται στην πραγματικότητα, και με ποιον τρόπο πρέπει αυτό να γνωστοποιηθεί στο κινητό τερματικό, προκειμένου να χρησιμοποιηθεί κατά τη διαδικασία βελτιστοποίησης;

Θα μελετήσουμε αρκετά εναλλακτικά σενάρια επίλυσης του προβλήματος αυτού, προκειμένου να καταλήξουμε στο επικρατέστερο.

💡 **Ιδέα:** Να δημιουργηθεί ένας πίνακας με κόσθη, τέτοιος ώστε να λαμβάνει υπόψη τις διαφοροποιήσεις του κόστους ανά υπηρεσία, ανάλογα με τον τύπο της τεχνολογίας που χρησιμοποιείται. Ένας τέτοιος πίνακας θα έχει την ακόλουθη, γενικά, μορφή:

		Technology			
		GSM	UMTS	WLAN	DVB
Φωνή	QoS-1	0	1	1	2
	QoS-2	1	2	2	3
	QoS-3	2	3	3	4
Video	...	...	...	...	...
	...	...	...	...	...
...	...	...	...	...	...

✘ **Μειονεκτήματα:** Δεν υπάρχει μια γενική τάση στην αγορά που να μας υπαγορεύει τις τιμές με τις οποίες θα πρέπει να συμπληρώσουμε τον πίνακα. Εξάλλου, είναι συχνό φαινόμενο οι διαφοροποιήσεις από τόπο σε τόπο αλλά και

από χρόνο σε χρόνο. Επομένως, για να μπορέσει να υλοποιηθεί η ιδέα αυτή, θα πρέπει να βελτιωθεί.

💡 **Ιδέα:** Να θεωρηθεί ότι το κόστος ανά KByte είναι τόσο χαμηλότερο όσο μεγαλύτερο είναι το τυπικό εύρος ζώνης της κάθε τεχνολογίας. Για παράδειγμα, στο GPRS (τυπικές δυνατότητες όσον αφορά το bandwidth: 45Kbps) το ανά KByte κόστος θα είναι πολύ μεγαλύτερο από όσο στο WLAN (11Mbps).

✘ **Μειονεκτήματα:** Μια τέτοια διαβάθμιση δεν είναι σίγουρο ότι θα ισχύσει απόλυτα στην πράξη. Επίσης, ναι μεν υπάρχει διαφοροποίηση ανάλογα με τον τύπο της τεχνολογίας, αλλά δεν υπάρχει διαφοροποίηση ανάλογα και με τον τύπο της υπηρεσίας. Επομένως, η ακρίβεια των αποτελεσμάτων της μεθόδου αυτής θα μπορούσε να αμφισβητηθεί.

💡 **Ιδέα:** Να θεωρηθεί ότι το ανά KByte κόστος είναι σταθερό, ανεξαρτήτως της συγκεκριμένης τεχνολογίας που χρησιμοποιείται.

✘ **Μειονεκτήματα:** Κάτι τέτοιο δεν ισχύει στην πράξη και, ουσιαστικά, αναιρεί τον παράγοντα κόστος από την αντικειμενική συνάρτηση. Επιπλέον, διάκριση ανάλογα με την υπηρεσία δεν υπάρχει.

💡 **Ιδέα:** Το ανά KByte κόστος να λαμβάνεται μαζί με τις υπόλοιπες πληροφορίες (πάροχος, τύπος τεχνολογίας, ισχύς σήματος, διαθέσιμο εύρος ζώνης) από κάθε access point.

✘ **Μειονεκτήματα:** Ποια υπηρεσία θα αφορά το κόστος αυτό και ποιο QoS; Είναι φανερό ότι στη λύση αυτή δεν υπάρχει διαφοροποίηση ανάλογα με τον τύπο της υπηρεσίας (ούτε και ανάλογα με το QoS). Εκτός και αν αποστέλλεται από κάθε access point ένα σύνολο πληροφοριών που αφορούν το κόστος κάθε υπηρεσίας για όλα τα QoS levels. Κάτι τέτοιο, όμως, θα υπερφόρτωνε το δίκτυο με πλεονάζουσα ως επί το πλείστον πληροφορία. Επίσης, η πληροφορία για το κόστος δεν αποτελεί L2 measurement.

💡 **Ιδέα:** Θα μπορούσε το τερματικό μας να λαμβάνει από κάποια οντότητα του δικτύου το «τιμολόγιο» όλων των υπηρεσιών και όλων των παρόχων. Η πληροφορία αυτή θα μπορούσε να είναι δομημένη σε XML μορφή, για εύκολη επεξεργασία. Έτσι, το τερματικό μας θα γνώριζε με ακρίβεια το κόστος κάθε υπηρεσίας, ανάλογα με το QoS, τον τύπο της τεχνολογίας αλλά ακόμη και τον πάροχο. Μάλιστα, η πληροφορία αυτή δεν είναι ανάγκη να λαμβάνεται κάθε φορά που ο χρήστης εκκινεί μια υπηρεσία, αλλά πολύ σπανιότερα, π.χ. μόνο μία φορά την ημέρα (αφού είναι πολύ σπάνιο να αλλάζουν δραστικά οι τιμές μέσα σε 24 ώρες) ή μόνο όταν το τερματικό κάνει power up. Ένα timestamp (χρονική σφραγίδα) μπορεί να καθορίζει για πόσο χρονικό διάστημα θα θεωρούνται έγκυρες οι πληροφορίες που έχει αποθηκευμένες το τερματικό στη μνήμη του. Με την παρέλευση του χρονικού αυτού διαστήματος, το τερματικό θα κάνει update των πληροφοριών του από το δίκτυο. Γενική δομή της πληροφορίας:



		Cosmote				Vodafone			TIM	
		GSM	UMTS	WLAN	DVB	GSM	UMTS	WLAN	...	...
Video	QoS-1									
	QoS-2									
	QoS-3									
E-mail	...									
	...									
...	...									

✘ **Μειονεκτήματα:** Ποια είναι η οντότητα του δικτύου που θα μπορούσε να αναλάβει αυτόν το ρόλο; Όταν ένας πάροχος μεταβάλει τις τιμές του, θα πρέπει να ενημερώνει πέρα από το δικό του σύστημα billing και τις συγκεκριμένες οντότητες του δικτύου.

☑ Παρά το ότι δε στερείται μειονεκτημάτων, η τελευταία αυτή ιδέα επικρατεί έναντι των άλλων, καθώς επιλύει προβλήματα που οι άλλες αδυνατούν να αντιμετωπίσουν επιτυχώς.

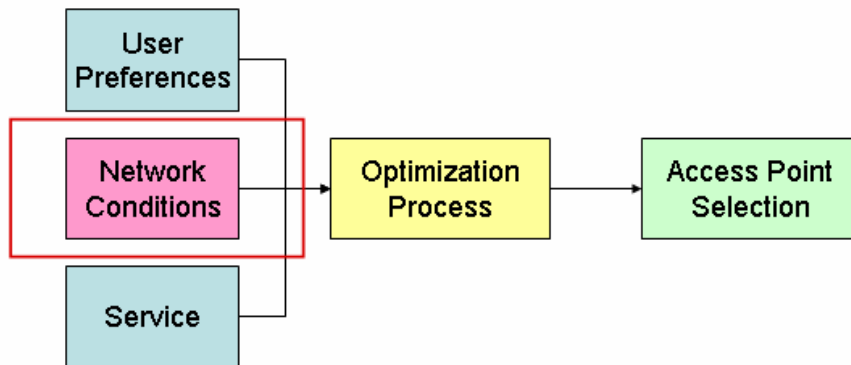
✓ Ο «πάροχος»:

Ο «πάροχος» εκφράζει τον προτιμώμενο από τη μεριά του χρήστη πάροχο της αιτούμενης υπηρεσίας. Η παράμετρος αυτή λαμβάνεται υπόψη μόνο όταν ο προτιμώμενος πάροχος συμπίπτει με τον πάροχο ενός access point. Αυτό σημαίνει ότι τα σημεία πρόσβασης που ανήκουν στον εν λόγω πάροχο θα πρέπει να «πριμοδοτηθούν». Το πριμ αυτό εξαρτάται από τη θέση του παράγοντα «πάροχος» στην ταξινομημένη λίστα του προφίλ.

➤ Κατηγορία 2<sup>η</sup> – Καθορισμός ποιότητας υπηρεσιών:

Όσον αφορά τη δεύτερη κατηγορία ρυθμίσεων του προφίλ, αυτή αναφέρεται στον καθορισμό των προτιμήσεων του χρήστη σχετικά με το μέγιστο επίπεδο ποιότητας που είναι διατεθειμένος να πληρώσει, για κάθε υπηρεσία. Όπως διαπιστώσαμε και στο κεφάλαιο 3, υπάρχουν πολλές υπηρεσίες που είναι δυνατό να διατεθούν σε παραπάνω από ένα επίπεδα ποιότητας. Για τις υπηρεσίες αυτές, ο χρήστης θα πρέπει να επιλέξει ποιο είναι το μέγιστο επίπεδο ποιότητας που τον ενδιαφέρει, ώστε να περιορίσει με τον τρόπο αυτό το μέγιστο κόστος που είναι διατεθειμένος να καταβάλει. Τονίζουμε εκ νέου ότι πρόκειται για το μέγιστο επιθυμητό επίπεδο ποιότητας και όχι για το ελάχιστο, αφού ούτως ή άλλως το τελευταίο δεν μπορεί να καθοριστεί παρά μόνο από τις συνθήκες του δικτύου. Δεν έχει νόημα, δηλαδή, ο χρήστης να επιλέξει π.χ. ελάχιστο επίπεδο ποιότητας 3 για μια υπηρεσία, αφού, σε περίπτωση που το δίκτυο δεν μπορεί να του παράσχει επίπεδο ποιότητας 3, είναι προτιμότερο να του δώσει την υπηρεσία σε επίπεδο ποιότητας 2 ή 1, παρά να αρνηθεί εντελώς την παροχή της. Από την άλλη, καθορίζοντας μέγιστο επίπεδο ποιότητας ίσο με 3 για μια υπηρεσία, εξασφαλίζει ότι δε θα χρεωθεί κόστος μεγαλύτερο από αυτό του επιπέδου 3.

#### 4.1.5.2 Δεύτερο στοιχείο εισόδου – Οι συνθήκες δικτύου

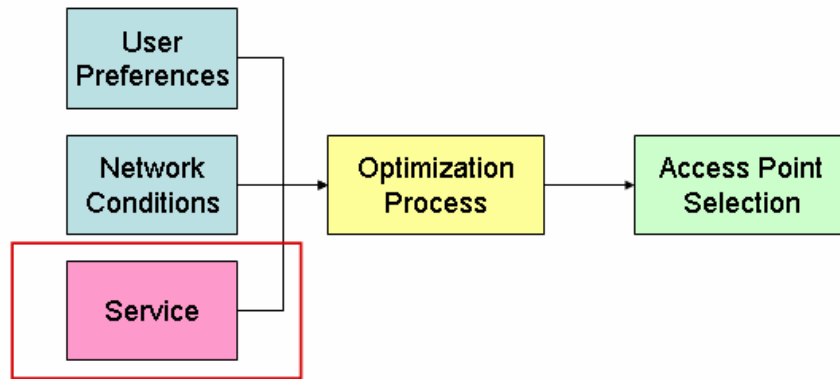


Οι συνθήκες που επικρατούν στο δίκτυο συνιστούν το δεύτερο στοιχείο εισόδου του αλγορίθμου του IIS.

Οι συνθήκες αυτές εισάγονται στο IIS υπό τη μορφή μετρήσεων που έχουν ληφθεί μέσω του Interface Adaptation module. Οι μετρήσεις αυτές οργανώνονται σε ομάδες πληροφοριών που αφορούν κάθε access point χωριστά. Συγκεκριμένα, για κάθε access point που «βλέπει» το κινητό τερματικό, υπάρχουν διαθέσιμες οι εξής πληροφορίες:

- *IP Διεύθυνση (IP Address)*: Μοναδική διεύθυνση που χαρακτηρίζει καθένα access point και χρησιμοποιείται για τη σύνδεση με αυτό. Μπορεί να είναι τύπου IPv4 ή IPv6.
- *Πάροχος Δικτύου (Network Provider)*: Η πληροφορία αυτή κωδικοποιείται με τη χρήση ενός μοναδικού κωδικού αριθμού, διαφορετικού για κάθε πάροχο.
- *Τύπος Τεχνολογίας Πρόσβασης (Access Technology)*: Η πληροφορία αυτή κωδικοποιείται και πάλι με τη χρήση ενός μοναδικού κωδικού αριθμού, διαφορετικού για κάθε τεχνολογία (π.χ. GSM, UMTS, κ.λπ.).
- *Ισχύς Λαμβανόμενου Σήματος (Signal Strength)*: Αντικατοπτρίζει την ποιότητα του σήματος που λαμβάνει το τερματικό από το συγκεκριμένο access point. Εκφράζεται σε μια αφηρημένη μορφή και, συγκεκριμένα, σε μια κλίμακα από 1 έως 5, με το 1 να αντιστοιχεί σε low signal, ενώ το 5 σε full signal strength.
- *Διαθέσιμο Εύρος Ζώνης (Available Bandwidth)*: Αντιστοιχεί στο εύρος ζώνης, σε Kbps, που είναι διαθέσιμο στο συγκεκριμένο access point τη στιγμή της μέτρησης.

#### 4.1.5.3 Τρίτο στοιχείο εισόδου – Η υπηρεσία



Το τρίτο και τελευταίο στοιχείο εισόδου στον αλγόριθμο του IIS είναι η υπηρεσία για την οποία ζητείται ο αλγόριθμος να τρέξει. Την υπηρεσία αυτή θα την αποκαλούμε από εδώ και στο εξής «αιτούμενη υπηρεσία». Ο προσδιορισμός της αιτούμενης υπηρεσίας είναι απαραίτητος, διότι τόσο το κόστος όσο και τα επίπεδα ποιότητας είναι διαφορετικά για κάθε μία υπηρεσία.

#### 4.1.6 Τράπεζα Πληροφοριών

Από την παραπάνω ανάλυση, γίνεται φανερό ότι είναι αναγκαία η πρόσβαση του τερματικού σε μια ομάδα πληροφοριών απαραίτητων για τη λειτουργικότητα της διαδικασίας βελτιστοποίησης. Οι πληροφορίες αυτές συγκεντρώνονται, με τη μορφή αρχείων, σε μια «κεντρική τράπεζα πληροφοριών» στο τερματικό, στην οποία θα μπορεί να έχει πρόσβαση κάθε sub-module που τις χρειάζεται.

Αναλυτικά, στην τράπεζα αυτή θα πρέπει να περιλαμβάνονται οι ακόλουθες πληροφορίες:

- Κατάλογος των υπηρεσιών που υποστηρίζει το τερματικό:

Ο κατάλογος αυτός εξυπηρετεί διπλό σκοπό. Αφενός μεν παρέχει στο λογισμικό του τερματικού, ανά πάσα στιγμή, τη γνώση για το πόσες και ποιες υπηρεσίες μπορεί να τρέξει το τερματικό και αφετέρου περιέχει την αντιστοίχιση κάθε υπηρεσίας με το μοναδικό κωδικό αριθμό που την χαρακτηρίζει. Για τυχόν προσθήκη νέας υπηρεσίας στις δυνατότητες του τερματικού (π.χ. σε περίπτωση αναβάθμισης), αρκεί μια ενημέρωση αυτού του αρχείου, χωρίς αλλαγές στον πηγαίο κώδικα του IIS module.

Το αρχείο αυτό είναι ένα απλό properties file, με όνομα *services.txt*, που για τις υπηρεσίες που παραθέσαμε στο κεφάλαιο 3 έχει την ακόλουθη μορφή:

```
#services.txt
#This is the file that contains the supported services
1=Voice Call
2=SMS
3=MMS
```

```
4=e-mail
5=Video Call
6=Video Stream
7=Web Browsing
```

Οι κωδικοί που αναφέρονται σε αυτό το αρχείο επιλέχθηκαν τυχαία, καθώς η τιμή του κωδικού δεν παίζει κανένα ιδιαίτερο ρόλο, αρκεί να είναι μοναδική για κάθε υπηρεσία.

▪ Κατάλογος προτεινόμενων παρόχων και κατάλογος υποστηριζόμενων τεχνολογιών:

Ο πρώτος από τους δύο αυτούς καταλόγους αποτελεί μια λίστα με προτεινόμενους παρόχους δικτύου. Η αναγνώριση των παρόχων αυτών ενσωματώθηκε ως δυνατότητα από τον κατασκευαστή της συσκευής. Για κάθε πάροχο που περιλαμβάνεται στη λίστα, υπάρχει η αντιστοίχιση του ονόματός του με το μοναδικό κωδικό αριθμό που τον χαρακτηρίζει. Σε περίπτωση αναβάθμισης της συσκευής, με στόχο να διευρυνθεί ή τροποποιηθεί η λίστα των προτεινόμενων παρόχων, αρκεί η ενημέρωση του συγκεκριμένου καταλόγου, χωρίς να απαιτείται κάποια αλλαγή στον κώδικα του λογισμικού του τερματικού.

Το αρχείο που ενσωματώνει αυτόν τον κατάλογο είναι ένα properties file (με όνομα *providers.txt*), η μορφή του οποίου έχει ως εξής:

```
#providers.txt
#This file contains some predefined providers
1=Cosmote
2=Vodafone
3=TIM
4=QTelecom
5=Forthnet
6=Hellas On Line
```

Οι κωδικοί που αναφέρονται σε αυτό το αρχείο επιλέχθηκαν τυχαία, καθώς η τιμή του κωδικού δεν παίζει κανένα ιδιαίτερο ρόλο, αρκεί να είναι μοναδική για κάθε πάροχο.

Ο δεύτερος κατάλογος, τώρα, αφορά τις υποστηριζόμενες από το τερματικό τεχνολογίες πρόσβασης. Περιλαμβάνει, δηλαδή, όλες τις τεχνολογίες που υποστηρίζει η συσκευή, καθώς επίσης και την αντιστοιχία κάθε μιας τεχνολογίας με το μοναδικό κωδικό της αριθμό. Σε περίπτωση που η συσκευή αναβαθμιστεί, με σκοπό να υποστηρίζει νέες τεχνολογίες πρόσβασης, αρκεί η ενημέρωση του συγκεκριμένου καταλόγου, χωρίς να χρειάζεται να τροποποιηθεί το λογισμικό του τερματικού.

Το αρχείο που περιέχει τον εν λόγω κατάλογο είναι και αυτό ένα απλό properties file (με όνομα *technologies.txt*), το οποίο, για την περίπτωση μιας συσκευής που υποστηρίζει και τις 4 ασύρματες τεχνολογίες για τις οποίες έχει γίνει λόγος (GSM, UMTS, WLAN και DVB), έχει την ακόλουθη μορφή:

```
#technologies.txt
#This file contains the supported access technologies
```

1=GSM 2=UMTS 3=WLAN 4=DVB
------------------------------------

Φυσικά, και πάλι οι κωδικοί που αναφέρονται σε αυτό το αρχείο επιλέχθηκαν τυχαία, καθώς η τιμή του κωδικού δεν παίζει κανένα ιδιαίτερο ρόλο, αρκεί να είναι μοναδική για κάθε τεχνολογία πρόσβασης.

- Κατάλογος των διαθέσιμων επιπέδων ποιότητας κάθε υπηρεσίας:

Ο κατάλογος αυτός περιέχει, για κάθε μια υποστηριζόμενη υπηρεσία ξεχωριστά, τον αριθμό των επιπέδων ποιότητας στα οποία μπορεί αυτή να διατεθεί, όπως επίσης και το ακριβές bitrate (σε Kbps) που αντιστοιχεί σε κάθε επίπεδο. Στον κατάλογο αυτό πρέπει να ανατρέξει το IIS module, προκειμένου να προσδιορίσει τα επίπεδα ποιότητας που μπορεί να προσφέρει στην αιτούμενη υπηρεσία ένα access point, δεδομένου του διαθέσιμου εύρους ζώνης του.

Ο κατάλογος αυτός βρίσκεται αποθηκευμένος στην κεντρική τράπεζα πληροφοριών με τη μορφή ενός XML αρχείου (*services.xml*) που έχει την ακόλουθη δομή:

```
<?xml version="1.0"?>
<!-- services.xml -->
<services>
  <service code="1" name="Voice Call" availQoSLevels="5">
    <QoS level="1">4</QoS>
    <QoS level="2">8</QoS>
    <QoS level="3">16</QoS>
    <QoS level="4">20</QoS>
    <QoS level="5">25</QoS>
  </service>
  <service code="2" name="SMS" availQoSLevels="1">
    <QoS level="1">1</QoS>
  </service>
  <service code="3" name="MMS" availQoSLevels="2">
    <QoS level="1">8</QoS>
    <QoS level="2">32</QoS>
  </service>
  <service code="4" name="e-mail" availQoSLevels="1">
    <QoS level="1">32</QoS>
  </service>
  <service code="5" name="Video Call" availQoSLevels="5">
    <QoS level="1">32</QoS>
    <QoS level="2">64</QoS>
    <QoS level="3">128</QoS>
    <QoS level="4">256</QoS>
    <QoS level="5">384</QoS>
  </service>
  <service code="6" name="Video Stream" availQoSLevels="5">
    <QoS level="1">64</QoS>
    <QoS level="2">128</QoS>
    <QoS level="3">512</QoS>
    <QoS level="4">1024</QoS>
  </service>
</services>
```

```

        <QoS level="5">2000</QoS>
    </service>
    <service code="7" name="Web Browsing" availQoSLevels="5">
        <QoS level="1">32</QoS>
        <QoS level="2">64</QoS>
        <QoS level="3">128</QoS>
        <QoS level="4">384</QoS>
        <QoS level="5">1024</QoS>
    </service>
</services>

```

Το παραπάνω αρχείο αποτελεί, ουσιαστικά, την αναπαράσταση σε XML μορφή των πινάκων αντιστοίχισης επιπέδων ποιότητας με bitrates, που παρατέθηκαν στο κεφάλαιο 3.

▪ Κατάλογος κόστους υπηρεσιών:

Στον κατάλογο αυτό περιλαμβάνονται οι πληροφορίες που αφορούν το κόστος παροχής κάθε υπηρεσίας υπό συγκεκριμένη τεχνολογία πρόσβασης, σε συγκεκριμένο επίπεδο ποιότητας και από συγκεκριμένο πάροχο. Έχουν, δηλαδή, την παρακάτω δομή, η οποία παρουσιάστηκε αναλυτικά στην παράγραφο 1.4.5.

		Cosmote				Vodafone			TIM	
		GSM	UMTS	WLAN	DVB	GSM	UMTS	WLAN	...	...
Video	QoS-1									
	QoS-2									
	QoS-3									
E-mail	...									
	...									
...	...									

Η πληροφορία αυτή, για λόγους εύκολης επεξεργασίας, αλλά και προκειμένου να έχει πρότυπη δομή, αναπαρίσταται σε XML μορφή. Μπορούμε να θεωρήσουμε ότι, κάθε φορά που το κινητό τερματικό τίθεται σε λειτουργία ή ανά τακτά αραιά χρονικά διαστήματα, γίνεται ενημέρωση της πληροφορίας αυτής, ώστε να είναι η πλέον πρόσφατη. Στο αρχείο αυτό ανατρέχει το IIS module με στόχο τον προσδιορισμό του κόστους καθεμιάς από τις επιλογές σύνδεσης.

Ένα τμήμα του αρχείου XML (*costs.xml*) που περιέχει τις πληροφορίες κόστους που χρησιμοποιήθηκαν στην εφαρμογή μας είναι το ακόλουθο:

```

<?xml version="1.0"?>
<!-- costs.xml -->
<costs>
    <service code = "1" maxCost="0.032">
        <QoS level = "1">
            <provider code = "1">
                <technology code = "1">0.003</technology>
                <technology code = "2">0.005</technology>
                <technology code = "3">0.002</technology>
            </provider>
        </QoS>
    </service>
</costs>

```

```

        <technology code = "4">0.006</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.0032</technology>
        <technology code = "2">0.0055</technology>
        <technology code = "3">0.0016</technology>
        <technology code = "4">0.0057</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.0028</technology>
        <technology code = "2">0.0047</technology>
        <technology code = "3">0.0019</technology>
        <technology code = "4">0.0055</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.0032</technology>
        <technology code = "2">0.006</technology>
        <technology code = "3">0.0018</technology>
        <technology code = "4">0.0058</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.0031</technology>
        <technology code = "2">0.0055</technology>
        <technology code = "3">0.0016</technology>
        <technology code = "4">0.005</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.0029</technology>
        <technology code = "2">0.0054</technology>
        <technology code = "3">0.0017</technology>
        <technology code = "4">0.0054</technology>
    </provider>
</QoS>
<QoS level = "2">
    <provider code = "1">
        <technology code = "1">0.0055</technology>
        <technology code = "2">0.009</technology>
        <technology code = "3">0.004</technology>
        <technology code = "4">0.01</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.006</technology>
        <technology code = "2">0.0098</technology>
        <technology code = "3">0.003</technology>
        <technology code = "4">0.01</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.0053</technology>
        <technology code = "2">0.0087</technology>
        <technology code = "3">0.0038</technology>
        <technology code = "4">0.009</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.0056</technology>
        <technology code = "2">0.0091</technology>
        <technology code = "3">0.0039</technology>
        <technology code = "4">0.011</technology>

```

```

</provider>
<provider code = "5">
  <technology code = "1">0.0054</technology>
  <technology code = "2">0.009</technology>
  <technology code = "3">0.0032</technology>
  <technology code = "4">0.0089</technology>
</provider>
<provider code = "6">
  <technology code = "1">0.0053</technology>
  <technology code = "2">0.0086</technology>
  <technology code = "3">0.031</technology>
  <technology code = "4">0.0091</technology>
</provider>
</QoS>
<QoS level = "3">
  <provider code = "1">
    <technology code = "1">0.01</technology>
    <technology code = "2">0.015</technology>
    <technology code = "3">0.0055</technology>
    <technology code = "4">0.017</technology>
  </provider>
  <provider code = "2">
    <technology code = "1">0.012</technology>
    <technology code = "2">0.017</technology>
    <technology code = "3">0.0043</technology>
    <technology code = "4">0.018</technology>
  </provider>
  <provider code = "3">
    <technology code = "1">0.01</technology>
    <technology code = "2">0.011</technology>
    <technology code = "3">0.0055</technology>
    <technology code = "4">0.018</technology>
  </provider>
  <provider code = "4">
    <technology code = "1">0.011</technology>
    <technology code = "2">0.015</technology>
    <technology code = "3">0.0056</technology>
    <technology code = "4">0.017</technology>
  </provider>
  <provider code = "5">
    <technology code = "1">0.012</technology>
    <technology code = "2">0.014</technology>
    <technology code = "3">0.0044</technology>
    <technology code = "4">0.017</technology>
  </provider>
  <provider code = "6">
    <technology code = "1">0.011</technology>
    <technology code = "2">0.013</technology>
    <technology code = "3">0.005</technology>
    <technology code = "4">0.016</technology>
  </provider>
</QoS>
<QoS level = "4">
  <provider code = "1">
    <technology code = "1">0.012</technology>
    <technology code = "2">0.02</technology>
    <technology code = "3">0.0075</technology>

```



```

        <technology code = "4">0.024</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.014</technology>
        <technology code = "2">0.02</technology>
        <technology code = "3">0.006</technology>
        <technology code = "4">0.024</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.026</technology>
        <technology code = "3">0.008</technology>
        <technology code = "4">0.027</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.014</technology>
        <technology code = "2">0.019</technology>
        <technology code = "3">0.008</technology>
        <technology code = "4">0.023</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.016</technology>
        <technology code = "2">0.017</technology>
        <technology code = "3">0.007</technology>
        <technology code = "4">0.023</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.014</technology>
        <technology code = "2">0.02</technology>
        <technology code = "3">0.0065</technology>
        <technology code = "4">0.025</technology>
    </provider>
</QoS>
<QoS level = "5">
    <provider code = "1">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.025</technology>
        <technology code = "3">0.01</technology>
        <technology code = "4">0.03</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.024</technology>
        <technology code = "3">0.009</technology>
        <technology code = "4">0.032</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.019</technology>
        <technology code = "2">0.03</technology>
        <technology code = "3">0.013</technology>
        <technology code = "4">0.032</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.017</technology>
        <technology code = "2">0.023</technology>
        <technology code = "3">0.013</technology>
        <technology code = "4">0.028</technology>

```

```

        </provider>
        <provider code = "5">
            <technology code = "1">0.02</technology>
            <technology code = "2">0.022</technology>
            <technology code = "3">0.011</technology>
            <technology code = "4">0.031</technology>
        </provider>
        <provider code = "6">
            <technology code = "1">0.018</technology>
            <technology code = "2">0.021</technology>
            <technology code = "3">0.01</technology>
            <technology code = "4">0.029</technology>
        </provider>
    </QoS>
</service>
<service code = "2" maxCost="0.13">
    <QoS level = "1">
        <provider code = "1">
            <technology code = "1">0.09</technology>
            <technology code = "2">0.09</technology>
            <technology code = "3">0.013</technology>
            <technology code = "4">0.1</technology>
        </provider>
        <provider code = "2">
            <technology code = "1">0.085</technology>
            <technology code = "2">0.085</technology>
            <technology code = "3">0.015</technology>
            <technology code = "4">0.11</technology>
        </provider>
        <provider code = "3">
            <technology code = "1">0.07</technology>
            <technology code = "2">0.07</technology>
            <technology code = "3">0.012</technology>
            <technology code = "4">0.12</technology>
        </provider>
        <provider code = "4">
            <technology code = "1">0.075</technology>
            <technology code = "2">0.075</technology>
            <technology code = "3">0.013</technology>
            <technology code = "4">0.1</technology>
        </provider>
        <provider code = "5">
            <technology code = "1">0.08</technology>
            <technology code = "2">0.08</technology>
            <technology code = "3">0.017</technology>
            <technology code = "4">0.11</technology>
        </provider>
        <provider code = "6">
            <technology code = "1">0.085</technology>
            <technology code = "2">0.085</technology>
            <technology code = "3">0.014</technology>
            <technology code = "4">0.13</technology>
        </provider>
    </QoS>
</service>
<!-- ... -->
</costs>

```

Το πλήρες αρχείο με τα κόστη παρατίθεται στο τέλος της εργασίας, μαζί με τον πηγαίο κώδικα.

Στο σημείο αυτό, πρέπει να σταθούμε στην παρουσίαση της γενικής φιλοσοφίας, σύμφωνα με την οποία συμπληρώθηκε το παραπάνω αρχείο. Αρχικά, πρέπει να διευκρινίσουμε ότι οι τιμές κόστους που περιλαμβάνει το αρχείο αντιπροσωπεύουν κόστος ανά μονάδα όγκου ή ανά μονάδα χρόνου, ανάλογα με τον τύπο της υπηρεσίας. Σε καμιά περίπτωση δεν πρέπει να γίνει σύγκριση κόστους μεταξύ δυο διαφορετικών υπηρεσιών, καθώς αυτό έχει προκύψει με βάση τα ιδιαίτερα χαρακτηριστικά κάθε υπηρεσίας. Σύγκριση υφίσταται, όπως εξάλλου κάνει και ο αλγόριθμος IIS, μόνο μεταξύ επιπέδου ποιότητας, τεχνολογίας και παρόχου της ίδιας υπηρεσίας.

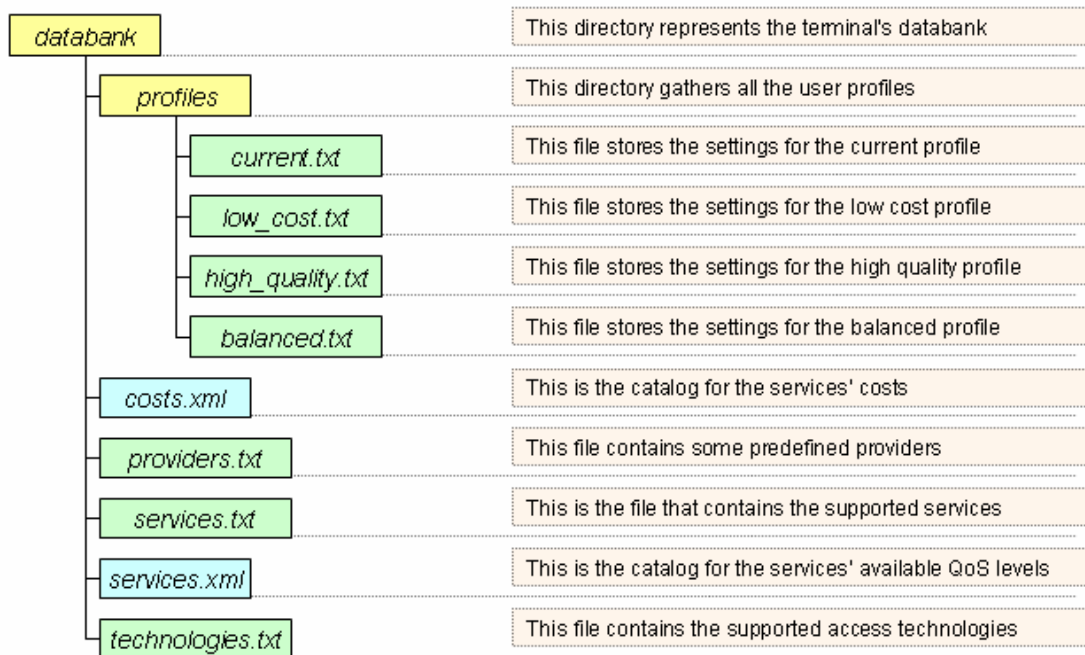
Οι βασικές αρχές σύμφωνα με τις οποίες συμπληρώθηκαν οι τιμές στο αρχείο είναι οι εξής:

- Όσο υψηλότερο είναι το επίπεδο ποιότητας, τόσο υψηλότερο είναι και το σχετικό κόστος.
- Γενικά, η τεχνολογία WLAN προσφέρει, συνήθως, πιο φθηνά τις διάφορες υπηρεσίες.
- Η τεχνολογία UMTS προσφέρει, σε γενικές γραμμές, τις υπηρεσίες με αυξημένο κόστος σε σύγκριση με την τεχνολογία GSM.
- Η τεχνολογία DVB είναι ακριβότερη σε σύγκριση με τις υπόλοιπες για τις περισσότερες υπηρεσίες, με εξαίρεση εκείνες που σχετίζονται με την αποστολή βίντεο, όπως είναι οι υπηρεσίες Video Call και Video Stream.

▪ Προφίλ χρήστη:

Όπως είδαμε στην παράγραφο 3.2, το προφίλ του χρήστη του τερματικού αποθηκεύεται και αυτό σε ένα αρχείο. Το αρχείο αυτό είναι ένα properties file, του οποίου τη μορφή έχουμε ήδη παραθέσει στην υπο-ενότητα «Αποθήκευση ή/και Ενεργοποίηση (προφίλ)» της συγκεκριμένης παραγράφου. Επειδή, όπως είδαμε, υπάρχουν 4 είδη προφίλ («*Current*», «*Low Cost*», «*High Quality*», «*Balanced*») διαθέσιμα στο τερματικό, η τράπεζα πληροφοριών περιλαμβάνει και τα 4 αντίστοιχα properties files: *current.txt*, *low\_cost.txt*, *high\_quality.txt* και *balanced.txt*.

Κατά συνέπεια, σύμφωνα με τα όσα έχουμε παραθέσει στην παράγραφο αυτή, η κεντρική τράπεζα πληροφοριών έχει τη μορφή που απεικονίζεται στο επόμενο σχήμα:



Σημειωτέον ότι από το παραπάνω σχήμα λείπουν ορισμένα μικρά αρχεία παραμετροποίησης, για τα οποία θα γίνει λόγος στην παράγραφο 4.2.4, στην οποία και θα παρατεθεί το πλήρες σχήμα της τράπεζας πληροφοριών.

#### 4.1.7 Στοιχεία Εξόδου

Στη συνέχεια, θα επικεντρωθούμε στο ποια θα πρέπει να είναι τα στοιχεία εξόδου του συστήματος βελτιστοποίησης. Ο αλγόριθμος, λοιπόν, του IIS module, λαμβάνοντας υπόψη του τα στοιχεία εισόδου, όπως αυτά παρουσιάστηκαν στην παράγραφο 4.1.5, θα πρέπει να αποφανθεί, χρησιμοποιώντας μια αντικειμενική συνάρτηση, για το επίπεδο ποιότητας στο οποίο θα ανατεθεί η αιτούμενη υπηρεσία και το σημείο πρόσβασης από το οποίο θα εξυπηρετηθεί.

Προκειμένου να έχουμε βέλτιστη κατανομή πόρων, αντί μιας τελικής απόφασης, ο αλγόριθμος δημιουργεί μια ιεραρχημένη λίστα υποψήφιων επιλογών, όπου κάθε τέτοια επιλογή είναι ένα ζεύγος επιπέδου ποιότητας και σημείου πρόσβασης. Το πρώτο στοιχείο αυτής της λίστας αποστέλλεται προς έγκριση στην οντότητα διαχείρισης του δικτύου. Αν η απάντηση είναι θετική, τότε πραγματοποιείται ανάθεση της αιτούμενης υπηρεσίας στο συγκεκριμένο επίπεδο ποιότητας και το συγκεκριμένο σημείο πρόσβασης. Σε αντίθεση περίπτωση, αποστέλλεται προς έγκριση η δεύτερη στη σειρά επιλογή της ιεραρχημένης λίστας, κοκ. Με αυτόν τον τρόπο, ικανοποιούνται και οι δύο βασικές αρχές που διέπουν την ανάθεση υπηρεσιών, δηλαδή η αυτονομία του τερματικού, αλλά και ο έλεγχος από την πλευρά του δικτύου.

#### 4.1.8 Περιορισμοί

Από την περιγραφή των στοιχείων εξόδου προκύπτει, όσον αφορά την έξοδο του συστήματος βελτιστοποίησης, ότι υπάρχουν κάποιοι φυσικοί περιορισμοί, που πρέπει να ικανοποιούνται, ώστε η λύση να θεωρείται αποδεκτή. Οι περιορισμοί αυτοί είναι οι ακόλουθοι:

Πρώτον, κάθε αιτούμενη υπηρεσία πρέπει να ανατίθεται σε αποδεκτό επίπεδο ποιότητας. Αυτό πρακτικά σημαίνει ότι σημεία πρόσβασης που δεν μπορούν να ικανοποιήσουν το χαμηλότερο δυνατό επίπεδο ποιότητας μιας υπηρεσίας (επίπεδο 1), απορρίπτονται από τη λίστα των υποψήφιων σημείων πρόσβασης.

Δεύτερον, πρέπει να ικανοποιούνται με τη δηλωμένη σειρά προτεραιότητας οι προτιμήσεις του χρήστη. Αυτό σημαίνει ότι το σύστημα πρέπει να λαμβάνει απόφαση βάσει των προτιμήσεων του χρήστη. Επίσης, η σειρά προτεραιότητας στην τελική λίστα επιλογών που διαμορφώνεται πρέπει να τηρείται απαρέγκλιτα.

Τρίτον, το επίπεδο ποιότητας στο οποίο ανατίθεται μια υπηρεσία δεν πρέπει, σε καμία περίπτωση, να υπερβαίνει το μέγιστο επιθυμητό επίπεδο ποιότητας που έχει καθορίσει στο προφίλ του ο χρήστης για τη συγκεκριμένη υπηρεσία. Σε αντίθετη περίπτωση, θα καταστρατηγείτο η επιθυμία του χρήστη να θέσει ένα πλαφόν στο κόστος που είναι διατεθειμένος να πληρώσει.

#### 4.1.9 Περιπτώσεις Εκκίνησης της Διαδικασίας Βελτιστοποίησης

Στην παράγραφο αυτή, θα εξετάσουμε σε ποιες περιπτώσεις πρέπει να γίνεται εκκίνηση της διαδικασίας βελτιστοποίησης, με άλλα λόγια πότε πρέπει να ενεργοποιείται (να γίνεται «trigger») ο αλγόριθμος του IIS. Διακρίνουμε τις εξής περιπτώσεις, κατά τη λειτουργία του τερματικού:

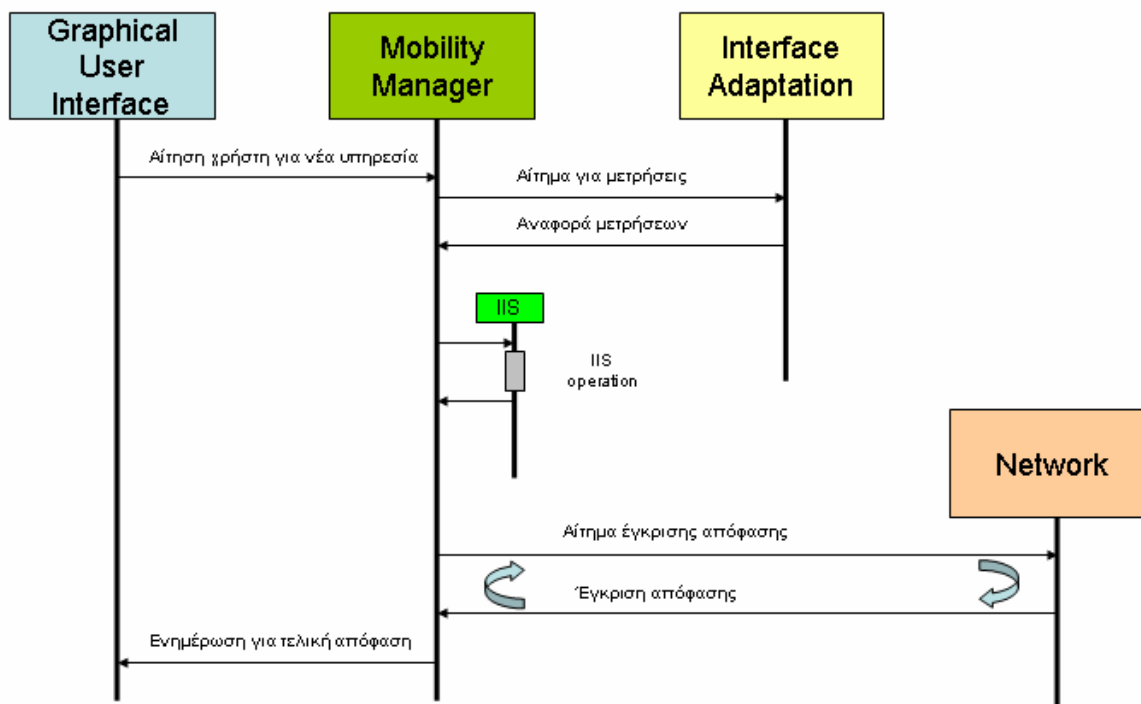
- Αίτηση για παροχή νέας υπηρεσίας:

Πρόκειται για την περίπτωση όπου ο χρήστης ζητά την εκκίνηση μιας νέας υπηρεσίας, μέσω της γραφικής διεπαφής. Προφανώς, θα πρέπει να τρέξει ο αλγόριθμος του IIS, προκειμένου να διαπιστωθεί το κατάλληλο επίπεδο ποιότητας και το καλύτερο access point στο οποίο μπορεί να ανατεθεί η υπηρεσία.

Όπως είδαμε και στο κεφάλαιο 3, ο χρήστης είναι σε θέση, μέσα από τη γραφική διεπαφή, να επιλέξει την υπηρεσία που τον ενδιαφέρει και, κατόπιν, να πατήσει το κουμπί που αντιστοιχεί στην εκκίνησή της, όπως φαίνεται και από τα screenshots που ακολουθούν.



Με το πάτημα αυτού του κουμπιού, ο αλγόριθμος του IIS θα πρέπει να τίθεται αμέσως σε εφαρμογή. Το παρακάτω ακολουθιακό διάγραμμα αναπαριστά με λεπτομέρεια τη ροή της πληροφορίας και την ανταλλαγή μηνυμάτων μεταξύ των διαφόρων modules του τερματικού, για την περίπτωση της αίτησης νέας υπηρεσίας από τη μεριά του χρήστη.

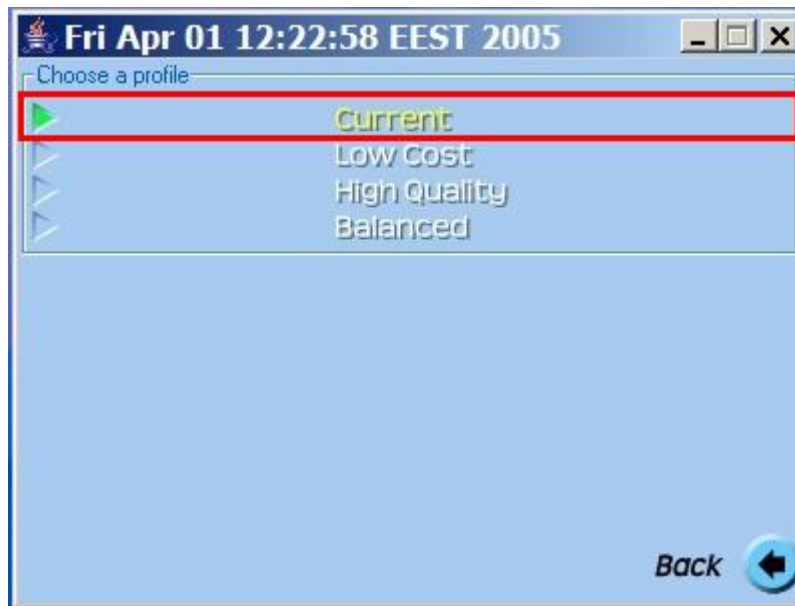


Όπως φαίνεται και στο παραπάνω σχήμα, ο χρήστης, μέσω της γραφικής διεπαφής, κάνει αίτηση για μια νέα υπηρεσία. Ο Mobility Manager, τότε, αναλαμβάνει να αποκτήσει, μέσω του Interface Adaptation module, τις τρέχουσες μετρήσεις για την κατάσταση του δικτύου και, αμέσως μετά, να εκκινήσει τον αλγόριθμο του IIS. Μόλις βρεθεί η βέλτιστη λύση, ο Mobility Manager την προτείνει στην οντότητα διαχείρισης του δικτύου και ζητά έγκριση για αυτήν. Σε περίπτωση απόρριψης, ο Mobility Manager προτείνει την αμέσως καλύτερη λύση, κοκ. Μόλις ληφθεί μια έγκριση, η γραφική διεπαφή ενημερώνεται από το Mobility Manager module για την τελική απόφαση, ώστε να απεικονίσει πληροφορίες σχετικές με τη σύνδεση που εγκαταστάθηκε.

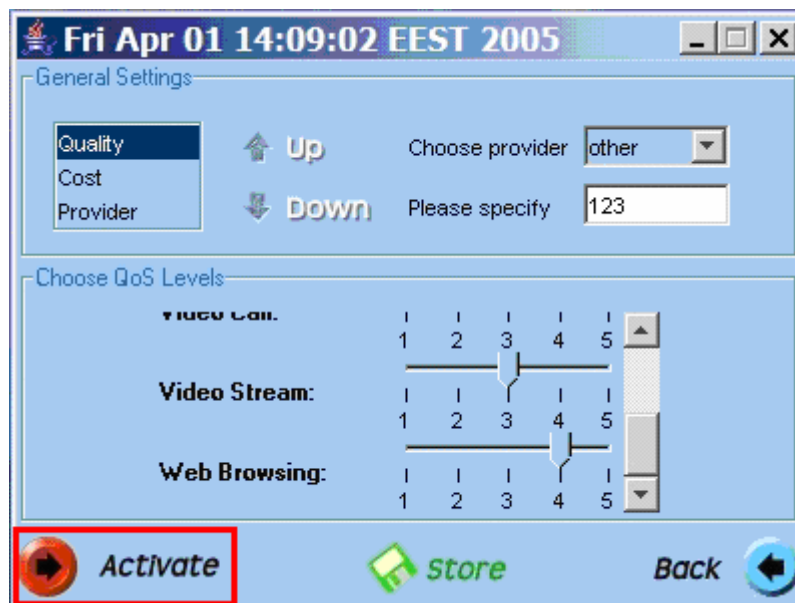
- Αλλαγή του προφίλ χρήστη:

Πρόκειται για την περίπτωση όπου ο χρήστης κάνει κάποια αλλαγή στις παραμέτρους του τρέχοντος προφίλ, μεταβάλλοντας κάποια από τις τρέχουσες ρυθμίσεις που αντιπροσωπεύουν τις προτιμήσεις του. Ο αλγόριθμος του IIS θα πρέπει και πάλι να τρέξει, και μάλιστα για όλες τις υπηρεσίες που βρίσκονται σε εξέλιξη, έτσι ώστε να ικανοποιηθούν οι «καινούριες» προτιμήσεις του χρήστη.

Όπως είδαμε και στο κεφάλαιο 3, ο χρήστης μπορεί, μέσω της γραφικής διεπαφής, να τροποποιήσει το προφίλ του και, στη συνέχεια, να το αποθηκεύσει. Αν το προφίλ που έχει επιλέξει να επεξεργασθεί είναι το τρέχον, όπως απεικονίζει το παρακάτω screenshot, τότε θα εκκινηθεί και ο αλγόριθμος του IIS, σε περίπτωση που το τροποποιήσει και αποθηκεύσει τις αλλαγές.

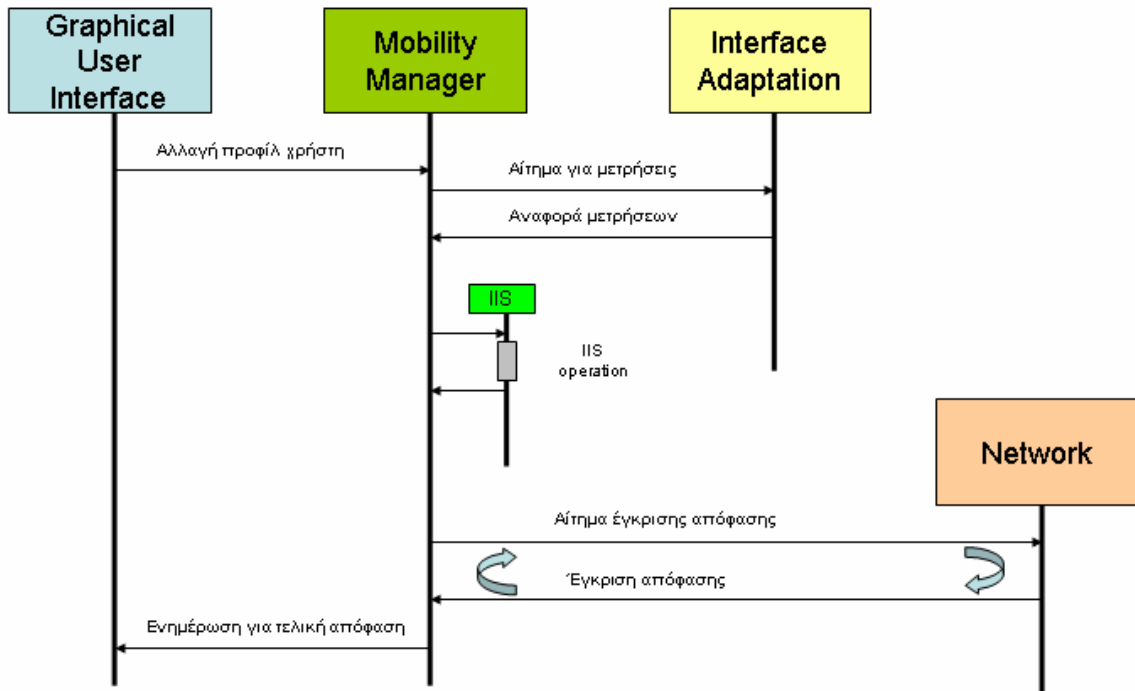


Το ίδιο θα συμβεί και στην περίπτωση που ο χρήστης επιλέξει οποιοδήποτε προφίλ (όχι απαραίτητα το τρέχον) και, έπειτα, πατήσει το κουμπί «Ενεργοποίηση» («Activate»), αφού κάνει τις αλλαγές που επιθυμεί.



Το ακολουθιακό διάγραμμα για την περίπτωση που μόλις εξετάσαμε είναι παρόμοιο με αυτό της περίπτωσης όπου έχουμε αίτηση νέας υπηρεσίας και δίνεται αμέσως πιο κάτω.



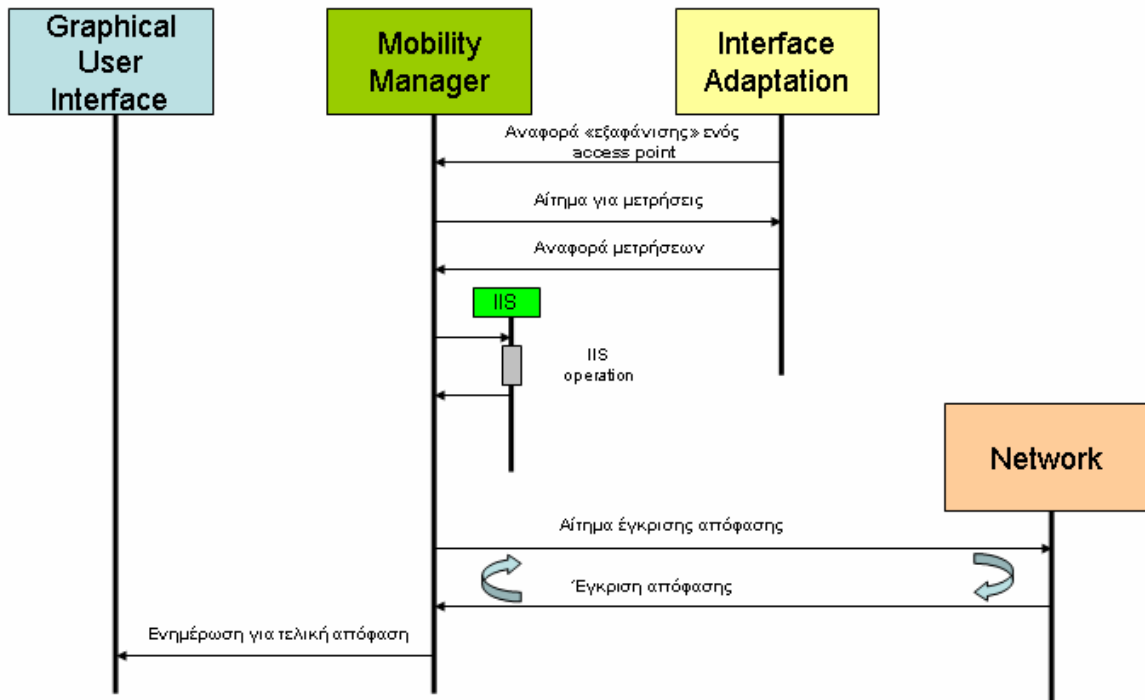


Όπως φαίνεται και στο παραπάνω σχήμα, η ακολουθία ενεργειών που λαμβάνει χώρα είναι ίδια με εκείνη της προηγούμενης περίπτωσης, με τη διαφορά ότι η εκκίνηση της όλης διαδικασίας οφείλεται σε αλλαγή στο τρέχον προφίλ του χρήστη, μέσα από το αντίστοιχο πάνελ της γραφικής διεπαφής, και η εκτέλεση του αλγορίθμου IIS γίνεται για όλες τις τρέχουσες υπηρεσίες.

- Υποβάθμιση λαμβανόμενου σήματος από τρέχον access point:

Ας θεωρήσουμε ότι μία ή περισσότερες υπηρεσίες λαμβάνονται από ένα συγκεκριμένο σημείο πρόσβασης και ότι κάποια στιγμή το σήμα που λαμβάνεται από αυτό εξασθενεί κάτω από ένα όριο, ώστε πρακτικά να είναι σαν το τερματικό μας να μη «βλέπει» πλέον το εν λόγω access point. Στην περίπτωση αυτή, θα πρέπει να εκκινηθεί και πάλι η διαδικασία βελτιστοποίησης για όλες τις υπηρεσίες που λαμβάνονταν από αυτό το σημείο πρόσβασης.

Οι ενέργειες που εκτελούνται από τα διάφορα modules του τερματικού, στην περίπτωση κατά την οποία διαπιστωθεί μια τέτοια υποβάθμιση του λαμβανόμενου σήματος, παρουσιάζονται στο διάγραμμα που ακολουθεί.

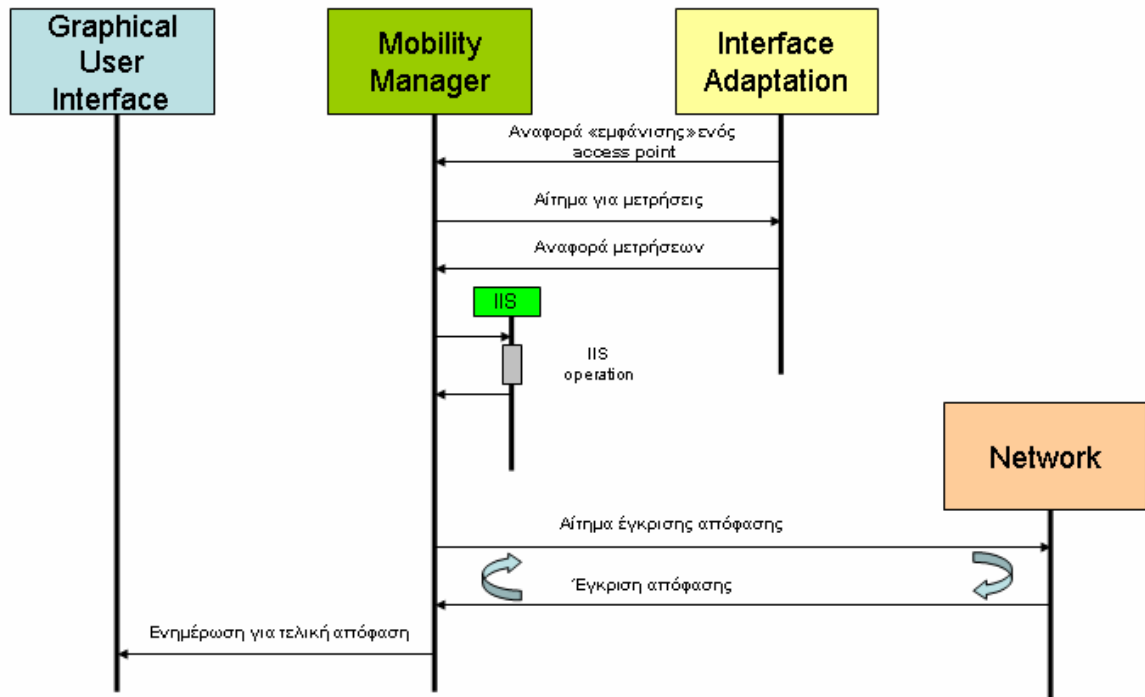


Όπως φαίνεται και στο παραπάνω σχήμα, μόλις το Interface Adaptation module διαπιστώσει την εξαφάνιση ενός access point, το αναφέρει στο Mobility Manager. Εκείνος, τότε, ενημερώνεται για τις τρέχουσες μετρήσεις και εκκινεί τον αλγόριθμο του IIS για όσες υπηρεσίες λαμβάνονταν από το συγκεκριμένο access point. Ακολουθεί η γνωστή διαδικασία λήψης έγκρισης από το δίκτυο και ενημέρωσης της γραφικής διεπαφής.

- Εμφάνιση καινούριου σημείου πρόσβασης:

Καθώς το τερματικό βρίσκεται σε κίνηση, καινούρια σημεία πρόσβασης μπορούν να βρεθούν στην εμβέλειά του. Τα καινούρια αυτά σημεία πρόσβασης μπορεί να υπερτερούν εκείνων που «έβλεπε» μέχρι στιγμής το τερματικό και, ως εκ τούτου, είναι χρήσιμο ο αλγόριθμος του IIS να τρέξει και πάλι, και μάλιστα για όλες τις υπηρεσίες που βρίσκονται σε εξέλιξη.

Το παρακάτω ακολουθιακό διάγραμμα αναπαριστά με λεπτομέρεια τη ροή της πληροφορίας και την ανταλλαγή μηνυμάτων μεταξύ των διαφόρων modules του τερματικού, για την περίπτωση της εμφάνισης καινούριου σημείου πρόσβασης στην περιοχή του κινητού τερματικού.



Όπως φαίνεται και στο παραπάνω σχήμα, μόλις το Interface Adaptation module διαπιστώσει την εμφάνιση ενός καινούριου access point, το αναφέρει στο Mobility Manager. Εκείνος, τότε, ενημερώνεται για τις τρέχουσες μετρήσεις και εκκινεί τον αλγόριθμο του IIS για όλες τις τρέχουσες υπηρεσίες. Ακολουθεί η γνωστή διαδικασία λήψης έγκρισης από το δίκτυο και ενημέρωσης της γραφικής διεπαφής.

## 4.2 Λύση Προβλήματος

### 4.2.1 Εισαγωγή

Έχοντας προδιαγράψει τα στοιχεία εισόδου, τα στοιχεία εξόδου και τους περιορισμούς του συστήματος, προχωρούμε τώρα στην παρουσίαση του πυρήνα του αλγορίθμου IIS, στη διαδικασία, δηλαδή, βελτιστοποίησης μέσω της οποίας το κινητό τερματικό καταλήγει στην τελική επιλογή σημείου πρόσβασης και επιπέδου ποιότητας, για δεδομένη υπηρεσία.

Η κεντρική ιδέα είναι ότι η τελική επιλογή στην οποία θα καταλήξει ο αλγόριθμος του IIS θα πρέπει να μεγιστοποιεί μια αντικειμενική συνάρτηση. Μια τέτοια αντικειμενική συνάρτηση θα πρέπει να θεωρεί ως θετική ποσότητα την ποιότητα και ως αρνητική το κόστος. Σκοπός, λοιπόν, είναι η μεγιστοποίηση της διαφοράς της ποιότητας που προσφέρει μια επιλογή μείον το κόστος που της αντιστοιχεί. Η αντικειμενική συνάρτηση, δηλαδή, πρέπει να έχει μια μορφή του τύπου:

$$Objective\_Function(Quality, Cost) = a \times Quality - b \times Cost$$

όπου a και b κατάλληλοι συντελεστές.

Στην επόμενη παράγραφο, παρουσιάζεται αναλυτικότερα η μαθηματική μορφή της αντικειμενικής συνάρτησης, η οποία συμπεριλαμβάνει όλες τις παραμέτρους που διαδραματίζουν ρόλο στη λήψη της τελικής απόφασης.

#### 4.2.2 Μαθηματική Διατύπωση

Προηγουμένως, είδαμε ποια είναι, γενικά, η κατάλληλη μορφή της αντικειμενικής συνάρτησης που χρησιμοποιεί ο αλγόριθμος βελτιστοποίησης του IIS. Στη συνέχεια, θα εξετάσουμε ποια είναι η ακριβής μορφή της παραπάνω συνάρτησης, διατυπώνοντας ταυτόχρονα το πρόβλημα από μαθηματική σκοπιά.

Τα σημεία τα οποία πρέπει να ληφθούν για την κατάστρωση της τελικής συνάρτησης είναι τα εξής:

- Για την τελική επιλογή, οφείλουν να εξεταστούν όλα τα υποψήφια σημεία πρόσβασης.
- Κάθε διαφορετικό επίπεδο ποιότητας στο οποίο μπορεί να προσφέρει την αιτούμενη υπηρεσία ένα σημείο πρόσβασης οφείλει να εξεταστεί ξεχωριστά.
- Πέραν της ποιότητας και του κόστους, ένας τρίτος παράγοντας που πρέπει να ληφθεί υπόψη είναι ο πάροχος της συγκεκριμένης υπηρεσίας. Ο παράγοντας αυτός, όμως, πρέπει να συνεισφέρει σε μία επιλογή μόνο όταν συμπίπτει με την προτίμηση του χρήστη για τον πάροχο, όπως αυτή έχει δηλωθεί στο προφίλ του.
- Οι τιμές των συντελεστών με τους οποίους θα πολλαπλασιαστούν οι παραπάνω τρεις παράγοντες πρέπει να αντανakλούν την κατάταξη των παραγόντων αυτών στο προφίλ του χρήστη.

Με βάση, λοιπόν, τα παραπάνω, καταλήγουμε στην εξής μορφή της αντικειμενικής συνάρτησης:

- Έστω  $\mathbf{P}$  το σύνολο των υποψήφιων σημείων πρόσβασης (access points), που «βλέπει» το τερματικό,  $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n\}$ ,  $n \in \mathbb{N}$ .
- Έστω  $\mathbf{Q}(\mathbf{p})$  το σύνολο των επιπέδων ποιότητας στα οποία μπορεί να προσφέρει την υπό εξέταση υπηρεσία το σημείο πρόσβασης  $\mathbf{p}$ ,  $\mathbf{Q}(\mathbf{p}) = \{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m\}$ ,  $m \in \mathbb{N}$ .
- Έστω **qualityFactor**, **costFactor**, **providerFactor** οι συντελεστές των παραγόντων «ποιότητα», «κόστος» και «πάροχος», όπως έχουν διαμορφωθεί από την κατάταξή τους στο προφίλ του χρήστη.
- Η αντικειμενική συνάρτηση ορίζεται ως εξής:

$$\begin{aligned} \mathbf{Objective\_Function}(p, q) = & \mathbf{qualityFactor} \times \mathbf{RealQuality}(p, q) - \\ & \mathbf{costFactor} \times \mathbf{Cost}(p, q) + \\ & \mathbf{providerFactor} \times \mathbf{Provider}(p) \end{aligned}$$

όπου  $p \in P$  και  $q \in Q(p)$ .

- Οι τιμές των συντελεστών  $qualityFactor$ ,  $costFactor$  και  $providerFactor$  και οι τιμές των παραγόντων  $RealQuality(p,q)$ ,  $Cost(p,q)$  και  $Provider(p)$ , όπου  $p \in P$  και  $q \in Q(p)$ , είναι πραγματικοί αριθμοί. Ο παράγοντας **RealQuality(p,q)** αντιπροσωπεύει την πραγματική ποιότητα που είναι σε θέση να προσφέρει το σημείο πρόσβασης  $p$  για το επίπεδο ποιότητας  $q$ , όπως αυτή έχει διαμορφωθεί υπό την επίδραση της ποιότητας του λαμβανόμενου σήματος. Περισσότερες λεπτομέρειες θα δοθούν παρακάτω. Ο παράγοντας **Cost(p,q)** εκφράζει το (ανά μονάδα όγκου ή χρόνου) κόστος με το οποίο παρέχει την αιτούμενη υπηρεσία το σημείο πρόσβασης  $p$ , στο επίπεδο ποιότητας  $q$ . Ο παράγοντας **Provider(p)**, τέλος, μπορεί να πάρει τιμή μη μηδενική ή μηδενική, ανάλογα με το αν ο πάροχος του σημείου πρόσβασης  $p$  συμπίπτει ή όχι, αντίστοιχα, με τον προτιμώμενο από το χρήστη πάροχο.

Ο αλγόριθμος του IIS καταλήγει σε μια βέλτιστη επιλογή, η οποία πρέπει να ικανοποιεί την απαίτηση:

$$BestChoice = \max_{\forall p \in P} \left\{ \max_{\forall q \in Q(p)} \{ Objective\_Function(p,q) \} \right\}$$

#### 4.2.3 Βήματα Επίλυσης του Προβλήματος

Προχωρούμε τώρα στην αναλυτική παρουσίαση των βημάτων που ακολουθούνται κατά τη διαδικασία επίλυσης του προβλήματος. Πρόκειται για τα βήματα που ακολουθεί επακριβώς ο αλγόριθμος του IIS.

- Συλλογή πληροφοριών για τα διαθέσιμα access points:

Πρώτο βήμα της διαδικασίας αποτελεί η ανάκτηση των τρεχουσών μετρήσεων, η ενημέρωση, δηλαδή, για τα access points που «βλέπει» το κινητό τερματικό, τη συγκεκριμένη στιγμή. Για κάθε τέτοιο access point, υπάρχουν διαθέσιμες οι εξής πληροφορίες:

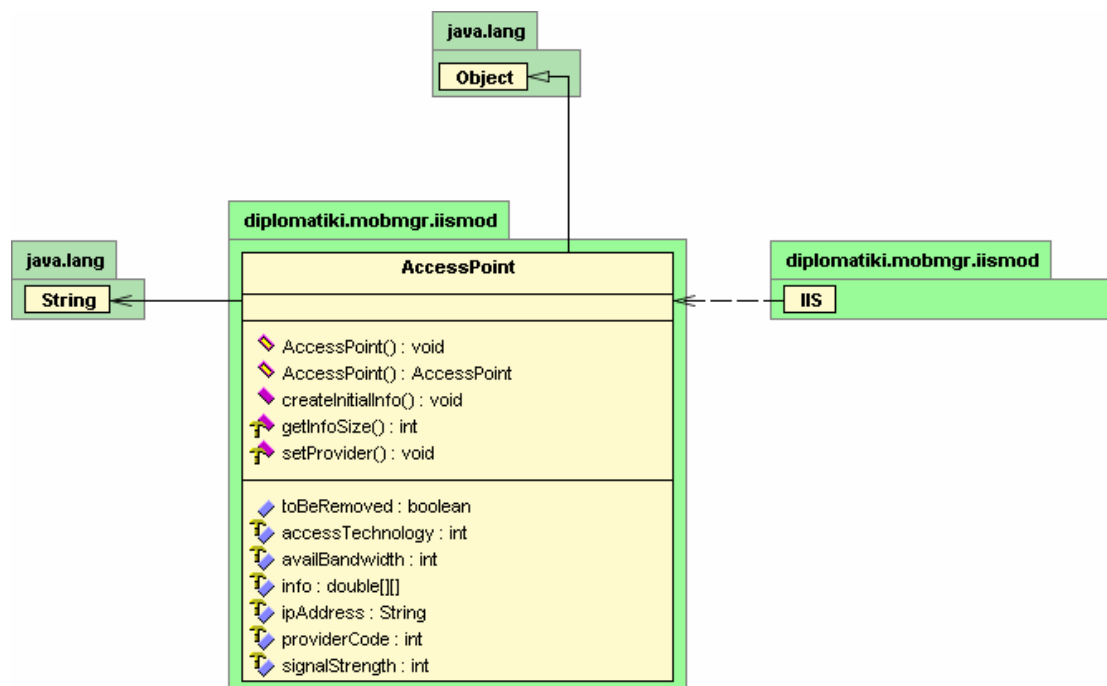
- ✓ Πάροχος δικτύου (Provider) στον οποίο ανήκει το access point.
- ✓ Τύπος τεχνολογίας (Access Technology) του access point.
- ✓ Ποιότητα του σήματος (Signal Strength) που λαμβάνεται από το access point (κλίμακα 1 έως 5).
- ✓ Διαθέσιμο εύρος ζώνης (Available Bandwidth) στο access point (σε Kbps).

Οι πληροφορίες αυτές συλλέγονται από κατάλληλους drivers του τερματικού και, μέσω του Interface Adaptation module, γίνονται διαθέσιμες στο Mobility Manager. Η δομή στην οποία διατίθενται είναι ένας πίνακας της ακόλουθης μορφής:

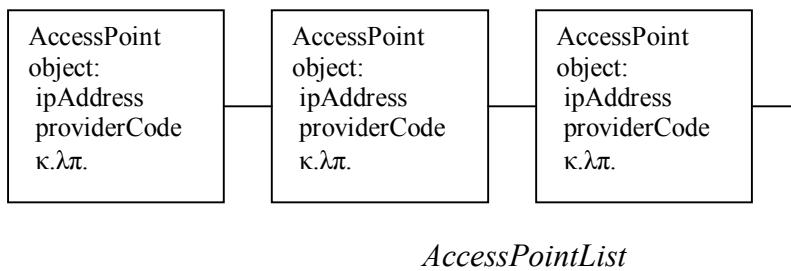
IP Address	Provider	Technology	Signal Strength	Available Bandwidth
147.102.10.2	1 (κωδικός του παρόχου «Cosmote»)	3 (κωδικός της τεχνολογίας WLAN)	5 (full strength)	1500 (σε Kbps)
...	...	...	...	...

▪ Δημιουργία λίστας διαθέσιμων access points:

Αρχικά, το IIS module, παίρνοντας αυτόν πίνακα, πρέπει να τον μετασχηματίσει σε μια δομή που να είναι περισσότερο κατάλληλη για αντικειμενοστρεφή επεξεργασία. Για κάθε ένα, λοιπόν, διαθέσιμο access point, δημιουργεί ένα αντικείμενο της κλάσης AccessPoint. Ένα αντικείμενο της κλάσης αυτής αντιπροσωπεύει πλήρως μια οντότητα access point, περιλαμβάνοντας όλες τις ιδιότητές του. Ουσιαστικά, αποτελεί την αναπαράσταση σε κώδικα του πραγματικού access point.



Κάθε τέτοιο αντικείμενο της κλάσης AccessPoint που κατασκευάζεται ενσωματώνεται σε μια λίστα (στο εξής, θα την καλούμε και AccessPointList) υποψήφιων access points. Στο τέλος της παραπάνω διαδικασίας, η AccessPointList που έχει δημιουργηθεί περιλαμβάνει όλες τις πληροφορίες που αφορούν κάθε access point, αποτελεί, δηλαδή, ουσιαστικά, ένα μετασχηματισμό του πίνακα με τις μετρήσεις, που παραλαμβάνεται μέσω του Interface Adaptation module, σε μια δομή εύκολη για επεξεργασία.



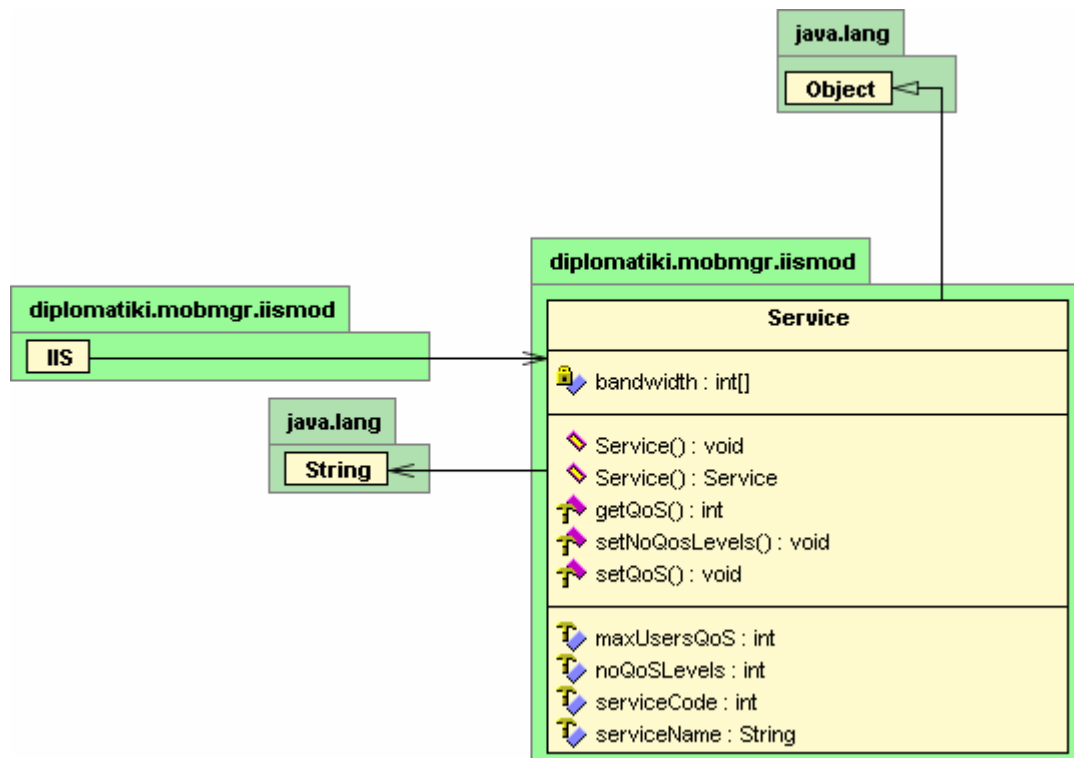
Σχηματικά, το στάδιο επεξεργασίας που μόλις περιγράψαμε έχει ως εξής:



- Δημιουργία αντικειμένου Service:

Το IIS module ενεργοποιείται να τρέξει με παράμετρο τον κωδικό της αιτούμενης υπηρεσίας. Επειδή, όμως, οι πληροφορίες που πρέπει να γνωρίζει για την υπηρεσία δεν περιορίζονται στον κωδικό της αριθμό, είναι απαραίτητη η δημιουργία ενός αντικειμένου της κλάσης Service, το οποίο θα ενθυλακώνει περισσότερες λεπτομέρειες σχετικά με την αιτούμενη υπηρεσία. Κυρίως θα περιέχει τους ρυθμούς μετάδοσης (bitrates) που αντιστοιχούν σε κάθε διαθέσιμο επίπεδο ποιότητας, αλλά και την προτίμηση του χρήστη αναφορικά με το μέγιστο επίπεδο ποιότητας που επιθυμεί. Τα στοιχεία αυτά είναι σε θέση να τα ανακτήσει εύκολα το IIS module από την τράπεζα πληροφοριών, συγκεκριμένα δε από τα αρχεία *services.xml* και *current.txt*.

Το αντικείμενο τύπου Service που δημιουργείται έχει την ακόλουθη μορφή:



- Στάδιο προσδιορισμού του παράγοντα «ποιότητα» (Find Quality):

Στο στάδιο αυτό, το IIS module μεταφράζει το διαθέσιμο bandwidth καθενός access point σε διαθέσιμα επίπεδα ποιότητας (QoS levels). Δηλαδή, προσδιορίζει τα QoS levels στα οποία ένα access point είναι σε θέση να προσφέρει τη συγκεκριμένη υπηρεσία.

Προκειμένου να ικανοποιηθεί η απαίτηση ανάθεσης της υπηρεσίας σε ένα ελάχιστο αποδεκτό επίπεδο ποιότητας, το IIS module εξετάζει το διαθέσιμο εύρος ζώνης κάθε σημείου πρόσβασης και αφαιρεί από την AccessPointList όσα από αυτά αδυνατούν (λόγω εξαιρετικά χαμηλού διαθέσιμου bandwidth) να παρέχουν την υπηρεσία ακόμα και στο ελάχιστο επίπεδο ποιότητας (επίπεδο 1).

Εκτός αυτού, πραγματοποιεί και έναν επιπλέον έλεγχο. Από τα διαθέσιμα επίπεδα ποιότητας αφαιρεί εκείνα τα οποία υπερβαίνουν το επιθυμητό από το χρήστη μέγιστο επίπεδο ποιότητας. Την πληροφορία αυτήν την ανακτά μέσω του αντικειμένου Service, το οποίο έχει ήδη κατασκευάσει. Αυτό γίνεται προκειμένου να μην υπάρχει κίνδυνος να καταστρατηγηθεί η βασική επιθυμία του χρήστη για περιορισμό του κόστους.

Ερχόμαστε, τώρα, στη σημαντικότερη λειτουργία αυτού του σταδίου, που δεν είναι άλλη από τον προσδιορισμό του παράγοντα «ποιότητα» που χρησιμοποιείται κατά τον υπολογισμό της αντικειμενικής συνάρτησης. Είναι εμφανές ότι στη διαμόρφωση αυτού του παράγοντα, πέρα από το ονομαστικό επίπεδο ποιότητας, συμμετέχει και η ένταση του λαμβανόμενου, από το access point, σήματος. Αυτό γιατί, ακόμα και αν έχουμε υψηλό επίπεδο ποιότητας, όσον αφορά το bandwidth, η παροχή της υπηρεσίας από ένα σημείο πρόσβασης δε γίνεται με τον καλύτερο τρόπο όταν η ένταση του λαμβανόμενου σήματος είναι πολύ χαμηλή. Επιβάλλεται, συνεπώς, η εισαγωγή μιας



νέας έννοιας, που εφ' εξής θα καλείται «πραγματική ποιότητα», η οποία θα αντικατοπτρίζει αυτόν ακριβώς το συνδυασμό ονομαστικού επιπέδου ποιότητας και ισχύος του λαμβανόμενου σήματος. Πρόκειται για τον παράγοντα  $RealQuality(p,q)$ , με  $p \in P$  και  $q \in Q(p)$ , που αναφέρθηκε στην παράγραφο της μαθηματικής διατύπωσης (4.2.2).

Οι τρόποι με τους οποίους μπορεί να εκφραστεί η επίδραση της έντασης του λαμβανόμενου σήματος στον καθορισμό της τελικής ποιότητας είναι πολλοί. Οι επικρατέστεροι από αυτούς είναι δύο: μέσω ενός «πολλαπλασιαστικού παράγοντα» ή μέσω ενός «προσθετικού παράγοντα». Στη συγκεκριμένη υλοποίηση, υιοθετήθηκε η ιδέα του «πολλαπλασιαστικού παράγοντα». Σύμφωνα με αυτήν, η ονομαστική τιμή κάθε επιπέδου ποιότητας (επί παραδείγματι 2, για το QoS level-2) πολλαπλασιάζεται με έναν πραγματικό αριθμό (μικρότερο ή ίσο της μονάδας), ο οποίος εξαρτάται από την ένταση του λαμβανόμενου σήματος, έτσι ώστε να διαμορφωθεί μια τιμή, χαμηλότερη ή ίση της ονομαστικής τιμής, που να αντικατοπτρίζει την πραγματική ποιότητα.

Η ποιότητα του λαμβανόμενου σήματος εκφράζεται σε μια κλίμακα από 1 έως 5. Για καθένα από αυτά τα επίπεδα σήματος, ορίστηκε και ο αντίστοιχος πολλαπλασιαστικός παράγοντας, σύμφωνα με τον πίνακα που ακολουθεί:

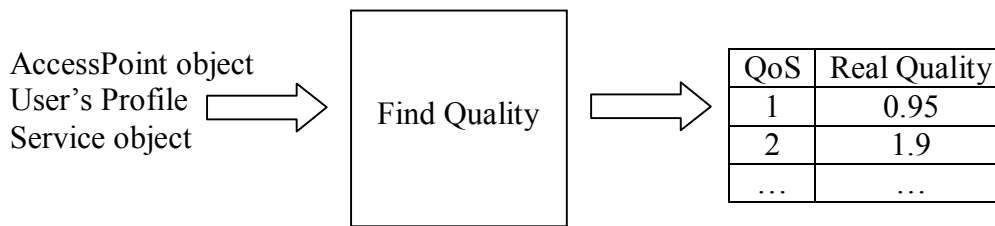
Signal Strength Level	Factor
1	0.4
2	0.6
3	0.8
4	0.95
5	1.0

Έτσι, για παράδειγμα, ένταση σήματος 5 δε θα υποβαθμίσει το επίπεδο ποιότητας, ενώ αντίθετα ένταση σήματος 1, θα προκαλέσει υποβάθμιση 60%.

Οι πολλαπλασιαστικοί αυτοί παράγοντες είναι αποθηκευμένοι σε ένα properties file, το οποίο συμπεριλαμβάνεται στην κεντρική τράπεζα πληροφοριών του τερματικού. Το αρχείο αυτό, με όνομα *signal\_factors.txt*, έχει ως εξής:

```
#signal_factors.txt
#This file contains the appropriate signal factor for each signal strength level
1=0.4
2=0.6
3=0.8
4=0.95
5=1.0
```

Σχηματικά, το στάδιο επεξεργασίας που μόλις περιγράψαμε έχει ως εξής:



Το στάδιο αυτό επεξεργασίας πρέπει να εφαρμοστεί σε καθένα από τα διαθέσιμα access points που περιλαμβάνονται στη λίστα AccessPointList. Η πληροφορία εξόδου (πίνακας του σχήματος), που αφορά τα υποστηριζόμενα επίπεδα ποιότητας και τα αντίστοιχα επίπεδα πραγματικής ποιότητας, εγγράφεται και αυτή μέσα στο ίδιο το αντικείμενο της κλάσης AccessPoint (AccessPoint object), το οποίο είναι διαμορφωμένο πλέον για τη συγκεκριμένη υπηρεσία (περιέχει, δηλαδή, πληροφορίες εξειδικευμένες για τη συγκεκριμένη υπηρεσία).

- Στάδιο προσδιορισμού του παράγοντα «κόστος» (Find Cost):

Στο προηγούμενο στάδιο, προσδιορίστηκε το σύνολο των υποστηριζόμενων επιπέδων ποιότητας και πραγματικής ποιότητας καθενός access point. Επόμενο βήμα είναι ο προσδιορισμός του (ανά μονάδα όγκου ή χρόνου) κόστους που αντιστοιχεί σε καθένα από τα επίπεδα αυτά.

Όπως είδαμε στην παράγραφο 4.1.5, το κόστος παροχής μιας υπηρεσίας εξαρτάται άμεσα από το είδος της υπηρεσίας αυτής, το ονομαστικό επίπεδο ποιότητας, το είδος της τεχνολογίας και τον πάροχο που την προσφέρει. Το IIS module, λοιπόν, γνωρίζοντας όλα τα παραπάνω στοιχεία για ένα access point, ανατρέχει στην κεντρική τράπεζα πληροφοριών και, συγκεκριμένα, στον κατάλογο κόστους υπηρεσίας (αρχείο *costs.xml*, σύμφωνα και με την παράγραφο 4.1.6) και ανακτά το αντίστοιχο κόστος.

Στο σημείο αυτό, τίθεται ένα σημαντικό ζήτημα: Όπως είδαμε από το αρχείο *costs.xml*, οι τιμές που περιέχει αντικατοπτρίζουν ανά μονάδα όγκου ή ανά μονάδα χρόνου κόστος. Για παράδειγμα, για την υπηρεσία Voice Call (κωδικός 1), στο επίπεδο ποιότητας 1, για τον πάροχο Cosmote (κωδικός 1) και με την τεχνολογία GSM (κωδικός 1), η τιμή του κόστους είναι 0.003. Η τάξη μεγέθους της παραπάνω τιμής, όμως, διαφέρει ριζικά από την τάξη μεγέθους των τυπικών τιμών που λαμβάνει ο παράγοντας «πραγματική ποιότητα», με τον οποίο και πρέπει να συγκριθεί, στα πλαίσια του υπολογισμού της αντικειμενικής συνάρτησης.

Γίνεται φανερό ότι, προκειμένου να μπορούν να συγκριθούν με επιτυχία αυτά τα δύο μεγέθη (πραγματική ποιότητα και κόστος), πρέπει οι τιμές του κόστους να «κανονικοποιηθούν», έτσι ώστε να έχουν την ίδια τάξη μεγέθους και να έχει νόημα η σύγκριση. Σε κάθε άλλη περίπτωση, προφανώς, οι επιλογές με υψηλότερο επίπεδο ποιότητας θα υπερτερούσαν πάντα έναντι αυτών με χαμηλό επίπεδο ποιότητας, αφού το χαμηλό τους κόστους δε θα είχε καμία θετική επίδραση.

Πώς, όμως, μπορεί να γίνει μια τέτοια κανονικοποίηση; Το ζητούμενο είναι ο παράγοντας «κόστος» να παίρνει τιμές στο ίδιο πεδίο τιμών με αυτό του παράγοντα της πραγματικής ποιότητας. Ένας τρόπος για να επιτευχθεί αυτό είναι η διαίρεση του

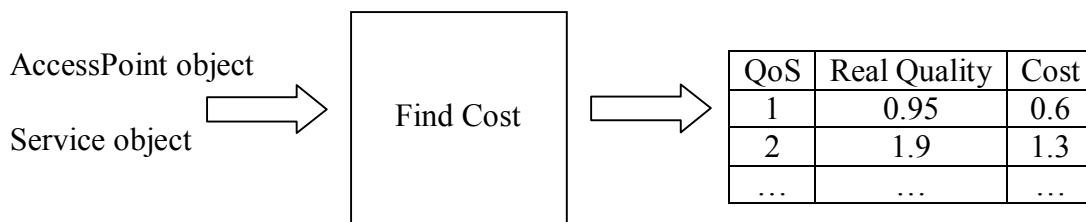
εκάστοτε κόστους που ανακτούμε από το αρχείο με το μέγιστο κόστος που συναντούμε στην υπό εξέταση υπηρεσία, ανεξάρτητα από τεχνολογία, πάροχο και επίπεδο ποιότητας. Αυτή η επιλογή μάς δίνει τη δυνατότητα να συγκρίνουμε καλύτερα επιλογές από διαφορετική τεχνολογία, πάροχο και επίπεδο ποιότητας. Η παραπάνω ενέργεια, όμως, δεν είναι αρκετή, διότι, προκειμένου να φέρουμε το κόστος στην ίδια κλίμακα με αυτήν της πραγματικής ποιότητας, πρέπει να κάνουμε ένα ακόμη βήμα: Να πολλαπλασιάσουμε το αποτέλεσμα της παραπάνω διαίρεσης με το συνολικό αριθμό διαθέσιμων επιπέδων ποιότητας για τη συγκεκριμένη υπηρεσία. Έτσι, τόσο η πραγματική ποιότητα όσο και το κόστος λαμβάνουν τιμές στην περιοχή από μηδέν έως και το μέγιστο αριθμό επιπέδων ποιότητας και, ως εκ τούτου, καθίσταται δυνατή μια πιο «δίκαιη» σύγκρισή τους.

Ακολουθεί η μαθηματική διατύπωση της διαδικασίας κανονικοποίησης:

- ✓ Έστω **maxCost** το μέγιστο κόστος που εμφανίζεται σε μια υπηρεσία.
- ✓ Έστω **noQoSLevels** ο συνολικός αριθμός διαθέσιμων επιπέδων ποιότητας για αυτήν την υπηρεσία.
- ✓ Η κανονικοποιημένη τιμή **Cost(p,q)** του κόστους, με  $p \in P$  και  $q \in Q(p)$ , συναρτήσει της τιμής **actualCost(p,q)**, με  $p \in P$  και  $q \in Q(p)$ , που λαμβάνεται από την κεντρική τράπεζα πληροφοριών, δίνεται από τη σχέση:

$$Cost(p,q) = \frac{noQoSLevels}{max\ Cost} \times actualCost(p,q)$$

Η πληροφορία αυτή, αναφορικά με το κόστος καθενός επιπέδου ποιότητας, εγγράφεται επίσης στα αντικείμενα τύπου AccessPoint. Σχηματικά, η διαδικασία που περιγράφηκε, καθώς και η εγγραφή της πληροφορίας στα AccessPoint αντικείμενα, παρουσιάζεται παρακάτω:



▪ Υπολογισμός αντικειμενικής συνάρτησης (Objective Function):

Έχοντας προσδιορίσει την πραγματική ποιότητα και το κόστος που αντιστοιχεί σε κάθε υποστηριζόμενο ονομαστικό επίπεδο ποιότητας, το μόνο που απομένει, πριν τον υπολογισμό της αντικειμενικής συνάρτησης, είναι ο προσδιορισμός του τρίτου της παράγοντα, που αντιστοιχεί στον πάροχο. Σε αυτό το βήμα, το IIS module ελέγχει αν ο πάροχος ενός access point συμπίπτει με τον πάροχο που έχει καθορίσει ως προτιμώμενο ο χρήστης. Στην περίπτωση αυτή, θα πρέπει, όπως έχουμε πει, να προμοδοτηθεί το συγκεκριμένο access point έναντι εκείνων που ανήκουν σε διαφορετικό πάροχο.

Ας θυμηθούμε, στο σημείο αυτό, τον τύπο της αντικειμενικής συνάρτησης (παράγραφος 4.2.2):

$$\begin{aligned} \text{Objective\_Function}(p, q) = & \text{qualityFactor} \times \text{RealQuality}(p, q) - \\ & \text{costFactor} \times \text{Cost}(p, q) + \\ & \text{providerFactor} \times \text{Provider}(p) \end{aligned}$$

όπου  $p \in P$  και  $q \in Q(p)$ .

Οι τιμές των παραγόντων  $\text{RealQuality}(p, q)$  και  $\text{Cost}(p, q)$  έχουν προσδιορισθεί στα δύο προηγούμενα στάδια. Η τιμή του παράγοντα  $\text{Provider}(p)$  θα πρέπει να είναι μηδενική σε περίπτωση που η προτίμηση του χρήστη δεν ταιριάζει με τον πάροχο του σημείου πρόσβασης  $p$ . Αντίθετα, πρέπει να είναι μη μηδενική όταν υπάρχει «ταίριασμα».

Τίθεται, ωστόσο, εδώ ένα ακόμη ζήτημα: Ποια πρέπει να είναι αυτή η μη μηδενική τιμή του παράγοντα  $\text{Provider}(p)$ ; Δύο είναι οι επικρατέστερες εκδοχές: Σύμφωνα με την πρώτη, το πριμ που θα δίνεται στα access points που ανήκουν στον προτιμώμενο πάροχο θα έχει μια σταθερή και συγκεκριμένη τιμή, ενώ στη δεύτερη το πριμ αυτό θα εξαρτάται από την τιμή της διαφοράς:

$$\text{qualityFactor} \times \text{RealQuality}(p, q) - \text{costFactor} \times \text{Cost}(p, q)$$

και θα είναι ένα ποσοστό της. Με άλλα λόγια, θα είναι ένα ποσοστό της «βαθμολογίας» που θα έπαιρνε το access point σε περίπτωση που δε λαμβανόταν υπόψη κατά τον προσδιορισμό της ο τρίτος παράγοντας.

Το μειονέκτημα της πρώτης εκδοχής είναι προφανές: Είναι δύσκολο να βρεθεί μια σταθερή τιμή που να είναι της ίδιας τάξης μεγέθους με την εκάστοτε τιμή της διαφοράς  $\text{qualityFactor} \times \text{RealQuality}(p, q) - \text{costFactor} \times \text{Cost}(p, q)$ . Ακόμη και αν προσδιορισθεί μια τέτοια τιμή, η επιλογή της θα είναι αυθαίρετη, με κίνδυνο να αλλοιώσει το αποτέλεσμα με «άδικο» για τα υπόλοιπα access points τρόπο. Αντίθετα, η δεύτερη εκδοχή ακολουθεί παρόμοια προσέγγιση με αυτήν που ακολουθήθηκε κατά τη διαδικασία κανονικοποίησης του κόστους. Δηλαδή, το πριμ είναι τέτοιο, ώστε τελικά ο παράγοντας  $\text{Provider}(p)$  να είναι της ίδιας τάξης μεγέθους με την εν λόγω διαφορά και, επομένως, η σύγκριση να είναι πιο δίκαιη.

Έτσι, λοιπόν, σε περίπτωση που ο παράγοντας πάροχος πρέπει να ληφθεί υπόψη, η συνεισφορά του επελέγη να είναι, στην υλοποίηση που ακολουθήθηκε, 50% επί της διαφοράς  $\text{qualityFactor} \times \text{RealQuality}(p, q) - \text{costFactor} \times \text{Cost}(p, q)$ , χωρίς βεβαίως και να ξεχνάμε και το συντελεστή  $\text{providerFactor}$ , για τον οποίο θα γίνει λόγος παρακάτω. Συνεπώς, έχουμε, σύμφωνα με τα παραπάνω:

$$\text{Provider}(p) = \begin{cases} 0, & \text{αν ο παροχος του access point } p \text{ δε συμπιπτει με τον προτιμωμενο} \\ 0.5 \times \text{Abs}[\text{qualityFactor} \times \text{RealQuality}(p, q) - \text{costFactor} \times \text{Cost}(p, q)], & \\ \text{αν συμπιπτει} \end{cases}$$

όπου με  $Abs[x]$  συμβολίζεται η απόλυτη τιμή του μεγέθους  $x$ .

Την τιμή του ποσοστού αυτού (50%) την ανακτά το IIS module από ένα properties file, με όνομα *provider\_bonus.txt*, το οποίο έχει την ακόλουθη τετριμμένη μορφή:

```
#provider_bonus.txt
#This file contains the percentage of the provider bonus
#in case that there is a match with the user's preferences
bonus=0.5
```

Κατά αυτόν τον τρόπο, αν επιθυμούμε να διαφοροποιήσουμε τη συνεισφορά του παράγοντα «πάροχος» δεν έχουμε παρά να αλλάξουμε την τιμή μέσα σε αυτό το αρχείο, χωρίς να χρειάζεται κάποια επέμβαση στον κώδικα του IIS.

Φτάνουμε, τώρα, στο τελικό βήμα της διαδικασίας, στον υπολογισμό, δηλαδή, της τιμής της αντικειμενικής συνάρτησης, για κάθε υποψήφια επιλογή ανάθεσης. Το μόνο που υπολείπεται, πλέον, είναι ο καθορισμός των συντελεστών *qualityFactor*, *costFactor* και *providerFactor*, οι οποίοι αντανakλούν τη βαρύτητα που δίνει ο χρήστης στους αντίστοιχους παράγοντες. Όπως είδαμε στο κεφάλαιο 3, ο χρήστης, μέσω της γραφικής διεπαφής, ταξινομεί τους παράγοντες «ποιότητα», «κόστος» και «πάροχος». Η θέση των παραγόντων σε αυτήν την ιεραρχημένη λίστα είναι εκείνη που καθορίζει τους αντίστοιχους συντελεστές. Για παράδειγμα, εάν ο χρήστης τοποθετήσει στη λίστα πρώτα την «ποιότητα», μετά το «κόστος» και, κατόπιν, τον «πάροχο», θα πρέπει να ισχύει η συνθήκη:

$$qualityFactor > costFactor > providerFactor$$

Στην προσέγγιση που ακολουθήθηκε, ο πρώτος εκ των τριών παραγόντων ορίστηκε να έχει βαρύτητα 80%, ο δεύτερος 60% και ο τρίτος 40%. Συνεπώς, για το παραπάνω παράδειγμα, θα ισχύουν τα εξής:

$$qualityFactor = 0.8$$

$$costFactor = 0.6$$

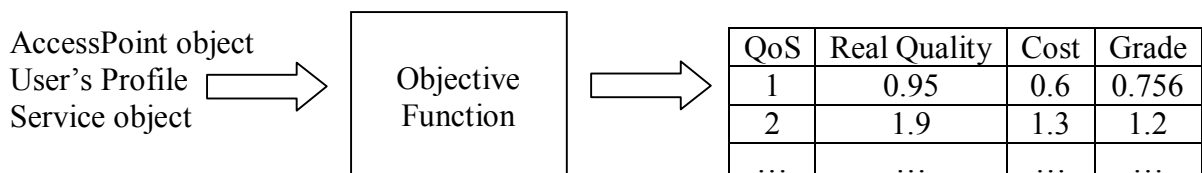
$$providerFactor = 0.4$$

$$Objective\_Function(p, q) = 0.8 \times RealQuality(p, q) - \\ 0.6 \times Cost(p, q) + \\ 0.4 \times Provider(p)$$

Οι τιμές των τριών αυτών συντελεστών (0.8, 0.6, 0.4) καταγράφονται σε αρχείο της κεντρικής τράπεζας πληροφοριών, έτσι ώστε το IIS module και να έχει άμεση πρόσβαση σε αυτούς, αλλά και να μπορούν πολύ εύκολα να αλλάξθούν. Το αρχείο αυτό είναι ένα απλό properties file, με όνομα *factors.txt* και την κάτωθι μορφή:

```
#factors.txt
#This file contains the factors used by the objective function
#a is the most important factor, while c is the least important
a=0.8
b=0.6
c=0.4
```

Γνωρίζοντας, πλέον, το IIS module όλα όσα πρέπει να ξέρει, προχωρά στον υπολογισμό της τιμής της αντικειμενικής συνάρτησης, για κάθε επιλογή ανάθεσης ξεχωριστά. Έτσι, για κάθε ένα επιτρεπτό επίπεδο ποιότητας ενός access point, υπολογίζεται και ένας βαθμός που αντικατοπτρίζει πρακτικά το κατά πόσο συμφέρει το τερματικό η συγκεκριμένη επιλογή ανάθεσης. Ο συγκεκριμένος βαθμός αποθηκεύεται και αυτός στο AccessPoint object, συμπληρώνοντας έτσι το σύνολο των πληροφοριών που απαιτούνται για την τελική επιλογή. Σχηματικά, το στάδιο επεξεργασίας που μόλις περιγράφηκε έχει ως εξής:



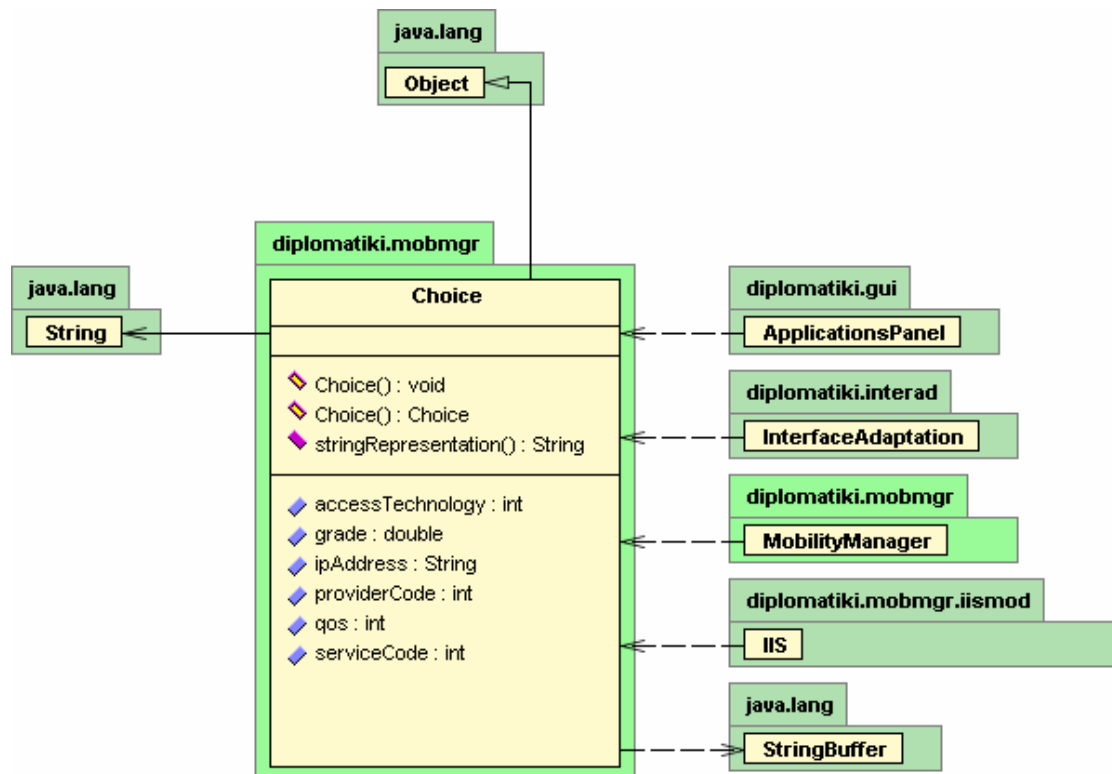
▪ Εύρεση της βέλτιστης λύσης (Find Best):

Όταν τα στάδια «Find Quality», «Find Cost» και «Objective Function» πραγματοποιηθούν για όλα τα access points, το μόνο που απομένει είναι η ταξινόμηση των διαφόρων επιλογών ανάθεσης κατά φθίνουσα σειρά βαθμολογίας, έτσι ώστε στην πρώτη θέση να βρίσκεται η βέλτιστη λύση και στην τελευταία η λιγότερο συμφέρουσα.

Για το σκοπό αυτό, το IIS module διατρέχει, αρχικά, ολόκληρη τη λίστα AccessPointList και για κάθε μία δυνατή επιλογή ανάθεσης κατασκευάζει ένα αντικείμενο τύπου Choice. Η κλάση Choice αποτελεί την αναπαράσταση, σε επίπεδο κώδικα, μιας επιλογής ανάθεσης. Για το λόγο αυτό, περιλαμβάνει τις εξής πληροφορίες:

- ✓ Τον κωδικό της αιτούμενης/υπό ανάθεση υπηρεσίας
- ✓ Το επίπεδο ποιότητας στο οποίο προτείνεται ανατεθεί η υπηρεσία
- ✓ Τη διεύθυνση IP του σημείου πρόσβασης στο οποίο προτείνεται να ανατεθεί
- ✓ Τον κωδικό της τεχνολογίας που χρησιμοποιεί αυτό το σημείο πρόσβασης
- ✓ Ο κωδικός του παρόχου στον οποίο αυτό ανήκει
- ✓ Ο βαθμός της συγκεκριμένης επιλογής

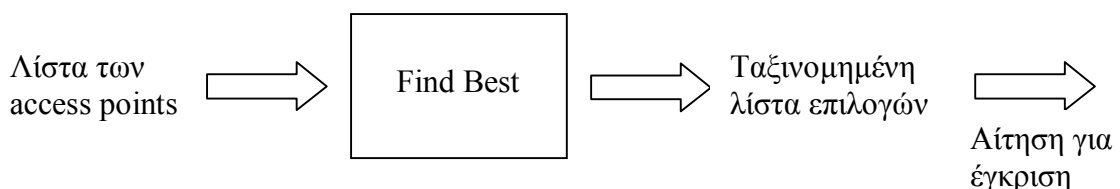
Το διάγραμμα της κλάσης αυτής φαίνεται στο επόμενο σχήμα:



Όπως είναι εμφανές, λοιπόν, για κάθε ένα αντικείμενο τύπου `AccessPoint` που περιέχεται στην `AccessPointList`, θα κατασκευαστούν τόσα αντικείμενα τύπου `Choice` όσα είναι τα επιτρεπτά επίπεδα ποιότητας που περιλαμβάνει ο πίνακας των πληροφοριών του. Τα καινούρια αυτά αντικείμενα, τύπου `Choice`, προστίθενται σε μια καινούρια λίστα, με όνομα `choices`. Η λίστα `choices` είναι, λοιπόν, η λίστα όλων των δυνατών επιλογών ανάθεσης και έχει μέγεθος μεγαλύτερο ή ίσο της `AccessPointList`.

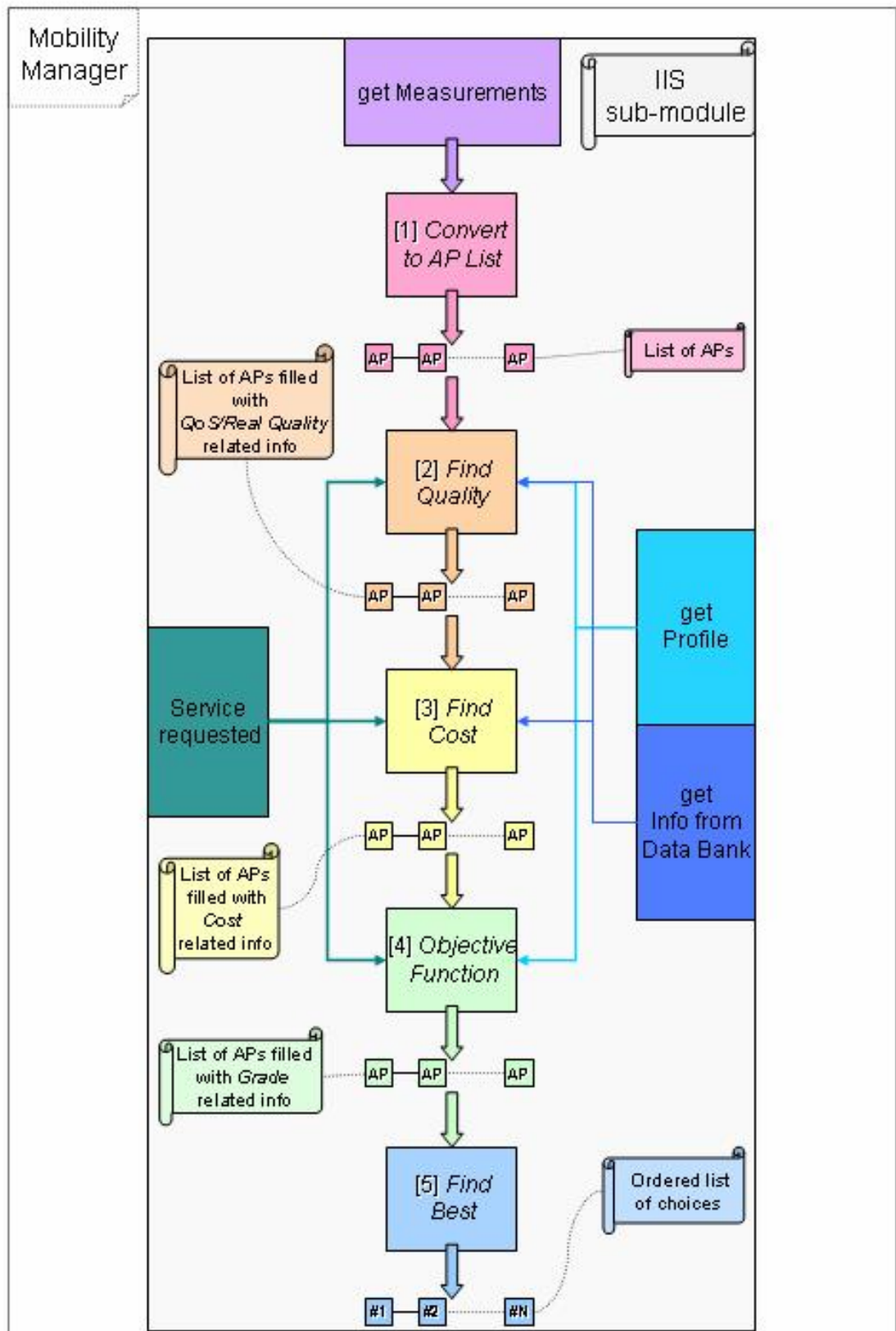
Το μοναδικό βήμα που απομένει είναι η ταξινόμηση, κατά φθίνουσα σειρά βαθμολογίας, της λίστας `choices`. Το βήμα αυτό είναι εξαιρετικά απλό και μπορεί να πραγματοποιηθεί με τη βοήθεια ενός από τους γνωστούς αλγόριθμους ταξινόμησης (όπως, για παράδειγμα, με τον αλγόριθμο «BubbleSort» που χρησιμοποιήθηκε στη συγκεκριμένη υλοποίηση).

Το τελευταίο αυτό στάδιο του αλγορίθμου του IIS παρουσιάζεται εποπτικά στο διάγραμμα που ακολουθεί:



Η ταξινομημένη λίστα επιλογών διοχετεύεται στο `Mobility Manager` module, προκειμένου το τελευταίο να αναλάβει την αποστολή αίτησης έγκρισης προς την οντότητα δικτύου.

Συγκεντρωτικά, η όλη διαδικασία που ακολουθείται από το IIS module και περιγράφηκε στην παράγραφο αυτή συνοψίζεται στο διάγραμμα που ακολουθεί ():

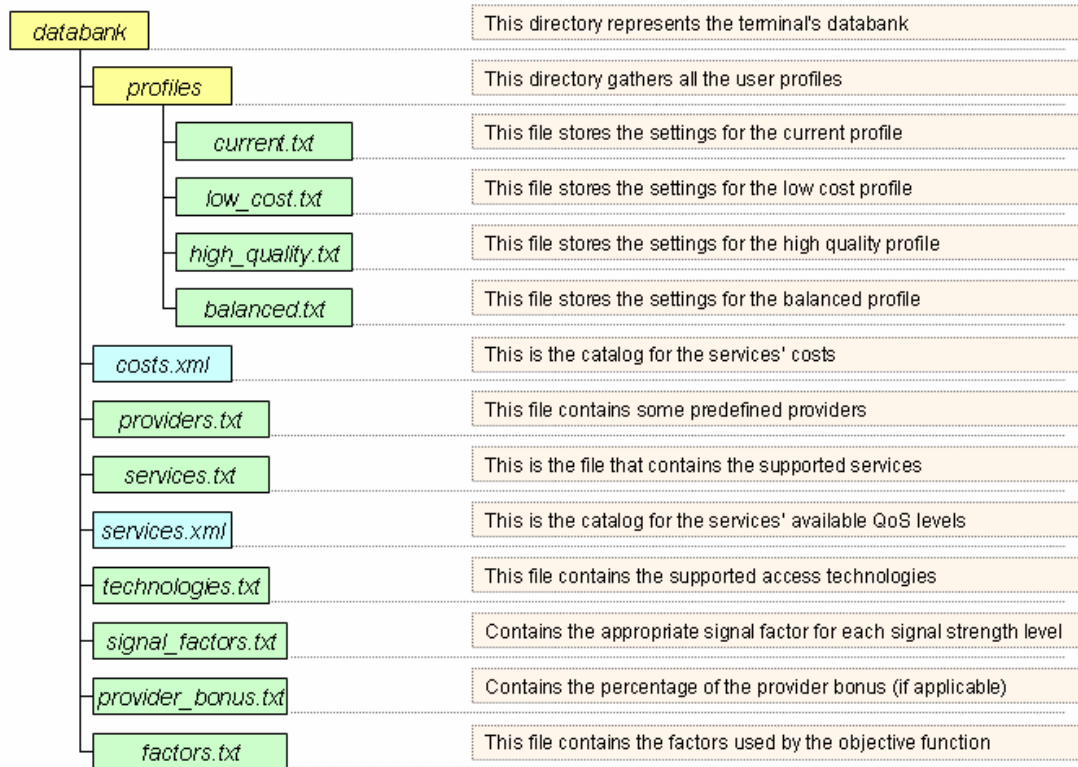


Διάγραμμα Απεικόνισης του IIS Αλγορίθμου (AP = Access Point)



#### 4.2.4 Ολοκληρωμένη Μορφή της Τράπεζας Πληροφοριών

Κλείνουμε την παράγραφο 4.2 με την παρουσίαση της ολοκληρωμένης μορφής της κεντρικής τράπεζας πληροφοριών. Σύμφωνα με τα όσα αναφέρθηκαν στα προηγούμενα, το σχήμα που παρατέθηκε στην παράγραφο 4.1.6 πρέπει να συμπληρωθεί και να έχει ως εξής:





## 5. Ειδικά Θέματα Υλοποίησης

### 5.1 Εισαγωγή

Στο κεφάλαιο 2, δόθηκε μία σύντομη περιγραφή των συνιστωσών που απαρτίζουν την αρχιτεκτονική διαχείρισης ενός τερματικού. Στο κεφάλαιο 3, παρουσιάστηκε λεπτομερώς η συνιστώσα προτιμήσεων χρήστη, ενώ, στο κεφάλαιο 4, περιγράφηκε, με αναλυτικό τρόπο, η διαδικασία ευφυούς επιλογής δικτύου πρόσβασης.

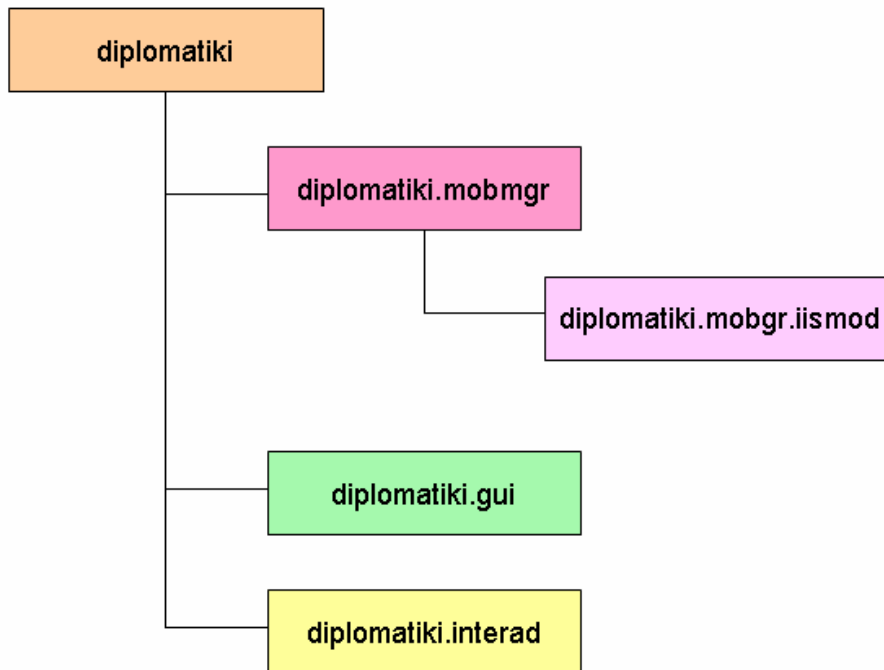
Στις παραγράφους που ακολουθούν θα ασχοληθούμε με ορισμένα ειδικά θέματα της υλοποίησης που ακολουθήθηκε, αναλύοντας με περισσότερη λεπτομέρεια, από τεχνικής πλευράς, τις διάφορες συνιστώσες (modules) του συστήματος.

Κάθε module του συστήματος διαχείρισης του τερματικού υλοποιείται, λοιπόν, και από ένα διαφορετικό πακέτο κλάσεων της Java. Με αυτόν τον τρόπο, διαφυλάσσεται η αυτονομία του κάθε module και ακολουθείται επακριβώς (και στην πράξη) η θεωρητική δόμηση του συστήματος.

Έτσι, μια πρώτη ματιά στον πηγαίο κώδικα φανερώνει αμέσως ότι η αντιστοιχία modules – πακέτων έχει ως εξής:

- ✓ *Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου (Interface Adaptation module – IA):* Υλοποιείται από το πακέτο *diplomatiki.interad*.
- ✓ *Συνιστώσα Διαχείρισης Κινητικότητας (Mobility Manager module):* Υλοποιείται από το πακέτο *diplomatiki.mobmgr*.
- ✓ *Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης (IIS sub-module):* Υλοποιείται από το πακέτο *diplomatiki.mobmgr.iismod*, που εμπεριέχεται στο προηγούμενο.
- ✓ *Γραφική Διεπαφή (Graphical User Interface – GUI):* Υλοποιείται από το πακέτο *diplomatiki.gui*. Το γραφικό κομμάτι της Συνιστώσας Προτιμήσεων Χρήστη (User Preferences module) υλοποιείται, επίσης, σε αυτό το πακέτο.
- ✓ *Οντότητα Διαχείρισης του Δικτύου (Network Management System Entity):* Υλοποιείται από το πακέτο *networkside.network*. Σημειωτέον ότι το πακέτο αυτό είναι τελείως ανεξάρτητο από τα υπόλοιπα, δε χρησιμοποιεί, δηλαδή, καμία απολύτως κλάση τους, αφού βρίσκεται σε οντότητα του δικτύου και όχι στο τερματικό. Αποτελεί, με άλλα λόγια, κομμάτι του λογισμικού των οντοτήτων του δικτύου και όχι της αρχιτεκτονικής διαχείρισης ενός τερματικού.

Όπως παρατηρούμε, λοιπόν, το πακέτο *diplomatiki*, μαζί με τα υπο-πακέτα του, συγκεντρώνει όλες τις κλάσεις που πρέπει να ενσωματωθούν στο λογισμικό διαχείρισης ενός τερματικού. Το σχήμα που ακολουθεί απεικονίζει τη γενική δομή του πακέτου *diplomatiki*.



Στις παραγράφους που ακολουθούν, θα αναφερθούμε σε καθένα από τα παραπάνω modules ξεχωριστά. Για ακόμα περισσότερες λεπτομέρειες, μπορεί κανείς να ανατρέξει στον πηγαίο κώδικα, στο παράρτημα της εργασίας.

## 5.2 Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου

### 5.2.1 Γενική Περιγραφή

Το IA module είναι ιδιαίτερος σημαντικό, διότι καθιστά το σύστημα ικανό να τρέχει ολοκληρωμένες προσομοιώσεις. Χάρη στο module αυτό, επιτυγχάνεται, πρώτα από όλα, η ανάκτηση των μετρήσεων που αφορούν τις συνθήκες του δικτύου και η «τροφοδοσία» τους στην IIS συνιστώσα. Επίσης, όπως έχουμε εξηγήσει, το module έχει την ικανότητα να ειδοποιεί, σε ορισμένες περιπτώσεις, το Mobility Manager για την ανάγκη εκκίνησης του αλγορίθμου του IIS.

Το πακέτο *diplomatiki.interad* περιλαμβάνει μία μόνο κλάση, την *InterfaceAdaptation*, η οποία και υλοποιεί το IA module. Αναφέρουμε, αμέσως πιο κάτω, τα βασικά συστατικά της υλοποίησης του συγκεκριμένου module, που είναι τα εξής:

- ✓ Ένα αρχείο που περιέχει μετρήσεις, προσομοιώνοντας έτσι τη λήψη μετρήσεων από το περιβάλλον του τερματικού.
- ✓ Ένας πίνακας (*currentMeasurements*) που εμπεριέχει το τρέχον σετ μετρήσεων.
- ✓ Ένας πίνακας (*previousMeasurements*) που εμπεριέχει το αμέσως προηγούμενο (χρονικά) σετ μετρήσεων.
- ✓ Μία μέθοδος (με όνομα *measure()*) που φροντίζει να συμπληρώνει τους δύο παραπάνω πίνακες, διαβάζοντας το αρχείο μετρήσεων.

- ✓ Μέθοδοι που «επιστρέφουν» (*get*) ή «θέτουν» (*set*) τους δύο προαναφερθέντες πίνακες.
- ✓ Μέθοδοι που ελέγχουν για την εμφάνιση καινούριου σημείου πρόσβασης ή την εξαφάνιση ενός χρησιμοποιούμενου.

### 5.2.2 Αρχείο Μετρήσεων

Σε μια προσομοίωση, η λήψη των μετρήσεων επιτυγχάνεται με την ανάγνωση από το IA module ενός αρχείου. Το αρχείο αυτό έχει ειδική δομή, είναι τύπου XML, και περιέχει τις μετρήσεις.

Οι μετρήσεις είναι κωδικοποιημένες στο αρχείο με ένα συγκεκριμένο τρόπο. Ένα αρχείο μετρήσεων αντιστοιχεί, συνήθως, και σε ένα τυπικό σενάριο λειτουργίας του τερματικού. Ενδεικτικό παράδειγμα ενός τέτοιου σεναρίου είναι η μετάβαση ενός εργαζόμενου από το σπίτι στη δουλειά του.

Οι μετρήσεις χωρίζονται σε διάφορα σετ, καθένα από τα οποία υποτίθεται ότι αντιστοιχεί και σε μια διαφορετική χρονική περίοδο του σεναρίου. Προτού προβούμε σε περαιτέρω ανάλυση, ας δούμε ένα τμήμα ενός τέτοιου αρχείου μετρήσεων:

---

```
<?xml version="1.0"?>
<measurements>
  <measurement timer="0"><!--HOME-->
    <AP ip="147.102.1.1" providerCode="1" technology="1"
      signal="5" availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1"
      signal="5" availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2"
      signal="5" availBandwidth="2000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2"
      signal="5" availBandwidth="2000"/>
    <AP ip="147.102.4.1" providerCode="5" technology="4"
      signal="5" availBandwidth="4000"/>
    <AP ip="147.102.4.2" providerCode="6" technology="4"
      signal="5" availBandwidth="4000"/>
  </measurement>

  <measurement timer="1"><!--HOME => CAR-->
    <AP ip="147.102.1.1" providerCode="1" technology="1"
      signal="2" availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1"
      signal="2" availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2"
      signal="3" availBandwidth="2000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2"
      signal="2" availBandwidth="2000"/>
    <AP ip="147.102.4.1" providerCode="5" technology="4"
      signal="1" availBandwidth="4000"/>
    <!--NEW-->
    <AP ip="147.102.1.11" providerCode="1" technology="1"
      signal="3" availBandwidth="170"/>
  </measurement>
</measurements>
```

```

    <AP ip="147.102.1.12" providerCode="2" technology="1"
    signal="3" availBandwidth="170"/>
    <AP ip="147.102.2.11" providerCode="1" technology="2"
    signal="3" availBandwidth="2000"/>
    <AP ip="147.102.2.12" providerCode="3" technology="2"
    signal="3" availBandwidth="2000"/>
</measurement>

<measurement timer="2"><!--CAR-->
    <AP ip="147.102.1.11" providerCode="1" technology="1"
    signal="4" availBandwidth="170"/>
    <AP ip="147.102.1.12" providerCode="2" technology="1"
    signal="5" availBandwidth="170"/>
    <AP ip="147.102.2.11" providerCode="1" technology="2"
    signal="4" availBandwidth="2000"/>
    <AP ip="147.102.2.12" providerCode="3" technology="2"
    signal="5" availBandwidth="2000"/>
</measurement>
<!--...-->
</measurements>

```

---

Κάθε σετ μετρήσεων προσδιορίζεται μεταξύ των ετικετών (tags) *<measurement>* και *</measurement>*. Η ιδιότητα (attribute) *timer* καθορίζει τη χρονική περίοδο στην οποία αντιστοιχεί το σετ των μετρήσεων. Έτσι, το πρώτο σετ μετρήσεων θα ισχύει για την πρώτη χρονική περίοδο (*timer* = 0), το δεύτερο για την αμέσως επόμενη (*timer* = 1), κοκ. Μέσα από το πρόγραμμα, μπορεί να καθοριστεί πόσο θα διαρκεί μία τέτοια χρονική περίοδος, για παράδειγμα 30 δευτερόλεπτα.

Με τον τρόπο αυτό, επιτυγχάνεται η επίδειξη μιας χρονικής αλληλουχίας γεγονότων σε ένα σενάριο προσομοίωσης. Κατά συνέπεια, όλες οι αιτήσεις υπηρεσιών που θα συμβούν τα πρώτα 30 δευτερόλεπτα (πρώτη χρονική περίοδος) θα εξεταστούν από τον αλγόριθμο IIS βάσει του πρώτου σετ μετρήσεων. Όλες οι αιτήσεις που θα συμβούν τα επόμενα 30 δευτερόλεπτα θα εξεταστούν βάσει του δεύτερου σετ μετρήσεων, κοκ.

Όπως μπορούμε να παρατηρήσουμε, ένα στοιχείο (element) τύπου *measurement* περιλαμβάνει, ως παιδιά του, πολλά στοιχεία τύπου *AP*. Τα στοιχεία αυτά αντιστοιχούν στα σημεία πρόσβασης (access points) τα οποία «βλέπει» το τερματικό τη δεδομένη χρονική περίοδο. Οι πληροφορίες που συνοδεύουν κάθε σημείο πρόσβασης είναι πέντε και είναι οι εξής:

- ✓ IP Διεύθυνση (attribute *ip*)
- ✓ Κωδικός Παρόχου (attribute *providerCode*)
- ✓ Τεχνολογία Πρόσβασης (attribute *technology*)
- ✓ Επίπεδο Ισχύος του Λαμβανόμενου Σήματος (attribute *signal*)
- ✓ Διαθέσιμο Εύρος Ζώνης (attribute *availBandwidth*)

Οι τιμές των ιδιοτήτων των στοιχείων *AP* είναι αυτές που αντικατοπτρίζουν το δικτυακό περιβάλλον στο οποίο κινείται το τερματικό.

Ένα σημείο που αξίζει να θίξουμε είναι ότι, σε όλα τα αρχεία μετρήσεων που κατασκευάστηκαν για τους σκοπούς της παρούσας εργασίας, οι διευθύνσεις IP των εκάστοτε σημείων πρόσβασης έχουν επιλεγεί βάσει της εξής λογικής:

- ✓ Τα δύο πρώτα bytes είναι πάντοτε 147.102.
- ✓ Η τιμή του τρίτου byte εξαρτάται από την τεχνολογία πρόσβασης την οποία υποστηρίζει το σημείο πρόσβασης. Αν είναι τύπου GSM, τότε έχει την τιμή 1. αν είναι τύπου UMTS, τότε έχει την τιμή 2. Αν είναι τύπου WLAN, τότε έχει την τιμή 3. Αν είναι τύπου DVB, τότε έχει την τιμή 4.
- ✓ Το τελευταίο byte διαφοροποιεί σημεία πρόσβασης της ίδιας τεχνολογίας.

Τονίζεται ότι η παραπάνω σύμβαση ακολουθήθηκε για τη διευκόλυνση της παρακολούθησης των αποτελεσμάτων των προσομοιώσεων και μόνο. Η υλοποίηση δεν εξαρτάται σε καμία περίπτωση από τη μορφή των διευθύνσεων των διαθέσιμων σημείων πρόσβασης.

Αξίζει να σημειωθεί, τέλος, ότι όλα όσα αναφέρθηκαν στην παράγραφο αυτή θα φανούν χρήσιμα και στην κατανόηση του επόμενου κεφαλαίου (κεφάλαιο 6), που ασχολείται με την εκτέλεση προσομοιώσεων και την εξαγωγή αποτελεσμάτων.

### 5.2.3 Πίνακες Μετρήσεων

Το IA module περιέχει δύο πολύ σημαντικούς πίνακες. Ο ένας (*currentMeasurements*) περιλαμβάνει το τρέχον σετ μετρήσεων, δηλαδή αυτό που αντιστοιχεί στην τρέχουσα χρονική περίοδο, ενώ ο δεύτερος (*previousMeasurements*) περιλαμβάνει το αμέσως προηγούμενο σετ μετρήσεων, δηλαδή εκείνο που αντιστοιχούσε στην προηγούμενη χρονική περίοδο.

Οι πίνακες αυτοί έχουν πέντε στήλες, όσα είναι και τα χαρακτηριστικά ενός σημείου πρόσβασης. Η μορφή τους, λοιπόν, είναι η εξής:

IP Address	Provider	Technology	Signal Strength	Available Bandwidth
147.102.1.1	1	1	5	170
...	...	...	...	...

Ο πίνακας *currentMeasurements* αποτελεί και τη διεπαφή του IA module προς την πλευρά του Mobility Manager και του IIS. Όποτε χρειάζεται, δηλαδή, να τρέξει ο IIS αλγόριθμος, χρησιμοποιεί ως είσοδο τον πίνακα αυτόν.

Ο λόγος για τον οποίο το τερματικό διατηρεί στη μνήμη του και τον πίνακα *previousMeasurements* είναι για να μπορεί να ελέγξει (συγκρίνοντας τον πίνακα *currentMeasurements* με τον *previousMeasurements*) αν έχει εμφανιστεί κάποιο καινούριο σημείο πρόσβασης. Αν, πράγματι, συμβαίνει κάτι τέτοιο, τότε ίσως το καινούριο σημείο πρόσβασης να αποτελεί μια καλύτερη επιλογή ανάθεσης για κάποια ή κάποιες από τις τρέχουσες υπηρεσίες.

#### 5.2.4 Μέθοδος *measure()*

Η μέθοδος αυτή αποτελεί, ίσως, τον πυρήνα του IA module, αφού αναλαμβάνει την κατασκευή του πίνακα τρεχουσών μετρήσεων (*currentMeasurements*), με βάση το αρχείο μετρήσεων. Με άλλα λόγια, διαβάζει το XML έγγραφο και καταχωρεί στον πίνακα *currentMeasurements* τις τρέχουσες μετρήσεις.

#### 5.2.5 Μέθοδοι Ελέγχου Εμφάνισης ή Εξαφάνισης Σημείων Πρόσβασης

Ανάμεσα στα βασικά καθήκοντα του IA module είναι να παρακολουθεί μήπως «εμφανιστεί» ή «εξαφανιστεί» κάποιο σημείο πρόσβασης. Τις λειτουργίες αυτές αναλαμβάνουν να υλοποιήσουν οι μέθοδοι *checkForNewAP()* και *checkForAPLoss()*, αντίστοιχα.

Η πρώτη εκ των δύο ελέγχει αν όλες οι εγγραφές του πίνακα *currentMeasurements* υπάρχουν και στον πίνακα *previousMeasurements*. Αν όχι, τότε οι εγγραφές που δεν υπάρχουν αντιστοιχούν σε «καινούρια» σημεία πρόσβασης. Η δεύτερη μέθοδος παραλαμβάνει, αρχικά, από το Mobility Manager τη λίστα τρεχουσών υπηρεσιών. Εν συνεχεία, ελέγχει αν όλες οι υπηρεσίες της λίστας εξυπηρετούνται από σημεία πρόσβασης που υπάρχουν και στον πίνακα *currentMeasurements*. Αν κάτι τέτοιο δεν ισχύει για κάποια υπηρεσία, αυτό σημαίνει ότι, προκειμένου να συνεχίσει να λαμβάνεται η συγκεκριμένη υπηρεσία, θα πρέπει να γίνει διαπομπή, διότι το «παλαιό» σημείο πρόσβασης δε βρίσκεται πλέον στην εμβέλεια του κινητού τερματικού ή, γενικότερα, το σήμα από αυτό είναι πολύ υποβαθμισμένο.

Στην πρώτη περίπτωση (καινούριο σημείο πρόσβασης), ο Mobility Manager ειδοποιείται ώστε να εκτελεστεί ο αλγόριθμος του IIS για όλες τις τρέχουσες υπηρεσίες. Στη δεύτερη περίπτωση (απώλεια σημείου πρόσβασης), εκτελείται ο αλγόριθμος του IIS για όσες από τις τρέχουσες υπηρεσίες χρησιμοποιούσαν το σημείο πρόσβασης που εξαφανίστηκε.

#### 5.2.6 Συμπληρωματικά Στοιχεία

Αξίζει, τέλος, να σημειωθεί ότι το IA module τρέχει σε ξεχωριστό thread (νήμα). Το νήμα αυτό φροντίζει να εκτελεί περιοδικά τις μεθόδους που περιγράψαμε στις προηγούμενες παραγράφους.

### 5.3 Συνιστώσα Διαχείρισης Κινητικότητας

#### 5.3.1 Γενική Περιγραφή

Όπως έχει τονιστεί στο κεφάλαιο 2, ο Mobility Manager συνιστά το σπουδαιότερο module του συστήματος διαχείρισης ενός τερματικού, αφού αποτελεί, ουσιαστικά, την «καρδιά» του συστήματος αυτού, με άλλα λόγια είναι η συνιστώσα με τον



κεντρικότερο ρόλο. Το πακέτο *diplomatiiki.mobmgr*, εκτός από το υπο-πακέτο *dimplomatiiki.mobmgr.iismod*, περιέχει την κλάση *MobilityManager*, που είναι και η σημαντικότερη του εν λόγω πακέτου, καθώς και μερικές βοηθητικές κλάσεις, όπως είναι η κλάση *Choice*, για την οποία έχει γίνει ήδη λόγος στην παράγραφο 4.2.3.

Δύο είναι τα βασικά συστατικά της κλάσης *MobilityManager*, η οποία αποτελεί την αναπαράσταση, σε επίπεδο κώδικα, του Mobility Manager module:

- ✓ Η μέθοδος *executeIIS()*, η οποία αναλαμβάνει να φέρνει εις πέρας τις περισσότερες από τις λειτουργίες που πρέπει να επιτελεί ο Mobility Manager, η βασικότερη εκ των οποίων είναι, βέβαια, η ενεργοποίηση του IIS sub-module, όπως καταδεικνύει και το όνομα της εν λόγω μεθόδου.
- ✓ Μια λίστα με τις τρέχουσες υπηρεσίες, μαζί με ένα σύστημα προσπέλασης και επεξεργασίας της.

### 5.3.2 Μέθοδος *executeIIS()*

Η μέθοδος αυτή, που αποτελεί τον πυρήνα του Mobility Manager module, εκτελεί, βήμα προς βήμα, τις εξής ενέργειες:

- ✓ Σε πρώτη φάση, ανακτά τον πίνακα των μετρήσεων από το IA module.
- ✓ Στη συνέχεια, ενεργοποιεί το IIS module, το οποίο επιστρέφει στο Mobility Manager μία στοίβα ταξινομημένων επιλογών.
- ✓ Έπειτα, και αφού ελέγξει αν η στοίβα περιέχει στοιχεία ή είναι κενή (κάτι που θα σήμαινε ότι καμία σύνδεση δεν μπορεί να επιτευχθεί), εκκινεί τη διαδικασία επικοινωνίας με την πλησιέστερη οντότητα διαχείρισης του δικτύου. Κατά τη διαδικασία αυτή, εξάγει το πρώτο στοιχείο της στοίβας και το αποστέλλει στην οντότητα διαχείρισης του δικτύου. Εν συνεχεία, αναμένει την απάντηση του δικτύου. Αν η επιλογή γίνει αποδεκτή, η επικοινωνία τερματίζεται. Αν όχι, τότε η διαδικασία επαναλαμβάνεται με την υποβολή προς έγκριση της δεύτερης επικρατέστερης επιλογής (δεύτερο στοιχείο της στοίβας).
- ✓ Η ολοκλήρωση της επικοινωνίας με το δίκτυο σημαίνει ότι η εξυπηρέτηση της αιτούμενης υπηρεσίας ανατέθηκε σε συγκεκριμένο (το βέλτιστο δυνατό) σημείο πρόσβασης και επίπεδο ποιότητας. Επομένως, πρέπει να γίνουν δύο ακόμη σημαντικές ενέργειες: Πρώτον, να ενημερωθεί η λίστα των τρεχουσών υπηρεσιών και, δεύτερον, να ενημερωθεί το GUI με τις νέες πληροφορίες ανάθεσης.
- ✓ Προκειμένου να υλοποιηθεί το πρώτο, ελέγχεται, αρχικά, αν η υπηρεσία για την οποία ενεργοποιήθηκε ο αλγόριθμος του IIS συγκαταλέγεται στις τρέχουσες υπηρεσίες (δηλαδή, αν υπάρχει στη λίστα τρεχουσών υπηρεσιών) ή όχι. Αν επρόκειτο για τρέχουσα υπηρεσία, τότε οι πληροφορίες ανάθεσης που περιείχε η λίστα για την υπηρεσία αυτή πρέπει να αντικατασταθούν. Αν επρόκειτο για καινούρια υπηρεσία, τότε πρέπει να προστεθεί στη λίστα τρεχουσών υπηρεσιών.

- ✓ Προκειμένου να υλοποιηθεί το δεύτερο, κατασκευάζεται ένα αντικείμενο της κλάσης *InfoStruct* το οποίο περιέχει όλες τις απαραίτητες πληροφορίες ανάθεσης που πρέπει να απεικονιστούν στο GUI. Η μέθοδος *executeIIS()* επιστρέφει αυτό το αντικείμενο.

### 5.3.3 Λίστα Τρεχουσών Υπηρεσιών

Πρόκειται για μια λίστα αντικειμένων τύπου *Choice* (βλ. και παράγραφο 4.2.3), χάρη στην οποία το σύστημα διαχείρισης του τερματικού μπορεί να παρακολουθεί ποιες υπηρεσίες τρέχουν στο τερματικό και πού έχει ανατεθεί κάθε μία. Ο *Mobility Manager* φροντίζει να κρατά τη λίστα αυτή ενήμερη, ενώ, επίσης, προσφέρει μεθόδους για την προσπέλασή της, τη διαγραφή μιας υπηρεσίας από αυτήν και την επανεξέταση όλων των υπηρεσιών που περιλαμβάνει.

## 5.4 Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης

Η λειτουργία του *IIS sub-module*, δηλαδή της συνιστώσας ευφυούς επιλογής δικτύου πρόσβασης, περιγράφηκε αναλυτικά στο κεφάλαιο 4, στο οποίο παρουσιάστηκε, βήμα προς βήμα, ο αλγόριθμος βελτιστοποίησης. Στην παράγραφο αυτή, αξίζει να συμπληρώσουμε ότι το *module* αυτό υλοποιείται από το πακέτο *diploimatiki.mobmgr.iismod*. Η σημαντικότερη κλάση του εν λόγω πακέτου είναι η κλάση *IIS*, ενώ υπάρχουν και οι βοηθητικές κλάσεις *AccessPoint* και *Service*, για τις οποίες έχει γίνει ήδη λόγος στην παράγραφο 4.2.3.

Ο αλγόριθμος βελτιστοποίησης εκκινείται με κλήση της μεθόδου *execute()* ενός αντικειμένου της κλάσης *IIS*. Η μέθοδος αυτή εκτελεί τα βήματα που έχουν περιγραφεί στην παράγραφο 4.2.3. Κάθε τέτοιο βήμα αντιστοιχεί και σε μία ξεχωριστή μέθοδο. Ακολουθεί η αντιστοιχία βημάτων – μεθόδων:

- ✓ Συλλογή πληροφοριών για τα διαθέσιμα *access points* – Μέθοδος *getMeasurements()*.
- ✓ Δημιουργία λίστας διαθέσιμων *access points* – Μέθοδος *convertToAPList()*.
- ✓ Δημιουργία αντικειμένου *Service* – Μέθοδος *constructServiceItem()*.
- ✓ Στάδιο προσδιορισμού του παράγοντα «ποιότητα» – Μέθοδος *findQuality()*.
- ✓ Στάδιο προσδιορισμού του παράγοντα «κόστος» – Μέθοδος *findCost()*.
- ✓ Υπολογισμός αντικειμενικής συνάρτησης – Μέθοδος *objectiveFunction()*.
- ✓ Εύρεση της βέλτιστης λύσης – Μέθοδος *findBest()*.

Φυσικά, υπάρχουν και πολλές βοηθητικές μέθοδοι, όπως, για παράδειγμα, η μέθοδος *bubbleSort()*, η οποία ταξινομεί έναν πίνακα με τη μέθοδο της «φουσαλίδας» (*Bubble Sort*).

## 5.5 Γραφική Διεπαφή

### 5.5.1 Γενική Περιγραφή – Ο Ρόλος της Κλάσης *MenuFrame*

Η γραφική διεπαφή χρήστη περιγράφηκε στο κεφάλαιο 3, στα πλαίσια της παρουσίασης της Συνιστώσας Προτιμήσεων Χρήστη (UP module). Υλοποιείται από το πακέτο *diplomatiki.gui*, το οποίο είναι ένα ευρύ σύνολο κλάσεων. Όπως είδαμε, σκοπός είναι το τελικό αποτέλεσμα να δίνει (να προσομοιώνει) την αίσθηση ενός κινητού τερματικού.

Αν και υπάρχουν πολλές κλάσεις στο πακέτο και όλες τους υλοποιούν κάποιο σημαντικό κομμάτι της γραφικής διεπαφής, η κλάση με τον κεντρικότερο ρόλο είναι η *MenuFrame*. Η κλάση αυτή κατασκευάζει το κεντρικό μενού του GUI (παράγραφος 3.1.3), αλλά, επιπλέον, δίνει την εντολή για την κατασκευή των διαφόρων πάνελ που είδαμε στο κεφάλαιο 3. Πρόκειται για τα εξής πάνελ:

- ✓ Τα πάνελ των 7 υποστηριζόμενων υπηρεσιών (παράγραφος 3.1.2).
- ✓ Το πάνελ επιλογής ενός εκ των 4 διαθέσιμων προφίλ προτιμήσεων (παράγραφος 3.2).
- ✓ Το πάνελ απεικόνισης πληροφοριών σύνδεσης (παράγραφος 3.1.3).

Καθένα από τα πάνελ που μόλις αναφέραμε υλοποιείται από μία ξεχωριστή κλάση. Για παράδειγμα, το πάνελ της υπηρεσίας Voice Call υλοποιείται από την κλάση *VoiceCallPanel*. Όλες, όμως, αυτές οι κλάσεις κληρονομούν από την ίδια κλάση με όνομα *NextPanel*. Η τελευταία συγκεντρώνει όλα τα κοινά χαρακτηριστικά των πάνελ που αναφέρθηκαν, ώστε να μη χρειάζεται να επανακαθορίζονται σε καθένα από αυτά.

Η κλάση *MenuFrame* δίνει, επίσης, την εντολή για να κατασκευαστεί ένα αντικείμενο τύπου *MobilityManager* (βλ. και παράγραφο 5.3.1). Επιπρόσθετα, κατασκευάζει και εκκινεί το thread (νήμα) το οποίο είναι υπεύθυνο για την απεικόνιση της ημερομηνίας και ώρας στην μπάρα τίτλου του παραθύρου της εφαρμογής. Το thread αυτό είναι ένα αντικείμενο της κλάσης *CalendarTitle*.

### 5.5.2 Συμπληρωματικά Στοιχεία

Μια αναλυτικότερη περιγραφή της προτεινόμενης υλοποίησης ξεφεύγει από τους στόχους της παρούσας εργασίας. Εξάλλου, περισσότερες λεπτομέρειες μπορούν να βρεθούν στο παράρτημα της εργασίας, στο οποίο υπάρχει ο κώδικας της εφαρμογής. Θα σταθούμε, ωστόσο, σε ορισμένα ακόμη στοιχεία που αφορούν την υλοποίηση του GUI.

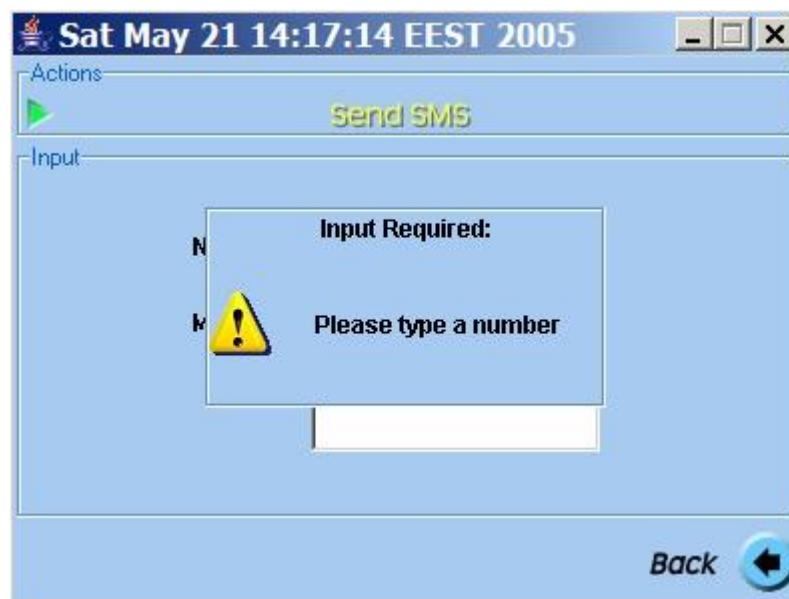
Όπως είδαμε στο κεφάλαιο 3, ο χρήστης έχει τη δυνατότητα να παρακολουθήσει τις πληροφορίες σύνδεσης για κάθε μία τρέχουσα υπηρεσία ξεχωριστά. Το screenshot που ακολουθεί δείχνει τον τρόπο απεικόνισης των πληροφοριών αυτών για την περίπτωση της υπηρεσίας Voice Call.



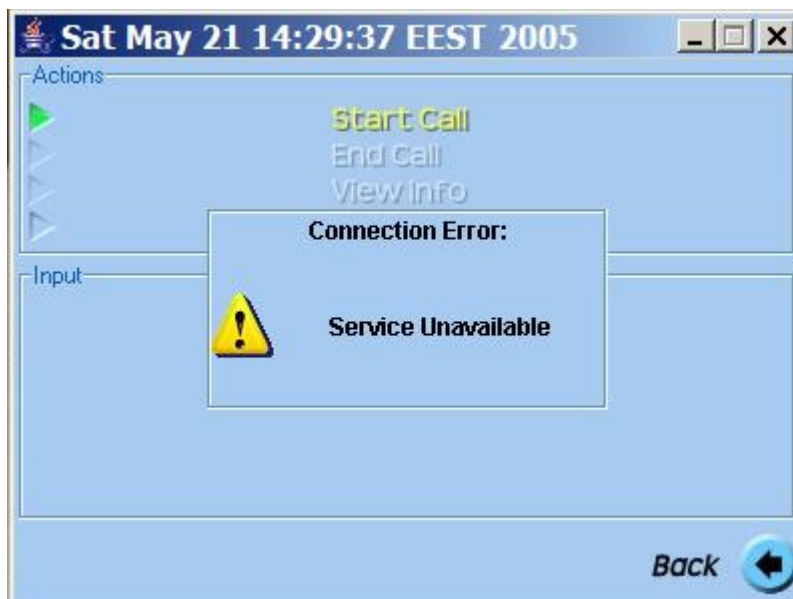
Το παραπάνω πάνελ υλοποιείται με τη βοήθεια των κλάσεων *InfoPanel* και *ConnectionInfoPanel*. Όπως παρατηρούμε, αποτελείται από δύο τμήματα. Το πάνω τμήμα του παρουσιάζει πληροφορίες σχετικά με την υπηρεσία την οποία το πάνελ αφορά. Το κάτω τμήμα του παρουσιάζει πληροφορίες σχετικά με τη σύνδεση μέσω της οποίας εξυπηρετείται η συγκεκριμένη υπηρεσία.

Μία από τις πολλές βοηθητικές κλάσεις του GUI είναι η *MessageWindow*. Η κλάση αυτή χρίζει ειδικής αναφοράς, μιας και βοηθά στην ειδοποίηση του χρήστη με κατάλληλα μηνύματα. Γενικά, τα μηνύματα αυτά είναι δύο ειδών:

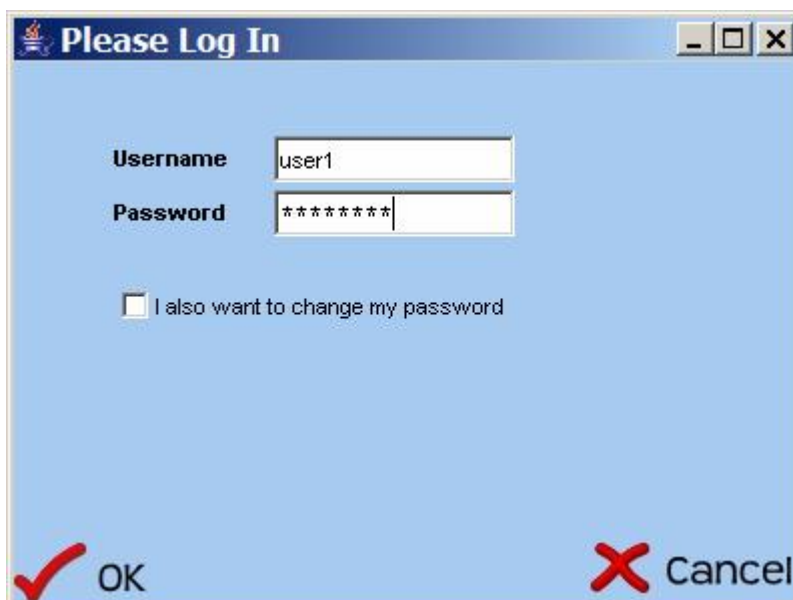
- ✓ Μηνύματα που ειδοποιούν το χρήστη ότι έχει κάνει κάποιο λάθος, π.χ. έχει ξεχάσει να συμπληρώσει τον αριθμό του παραλήπτη ενός SMS μηνύματος, όπως φαίνεται στο ακόλουθο screenshot.

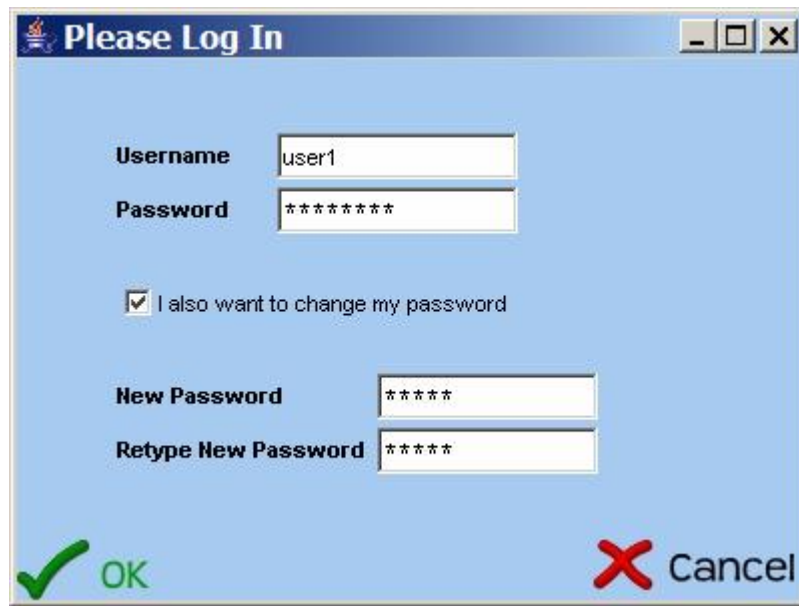


- ✓ Μηνύματα που ειδοποιούν το χρήστη ότι μια αιτούμενη υπηρεσία δεν μπορεί να εξυπηρετηθεί, λόγω της κατάστασης του δικτύου.



Κλείνουμε την παράγραφο αυτή αναφερόμενοι σε ένα ακόμη ειδικό θέμα: τη διαδικασία μέσω της οποίας ο χρήστης κάνει log in στο τερματικό. Για το σκοπό αυτό, κατασκευάστηκε ένα παράθυρο μέσω του οποίου ο χρήστης συμπληρώνει το όνομα (username) και τον κωδικό πρόσβασής του (password). Επίσης, μπορεί, εφόσον το επιθυμεί, να αλλάξει τον κωδικό αυτό, εισάγοντας πρώτα τον παλιό και έπειτα το νέο κωδικό. Τα δύο παρακάτω screenshots εικονίζουν το παράθυρο αυτό. Σημειώνεται ότι η υλοποίηση του παραθύρου, όπως και ο έλεγχος για το αν τα στοιχεία που συμπλήρωσε ο χρήστης είναι σωστά, γίνεται από την κλάση *PINFrame*.

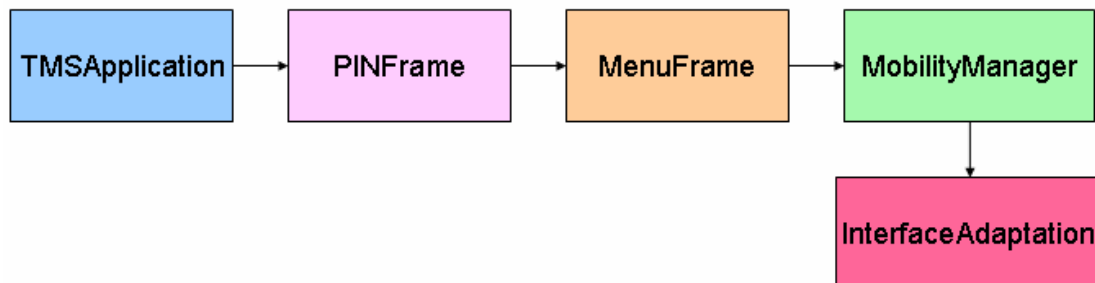




## 5.6 Συμπληρωματικά Σχήματα

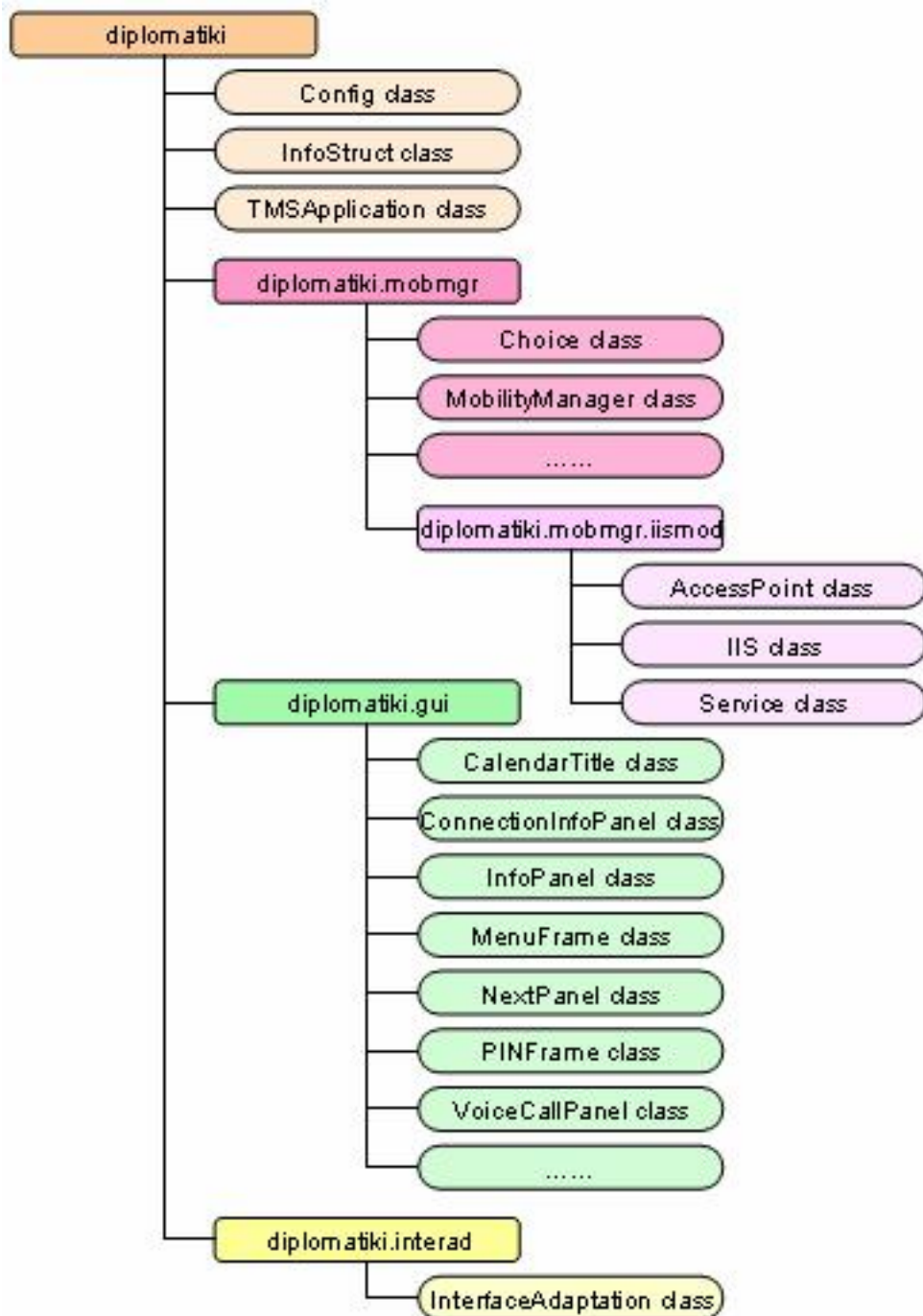
Στο σημείο αυτό, είμαστε πλέον σε θέση να συμπληρώσουμε την περιγραφή της εφαρμογής που υλοποιήθηκε με δύο επιπλέον σχήματα.

Στο πρώτο, παρουσιάζεται η αλληλουχία στην κατασκευή των διαφόρων modules της εφαρμογής, κατά την εκκίνησή της.



Έτσι, λοιπόν, όπως φαίνεται στο παραπάνω σχήμα, όταν δίνεται εντολή για το άνοιγμα της εφαρμογής, η *TMSApplication* κλάση (στη μέθοδο *main()* αυτής) κατασκευάζει ένα αντικείμενο τύπου *PINFrame*. Σε περίπτωση που ο χρήστης συμπληρώσει σωστά το όνομα και τον κωδικό πρόσβασής του, κατασκευάζεται ένα αντικείμενο τύπου *MenuFrame*. Το *MenuFrame* δίνει την εντολή κατασκευής όλων των απαραίτητων πάνελ του GUI, αλλά και ενός αντικειμένου τύπου *MobilityManager*. Τέλος, ο *MobilityManager* περιέχει την εντολή κατασκευής ενός αντικειμένου τύπου *InterfaceAdaptation*, το οποίο αρχίζει να εκτελείται στο δικό του νήμα.

Το δεύτερο σχήμα, που παραθέτουμε αμέσως πιο κάτω, αποτελεί μια πιο αναλυτική μορφή του σχήματος που παρατέθηκε στην εισαγωγή του κεφαλαίου 5 (παράγραφος 5.1). Στο σχήμα αυτό δεν καταγράφονται όλες οι κλάσεις τις οποίες περιλαμβάνει η εφαρμογή, αλλά μόνο εκείνες για τις οποίες έχει γίνει λόγος στα παραπάνω.



Ειδικά η κλάση *Config* του πακέτου *diplomatiki* συγκεντρώνει σταθερές, και μερικές στατικές (static) μεταβλητές, ιδιαίτερα χρήσιμες για την παραμετροποίηση της εφαρμογής (π.χ. καταγράφει τις διαδρομές (paths) προς τα αρχεία και τα εικονίδια που χρησιμοποιεί η εφαρμογή).

## 5.7 Υλοποίηση Υπηρεσιών

Από τις 7 υπηρεσίες που είναι διαθέσιμες στο κινητό τερματικό (βλ. παράγραφο 3.1.2), έχει υλοποιηθεί, ενδεικτικά, η υπηρεσία Email. Η υπηρεσία αυτή υλοποιείται από τις κλάσεις *Emailer* και *EmailerService*, που ανήκουν στο πακέτο *diplomatiki.services*. Η κλάση *Emailer* παρέχει μία μέθοδο για την αποστολή ενός απλού ηλεκτρονικού μηνύματος, ενώ η *EmailerService* αποτελεί υπο-κλάση της πρώτης, με τη διαφορά ότι ένα αντικείμενό της μπορεί να εκτελεστεί σε ξεχωριστό νήμα.

Ένα αντικείμενο τύπου *EmailerService* κατασκευάζεται, με τις κατάλληλες παραμέτρους, και εκκινείται, μόλις ο χρήστης επιλέξει από το GUI την αποστολή κάποιου e-mail.

## 5.8 Οντότητα Διαχείρισης του Δικτύου

Η υλοποίηση του συστήματος διαχείρισης του δικτύου (Network Management System) γίνεται από το πακέτο *networkside.network* και, ειδικότερα, από τις κλάσεις *NetworkEntity* και *NetworkEntityService*. Φυσικά, στόχος της συγκεκριμένης υλοποίησης είναι η προσομοίωση της επικοινωνίας τερματικού και δικτύου (προκειμένου να υποβληθεί προς έγκριση η επιλογή δικτύου πρόσβασης) και όχι η ρεαλιστική και πλήρης κατασκευή μιας οντότητας διαχείρισης του δικτύου.

Ένα αντικείμενο τύπου *NetworkEntity* «ακούει» σε μια συγκεκριμένη θύρα (port) επικοινωνίας, περιμένοντας αιτήσεις από τερματικά. Μόλις λάβει μια τέτοια αίτηση, δημιουργεί ένα αντικείμενο τύπου *NetworkEntityService*, στο οποίο και αναθέτει την εξυπηρέτηση του τερματικού. Το *NetworkEntityService* αντικείμενο εκτελείται σε ξεχωριστό νήμα. Αρχικά, κάνει λήψη της επιλογής του τερματικού σχετικά με το σημείο πρόσβασης και το επίπεδο ποιότητας στο οποίο θα ανατεθεί η εξυπηρέτηση μιας συγκεκριμένης υπηρεσίας. Έπειτα, αποφασίζει, με τυχαίο τρόπο, αν θα κάνει δεκτή ή όχι την επιλογή αυτή και στέλνει πίσω το κατάλληλο μήνυμα. Σε περίπτωση απόρριψης, αναμένει να λάβει τη δεύτερη κατά σειρά επιλογή του τερματικού και ακολουθεί την ίδια διαδικασία. Σε περίπτωση αποδοχής, η επικοινωνία λύεται.

Η πιθανότητα απόρριψης μιας επιλογής ανάθεσης μπορεί να καθοριστεί μέσα από το πρόγραμμα, π.χ. στο 0%, στο 20%, κ.λπ. Βέβαια, στην πράξη, η αποδοχή ή μη της επιλογής των τερματικών θα εξαρτάτο από την εφαρμογή των κριτηρίων του συστήματος διαχείρισης του δικτύου (π.χ. φορτίο του δικτύου, επιχειρηματικά μοντέλα, κ.λπ.).



## 6 Εκτέλεση Προσομοιώσεων

### 6.1 Εισαγωγή

Στο κεφάλαιο αυτό, θα παρουσιαστούν με αρκετές λεπτομέρειες τα αποτελέσματα του αλγορίθμου που προέκυψαν μετά από την εκτέλεση προσομοιώσεων βασισμένων σε δύο τυπικά σενάρια. Η κατασκευή των σεναρίων έγινε με γνώμονα την κάλυψη όσο το δυνατόν περισσότερων περιπτώσεων επιλογής ασύρματου δικτύου πρόσβασης.

Για καθένα από τα δύο σενάρια που θα παρατεθούν στις επόμενες παραγράφους, θα δοθεί, αρχικά, μια πολύ σύντομη περιγραφή του θέματός τους και, εν συνεχεία, θα παρουσιαστεί αναλυτικά, στα πλαίσια της λειτουργίας του αλγορίθμου, κάθε περίπτωση στην οποία εγείρεται το πρόβλημα της επιλογής δικτύου πρόσβασης.

Αξίζει να σημειωθεί ότι το Mobility Manager module και το IIS sub-module καταγράφουν, για διευκόλυνση των προσομοιώσεων, τα αποτελέσματα κάθε εκτέλεσης του αλγορίθμου βελτιστοποίησης σε ένα αρχείο κειμένου (log file). Στο αρχείο αυτό, κάθε φορά που ενεργοποιείται ο αλγόριθμος βελτιστοποίησης, καταγράφονται τα εξής στοιχεία:

- ✓ Το όνομα της αιτούμενης υπηρεσίας.
- ✓ Ο πίνακας των μετρήσεων (συνθήκες δικτύου).
- ✓ Οι προτιμήσεις του χρήστη (μέγιστο επιθυμητό επίπεδο ποιότητας) για τη συγκεκριμένη υπηρεσία, καθώς και οι γενικές ρυθμίσεις του προφίλ.
- ✓ Ένας πίνακας ο οποίος παρουσιάζει όλες τις δυνατές επιλογές ανάθεσης (σημείο πρόσβασης και επίπεδο ποιότητας) και, για κάθε μία από αυτές, τις τιμές των παραμέτρων «ποιότητα», «πραγματική ποιότητα», «κόστος», «πάροχος», καθώς, επίσης, και τις τιμές των αντίστοιχων συντελεστών βαρύτητας. Τέλος, τυπώνεται και η βαθμολογία (δηλαδή, η τιμή της αντικειμενικής συνάρτησης που περιγράφηκε στο κεφάλαιο 4) της κάθε επιλογής.
- ✓ Ένας πίνακας ο οποίος παρουσιάζει ταξινομημένες, σύμφωνα με τη βαθμολογία τους, τις διάφορες επιλογές ανάθεσης.
- ✓ Τέλος, τυπώνονται τα στοιχεία της επιλογής ανάθεσης που θα υποβληθεί προς έγκριση στο δίκτυο, καθώς και η απόφαση του δικτύου (έγκριση ή απόρριψη). Σε περίπτωση απόρριψης, η επόμενη κατά σειρά επιλογή ανάθεσης υποβάλλεται προς έγκριση, κ.ο.κ.

Αξίζει, ακόμη, να διευκρινιστεί και ένα θέμα που αφορά την ακρίβεια των αριθμητικών αποτελεσμάτων στα οποία καταλήγει ο αλγόριθμος βελτιστοποίησης. Όπως μπορεί κανείς να παρατηρήσει από τα στοιχεία του log file, οι τιμές των βαθμολογιών δεν είναι απόλυτα ακριβείς, για παράδειγμα, αντί της αναμενόμενης βαθμολογίας 2.654545, μπορεί να έχουμε το αποτέλεσμα 2.6545454545454548.

Το θέμα αυτό σχετίζεται με τον τρόπο αναπαράστασης των δεκαδικών αριθμών στη Java και δεν οφείλεται σε κάποιο σφάλμα. Χωρίς να υπεισέλθουμε σε περισσότερες λεπτομέρειες, αρκεί να αναφέρουμε ότι πολλοί δεκαδικοί αριθμοί, όπως, για παράδειγμα, ο αριθμός 0.1, δεν έχουν ακριβή αναπαράσταση σε έναν υπολογιστή. Για το λόγο αυτό, μπορεί να παρατηρηθούν ορισμένες διαφορές μεταξύ των αναμενόμενων και των πραγματικών αποτελεσμάτων, σαν αυτή που αναφέρθηκε

παραπάνω. Οι διαφορές αυτές, όμως, είναι πολύ μικρής τάξεως και με κανέναν τρόπο δεν επηρεάζουν τη λήψη σωστής απόφασης από τον αλγόριθμο βελτιστοποίησης. Εξάλλου, ενδιαφερόμαστε για το αν μια βαθμολογία είναι μεγαλύτερη ή μικρότερη από τις υπόλοιπες και όχι για την ακριβή τιμή της (τουλάχιστον όχι με ακρίβεια απεριόριστου αριθμού δεκαδικών ψηφίων). Σε περίπτωση που ο αναγνώστης επιθυμεί να ενημερωθεί για το θέμα περισσότερο, μπορεί να ανατρέξει στην εξής ιστοσελίδα της Java: <http://java.sun.com/developer/JDCTechTips/2001/tt0807.html>

## 6.2 Σενάριο 1 – «Μέσα σε ένα πανεπιστημιακό campus»

### 6.2.1 Γενική Περιγραφή

Το πρώτο σενάριο αφορά τις κινήσεις ενός καθηγητή στο χώρο ενός πανεπιστημιακού campus. Ο καθηγητής πραγματοποιεί διάφορες μετακινήσεις στους χώρους του campus, στους οποίους, ανάλογα με την τοποθεσία, υπάρχει κάλυψη από GSM, UMTS και WLAN. Στην επόμενη παράγραφο, θα εξετάσουμε αναλυτικότερα την κάθε φάση αυτού του σεναρίου.

Σημειώνεται ότι, στα screenshots που θα παρατεθούν στην επόμενη παράγραφο, καταγράφονται σε κάθε τοποθεσία τα σημεία πρόσβασης που την καλύπτουν. Με σκίαση καταγράφονται τα σημεία πρόσβασης το σήμα των οποίων, όπως αυτό λαμβάνεται από τον καθηγητή, είναι ασθενές. Επίσης, οι υπηρεσίες τις οποίες λαμβάνει ο καθηγητής μία δεδομένη στιγμή παρουσιάζονται δίπλα από τα αντίστοιχα σημεία πρόσβασης.

Τέλος, διευκρινίζεται ότι στο σενάριο αυτό η πιθανότητα απόρριψης μιας πρότασης του τερματικού από το σύστημα διαχείρισης του δικτύου έχει καθορισθεί στο 0%.

### 6.2.2 Εξέλιξη Σεναρίου

#### 1. «Μέσα στο εργαστήριο»:

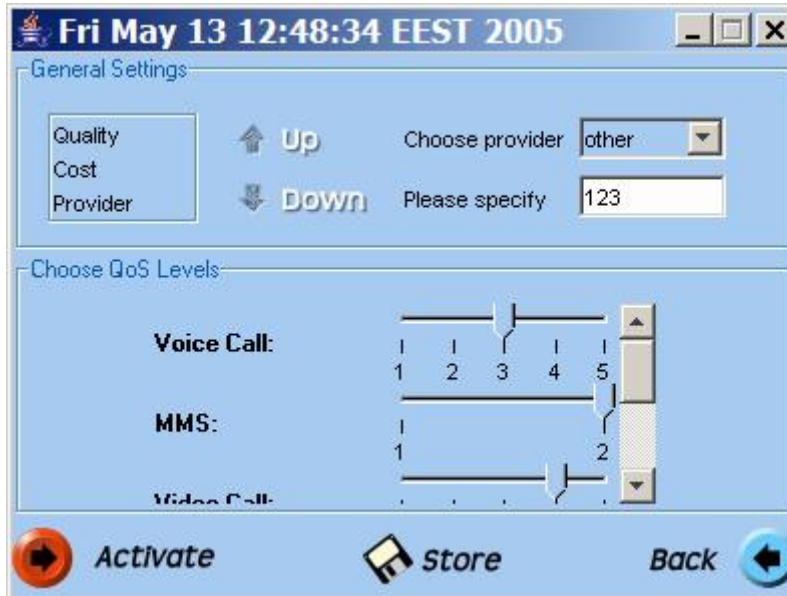
«Ο καθηγητής κος Κεραϊόπουλος εργάζεται εντατικά πάνω στο καινούριο του project, μέσα στο Εργαστήριο Τηλεπικοινωνιών του campus. Στη θέση του παρέχεται κάλυψη τόσο από GSM και UMTS, όσο και από αρκετά WLANs, τα οποία λειτουργούν στην περιοχή του εργαστηρίου. Ο καθηγητής έχει εκκινήσει την υπηρεσία Web Browsing και, μέσω αυτής, πραγματοποιεί τη λήψη (downloading) ενός αρχείου μεγάλου μεγέθους».

#### ▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing

- Προφίλ Προτιμήσεων:

Το προφίλ προτιμήσεων το οποίο έχει ενεργό την παρούσα στιγμή είναι το εξής:



Όπως παρατηρούμε, οι γενικές ρυθμίσεις που έχει καταχωρήσει είναι οι εξής:

- ✓ Ιεράρχηση των παραγόντων «ποιότητα», «κόστος» και «πάροχος»:
  - Ποιότητα
  - Κόστος
  - Πάροχος
- ✓ Προτιμώμενος πάροχος: Έχει δηλώσει ένα δικό του πάροχο και, συγκεκριμένα, τον πάροχο με κωδικό αριθμό «123». Σημειώνεται, πάντως, ότι στο σενάριο αυτό κανείς από τους παρόχους δικτύου που παρέχουν κάλυψη στο campus δεν είναι ο προτιμώμενος.

Όσον αφορά τις ανά υπηρεσία ρυθμίσεις, αυτές φαίνονται στον επόμενο πίνακα:

Όνομα Υπηρεσίας	Μέγιστο Επιθυμητό Επίπεδο Ποιότητας
Voice Call	3
MMS	2
Video Call	4
Video Stream	3
Web Browsing	4

Υπενθυμίζεται ότι οι υπηρεσίες SMS και Email μπορούν να διατεθούν σε ένα μοναδικό επίπεδο ποιότητας.

- Screenshots:

Ακολουθεί μια εποπτική εικόνα στην οποία φαίνονται τα διαθέσιμα σημεία πρόσβασης την τρέχουσα χρονική στιγμή.



- Log File:

Οι λεπτομέρειες παροχής της υπηρεσίας Web Browsing, όπως αυτές προέκυψαν ως αποτέλεσμα του αλγορίθμου βελτιστοποίησης και καταγράφηκαν στο log file, είναι οι εξής:

---

# Fri May 13 12:00:54 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.3.1	4	3	5	2000
147.102.3.2	5	3	5	3000
147.102.3.3	6	3	5	2000
147.102.1.1	1	1	4	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	5	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.3.1	0.8	1	1	54,5455	0,002	0	0,69091
147.102.3.1	0.8	2	2	54,5455	0,006	0	1,27273
147.102.3.1	0.8	3	3	54,5455	0,008	0	1,96364
147.102.3.1	0.8	4	4	54,5455	0,011	0	2,6
147.102.3.2	0.8	1	1	54,5455	0,0023	0	0,67455
147.102.3.2	0.8	2	2	54,5455	0,0071	0	1,21273
147.102.3.2	0.8	3	3	54,5455	0,009	0	1,90909
147.102.3.2	0.8	4	4	54,5455	0,012	0	2,54545
147.102.3.3	0.8	1	1	54,5455	0,0019	0	0,69636
147.102.3.3	0.8	2	2	54,5455	0,0055	0	1,3
147.102.3.3	0.8	3	3	54,5455	0,0078	0	1,97455
147.102.3.3	0.8	4	4	54,5455	0,01	0	2,65455
147.102.1.1	0.8	1	0,95	54,5455	0,008	0	0,32364
147.102.1.1	0.8	2	1,9	54,5455	0,016	0	0,64727
147.102.1.1	0.8	3	2,85	54,5455	0,024	0	0,97091
147.102.1.2	0.8	1	1	54,5455	0,009	0	0,30909
147.102.1.2	0.8	2	2	54,5455	0,015	0	0,78182
147.102.1.2	0.8	3	3	54,5455	0,022	0	1,2
147.102.2.1	0.8	1	1	54,5455	0,01	0	0,25455
147.102.2.1	0.8	2	2	54,5455	0,018	0	0,61818
147.102.2.1	0.8	3	3	54,5455	0,029	0	0,81818
147.102.2.1	0.8	4	4	54,5455	0,039	0	1,07273
147.102.2.2	0.8	1	0,95	54,5455	0,009	0	0,26909

147.102.2.2	0.8	2	1,9	54,5455	0,019	0	0,48364
147.102.2.2	0.8	3	2,85	54,5455	0,028	0	0,75273
147.102.2.2	0.8	4	3,8	54,5455	0,04	0	0,85818

Sorted Array of Choices: (26 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.3	4	2,65455
2.	147.102.3.1	4	2,6
3.	147.102.3.2	4	2,54545
4.	147.102.3.3	3	1,97455
5.	147.102.3.1	3	1,96364
6.	147.102.3.2	3	1,90909
7.	147.102.3.3	2	1,3
8.	147.102.3.1	2	1,27273
9.	147.102.3.2	2	1,21273
10.	147.102.1.2	3	1,2
11.	147.102.2.1	4	1,07273
12.	147.102.1.1	3	0,97091
13.	147.102.2.2	4	0,85818
14.	147.102.2.1	3	0,81818
15.	147.102.1.2	2	0,78182
16.	147.102.2.2	3	0,75273
17.	147.102.3.3	1	0,69636
18.	147.102.3.1	1	0,69091
19.	147.102.3.2	1	0,67455
20.	147.102.1.1	2	0,64727
21.	147.102.2.1	2	0,61818
22.	147.102.2.2	2	0,48364
23.	147.102.1.1	1	0,32364
24.	147.102.1.2	1	0,30909
25.	147.102.2.2	1	0,26909
26.	147.102.2.1	1	0,25455

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3            Grade = 2.6545454545454548

IP Address = 147.102.3.3        Provider Code = 6

QoS Level = 4                    Service Code = 7

The terminal's request has been ACCEPTED!

▪ Αναλυτικό Παράδειγμα:

Μιας και πρόκειται για την πρώτη φάση του σεναρίου, ας εξηγήσουμε λίγο αναλυτικότερα τα αποτελέσματα του αλγορίθμου, όπως αυτά παρουσιάζονται στο log file.

Όπως παρατηρούμε, αρχικά τυπώνεται το όνομα της υπηρεσίας για την οποία τρέχει ο αλγόριθμος. Εν συνεχεία, παρατίθεται ο πίνακας των τρεχουσών μετρήσεων. Όπως

έχει εξηγηθεί, τον πίνακα αυτόν τον παραλαμβάνει ο Mobility Manager (και το IIS sub-module του) από το IA module, υπεύθυνο για τη συλλογή, οργάνωση και έλεγχο των μετρήσεων. Οι μετρήσεις είναι καταχωρημένες σε ένα αρχείο τύπου XML και από αυτό διαβάζονται από το IA module. Έτσι, για παράδειγμα, για αυτήν τη φάση του σεναρίου το αντίστοιχο κομμάτι του XML αρχείου έχει ως εξής:

---

```
<measurement timer="0"><!--LAB-->
  <AP ip="147.102.3.1" providerCode="4" technology="3" signal="5"
    availBandwidth="2000"/>
  <AP ip="147.102.3.2" providerCode="5" technology="3" signal="5"
    availBandwidth="3000"/>
  <AP ip="147.102.3.3" providerCode="6" technology="3" signal="5"
    availBandwidth="2000"/>
  <AP ip="147.102.1.1" providerCode="1" technology="1" signal="4"
    availBandwidth="170"/>
  <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
  <AP ip="147.102.2.1" providerCode="1" technology="2" signal="5"
    availBandwidth="1000"/>
  <AP ip="147.102.2.2" providerCode="3" technology="2" signal="4"
    availBandwidth="1000"/>
</measurement>
```

---

Το IA module, διαβάζοντας το παραπάνω τμήμα του αρχείου (παρατηρούμε ότι πρόκειται για το τμήμα για το οποίο ισχύει: timer = 0, δηλαδή για το πρώτο του τμήμα), κατασκευάζει τον πίνακα τον οποίο βλέπουμε στο log file. Ο πίνακας αυτός αποτελεί μία εκ των εισόδων του αλγορίθμου.

Η έτερη σημαντικότερη είσοδος είναι το προφίλ του χρήστη, οι βασικές ρυθμίσεις του οποίου τυπώνονται, αμέσως μετά, από το IIS sub-module στο log file.

Ακολουθεί ο πίνακας με όλες τις δυνατές επιλογές ανάθεσης (σημείο πρόσβασης και επίπεδο ποιότητας), καθώς και τα χαρακτηριστικά της κάθε μιας. Τα χαρακτηριστικά αυτά είναι ο παράγοντας της ποιότητας (στήλη «QoS Level»), της πραγματικής ποιότητας («Real Quality»), συνδυασμός του ονομαστικού επιπέδου ποιότητας και του επιπέδου του λαμβανόμενου σήματος), του κόστους («Cost») και του παρόχου («Provider Factor»). Για διευκόλυνση, τυπώνονται και οι συντελεστές της ποιότητας («Quality Factor») και του κόστους («Cost Factor»). Η τελευταία στήλη («Grade») αντιστοιχεί στη βαθμολογία και προκύπτει εύκολα από τις υπόλοιπες τιμές του πίνακα ως εξής:

$$Grade = QualityFactor \times RealQuality - CostFactor \times Cost$$

αν η τιμή «Provider Factor» είναι μηδενική. Διαφορετικά, αν είναι μη μηδενική, τότε έχουμε:

$$Grade = QualityFactor \times RealQuality - CostFactor \times Cost + 0.5 \times ProviderFactor \times Abs[QualityFactor \times RealQuality - CostFactor \times Cost]$$



διότι πρέπει να προσθέσουμε και την πρωτοδότηση της ταύτισης του παρόχου του υποψήφιου σημείου πρόσβασης με τον προτιμώμενο από το χρήστη πάροχο, σύμφωνα με όσα έχουν εξηγηθεί στο κεφάλαιο 4.

Ο αμέσως επόμενος πίνακας στο log file περιλαμβάνει συνοπτικά όλες τις επιλογές ανάθεσης (σημείο πρόσβασης και επίπεδο ποιότητας υπηρεσίας), ταξινομημένες, όμως, κατά φθίνουσα σειρά βαθμολογίας.

Στο τελευταίο τμήμα, ο Mobility Manager καταγράφει τα στοιχεία της επιλογής ανάθεσης που πρόκειται να προτείνει στο σύστημα διαχείρισης του δικτύου. Η επιλογή αυτή είναι εκείνη που βρίσκεται στην πρώτη θέση του προαναφερθέντος πίνακα. Έπειτα, τυπώνεται η απόφαση του δικτύου («ACCEPTED» ή «REJECTED»). Σε περίπτωση που έχουμε απόρριψη, προτείνεται στο δίκτυο η δεύτερη κατά σειρά επιλογή ανάθεσης, κοκ.

Όσα αναφέρθηκαν στο σημείο αυτό ισχύουν και για τις υπόλοιπες φάσεις του σεναρίου, όπως, βέβαια, και για το επόμενο σενάριο.

▪ Παρατηρήσεις:

Όπως βλέπουμε από τα παραπάνω, η επιλογή η οποία συγκέντρωσε τη μεγαλύτερη βαθμολογία (τιμή αντικειμενικής συνάρτησης), σύμφωνα με τη διαδικασία που περιγράφηκε στο κεφάλαιο 4, είναι η παροχή της υπηρεσίας από ένα από τα τρία WLANs, από τον πάροχο με κωδικό αριθμό 6 και σε επίπεδο ποιότητας 4. Η επιλογή αυτή, μετά από σχετική αίτηση, έγινε αποδεκτή από το δίκτυο.

Μία λεπτομέρεια στην οποία πρέπει να σταθούμε είναι ότι το μέγιστο επίπεδο ποιότητας που εξετάζεται στη λίστα των επιλογών είναι το επίπεδο 4. Αυτό είναι απόλυτα δικαιολογημένο, καθώς, όπως είδαμε, ο χρήστης μέσα από τη γραφική διεπαφή έχει δηλώσει ότι για την υπηρεσία Web Browsing επιθυμεί να λαμβάνει και να πληρώνει μέγιστο επίπεδο ποιότητας 4.

Επίσης, παρατηρούμε ότι η επιλογή λήψης της υπηρεσίας από σημείο πρόσβασης που υποστηρίζει την τεχνολογία GSM σε επίπεδο ποιότητας 4 δεν εξετάζεται από τον αλγόριθμο, διότι το διαθέσιμο bandwidth του σημείου δεν επαρκεί για το δεδομένο επίπεδο ποιότητας.

2. «Απομακρυνόμενος από το εργαστήριο»:

«Ξαφνικά, ο καθηγητής διαπιστώνει ότι η ώρα πέρασε και θα αργήσει στη διάλεξη που πρόκειται να δώσει. Έντρομος, πετάγεται έξω από το εργαστήριο και κατευθύνεται προς το γραφείο του για να πάρει το CD με τις διαφάνειες της παρουσίασης. Συγχρόνως, στέλνει SMS σε ένα συνάδελφό του για να του θυμίσει τη μεσημεριανή τους συνάντηση. Το downloading του αρχείου (υπηρεσία Web Browsing), το οποίο είχε ξεκινήσει μέσα στο εργαστήριο, συνεχίζεται. Καθώς απομακρύνεται από το εργαστήριο, το σήμα που λαμβάνεται από τα WLANs εξασθενεί».



▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ SMS

▪ Screenshots:





▪ Log File:

---

# Fri May 13 12:01:08 EEST 2005:

# Service SMS has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.3.1	4	3	1	2000
147.102.3.2	5	3	2	3000
147.102.3.3	6	3	1	2000
147.102.1.1	1	1	5	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	4	1000
147.102.2.2	3	2	5	1000

The user's maximum preferred QoS level is QoS-1  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.3.1	0.8	1	0,4	4,6154	0,013	0	0,26
147.102.3.2	0.8	1	0,6	4,6154	0,017	0	0,40154
147.102.3.3	0.8	1	0,4	4,6154	0,014	0	0,25538
147.102.1.1	0.8	1	1	4,6154	0,09	0	0,38462
147.102.1.2	0.8	1	1	4,6154	0,085	0	0,40769
147.102.2.1	0.8	1	0,95	4,6154	0,09	0	0,34462

147.102.2.2	0.8	1	1	4,6154	0,07	0	0,47692
-------------	-----	---	---	--------	------	---	---------

Sorted Array of Choices: (7 choices)

	IP Address	QoS Level	Grade
1.	147.102.2.2	1	0,47692
2.	147.102.1.2	1	0,40769
3.	147.102.3.2	1	0,40154
4.	147.102.1.1	1	0,38462
5.	147.102.2.1	1	0,34462
6.	147.102.3.1	1	0,26
7.	147.102.3.3	1	0,25538

A connection with the following details is being requested from the nearest network entity:

Access Technology = 2      Grade = 0.47692307692307695

IP Address = 147.102.2.2      Provider Code = 3

QoS Level = 1      Service Code = 2

The terminal's request has been ACCEPTED!

▪ Παρατηρήσεις:

Όπως παρατηρούμε, η υπηρεσία Web Browsing συνεχίζεται να λαμβάνεται από το ίδιο σημείο πρόσβασης, αν και η ένταση του σήματος που λαμβάνει ο χρήστης από αυτό είναι σαφώς μειωμένη. Ο αλγόριθμος, σωστά, δεν επανεξετάζει την υπηρεσία αυτή, καθώς η ένταση του σήματος είναι μεν χαμηλή, αλλά επαρκεί για την παροχή της υπηρεσίας και, επομένως, δε χρειάζεται να επιβαρυνθεί το σύστημα με μία επιπλέον διαπομπή.

Αντίθετα, ο αλγόριθμος εκκινείται για την υπηρεσία SMS και καταλήγει στην ανάθεσή της σε ένα σημείο πρόσβασης που υποστηρίζει την τεχνολογία UMTS. Η επιλογή αυτή γίνεται τελικά αποδεκτή από το δίκτυο.

3. «Στη διαδρομή προς το γραφείο»:

«Καθώς ο καθηγητής έχει απομακρυνθεί από το εργαστήριο, η ευρυζωνική σύνδεση που του παρείχαν τα WLANs έχει χαθεί. Εν τω μεταξύ, ο καθηγητής τηλεφωνεί στο βοηθό του (Voice Call), προκειμένου να τον ενημερώσει για το λόγο της μικρής του καθυστέρησης. Έπειτα από λίγο, τερματίζει τη συνομιλία».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Voice Call

▪ Screenshots:



Service Info

 Browsing URL: [www.ntua.gr](http://www.ntua.gr)

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**

Provider: **Vodafone**

QoS Level: **3**

IP Address: **147.102.1.2**

Back 



- Log File:

---

# Fri May 13 12:01:27 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	5	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	4	1000
147.102.2.2	3	2	5	1000

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	1	54,5455	0,008	0	0,36364
147.102.1.1	0.8	2	2	54,5455	0,016	0	0,72727
147.102.1.1	0.8	3	3	54,5455	0,024	0	1,09091
147.102.1.2	0.8	1	1	54,5455	0,009	0	0,30909
147.102.1.2	0.8	2	2	54,5455	0,015	0	0,78182
147.102.1.2	0.8	3	3	54,5455	0,022	0	1,2
147.102.2.1	0.8	1	0,95	54,5455	0,01	0	0,21455
147.102.2.1	0.8	2	1,9	54,5455	0,018	0	0,53818
147.102.2.1	0.8	3	2,85	54,5455	0,029	0	0,69818

147.102.2.1	0.8	4	3,8	54,5455	0,039	0	0,91273
147.102.2.2	0.8	1	1	54,5455	0,009	0	0,30909
147.102.2.2	0.8	2	2	54,5455	0,019	0	0,56364
147.102.2.2	0.8	3	3	54,5455	0,028	0	0,87273
147.102.2.2	0.8	4	4	54,5455	0,04	0	1,01818

Sorted Array of Choices: (14 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.2	3	1,2
2.	147.102.1.1	3	1,09091
3.	147.102.2.2	4	1,01818
4.	147.102.2.1	4	0,91273
5.	147.102.2.2	3	0,87273
6.	147.102.1.2	2	0,78182
7.	147.102.1.1	2	0,72727
8.	147.102.2.1	3	0,69818
9.	147.102.2.2	2	0,56364
10.	147.102.2.1	2	0,53818
11.	147.102.1.1	1	0,36364
12.	147.102.2.2	1	0,30909
13.	147.102.1.2	1	0,30909
14.	147.102.2.1	1	0,21455

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 1.2000000000000004

IP Address = 147.102.1.2      Provider Code = 2

QoS Level = 3      Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:01:29 EEST 2005:

# Service VOICE CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	5	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	4	1000
147.102.2.2	3	2	5	1000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123



The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	1	93,75	0,003	0	0,51875
147.102.1.1	0.8	2	2	93,75	0,0055	0	1,08438
147.102.1.1	0.8	3	3	93,75	0,01	0	1,4625
147.102.1.2	0.8	1	1	93,75	0,0032	0	0,5
147.102.1.2	0.8	2	2	93,75	0,006	0	1,0375
147.102.1.2	0.8	3	3	93,75	0,012	0	1,275
147.102.2.1	0.8	1	0,95	93,75	0,005	0	0,29125
147.102.2.1	0.8	2	1,9	93,75	0,009	0	0,67625
147.102.2.1	0.8	3	2,85	93,75	0,015	0	0,87375
147.102.2.2	0.8	1	1	93,75	0,0047	0	0,35938
147.102.2.2	0.8	2	2	93,75	0,0087	0	0,78438
147.102.2.2	0.8	3	3	93,75	0,011	0	1,36875

Sorted Array of Choices: (12 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.1	3	1,4625
2.	147.102.2.2	3	1,36875
3.	147.102.1.2	3	1,275
4.	147.102.1.1	2	1,08438
5.	147.102.1.2	2	1,0375
6.	147.102.2.1	3	0,87375
7.	147.102.2.2	2	0,78438
8.	147.102.2.1	2	0,67625
9.	147.102.1.1	1	0,51875
10.	147.102.1.2	1	0,5
11.	147.102.2.2	1	0,35938
12.	147.102.2.1	1	0,29125

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 1.4625000000000004

IP Address = 147.102.1.1      Provider Code = 1

QoS Level = 3      Service Code = 1

The terminal's request has been ACCEPTED!

▪ Παρατηρήσεις:

Όπως παρατηρούμε, η «εξαφάνιση» των WLANs έχει ως αποτέλεσμα την εκκίνηση του αλγορίθμου, προκειμένου η υπηρεσία Web Browsing, η οποία λαμβανόταν από ένα από αυτά, να ανατεθεί σε ένα σημείο διαφορετικής τεχνολογίας πρόσβασης. Το αποτέλεσμα που προέκυψε μέσα από τη διαδικασία βελτιστοποίησης, όπως φαίνεται και στο log file, είναι η ανάθεση της υπηρεσίας σε ένα σημείο πρόσβασης GSM και σε επίπεδο ποιότητας 3.

Στο σημείο αυτό, πιθανόν να εγείρεται η απορία γιατί τελικά επελέγη η τεχνολογία GSM, η οποία αδυνατεί να παρέχει επίπεδο ποιότητας 4, αντί της τεχνολογίας UMTS (η οποία έχει αυτήν τη δυνατότητα). Η απάντηση αυτή στηρίζεται στην ουσία του αλγορίθμου IIS, που είναι η διαδικασία της βελτιστοποίησης. Μπορεί, δηλαδή, η τεχνολογία GSM να παρέχει ελαφρώς κατώτερη ποιότητα, έναντι της τεχνολογίας UMTS, ο συνδυασμένος παράγοντας, όμως, «ποιότητα – κόστος» υπερτερεί στην πρώτη περίπτωση. Δηλαδή, το πλεονέκτημα που έχει η επιλογή UMTS δεν επαρκεί, προκειμένου να καλύψει τη μεγάλη τιμή κόστους στην οποία αυτή το προσφέρει. Σημειώνεται, ακόμη, ότι μεταξύ των δύο σημείων πρόσβασης GSM επελέγη εκείνο που προσφέρει την ίδια ποιότητα στο χαμηλότερο κόστος.

Όσον αφορά την υπηρεσία Voice Call, αυτή ανατίθεται στο άλλο σημείο πρόσβασης GSM και σε επίπεδο ποιότητας 3, καθώς, όπως είπαμε και παραπάνω, ο συνδυασμένος παράγοντας «ποιότητα – κόστος» είναι καλύτερος για αυτήν την επιλογή.

#### 4. «Μέσα στο γραφείο»:

«Φτάνοντας στο γραφείο του, ο καθηγητής ψάχνει για τις διαφάνειες της διάλεξης. Καθώς η διαδικασία αυτή του παίρνει αρκετή ώρα, το τερματικό του ξαναελέγχει τα διαθέσιμα access points και διαπιστώνει την ύπαρξη ενός WLAN που καλύπτει το γραφείο του καθηγητή. Έχοντας ψάξει παντού, ο καθηγητής πραγματοποιεί μία βίντεο κλήση (Video Call) στο βοηθό του, προκειμένου να τον ρωτήσει αν γνωρίζει πού βρίσκεται το CD με τις διαφάνειες. Εκείνος του απαντά πως πιθανότατα ο καθηγητής έχει ξεχάσει το CD στο αυτοκίνητό του, που είναι σταθμευμένο στο υπόγειο πάρκινγκ του campus. Η βίντεο κλήση τερματίζεται».

#### ▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Call



▪ Screenshots:



Service Info

 Browsing URL: [www.ntua.gr](http://www.ntua.gr)

Connection Details

The requested service has been provided with the following specifications:

Technology: **WLAN**

Provider: **Hellas On Line**

QoS Level: **4**

IP Address: **147.102.3.11**

Back 



▪ Log File:

---

# Fri May 13 12:01:47 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	4	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	4	1000
147.102.2.2	3	2	5	1000
147.102.3.11	6	3	5	2500

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,95	54,5455	0,008	0	0,32364
147.102.1.1	0.8	2	1,9	54,5455	0,016	0	0,64727
147.102.1.1	0.8	3	2,85	54,5455	0,024	0	0,97091
147.102.1.2	0.8	1	1	54,5455	0,009	0	0,30909
147.102.1.2	0.8	2	2	54,5455	0,015	0	0,78182
147.102.1.2	0.8	3	3	54,5455	0,022	0	1,2
147.102.2.1	0.8	1	0,95	54,5455	0,01	0	0,21455
147.102.2.1	0.8	2	1,9	54,5455	0,018	0	0,53818

147.102.2.1	0.8	3	2,85	54,5455	0,029	0	0,69818
147.102.2.1	0.8	4	3,8	54,5455	0,039	0	0,91273
147.102.2.2	0.8	1	1	54,5455	0,009	0	0,30909
147.102.2.2	0.8	2	2	54,5455	0,019	0	0,56364
147.102.2.2	0.8	3	3	54,5455	0,028	0	0,87273
147.102.2.2	0.8	4	4	54,5455	0,04	0	1,01818
147.102.3.11	0.8	1	1	54,5455	0,0019	0	0,69636
147.102.3.11	0.8	2	2	54,5455	0,0055	0	1,3
147.102.3.11	0.8	3	3	54,5455	0,0078	0	1,97455
147.102.3.11	0.8	4	4	54,5455	0,01	0	2,65455

Sorted Array of Choices: (18 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.11	4	2,65455
2.	147.102.3.11	3	1,97455
3.	147.102.3.11	2	1,3
4.	147.102.1.2	3	1,2
5.	147.102.2.2	4	1,01818
6.	147.102.1.1	3	0,97091
7.	147.102.2.1	4	0,91273
8.	147.102.2.2	3	0,87273
9.	147.102.1.2	2	0,78182
10.	147.102.2.1	3	0,69818
11.	147.102.3.11	1	0,69636
12.	147.102.1.1	2	0,64727
13.	147.102.2.2	2	0,56364
14.	147.102.2.1	2	0,53818
15.	147.102.1.1	1	0,32364
16.	147.102.2.2	1	0,30909
17.	147.102.1.2	1	0,30909
18.	147.102.2.1	1	0,21455

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 2.6545454545454548

IP Address = 147.102.3.11      Provider Code = 6

QoS Level = 4      Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:01:49 EEST 2005:

# Service VIDEO CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	4	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	4	1000

147.102.2.2	3	2	5	1000
147.102.3.11	6	3	5	2500

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,95	2,521	0,2	0	0,2558
147.102.1.1	0.8	2	1,9	2,521	0,4	0	0,5116
147.102.1.1	0.8	3	2,85	2,521	0,6	0	0,76739
147.102.1.2	0.8	1	1	2,521	0,2	0	0,2958
147.102.1.2	0.8	2	2	2,521	0,41	0	0,56639
147.102.1.2	0.8	3	3	2,521	0,63	0	0,81176
147.102.2.1	0.8	1	0,95	2,521	0,25	0	0,12975
147.102.2.1	0.8	2	1,9	2,521	0,46	0	0,36034
147.102.2.1	0.8	3	2,85	2,521	0,66	0	0,61613
147.102.2.1	0.8	4	3,8	2,521	0,92	0	0,72067
147.102.2.2	0.8	1	1	2,521	0,24	0	0,19496
147.102.2.2	0.8	2	2	2,521	0,46	0	0,44034
147.102.2.2	0.8	3	3	2,521	0,67	0	0,71092
147.102.2.2	0.8	4	4	2,521	0,92	0	0,88067
147.102.3.11	0.8	1	1	2,521	0,16	0	0,39664
147.102.3.11	0.8	2	2	2,521	0,29	0	0,86891
147.102.3.11	0.8	3	3	2,521	0,45	0	1,26555
147.102.3.11	0.8	4	4	2,521	0,62	0	1,63697

Sorted Array of Choices: (18 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.11	4	1,63697
2.	147.102.3.11	3	1,26555
3.	147.102.2.2	4	0,88067
4.	147.102.3.11	2	0,86891
5.	147.102.1.2	3	0,81176
6.	147.102.1.1	3	0,76739
7.	147.102.2.1	4	0,72067
8.	147.102.2.2	3	0,71092
9.	147.102.2.1	3	0,61613
10.	147.102.1.2	2	0,56639
11.	147.102.1.1	2	0,5116
12.	147.102.2.2	2	0,44034
13.	147.102.3.11	1	0,39664
14.	147.102.2.1	2	0,36034
15.	147.102.1.2	1	0,2958

16.	147.102.1.1	1	0,2558
17.	147.102.2.2	1	0,19496
18.	147.102.2.1	1	0,12975

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 1.6369747899159666  
 IP Address = 147.102.3.11      Provider Code = 6  
 QoS Level = 4      Service Code = 5  
 The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Στο σημείο αυτό, η εμφάνιση ενός καινούριου WLAN access point έχει ως αποτέλεσμα την επανεξέταση όλων των τρεχουσών υπηρεσιών, προκειμένου να διαπιστωθεί αν το νέο σημείο πρόσβασης παρέχει βελτιωμένες συνθήκες σύνδεσης. Έτσι, λοιπόν, η υπηρεσία Web Browsing επανεξετάζεται και αποφασίζεται η ανάθεσή της στο WLAN.

Όσον αφορά την αιτούμενη υπηρεσία Video Call, και αυτή μετά από τη διαδικασία βελτιστοποίησης ανατίθεται στο ίδιο access point. Σημειωτέον ότι τα σημεία πρόσβασης που υποστηρίζουν την τεχνολογία GSM, λόγω του περιορισμένου εύρους ζώνης τους, δεν μπορούν να προσφέρουν τη συγκεκριμένη υπηρεσία σε επίπεδο ποιότητας 4.

5. «Απομακρυνόμενος από γραφείο»:

«Δυστυχώς οι διαφάνειες δε βρέθηκαν στο γραφείο και, έτσι, ο καθηγητής αναγκάζεται να κατευθυνθεί στο υπόγειο πάρκινγκ, προκειμένου να πάρει το χαρτοφύλακά του από το αυτοκίνητο. Βγαίνοντας και πάλι έξω από το γραφείο, η κάλυψη που είχε από το WLAN σημείο χάνεται. Ο καθηγητής στέλνει στο βοηθό του ένα MMS, για να τον ειδοποιήσει ότι κατευθύνεται προς το υπόγειο πάρκινγκ και ότι η καθυστέρηση θα είναι λίγο μεγαλύτερη».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ MMS

▪ Screenshots:



Service Info

 Browsing URL: [www.ntua.gr](http://www.ntua.gr)

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**

Provider: **Vodafone**

QoS Level: **3**

IP Address: **147.102.1.2**

Back 





▪ Log File:

---

# Fri May 13 12:02:08 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	5	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	5	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	1	54,5455	0,008	0	0,36364
147.102.1.1	0.8	2	2	54,5455	0,016	0	0,72727
147.102.1.1	0.8	3	3	54,5455	0,024	0	1,09091
147.102.1.2	0.8	1	1	54,5455	0,009	0	0,30909
147.102.1.2	0.8	2	2	54,5455	0,015	0	0,78182
147.102.1.2	0.8	3	3	54,5455	0,022	0	1,2
147.102.2.1	0.8	1	1	54,5455	0,01	0	0,25455
147.102.2.1	0.8	2	2	54,5455	0,018	0	0,61818
147.102.2.1	0.8	3	3	54,5455	0,029	0	0,81818

147.102.2.1	0.8	4	4	54,5455	0,039	0	1,07273
147.102.2.2	0.8	1	0,95	54,5455	0,009	0	0,26909
147.102.2.2	0.8	2	1,9	54,5455	0,019	0	0,48364
147.102.2.2	0.8	3	2,85	54,5455	0,028	0	0,75273
147.102.2.2	0.8	4	3,8	54,5455	0,04	0	0,85818

Sorted Array of Choices: (14 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.2	3	1,2
2.	147.102.1.1	3	1,09091
3.	147.102.2.1	4	1,07273
4.	147.102.2.2	4	0,85818
5.	147.102.2.1	3	0,81818
6.	147.102.1.2	2	0,78182
7.	147.102.2.2	3	0,75273
8.	147.102.1.1	2	0,72727
9.	147.102.2.1	2	0,61818
10.	147.102.2.2	2	0,48364
11.	147.102.1.1	1	0,36364
12.	147.102.1.2	1	0,30909
13.	147.102.2.2	1	0,26909
14.	147.102.2.1	1	0,25455

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 1.2000000000000004

IP Address = 147.102.1.2      Provider Code = 2

QoS Level = 3      Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:02:09 EEST 2005:

# Service MMS has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	5	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	5	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-2  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123



The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	1	2,3529	0,25	0	0,21176
147.102.1.1	0.8	2	2	2,3529	0,4	0	0,65882
147.102.1.2	0.8	1	1	2,3529	0,24	0	0,23529
147.102.1.2	0.8	2	2	2,3529	0,41	0	0,63529
147.102.2.1	0.8	1	1	2,3529	0,26	0	0,18824
147.102.2.1	0.8	2	2	2,3529	0,4	0	0,65882
147.102.2.2	0.8	1	0,95	2,3529	0,2	0	0,28941
147.102.2.2	0.8	2	1,9	2,3529	0,51	0	0,32

Sorted Array of Choices: (8 choices)

	IP Address	QoS Level	Grade
1.	147.102.2.1	2	0,65882
2.	147.102.1.1	2	0,65882
3.	147.102.1.2	2	0,63529
4.	147.102.2.2	2	0,32
5.	147.102.2.2	1	0,28941
6.	147.102.1.2	1	0,23529
7.	147.102.1.1	1	0,21176
8.	147.102.2.1	1	0,18824

A connection with the following details is being requested from the nearest network entity:

Access Technology = 2      Grade = 0.6588235294117647

IP Address = 147.102.2.1      Provider Code = 1

QoS Level = 2      Service Code = 3

The terminal's request has been ACCEPTED!

▪ Παρατηρήσεις:

Η κάλυψη από το WLAN access point έχει πια χαθεί και, συνεπώς, η υπηρεσία Web Browsing πρέπει να ανατεθεί σε νέο σημείο πρόσβασης. Τα διαθέσιμα σημεία τη δεδομένη στιγμή είναι 4 και υποστηρίζουν τις τεχνολογίες GSM και UMTS. Τελικά, για τους λόγους που αναφέρθηκαν στην περίπτωση 3 («Στη διαδρομή προς το γραφείο»), αποφασίζεται η ανάθεσή της σε access point της τεχνολογίας GSM και σε επίπεδο ποιότητας 3.

Η αιτούμενη υπηρεσία MMS εξυπηρετείται, όπως φαίνεται από το log file, από access point της τεχνολογίας UMTS στο μέγιστο δυνατό επίπεδο ποιότητας.

6. «Ανάμεσα στα δέντρα»:

«Καθώς διασχίζει το πάρκο που οδηγεί στο υπόγειο πάρκινγκ, τα πυκνά φυλλώματα των δέντρων προκαλούν έντονες διαλείψεις στο σήμα του. Μάλιστα, η κάλυψη που

είχε από το access point από το οποίο λάμβανε την υπηρεσία Web Browsing χάνεται εντελώς. Προκειμένου να μην αφήσει ούτε ένα λεπτό να πάει χαμένο, ξεκινά την υπηρεσία Video Stream, μέσω της οποίας λαμβάνει και παρακολουθεί την ομιλία ενός έγκριτου συναδέλφου του σε πανεπιστήμιο του εξωτερικού».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Stream

▪ Screenshots:



Service Info



Browsing URL: [www.ntua.gr](http://www.ntua.gr)

---

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**  
 Provider: **Cosmote**  
 QoS Level: **1**  
 IP Address: **147.102.1.1**

Back 

Service Info



Video Stream from: [www.mit.edu/speech](http://www.mit.edu/speech)

---

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**  
 Provider: **Cosmote**  
 QoS Level: **1**  
 IP Address: **147.102.1.1**

Back 

- Log File:

---

# Fri May 13 12:02:28 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	1	170
147.102.2.1	1	2	1	1000
147.102.2.2	3	2	1	1000

The user's maximum preferred QoS level is QoS-4 and the user's general preferences are:

1. Quality
2. Cost

3. Provider  
and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,4	54,5455	0,008	0	-0,11636
147.102.1.1	0.8	2	0,8	54,5455	0,016	0	-0,23273
147.102.1.1	0.8	3	1,2	54,5455	0,024	0	-0,34909
147.102.2.1	0.8	1	0,4	54,5455	0,01	0	-0,22545
147.102.2.1	0.8	2	0,8	54,5455	0,018	0	-0,34182
147.102.2.1	0.8	3	1,2	54,5455	0,029	0	-0,62182
147.102.2.1	0.8	4	1,6	54,5455	0,039	0	-0,84727
147.102.2.2	0.8	1	0,4	54,5455	0,009	0	-0,17091
147.102.2.2	0.8	2	0,8	54,5455	0,019	0	-0,39636
147.102.2.2	0.8	3	1,2	54,5455	0,028	0	-0,56727
147.102.2.2	0.8	4	1,6	54,5455	0,04	0	-0,90182

Sorted Array of Choices: (11 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.1	1	-0,11636
2.	147.102.2.2	1	-0,17091
3.	147.102.2.1	1	-0,22545
4.	147.102.1.1	2	-0,23273
5.	147.102.2.1	2	-0,34182
6.	147.102.1.1	3	-0,34909
7.	147.102.2.2	2	-0,39636
8.	147.102.2.2	3	-0,56727
9.	147.102.2.1	3	-0,62182
10.	147.102.2.1	4	-0,84727
11.	147.102.2.2	4	-0,90182

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = -0.11636363636363634

IP Address = 147.102.1.1      Provider Code = 1

QoS Level = 1      Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:02:29 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	1	170
147.102.2.1	1	2	1	1000
147.102.2.2	3	2	1	1000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,4	3,1579	0,15	0	-0,15368
147.102.1.1	0.8	2	0,8	3,1579	0,3	0	-0,30737
147.102.2.1	0.8	1	0,4	3,1579	0,17	0	-0,21684
147.102.2.1	0.8	2	0,8	3,1579	0,33	0	-0,40211
147.102.2.1	0.8	3	1,2	3,1579	0,48	0	-0,55579
147.102.2.2	0.8	1	0,4	3,1579	0,16	0	-0,18526
147.102.2.2	0.8	2	0,8	3,1579	0,35	0	-0,46526
147.102.2.2	0.8	3	1,2	3,1579	0,47	0	-0,52421

Sorted Array of Choices: (8 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.1	1	-0,15368
2.	147.102.2.2	1	-0,18526
3.	147.102.2.1	1	-0,21684
4.	147.102.1.1	2	-0,30737
5.	147.102.2.1	2	-0,40211
6.	147.102.2.2	2	-0,46526
7.	147.102.2.2	3	-0,52421
8.	147.102.2.1	3	-0,55579

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = -0.1536842105263157

IP Address = 147.102.1.1      Provider Code = 1

QoS Level = 1      Service Code = 6

The terminal's request has been ACCEPTED!

▪ Παρατηρήσεις:

Τη συγκεκριμένη στιγμή, η κάλυψη από το GSM access point το οποίο εξυπηρετούσε την υπηρεσία Web Browsing έχει χαθεί. Επομένως, επανεκκινείται ο αλγόριθμος IIS για αυτήν την υπηρεσία και καταλήγει σε ανάθεσή της σε άλλο GSM access point και σε επίπεδο ποιότητας 1.

Η νέα υπηρεσία Video Stream ανατίθεται και αυτή στο ίδιο access point και σε επίπεδο ποιότητας 1, καθώς αυτό προσφέρει τον καλύτερο συνδυασμό ποιότητας – κόστους.

7. «Μέσα στο υπόγειο πάρκινγκ»:

«Επιτέλους, ο καθηγητής βρίσκεται μέσα στο υπόγειο πάρκινγκ και κατευθύνεται προς το αυτοκίνητό του. Στο χώρο αυτό, το σήμα που λαμβάνει από τα διάφορα σημεία πρόσβασης είναι επίσης πολύ ασθενές, ενώ έχει χαθεί και πλήρως η κάλυψη από ένα ακόμη σημείο πρόσβασης. Ευτυχώς, ο χαρτοφύλακός του είναι μέσα στο αυτοκίνητο, οπότε ο καθηγητής τον παίρνει και κατευθύνεται προς την έξοδο. Φυσικά, δεν παραλείπει να τηλεφωνήσει στο βοηθό του (Voice Call), για να τον ενημερώσει για την εύρεση του πολυπόθητου χαρτοφύλακα. Το downloading του αρχείου και η παρακολούθηση της ομιλίας συνεχίζονται, ενώ μετά από λίγο η φωνητική κλήση τερματίζεται».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Stream
- ✓ Voice Call

▪ Screenshots:





Service Info

 Browsing URL: [www.ntua.gr](http://www.ntua.gr)

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**  
Provider: **Cosmote**  
QoS Level: **1**  
IP Address: **147.102.1.1**

Back 

Service Info

 Video Stream from: [www.mit.edu/speech](http://www.mit.edu/speech)

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**  
Provider: **Cosmote**  
QoS Level: **1**  
IP Address: **147.102.1.1**

Back 

Service Info

 Call to: **6973811111**

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**  
Provider: **Cosmote**  
QoS Level: **2**  
IP Address: **147.102.1.1**

Back 

▪ Log File:

---

# Fri May 13 12:02:49 EEST 2005:

# Service VOICE CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	1	170
147.102.2.1	1	2	1	1000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,4	93,75	0,003	0	0,03875
147.102.1.1	0.8	2	0,8	93,75	0,0055	0	0,12438
147.102.1.1	0.8	3	1,2	93,75	0,01	0	0,0225
147.102.2.1	0.8	1	0,4	93,75	0,005	0	-0,14875
147.102.2.1	0.8	2	0,8	93,75	0,009	0	-0,20375
147.102.2.1	0.8	3	1,2	93,75	0,015	0	-0,44625

Sorted Array of Choices: (6 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.1	2	0,12438
2.	147.102.1.1	1	0,03875
3.	147.102.1.1	3	0,0225
4.	147.102.2.1	1	-0,14875
5.	147.102.2.1	2	-0,20375
6.	147.102.2.1	3	-0,44625

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.12437500000000012

IP Address = 147.102.1.1      Provider Code = 1

QoS Level = 2      Service Code = 1

The terminal's request has been ACCEPTED!

---



▪ Παρατηρήσεις:

Ένα ακόμη access point έχει «εξαφανισθεί». Καθώς, όμως, οι τρέχουσες υπηρεσίες Web Browsing και Video Stream δεν εξυπηρετούντο από αυτό το access point, δεν υπάρχει λόγος επανεξέτασής τους και συνεχίζουν να παρέχονται από το ίδιο σημείο πρόσβασης.

Ο αλγόριθμος βελτιστοποίησης εκκινείται μόνο για τη νέα αιτούμενη υπηρεσία Voice Call και αποφασίζει την ανάθεσή της στο ίδιο access point από το οποίο εξυπηρετούνται και οι δύο τρέχουσες υπηρεσίες. Η απόφασή του γίνεται αποδεκτή και από την οντότητα διαχείρισης του δικτύου.

8. «Κατευθυνόμενος προς το αμφιθέατρο»:

«Καθώς ο καθηγητής εξέρχεται από το υπόγειο πάρκινγκ, το σήμα βελτιώνεται και τα 4 γνωστά σημεία πρόσβασης είναι διαθέσιμα και πάλι. Με γοργό βηματισμό, κατευθύνεται προς το αμφιθέατρο. Η αγωνία του για τυχόν αποχώρηση των φοιτητών τον αναγκάζει να πραγματοποιήσει ακόμη μία κλήση στο βοηθό του, ο οποίος βρίσκεται ήδη εκεί. Το downloading του αρχείου και η παρακολούθηση της ομιλίας συνεχίζονται. Διαπιστώνοντας ότι πρέπει να ολοκληρώσει το downloading του αρχείου γρηγορότερα, προβαίνει σε μία αλλαγή στο ενεργό προφίλ του και, συγκεκριμένα, στην επιλογή του επιπέδου 5 ως ανώτερου επιπέδου ποιότητας για την υπηρεσία Web Browsing».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Stream
- ✓ Voice Call

▪ Screenshots:



Service Info

 Browsing URL: [www.ntua.gr](http://www.ntua.gr)

Connection Details

The requested service has been provided with the following specifications:

Technology: **GSM**

Provider: **Vodafone**

QoS Level: **3**

IP Address: **147.102.1.2**

Back 

Service Info

 Video Stream from: **www.mit.edu/speech**

Connection Details

The requested service has been provided with the following specifications:

Technology: **UMTS**  
 Provider: **TIM**  
 QoS Level: **3**  
 IP Address: **147.102.2.2**

Back 

Service Info

 Call to: **6973811111**

Connection Details

The requested service has been provided with the following specifications:

Technology: **UMTS**  
 Provider: **TIM**  
 QoS Level: **3**  
 IP Address: **147.102.2.2**

Back 

▪ Log File:

---

# Fri May 13 12:03:08 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	3	170
147.102.1.2	2	1	4	170
147.102.2.1	1	2	2	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-4 and the user's general preferences are:

1. Quality
2. Cost

3. Provider  
and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,8	54,5455	0,008	0	0,20364
147.102.1.1	0.8	2	1,6	54,5455	0,016	0	0,40727
147.102.1.1	0.8	3	2,4	54,5455	0,024	0	0,61091
147.102.1.2	0.8	1	0,95	54,5455	0,009	0	0,26909
147.102.1.2	0.8	2	1,9	54,5455	0,015	0	0,70182
147.102.1.2	0.8	3	2,85	54,5455	0,022	0	1,08
147.102.2.1	0.8	1	0,6	54,5455	0,01	0	-0,06545
147.102.2.1	0.8	2	1,2	54,5455	0,018	0	-0,02182
147.102.2.1	0.8	3	1,8	54,5455	0,029	0	-0,14182
147.102.2.1	0.8	4	2,4	54,5455	0,039	0	-0,20727
147.102.2.2	0.8	1	0,95	54,5455	0,009	0	0,26909
147.102.2.2	0.8	2	1,9	54,5455	0,019	0	0,48364
147.102.2.2	0.8	3	2,85	54,5455	0,028	0	0,75273
147.102.2.2	0.8	4	3,8	54,5455	0,04	0	0,85818

Sorted Array of Choices: (14 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.2	3	1,08
2.	147.102.2.2	4	0,85818
3.	147.102.2.2	3	0,75273
4.	147.102.1.2	2	0,70182
5.	147.102.1.1	3	0,61091
6.	147.102.2.2	2	0,48364
7.	147.102.1.1	2	0,40727
8.	147.102.2.2	1	0,26909
9.	147.102.1.2	1	0,26909
10.	147.102.1.1	1	0,20364
11.	147.102.2.1	2	-0,02182
12.	147.102.2.1	1	-0,06545
13.	147.102.2.1	3	-0,14182
14.	147.102.2.1	4	-0,20727

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 1.0799999999999998

IP Address = 147.102.1.2      Provider Code = 2

QoS Level = 3      Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:03:08 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	3	170
147.102.1.2	2	1	4	170
147.102.2.1	1	2	2	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,8	3,1579	0,15	0	0,16632
147.102.1.1	0.8	2	1,6	3,1579	0,3	0	0,33263
147.102.1.2	0.8	1	0,95	3,1579	0,14	0	0,31789
147.102.1.2	0.8	2	1,9	3,1579	0,32	0	0,50947
147.102.2.1	0.8	1	0,6	3,1579	0,17	0	-0,05684
147.102.2.1	0.8	2	1,2	3,1579	0,33	0	-0,08211
147.102.2.1	0.8	3	1,8	3,1579	0,48	0	-0,07579
147.102.2.2	0.8	1	0,95	3,1579	0,16	0	0,25474
147.102.2.2	0.8	2	1,9	3,1579	0,35	0	0,41474
147.102.2.2	0.8	3	2,85	3,1579	0,47	0	0,79579

Sorted Array of Choices: (10 choices)

	IP Address	QoS Level	Grade
1.	147.102.2.2	3	0,79579
2.	147.102.1.2	2	0,50947
3.	147.102.2.2	2	0,41474
4.	147.102.1.1	2	0,33263
5.	147.102.1.2	1	0,31789
6.	147.102.2.2	1	0,25474
7.	147.102.1.1	1	0,16632
8.	147.102.2.1	1	-0,05684
9.	147.102.2.1	3	-0,07579
10.	147.102.2.1	2	-0,08211

A connection with the following details is being requested from the nearest network entity:

Access Technology = 2      Grade = 0.7957894736842104

IP Address = 147.102.2.2      Provider Code = 3

QoS Level = 3      Service Code = 6

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:03:09 EEST 2005:

# Service VOICE CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	3	170
147.102.1.2	2	1	4	170
147.102.2.1	1	2	2	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,8	93,75	0,003	0	0,35875
147.102.1.1	0.8	2	1,6	93,75	0,0055	0	0,76438
147.102.1.1	0.8	3	2,4	93,75	0,01	0	0,9825
147.102.1.2	0.8	1	0,95	93,75	0,0032	0	0,46
147.102.1.2	0.8	2	1,9	93,75	0,006	0	0,9575
147.102.1.2	0.8	3	2,85	93,75	0,012	0	1,155
147.102.2.1	0.8	1	0,6	93,75	0,005	0	0,01125
147.102.2.1	0.8	2	1,2	93,75	0,009	0	0,11625
147.102.2.1	0.8	3	1,8	93,75	0,015	0	0,03375
147.102.2.2	0.8	1	0,95	93,75	0,0047	0	0,31937
147.102.2.2	0.8	2	1,9	93,75	0,0087	0	0,70438
147.102.2.2	0.8	3	2,85	93,75	0,011	0	1,24875

Sorted Array of Choices: (12 choices)

	IP Address	QoS Level	Grade
1.	147.102.2.2	3	1,24875
2.	147.102.1.2	3	1,155
3.	147.102.1.1	3	0,9825
4.	147.102.1.2	2	0,9575
5.	147.102.1.1	2	0,76438
6.	147.102.2.2	2	0,70438
7.	147.102.1.2	1	0,46
8.	147.102.1.1	1	0,35875
9.	147.102.2.2	1	0,31937
10.	147.102.2.1	2	0,11625
11.	147.102.2.1	3	0,03375
12.	147.102.2.1	1	0,01125

A connection with the following details is being requested from the nearest network entity:

Access Technology = 2      Grade = 1.2487499999999998

IP Address = 147.102.2.2      Provider Code = 3

QoS Level = 3      Service Code = 1

The terminal's request has been ACCEPTED!

-----  
 # Fri May 13 12:03:08 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	3	170
147.102.1.2	2	1	4	170
147.102.2.1	1	2	2	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-4  
 and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,8	54,5455	0,008	0	0,20364
147.102.1.1	0.8	2	1,6	54,5455	0,016	0	0,40727
147.102.1.1	0.8	3	2,4	54,5455	0,024	0	0,61091
147.102.1.2	0.8	1	0,95	54,5455	0,009	0	0,26909
147.102.1.2	0.8	2	1,9	54,5455	0,015	0	0,70182
147.102.1.2	0.8	3	2,85	54,5455	0,022	0	1,08
147.102.2.1	0.8	1	0,6	54,5455	0,01	0	-0,06545
147.102.2.1	0.8	2	1,2	54,5455	0,018	0	-0,02182
147.102.2.1	0.8	3	1,8	54,5455	0,029	0	-0,14182
147.102.2.1	0.8	4	2,4	54,5455	0,039	0	-0,20727
147.102.2.2	0.8	1	0,95	54,5455	0,009	0	0,26909
147.102.2.2	0.8	2	1,9	54,5455	0,019	0	0,48364
147.102.2.2	0.8	3	2,85	54,5455	0,028	0	0,75273
147.102.2.2	0.8	4	3,8	54,5455	0,04	0	0,85818

Sorted Array of Choices: (14 choices)

- |    | IP Address  | QoS Level | Grade |
|----|-------------|-----------|-------|
| 1. | 147.102.1.2 | 3         | 1,08  |

2.	147.102.2.2	4	0,85818
3.	147.102.2.2	3	0,75273
4.	147.102.1.2	2	0,70182
5.	147.102.1.1	3	0,61091
6.	147.102.2.2	2	0,48364
7.	147.102.1.1	2	0,40727
8.	147.102.2.2	1	0,26909
9.	147.102.1.2	1	0,26909
10.	147.102.1.1	1	0,20364
11.	147.102.2.1	2	-0,02182
12.	147.102.2.1	1	-0,06545
13.	147.102.2.1	3	-0,14182
14.	147.102.2.1	4	-0,20727

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 1.0799999999999998  
IP Address = 147.102.1.2      Provider Code = 2  
QoS Level = 3      Service Code = 7  
The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:03:08 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	3	170
147.102.1.2	2	1	4	170
147.102.2.1	1	2	2	1000
147.102.2.2	3	2	4	1000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,8	3,1579	0,15	0	0,16632
147.102.1.1	0.8	2	1,6	3,1579	0,3	0	0,33263
147.102.1.2	0.8	1	0,95	3,1579	0,14	0	0,31789
147.102.1.2	0.8	2	1,9	3,1579	0,32	0	0,50947
147.102.2.1	0.8	1	0,6	3,1579	0,17	0	-0,05684
147.102.2.1	0.8	2	1,2	3,1579	0,33	0	-0,08211



147.102.2.1	0.8	3	1,8	3,1579	0,48	0	-0,07579
147.102.2.2	0.8	1	0,95	3,1579	0,16	0	0,25474
147.102.2.2	0.8	2	1,9	3,1579	0,35	0	0,41474
147.102.2.2	0.8	3	2,85	3,1579	0,47	0	0,79579

Sorted Array of Choices: (10 choices)

	IP Address	QoS Level	Grade
1.	147.102.2.2	3	0,79579
2.	147.102.1.2	2	0,50947
3.	147.102.2.2	2	0,41474
4.	147.102.1.1	2	0,33263
5.	147.102.1.2	1	0,31789
6.	147.102.2.2	1	0,25474
7.	147.102.1.1	1	0,16632
8.	147.102.2.1	1	-0,05684
9.	147.102.2.1	3	-0,07579
10.	147.102.2.1	2	-0,08211

A connection with the following details is being requested from the nearest network entity:

Access Technology = 2      Grade = 0.7957894736842104

IP Address = 147.102.2.2      Provider Code = 3

QoS Level = 3      Service Code = 6

The terminal's request has been ACCEPTED!

▪ Παρατηρήσεις:

Στο σημείο αυτό, γίνονται και πάλι διαθέσιμα τα 4 γνωστά σημεία πρόσβασης, επομένως ο αλγόριθμος επανεξετάζει το σύνολο των τρεχουσών υπηρεσιών, δηλαδή τις υπηρεσίες Web Browsing και Video Stream. Για τη μεν πρώτη, αποφασίζει την ανάθεσή της σε δίκτυο GSM και σε επίπεδο ποιότητας 3. Για τη δεύτερη, αποφασίζει την ανάθεσή της σε δίκτυο UMTS και σε επίπεδο ποιότητας 3.

Πλέον αυτών των δύο, ο αλγόριθμος τρέχει και για τη νέα αιτούμενη υπηρεσία Voice Call, την εξυπηρέτηση της οποίας αποφασίζει να αναθέσει σε δίκτυο UMTS και σε επίπεδο ποιότητας 3.

Μετά από τον τερματισμό του Voice Call, ο καθηγητής αποφασίζει να τροποποιήσει το προφίλ του. Για το λόγο αυτό, ο αλγόριθμος IIS ξανατρέχει για τις υπηρεσίες Web Browsing και Video Stream. Ωστόσο, καταλήγει, σωστά, στις ίδιες αποφάσεις. Αυτό συμβαίνει, επειδή για την υπηρεσία Video Stream δεν έχει αλλάξει κάποια ρύθμιση, ενώ για την υπηρεσία Web Browsing είδαμε ότι επελέγη ως μέγιστο επιθυμητό το επίπεδο ποιότητας 5. Ωστόσο, το επίπεδο αυτό δεν μπορεί να υποστηριχθεί ούτε από τα GSM ούτε από τα UMTS access points, λόγω του μικρότερου διαθέσιμου εύρους ζώνης τους.

9. «Μέσα στο αμφιθέατρο»:

«Με την είσοδό του στο αμφιθέατρο, ο καθηγητής, με ανακούφιση ή απαγοήτευση, διαπιστώνει ότι οι φοιτητές δεν έχουν αποχωρήσει ακόμη λόγω της καθυστέρησης! Ευτυχώς, πάντως, μέσα στο αμφιθέατρο υπάρχει και πάλι κάλυψη WLAN. Η ομιλία του καθηγητή, την οποία λάμβανε μέσω Video Streaming, έχει σχεδόν ολοκληρωθεί και γι' αυτό στέλνει ένα σύντομο email με εύσημα και απορίες στο συνάδελφό του. Το downloading του αρχείου δεν το σταματά, καθώς δεν έχει ολοκληρωθεί ακόμα».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Stream
- ✓ Email

▪ Screenshots:



Service Info

 Browsing URL: [www.ntua.gr](http://www.ntua.gr)

Connection Details

The requested service has been provided with the following specifications:

Technology: **WLAN**  
Provider: **Forthnet**  
QoS Level: **5**  
IP Address: **147.102.3.21**

Back 

Service Info

 Video Stream from: [www.mit.edu/speech](http://www.mit.edu/speech)

Connection Details

The requested service has been provided with the following specifications:

Technology: **WLAN**  
Provider: **Forthnet**  
QoS Level: **3**  
IP Address: **147.102.3.21**

Back 

Service Info

 Sending e-mail to: [prof@mit.edu](mailto:prof@mit.edu)

Connection Details

The requested service has been provided with the following specifications:

Technology: **WLAN**  
Provider: **Forthnet**  
QoS Level: **1**  
IP Address: **147.102.3.21**

100%

Back 

▪ Log File:

---

# Fri May 13 12:03:28 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	2	170
147.102.1.2	2	1	3	170
147.102.2.1	1	2	3	1000
147.102.2.2	3	2	3	1000
147.102.3.21	5	3	5	3000

The user's maximum preferred QoS level is QoS-5  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,6	54,5455	0,008	0	0,04364
147.102.1.1	0.8	2	1,2	54,5455	0,016	0	0,08727
147.102.1.1	0.8	3	1,8	54,5455	0,024	0	0,13091
147.102.1.2	0.8	1	0,8	54,5455	0,009	0	0,14909
147.102.1.2	0.8	2	1,6	54,5455	0,015	0	0,46182
147.102.1.2	0.8	3	2,4	54,5455	0,022	0	0,72
147.102.2.1	0.8	1	0,8	54,5455	0,01	0	0,09455
147.102.2.1	0.8	2	1,6	54,5455	0,018	0	0,29818
147.102.2.1	0.8	3	2,4	54,5455	0,029	0	0,33818
147.102.2.1	0.8	4	3,2	54,5455	0,039	0	0,43273
147.102.2.2	0.8	1	0,8	54,5455	0,009	0	0,14909
147.102.2.2	0.8	2	1,6	54,5455	0,019	0	0,24364
147.102.2.2	0.8	3	2,4	54,5455	0,028	0	0,39273
147.102.2.2	0.8	4	3,2	54,5455	0,04	0	0,37818
147.102.3.21	0.8	1	1	54,5455	0,0023	0	0,67455
147.102.3.21	0.8	2	2	54,5455	0,0071	0	1,21273
147.102.3.21	0.8	3	3	54,5455	0,009	0	1,90909
147.102.3.21	0.8	4	4	54,5455	0,012	0	2,54545
147.102.3.21	0.8	5	5	54,5455	0,015	0	3,18182

Sorted Array of Choices: (19 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.21	5	3,18182
2.	147.102.3.21	4	2,54545

3.	147.102.3.21	3	1,90909
4.	147.102.3.21	2	1,21273
5.	147.102.1.2	3	0,72
6.	147.102.3.21	1	0,67455
7.	147.102.1.2	2	0,46182
8.	147.102.2.1	4	0,43273
9.	147.102.2.2	3	0,39273
10.	147.102.2.2	4	0,37818
11.	147.102.2.1	3	0,33818
12.	147.102.2.1	2	0,29818
13.	147.102.2.2	2	0,24364
14.	147.102.2.2	1	0,14909
15.	147.102.1.2	1	0,14909
16.	147.102.1.1	3	0,13091
17.	147.102.2.1	1	0,09455
18.	147.102.1.1	2	0,08727
19.	147.102.1.1	1	0,04364

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 3.1818181818181817  
IP Address = 147.102.3.21    Provider Code = 5  
QoS Level = 5                Service Code = 7  
The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:03:28 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	2	170
147.102.1.2	2	1	3	170
147.102.2.1	1	2	3	1000
147.102.2.2	3	2	3	1000
147.102.3.21	5	3	5	3000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,6	3,1579	0,15	0	0,00632

147.102.1.1	0.8	2	1,2	3,1579	0,3	0	0,01263
147.102.1.2	0.8	1	0,8	3,1579	0,14	0	0,19789
147.102.1.2	0.8	2	1,6	3,1579	0,32	0	0,26947
147.102.2.1	0.8	1	0,8	3,1579	0,17	0	0,10316
147.102.2.1	0.8	2	1,6	3,1579	0,33	0	0,23789
147.102.2.1	0.8	3	2,4	3,1579	0,48	0	0,40421
147.102.2.2	0.8	1	0,8	3,1579	0,16	0	0,13474
147.102.2.2	0.8	2	1,6	3,1579	0,35	0	0,17474
147.102.2.2	0.8	3	2,4	3,1579	0,47	0	0,43579
147.102.3.21	0.8	1	1	3,1579	0,14	0	0,35789
147.102.3.21	0.8	2	2	3,1579	0,23	0	0,87368
147.102.3.21	0.8	3	3	3,1579	0,37	0	1,23158

Sorted Array of Choices: (13 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.21	3	1,23158
2.	147.102.3.21	2	0,87368
3.	147.102.2.2	3	0,43579
4.	147.102.2.1	3	0,40421
5.	147.102.3.21	1	0,35789
6.	147.102.1.2	2	0,26947
7.	147.102.2.1	2	0,23789
8.	147.102.1.2	1	0,19789
9.	147.102.2.2	2	0,17474
10.	147.102.2.2	1	0,13474
11.	147.102.2.1	1	0,10316
12.	147.102.1.1	2	0,01263
13.	147.102.1.1	1	0,00632

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 1.2315789473684213

IP Address = 147.102.3.21      Provider Code = 5

QoS Level = 3      Service Code = 6

The terminal's request has been ACCEPTED!

-----  
# Fri May 13 12:03:30 EEST 2005:

# Service E-MAIL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	2	170
147.102.1.2	2	1	3	170
147.102.2.1	1	2	3	1000
147.102.2.2	3	2	3	1000
147.102.3.21	5	3	5	3000

The user's maximum preferred QoS level is QoS-1  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 123

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.8	1	0,6	0,7692	0,58	0	0,03385
147.102.1.2	0.8	1	0,8	0,7692	0,59	0	0,18615
147.102.2.1	0.8	1	0,8	0,7692	0,59	0	0,18615
147.102.2.2	0.8	1	0,8	0,7692	0,59	0	0,18615
147.102.3.21	0.8	1	1	0,7692	0,0046	0	0,79646

Sorted Array of Choices: (5 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.21	1	0,79646
2.	147.102.2.2	1	0,18615
3.	147.102.2.1	1	0,18615
4.	147.102.1.2	1	0,18615
5.	147.102.1.1	1	0,03385

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 0.7964615384615386

IP Address = 147.102.3.21      Provider Code = 5

QoS Level = 1      Service Code = 4

The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:


Στο σημείο αυτό, γίνεται αντιληπτή από το τερματικό η εμφάνιση ενός νέου σημείου πρόσβασης WLAN, γεγονός που οδηγεί στην επανεξέταση των τρεχουσών υπηρεσιών. Το Web Browsing ανατίθεται στο νέο αυτό σημείο πρόσβασης και σε επίπεδο ποιότητας 5, το οποίο μπορεί πλέον να εξυπηρετηθεί. Το Video Stream ανατίθεται επίσης στο ίδιο access point και στο μέγιστο επιτρεπτό επίπεδο ποιότητας, δηλαδή στο επίπεδο 3.

Η νέα υπηρεσία Email, λόγω του σαφώς μικρότερου κόστους, ανατίθεται και αυτή στο ίδιο σημείο πρόσβασης.

## 6.3 Σενάριο 2 – «Από το σπίτι στη δουλειά»

### 6.3.1 Γενική Περιγραφή

Το δεύτερο σενάριο αφορά έναν εργαζόμενο ο οποίος ξεκινά το πρωί από το σπίτι του για να πάει στη δουλειά του. Το γραφείο του είναι μακριά και, προκειμένου να φτάσει εκεί, χρειάζεται να χρησιμοποιήσει διάφορα μεταφορικά μέσα. Κατά τη διάρκεια της μετακίνησής του, χρησιμοποιεί διάφορες υπηρεσίες από το 4G τερματικό του, τις οποίες και λαμβάνει κάθε φορά από το βέλτιστο δυνατό δίκτυο (GSM, UMTS, WLAN ή DVB) και στο καταλληλότερο επίπεδο ποιότητας. Στην επόμενη παράγραφο, θα εξετάσουμε αναλυτικότερα την κάθε φάση αυτού του σεναρίου.

Σημειώνεται ότι, στα screenshots που θα παρατεθούν στην επόμενη παράγραφο, καταγράφονται σε κάθε τοποθεσία τα σημεία πρόσβασης που την καλύπτουν. Με σκίαση καταγράφονται τα σημεία πρόσβασης το σήμα των οποίων, όπως αυτό λαμβάνεται από τον εργαζόμενο, είναι ασθενές. Επίσης, οι υπηρεσίες τις οποίες λαμβάνει ο εργαζόμενος μία δεδομένη στιγμή παρουσιάζονται δίπλα από τα αντίστοιχα σημεία πρόσβασης, ενώ με τη χαρακτηριστική εικόνα  πλαισιώνονται τα access points που ανήκουν στον προτιμώμενο από το χρήστη πάροχο.

Τέλος, διευκρινίζεται ότι στο σενάριο αυτό η πιθανότητα απόρριψης μιας πρότασης του τερματικού από το σύστημα διαχείρισης του δικτύου έχει καθορισθεί στο 20%.

### 6.3.2 Εξέλιξη Σεναρίου

#### 1. «Μέσα στο σπίτι»:

«Είναι πρωί και ο εργαζόμενος κος Εργατικόπουλος βρίσκεται σπίτι του και ετοιμάζεται να φύγει για τη δουλειά του. Παρακολουθεί ζωντανά τις τελευταίες εξελίξεις στο Χρηματιστήριο, από την οθόνη του τερματικού του, μέσω της υπηρεσίας Video Stream. Στο χώρο του σπιτιού του, υπάρχει κάλυψη τόσο από GSM και UMTS δίκτυα, όσο και από DVB».

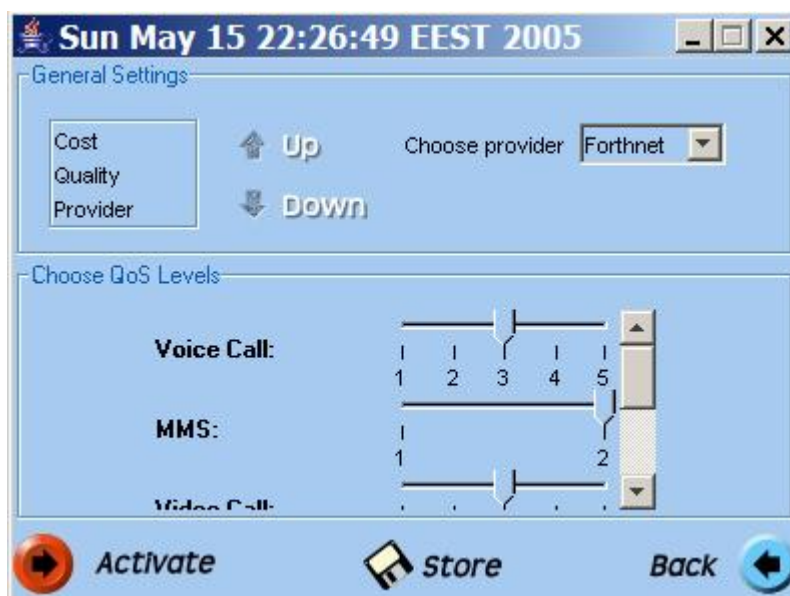
#### ▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Video Stream

#### ▪ Προφίλ Προτιμήσεων:

Το προφίλ προτιμήσεων το οποίο έχει ενεργό την παρούσα στιγμή είναι το εξής:





Όπως παρατηρούμε, οι γενικές ρυθμίσεις που έχει καταχωρήσει είναι οι εξής:

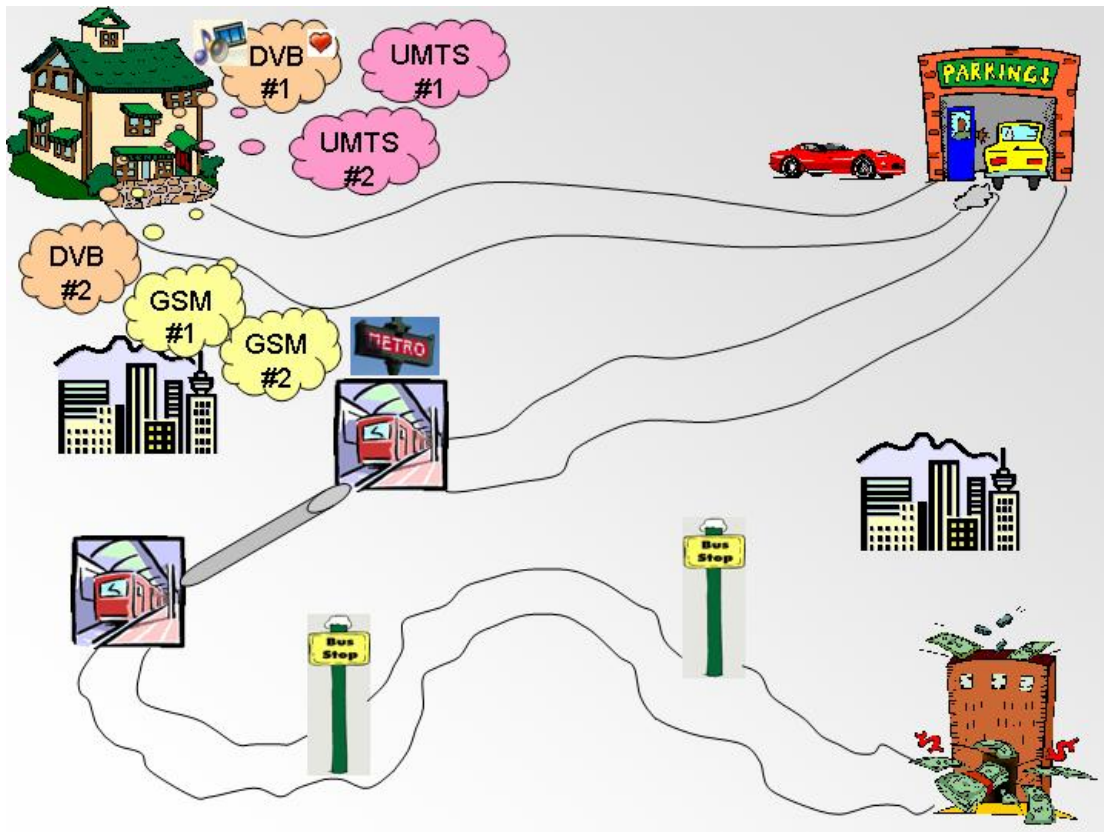
- ✓ Ιεράρχηση των παραγόντων «ποιότητα», «κόστος» και «πάροχος»:
  - i. Κόστος
  - ii. Ποιότητα
  - iii. Πάροχος
- ✓ Προτιμώμενος πάροχος: Ως προτιμώμενο έχει διαλέξει τον πάροχο «Forthnet», ο κωδικός αριθμός του οποίου είναι ο «5».

Όσον αφορά τις ανά υπηρεσία ρυθμίσεις, αυτές φαίνονται στον επόμενο πίνακα:

Όνομα Υπηρεσίας	Μέγιστο Επιθυμητό Επίπεδο Ποιότητας
Voice Call	3
MMS	2
Video Call	3
Video Stream	4
Web Browsing	5

Υπενθυμίζεται ότι οι υπηρεσίες SMS και Email μπορούν να διατεθούν σε ένα μοναδικό επίπεδο ποιότητας.

▪ Screenshots:



▪ Log File:

---

# Sat May 14 13:58:41 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	5	170
147.102.1.2	2	1	5	170
147.102.2.1	1	2	5	2000
147.102.2.2	3	2	5	2000
147.102.4.1	5	4	5	4000
147.102.4.2	6	4	5	4000

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Cost
  2. Quality
  3. Provider
- and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.6	1	1	4,2105	0,15	0	-0,03158
147.102.1.1	0.6	2	2	4,2105	0,3	0	-0,06316
147.102.1.2	0.6	1	1	4,2105	0,14	0	0,01053
147.102.1.2	0.6	2	2	4,2105	0,32	0	-0,14737
147.102.2.1	0.6	1	1	4,2105	0,17	0	-0,11579
147.102.2.1	0.6	2	2	4,2105	0,33	0	-0,18947
147.102.2.1	0.6	3	3	4,2105	0,48	0	-0,22105
147.102.2.1	0.6	4	4	4,2105	0,64	0	-0,29474
147.102.2.2	0.6	1	1	4,2105	0,16	0	-0,07368
147.102.2.2	0.6	2	2	4,2105	0,35	0	-0,27368
147.102.2.2	0.6	3	3	4,2105	0,47	0	-0,17895
147.102.2.2	0.6	4	4	4,2105	0,65	0	-0,33684
147.102.4.1	0.6	1	1	4,2105	0,0092	0,4	0,67352
147.102.4.1	0.6	2	2	4,2105	0,19	0,4	0,48
147.102.4.1	0.6	3	3	4,2105	0,35	0,4	0,39158
147.102.4.1	0.6	4	4	4,2105	0,53	0,4	0,20211
147.102.4.2	0.6	1	1	4,2105	0,0089	0	0,56253
147.102.4.2	0.6	2	2	4,2105	0,2	0	0,35789
147.102.4.2	0.6	3	3	4,2105	0,35	0	0,32632
147.102.4.2	0.6	4	4	4,2105	0,51	0	0,25263

Sorted Array of Choices: (20 choices)

	IP Address	QoS Level	Grade
1.	147.102.4.1	1	0,67352
2.	147.102.4.2	1	0,56253
3.	147.102.4.1	2	0,48
4.	147.102.4.1	3	0,39158
5.	147.102.4.2	2	0,35789
6.	147.102.4.2	3	0,32632
7.	147.102.4.2	4	0,25263
8.	147.102.4.1	4	0,20211
9.	147.102.1.2	1	0,01053
10.	147.102.1.1	1	-0,03158
11.	147.102.1.1	2	-0,06316
12.	147.102.2.2	1	-0,07368
13.	147.102.2.1	1	-0,11579
14.	147.102.1.2	2	-0,14737
15.	147.102.2.2	3	-0,17895
16.	147.102.2.1	2	-0,18947
17.	147.102.2.1	3	-0,22105
18.	147.102.2.2	2	-0,27368
19.	147.102.2.1	4	-0,29474
20.	147.102.2.2	4	-0,33684

A connection with the following details is being requested from the nearest network entity:

Access Technology = 4      Grade = 0.6735157894736842

IP Address = 147.102.4.1      Provider Code = 5  
QoS Level = 1                      Service Code = 6  
The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Η διαδικασία βελτιστοποίησης καταλήγει στην ανάθεση της υπηρεσίας Video Stream σε σημείο πρόσβασης τεχνολογίας DVB και, μάλιστα, σε εκείνο το σημείο το οποίο ανήκει στον πάροχο που προτιμά ο χρήστης. Η επιλογή αυτή, η οποία και γίνεται αποδεκτή από το δίκτυο, παρέχει την αιτούμενη υπηρεσία σε επίπεδο ποιότητας 1. Η απόφαση αυτή δικαιολογείται ως εξής: Ο συγκεκριμένος χρήστης, στο τρέχον προφίλ προτιμήσεών του, έχει θέσει ως πρωταρχική παράμετρο το κόστος. Έτσι, είναι φυσικό ο αλγόριθμος να ψάξει για την οικονομικότερη λύση. Το στοιχείο αυτό, σε συνδυασμό με την επιλογή του χρήστη όσον αφορά τον πάροχο, οδηγεί στη συγκεκριμένη τελική ανάθεση.

Επίσης, παρατηρούμε ότι τα σημεία πρόσβασης της τεχνολογίας GSM δεν μπορούν να παρέχουν την υπηρεσία σε επίπεδο ποιότητας ανώτερο του 2, λόγω του περιορισμένου διαθέσιμου bandwidth τους, γι' αυτό και η επιλογή αυτή δεν εξετάζεται καθόλου από τον αλγόριθμο βελτιστοποίησης.

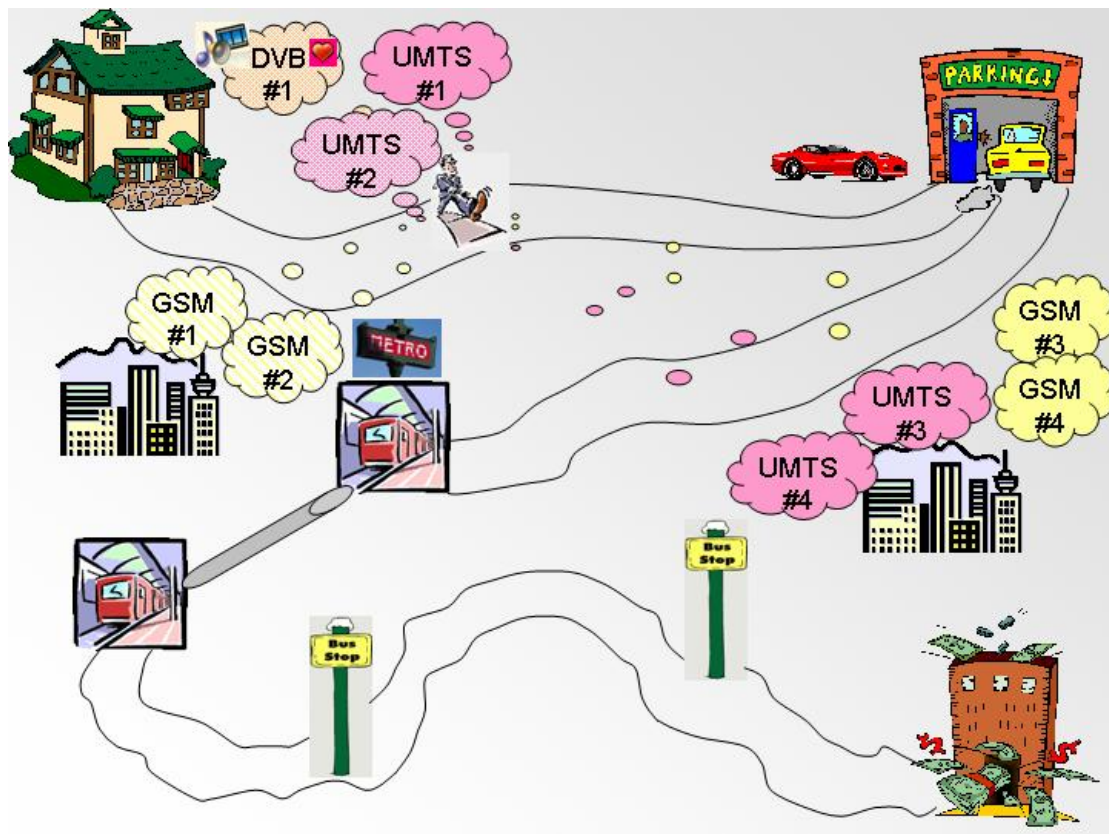
2. «Έξω από το σπίτι»:

«Η ώρα περνάει και ο εργαζόμενος πρέπει να ξεκινήσει για τη δουλειά του. Βγαίνει έξω από το σπίτι του και κατευθύνεται προς το πάρκινγκ όπου έχει σταθμεύσει το αυτοκίνητο του. Το τερματικό του λαμβάνει πιο αδύναμο σήμα από τα access points που «έβλεπε» μέχρι στιγμής, ενώ ταυτόχρονα ορισμένα καινούρια σημεία πρόσβασης κάνουν την εμφάνισή τους. Καθώς προχωρά στο δρόμο, συνεχίζει να παρακολουθεί τα τελευταία νέα, μέσω του Video Streaming».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Video Stream

▪ Screenshots:



▪ Log File:

---

# Sat May 14 13:58:54 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.1	1	1	2	170
147.102.1.2	2	1	2	170
147.102.2.1	1	2	3	2000
147.102.2.2	3	2	2	2000
147.102.4.1	5	4	1	4000
147.102.1.11	1	1	3	170
147.102.1.12	2	1	3	170
147.102.2.11	1	2	3	2000
147.102.2.12	3	2	3	2000

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Cost
2. Quality
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.1	0.6	1	0,6	4,2105	0,15	0	-0,27158
147.102.1.1	0.6	2	1,2	4,2105	0,3	0	-0,54316
147.102.1.2	0.6	1	0,6	4,2105	0,14	0	-0,22947
147.102.1.2	0.6	2	1,2	4,2105	0,32	0	-0,62737
147.102.2.1	0.6	1	0,8	4,2105	0,17	0	-0,23579
147.102.2.1	0.6	2	1,6	4,2105	0,33	0	-0,42947
147.102.2.1	0.6	3	2,4	4,2105	0,48	0	-0,58105
147.102.2.1	0.6	4	3,2	4,2105	0,64	0	-0,77474
147.102.2.2	0.6	1	0,6	4,2105	0,16	0	-0,31368
147.102.2.2	0.6	2	1,2	4,2105	0,35	0	-0,75368
147.102.2.2	0.6	3	1,8	4,2105	0,47	0	-0,89895
147.102.2.2	0.6	4	2,4	4,2105	0,65	0	-1,29684
147.102.4.1	0.6	1	0,4	4,2105	0,0092	0,4	0,24152
147.102.4.1	0.6	2	0,8	4,2105	0,19	0,4	-0,256
147.102.4.1	0.6	3	1,2	4,2105	0,35	0,4	-0,60295
147.102.4.1	0.6	4	1,6	4,2105	0,53	0,4	-1,01726
147.102.1.11	0.6	1	0,8	4,2105	0,15	0	-0,15158
147.102.1.11	0.6	2	1,6	4,2105	0,3	0	-0,30316
147.102.1.12	0.6	1	0,8	4,2105	0,14	0	-0,10947
147.102.1.12	0.6	2	1,6	4,2105	0,32	0	-0,38737
147.102.2.11	0.6	1	0,8	4,2105	0,17	0	-0,23579
147.102.2.11	0.6	2	1,6	4,2105	0,33	0	-0,42947
147.102.2.11	0.6	3	2,4	4,2105	0,48	0	-0,58105
147.102.2.11	0.6	4	3,2	4,2105	0,64	0	-0,77474
147.102.2.12	0.6	1	0,8	4,2105	0,16	0	-0,19368
147.102.2.12	0.6	2	1,6	4,2105	0,35	0	-0,51368
147.102.2.12	0.6	3	2,4	4,2105	0,47	0	-0,53895
147.102.2.12	0.6	4	3,2	4,2105	0,65	0	-0,81684

Sorted Array of Choices: (28 choices)

	IP Address	QoS Level	Grade
1.	147.102.4.1	1	0,24152
2.	147.102.1.12	1	-0,10947
3.	147.102.1.11	1	-0,15158
4.	147.102.2.12	1	-0,19368
5.	147.102.1.2	1	-0,22947
6.	147.102.2.11	1	-0,23579
7.	147.102.2.1	1	-0,23579
8.	147.102.4.1	2	-0,256
9.	147.102.1.1	1	-0,27158
10.	147.102.1.11	2	-0,30316
11.	147.102.2.2	1	-0,31368
12.	147.102.1.12	2	-0,38737
13.	147.102.2.11	2	-0,42947

14.	147.102.2.1	2	-0,42947
15.	147.102.2.12	2	-0,51368
16.	147.102.2.12	3	-0,53895
17.	147.102.1.1	2	-0,54316
18.	147.102.2.11	3	-0,58105
19.	147.102.2.1	3	-0,58105
20.	147.102.4.1	3	-0,60295
21.	147.102.1.2	2	-0,62737
22.	147.102.2.2	2	-0,75368
23.	147.102.2.11	4	-0,77474
24.	147.102.2.1	4	-0,77474
25.	147.102.2.12	4	-0,81684
26.	147.102.2.2	3	-0,89895
27.	147.102.4.1	4	-1,01726
28.	147.102.2.2	4	-1,29684

A connection with the following details is being requested from the nearest network entity:

Access Technology = 4      Grade = 0.2415157894736842  
IP Address = 147.102.4.1      Provider Code = 5  
QoS Level = 1      Service Code = 6  
The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Στο σημείο αυτό, έχουμε την εμφάνιση 4 νέων σημείων πρόσβασης, οπότε ο αλγόριθμος επανεξετάζει την υπηρεσία Video Stream, προκειμένου να διαπιστώσει αν αυτή μπορεί να εξυπηρετηθεί καλύτερα από κάποιο από τα νέα σημεία πρόσβασης. Όπως παρατηρούμε, όμως, στο log file, ο συνδυασμός access point και επιπέδου ποιότητας που ίσχυε στην προηγούμενη ανάθεση επικρατεί και πάλι και, συνεπώς, δεν πραγματοποιείται διαπομπή. Ο λόγος που συμβαίνει αυτό, παρόλο που το λαμβανόμενο σήμα από το συγκεκριμένο σημείο πρόσβασης έχει εξασθενήσει, εστιάζεται και πάλι στον παράγοντα «κόστος», αφού η υπηρεσία Video Stream παρέχεται με χαμηλότερο κόστος μέσω της τεχνολογίας DVB, σε σύγκριση με τις άλλες τεχνολογίες πρόσβασης. Επιπλέον, κανένα από τα καινούρια access points δεν ανήκει στον προτιμώμενο πάροχο.

3. «Μέσα στο αυτοκίνητο»:

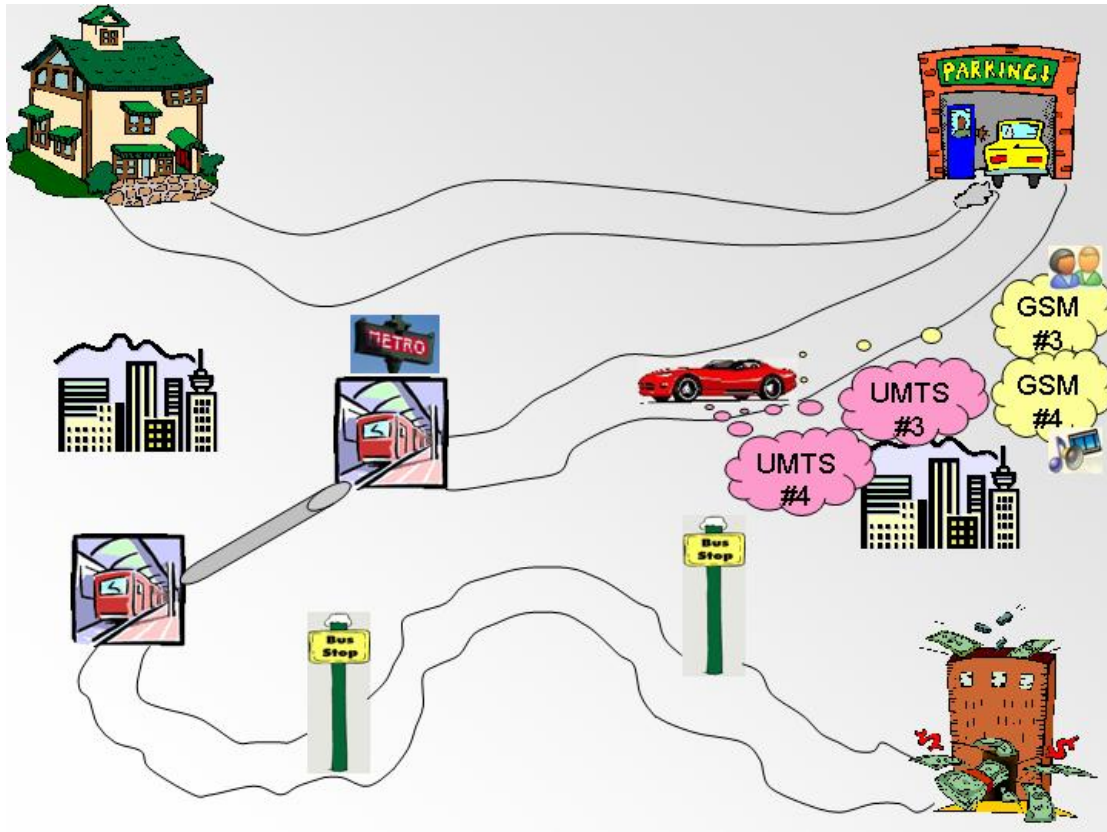
«Μπαίνει μέσα στο αυτοκίνητό του και ξεκινά με κατεύθυνση τον πλησιέστερο σταθμό του μετρό. Ενώ βρίσκεται μέσα στο αυτοκίνητο, κάνει μία κλήση (Voice Call) στη γραμματέα του γραφείου, για να επιβεβαιώσει την ώρα που αρχίζει το σημερινό meeting. Ύστερα από λίγο, τερματίζει την κλήση, όπως, επίσης, και την υπηρεσία Video Stream, για να μην του αποσπά την προσοχή, καθώς οδηγεί».



- Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Video Stream
- ✓ Voice Call

- Screenshots:



- Log File:

---

# Sat May 14 13:59:14 EEST 2005:

# Service VIDEO STREAM has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.11	1	1	4	170
147.102.1.12	2	1	5	170
147.102.2.11	1	2	4	2000
147.102.2.12	3	2	5	2000

The user's maximum preferred QoS level is QoS-4  
and the user's general preferences are:

1. Cost
2. Quality



3. Provider  
and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.11	0.6	1	0,95	4,2105	0,15	0	-0,06158
147.102.1.11	0.6	2	1,9	4,2105	0,3	0	-0,12316
147.102.1.12	0.6	1	1	4,2105	0,14	0	0,01053
147.102.1.12	0.6	2	2	4,2105	0,32	0	-0,14737
147.102.2.11	0.6	1	0,95	4,2105	0,17	0	-0,14579
147.102.2.11	0.6	2	1,9	4,2105	0,33	0	-0,24947
147.102.2.11	0.6	3	2,85	4,2105	0,48	0	-0,31105
147.102.2.11	0.6	4	3,8	4,2105	0,64	0	-0,41474
147.102.2.12	0.6	1	1	4,2105	0,16	0	-0,07368
147.102.2.12	0.6	2	2	4,2105	0,35	0	-0,27368
147.102.2.12	0.6	3	3	4,2105	0,47	0	-0,17895
147.102.2.12	0.6	4	4	4,2105	0,65	0	-0,33684

Sorted Array of Choices: (12 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.12	1	0,01053
2.	147.102.1.11	1	-0,06158
3.	147.102.2.12	1	-0,07368
4.	147.102.1.11	2	-0,12316
5.	147.102.2.11	1	-0,14579
6.	147.102.1.12	2	-0,14737
7.	147.102.2.12	3	-0,17895
8.	147.102.2.11	2	-0,24947
9.	147.102.2.12	2	-0,27368
10.	147.102.2.11	3	-0,31105
11.	147.102.2.12	4	-0,33684
12.	147.102.2.11	4	-0,41474

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.010526315789473606  
 IP Address = 147.102.1.12      Provider Code = 2  
 QoS Level = 1      Service Code = 6  
 The terminal's request has been ACCEPTED!

-----  
 # Sat May 14 13:59:22 EEST 2005:

# Service VOICE CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.11	1	1	4	170

147.102.1.12	2	1	5	170
147.102.2.11	1	2	4	2000
147.102.2.12	3	2	5	2000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Cost
  2. Quality
  3. Provider
- and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.11	0.6	1	0,95	125	0,003	0	0,195
147.102.1.11	0.6	2	1,9	125	0,0055	0	0,4525
147.102.1.11	0.6	3	2,85	125	0,01	0	0,46
147.102.1.12	0.6	1	1	125	0,0032	0	0,2
147.102.1.12	0.6	2	2	125	0,006	0	0,45
147.102.1.12	0.6	3	3	125	0,012	0	0,3
147.102.2.11	0.6	1	0,95	125	0,005	0	-0,055
147.102.2.11	0.6	2	1,9	125	0,009	0	0,015
147.102.2.11	0.6	3	2,85	125	0,015	0	-0,165
147.102.2.12	0.6	1	1	125	0,0047	0	0,0125
147.102.2.12	0.6	2	2	125	0,0087	0	0,1125
147.102.2.12	0.6	3	3	125	0,011	0	0,425

Sorted Array of Choices: (12 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.11	3	0,46
2.	147.102.1.11	2	0,4525
3.	147.102.1.12	2	0,45
4.	147.102.2.12	3	0,425
5.	147.102.1.12	3	0,3
6.	147.102.1.12	1	0,2
7.	147.102.1.11	1	0,195
8.	147.102.2.12	2	0,1125
9.	147.102.2.11	2	0,015
10.	147.102.2.12	1	0,0125
11.	147.102.2.11	1	-0,055
12.	147.102.2.11	3	-0,165

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.45999999999999974  
IP Address = 147.102.1.11      Provider Code = 1  
QoS Level = 3      Service Code = 1  
The terminal's request has been REJECTED.

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.4524999999999999  
IP Address = 147.102.1.11      Provider Code = 1  
QoS Level = 2      Service Code = 1  
The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Στη θέση αυτή, η κάλυψη από το access point το οποίο εξυπηρετούσε την υπηρεσία Video Stream έχει χαθεί, οπότε ο αλγόριθμος εκκινείται για ακόμη μία φορά, προκειμένου να αποφασίσει από ποιο σημείο πρόσβασης θα συνεχιστεί η λήψη της υπηρεσίας. Τελικά, αποφασίζεται η ανάθεσή της σε δίκτυο τεχνολογίας GSM και σε επίπεδο ποιότητας 1. Κανένα από τα διαθέσιμα access points δεν ανήκει στον προτιμώμενο πάροχο, οπότε ο παράγοντας που επηρεάζει με τη μεγαλύτερη βαρύτητα την τελική απόφαση είναι το κόστος.

Όσον αφορά τη νέα υπηρεσία Voice Call, αυτή ανατίθεται στο έτερο σημείο πρόσβασης GSM και σε επίπεδο ποιότητας 2. Στο σημείο αυτό, παρατηρούμε κάποια ιδιαιτερότητα: Η πρώτη επιλογή του αλγορίθμου, όπως φαίνεται και από την ταξινομημένη λίστα επιλογών στο log file, είναι η ανάθεση της υπηρεσίας στο ίδιο μεν access point, αλλά σε επίπεδο ποιότητας 3. Ο λόγος που τελικά δεν πραγματοποιήθηκε αυτή η ανάθεση οφείλεται σε απόρριψη της πρότασης του τερματικού από την οντότητα του δικτύου. Η συμπεριφορά αυτή του δικτύου υπόκειται σε όρους και εμπορικά μοντέλα που έχουν αναλυθεί σε προηγούμενο κεφάλαιο (κεφάλαιο 1). Έτσι, το τερματικό υπέβαλε προς έγκριση τη δεύτερη κατά σειρά επιλογή, η οποία και έγινε τελικά αποδεκτή.

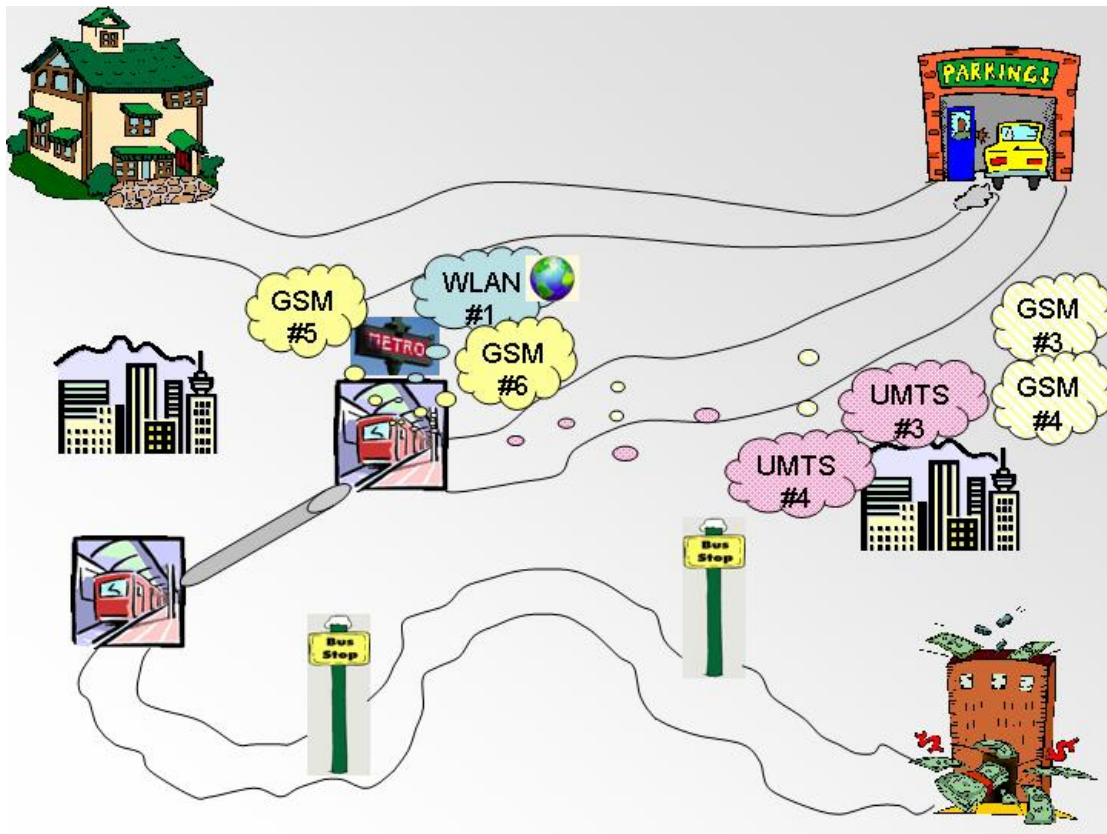
4. «Στο σταθμό του μετρό»:

«Φτάνοντας στο σταθμό του μετρό και, καθώς περιμένει να έρθει το τρένο, εκκινεί τη λήψη (downloading) ενός μεγάλου αρχείου, μέσω της υπηρεσίας Web Browsing. Στο σταθμό του μετρό, νέα σημεία πρόσβασης έχουν κάνει την εμφάνισή τους, μεταξύ των οποίων και ένα WLAN, το οποίο καλύπτει το χώρο του σταθμού».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing

▪ Screenshots:



▪ Log File:

# Sat May 14 13:59:36 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.11	1	1	1	170
147.102.1.12	2	1	1	170
147.102.2.11	1	2	1	2000
147.102.2.12	3	2	1	2000
147.102.1.21	1	1	5	170
147.102.1.22	2	1	4	170
147.102.3.1	4	3	4	3000

The user's maximum preferred QoS level is QoS-5  
and the user's general preferences are:

1. Cost
2. Quality
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.11	0.6	1	0,4	72,7273	0,008	0	-0,34182
147.102.1.11	0.6	2	0,8	72,7273	0,016	0	-0,68364
147.102.1.11	0.6	3	1,2	72,7273	0,024	0	-1,02545
147.102.1.12	0.6	1	0,4	72,7273	0,009	0	-0,41455
147.102.1.12	0.6	2	0,8	72,7273	0,015	0	-0,61091
147.102.1.12	0.6	3	1,2	72,7273	0,022	0	-0,88
147.102.2.11	0.6	1	0,4	72,7273	0,01	0	-0,48727
147.102.2.11	0.6	2	0,8	72,7273	0,018	0	-0,82909
147.102.2.11	0.6	3	1,2	72,7273	0,029	0	-1,38909
147.102.2.11	0.6	4	1,6	72,7273	0,039	0	-1,87636
147.102.2.11	0.6	5	2	72,7273	0,05	0	-2,43636
147.102.2.12	0.6	1	0,4	72,7273	0,009	0	-0,41455
147.102.2.12	0.6	2	0,8	72,7273	0,019	0	-0,90182
147.102.2.12	0.6	3	1,2	72,7273	0,028	0	-1,31636
147.102.2.12	0.6	4	1,6	72,7273	0,04	0	-1,94909
147.102.2.12	0.6	5	2	72,7273	0,051	0	-2,50909
147.102.1.21	0.6	1	1	72,7273	0,008	0	0,01818
147.102.1.21	0.6	2	2	72,7273	0,016	0	0,03636
147.102.1.21	0.6	3	3	72,7273	0,024	0	0,05455
147.102.1.22	0.6	1	0,95	72,7273	0,009	0	-0,08455
147.102.1.22	0.6	2	1,9	72,7273	0,015	0	0,04909
147.102.1.22	0.6	3	2,85	72,7273	0,022	0	0,11
147.102.3.1	0.6	1	0,95	72,7273	0,002	0	0,42455
147.102.3.1	0.6	2	1,9	72,7273	0,006	0	0,70364
147.102.3.1	0.6	3	2,85	72,7273	0,008	0	1,12818
147.102.3.1	0.6	4	3,8	72,7273	0,011	0	1,48
147.102.3.1	0.6	5	4,75	72,7273	0,014	0	1,83182

Sorted Array of Choices: (27 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.1	5	1,83182
2.	147.102.3.1	4	1,48
3.	147.102.3.1	3	1,12818
4.	147.102.3.1	2	0,70364
5.	147.102.3.1	1	0,42455
6.	147.102.1.22	3	0,11
7.	147.102.1.21	3	0,05455
8.	147.102.1.22	2	0,04909
9.	147.102.1.21	2	0,03636
10.	147.102.1.21	1	0,01818
11.	147.102.1.22	1	-0,08455
12.	147.102.1.11	1	-0,34182
13.	147.102.2.12	1	-0,41455
14.	147.102.1.12	1	-0,41455
15.	147.102.2.11	1	-0,48727
16.	147.102.1.12	2	-0,61091

17.	147.102.1.11	2	-0,68364
18.	147.102.2.11	2	-0,82909
19.	147.102.1.12	3	-0,88
20.	147.102.2.12	2	-0,90182
21.	147.102.1.11	3	-1,02545
22.	147.102.2.12	3	-1,31636
23.	147.102.2.11	3	-1,38909
24.	147.102.2.11	4	-1,87636
25.	147.102.2.12	4	-1,94909
26.	147.102.2.11	5	-2,43636
27.	147.102.2.12	5	-2,50909

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 1.8318181818181818

IP Address = 147.102.3.1      Provider Code = 4

QoS Level = 5      Service Code = 7

The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Στην τοποθεσία αυτή, είναι διαθέσιμα 7 access points. Η νέα αιτούμενη υπηρεσία Web Browsing ανατίθεται τελικά σε ασύρματο δίκτυο WLAN και σε επίπεδο ποιότητας 5. Τα πλεονεκτήματα αυτής της επιλογής είναι ότι η παροχή της συγκεκριμένης υπηρεσίας είναι ιδιαίτερα φτηνή πάνω από WLAN, ακόμα και για μεγάλα επίπεδα ποιότητας.

5. «Μέσα στο τρένο»:

«Επιβιβάζεται στο τρένο. Μέσα στο τρένο, το τερματικό του διαπιστώνει ότι το σήμα από ορισμένα σημεία πρόσβασης έχει εξασθενίσει σημαντικά. Ο εργαζόμενος συνεχίζει το downloading, ενώ, επίσης, αποστέλλει ένα e-mail σε ένα συνάδελφό του».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Email

▪ Screenshots:



▪ Log File:

# Sat May 14 13:59:54 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.21	1	1	1	170
147.102.1.31	2	1	4	170
147.102.1.32	4	1	3	170

The user's maximum preferred QoS level is QoS-5  
and the user's general preferences are:

1. Cost
2. Quality
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade

147.102.1.21	0.6	1	0,4	72,7273	0,008	0	-0,34182
147.102.1.21	0.6	2	0,8	72,7273	0,016	0	-0,68364
147.102.1.21	0.6	3	1,2	72,7273	0,024	0	-1,02545
147.102.1.31	0.6	1	0,95	72,7273	0,009	0	-0,08455
147.102.1.31	0.6	2	1,9	72,7273	0,015	0	0,04909
147.102.1.31	0.6	3	2,85	72,7273	0,022	0	0,11
147.102.1.32	0.6	1	0,8	72,7273	0,0085	0	-0,13818
147.102.1.32	0.6	2	1,6	72,7273	0,017	0	-0,27636
147.102.1.32	0.6	3	2,4	72,7273	0,025	0	-0,37818

Sorted Array of Choices: (9 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.31	3	0,11
2.	147.102.1.31	2	0,04909
3.	147.102.1.31	1	-0,08455
4.	147.102.1.32	1	-0,13818
5.	147.102.1.32	2	-0,27636
6.	147.102.1.21	1	-0,34182
7.	147.102.1.32	3	-0,37818
8.	147.102.1.21	2	-0,68364
9.	147.102.1.21	3	-1,02545

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.10999999999999965

IP Address = 147.102.1.31      Provider Code = 2

QoS Level = 3      Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Sat May 14 13:59:56 EEST 2005:

# Service E-MAIL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.21	1	1	1	170
147.102.1.31	2	1	4	170
147.102.1.32	4	1	3	170

The user's maximum preferred QoS level is QoS-1

and the user's general preferences are:

1. Cost
2. Quality
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality	QoS	Real	Cost	Cost	Provider	Grade
------------	---------	-----	------	------	------	----------	-------



	Factor	Level	Quality	Factor		Factor	
147.102.1.21	0.6	1	0,4	1,0256	0,58	0	-0,35487
147.102.1.31	0.6	1	0,95	1,0256	0,59	0	-0,03513
147.102.1.32	0.6	1	0,8	1,0256	0,58	0	-0,11487

Sorted Array of Choices: (3 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.31	1	-0,03513
2.	147.102.1.32	1	-0,11487
3.	147.102.1.21	1	-0,35487

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = -0.03512820512820525  
IP Address = 147.102.1.31      Provider Code = 2  
QoS Level = 1      Service Code = 4  
The terminal's request has been ACCEPTED!

▪ Παρατηρήσεις:

Στη θέση αυτή, η κάλυψη από το WLAN σημείο πρόσβασης, που εξυπηρετούσε την υπηρεσία Web Browsing, έχει χαθεί. Ο αλγόριθμος βελτιστοποίησης επανεξετάζει την υπηρεσία αυτή, προκειμένου να βρει μέσω ποιου σημείου πρόσβασης θα συνεχιστεί η λήψη της. Καταλήγει στην ανάθεσή της σε δίκτυο τεχνολογίας GSM και σε επίπεδο ποιότητας 3. Αυτό είναι το μέγιστο επίπεδο ποιότητας που μπορούν να προσφέρουν για τη συγκεκριμένη υπηρεσία τα διαθέσιμα, τη δεδομένη στιγμή, access points. Το σημείο πρόσβασης GSM που επιλέγεται υπερτερεί έναντι των άλλων δύο, διότι έχει καλύτερη στάθμη σήματος.

Η νέα υπηρεσία Email ανατίθεται, για τον ίδιο λόγο, και αυτή στο παραπάνω σημείο πρόσβασης.

6. «Εκτός του μετρώ»:

«Ο εργαζόμενος αποβιβάζεται από το τρένο και βγαίνει εκτός του σταθμού του μετρώ. Κατευθύνεται προς τη στάση των λεωφορείων, κοντά στο σταθμό. Επειδή έχει καθυστερήσει λίγο, εκκινεί μια βίντεο κλήση (Video Call), προκειμένου να μη χάσει την αρχή του meeting. Αφού έχει ξεκινήσει την υπηρεσία Video Call, διαπιστώνει ότι το παρεχόμενο επίπεδο ποιότητας δεν τον ικανοποιεί απόλυτα. Γι' αυτό, τροποποιεί το ενεργό προφίλ του, καθορίζοντας ως μέγιστο επιθυμητό επίπεδο ποιότητας για την υπηρεσία Video Call το επίπεδο 5 (αντί του 3), ενώ, επίσης, ανακατατάσσει τις τρεις βασικές παραμέτρους των γενικών ρυθμίσεων, τοποθετώντας ως πρώτη την ποιότητα, ως δεύτερη το κόστος και ως τρίτη τον πάροχο».

▪ Αιτούμενες και Τρέγουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Call

▪ Screenshots:

- ✓ Πριν την αλλαγή του προφίλ:



✓ Μετά την αλλαγή του προφίλ:



▪ Log File:

---

# Sat May 14 14:00:14 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.31	2	1	1	170
147.102.1.32	4	1	1	170
147.102.1.41	1	1	2	170
147.102.1.42	4	1	3	170
147.102.2.21	1	2	2	2000
147.102.2.22	3	2	3	2000

The user's maximum preferred QoS level is QoS-5

and the user's general preferences are:

1. Cost
2. Quality
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.31	0.6	1	0,4	72,7273	0,009	0	-0,41455
147.102.1.31	0.6	2	0,8	72,7273	0,015	0	-0,61091
147.102.1.31	0.6	3	1,2	72,7273	0,022	0	-0,88
147.102.1.32	0.6	1	0,4	72,7273	0,0085	0	-0,37818
147.102.1.32	0.6	2	0,8	72,7273	0,017	0	-0,75636
147.102.1.32	0.6	3	1,2	72,7273	0,025	0	-1,09818
147.102.1.41	0.6	1	0,6	72,7273	0,008	0	-0,22182
147.102.1.41	0.6	2	1,2	72,7273	0,016	0	-0,44364
147.102.1.41	0.6	3	1,8	72,7273	0,024	0	-0,66545
147.102.1.42	0.6	1	0,8	72,7273	0,0085	0	-0,13818
147.102.1.42	0.6	2	1,6	72,7273	0,017	0	-0,27636
147.102.1.42	0.6	3	2,4	72,7273	0,025	0	-0,37818
147.102.2.21	0.6	1	0,6	72,7273	0,01	0	-0,36727
147.102.2.21	0.6	2	1,2	72,7273	0,018	0	-0,58909
147.102.2.21	0.6	3	1,8	72,7273	0,029	0	-1,02909
147.102.2.21	0.6	4	2,4	72,7273	0,039	0	-1,39636
147.102.2.21	0.6	5	3	72,7273	0,05	0	-1,83636
147.102.2.22	0.6	1	0,8	72,7273	0,009	0	-0,17455
147.102.2.22	0.6	2	1,6	72,7273	0,019	0	-0,42182
147.102.2.22	0.6	3	2,4	72,7273	0,028	0	-0,59636
147.102.2.22	0.6	4	3,2	72,7273	0,04	0	-0,98909
147.102.2.22	0.6	5	4	72,7273	0,051	0	-1,30909

Sorted Array of Choices: (22 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.42	1	-0,13818
2.	147.102.2.22	1	-0,17455
3.	147.102.1.41	1	-0,22182
4.	147.102.1.42	2	-0,27636
5.	147.102.2.21	1	-0,36727
6.	147.102.1.42	3	-0,37818
7.	147.102.1.32	1	-0,37818
8.	147.102.1.31	1	-0,41455
9.	147.102.2.22	2	-0,42182
10.	147.102.1.41	2	-0,44364
11.	147.102.2.21	2	-0,58909
12.	147.102.2.22	3	-0,59636
13.	147.102.1.31	2	-0,61091
14.	147.102.1.41	3	-0,66545
15.	147.102.1.32	2	-0,75636
16.	147.102.1.31	3	-0,88
17.	147.102.2.22	4	-0,98909
18.	147.102.2.21	3	-1,02909
19.	147.102.1.32	3	-1,09818
20.	147.102.2.22	5	-1,30909
21.	147.102.2.21	4	-1,39636

22. 147.102.2.21 5 -1,83636

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1 Grade = -0.1381818181818183

IP Address = 147.102.1.42 Provider Code = 4

QoS Level = 1 Service Code = 7

The terminal's request has been ACCEPTED!

-----  
# Sat May 14 14:00:16 EEST 2005:

# Service VIDEO CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.31	2	1	1	170
147.102.1.32	4	1	1	170
147.102.1.41	1	1	2	170
147.102.1.42	4	1	3	170
147.102.2.21	1	2	2	2000
147.102.2.22	3	2	3	2000

The user's maximum preferred QoS level is QoS-3  
and the user's general preferences are:

1. Cost
2. Quality
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.31	0.6	1	0,4	3,3613	0,2	0	-0,43227
147.102.1.31	0.6	2	0,8	3,3613	0,41	0	-0,89815
147.102.1.31	0.6	3	1,2	3,3613	0,63	0	-1,39765
147.102.1.32	0.6	1	0,4	3,3613	0,19	0	-0,39866
147.102.1.32	0.6	2	0,8	3,3613	0,42	0	-0,93176
147.102.1.32	0.6	3	1,2	3,3613	0,59	0	-1,26319
147.102.1.41	0.6	1	0,6	3,3613	0,2	0	-0,31227
147.102.1.41	0.6	2	1,2	3,3613	0,4	0	-0,62454
147.102.1.41	0.6	3	1,8	3,3613	0,6	0	-0,93681
147.102.1.42	0.6	1	0,8	3,3613	0,19	0	-0,15866
147.102.1.42	0.6	2	1,6	3,3613	0,42	0	-0,45176
147.102.1.42	0.6	3	2,4	3,3613	0,59	0	-0,54319
147.102.2.21	0.6	1	0,6	3,3613	0,25	0	-0,48034
147.102.2.21	0.6	2	1,2	3,3613	0,46	0	-0,82622
147.102.2.21	0.6	3	1,8	3,3613	0,66	0	-1,13849

147.102.2.22	0.6	1	0,8	3,3613	0,24	0	-0,32672
147.102.2.22	0.6	2	1,6	3,3613	0,46	0	-0,58622
147.102.2.22	0.6	3	2,4	3,3613	0,67	0	-0,8121

Sorted Array of Choices: (18 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.42	1	-0,15866
2.	147.102.1.41	1	-0,31227
3.	147.102.2.22	1	-0,32672
4.	147.102.1.32	1	-0,39866
5.	147.102.1.31	1	-0,43227
6.	147.102.1.42	2	-0,45176
7.	147.102.2.21	1	-0,48034
8.	147.102.1.42	3	-0,54319
9.	147.102.2.22	2	-0,58622
10.	147.102.1.41	2	-0,62454
11.	147.102.2.22	3	-0,8121
12.	147.102.2.21	2	-0,82622
13.	147.102.1.31	2	-0,89815
14.	147.102.1.32	2	-0,93176
15.	147.102.1.41	3	-0,93681
16.	147.102.2.21	3	-1,13849
17.	147.102.1.32	3	-1,26319
18.	147.102.1.31	3	-1,39765

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = -0.15865546218487403

IP Address = 147.102.1.42      Provider Code = 4

QoS Level = 1      Service Code = 5

The terminal's request has been ACCEPTED!

# Sat May 14 14:00:30 EEST 2005:

-----

# Sat May 14 14:00:30 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.31	2	1	1	170
147.102.1.32	4	1	1	170
147.102.1.41	1	1	2	170
147.102.1.42	4	1	3	170
147.102.2.21	1	2	2	2000
147.102.2.22	3	2	3	2000

The user's maximum preferred QoS level is QoS-5

and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.31	0.8	1	0,4	54,5455	0,009	0	-0,17091
147.102.1.31	0.8	2	0,8	54,5455	0,015	0	-0,17818
147.102.1.31	0.8	3	1,2	54,5455	0,022	0	-0,24
147.102.1.32	0.8	1	0,4	54,5455	0,0085	0	-0,14364
147.102.1.32	0.8	2	0,8	54,5455	0,017	0	-0,28727
147.102.1.32	0.8	3	1,2	54,5455	0,025	0	-0,40364
147.102.1.41	0.8	1	0,6	54,5455	0,008	0	0,04364
147.102.1.41	0.8	2	1,2	54,5455	0,016	0	0,08727
147.102.1.41	0.8	3	1,8	54,5455	0,024	0	0,13091
147.102.1.42	0.8	1	0,8	54,5455	0,0085	0	0,17636
147.102.1.42	0.8	2	1,6	54,5455	0,017	0	0,35273
147.102.1.42	0.8	3	2,4	54,5455	0,025	0	0,55636
147.102.2.21	0.8	1	0,6	54,5455	0,01	0	-0,06545
147.102.2.21	0.8	2	1,2	54,5455	0,018	0	-0,02182
147.102.2.21	0.8	3	1,8	54,5455	0,029	0	-0,14182
147.102.2.21	0.8	4	2,4	54,5455	0,039	0	-0,20727
147.102.2.21	0.8	5	3	54,5455	0,05	0	-0,32727
147.102.2.22	0.8	1	0,8	54,5455	0,009	0	0,14909
147.102.2.22	0.8	2	1,6	54,5455	0,019	0	0,24364
147.102.2.22	0.8	3	2,4	54,5455	0,028	0	0,39273
147.102.2.22	0.8	4	3,2	54,5455	0,04	0	0,37818
147.102.2.22	0.8	5	4	54,5455	0,051	0	0,41818

Sorted Array of Choices: (22 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.42	3	0,55636
2.	147.102.2.22	5	0,41818
3.	147.102.2.22	3	0,39273
4.	147.102.2.22	4	0,37818
5.	147.102.1.42	2	0,35273
6.	147.102.2.22	2	0,24364
7.	147.102.1.42	1	0,17636
8.	147.102.2.22	1	0,14909
9.	147.102.1.41	3	0,13091
10.	147.102.1.41	2	0,08727
11.	147.102.1.41	1	0,04364
12.	147.102.2.21	2	-0,02182
13.	147.102.2.21	1	-0,06545
14.	147.102.2.21	3	-0,14182
15.	147.102.1.32	1	-0,14364

16. 147.102.1.31 1 -0,17091  
 17. 147.102.1.31 2 -0,17818  
 18. 147.102.2.21 4 -0,20727  
 19. 147.102.1.31 3 -0,24  
 20. 147.102.1.32 2 -0,28727  
 21. 147.102.2.21 5 -0,32727  
 22. 147.102.1.32 3 -0,40364

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1 Grade = 0.5563636363636366  
 IP Address = 147.102.1.42 Provider Code = 4  
 QoS Level = 3 Service Code = 7  
 The terminal's request has been REJECTED.

A connection with the following details is being requested from the nearest network entity:

Access Technology = 2 Grade = 0.41818181818181843  
 IP Address = 147.102.2.22 Provider Code = 3  
 QoS Level = 5 Service Code = 7  
 The terminal's request has been ACCEPTED!

-----  
 # Sat May 14 14:00:30 EEST 2005:

# Service VIDEO CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.31	2	1	1	170
147.102.1.32	4	1	1	170
147.102.1.41	1	1	2	170
147.102.1.42	4	1	3	170
147.102.2.21	1	2	2	2000
147.102.2.22	3	2	3	2000

The user's maximum preferred QoS level is QoS-5

and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.31	0.8	1	0,4	2,521	0,2	0	-0,1842
147.102.1.31	0.8	2	0,8	2,521	0,41	0	-0,39361
147.102.1.31	0.8	3	1,2	2,521	0,63	0	-0,62824



147.102.1.32	0.8	1	0,4	2,521	0,19	0	-0,15899
147.102.1.32	0.8	2	0,8	2,521	0,42	0	-0,41882
147.102.1.32	0.8	3	1,2	2,521	0,59	0	-0,52739
147.102.1.41	0.8	1	0,6	2,521	0,2	0	-0,0242
147.102.1.41	0.8	2	1,2	2,521	0,4	0	-0,0484
147.102.1.41	0.8	3	1,8	2,521	0,6	0	-0,07261
147.102.1.42	0.8	1	0,8	2,521	0,19	0	0,16101
147.102.1.42	0.8	2	1,6	2,521	0,42	0	0,22118
147.102.1.42	0.8	3	2,4	2,521	0,59	0	0,43261
147.102.2.21	0.8	1	0,6	2,521	0,25	0	-0,15025
147.102.2.21	0.8	2	1,2	2,521	0,46	0	-0,19966
147.102.2.21	0.8	3	1,8	2,521	0,66	0	-0,22387
147.102.2.21	0.8	4	2,4	2,521	0,92	0	-0,39933
147.102.2.21	0.8	5	3	2,521	1,1	0	-0,37311
147.102.2.22	0.8	1	0,8	2,521	0,24	0	0,03496
147.102.2.22	0.8	2	1,6	2,521	0,46	0	0,12034
147.102.2.22	0.8	3	2,4	2,521	0,67	0	0,23092
147.102.2.22	0.8	4	3,2	2,521	0,92	0	0,24067
147.102.2.22	0.8	5	4	2,521	1,13	0	0,35126

Sorted Array of Choices: (22 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.42	3	0,43261
2.	147.102.2.22	5	0,35126
3.	147.102.2.22	4	0,24067
4.	147.102.2.22	3	0,23092
5.	147.102.1.42	2	0,22118
6.	147.102.1.42	1	0,16101
7.	147.102.2.22	2	0,12034
8.	147.102.2.22	1	0,03496
9.	147.102.1.41	1	-0,0242
10.	147.102.1.41	2	-0,0484
11.	147.102.1.41	3	-0,07261
12.	147.102.2.21	1	-0,15025
13.	147.102.1.32	1	-0,15899
14.	147.102.1.31	1	-0,1842
15.	147.102.2.21	2	-0,19966
16.	147.102.2.21	3	-0,22387
17.	147.102.2.21	5	-0,37311
18.	147.102.1.31	2	-0,39361
19.	147.102.2.21	4	-0,39933
20.	147.102.1.32	2	-0,41882
21.	147.102.1.32	3	-0,52739
22.	147.102.1.31	3	-0,62824

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.4326050420168073  
IP Address = 147.102.1.42      Provider Code = 4

QoS Level = 3                      Service Code = 5  
The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Στη θέση αυτή, εμφανίζονται 4 ακόμη access points, οπότε ο αλγόριθμος επανεκκινείται για την υπηρεσία Web Browsing, προκειμένου να αποφανθεί για τυχόν καλύτερη επιλογή ανάθεσης. Πράγματι, το αποτέλεσμα της αντικειμενικής συνάρτησης οδηγεί σε ανάθεση αυτής της υπηρεσίας σε ένα νέο access point τύπου GSM και σε επίπεδο ποιότητας 1. Οι λόγοι εστιάζονται, κυρίως, στο κόστος, αλλά και στην υποβάθμιση του λαμβανόμενου σήματος από το προηγούμενο σημείο πρόσβασης.

Η νέα υπηρεσία Video Call ανατίθεται, για τους ίδιους λόγους, και αυτή στο παραπάνω σημείο πρόσβασης. Παρατηρούμε ότι το μέγιστο επίπεδο ποιότητας που εξετάζεται από τον αλγόριθμο για την υπηρεσία αυτή είναι το επίπεδο 3, καθώς έτσι έχει δηλώσει στις προτιμήσεις του ο χρήστης.

Στο σημείο αυτό, ο χρήστης επιθυμεί να αυξήσει το επίπεδο ποιότητας των λαμβανόμενων υπηρεσιών. Για το σκοπό αυτό, τροποποιεί το προφίλ των προτιμήσεών του ως εξής: Θέτει ως πρώτη παράμετρο την ποιότητα και ως δεύτερη το κόστος. Επιπλέον, δηλώνει ότι το μέγιστο επίπεδο ποιότητας που επιθυμεί για την υπηρεσία Video Call είναι το επίπεδο 5 (αντί του 3).

Η τροποποίηση των προτιμήσεων του χρήστη πυροδοτεί νέο τρέξιμο του αλγορίθμου, που οδηγεί στην ανάθεση της υπηρεσίας Web Browsing σε σημείο πρόσβασης της τεχνολογίας UMTS και σε επίπεδο ποιότητας 5. Παρατηρούμε, δηλαδή, ότι ο καθορισμός της ποιότητας ως πρωταρχικού παράγοντα επηρέασε τη διαδικασία λήψης αποφάσεων. Βέβαια, και εδώ αξίζει να σημειωθεί ότι η επιλογή αυτή είναι η δεύτερη επικρατέστερη επιλογή του τερματικού, καθώς η πρώτη επιλογή απορρίφθηκε από την οντότητα διαχείρισης του δικτύου.

Η δεύτερη τρέχουσα υπηρεσία, Video Call, παρέμεινε στο ίδιο σημείο πρόσβασης, αλλά το επίπεδο ποιότητάς της αυξήθηκε από 1 σε 3. Η παραμονή στο ίδιο σημείο πρόσβασης οφείλεται στο ότι ο παράγοντας «κόστος» μπορεί να υποβαθμίστηκε, δεν έπαψε, όμως, να επηρεάζει τη διαδικασία λήψης απόφασης. Ο χρήστης, εφόσον το επιθυμούσε, θα μπορούσε να περιορίσει ακόμα περισσότερο τη βαρύτητα του παράγοντα «κόστος», τοποθετώντας τον ως τρίτο και όχι ως δεύτερο. Αξιοσημείωτο είναι, ακόμη, περί της σωστής λειτουργίας του αλγορίθμου, ότι έχει προταθεί ως δεύτερη επικρατέστερη λύση η λήψη της υπηρεσίας σε επίπεδο ποιότητας 5 (δηλαδή, στη μέγιστη δυνατή ποιότητα) και από δίκτυο UMTS.

7. «Μέσα στο λεωφορείο»:

«Επιβιβάζεται στο λεωφορείο. Στη διαδρομή, εξακολουθεί να εκτελεί τις υπηρεσίες Video Call και Web Browsing».

- Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Call

- Screenshots:



- Log File:

- ✓ Δεν υπάρχει καινούρια εγγραφή στο log file.

- Παρατηρήσεις:

Στη θέση αυτή, παρατηρούμε ότι δεν υπάρχει κάποια νέα εγγραφή στο log file. Αυτό συμβαίνει, διότι δεν έχουν εμφανιστεί κάποια νέα access points ούτε έχει γίνει αίτηση για νέα υπηρεσία και το προφίλ του χρήστη παραμένει σταθερό. Η εξαφάνιση κάποιων access points δεν έχει ουσιαστική σημασία, αφού κανένα από αυτά δεν εξυπηρετούσε κάποια υπηρεσία. Συνεπώς, οι τρέχουσες υπηρεσίες συνεχίζουν να παρέχονται από τα ίδια σημεία πρόσβασης.

8. «Πλησιάζοντας στον προορισμό»:

«Επιτέλους φτάνει κοντά στο γραφείο. Αποβιβάζεται από το λεωφορείο και κατευθύνεται πεζός προς το κτίριο του γραφείου του. Συνεχίζει να παρακολουθεί το meeting, μέσω Video Call, και να κατεβάζει το αρχείο, μέσω Web Browsing. Επιπλέον, στέλνει ένα SMS στη σύζυγό του, για να την ειδοποιήσει ότι σχεδόν έφτασε στο γραφείο του».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Call
- ✓ SMS

▪ Screenshots:



▪ Log File:

---

# Sat May 14 14:00:54 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
------------	----------	------------	--------	---------------------

147.102.1.41	1	1	1	170
147.102.1.42	4	1	1	170
147.102.2.21	3	2	1	2000
147.102.1.51	1	1	3	170
147.102.1.52	4	1	3	170
147.102.2.31	1	2	3	1500

The user's maximum preferred QoS level is QoS-5  
and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.41	0.8	1	0,4	54,5455	0,008	0	-0,11636
147.102.1.41	0.8	2	0,8	54,5455	0,016	0	-0,23273
147.102.1.41	0.8	3	1,2	54,5455	0,024	0	-0,34909
147.102.1.42	0.8	1	0,4	54,5455	0,0085	0	-0,14364
147.102.1.42	0.8	2	0,8	54,5455	0,017	0	-0,28727
147.102.1.42	0.8	3	1,2	54,5455	0,025	0	-0,40364
147.102.2.21	0.8	1	0,4	54,5455	0,009	0	-0,17091
147.102.2.21	0.8	2	0,8	54,5455	0,019	0	-0,39636
147.102.2.21	0.8	3	1,2	54,5455	0,028	0	-0,56727
147.102.2.21	0.8	4	1,6	54,5455	0,04	0	-0,90182
147.102.2.21	0.8	5	2	54,5455	0,051	0	-1,18182
147.102.1.51	0.8	1	0,8	54,5455	0,008	0	0,20364
147.102.1.51	0.8	2	1,6	54,5455	0,016	0	0,40727
147.102.1.51	0.8	3	2,4	54,5455	0,024	0	0,61091
147.102.1.52	0.8	1	0,8	54,5455	0,0085	0	0,17636
147.102.1.52	0.8	2	1,6	54,5455	0,017	0	0,35273
147.102.1.52	0.8	3	2,4	54,5455	0,025	0	0,55636
147.102.2.31	0.8	1	0,8	54,5455	0,01	0	0,09455
147.102.2.31	0.8	2	1,6	54,5455	0,018	0	0,29818
147.102.2.31	0.8	3	2,4	54,5455	0,029	0	0,33818
147.102.2.31	0.8	4	3,2	54,5455	0,039	0	0,43273
147.102.2.31	0.8	5	4	54,5455	0,05	0	0,47273

Sorted Array of Choices: (22 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.51	3	0,61091
2.	147.102.1.52	3	0,55636
3.	147.102.2.31	5	0,47273
4.	147.102.2.31	4	0,43273
5.	147.102.1.51	2	0,40727
6.	147.102.1.52	2	0,35273
7.	147.102.2.31	3	0,33818

8.	147.102.2.31	2	0,29818
9.	147.102.1.51	1	0,20364
10.	147.102.1.52	1	0,17636
11.	147.102.2.31	1	0,09455
12.	147.102.1.41	1	-0,11636
13.	147.102.1.42	1	-0,14364
14.	147.102.2.21	1	-0,17091
15.	147.102.1.41	2	-0,23273
16.	147.102.1.42	2	-0,28727
17.	147.102.1.41	3	-0,34909
18.	147.102.2.21	2	-0,39636
19.	147.102.1.42	3	-0,40364
20.	147.102.2.21	3	-0,56727
21.	147.102.2.21	4	-0,90182
22.	147.102.2.21	5	-1,18182

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.6109090909090913  
IP Address = 147.102.1.51    Provider Code = 1  
QoS Level = 3                Service Code = 7  
The terminal's request has been ACCEPTED!

-----  
# Sat May 14 14:00:55 EEST 2005:

# Service VIDEO CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.41	1	1	1	170
147.102.1.42	4	1	1	170
147.102.2.21	3	2	1	2000
147.102.1.51	1	1	3	170
147.102.1.52	4	1	3	170
147.102.2.31	1	2	3	1500

The user's maximum preferred QoS level is QoS-5  
and the user's general preferences are:

1. Quality
  2. Cost
  3. Provider
- and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.41	0.8	1	0,4	2,521	0,2	0	-0,1842
147.102.1.41	0.8	2	0,8	2,521	0,4	0	-0,3684

147.102.1.41	0.8	3	1,2	2,521	0,6	0	-0,55261
147.102.1.42	0.8	1	0,4	2,521	0,19	0	-0,15899
147.102.1.42	0.8	2	0,8	2,521	0,42	0	-0,41882
147.102.1.42	0.8	3	1,2	2,521	0,59	0	-0,52739
147.102.2.21	0.8	1	0,4	2,521	0,24	0	-0,28504
147.102.2.21	0.8	2	0,8	2,521	0,46	0	-0,51966
147.102.2.21	0.8	3	1,2	2,521	0,67	0	-0,72908
147.102.2.21	0.8	4	1,6	2,521	0,92	0	-1,03933
147.102.2.21	0.8	5	2	2,521	1,13	0	-1,24874
147.102.1.51	0.8	1	0,8	2,521	0,2	0	0,1358
147.102.1.51	0.8	2	1,6	2,521	0,4	0	0,2716
147.102.1.51	0.8	3	2,4	2,521	0,6	0	0,40739
147.102.1.52	0.8	1	0,8	2,521	0,19	0	0,16101
147.102.1.52	0.8	2	1,6	2,521	0,42	0	0,22118
147.102.1.52	0.8	3	2,4	2,521	0,59	0	0,43261
147.102.2.31	0.8	1	0,8	2,521	0,25	0	0,00975
147.102.2.31	0.8	2	1,6	2,521	0,46	0	0,12034
147.102.2.31	0.8	3	2,4	2,521	0,66	0	0,25613
147.102.2.31	0.8	4	3,2	2,521	0,92	0	0,24067
147.102.2.31	0.8	5	4	2,521	1,1	0	0,42689

Sorted Array of Choices: (22 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.52	3	0,43261
2.	147.102.2.31	5	0,42689
3.	147.102.1.51	3	0,40739
4.	147.102.1.51	2	0,2716
5.	147.102.2.31	3	0,25613
6.	147.102.2.31	4	0,24067
7.	147.102.1.52	2	0,22118
8.	147.102.1.52	1	0,16101
9.	147.102.1.51	1	0,1358
10.	147.102.2.31	2	0,12034
11.	147.102.2.31	1	0,00975
12.	147.102.1.42	1	-0,15899
13.	147.102.1.41	1	-0,1842
14.	147.102.2.21	1	-0,28504
15.	147.102.1.41	2	-0,3684
16.	147.102.1.42	2	-0,41882
17.	147.102.2.21	2	-0,51966
18.	147.102.1.42	3	-0,52739
19.	147.102.1.41	3	-0,55261
20.	147.102.2.21	3	-0,72908
21.	147.102.2.21	4	-1,03933
22.	147.102.2.21	5	-1,24874

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1      Grade = 0.4326050420168073

IP Address = 147.102.1.52    Provider Code = 4  
 QoS Level = 3                    Service Code = 5  
 The terminal's request has been ACCEPTED!

-----  
 # Sat May 14 14:01:03 EEST 2005:

# Service SMS has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.41	1	1	1	170
147.102.1.42	4	1	1	170
147.102.2.21	3	2	1	2000
147.102.1.51	1	1	3	170
147.102.1.52	4	1	3	170
147.102.2.31	1	2	3	1500

The user's maximum preferred QoS level is QoS-1  
 and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.41	0.8	1	0,4	4,6154	0,09	0	-0,09538
147.102.1.42	0.8	1	0,4	4,6154	0,075	0	-0,02615
147.102.2.21	0.8	1	0,4	4,6154	0,07	0	-0,00308
147.102.1.51	0.8	1	0,8	4,6154	0,09	0	0,22462
147.102.1.52	0.8	1	0,8	4,6154	0,075	0	0,29385
147.102.2.31	0.8	1	0,8	4,6154	0,09	0	0,22462

Sorted Array of Choices: (6 choices)

	IP Address	QoS Level	Grade
1.	147.102.1.52	1	0,29385
2.	147.102.2.31	1	0,22462
3.	147.102.1.51	1	0,22462
4.	147.102.2.21	1	-0,00308
5.	147.102.1.42	1	-0,02615
6.	147.102.1.41	1	-0,09538

A connection with the following details is being requested from the nearest network entity:

Access Technology = 1            Grade = 0.29384615384615403  
 IP Address = 147.102.1.52    Provider Code = 4  
 QoS Level = 1                    Service Code = 2



The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Εδώ, έχουμε τόσο την εμφάνιση νέων access points όσο και την εξαφάνιση του access point το οποίο παρείχε την υπηρεσία Video Call. Έτσι, ο αλγόριθμος εκκινείται για το σύνολο των τρεχουσών υπηρεσιών. Για την υπηρεσία Web Browsing, αποφασίζει την ανάθεσή της σε δίκτυο GSM και σε επίπεδο ποιότητας 3. Για την υπηρεσία Video Call, αποφασίζει την ανάθεσή της σε άλλο σημείο πρόσβασης τύπου GSM και σε επίπεδο ποιότητας 3. Η νέα υπηρεσία SMS ανατίθεται και αυτή στο ίδιο σημείο πρόσβασης το οποίο εξυπηρετεί την υπηρεσία Video Call.

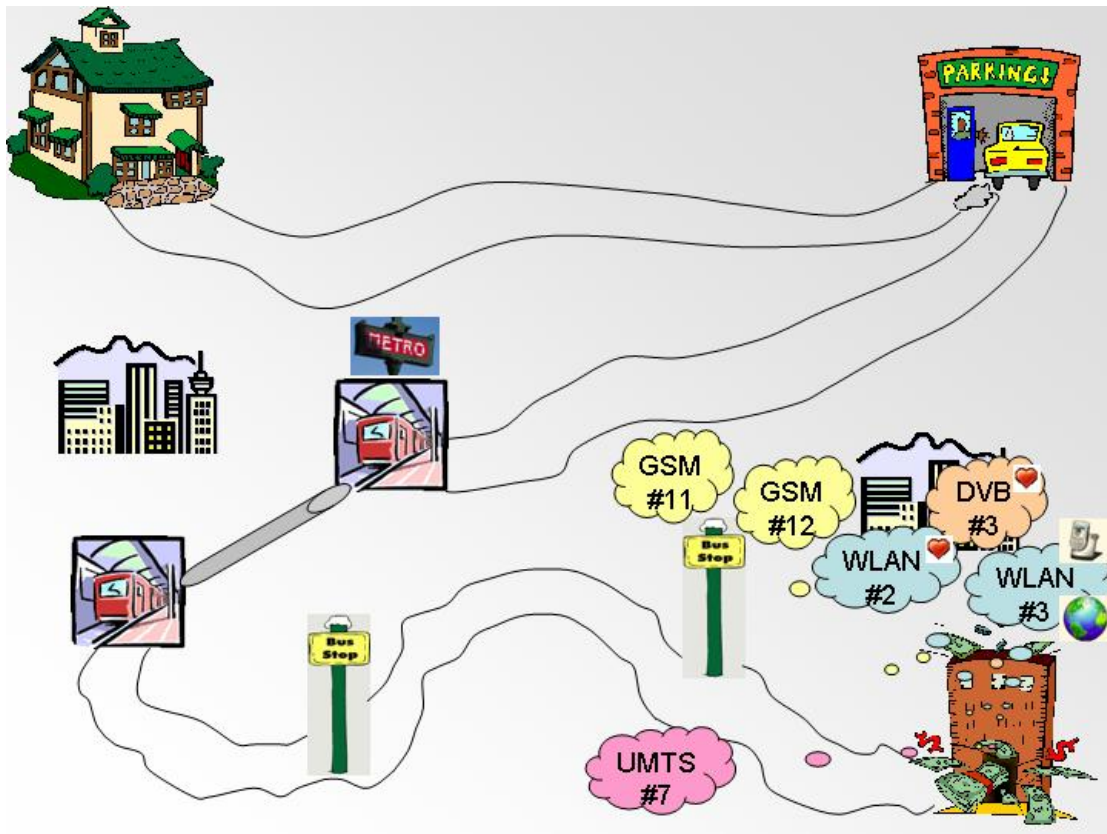
9. «Μέσα στο γραφείο»:

«Υστερα από όλη αυτήν την ταλαιπωρία, δεν του έχει μείνει και πολλή όρεξη για δουλειά! Ωστόσο, φτάνοντας στο γραφείο, πρέπει να πάει αμέσως στο meeting, που έχει αρχίσει. Μπαίνοντας στην αίθουσα συνεδριάσεων, τερματίζει την υπηρεσία Video Call, όπως και την υπηρεσία Web Browsing».

▪ Αιτούμενες και Τρέχουσες Υπηρεσίες:

- ✓ Web Browsing
- ✓ Video Call

▪ Screenshots:



▪ Log File:

# Sat May 14 14:01:15 EEST 2005:

# Service WEB BROWSING has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.51	1	1	5	170
147.102.1.52	4	1	4	170
147.102.2.31	1	2	5	1500
147.102.4.11	5	4	5	4000
147.102.3.11	5	3	2	3000
147.102.3.12	6	3	5	3500

The user's maximum preferred QoS level is QoS-5  
and the user's general preferences are:

1. Quality
  2. Cost
  3. Provider
- and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.51	0.8	1	1	54,5455	0,008	0	0,36364
147.102.1.51	0.8	2	2	54,5455	0,016	0	0,72727
147.102.1.51	0.8	3	3	54,5455	0,024	0	1,09091
147.102.1.52	0.8	1	0,95	54,5455	0,0085	0	0,29636
147.102.1.52	0.8	2	1,9	54,5455	0,017	0	0,59273
147.102.1.52	0.8	3	2,85	54,5455	0,025	0	0,91636
147.102.2.31	0.8	1	1	54,5455	0,01	0	0,25455
147.102.2.31	0.8	2	2	54,5455	0,018	0	0,61818
147.102.2.31	0.8	3	3	54,5455	0,029	0	0,81818
147.102.2.31	0.8	4	4	54,5455	0,039	0	1,07273
147.102.2.31	0.8	5	5	54,5455	0,05	0	1,27273
147.102.4.11	0.8	1	1	54,5455	0,0099	0,4	0,312
147.102.4.11	0.8	2	2	54,5455	0,022	0,4	0,48
147.102.4.11	0.8	3	3	54,5455	0,033	0,4	0,72
147.102.4.11	0.8	4	4	54,5455	0,044	0,4	0,96
147.102.4.11	0.8	5	5	54,5455	0,055	0,4	1,2
147.102.3.11	0.8	1	0,6	54,5455	0,0023	0,4	0,42545
147.102.3.11	0.8	2	1,2	54,5455	0,0071	0,4	0,68727
147.102.3.11	0.8	3	1,8	54,5455	0,009	0,4	1,13891
147.102.3.11	0.8	4	2,4	54,5455	0,012	0,4	1,51855
147.102.3.11	0.8	5	3	54,5455	0,015	0,4	1,89818
147.102.3.12	0.8	1	1	54,5455	0,0019	0	0,69636
147.102.3.12	0.8	2	2	54,5455	0,0055	0	1,3
147.102.3.12	0.8	3	3	54,5455	0,0078	0	1,97455
147.102.3.12	0.8	4	4	54,5455	0,01	0	2,65455
147.102.3.12	0.8	5	5	54,5455	0,014	0	3,23636

Sorted Array of Choices: (26 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.12	5	3,23636
2.	147.102.3.12	4	2,65455
3.	147.102.3.12	3	1,97455
4.	147.102.3.11	5	1,89818
5.	147.102.3.11	4	1,51855
6.	147.102.3.12	2	1,3
7.	147.102.2.31	5	1,27273
8.	147.102.4.11	5	1,2
9.	147.102.3.11	3	1,13891
10.	147.102.1.51	3	1,09091
11.	147.102.2.31	4	1,07273
12.	147.102.4.11	4	0,96
13.	147.102.1.52	3	0,91636
14.	147.102.2.31	3	0,81818
15.	147.102.1.51	2	0,72727
16.	147.102.4.11	3	0,72
17.	147.102.3.12	1	0,69636
18.	147.102.3.11	2	0,68727

```

19. 147.102.2.31 2 0,61818
20. 147.102.1.52 2 0,59273
21. 147.102.4.11 2 0,48
22. 147.102.3.11 1 0,42545
23. 147.102.1.51 1 0,36364
24. 147.102.4.11 1 0,312
25. 147.102.1.52 1 0,29636
26. 147.102.2.31 1 0,25455

```

A connection with the following details is being requested from the nearest network entity:

```

Access Technology = 3      Grade = 3.2363636363636363
IP Address = 147.102.3.12  Provider Code = 6
QoS Level = 5            Service Code = 7
The terminal's request has been ACCEPTED!

```

-----

# Sat May 14 14:01:15 EEST 2005:

# Service VIDEO CALL has been requested

The current measurements are:

IP Address	Provider	Technology	Signal	Available Bandwidth
147.102.1.51	1	1	5	170
147.102.1.52	4	1	4	170
147.102.2.31	1	2	5	1500
147.102.4.11	5	4	5	4000
147.102.3.11	5	3	2	3000
147.102.3.12	6	3	5	3500

The user's maximum preferred QoS level is QoS-5 and the user's general preferences are:

1. Quality
2. Cost
3. Provider

and the preferred provider is 5

The list of choices is:

IP Address	Quality Factor	QoS Level	Real Quality	Cost Factor	Cost	Provider Factor	Grade
147.102.1.51	0.8	1	1	2,521	0,2	0	0,2958
147.102.1.51	0.8	2	2	2,521	0,4	0	0,5916
147.102.1.51	0.8	3	3	2,521	0,6	0	0,88739
147.102.1.52	0.8	1	0,95	2,521	0,19	0	0,28101
147.102.1.52	0.8	2	1,9	2,521	0,42	0	0,46118
147.102.1.52	0.8	3	2,85	2,521	0,59	0	0,79261
147.102.2.31	0.8	1	1	2,521	0,25	0	0,16975
147.102.2.31	0.8	2	2	2,521	0,46	0	0,44034
147.102.2.31	0.8	3	3	2,521	0,66	0	0,73613

147.102.2.31	0.8	4	4	2,521	0,92	0	0,88067
147.102.2.31	0.8	5	5	2,521	1,1	0	1,22689
147.102.4.11	0.8	1	1	2,521	0,13	0,4	0,56672
147.102.4.11	0.8	2	2	2,521	0,27	0,4	1,10319
147.102.4.11	0.8	3	3	2,521	0,45	0,4	1,51866
147.102.4.11	0.8	4	4	2,521	0,64	0,4	1,90387
147.102.4.11	0.8	5	5	2,521	0,96	0,4	1,8958
147.102.3.11	0.8	1	0,6	2,521	0,15	0,4	0,12222
147.102.3.11	0.8	2	1,2	2,521	0,28	0,4	0,30494
147.102.3.11	0.8	3	1,8	2,521	0,46	0,4	0,3364
147.102.3.11	0.8	4	2,4	2,521	0,59	0,4	0,51913
147.102.3.11	0.8	5	3	2,521	0,7	0,4	0,76235
147.102.3.12	0.8	1	1	2,521	0,16	0	0,39664
147.102.3.12	0.8	2	2	2,521	0,29	0	0,86891
147.102.3.12	0.8	3	3	2,521	0,45	0	1,26555
147.102.3.12	0.8	4	4	2,521	0,62	0	1,63697
147.102.3.12	0.8	5	5	2,521	0,75	0	2,10924

Sorted Array of Choices: (26 choices)

	IP Address	QoS Level	Grade
1.	147.102.3.12	5	2,10924
2.	147.102.4.11	4	1,90387
3.	147.102.4.11	5	1,8958
4.	147.102.3.12	4	1,63697
5.	147.102.4.11	3	1,51866
6.	147.102.3.12	3	1,26555
7.	147.102.2.31	5	1,22689
8.	147.102.4.11	2	1,10319
9.	147.102.1.51	3	0,88739
10.	147.102.2.31	4	0,88067
11.	147.102.3.12	2	0,86891
12.	147.102.1.52	3	0,79261
13.	147.102.3.11	5	0,76235
14.	147.102.2.31	3	0,73613
15.	147.102.1.51	2	0,5916
16.	147.102.4.11	1	0,56672
17.	147.102.3.11	4	0,51913
18.	147.102.1.52	2	0,46118
19.	147.102.2.31	2	0,44034
20.	147.102.3.12	1	0,39664
21.	147.102.3.11	3	0,3364
22.	147.102.3.11	2	0,30494
23.	147.102.1.51	1	0,2958
24.	147.102.1.52	1	0,28101
25.	147.102.2.31	1	0,16975
26.	147.102.3.11	1	0,12222

A connection with the following details is being requested from the nearest network entity:

Access Technology = 3      Grade = 2.1092436974789917  
IP Address = 147.102.3.12    Provider Code = 6  
QoS Level = 5                Service Code = 5  
The terminal's request has been ACCEPTED!

---

▪ Παρατηρήσεις:

Στο σημείο αυτό, γίνεται αντιληπτή η εμφάνιση νέων σημείων πρόσβασης, οπότε ο αλγόριθμος εκκινείται για τις δύο τρέχουσες υπηρεσίες, Web Browsing και Video Call. Και για τις δύο, αποφασίζει την ανάθεσή τους σε δίκτυο WLAN και στο μέγιστο επίπεδο ποιότητας (επίπεδο 5). Είναι αξιοσημείωτο ότι, αν και υπήρχαν στην περιοχή δύο σημεία πρόσβασης που ανήκαν στον προτιμώμενο από το χρήστη πάροχο, κανένα από αυτά τα σημεία δεν επελέγη, καθώς η προτίμηση του χρήστη και ο καθορισμός του παράγοντα «πάροχος» στην τρίτη θέση δεν ήταν αρκετά προκειμένου να ξεπεράσουν το βελτιωμένο παράγοντα «ποιότητα – κόστος» του σημείου πρόσβασης που τελικά επελέγη.

#### 6.4 Σύνοψη

Στο κεφάλαιο αυτό, παρουσιάστηκε το πολύ ενδιαφέρον θέμα των αποτελεσμάτων στα οποία καταλήγει ο αλγόριθμος που κατασκευάστηκε, στα πλαίσια τυπικών σεναρίων. Περιγράφηκαν δύο σεναρία, στα οποία αναλύθηκε και αιτιολογήθηκε η έξοδος του αλγορίθμου για κάθε μία φάση τους.

Ειδικότερα, είδαμε την επιλογή ανάθεσης που προτείνει, κάθε φορά, ως επικρατέστερη ο αλγόριθμος, δεδομένων των συνθηκών δικτύου, των προτιμήσεων του χρήστη και της αιτούμενης υπηρεσίας. Είδαμε, ακόμη, πότε ο αλγόριθμος ενεργοποιείται και πότε όχι, αλλά και τον τρόπο με τον οποίο οι προτιμήσεις του χρήστη επηρεάζουν το τελικό αποτέλεσμα.

Γενικά, είχαμε την ευκαιρία να δούμε τον αλγόριθμο «σε δράση» για όσο το δυνατόν περισσότερες διαφορετικές περιπτώσεις επιλογής δικτύου πρόσβασης και επιπέδου ποιότητας. Στο επόμενο κεφάλαιο, θα συνοψισθούν και θα συζητηθούν ακόμη εκτενέστερα τα συμπεράσματα που προκύπτουν από την εφαρμογή του αλγορίθμου σε διάφορες περιστάσεις της καθημερινής ζωής.

## 7. Συμπεράσματα και Επεκτάσεις

### 7.1 Ανακεφαλαίωση

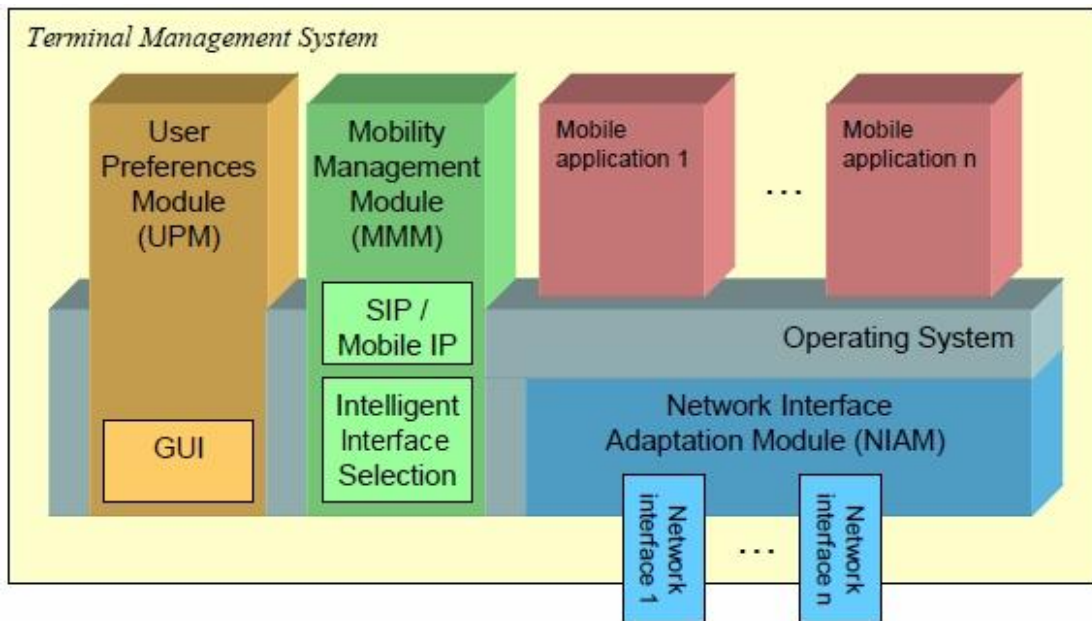
Στην παρούσα εργασία, ασχοληθήκαμε με τα συστήματα 4<sup>ης</sup> Γενιάς και, ιδιαίτερα, με το σύστημα διαχείρισης ενός 4G τερματικού, τόσο από θεωρητικής όσο και από πρακτικής σκοπιάς. Εξηγήσαμε ότι η ύπαρξη πολλών ετερογενών ασύρματων δικτύων πρόσβασης οδηγεί στην ανάγκη της ενιαίας αξιοποίησής τους. Κάτι τέτοιο μπορεί να επιτευχθεί ευκολότερα με την «αποκέντρωση» λειτουργιών από την πλευρά του δικτύου προς την πλευρά των τερματικών.

Μία βασικότερη τέτοια λειτουργία είναι η επιλογή του δικτύου πρόσβασης από το οποίο το τερματικό θα λάβει μια υπηρεσία. Η επιλογή αυτή δεν είναι μια απλή διαδικασία, αφού πρέπει να γίνει με γνώμονα τις συνθήκες του δικτύου τη δεδομένη στιγμή αλλά και τις προτιμήσεις του χρήστη. Η επιλογή αυτή πρέπει, με άλλα λόγια, να είναι η βέλτιστη, τόσο από πλευράς ποιότητας όσο και από πλευράς κόστους.

Το ποιο δίκτυο πρόσβασης (π.χ. GSM, UMTS, WLAN, DVB, κ.λπ.) θα χρησιμοποιηθεί για την εξυπηρέτηση μιας υπηρεσίας δεν αποτελεί τη μοναδική απόφαση που πρέπει να λάβει ένα τερματικό. Η πολλαπλότητα των επιπέδων ποιότητας στα οποία μπορούν να διατεθούν οι περισσότερες υπηρεσίες οδηγεί στην ανάγκη βέλτιστης επιλογής επιπέδου ποιότητας υπηρεσίας (QoS). Συνεπώς, η απόφαση που πρέπει να λάβει το κινητό τερματικό έχει δύο συνιστώσες: Πρώτον, την επιλογή του δικτύου και, ειδικότερα, του σημείου πρόσβασης που θα χρησιμοποιηθεί και, δεύτερον, την επιλογή του επιπέδου ποιότητας στο οποίο θα διατεθεί η υπηρεσία.

Η διαδικασία της επιλογής πρέπει να είναι διαφανής προς το χρήστη, κάτι που σημαίνει ότι δεν πρέπει να απαιτεί την παρέμβασή του, αφού, σε αντίθετη περίπτωση, θα ήταν χρονοβόρα και ενοχλητική. Από την άλλη, πρέπει να λαμβάνει υπόψη της και να ικανοποιεί στο ακέραιο τις προτιμήσεις του. Αυτό σημαίνει ότι είναι αναγκαία η ύπαρξη ενός εύχρηστου και αποτελεσματικού τρόπου εισαγωγής, επεξεργασίας και αποθήκευσης των προτιμήσεων των χρηστών στα κινητά τερματικά.

Για το λόγο αυτό, η αρχιτεκτονική του προτεινόμενου Συστήματος Διαχείρισης Τερματικού (Terminal Management System – TMS) αποτελείται από δύο βασικές συνιστώσες: Πρώτον, τη Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης (Intelligent Interface Selection module – IIS), η οποία είναι υπεύθυνη για τον ανεξάρτητο καθορισμό της βέλτιστης τεχνολογίας και σημείου πρόσβασης από τα οποία θα λαμβάνονται οι υπηρεσίες που ο χρήστης εκκινεί ή ήδη χρησιμοποιεί, με τον καλύτερο δυνατό τρόπο. Η προαναφερθείσα διαδικασία βελτιστοποίησης λαμβάνει υπόψη της μετρήσεις από τα σημεία πρόσβασης, που αντανακλούν την κατάσταση και τη διαθεσιμότητα πόρων του δικτύου, τις προτιμήσεις του χρήστη, καθώς και τις προδιαγραφές των υπηρεσιών. Δεύτερον, τη Συνιστώσα Προτιμήσεων Χρήστη (User Preferences module – UP), που πρόκειται για μια γραφική διεπαφή (GUI) μέσω της οποίας ο χρήστης μπορεί να καθορίζει, να αποθηκεύει και να τροποποιεί δυναμικά τις προτιμήσεις του. Η γραφική αυτή διεπαφή δίνει τη δυνατότητα στο χρήστη να προσδίδει διαφορετική βαρύτητα σε παραμέτρους που μπορεί να επηρεάζουν τη διαδικασία επιλογής δικτύου πρόσβασης, όπως είναι η «ποιότητα», ο «πάροχος» και το «κόστος».



Το παραπάνω σχήμα συνοψίζει την αρχιτεκτονική του συστήματος διαχείρισης ενός τερματικού 4<sup>ης</sup> Γενιάς. Όπως παρατηρούμε, εκτός του IIS και του UP module, υπάρχει η Συνιστώσα Προσαρμογής στους Οδηγούς Δικτύου (Interface Adaptation module – IA), καθώς και η Συνιστώσα Διαχείρισης Κινητικότητας (Mobility Manager module). Η πρώτη είναι υπεύθυνη για την οργάνωση των μετρήσεων που συλλέγει το τερματικό, μέσω των διεπαφών δικτύου που διαθέτει και χρησιμοποιώντας κατάλληλα πρωτόκολλα, από τα σημεία πρόσβασης που βρίσκονται γύρω του, καθώς και για τον έλεγχο εμφάνισης νέων σημείων πρόσβασης ή έντονης υποβάθμισης του λαμβανόμενου σήματος των χρησιμοποιούμενων σημείων πρόσβασης. Η δεύτερη είναι η συνιστώσα που εμπεριέχει το IIS module. Πέραν αυτού, έχει την ευθύνη της συνεργασίας με τα υπόλοιπα modules του συστήματος, της επικοινωνίας με το δίκτυο για την έγκριση των αποφάσεων ανάθεσης και της διατήρησης μιας λίστας με τις τρέχουσες υπηρεσίες.

Τα πλεονεκτήματα ενσωμάτωσης της λειτουργίας της επιλογής δικτύου πρόσβασης στο σύστημα διαχείρισης ενός τερματικού είναι πολλά και έχουν αναλυθεί στο κεφάλαιο 1 (παράγραφος 1.3.2). Θα μπορούσαμε, συνοπτικά, να αναφέρουμε ότι η μετάβαση της συγκεκριμένης λειτουργίας από την πλευρά του δικτύου στην πλευρά του τερματικού εννοείται από το ότι το τερματικό είναι σε θέση να γνωρίζει καλύτερα τόσο τα δικά του χαρακτηριστικά (δικτυακά interfaces και δυνατότητες που διαθέτει, υπηρεσίες που τρέχει) και τις προτιμήσεις του χρήστη (ο χρήστης εύκολα γνωστοποιεί τις προτιμήσεις του στο τερματικό, μέσω του UP module) όσο και τα χαρακτηριστικά του δικτυακού του περιβάλλοντος, ανά πάσα στιγμή. Επιπρόσθετα, μια τέτοια μετάβαση διαφυλάσσει την «αυτονομία» των χρηστών και περιορίζει το υπολογιστικό φορτίο που επωμίζονται οι οντότητες του δικτύου.

Στα πλαίσια της παρούσας εργασίας, υλοποιήθηκε σε Java το σύστημα διαχείρισης ενός τερματικού και παρουσιάστηκαν τα αποτελέσματα εκτέλεσης της εφαρμογής, βάσει συγκεκριμένων σεναρίων. Η συγκεκριμένη υλοποίηση παρατίθεται ολόκληρη στο παράρτημα της εργασίας.



Στο κεφάλαιο 1, δόθηκε το θεωρητικό υπόβαθρο που σχετίζεται με την παρούσα εργασία. Ιδιαίτερη έμφαση δόθηκε στην παρουσίαση της ιστορικής εξέλιξης των ασύρματων τεχνολογιών πρόσβασης, στην εισαγωγή και ανάλυση της ιδέας ABC (Always Best Connected) για τα συστήματα 4<sup>ης</sup> Γενιάς και στην περιγραφή των απαραίτητων δικτυακών και τερματικών υποδομών των 4G συστημάτων. Ειδικά σταθήκαμε στην ιδέα ABC, την οποία και υλοποιεί το προτεινόμενο σύστημα διαχείρισης τερματικού, σύμφωνα με την οποία ένας χρήστης πρέπει να είναι διαρκώς συνδεδεμένος με τον καλύτερο δυνατό τρόπο, συνδυάζοντας και αξιοποιώντας τα πλεονεκτήματα όλων των διαθέσιμων τεχνολογιών πρόσβασης.

Στο κεφάλαιο 2, περιγράφηκε η προτεινόμενη δομή του συστήματος διαχείρισης ενός 4G τερματικού. Ιδιαίτερη έμφαση δόθηκε στο ρόλο και τις βασικές λειτουργίες που επιτελεί η κάθε συνιστώσα του συστήματος.

Στο κεφάλαιο 3, περιγράφηκε αναλυτικά η Συνιστώσα Προτιμήσεων Χρήστη του συστήματος, με ταυτόχρονη παρουσίαση της γραφικής διεπαφής της εφαρμογής που υλοποιήθηκε. Στο κεφάλαιο αυτό, είχαμε την ευκαιρία να δούμε τη μορφή των περισσότερων μενού και παραθύρων της εφαρμογής, αλλά και να εξηγήσουμε τον τρόπο με τον οποίο μπορεί ένας χρήστης, απλά και γρήγορα, να διαμορφώσει το προφίλ του και να το καταχωρήσει στο τερματικό.

Στο κεφάλαιο 4, ασχοληθήκαμε εκτενώς με τη Συνιστώσα Ευφυούς Επιλογής Δικτύου Πρόσβασης. Διατυπώσαμε επακριβώς το πρόβλημα, που, όπως είδαμε, πρόκειται ουσιαστικά για ένα πρόβλημα βελτιστοποίησης, και παραθέσαμε, βήμα προς βήμα, τόσο τη μαθηματική όσο και την αλγοριθμική επίλυσή του. Αναφερθήκαμε σε διάφορες παραμέτρους του προβλήματος, όπως, για παράδειγμα, είναι η ανάκτηση πληροφοριών για το κόστος των υπηρεσιών, και για το πώς αυτές μπορούν να αντιμετωπισθούν.

Στο κεφάλαιο 5, ασχοληθήκαμε με περισσότερο ειδικά θέματα της υλοποίησης, δίνοντας τεχνικές λεπτομέρειες και μια εποπτική εικόνα του τρόπου υλοποίησης. Στο κεφάλαιο αυτό, είχαμε την ευκαιρία να δούμε ποια είναι τα βασικότερα πακέτα και οι κυριότερες κλάσεις που απαρτίζουν την εφαρμογή.

Στο κεφάλαιο 6, προχωρήσαμε στην επίδειξη των αποτελεσμάτων από την εκτέλεση της εφαρμογής. Έγινε προσομοίωση δύο τυπικών σεναρίων και σχολιάστηκαν τα αποτελέσματα του αλγορίθμου βελτιστοποίησης.

## **7.2 Συμπεράσματα Λειτουργίας του Αλγορίθμου**

Ο τρόπος με τον οποίο σχεδιάστηκε το προτεινόμενο σύστημα διαχείρισης ενός 4G τερματικού, αλλά και τα αποτελέσματα της εκτέλεσής του, τα οποία παρακολούθησαμε στο κεφάλαιο 6, μας οδηγούν σε συγκεκριμένα συμπεράσματα αναφορικά με τη λειτουργία του αλγορίθμου βελτιστοποίησης. Τα συμπεράσματα αυτά θα επιχειρήσουμε να συνοψίσουμε στα επόμενα.

- Ο αλγόριθμος βελτιστοποίησης αναζητά μεταξύ των δυνατών επιλογών ανάθεσης, την τρέχουσα χρονική στιγμή, την καλύτερη δυνατή. Για το σκοπό αυτό, βαθμολογεί καθεμιά από τις επιλογές αυτές και η επιλογή (σημείο πρόσβασης και επίπεδο ποιότητας) με τη μεγαλύτερη βαθμολογία είναι εκείνη που τελικώς επικρατεί.
- Στη διαμόρφωση της βαθμολογίας μιας επιλογής ανάθεσης, δεν παίζει ρόλο μόνο η ποιότητά της ή μόνο το κόστος της, αλλά ο συνδυασμένος παράγοντας ποιότητας – κόστους. Επομένως, πρέπει να υπάρχει μια ισορροπημένη σχέση μεταξύ της προσφερόμενης ποιότητας και του αντίστοιχου κόστους, προκειμένου μια επιλογή να θεωρηθεί τελικώς επικρατέστερη.
- Ειδικά για τον παράγοντα της ποιότητας, δεν παίζει ρόλο μόνο το ονομαστικό επίπεδο ποιότητας μιας επιλογής, αλλά και το επίπεδο ισχύος του λαμβανόμενου σήματος. Με άλλα λόγια, μπορεί ένα σημείο πρόσβασης να διαθέτει το απαιτούμενο εύρος ζώνης, ώστε να είναι σε θέση να προσφέρει ένα συγκεκριμένο επίπεδο ποιότητας, ωστόσο, αν το σήμα που λαμβάνεται από αυτό το σημείο πρόσβασης είναι πολύ αδύναμο, τότε υστερεί έναντι των υπολοίπων και αυτό λαμβάνεται υπόψη από τον αλγόριθμο βελτιστοποίησης. Κάτι τέτοιο είναι λογικό, αφού χαμηλή στάθμη σήματος σημαίνει και αυξημένη πιθανότητα να χρειαστεί να γίνει διαπομπή στο άμεσο μέλλον. Επομένως, καλόν είναι να επιλεγεί εξαρχής ένα πιο αξιόπιστο σημείο πρόσβασης, ώστε να ελαττωθεί το ενδεχόμενο μιας διαπομπής στο κοντινό μέλλον.
- Ο τρόπος με τον οποίο συμμετέχουν η «ποιότητα» και το «κόστος» στο συνδυασμένο παράγοντα ποιότητας – κόστους δεν είναι, φυσικά, ισοβαρής, αλλά εξαρτάται από τις προτιμήσεις του χρήστη. Συγκεκριμένα, ο χρήστης, κατατάσσοντας στο προφίλ του τους παράγοντες «ποιότητα», «κόστος» και «πάροχος», τους αποδίδει και την αντίστοιχη σχετική βαρύτητα. Ο συντελεστής βαρύτητας ενός παράγοντα είναι μεγαλύτερος όταν ο παράγοντας είναι τοποθετημένος υψηλότερα στην ιεραρχημένη λίστα.
- Τα σημεία πρόσβασης των οποίων ο πάροχος ταυτίζεται με τον προτιμώμενο από το χρήστη πάροχο λαμβάνουν ένα «πριμ» κατά τη βαθμολόγησή τους. Έτσι, ο χρήστης έχει τη δυνατότητα να καθορίζει ποιος είναι ο πάροχος που προτιμά ή εμπιστεύεται περισσότερο. Το ύψος της πριμοδότησης εξαρτάται από το πόσο ψηλά είναι τοποθετημένος ο παράγοντας πάροχος στην ιεραρχημένη λίστα παραγόντων και πρόκειται για ένα ποσοστό της απόλυτης τιμής του συνδυασμένου παράγοντα «ποιότητα – κόστος».
- Η διαδικασία βελτιστοποίησης λαμβάνει υπόψη της το σύνολο των προτιμήσεων του χρήστη. Κατά συνέπεια, είναι πιθανό ένα σημείο πρόσβασης το οποίο ανήκει στον προτιμώμενο πάροχο τελικά να μην επιλεγεί. Αυτό θα συμβαίνει επειδή ο συνδυασμένος παράγοντας ποιότητας – κόστους κάποιας άλλης επιλογής θα υπερκαλύπτει την πριμοδοτούμενη βαθμολογία του συγκεκριμένου σημείου πρόσβασης. Αυτό είναι σωστό και αναμενόμενο, αφού όλοι οι χρήστες ενδιαφέρονται για όσο το δυνατόν υψηλότερη ποιότητα με όσο το δυνατό χαμηλότερο κόστος και αυτό είναι κάτι που το εξασφαλίζει σε κάθε περίπτωση η αντικειμενική συνάρτηση. Σίγουρα δε θα ήταν προς το συμφέρον του χρήστη να

επιλεγεί μια λιγότερο συμφέρουσα ανάθεση μόνο επειδή ανήκει στον προτιμώμενο πάροχο.

- Σε περίπτωση που ο χρήστης δεν είναι απόλυτα ικανοποιημένος με τις επιλογές του τερματικού, μπορεί να προβεί σε αλλαγή του ενεργού προφίλ του. Ένας εύκολος τρόπος για να γίνει αυτό είναι μέσω της φόρτωσης ενός από τα τέσσερα διαθέσιμα προφίλ που είδαμε στο κεφάλαιο 3. Επίσης, μπορεί να προβεί σε οποιαδήποτε τροποποίηση, ανά πάσα στιγμή, και να την αποθηκεύσει. Η διαδικασία βελτιστοποίησης θα ξανατρέξει, προκειμένου να ικανοποιήσει τις νέες του προτιμήσεις.
- Είναι ιδιαίτερα σημαντική η κατάταξη των παραγόντων που θα επιλέξει ο χρήστης στο προφίλ του. Έτσι, για παράδειγμα, αν η «ποιότητα» είναι για αυτόν η σημαντικότερη παράμετρος, τότε πρέπει να την τοποθετήσει πρώτη. Τοποθετώντας το «κόστος» στην επόμενη θέση, δηλώνει ότι είναι λιγότερο σημαντικός παράγοντας σε σχέση με την «ποιότητα», ωστόσο εξακολουθεί να έχει ένα σημαντικό συντελεστή βαρύτητας. Επομένως, σε περίπτωση που το «κόστος» δεν τον ενδιαφέρει ιδιαίτερα ως παράμετρος της βελτιστοποίησης, θα πρέπει να τον τοποθετήσει στην τρίτη και όχι στη δεύτερη θέση. Αυτό ισχύει, βεβαίως, για οποιονδήποτε παράγοντα. Όπως γίνεται, λοιπόν, φανερό, το να επιλέξει ο χρήστης τη ρύθμιση που πραγματικά του καλύπτει τις ανάγκες είναι ιδιαίτερα σημαντικό. Για παράδειγμα, άλλες ρυθμίσεις θα πρέπει πιθανότατα να επιλέξει για το «εταιρικό» του προφίλ (το λογαριασμό πληρώνει η εταιρεία) και άλλες για το «προσωπικό» του (το λογαριασμό πληρώνει ο ίδιος). Για το πρώτο, προτεραιότητα θα έχει η «ποιότητα» (με το «κόστος» στη δεύτερη ή στην τρίτη θέση), ενώ το αντίστροφο θα πρέπει να ισχύει για το δεύτερο.
- Ο αλγόριθμος βελτιστοποίησης δεν εξετάζει εκείνες τις περιπτώσεις ανάθεσης που αντιστοιχούν σε επίπεδα ποιότητας μεγαλύτερα εκείνου που ο χρήστης έχει καθορίσει, μέσα από το προφίλ του, ως το μέγιστο επιτρεπτό. Έτσι, εξασφαλίζεται ότι το κόστος λήψης μιας υπηρεσίας δε θα υπερβεί το όριο που επιθυμεί ο χρήστης.
- Από την άλλη, είναι πολύ πιθανό ένα σημείο πρόσβασης να μην μπορεί να υποστηρίξει όλα τα επίπεδα ποιότητας, μέχρι το μέγιστο επιτρεπτό, λόγω περιορισμένου εύρους ζώνης. Το περιορισμένο αυτό εύρος ζώνης μπορεί να οφείλεται στις δυνατότητες που έχει η τεχνολογία πρόσβασης στην οποία ανήκει το συγκεκριμένο πρόσβασης ή στο μεγάλο φορτίο του τη δεδομένη χρονική στιγμή.
- Όπως έχουμε εξηγήσει, ο αλγόριθμος βελτιστοποίησης εκκινείται σε συγκεκριμένες περιπτώσεις (παράγραφος 4.1.9). Οι περιπτώσεις αυτές είναι, συνοπτικά, η αίτηση μιας καινούριας υπηρεσίας, η τροποποίηση του προφίλ χρήστη, η εμφάνιση καινούριου σημείου πρόσβασης ή η έντονη υποβάθμιση του λαμβανόμενου σήματος ενός από τα χρησιμοποιούμενα σημεία πρόσβασης. Εφόσον κάποιος από τους παραπάνω λόγους δε συντρέχει, τότε ο αλγόριθμος βελτιστοποίησης δεν εκκινείται, αφού οι υπηρεσίες συνεχίζουν να λαμβάνονται κατά τον καλύτερο δυνατό τρόπο. Στην περίπτωση που έχουμε μια μικρή υποβάθμιση του λαμβανόμενου σήματος, αλλά όχι σε σημείο που το σημείο πρόσβασης να χάνεται από την εμβέλεια του τερματικού, η λήψη της υπηρεσίας

συνεχίζεται από το ίδιο σημείο πρόσβασης, διότι δεν υπάρχει λόγος να επιβαρυνθεί το σύστημα με μία επιπλέον διαπομπή, εφόσον η ισχύς του σήματος επαρκεί για τη λήψη της υπηρεσίας.

### 7.3 Επεκτάσεις της Εργασίας

#### 7.3.1 Εισαγωγή

Κλείνοντας το τελευταίο αυτό κεφάλαιο, αξίζει να εξετάσουμε, εν συντομία, ορισμένα θέματα που θα μπορούσαν να διερευνηθούν περαιτέρω, ως επεκτάσεις της παρούσας εργασίας. Τα θέματα αυτά ποικίλλουν και κυμαίνονται από απλές προσθήκες ή τροποποιήσεις της εργασίας αυτής μέχρι περισσότερο ουσιώδεις επεκτάσεις. Στις παραγράφους που ακολουθούν, θα ασχοληθούμε με τα κυριότερα από αυτά τα θέματα.

#### 7.3.2 Ενσωμάτωση της Τεχνολογίας Πρόσβασης ως Παράγοντα του Προφίλ Χρήστη

Όπως έχουμε δει, οι τρεις βασικοί παράγοντες που λαμβάνονται υπόψη από την αντικειμενική συνάρτηση βελτιστοποίησης είναι η «ποιότητα», το «κόστος» και ο «πάροχος». Στην παράγραφο 4.2.2, ορίσαμε, μάλιστα την αντικειμενική συνάρτηση ως εξής:

$$\begin{aligned} \mathbf{Objective\_Function}(p,q) = & \mathbf{qualityFactor} \times \mathbf{RealQuality}(p,q) - \\ & \mathbf{costFactor} \times \mathbf{Cost}(p,q) + \\ & \mathbf{providerFactor} \times \mathbf{Provider}(p) \end{aligned}$$

όπου  $p \in P$  και  $q \in Q(p)$ .

Σε μία λίγο γενικότερη προσέγγιση, θα μπορούσε να συγκαταλέγεται μεταξύ των παραμέτρων που παίζουν ρόλο στη διαδικασία βελτιστοποίησης και η «τεχνολογία πρόσβασης». Με άλλα λόγια, θα μπορούσε η αντικειμενική συνάρτηση να συμπληρωθεί ως εξής:

$$\begin{aligned} \mathbf{Objective\_Function}(p,q) = & \mathbf{qualityFactor} \times \mathbf{RealQuality}(p,q) - \\ & \mathbf{costFactor} \times \mathbf{Cost}(p,q) + \\ & \mathbf{providerFactor} \times \mathbf{Provider}(p) + \\ & \mathbf{technologyFactor} \times \mathbf{Technology}(p) \end{aligned}$$

όπου **technologyFactor** είναι ο συντελεστής βαρύτητας του «καινούριου» παράγοντα «τεχνολογία πρόσβασης». Ο δε ορισμός του παράγοντα **Technology(p)** μπορεί να είναι παρόμοιος με εκείνον του παράγοντα **Provider(p)** (παράγραφος 4.2.3):

$$Technology(p) = \begin{cases} 0, & \text{αν η τεχνολογία του access point } p \\ & \text{δε συμπίπτει με την προτιμώμενη} \\ 0.5 \times Abs[qualityFactor \times RealQuality(p, q) - costFactor \times Cost(p, q)], & \\ & \text{αν συμπίπτει} \end{cases}$$

Με άλλα λόγια, στην περίπτωση αυτή, ο χρήστης θα επιλέγει στο προφίλ του την τεχνολογία πρόσβασης που προτιμά (π.χ. WLAN), όπως ακριβώς συμβαίνει και με την επιλογή του προτιμώμενου παρόχου. Σε περίπτωση που η τεχνολογία την οποία υποστηρίζει ένα υποψήφιο access point είναι ίδια με την προτιμώμενη από το χρήστη τεχνολογία πρόσβασης, τότε η αντίστοιχη επιλογή ανάθεσης θα προμοδοτείται κατά ένα ποσοστό.

Η προσέγγιση αυτή έχει το πλεονέκτημα ότι είναι γενικότερη από την αρχική και δίνει στο χρήστη τη δυνατότητα να καθορίσει, εφόσον το επιθυμεί, την τεχνολογία πρόσβασης που προτιμά. Από την άλλη, η αρχική προσέγγιση έχει το πλεονέκτημα ότι θεωρεί και καθιστά την επιλογή της κατάλληλης τεχνολογίας πρόσβασης πιο διαφανή προς το χρήστη. Έτσι, αναλαμβάνει το ίδιο το σύστημα διαχείρισης να βρει τη βέλτιστη τεχνολογία πρόσβασης χωρίς να ρωτά το χρήστη για το ποια θεωρεί εκείνος ότι είναι η καλύτερη. Είναι πολύ πιθανό, επίσης, ο χρήστης να μην είναι εξοικειωμένος με τα ειδικά χαρακτηριστικά της κάθε τεχνολογίας πρόσβασης και άρα να μη γνωρίζει τι να επιλέξει. Εξάλλου, οι χρήστες ενδιαφέρονται συνήθως για το τελικό αποτέλεσμα, δηλαδή τη βέλτιστη λήψη μιας υπηρεσίας, από πλευράς ποιότητας και κόστους, και όχι για το κάτω από ποια τεχνολογία πρόσβασης είναι εφικτή η λήψη αυτή.

### 7.3.3 Τροποποίηση της Μεθόδου Κανονικοποίησης του Κόστους

Όπως είδαμε στην παράγραφο 4.2.3 (Στάδιο προσδιορισμού του παράγοντα «κόστος»), οι τιμές του - ανά μονάδα χρόνου ή ανά μονάδα όγκου δεδομένων - κόστους των υπηρεσιών κανονικοποιούνται, προκειμένου να καταστούν συγκρίσιμες με τις τιμές του παράγοντα «ποιότητα», βάσει του τύπου:

$$Cost(p, q) = \frac{noQoSLevels}{\max Cost} \times actualCost(p, q)$$

όπου **maxCost** το μέγιστο κόστος που εμφανίζεται σε μια υπηρεσία, **noQoSLevels** ο συνολικός αριθμός διαθέσιμων επιπέδων ποιότητας για αυτήν την υπηρεσία, **Cost(p,q)** η κανονικοποιημένη τιμή του κόστους και **actualCost(p,q)**, η ανά μονάδα χρόνου ή όγκου τιμή του κόστους, με  $p \in P$  και  $q \in Q(p)$ .

Μια διαφορετική προσέγγιση θα ήταν το maxCost να μην αντιπροσώπευε το μέγιστο (ανά μονάδα) κόστος παροχής μιας υπηρεσίας, αλλά το μέγιστο (ανά μονάδα) κόστος μεταξύ των υποψήφιων επιλογών ανάθεσης τη δεδομένη στιγμή. Με άλλα λόγια, να εξετάζεται ποιο είναι το μέγιστο κόστος από τις διαθέσιμες επιλογές ανάθεσης και όχι ποιο είναι γενικά το μέγιστο κόστος και αυτό να χρησιμοποιείται ως τιμή του maxCost. Επίσης, θα μπορούσε ως τιμή του noQoSLevels να χρησιμοποιείται ο

αριθμός των επιτρεπτών (βάσει του προφίλ του χρήστη) επιπέδων ποιότητας και όχι ο συνολικός τους αριθμός.

Τόσο η μία όσο και η άλλη προσέγγιση είναι σωστές. Είναι θέμα της εκάστοτε υλοποίησης να επιλέξει μία εκ των δύο.

#### 7.3.4 Το Σύνολο των Τρεχουσών Υπηρεσιών ως Είσοδος του Αλγορίθμου

Όπως έχουμε τονίσει πολλές φορές, οι εισοδοί του αλγορίθμου βελτιστοποίησης είναι οι προτιμήσεις (προφίλ) του χρήστη, οι συνθήκες του δικτυακού περιβάλλοντος του τερματικού και η αιτούμενη υπηρεσία (παράγραφος 4.1.5). Η δε μαθηματική διατύπωση του προβλήματος διαμορφώνεται ως εξής (παράγραφος 4.2.2):

$$\begin{aligned} \mathbf{Objective\_Function}(p, q) = & \mathbf{qualityFactor} \times \mathbf{RealQuality}(p, q) - \\ & \mathbf{costFactor} \times \mathbf{Cost}(p, q) + \\ & \mathbf{providerFactor} \times \mathbf{Provider}(p) \end{aligned}$$

$$\mathbf{BestChoice} = \max_{\forall p \in P} \left\{ \max_{\forall q \in Q(p)} \{ \mathbf{Objective\_Function}(p, q) \} \right\}$$

Επεκτείνοντας την παραπάνω ιδέα, θα μπορούσαμε να θεωρήσουμε ως είσοδο του αλγορίθμου όχι την αιτούμενη υπηρεσία αλλά το σύνολο των τρεχουσών / αιτούμενων υπηρεσιών. Ως εκ τούτου, ο αλγόριθμος θα έπρεπε να αναζητήσει τη βέλτιστη κατανομή όλων των υπηρεσιών στα διαθέσιμα access points και στα επιτρεπτά επίπεδα ποιότητας.

Η εύρεση της βέλτιστης αυτής κατανομής είναι ένα πρόβλημα που μπορεί να λυθεί με εξαντλητική εξέταση των δυνατών επιλογών ανάθεσης. Η μαθηματική διατύπωση του προβλήματος τροποποιείται ως εξής:

- Έστω  $\mathbf{S}$  το σύνολο των τρεχουσών / αιτούμενων υπηρεσιών στο τερματικό:  

$$\mathbf{S} = \{s_1, s_2, \dots, s_k\}, k \in N$$
- Έστω  $\mathbf{P}$  το σύνολο των υποψήφιων σημείων πρόσβασης (access points), που «βλέπει» το τερματικό:  

$$\mathbf{P} = \{p_1, p_2, \dots, p_n\}, n \in N$$
- Έστω  $\mathbf{Q}(p)$  το σύνολο των επιπέδων ποιότητας στα οποία μπορεί να προσφέρει την υπό εξέταση υπηρεσία το σημείο πρόσβασης  $p$ :  

$$\mathbf{Q}(p, s) = \{q_1, q_2, \dots, q_m\}, m \in N, p \in P, s \in S$$
- *Σημείωση:* Κάθε  $p \in P$  έχει, σε πρώτη φάση, διαθέσιμο εύρος ζώνης,  $\mathbf{available\_bandwidth}(p)$ , όσο θα είχε χωρίς να τρέχουν οι  $s_1, s_2, \dots, s_k$ .
- *Μαθηματική Διατύπωση:*

$$MaxGrade(S, P) = \max_{\forall s \in S} \left\{ \max_{\forall p \in P} \left\{ \max_{\forall q \in Q(p,s)} \{ OF(p, q, s) + MaxGrade(S - \{s\}, P') \} \right\} \right\}$$

όπου  $P'$  είναι το  $P$  τροποποιημένο, έτσι ώστε, αντί του  $p$ , να εμπεριέχει το  $p'$ . Το  $p' \in P$  είναι τέτοιο, ώστε το διαθέσιμο εύρος ζώνης του να είναι ελαττωμένο κατά  $bandwidth(s, q)$ , δηλαδή:

$$available\_bandwidth(p') = available\_bandwidth(p) - bandwidth(s, q)$$

- *Σημείωση:* Με  $OF(p, q, s)$  συμβολίζεται η τιμή της αντικειμενικής συνάρτησης, της οποίας η μορφή είναι σχεδόν όμοια:

$$OF(p, q, s) = qualityFactor \times RealQuality(p, q, s) - \\ costFactor \times Cost(p, q, s) + \\ providerFactor \times Provider(p)$$

Η παραπάνω προσέγγιση έχει το πλεονέκτημα ότι, ανά πάσα στιγμή, οι υπηρεσίες ενός τερματικού έχουν καταναμηθεί κατά το βέλτιστο δυνατό τρόπο. Έτσι, όταν γίνεται μια καινούρια αίτηση, είναι πολύ πιθανό να γίνει ανακατανομή των υπηρεσιών που ήδη τρέχουν (π.χ. ανάθεσή τους σε διαφορετικά σημεία πρόσβασης ή επίπεδα ποιότητας), με στόχο να επιτευχθεί η καλύτερη δυνατή λήψη του συνόλου των υπηρεσιών. Από την άλλη, βέβαια, υπάρχουν και σοβαρά μειονεκτήματα: Υστερεί σε ταχύτητα έναντι της αρχικής προσέγγισης, ενώ, επίσης, επιβαρύνει το σύστημα με πολύ περισσότερες διαπομπές. Είναι χαρακτηριστικό ότι, όποτε ο χρήστης ζητά την έναρξη μιας καινούριας υπηρεσίας, οι υπηρεσίες που ήδη τρέχουν είναι πιθανό να επηρεάζονται και να αλλάζουν τρόπο λήψης.

### 7.3.5 Αυτόματος Καθορισμός Συντελεστών Βαρύτητας

Η σημαντικότερη, ίσως, επέκταση της παρούσας εργασίας εστιάζεται στον αυτόματο καθορισμό των συντελεστών βαρύτητας που χρησιμοποιεί η αντικειμενική συνάρτηση. Οι τιμές των συντελεστών αυτών εξαρτώνται από τις προτιμήσεις του χρήστη και, συγκεκριμένα, από την ιεράρχηση των παραγόντων «ποιότητα», «κόστος» και «πάροχος» που έχει κάνει ο χρήστης στο προφίλ του.

Η έννοια του αυτόματου καθορισμού των συντελεστών αυτών συνεπάγεται, λοιπόν, τη με αυτόματο τρόπο ιεράρχηση των τριών παραγόντων, κάθε φορά που ενεργοποιείται ο αλγόριθμος βελτιστοποίησης. Η διαδικασία του αυτόματου καθορισμού μπορεί να υλοποιηθεί μέσω ενός ευφυούς μοντέλου το οποίο θα στηρίζεται σε πληροφορίες του πλαισίου χρήσης (context) κατά τη στιγμή της αίτησης. Τέτοιου είδους πληροφορίες είναι οι ακόλουθες:

- Η τοποθεσία όπου βρίσκεται ο χρήστης.
- Η χρονική στιγμή της αίτησης.
- Το είδος της εφαρμογής που εκκινείται.

- Ο συνομιλητής ή παραλήπτης της υπηρεσίας (π.χ. ποιος είναι ο συνδρομητής τον οποίο καλεί ο χρήστης).

Οι πληροφορίες αυτές είναι εύκολο να ανακτηθούν (π.χ. η τοποθεσία μέσω GPS συστήματος, η χρονική στιγμή από το ρολόι του τερματικού, κ.λπ.) και μπορούν να αξιοποιηθούν προκειμένου να αποφασιστεί η πιθανότερη κατάταξη των τριών παραγόντων («ποιότητα», «κόστος» και «πάροχος»), βάσει και των προηγούμενων εμπειριών του συστήματος. Ένα σύστημα εκμάθησης του τερματικού, μέσω της αλληλεπίδρασής του με το χρήστη, θα βοηθά το σύστημα διαχείρισης να μάθει να λαμβάνει τη σωστή απόφαση.



## Βιβλιογραφία

- [1] E. Gustafsson and A. Jonsson, “*Always Best Connected*”, IEEE Wireless Communications, February 2003.
- [2] J. van Bommel, H. Teunissen, D.-J. Plas, A. Peddemors, “*A Reference Architecture for 4G Services*”, In: 7<sup>th</sup> WWRF Meeting in Eindhoven, The Netherlands, December 2002.
- [3] B. Busropan, J. van Loon, F. Vervuurt, R. van Eijk, “*Access Network Selection in Heterogeneous Networks and the Role of the Operator*”, In: 9<sup>th</sup> WWRF Meeting in Zurich, Switzerland, July 2003.
- [4] D. Chantrain, K. Handekyn, H. Vanderstraeten, “*The Soft Terminal: Extending Service Intelligence from the Network to the Terminal*”, Alcatel Telecommunications Review, 2<sup>nd</sup> Quarter 2000.
- [5] M. S. Bargh, H. Zandbelt, A. Peddemors, “*Managing Mobility in 4G Environments with Federating Service Platforms*”, EVOLUTE Workshop, November 2003.
- [6] M. Liebsch, A. Singh (editors), H. Chaskar, D. Funato, E. Shim, “*The Candidate Access Router Discovery Protocol*”, draft-ietf-seamoby-card-protocol-07.txt, work in progress.
- [7] E. Njedjou, P. Bertin, P. Reynolds, “*Motivation for Network Controlled Handoffs using IP Mobility between heterogeneous Wireless Access Networks*”, IETF Internet Draft, draft-njedjou-inter-an-handoffs-00.txt, June 2003, work in progress.
- [8] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, “*SIP: Session Initiation Protocol*”, RFC 3261, June 2002.
- [9] G. Koundourakis, N. Koutsouris, V. Stavroulaki, L. Papadopoulou, V. Tountopoulos, D. Kouis, P. Demestichas, N. Mitrou, “*Network and Service Management System for Optimising Service Delivery and Traffic Distribution in Composite Radio Environments*”, In: Proc. IST Mobile & Wireless Telecommunications Summit 2003, Aveiro, Portugal, June 2003.
- [10] P. Demestichas, N. Koutsouris, G. Koundourakis, K. Tsagkaris, A. Oikonomou, V. Stavroulaki, L. Papadopoulou, M. Theologou, G. Vivier, K. El Khazen, “*Management of Networks and Services in a Composite Radio Context*”, IEEE

Wireless Communications Magazine, August 2003 (Evolution towards 4G Mobile Communication Systems).

- [11] Μ. Ε. Θεολόγου, «*Δίκτυα Κινητών και Προσωπικών Επικοινωνιών*», Έκδοση Εθνικού Μετσόβιου Πολυτεχνείου, Αθήνα, Μάρτιος 2004.
- [12] Α. Κανάτας, Φ. Κωνσταντίνου, «*Συστήματα Κινητών Ραδιοεπικοινωνιών*», Εργαστήριο Κινητών Ραδιοεπικοινωνιών Εθνικού Μετσόβιου Πολυτεχνείου, Αθήνα, 13 Μαρτίου 2001.
- [13] R. Cadenhead, L. Lemay, «*Πλήρες Εγχειρίδιο της Java 2*», Απόδοση στα Ελληνικά: Γ. Β. Σαμαράς, Εκδόσεις Μ. Γκιούρδας, 2003.
- [14] Ε. Καγιάφας, Β. Λούμος, «*Τεχνολογία Πολυμέσων*», Έκδοση Εθνικού Μετσόβιου Πολυτεχνείου, Αθήνα, Ιούνιος 2002.
- [15] <http://www.cellular.co.za/stats/stats-main.htm>
- [16] <http://java.sun.com/developer/JDCTechTips/2001/tt0807.html>
- [17] <http://java.sun.com/docs/books/tutorial/uiswing/misc/icon.html>
- [18] <http://compnetworking.about.com/cs/wireless80211/a/aa80211standard.htm>
- [19] <http://www.gsmworld.com/technology/gprs/intro.shtml>
- [20] <http://computing.breinstorm.net/speech+bitrate+coding+coders+performance/>

## **ΠΑΡΑΡΤΗΜΑ**

### **A. πηγαίος κώδικας και UML διαγράμματα**

(κάθε κλάση του κώδικα συνοδεύεται και από το αντίστοιχο διάγραμμα)

### **B. αρχεία της τράπεζας πληροφοριών**

### **Γ. τεκμηρίωση κώδικα (documentation)**



## A. Πηγαίος Κώδικας και UML Διαγράμματα

```

package diplomatiki ;

/**
 * <p>Title: Term Paper - <code>Config</code> class</p>
5  *
 * <p>Description: This class encapsulates configuration information for the
 * TMS. All variables defined here class are class (static) variables and most
 * of them are also final.</p>
 *
10 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
15 * @version 1.0
 */
public class Config {
    //Base Path
    public final static String INIT_PATH =
20     "diplomatiki/databank/" ;
    //"I:/users/cdemest/diplomatiki/databank/";
    public final static String INIT_ICONS_PATH =
    "diplomatiki/images/" ;
    //"C:/Documents and Settings/Costas/My Documents/" +
25     //"My Pictures/My Icons/xara/";

    //For Progress Bars
    public final static int BAR_MAX_TIME = 3;

30     //For InterfaceAdaptation
    public final static String MEASUREMENTS_PATH =
        INIT_PATH + "measurements.xml" ;

    //For DeteriorationController
35     public final static boolean KILL_DC = true;
    public final static int RANDOM_MAX_NUMBER = 5;
    public final static int DETER_DEGREE = 5;

    //For InterfaceAdaptation, DeteriorationController
40     public final static long INITIAL_SLEEP = 20000;
    public final static long INTERVAL_SLEEP = 20000;

    //For MobilityManager, ApplicationsPanel
    public final static String SERVICES_CONFIG_FILE =
45     INIT_PATH + "services.txt" ;
    public final static String TECHNOLOGIES_FILE =
        INIT_PATH + "technologies.txt" ;

    //For MobilityManager
50     public final static String PROVIDERS_FILE =
        INIT_PATH + "providers.txt" ;
    public final static String NETWORK_ENTITY_ADDRESS = "localhost" ;
    public final static int NETWORK_ENTITY_PORT = 4413;

55     //For IIS
    public final static int REAL_QUALITY_COL = 0;
    public final static int COST_COL = 1;
    public final static int GRADE_COL = 2;
    public final static String SERVICES_FILE =
60     INIT_PATH + "services.xml" ;
    public final static String COSTS_FILE =
        INIT_PATH + "costs.xml" ;
    public final static String FACTORS_FILE =
        INIT_PATH + "factors.txt" ;
65     public final static String SIGNAL_FACTORS_FILE =
        INIT_PATH + "signal_factors.txt" ;
    public final static String PROVIDER_PERCENTAGE_FILE =
        INIT_PATH + "provider_bonus.txt" ;

70     //For EditPanel

```

```

public final static int PROFILE_CURRENT_CODE = 1;
public final static int PROFILE_LOW_COST_CODE = 2;
public final static int PROFILE_HIGH_QUALITY_CODE = 3;
public final static int PROFILE_BALANCED_CODE = 4;
75 public final static String EDIT_CURRENT_ICON_PATH =
    INIT_ICONS_PATH + "currentprofile.jpg" ;
public final static String EDIT_CURRENT_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "currentprofileselected.jpg" ;
public final static String EDIT_LOW_COST_ICON_PATH =
80 INIT_ICONS_PATH + "lowcostprofile.jpg" ;
public final static String EDIT_LOW_COST_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "lowcostprofileselected.jpg" ;
public final static String EDIT_HIGH_QUALITY_ICON_PATH =
    INIT_ICONS_PATH + "highqualityprofile.jpg" ;
85 public final static String EDIT_HIGH_QUALITY_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "highqualityprofileselected.jpg" ;
public final static String EDIT_BALANCED_ICON_PATH =
    INIT_ICONS_PATH + "balancedprofile.jpg" ;
public final static String EDIT_BALANCED_SELECTED_ICON_PATH =
90 INIT_ICONS_PATH + "balancedprofileselected.jpg" ;

//For EmailInputPanel, MMSInputPanel
public final static String LOAD_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "load.jpg" ;
95 public final static String LOAD_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "loadselected.jpg" ;

//For EmailPanel
public final static int EMAIL_SERVICE_CODE = 4;
100 public final static String EMAIL_START_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "sendemail.jpg" ;
public final static String EMAIL_START_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "sendemailselected.jpg" ;

105 //ForGeneralProfileSettingsPanel
public final static String UP_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "up.jpg" ;
public final static String UP_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "upselected.jpg" ;
110 public final static String DOWN_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "down.jpg" ;
public final static String DOWN_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "downselected.jpg" ;

115 //For InfoPanel
public final static String DEFAULT_ANIMATED_ICON_PATH =
    INIT_ICONS_PATH + "pieanimated.gif" ;

//For MenuFrame
120 public final static String JBUTTON1_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "voicecall.jpg" ;
public final static String JBUTTON1_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "voicecallselected.jpg" ;
public final static String JBUTTON2_DEFAULT_ICON_PATH =
125 INIT_ICONS_PATH + "email.jpg" ;
public final static String JBUTTON2_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "emailselected.jpg" ;
public final static String JBUTTON3_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "videostream.jpg" ;
130 public final static String JBUTTON3_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "videostreamselected.jpg" ;
public final static String JBUTTON4_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "sms.jpg" ;
public final static String JBUTTON4_SELECTED_ICON_PATH =
135 INIT_ICONS_PATH + "smsselected.jpg" ;
public final static String JBUTTON5_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "settings.jpg" ;
public final static String JBUTTON5_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "settingsselected.jpg" ;
140 public final static String JBUTTON6_DEFAULT_ICON_PATH =

```

```

        INIT_ICONS_PATH + "webbrowsing.jpg" ;
    public final static String JBUTTON6_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "webbrowsingselected.jpg" ;
    public final static String JBUTTON7_DEFAULT_ICON_PATH =
145     INIT_ICONS_PATH + "mms.jpg" ;
    public final static String JBUTTON7_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "mmsselected.jpg" ;
    public final static String JBUTTON8_DEFAULT_ICON_PATH =
        INIT_ICONS_PATH + "videocall.jpg" ;
150     public final static String JBUTTON8_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "videocallselected.jpg" ;
    public final static String JBUTTON9_DEFAULT_ICON_PATH =
        INIT_ICONS_PATH + "applications.jpg" ;
    public final static String JBUTTON9_SELECTED_ICON_PATH =
155     INIT_ICONS_PATH + "applicationsselected.jpg" ;

    //For MessageWindow
    public final static long SHOW_TIME = 2000;
    public final static String WARNING_ICON_PATH =
160     INIT_ICONS_PATH + "warning.jpg" ;

    //For MMSPanel
    public final static int MMS_SERVICE_CODE = 3;
    public final static String MMS_START_DEFAULT_ICON_PATH =
165     INIT_ICONS_PATH + "sendmms.jpg" ;
    public final static String MMS_START_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "sendmmsselected.jpg" ;

    //For ProfilePanel
170     public static String PROFILE_PATH_CURRENT =
        INIT_PATH + "profiles/" + "default/" + "current.txt" ; //Also, for IIS
    public static String PROFILE_PATH_LOW_COST =
        INIT_PATH + "profiles/" + "default/" + "low_cost.txt" ;
    public static String PROFILE_PATH_HIGH_QUALITY =
175     INIT_PATH + "profiles/" + "default/" + "high_quality.txt" ;
    public static String PROFILE_PATH_BALANCED =
        INIT_PATH + "profiles/" + "default/" + "balanced.txt" ;
    public final static String PROVIDERS_INVERSED_FILE =
        INIT_PATH + "providers_inversed.txt" ;
180     public final static String ACTIVATE_DEFAULT_ICON_PATH =
        INIT_ICONS_PATH + "activate.jpg" ;
    public final static String ACTIVATE_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "activateselected.jpg" ;
    public final static String STORE_DEFAULT_ICON_PATH =
185     INIT_ICONS_PATH + "store.jpg" ;
    public final static String STORE_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "storeselected.jpg" ;

    //For SMSPanel
190     public final static int SMS_SERVICE_CODE = 2;
    public final static String SMS_START_DEFAULT_ICON_PATH =
        INIT_ICONS_PATH + "sendsms.jpg" ;
    public final static String SMS_START_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "sendsmsselected.jpg" ;
195

    //For VideoCallPanel
    public final static int VIDEOCALL_SERVICE_CODE = 5;
    public final static String VIDEOCALL_START_DEFAULT_ICON_PATH =
        INIT_ICONS_PATH + "startvideocall.jpg" ;
200     public final static String VIDEOCALL_START_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "startvideocallselected.jpg" ;
    public final static String VIDEOCALL_START_DISABLED_ICON_PATH =
        INIT_ICONS_PATH + "startvideocalldisabled.jpg" ;
    public final static String VIDEOCALL_END_DEFAULT_ICON_PATH =
205     INIT_ICONS_PATH + "endvideocall.jpg" ;
    public final static String VIDEOCALL_END_SELECTED_ICON_PATH =
        INIT_ICONS_PATH + "endvideocallselected.jpg" ;
    public final static String VIDEOCALL_END_DISABLED_ICON_PATH =
        INIT_ICONS_PATH + "endvideocalldisabled.jpg" ;
210

```



```

//For VideoStreamPanel
public final static int VIDEOSTREAM_SERVICE_CODE = 6;
public final static String VIDEOSTREAM_START_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "startvideostream.jpg" ;
215 public final static String VIDEOSTREAM_START_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "startvideostreamselected.jpg" ;
public final static String VIDEOSTREAM_START_DISABLED_ICON_PATH =
    INIT_ICONS_PATH + "startvideostreamdisabled.jpg" ;
public final static String VIDEOSTREAM_END_DEFAULT_ICON_PATH =
220 INIT_ICONS_PATH + "endvideostream.jpg" ;
public final static String VIDEOSTREAM_END_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "endvideostreamselected.jpg" ;
public final static String VIDEOSTREAM_END_DISABLED_ICON_PATH =
    INIT_ICONS_PATH + "endvideostreamdisabled.jpg" ;
225

//For VoiceCallPanel
public final static int VOICECALL_SERVICE_CODE = 1;
public final static String VOICECALL_START_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "startcall.jpg" ;
230 public final static String VOICECALL_START_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "startcallselected.jpg" ;
public final static String VOICECALL_START_DISABLED_ICON_PATH =
    INIT_ICONS_PATH + "startcalldisabled.jpg" ;
public final static String VOICECALL_END_DEFAULT_ICON_PATH =
235 INIT_ICONS_PATH + "endcall.jpg" ;
public final static String VOICECALL_END_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "endcallselected.jpg" ;
public final static String VOICECALL_END_DISABLED_ICON_PATH =
    INIT_ICONS_PATH + "endcalldisabled.jpg" ;
240 public final static String VOICECALL_ALTERNATIVE_ANIMATED_ICON_PATH =
    INIT_ICONS_PATH + "voicecallanimated.gif" ;

//For WebBrowsingPanel
public final static int WEBBROWSING_SERVICE_CODE = 7;
245 public final static String WEBBROWSING_START_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "startwebbrowsing.jpg" ;
public final static String WEBBROWSING_START_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "startwebbrowsingselected.jpg" ;
public final static String WEBBROWSING_START_DISABLED_ICON_PATH =
250 INIT_ICONS_PATH + "startwebbrowsingdisabled.jpg" ;
public final static String WEBBROWSING_END_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "endwebbrowsing.jpg" ;
public final static String WEBBROWSING_END_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "endwebbrowsingselected.jpg" ;
255 public final static String WEBBROWSING_END_DISABLED_ICON_PATH =
    INIT_ICONS_PATH + "endwebbrowsingdisabled.jpg" ;

//For View Info Button
public final static String INFO_DEFAULT_ICON_PATH =
260 INIT_ICONS_PATH + "viewinfo.jpg" ;
public final static String INFO_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "viewinfoselected.jpg" ;
public final static String INFO_DISABLED_ICON_PATH =
    INIT_ICONS_PATH + "viewinfodisabled.jpg" ;
265

//For Edit Settings Button
public final static String EDIT_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "edit.jpg" ;
public final static String EDIT_SELECTED_ICON_PATH =
270 INIT_ICONS_PATH + "editselected.jpg" ;

//For Back Button (NextPanel)
public final static String BACK_BUTTON_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "back.jpg" ;
275 public final static String BACK_BUTTON_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "backselected.jpg" ;

//For Log File
public final static String LOG_FILE = INIT_PATH + "log.txt" ;
280

```

**Config.java**

```
//For PINFrame
public final static String OK_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "OK.jpg";
public final static String OK_SELECTED_ICON_PATH =
285    INIT_ICONS_PATH + "OKselected.jpg" ;
public final static String CANCEL_DEFAULT_ICON_PATH =
    INIT_ICONS_PATH + "Cancel.jpg" ;
public final static String CANCEL_SELECTED_ICON_PATH =
    INIT_ICONS_PATH + "Cancelselected.jpg" ;
290 public final static String PSW_FILE =
    INIT_PATH + "access_denied/psw.txt" ;

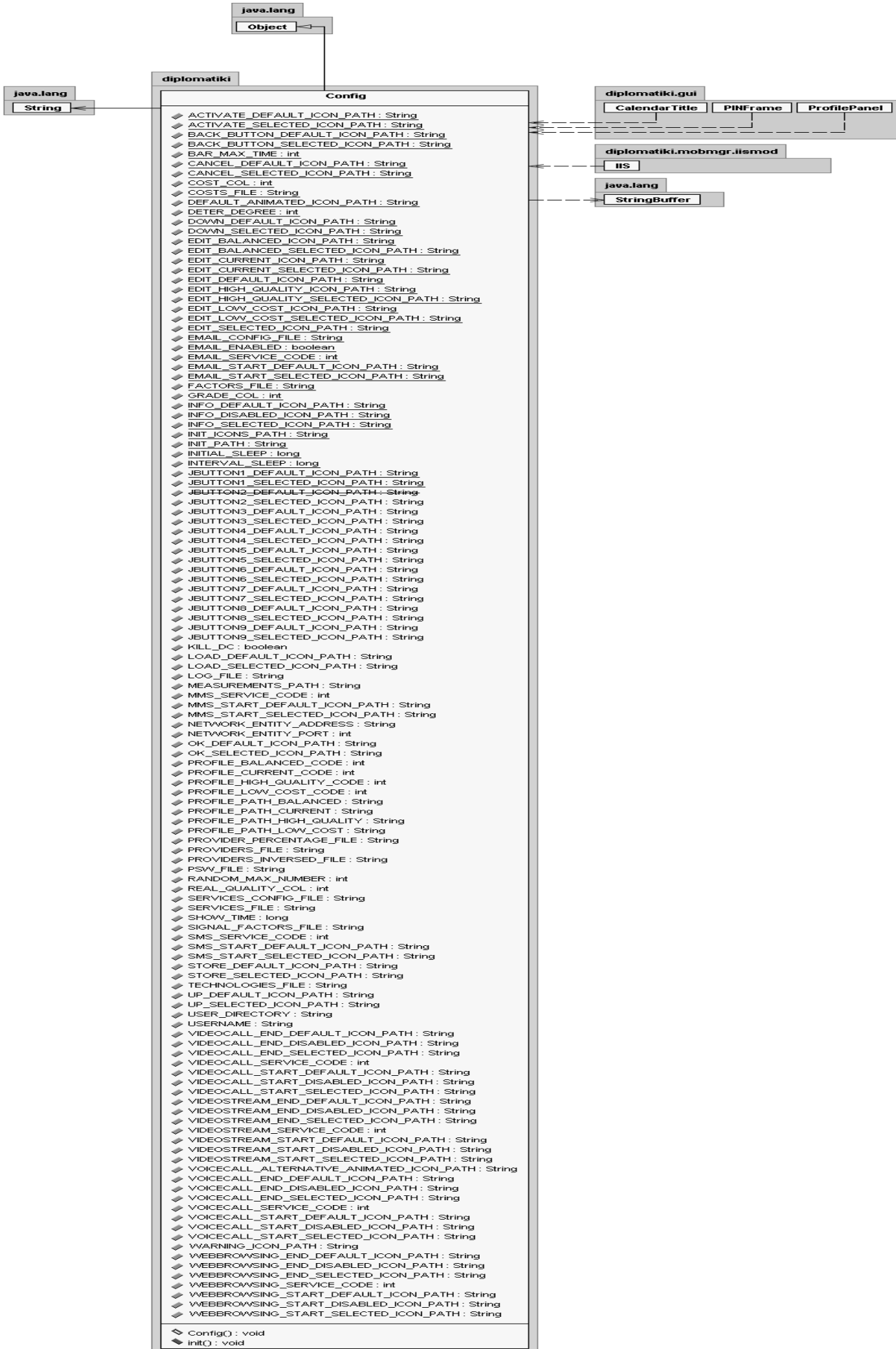
//For MenuFrame's Title
public static String USERNAME = "";
295

//For all Profile Paths
public static String USER_DIRECTORY = "";

//For Emailer
300 public final static String EMAIL_CONFIG_FILE =
    INIT_PATH + "services/" + "email_config.txt" ;
public final static boolean EMAIL_ENABLED = true;

/**
305  * Initiates the values of some static variables, including the name of the
  * current user, and the paths to the current user's stored profiles.
  *
  * @param username The user's name. In general, the TMS can support multiple
  * users
  */
310 public static void init(String username) {
    USERNAME = username;
    USER_DIRECTORY = USERNAME + "/";
    PROFILE_PATH_CURRENT =
315     INIT_PATH + "profiles/" + USER_DIRECTORY + "current.txt" ;
    PROFILE_PATH_LOW_COST =
        INIT_PATH + "profiles/" + USER_DIRECTORY + "low_cost.txt" ;
    PROFILE_PATH_HIGH_QUALITY =
        INIT_PATH + "profiles/" + USER_DIRECTORY + "high_quality.txt" ;
320     PROFILE_PATH_BALANCED =
        INIT_PATH + "profiles/" + USER_DIRECTORY + "balanced.txt" ;
}

/**
325  * Creates a new <code>Config</code> instance. This constructor is never used,
  * as all variables in this class are static.
  */
public Config () {
330 }
}
```



```

package diplomatiki ;

/**
 * <p>Title: Term Paper - <code>InfoStruct</code> class</p>
5  *
 * <p>Description: Instances of this class encapsulate all the information that
 * the GUI needs to know (and present), concerning an open connection of the
 * terminal.</p>
 *
10 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
15 * @version 1.0
 */
public class InfoStruct {
    private String ipAddress ;
    private String providerName ;
20    private String accessTechnology ;
    private String serviceName ;
    private String qos ;

    /**
25     * Creates a new <code>InfoStruct</code> instance.
    */
    public InfoStruct () {
    }

30    /**
     * Creates a new <code>InfoStruct</code> instance with the specified IP
     * Address, Provider Name, Access Technology, Service Name and QoS Level. This
     * means that the specified service has been allocated to the specified QoS
     * level and access point. These pieces of information are presented in the
35     * terminal's GUI.
     *
     * @param ipAddress The access point's IP address
     * @param providerName The access point's provider name
     * @param accessTechnology The point's access technology
40     * @param serviceName The name of the service
     * @param qos The QoS level
    */
    public InfoStruct (String ipAddress , String providerName ,
45        String accessTechnology ,
        String serviceName , String qos) {
        this.ipAddress = ipAddress ;
        this.providerName = providerName ;
        this.accessTechnology = accessTechnology ;
        this.serviceName = serviceName ;
50        this.qos = qos ;
    }

    /**
55     * Sets the access point's IP address.
     *
     * @param ipAddress The access point's IP address
    */
    public void setIpAddress (String ipAddress) {
60        this.ipAddress = ipAddress ;
    }

    /**
     * Sets the access point's provider name.
     *
65     * @param providerName The access point's provider name
    */
    public void setProviderName (String providerName) {
70        this.providerName = providerName ;
    }
}

```

```
/**
 * Sets the point's access technology.
 *
 * @param accessTechnology The point's access technology
 */
75 public void setAccessTechnology (String accessTechnology ) {
    this.accessTechnology = accessTechnology ;
}

80 /**
 * Sets the name of the service
 *
 * @param serviceName The service's name
 */
85 public void setServiceName (String serviceName ) {
    this.serviceName = serviceName ;
}

90 /**
 * Sets the QoS level.
 *
 * @param qos The QoS level
 */
95 public void setQos (String qos) {
    this.qos = qos;
}

100 /**
 * Returns the access point's IP address.
 *
 * @return The access point's IP address
 */
105 public String getIpAddress () {
    return ipAddress ;
}

110 /**
 * Returns the access point's provider name.
 *
 * @return The access point's provider name
 */
115 public String getProviderName () {
    return providerName ;
}

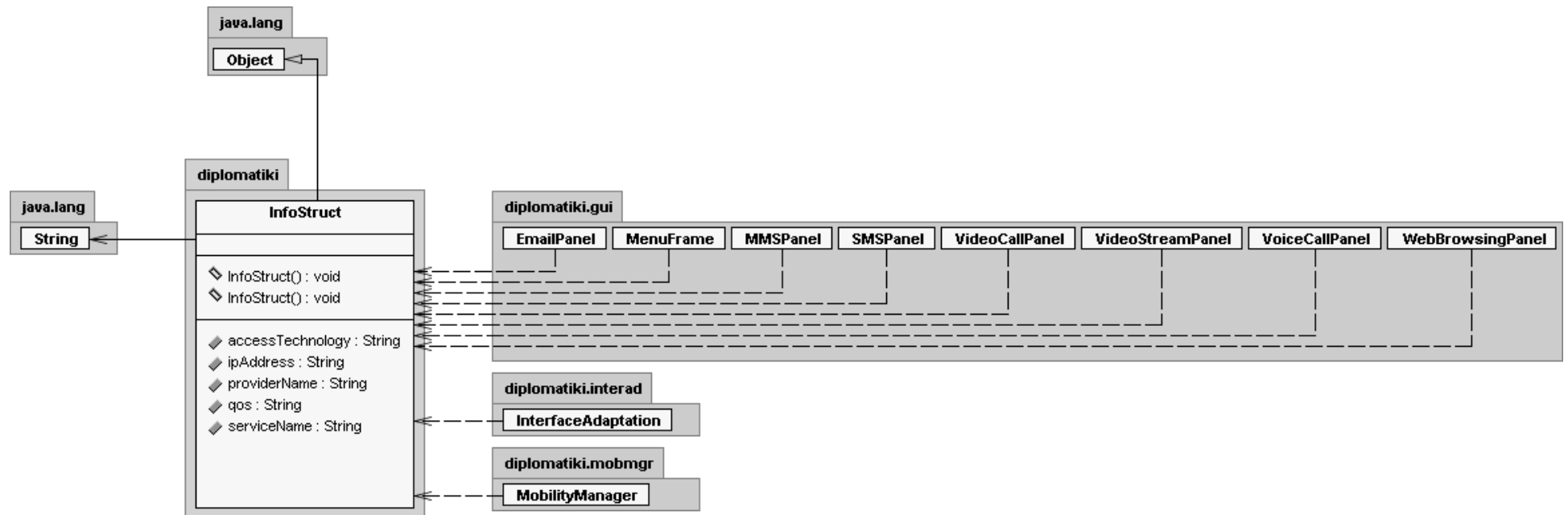
120 /**
 * Returns the point's access technology.
 *
 * @return The point's access technology
 */
125 public String getAccessTechnology () {
    return accessTechnology ;
}

130 /**
 * Returns the name of the service.
 *
 * @return The service's name
 */
135 public String getServiceName () {
    return serviceName ;
}

140 /**
 * Returns the QoS level.
 *
 * @return The QoS level
 */
145 public String getQos () {
    return qos;
}
```

InfoStruct.java

```
}  
}
```

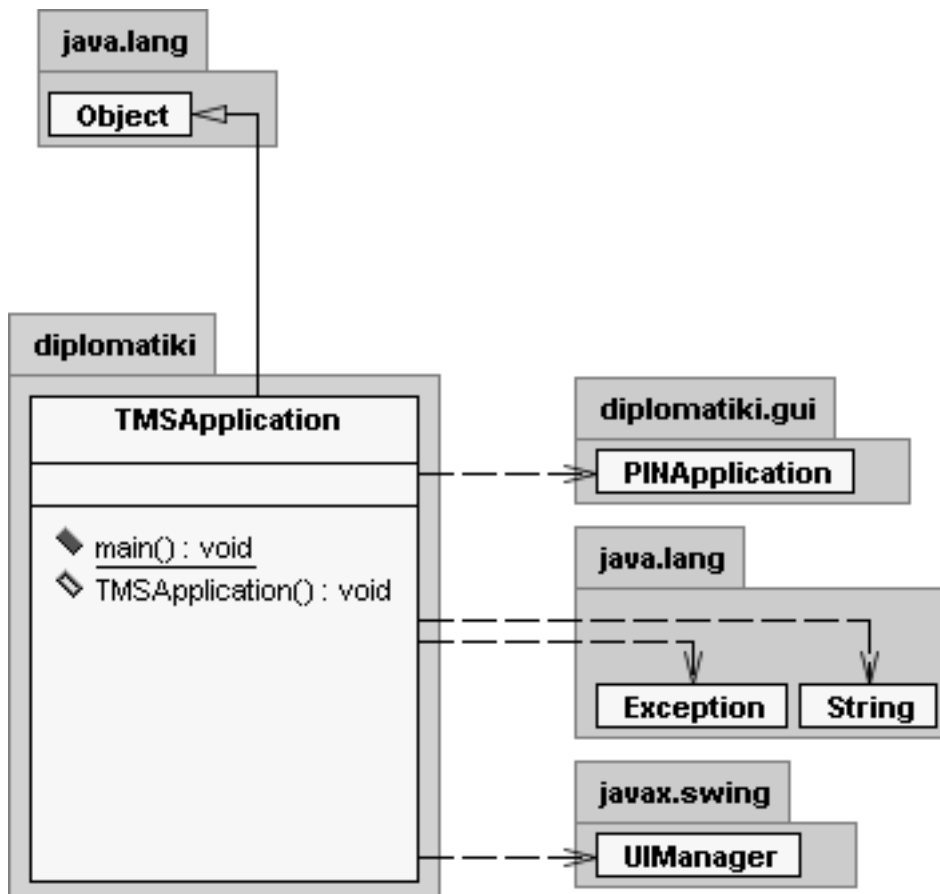


```
package diplomatiki ;

import javax.swing.UIManager ;
import diplomatiki .gui.PINApplication ;
5
/**
 * <p>Title: Term Paper - <code>TMSApplication</code> class</p>
 *
 * <p>Description: This class holds the <code>main()</code> method. The execution
10 * of the Terminal Management System (TMS) begins from this point.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
15 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
public class TMSApplication {
20
    /**
     * Creates a new <code>TMSApplication</code> instance. It is not necessary to
     * use this constructor, as the only method in this class is the
     * <code>main()</code> method.
25     */
    public TMSApplication () {
    }

    /**
30     * Starts the TMS application. Sets the GUI's "look and feel" and launches the
     * <code>PINApplication</code>.
     *
     * @param args No command line arguments are necessary
     */
35     public static void main (String [] args) {
        try {
            UIManager .setLookAndFeel (UIManager .getSystemLookAndFeelClassName ());
        }
        catch (Exception e) {
40             e.printStackTrace ();
        }
        new PINApplication ();
    }
45 }
```





**Choice.java**

```
package diplomatiki .mobmgr ;

/**
 * <p>Title: Term Paper - <code>Choice</code> class</p>
5  *
 * <p>Description: An instance of this class represents an allocation choice</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
10 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
15 public class Choice {
    private String ipAddress = "none";
    private int providerCode = -1;
    private int accessTechnology = -1;
    private int serviceCode = -1;
20 private int qos = -1;
    private double grade = Double.NEGATIVE_INFINITY ;

    /**
     * Creates a new <code>Choice</code> instance.
25  */
    public Choice () {
    }

    /**
30  * Creates a new <code>Choice</code> instance with the specified IP Address,
     * Provider Code, Access Technology, Service Code, QoS Level and Grade.
     *
     * @param ipAddress The IP address of a candidate access point
     * @param providerCode The access point's provider code
35  * @param accessTechnology The point's access technology
     * @param serviceCode The code of the requested service
     * @param qos A candidate QoS level
     * @param grade The grade (rating) of this allocation choice
     */
40  public Choice (String ipAddress , int providerCode , int accessTechnology ,
                 int serviceCode , int qos , double grade) {
        this.ipAddress = ipAddress ;
        this.providerCode = providerCode ;
        this.accessTechnology = accessTechnology ;
45  this.serviceCode = serviceCode ;
        this.qos = qos ;
        this.grade = grade ;
    }

50  /**
     * Sets the candidate access point's IP address
     *
     * @param ipAddress The candidate's IP address
     */
55  public void setIpAddress (String ipAddress) {
        this.ipAddress = ipAddress ;
    }

60  /**
     * Sets the candidate access point's provider code
     *
     * @param providerCode The candidate's provider code
     */
65  public void setProviderCode (int providerCode) {
        this.providerCode = providerCode ;
    }

70  /**
     * Sets the candidate access point's access technology.
     *

```

**Choice.java**

```
    * @param accessTechnology Sets the candidate's access technology
    */
    public void setAccessTechnology (int accessTechnology ) {
75     this.accessTechnology = accessTechnology ;
    }

    /**
    * Sets the code of the requested service.
    *
    * @param serviceCode The code of the requested service
    */
80     public void setServiceCode (int serviceCode ) {
        this.serviceCode = serviceCode ;
    }

85     /**
    * Sets the QoS level of this allocation choice.
    *
    * @param qos The QoS level of this choice
    */
90     public void setQos (int qos) {
        this.qos = qos ;
    }

95     /**
    * Sets the grade (rating) of this allocation choice.
    *
    * @param grade The grade of this choice
    */
100    public void setGrade (double grade) {
        this.grade = grade ;
    }

    /**
105    * Returns the candidate access point's IP address.
    *
    * @return The candidate's IP address
    */
    public String getIpAddress () {
110    return ipAddress ;
    }

    /**
    * Returns the candidate access point's provider code.
    *
    * @return The candidate's provider code
    */
115    public int getProviderCode () {
        return providerCode ;
    }

120    /**
    * Returns the candidate access point's access technology.
    *
    * @return Returns the candidate's access technology
    */
125    public int getAccessTechnology () {
        return accessTechnology ;
    }

130    /**
    * Returns the code of the requested service.
    *
    * @return The code of the requested service
    */
135    public int getServiceCode () {
        return serviceCode ;
    }

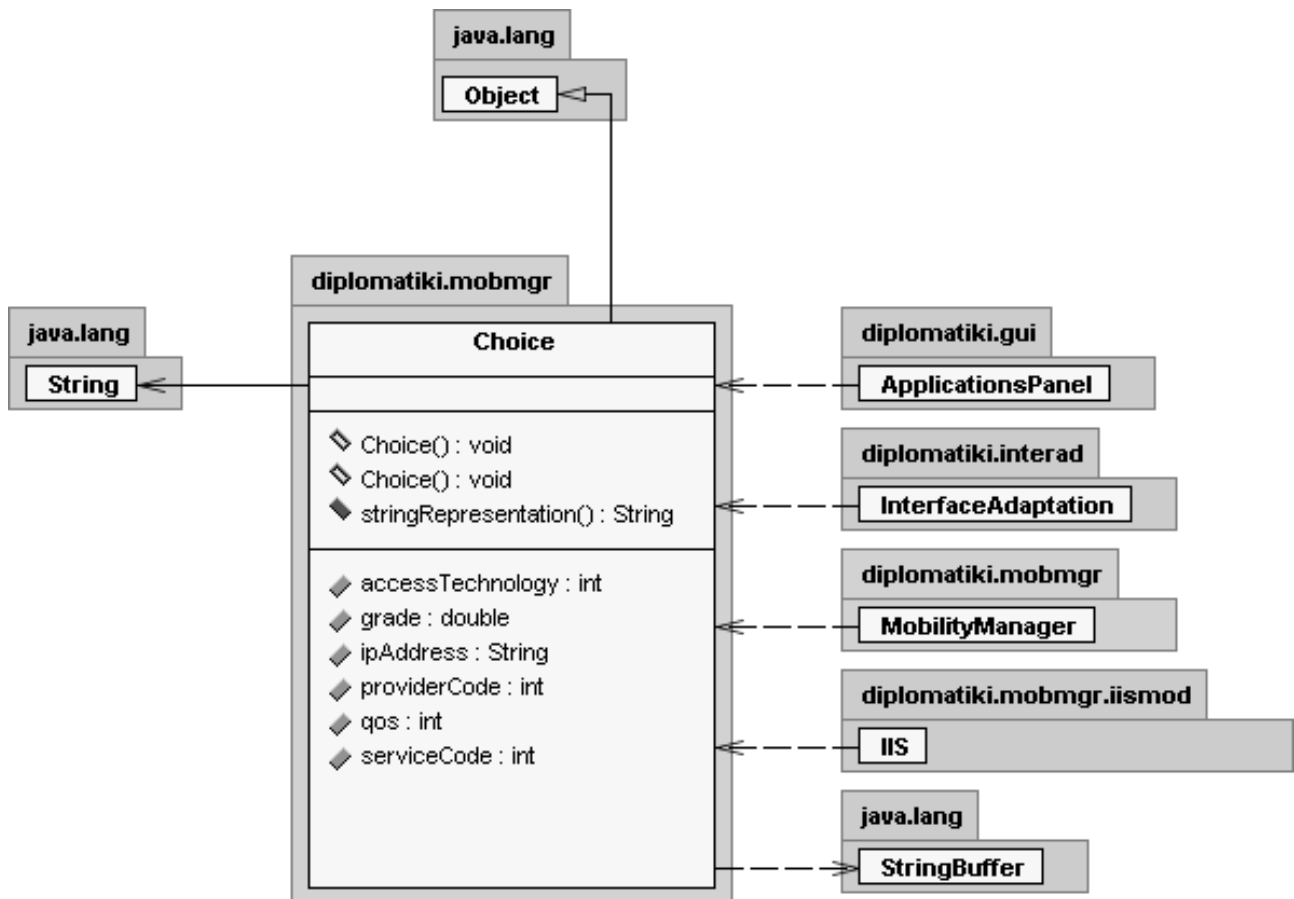
140    /**
```

**Choice.java**

```
    * Returns the QoS level of this allocation choice
    *
    * @return The QoS level of this choice
    */
145  public int getQos () {
        return qos;
    }

    /**
150  * Returns the grade (rating) of this allocation choice.
    *
    * @return The grade of this choice
    */
155  public double getGrade () {
        return grade;
    }

    /**
160  * Returns a string with all the details of this allocation choice.
    *
    * @return A string fully representing this choice
    */
    public String stringRepresentation () {
        String choiceStringRepresentation =
165         "Access Technology = " + getAccessTechnology () +
            "\tGrade = " + getGrade () +
            "\tIP Address = " + getIpAddress () +
            "\tProvider Code = " + getProviderCode () +
170         "\tQoS Level = " + getQos () +
            "\tService Code = " + getServiceCode ();
        return choiceStringRepresentation ;
    }
}
175
```



```

package diplomatiki .mobmgr ;

import java.util.*;
import diplomatiki .Config;
5
/**
 * <p>Title: Term Paper - <code>DeteriorationController</code> class</p>
 *
 * <p>Description: This class can be used to control the deterioration degree of
10 * the running services. More specifically, it uses a random number generator,
 * in order to simulate if the signal from a specific access point is severely
 * deteriorated or not. If so, the IIS algorithm should be triggered. This class
 * has been used in the early stages of the implementation, but is no longer
 * supported.</p>
15 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
20 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 * @deprecated This class is no longer used by the Terminal Management System
 * (TMS)
 */
25 public class DeteriorationController
    extends Thread {
    private final int RANDOM_MAX_NUMBER = Config.RANDOM_MAX_NUMBER ;
    private final int DETER_DEGREE = Config.DETER_DEGREE ;
    private final long INITIAL_SLEEP = Config.INITIAL_SLEEP ;
30 private final long INTERVAL_SLEEP = Config.INTERVAL_SLEEP ;

    private MobilityManager mm;

    /**
35 * Creates a new <code>DeteriorationController</code> instance.
    */
    public DeteriorationController () {
    }

    /**
40 * Creates a new <code>DeteriorationController</code> instance that is
 * associated with the specified <code>MobilityManager</code> object.
 *
 * @param mm A reference to the Mobility Manager module
45 */
    public DeteriorationController (MobilityManager mm) {
        this.mm = mm;
    }

    /**
50 * A <code>DeteriorationController</code> instance runs in a separate thread.
 * This is the entry point of this thread. Special care has been taken in
 * order to make the TMS thread-safe.
    */
55 public void run() {
    if (Config.KILL_DC) {
        return;
    }

    Random randomizer = new Random ();

    try {
        Thread.currentThread ().sleep (INITIAL_SLEEP );
    }
65 catch (InterruptedException ex) {
    ex.printStackTrace ();
    }

    while (true) {
70     try {

```

```

        Thread.currentThread ().sleep (INTERVAL_SLEEP );
    }
    catch (InterruptedException ex) {
        ex.printStackTrace ();
75    }

    synchronized (mm.getRunningServices ()) {

        System.out.println ("\nDC: Checking for deterioration...\n" );
80    boolean [] alreadyChecked = new boolean [mm.getRunningServices ().size ()];
        for (int k = 0; k < alreadyChecked .length; k++) {
            alreadyChecked [k] = false ;
        }

85    for (int i = 0; i < mm.getRunningServices ().size (); i++) {
        int decision = randomizer .nextInt (RANDOM_MAX_NUMBER );
        if ( (decision <= DETER_DEGREE ) && (!alreadyChecked [i])) {
            Choice ch = (Choice) mm.getRunningServices ().get (i);

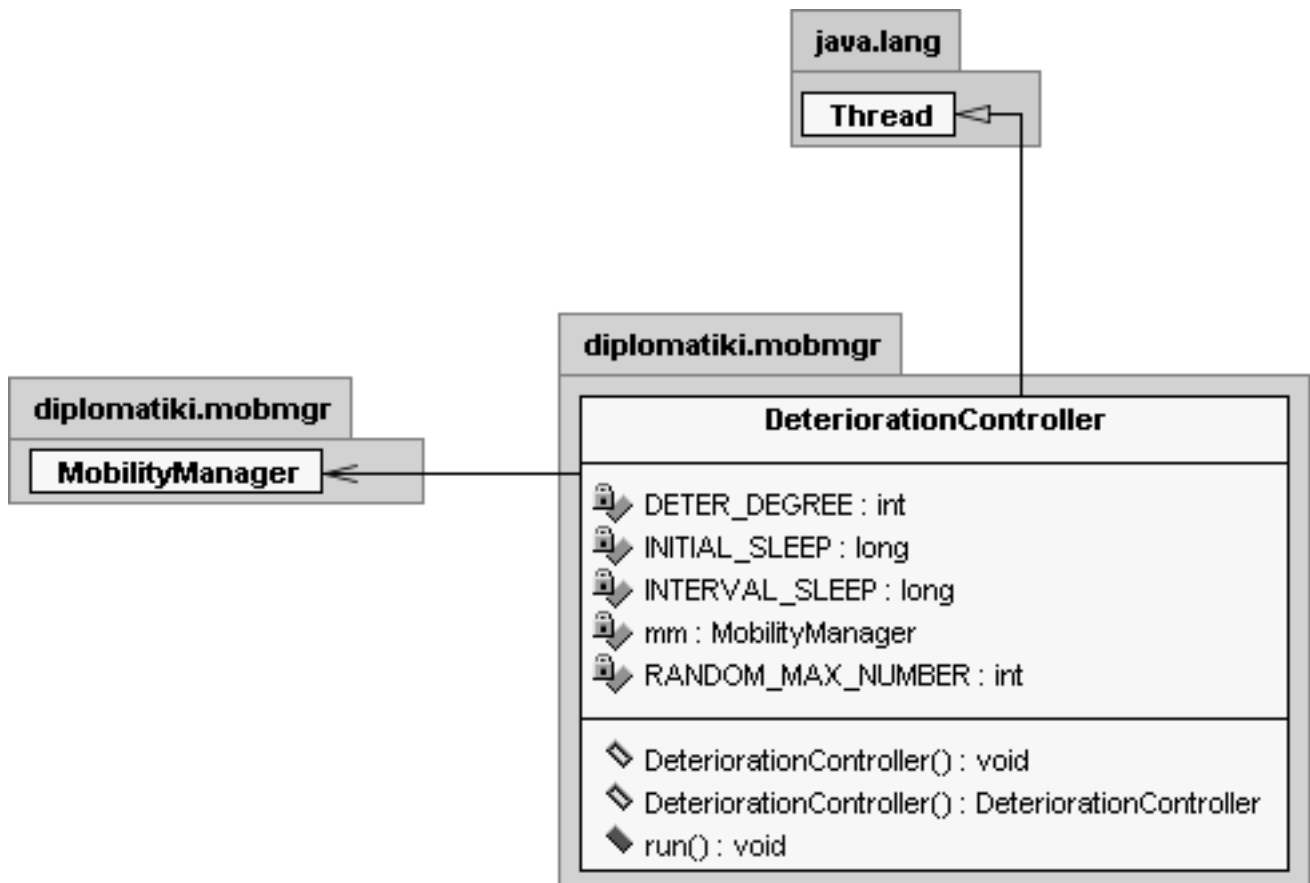
90    for (int j = 0; j < mm.getRunningServices ().size (); j++) {
            Choice choice = (Choice) mm.getRunningServices ().get (j);
            if ( (!alreadyChecked [j]) &&
                (choice.getIpAddress ().equals (ch.getIpAddress ()))) {
95    System.out.println (
                "\nDC:!-- Deterioration detected in service " +
                choice.getServiceCode () +
                " from AP " + choice.getIpAddress () + "\n");
                alreadyChecked [j] = true ;
                diplomatiki .InfoStruct myInfo = mm.executeIIS (choice .
                getServiceCode (), false , i);
                mm.getMenuFrame ().refreshAppropriateConnectionInfo (choice .
                getServiceCode (), myInfo);
            } //end of if(j)
        } //end of for(j)

100    } //end of if(decision)
    } //end of for(i)

105    } //end of synchronized
    } //end of while

110 }
}
}

```





```

package diplomatiki .mobmgr .iismod ;

/**
 * <p>Title: Term Paper - <code>AccessPoint</code> class</p>
5  *
 * <p>Description: Objects of this class represent access points which the
 * terminal can connect to.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
10 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
15 */
public class AccessPoint {
    private String ipAddress ;
    private int providerCode ;
    private int accessTechnology ;
20 private int signalStrength ;
    private int availBandwidth ;
    private double [][] info = null ;
    private boolean toBeRemoved = false ;

25 /**
 * Creates a new <code>AccessPoint</code> instance.
 */
    public AccessPoint () {
30 }

    /**
 * Creates a new <code>AccessPoint</code> instance with the specified IP
 * Address, Provider Code, Access Technology, Signal Strength and Bandwidth.
 *
35 * @param ipAddress The access point's IP address
 * @param providerCode The access point's provider code
 * @param accessTechnology The point's access technology
 * @param signalStrength The strength of the signal currently received by the
 * access point
40 * @param availBandwidth The access point's available bandwidth (in Kbps)
 */
    public AccessPoint (String ipAddress , int providerCode , int accessTechnology ,
        int signalStrength , int availBandwidth ) {
45         this.ipAddress = ipAddress ;
         this.providerCode = providerCode ;
         this.accessTechnology = accessTechnology ;
         this.signalStrength = signalStrength ;
         this.availBandwidth = availBandwidth ;
50     }

    /**
 * Sets the access point's IP address.
 *
 * @param ipAddress The new IP address
55 */
    void setIpAddress (String ipAddress ) {
         this.ipAddress = ipAddress ;
    }

60 /**
 * Sets the access point's provider code.
 *
 * @param providerCode The new provider code
 */
65 void setProvider (int providerCode ) {
         this.providerCode = providerCode ;
    }

    /**
70 * Sets the point's access technology.

```

```

*
* @param accessTechnology The new access technology code
*/
75 void setAccessTechnology (int accessTechnology ) {
    this.accessTechnology = accessTechnology ;
}

/**
* Sets the access point's signal strength.
80 *
* @param signalStrength The new signal strength
*/
void setSignalStrength (int signalStrength ) {
85     this.signalStrength = signalStrength ;
}

/**
* Sets the access point's available bandwidth.
*
90 * @param availBandwidth The new available bandwidth
*/
void setAvailBandwidth (int availBandwidth ) {
    this.availBandwidth = availBandwidth ;
95 }

/**
* Creates the info table for this access point.
*
* @param noRows Defines the size of the table. The actual number of rows
100 * created is the value of this argument plus one (<code>noRows</code>+1).
* The number of columns is always equal to <code>3</code>
*/
public void createInitialInfo (int noRows) {
105     info = new double [noRows + 1][3];
}

/**
* Sets the value of the table element specified. The two first arguments
* specify the element's position, while the third specifies the new value.
110 *
* @param indexQoS The table row of the element
* @param column The table column of the element
* @param value The new value of the element specified
*/
115 void setInfo (int indexQoS , int column , double value) {
    info [indexQoS ][column] = value ;
}

/**
120 * Sets the <code>toBeRemoved</code> property (instance variable) of this
* object. If <code>>true</code>, then this access point should be removed from
* the list of access points maintained by the IIS sub-module.
*
* @param toBeRemoved The new value of the <code>toBeRemoved</code> property.
125 */
public void setToBeRemoved (boolean toBeRemoved ) {
    this.toBeRemoved = toBeRemoved ;
}

130 /**
* Returns the access point's IP address.
*
* @return The access point's IP address
*/
135 String getIpAddress () {
    return ipAddress ;
}

/**
140 * Returns the access point's provider code.

```

```

    *
    * @return The access point's provider code
    */
145  int getProviderCode () {
        return providerCode ;
    }

    /**
    * Returns the point's access technology.
    *
    * @return The point's access technology
    */
150  int getAccessTechnology () {
        return accessTechnology ;
155  }

    /**
    * Returns the access point's signal strength.
    *
    * @return The access point's signal strength
    */
160  int getSignalStrength () {
        return signalStrength ;
165  }

    /**
    * Returns the access point's available bandwidth.
    *
    * @return The access point's available bandwidth
    */
170  int getAvailBandwidth () {
        return availBandwidth ;
    }

175  /**
    * Returns the number of rows of the access point's info table. If the table
    * is <code>null</code>, it returns <code>0</code>.
    *
    * @return The number of rows of the info table
180  */
    int getInfoSize () {
        if (info == null) {
            return 0;
        }
185  return info.length;
    }

    /**
    * Returns the value of the specified table element. The element's position is
    * specified by the two arguments.
    *
    * @param qos The element's table row
    * @param column The element's table column
    * @return The element's value
195  */
    double getInfo (int qos, int column) {
        return info [qos] [column];
    }

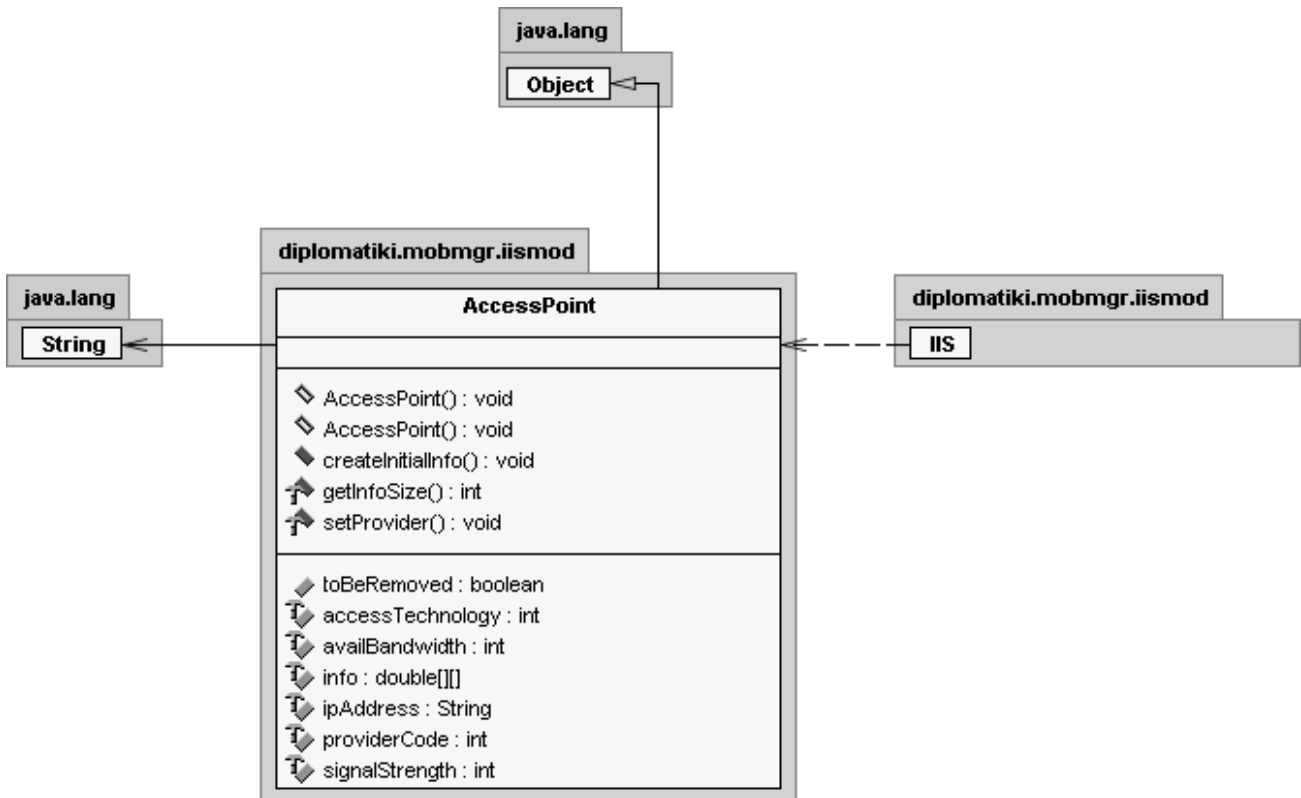
200  /**
    * Returns the value of the <code>toBeRemoved</code> property (instance
    * variable). If <code>true</code>, then this access point should be removed
    * from the list of access points maintained by the IIS sub-module.
    *
    * @return The value of the <code>toBeRemoved</code> property.
    */
205  public boolean isToBeRemoved () {
        return toBeRemoved ;
    }

210

```

**AccessPoint.java**

```
}
```



```

package diplomatiki .mobmgr .iismod ;

import java.io.*;
import java.util.*;
5 import java.text.NumberFormat ;

import org.jdom.*;
import org.jdom.input.*;
import diplomatiki .*;
10 import diplomatiki .mobmgr .*;

/**
 * <p>Title: Term Paper - <code>IIS</code> class</p>
 *
15 * <p>Description: This is the main class implementing the IIS sub-module of a
 * TMS.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
20 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
25 public class IIS {
    private final int REAL_QUALITY_COL = Config.REAL_QUALITY_COL ;
    private final int COST_COL = Config.COST_COL ;
    private final int GRADE_COL = Config.GRADE_COL ;

30     private final String SERVICES_FILE = Config.SERVICES_FILE ;
    private final String COSTS_FILE = Config.COSTS_FILE ;
    private final String FACTORS_FILE = Config.FACTORS_FILE ;
    private final String SIGNAL_FACTORS_FILE = Config.SIGNAL_FACTORS_FILE ;
    private final String CURRENT_PROFILE_PATH = Config.PROFILE_PATH_CURRENT ;
35     private final String PROVIDER_PERCENTAGE_FILE = Config.
        PROVIDER_PERCENTAGE_FILE ;

    private Vector APList = new Vector (), choices = new Vector ();
    private Service myService ;
40     private Stack sortedChoices ;

    private MobilityManager mm;

    private PrintWriter fileout = null;
45

    /**
     * Creates a new <code>IIS</code> instance.
     */
    public IIS () {
50     }

    /**
     * Creates a new <code>IIS</code> instance that belongs to the specified
     * <code>MobilityManager</code> object.
55     *
     * @param mm The <code>MobilityManager</code> object that represents the
     * module which contains the IIS sub-module
     */
    public IIS (MobilityManager mm) {
60         this .mm = mm;
    }

    /**
     * Executes the Intelligent Interface Selection (IIS) algorithm.
65     *
     * @param serviceCode The code of the service for which the IIS algorithm
     * will be executed
     * @return An ordered list of allocation choices
     */
70     public Stack execute (int serviceCode ) {

```

```

    if (serviceCode <= 0) {
        return null;
    }
    openLogFile ();
75    myService = constructServiceItem (serviceCode );
    convertToAPList (getMeasurements ());
    writeInitialInfoToLogFile ();
    for (Iterator i = APList.iterator (); i.hasNext (); ) {
        AccessPoint currAP = (AccessPoint ) i.next ();
80        findQuality (currAP);
        findCost (currAP);
        objectiveFunction (currAP);
    }
    findBest ();
85
    for (int j = 0; j < APList.size (); j++) {
        for (Iterator i = APList.iterator (); i.hasNext (); ) {
            AccessPoint currAP = (AccessPoint ) i.next ();
90            if (currAP.isToBeRemoved ()) {
                APList.removeElement (currAP);
                break ;
            }
        }
    }
95
    return sortedChoices ;
}

/**
100 * Acquires the table of Measurements.
 *
 * @return The table of Measurements
 */
private String [][] getMeasurements () {
105     return mm.getMeasurements ();
}

/**
110 * Converts the table of Measurements to a list of <code>AccessPoint</code>
 * objects.
 *
 * @param measurements The table of Measurements
 */
private void convertToAPList (String [][] measurements ) {
115     int size = measurements .length;
    for (int i = 0; i < size; i++) {
        AccessPoint myAP =
            new AccessPoint (measurements [i][0],
120                new Integer (measurements [i][1]).intValue (),
                new Integer (measurements [i][2]).intValue (),
                new Integer (measurements [i][3]).intValue (),
                new Integer (measurements [i][4]).intValue ());
        APList.add (myAP);
    }
125 }

/**
 * Constructs a <code>Service</code> item.
 *
130 * @param serviceCode The code of the service for which the
 * <code>Service</code> object should be constructed
 * @return The <code>Service</code> object
 */
private Service constructServiceItem (int serviceCode ) {
135     SAXBuilder builder = new SAXBuilder ();
    try {
        Document doc = builder.build (new FileInputStream (SERVICES_FILE ));
        Element el = findServiceElement (serviceCode , doc.getRootElement ());
        if (el == null) {
140             return null;
        }
    }
}

```

```

    }

    Properties currentConfig = loadPropertiesFile (CURRENT_PROFILE_PATH );
    int maxUserQoS = new Integer (currentConfig .getProperty (" + serviceCode )).
145     intValue ();

    //The third actual parameter represents the user's settings
    Service s = new
        Service (el.getAttributeValue ("name"), serviceCode , maxUserQoS ,
150         new Integer (el.getAttributeValue ("availQoSLevels" )).
            intValue ());

    List lst = el.getChildren ("QoS");
    for (Iterator i = lst.iterator (); i.hasNext (); ) {
        Element elm = (Element) i.next ();
155     s.setQoS (new Integer (elm.getAttributeValue ("level" )).intValue (),
        new Integer (elm.getText ()).intValue ());
    }
    return s;
}
160 catch (IOException ex) {
    ex.printStackTrace ();
}
catch (JDOMException ex) {
    ex.printStackTrace ();
165 }
return null;
}

/**
170 * Finds the appropriate <i>service</i> element from the "services.xml" file
* of the databank. Returns the <i>service</i> element that has the specified
* attribute value. The name of this attribute is "code".
*
* @param serviceCode The value of the attribute "code". The search performed
175 * is based on this value
* @param rootElement The parent element of the element that is searched for
* @return The appropriate <i>service</i> element
*/
private Element findServiceElement (int serviceCode , Element rootElement ) {
180 List mixedContent = rootElement .getContent ();
for (Iterator i = mixedContent .iterator (); i.hasNext (); ) {
    Object o = i.next ();
    if (o instanceof Element) {
        Element el = (Element) o;
185 if (new Integer (el.getAttributeValue ("code" )).
            intValue () == serviceCode ) {
            return el;
        }
    }
190 }
return null;
}

/**
195 * The "Find Quality" stage of the IIS algorithm. This stage finds the QoS
* levels supported by the given access point. It also calculates the
* respective <i>real</i> QoS levels.
*
* @param myAP The access point for which this stage of the algorithm will be
200 * executed
*/
private void findQuality (AccessPoint myAP) {
    int i = 1, max = 0;
    while (i <= myService .getNoQoSLevels ()) {
205 if (myAP.getAvailBandwidth () >= myService .getQoS (i)) {
        max = i;
    }
    else {
        break ;
210 }
}

```



```

        i++;
    }
    if (max == 0) {
215     //APList.remove(myAP);
        myAP.setToBeRemoved (true);
        return;
    }

220     if (max > myService.getMaxUsersQoS ()) {
        max = myService.getMaxUsersQoS ();
    }

    myAP.createInitialInfo (max);

225     Properties signalFactorConfig = loadPropertiesFile (SIGNAL_FACTORS_FILE );
    double multFactor = new
        Double (signalFactorConfig.getProperty (" " + myAP.getSignalStrength ())).
        doubleValue ();
230     for (i = 1; i <= max; i++) {
        myAP.setInfo (i, REAL_QUALITY_COL , multFactor * i);
    }
}

235 /**
 * The "Find Cost" stage of the IIS algorithm. This stage finds the cost of
 * all the possible allocation choices for the given access point.
 *
 * @param myAP The access point for which this stage of the algorithm will be
240 *   executed
 */
private void findCost (AccessPoint myAP) {
    if (myAP == null) {
245     return;
    }

    SAXBuilder builder = new SAXBuilder ();
    try {
        Document doc = builder.build (new FileInputStream (COSTS_FILE ));
250     Element service = findServiceElement (myService.getServiceCode (),
        doc.getRootElement ());

        for (int i = 1; i < myAP.getInfoSize (); i++) {
            Element qos =
255     findChildByAttribute (service, "level", new Integer (i).toString ());

            Element provider =
                findChildByAttribute (qos, "code",
260     new Integer (myAP.getProviderCode ()).
                toString ());

            Element technology =
                findChildByAttribute (provider, "code",
265     new Integer (myAP.getAccessTechnology ()).
                toString ());

            /*
            System.out.println(">> info size = "+myAP.getInfoSize()+
                " i="+i+
270     " qos="+qos.toString()+
                " provider="+provider.toString ();
            System.out.println(" technology = "+technology.toString()+
                " cost = "+technology.getText ());
            */

275     if (technology == null) {
        myAP.setInfo (i, COST_COL , 1000.0);
        if (i == 1) {
            myAP.setToBeRemoved (true);
280     }
    }
}

```

```

    }
    else {
        myAP.setInfo (i, COST_COL,
                    new Double (technology .getText ()).doubleValue ());
285     }
    }
}
catch (IOException ex) {
    ex.printStackTrace ();
290 }
catch (JDOMException ex) {
    ex.printStackTrace ();
}
}
295
/**
 * Finds the appropriate child element of the given element, in an XML
 * document, by using the given attribute name/attribute value pair as a
 * search criteria.
300 *
 * @param parentEl The parent element of the element that we search for
 * @param attrName The name of the attribute
 * @param attrValue The value of the attribute
 * @return The appropriate XML element
305 */
private Element findChildByAttribute (Element parentEl , String attrName ,
                                     String attrValue) {
    List children = parentEl .getChildren ();
    for (Iterator i = children .iterator (); i.hasNext (); ) {
310     Element el = (Element) i.next ();
        if (el.getAttributeValue (attrName).equals (attrValue)) {
            return el;
        }
    }
315
    return null;
}

/**
320 * The "Objective Function" stage of the IIS algorithm. In this stage, we
 * calculate the value of the objective function for all the possible
 * allocation choices with respect to the given access point.
 *
 * @param myAP The access point for which this stage of the algorithm will be
325 *   executed
 */
private void objectiveFunction (AccessPoint myAP) {
    if (myAP == null) {
330     return;
    }

    double maxCost = 1.0;
    int qosLevels = 5;

335     try {
        SAXBuilder builder = new SAXBuilder ();
        Document doc = builder .build (new FileInputStream (COSTS_FILE));
        Element service = findServiceElement (myService .getServiceCode (),
                                             doc.getRootElement ());
340     maxCost = new Double (service .getAttributeValue ("maxCost")).doubleValue ();
    }
    catch (IOException ex) {
        ex.printStackTrace ();
    }
345     catch (JDOMException ex) {
        ex.printStackTrace ();
    }

    try {
350     SAXBuilder builder = new SAXBuilder ();

```

```

Document doc = builder .build (new FileInputStream (SERVICES_FILE ));
Element service = findServiceElement (myService .getServiceCode (),
doc.getRootElement ());
qosLevels = new Integer (service .getAttributeValue ("availQoSLevels" )).
355     intValue ();
}
catch (IOException ex) {
    ex.printStackTrace ();
}
360 catch (JDOMException ex) {
    ex.printStackTrace ();
}

Properties providerBonusConfig = loadPropertiesFile (
365     PROVIDER_PERCENTAGE_FILE );

double providerBonus = new Double (providerBonusConfig .getProperty ("bonus" )).
    doubleValue ();

370 Properties factorConfig = loadPropertiesFile (FACTORS_FILE );

double a = new Double (factorConfig .getProperty ("a" )).doubleValue ();
double b = new Double (factorConfig .getProperty ("b" )).doubleValue ();
double c = new Double (factorConfig .getProperty ("c" )).doubleValue ();
375

//Set the values of the 3 main factors, according to the user's preferences
double qualityFactor = 0.0;
double costFactor = 0.0;
double providerFactor = 0.0;
380

Properties currentConfig = loadPropertiesFile (CURRENT_PROFILE_PATH );
if (currentConfig .getProperty ("factor1" ).equals ("Quality" )) {
    qualityFactor = a;
}
385 else if (currentConfig .getProperty ("factor1" ).equals ("Cost" )) {
    costFactor = (a * qosLevels) / maxCost;
}
else {
    if (new Integer (currentConfig .getProperty ("providerCode" )).intValue () ==
390         myAP .getProviderCode ()) {
        providerFactor = a;
    }
}
if (currentConfig .getProperty ("factor2" ).equals ("Quality" )) {
395     qualityFactor = b;
}
else if (currentConfig .getProperty ("factor2" ).equals ("Cost" )) {
    costFactor = (b * qosLevels) / maxCost;
}
400 else {
    if (new Integer (currentConfig .getProperty ("providerCode" )).intValue () ==
        myAP .getProviderCode ()) {
        providerFactor = b;
    }
405 }
if (currentConfig .getProperty ("factor3" ).equals ("Quality" )) {
    qualityFactor = c;
}
else if (currentConfig .getProperty ("factor3" ).equals ("Cost" )) {
410     costFactor = (c * (double) qosLevels) / maxCost;
}
else {
    if (new Integer (currentConfig .getProperty ("providerCode" )).intValue () ==
415         myAP .getProviderCode ()) {
        providerFactor = c;
    }
}

for (int i = 1; i < myAP .getInfoSize (); i++) {
420     double grade = qualityFactor * myAP .getInfo (i, REAL_QUALITY_COL ) -

```

```

        costFactor * myAP.getInfo (i, COST_COL);

        /*
        System.out.println("AP IP = " + myAP.getIpAddress());
425     System.out.println("qualityFactor*myAP.getInfo(i, REAL_QUALITY_COL) = "+
            qualityFactor*myAP.getInfo(i,REAL_QUALITY_COL));
        System.out.println("costFactor*myAP.getInfo(i, COST_COL) = "+
            costFactor*myAP.getInfo(i, COST_COL));
        */

430     grade = grade + Math.abs(grade * providerFactor * providerBonus );
        myAP.setInfo (i, GRADE_COL , grade);

        /*
435     System.out.println("Quality Factor = " + qualityFactor);
        System.out.println("Real Quality = " + myAP.getInfo(i, REAL_QUALITY_COL));
        System.out.println("Cost Factor = " + costFactor);
        System.out.println("Cost = " + myAP.getInfo(i, COST_COL));
        System.out.println("Provider Factor = " + providerFactor);
440     System.out.println("Grade = " + grade);
        System.out.println("*****");
        */

        NumberFormat nf4 = NumberFormat .getInstance ();
445     NumberFormat nf5 = NumberFormat .getInstance ();
        nf4.setMaximumFractionDigits (4);
        nf5.setMaximumFractionDigits (5);
        writeOFDataToLogFile (myAP.getIpAddress () + "\t" +
            qualityFactor + "\t\t" +
450     i + "\t\t" +
            nf5.format (myAP.getInfo (i, REAL_QUALITY_COL )) +
            "\t\t" +
            nf4.format (costFactor ) + "\t\t" +
            nf5.format (myAP.getInfo (i, COST_COL )) + "\t\t" +
455     nf5.format (providerFactor ) + "\t\t" +
            nf5.format (grade));
    }
}

460 /**
 * Loads a Properties file.
 *
 * @param filePath The path of the properties file
 * @return A <code>Properties</code> instance that represents the properties
465 * file's contents
 */
private Properties loadPropertiesFile (String filePath) {
    Properties config = new Properties ();
    InputStream input = null;
470     try {
        input = new FileInputStream (filePath);
        config.load (input);
    }
    catch (IOException ex) {
475     System.err.println ("Cannot open and load properties file." );
    }
    finally {
        try {
            if (input != null) {
480     input.close ();
            }
        }
        catch (IOException ex) {
485     System.err.println ("Cannot close properties file." );
        }
    }

    return config;
}
490

```

```

/**
 * Converts the list of access points to a list of choices
 * (<code>Choice</code> objects).
 */
495 private void convertToChoicesList () {
    for (Iterator i = APList.iterator (); i.hasNext (); ) {
        AccessPoint myAP = (AccessPoint) i.next ();
        for (int j = 1; j < myAP.getInfoSize (); j++) {
500             Choice myChoice = new Choice (myAP.getIpAddress (),
                                                myAP.getProviderCode (),
                                                myAP.getAccessTechnology (),
                                                myService.getServiceCode (), j,
                                                myAP.getInfo (j, GRADE_COL ));
505             choices.add (myChoice );
        }
    }
}

/**
510 * Sorts all the possible allocation choices according to their grade.
 */
private void findBest () {
    convertToChoicesList ();
    Choice [] myArray = new Choice [0];
515    myArray = (Choice []) choices.toArray (myArray);

    for (int i=0;i<myArray.length;i++) {
520         System.out.println ("i = "+i+" grade = "+myArray[i].getGrade ());
    }
}

bubbleSort (myArray);

525 /*
    System.out.println ("\nSorted Array of Choices: (" + myArray.length +
        " choices)");
    for (int i = 0; i < myArray.length; i++) {
        System.out.println ("i = " + i + " grade = " + myArray[i].getGrade ());
530     }
    */

writeSortedListToLogFile (myArray);
closeLogFile ();

535 sortedChoices = new Stack ();
for (int i = 0; i < myArray.length; i++) {
    sortedChoices.push (myArray [i]);
}
540 }

/**
 * Sorts a table using the "Bubble Sort" algorithm.
 *
545 * @param choiceArray The table that should be sorted
 */
private void bubbleSort (Choice [] choiceArray) {
    Choice temp = new Choice ();
    int bottom, i;
550    boolean exchanged = true;

    bottom = choiceArray.length - 2;

    while (exchanged) {
555         exchanged = false;
        for (i = 0; i <= bottom; i++) {
            if (choiceArray [i].getGrade () > choiceArray [i + 1].getGrade ()) {
                temp = choiceArray [i];
                choiceArray [i] = choiceArray [i + 1];
560                 choiceArray [i + 1] = temp;
            }
        }
    }
}

```

```

        exchanged = true;
    }
}
565     bottom--;
    }
}

570 /**
 * Opens a log file in order to append log data.
 */
private void openLogFile () {
575     try {
        fileout = new
            PrintWriter (new
                BufferedWriter (new FileWriter (Config.LOG_FILE , true)));
    }
    catch (IOException ex) {
580         System.err.println ("Cannot open log file." );
    }
}

/**
585 * Writes all the appropriate initial info to the log file, including the
 * current date/time, service name, current measurements, user preferences,
 * and more.
 */
private void writeInitialInfoToLogFile () {
590     fileout.println (System.getProperty ("line.separator" ) +
        "-----" +
        "-----" +
        "-----" +
        System.getProperty ("line.separator" ));
595
    fileout.println ("# " +
        new java.util.Date (System.currentTimeMillis ()).
            toString () +
        ": " +
600         System.getProperty ("line.separator" ));

    fileout.println ("# Service " + myService.getServiceName ().toUpperCase () +
        " has been requested" );

605     fileout.println ();
    mm.writeMeasurementsToLogFile (fileout);
    fileout.println ();

    fileout.println ("The user's maximum preferred QoS level is QoS-" +
610         myService.getMaxUsersQoS ());

    fileout.println ("and the user's general preferences are: " );
    Properties currentConfig = loadPropertiesFile (CURRENT_PROFILE_PATH );
    fileout.println ("1. " + currentConfig.getProperty ("factor1" ) +
615         System.getProperty ("line.separator" ) +
        "2. " + currentConfig.getProperty ("factor2" ) +
        System.getProperty ("line.separator" ) +
        "3. " + currentConfig.getProperty ("factor3" ) +
        System.getProperty ("line.separator" ) +
620         "and the preferred provider is " +
        currentConfig.getProperty ("providerCode" ));

    fileout.println (System.getProperty ("line.separator" ) +
        "The list of choices is: " );
625     fileout.println ("IP Address\tQuality Factor\tQoS Level\tReal Quality\t" +
        "Cost Factor\tCost\t\tProvider Factor\tGrade" );
}

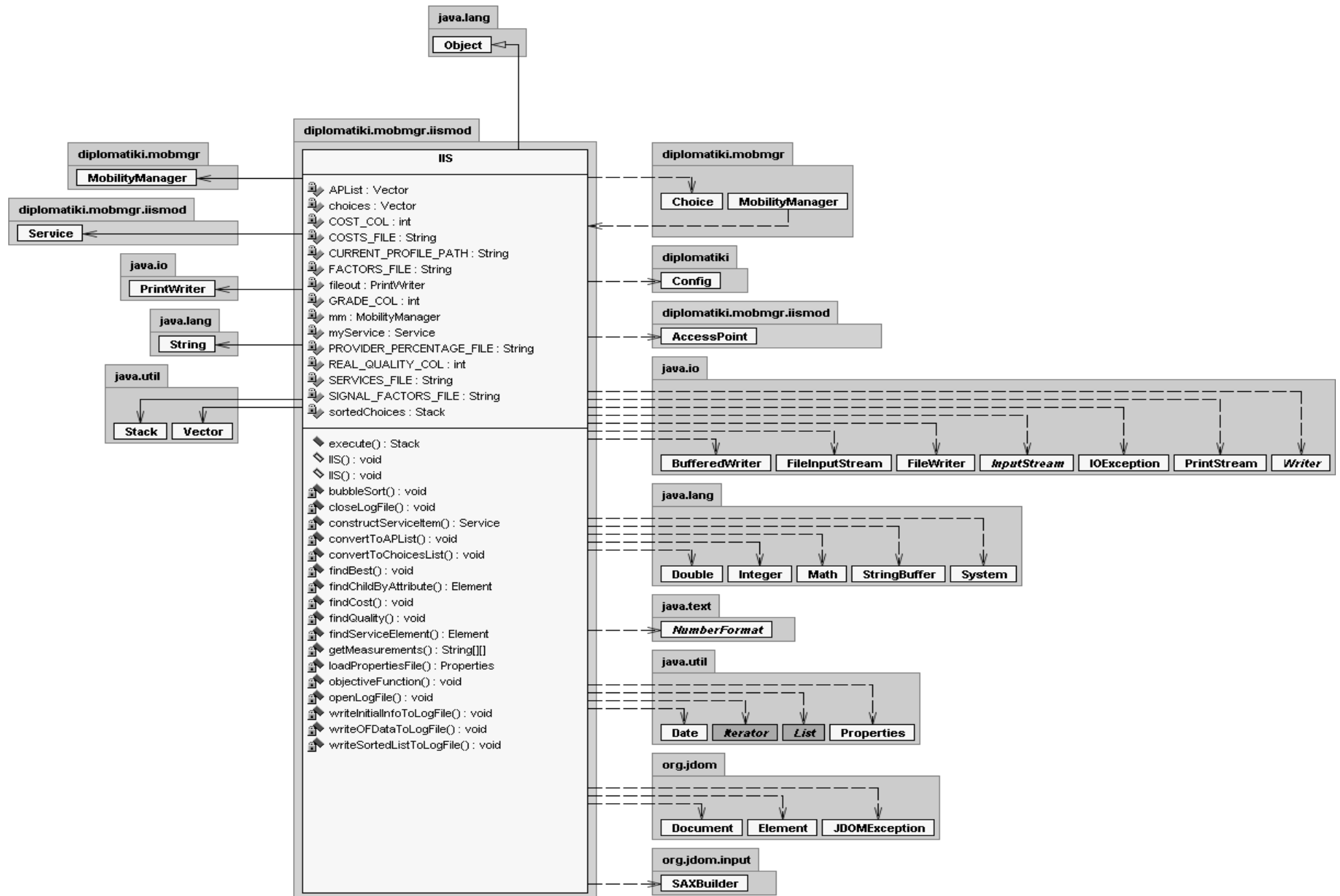
/**
630 * Prints the grade of an allocation choice and some intermediate calculation

```

```
        * data to the log file.
        *
        * @param ofData The piece of info that should be printed
        */
635  private void writeOFDataToLogFile (String ofData) {
        fileout.println (ofData);
    }

    /**
640  * Prints the sorted list of allocation choices to the log file.
        *
        * @param myArray The sorted table of allocation choices
        */
645  private void writeSortedListToLogFile (Choice [] myArray) {
        fileout.println (System.getProperty ("line.separator" ) +
            "Sorted Array of Choices: (" + myArray.length +
            " choices)" );
        fileout.println ("\tIP Address\tQoS Level\tGrade" );
        NumberFormat nf5 = NumberFormat .getInstance ();
650  nf5.setMaximumFractionDigits (5);
        for (int i = myArray.length - 1; i >= 0; i--) {
            fileout.println (myArray.length - i + ".\t" +
                myArray [i].getIpAddress () + "\t" +
                myArray [i].getQos () + "\t\t" +
655  nf5.format (myArray [i].getGrade ());
        }
    }

    /**
660  * Closes the log file.
        */
        private void closeLogFile () {
            fileout.flush ();
            fileout.close ();
665  }
    }
```





```

package diplomatiki .mobmgr .iismod ;

/**
 * <p>Title: Term Paper - <code>Service</code> class</p>
5  *
 * <p>Description: Objects of this class encapsulate information about a
 * requested service.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
10 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
15 */
public class Service {
    private String serviceName ;
    private int serviceCode ;
    private int maxUsersQoS ;
20 private int noQoSLevels ;
    private int[] bandwidth ;

    /**
     * Creates a new <code>Service</code> instance.
25 */
    public Service () {
    }

    /**
30 * Creates a new <code>Service</code> instance with the specified Name, Code,
 * Maximum Preferred QoS Level and Number of QoS Levels.
 *
 * @param serviceName The service's name
 * @param serviceCode The service's code
35 * @param maxUsersQoS The service's maximum QoS level (this is defined by the
 * user's preferences)
 * @param noQoSLevels The total number of QoS levels for this service
 */
    public Service (String serviceName , int serviceCode , int maxUsersQoS ,
40 int noQoSLevels ) {
        this .serviceName = serviceName ;
        this .serviceCode = serviceCode ;
        this .maxUsersQoS = maxUsersQoS ;
        this .noQoSLevels = noQoSLevels ;
45 bandwidth = new int [noQoSLevels + 1];
    }

    /**
50 * Sets the service's name.
 *
 * @param serviceName The new name
 */
    void setServiceName (String serviceName ) {
55 this .serviceName = serviceName ;
    }

    /**
 * Sets the service's code.
 *
60 * @param serviceCode The new code
 */
    void setServiceCode (int serviceCode ) {
        this .serviceCode = serviceCode ;
    }

65 /**
 * Sets the maximum preferred QoS level.
 *
 * @param maxUsersQoS The new maximum QoS level
70 */

```

```

    void setMaxUsersQoS (int maxUsersQoS ) {
        this.maxUsersQoS = maxUsersQoS ;
    }

75  /**
    * Sets the total number of QoS levels for this service.
    *
    * @param noQoSLevels The new total number of QoS levels
    */
80  void setNoQoSLevels (int noQoSLevels ) {
        this.noQoSLevels = noQoSLevels ;
    }

    /**
85  * Sets the bitrate for the specified QoS level. All bitrates are kept in a
    * table. For example, if the bitrate of the QoS level 2 is 8 Kbps, then this
    * method should be called as follows:
    *
    * <p><code> Service s = new Service();
90  * <br>s.setQoS(2, 8); </code></p>
    *
    * @param qosIndex The QoS level
    * @param bitRate The bitrate of the <code>qosIndex</code> QoS level
    */
95  void setQoS (int qosIndex , int bitRate) {
        bandwidth [qosIndex] = bitRate ;
    }

    /**
100  * Returns the service's name.
    *
    * @return The service's name
    */
    String getServiceName () {
105  return serviceName ;
    }

    /**
110  * Returns the service's code.
    *
    * @return The service's code
    */
    int getServiceCode () {
115  return serviceCode ;
    }

    /**
    * Returns the maximum QoS level (according to the user's preferences).
    *
120  * @return The maximum preferred QoS level
    */
    int getMaxUsersQoS () {
        return maxUsersQoS ;
    }

125  /**
    * Returns the total number of QoS levels for this service.
    *
    * @return The total number of QoS levels
    */
130  int getNoQoSLevels () {
        return noQoSLevels ;
    }

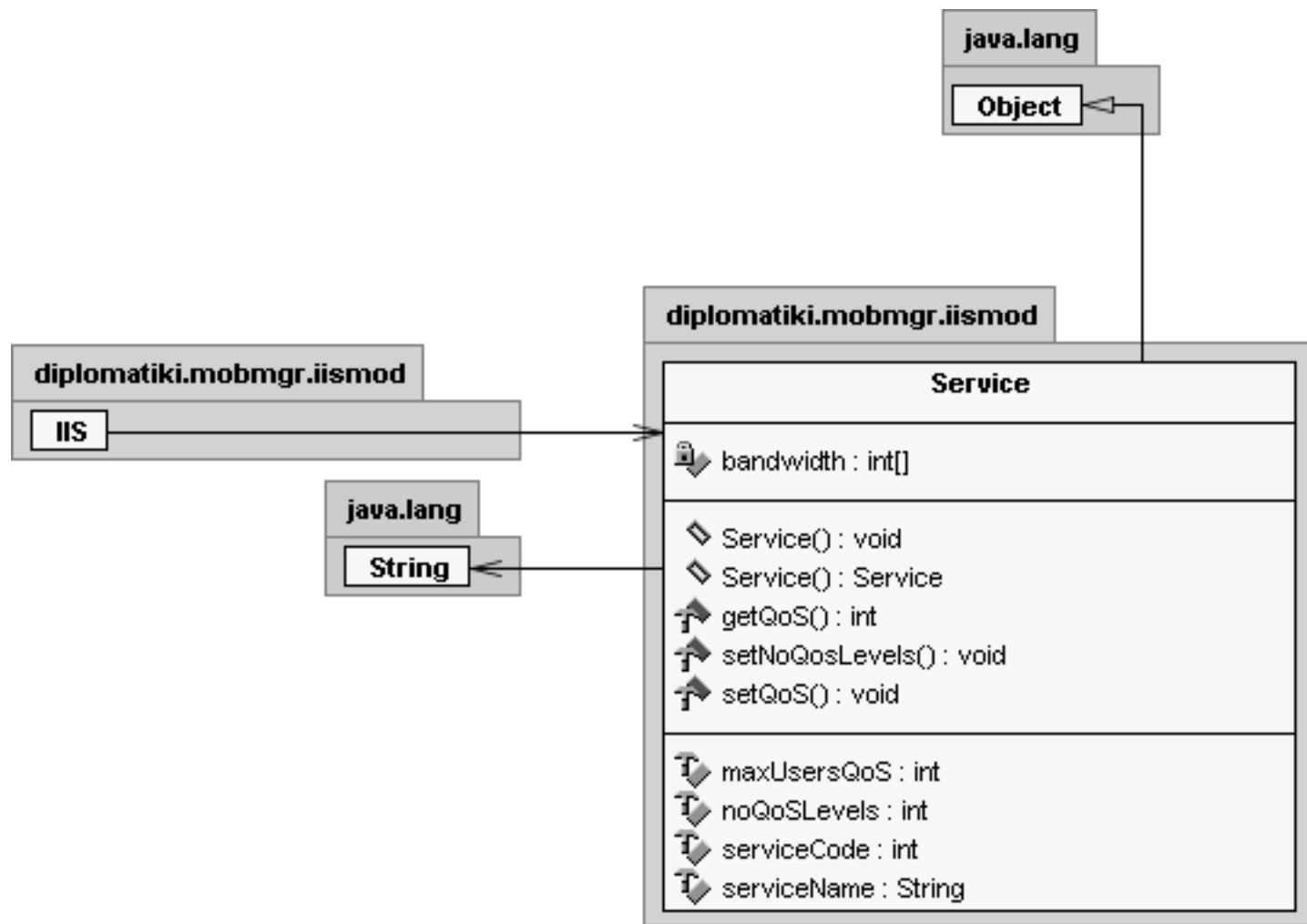
135  /**
    * Returns the bitrate of the specified QoS level.
    *
    * @param qosIndex The QoS level
    * @return The bitrate of the <code>qosIndex</code> QoS level
140  */

```

Service.java

```
int getQoS(int qosIndex) {  
    return bandwidth[qosIndex];  
}
```

145 }



**InterfaceAdaptation.java**

```
package diplomatiki .interad ;

import java.io.*;
import java.util.*;
5 import org.jdom.*;
import org.jdom.input.*;
import diplomatiki .*;
import diplomatiki .gui.*;
import diplomatiki .mobmgr.*;
10
/**
 * <p>Title: Term Paper - <code>InterfaceAdaptation</code> class</p>
 *
 * <p>Description: This is the main class implementing the Interface Adaptation
15 * (IA) module of a TMS. An <code>InterfaceAdaptation</code> instance runs in a
 * separate thred, so special care has been taken in order to make its methods
 * thread-safe.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
20 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
25 */
public class InterfaceAdaptation
    extends Thread {
    private final String MEASUREMENTS_PATH = Config.MEASUREMENTS_PATH ;
    private final long INITIAL_SLEEP = Config.INITIAL_SLEEP ;
30 private final long INTERVAL_SLEEP = Config.INTERVAL_SLEEP ;

    private MobilityManager mm = null;
    private String [][] currentMeasurements , previousMeasurements ;
    private SAXBuilder builder = null;
35 private Document doc = null;
    private int max_timer = 1;
    private int timer = -1;

    private InterfaceAdaptationLogFrame iaf = new InterfaceAdaptationLogFrame ();
40
    /**
     * Sets the table of current Measurements.
     *
     * @param currentMeasurements The new table of current Measurements
45 */
    public synchronized void setCurrentMeasurements (String [][]
                                                    currentMeasurements ) {
        this.currentMeasurements = currentMeasurements ;
    }
50
    /**
     * Sets the table of previous Measurements.
     *
     * @param previousMeasurements The new table of previous Measurements
55 */
    public void setPreviousMeasurements (String [][] previousMeasurements ) {
        this.previousMeasurements = previousMeasurements ;
    }
60
    /**
     * Returns the table of current Measurements.
     *
     * @return The table of current Measurements
     */
65 public synchronized String [][] getCurrentMeasurements () {
    return currentMeasurements ;
}

    /**
70 * Returns the table of previous Measurements.
```

**InterfaceAdaptation.java**

```

    *
    * @return The table of previous Measurements.
    */
75  public String [][] getPreviousMeasurements () {
    return previousMeasurements ;
}

/**
 * Creates a new <code>InterfaceAdaptation</code> instance.
80  */
public InterfaceAdaptation () {
}

/**
85  * Creates a new <code>InterfaceAdaptation</code> instance, giving it also a
 * reference to the TMS's Mobility Manager module. This constructor also
 * initiates (by invoking the <code>init()</code> method) and starts the IA
 * thread.
 *
90  * @param mm A reference to the Mobility Manager module
 */
public InterfaceAdaptation (MobilityManager mm) {
    this.mm = mm;
    max_timer = init ();
95  start ();
}

/**
100  * The entry point of the IA thread. This thread stays alive until the whole
 * TMS is terminated.
 */
public void run () {
    setCurrentMeasurements (measure ());
    setPreviousMeasurements (getCurrentMeasurements ());
105
    /*
    try {
        Thread.currentThread().sleep(INITIAL_SLEEP);
    }
110  catch (InterruptedException ex) {
        ex.printStackTrace();
    }
    */

115  while (true) {
    try {
        Thread.currentThread ().sleep (INTERVAL_SLEEP );
    }
    catch (InterruptedException ex) {
120  ex.printStackTrace ();
    }

    setPreviousMeasurements (getCurrentMeasurements ());
    setCurrentMeasurements (measure ());
125  if (!checkForNewAP ()) {
        checkForAPLoss ();
    }
}
}

130  /**
 * Retrieves the current Measurements from the measurements XML file.
 *
 * @return A table with the current Measurements
 */
135  public String [][] measure () {
    timer++;

    //System.out.println("New measurements acquired, timer = " + timer);
140  iaf.printTimer (timer);
}

```

```

Element curmsr = findChildByAttribute (doc.getRootElement (), "timer",
                                     "" + (timer % max_timer));
List entries = curmsr.getChildren ();
145 String [][] measurements = new String [entries.size ()][5];
int row = 0;
for (Iterator i = entries.iterator (); i.hasNext (); ) {
    Element el = (Element) i.next ();
    measurements [row][0] = el.getAttributeValue ("ip");
150 measurements [row][1] = el.getAttributeValue ("providerCode" );
    measurements [row][2] = el.getAttributeValue ("technology" );
    measurements [row][3] = el.getAttributeValue ("signal" );
    measurements [row][4] = el.getAttributeValue ("availBandwidth" );
    row++;
155 }

return measurements ;
}

160 /**
 * Checks if an access point that is in use by the terminal has disappeared.
 */
private void checkForAPLoss () {
165     synchronized (mm.getRunningServices ()) {

        boolean loss = true;
        for (int i = 0; i < mm.getRunningServices ().size (); i++) {
            loss = true;
            Choice ch = (Choice) mm.getRunningServices ().get(i);
170             for (int j = 0; j < getCurrentMeasurements ().length; j++) {
                 if (getCurrentMeasurements ()[j][0].equals (ch.getIpAddress ())) {
                     loss = false;
                     break;
                 }
            }
175             if (loss) {
                //System.out.println(i+" => loss found for service "+
                //                ch.getServiceCode());
                InfoStruct myInfo = mm.executeIIS (ch.getServiceCode (), false, i);
180                 //Workaround in case of removing the examined service
                 if (myInfo == null) {
                     i--;
                 }
                mm.getMenuFrame ().
185                 refreshAppropriateConnectionInfo (ch.getServiceCode (), myInfo);
            } //end of if(loss)
        } //end of for(i)

    } //end of synchronized
190 }

/**
 * Checks if a new access point has appeared. If so, this method returns
 * <code>true</code>.
195 *
 * @return <code>True</code> if a new access point has appeared, otherwise
 * <code>false</code>
 */
private boolean checkForNewAP () {
200     boolean found = false;
    boolean refresh = false;
    for (int i = 0; i < getCurrentMeasurements ().length; i++) {
        found = false;
        for (int j = 0; j < getPreviousMeasurements ().length; j++) {
205             if (getCurrentMeasurements ()[i][0].
                 equals (getPreviousMeasurements ()[j][0])) {
                     found = true;
                     break;
                 }
            }
210         }
    }
}

```

```

        if (!found) {
            refresh = true;
            break;
        }
215     }
        if (refresh) {
            mm.reExamineAllRunningServices ();
        }
        return refresh;
220     }

/**
 * Finds the appropriate child element of the given element, in an XML
 * document, by using the given attribute name/attribute value pair as a
225 * search criteria.
 *
 * @param parentEl The parent element of the element that we search for
 * @param attrName The name of the attribute
 * @param attrValue The value of the attribute
230 * @return The appropriate XML element
 */
private Element findChildByAttribute (Element parentEl, String attrName,
                                     String attrValue) {
    List children = parentEl.getChildren ();
235     for (Iterator i = children.iterator (); i.hasNext (); ) {
        Element el = (Element) i.next ();
        if (el.getAttributeValue (attrName).equals (attrValue)) {
            return el;
        }
240     }

    return null;
}

/**
 * Prints the specified table of Measurements to the system's console.
 *
 * @param measurements The table of Measurements to be printed
 */
250 public void printMeasurements (String [][] measurements) {
    System.out.println (
        "IP Address \t Provider \t Technology \t Signal Strength \t " +
        "Available Bandwidth" );
    for (int i = 0; i < measurements.length; i++) {
255         System.out.println (measurements [i][0] + "\t" + measurements [i][1] + "\t" +
            measurements [i][2] + "\t" + measurements [i][3] + "\t" +
            measurements [i][4]);
    }
}

/**
 * Finds out the maximum value of the <code>timer</code> instance variable
 * from the measurements XML document, and returns this value.
 *
265 * @return The maximum value of the <code>timer</code> instance variable
 */
private int init () {
    builder = new SAXBuilder ();
    try {
270         doc = builder.build (new FileInputStream (MEASUREMENTS_PATH));
    }
    catch (IOException ex) {
        ex.printStackTrace ();
    }
    catch (JDOMException ex) {
275         ex.printStackTrace ();
    }

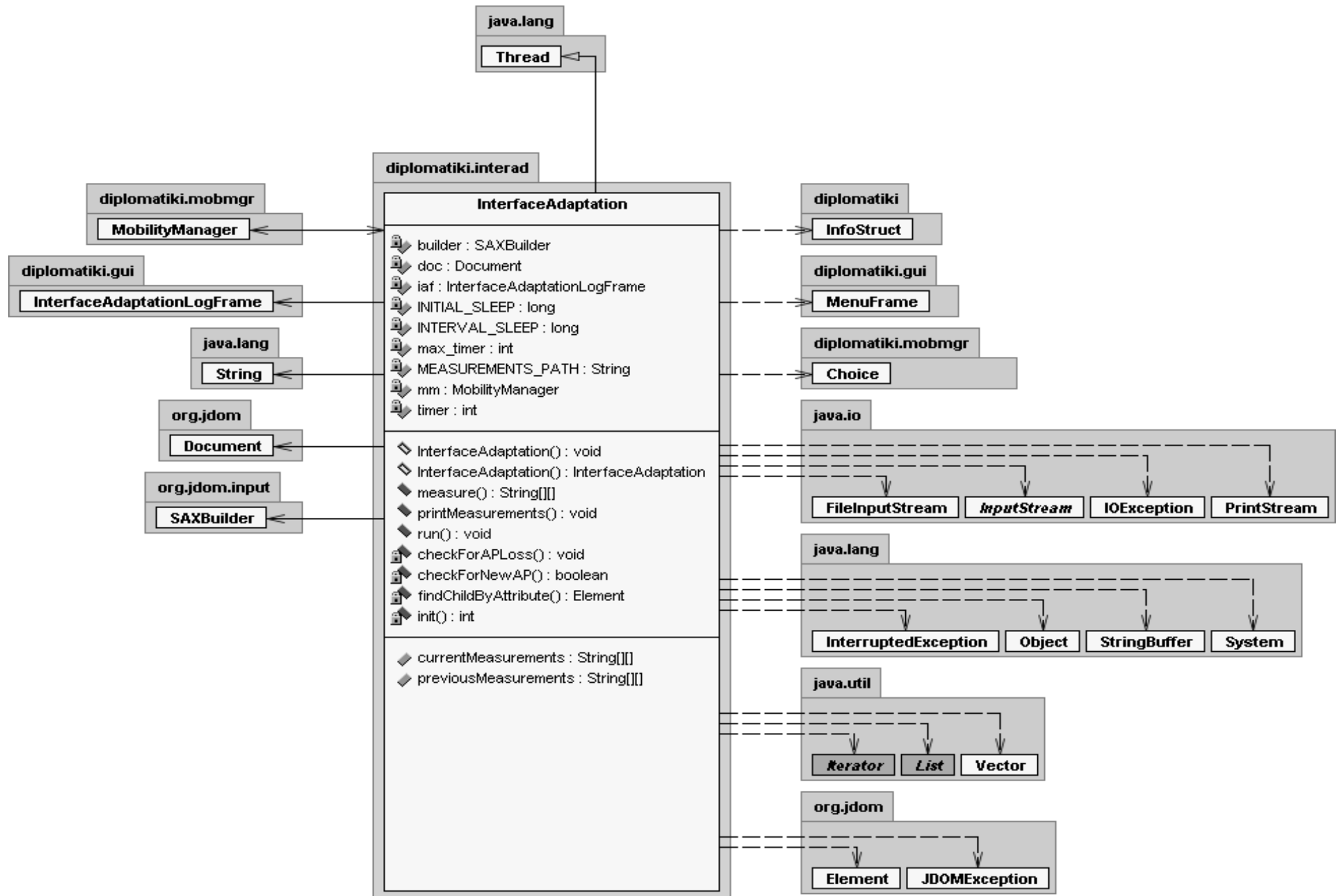
    return doc.getRootElement ().getChildren ().size ();
280 }

```



```
InterfaceAdaptation.java
```

```
}
```



**InterfaceAdaptationDeprecated.java**

```
package diplomatiki .interad ;

import java.io.*;
import java.util.*;
5 import org.jdom.*;
import org.jdom.input.*;
import diplomatiki .Config;

/**
10 * <p>Title: Term Paper - <code>InterfaceAdaptationDeprecated</code> class</p>
*
* <p>Description: This class has been used in order to implement the Interface
* Adaptation (IA) module in the early stages of development. It uses a
* diferrent method to retrieve link-layer measurements and is no longer
15 * supported. It doesn't check for access point appearance or disapearance.</p>
*
* <p>Copyright: Copyright (c) 2005</p>
*
* <p>Company: Jadacode</p>
20 *
* @author Evgenia Adamopoulou - Konstantinos Demestichas
* @version 1.0
* @deprecated This class is no longer used by the Terminal Management System
* (TMS)
25 */
public class InterfaceAdaptationDeprecated {
    private final String MEASUREMENTS_PATH = Config.MEASUREMENTS_PATH ;

    private SAXBuilder builder = null;
30 private Document doc = null;
private int max_timer = 1;
private int timer = -1;

/**
35 * Creates a new <code>InterfaceAdaptation</code> instance. It also invokes
* the <code>init()</code> method.
*/
public InterfaceAdaptationDeprecated () {
40     max_timer = init();
}

/**
* Retrieves the current Measurements from the measurements XML file.
*
45 * @param newmsr If <code>>true</code> then new Measurements should be
* retrieved. The <code>timer</code> instance variable is increased only if
* newmsr is <code>>true</code>
* @return A table with the current Measurements
*/
50 public String [][] measure (boolean newmsr) {
    if ( (newmsr) || (timer < 0)) {
        timer++;
    }

    Element curmsr = findChildByAttribute (doc.getRootElement (), "timer",
65     "" + (timer % max_timer));
    List entries = curmsr.getChildren ();
    String [][] measurements = new String [entries.size ()][5];
    int row = 0;
    for (Iterator i = entries.iterator (); i.hasNext (); ) {
        Element el = (Element) i.next ();
        measurements [row][0] = el.getAttributeValue ("ip");
        measurements [row][1] = el.getAttributeValue ("providerCode" );
        measurements [row][2] = el.getAttributeValue ("technology" );
65     measurements [row][3] = el.getAttributeValue ("signal" );
        measurements [row][4] = el.getAttributeValue ("availBandwidth" );
        row++;
    }

70     return measurements ;
}
```

```

    }

    /**
     * Increases the <code>timer</code> instance variable by 1.
75     */
    public void increaseTimer () {
        timer++;
    }

80     /**
     * Finds out the maximum value of the <code>timer</code> instance variable
     * from the measurements XML document, and returns this value.
     *
     * @return The maximum value of the <code>timer</code> instance variable
85     */
    private int init () {
        builder = new SAXBuilder ();
        try {
            doc = builder .build (new FileInputStream (MEASUREMENTS_PATH ));
90        }
        catch (IOException ex) {
            ex.printStackTrace ();
        }
        catch (JDOMException ex) {
95            ex.printStackTrace ();
        }

        return doc.getRootElement ().getChildren ().size ();
    }

100    /**
     * Finds the appropriate child element of the given element, in an XML
     * document, by using the given attribute name/attribute value pair as a
     * search criteria.
105     *
     * @param parentEl The parent element of the element that we search for
     * @param attrName The name of the attribute
     * @param attrValue The value of the attribute
     * @return The appropriate XML element
110     */
    private Element findChildByAttribute (Element parentEl , String attrName ,
                                         String attrValue) {
        List children = parentEl .getChildren ();
        for (Iterator i = children .iterator (); i.hasNext (); ) {
115            Element el = (Element) i.next ();
            if (el.getAttributeValue (attrName) .equals (attrValue) ) {
                return el;
            }
        }

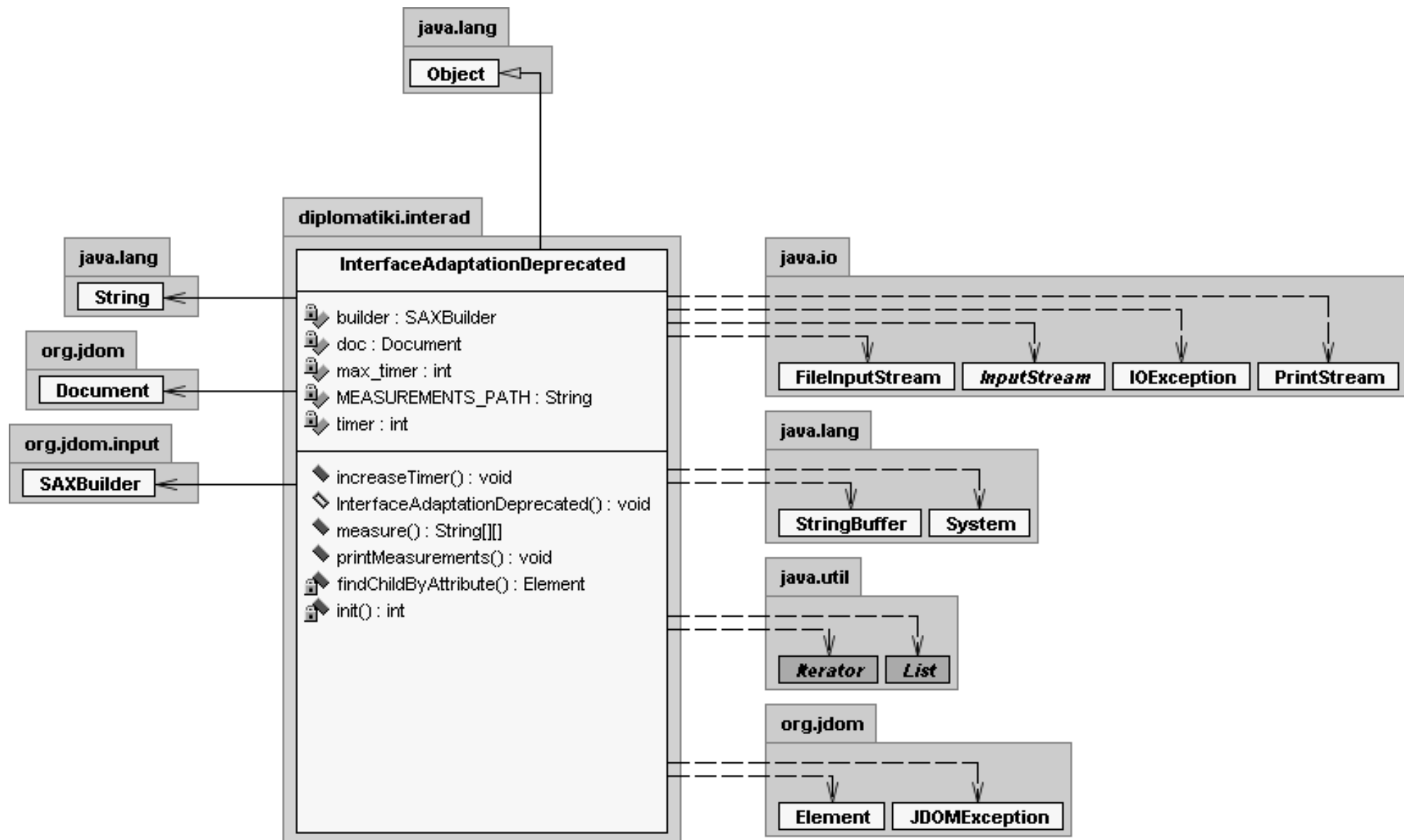
120        return null;
    }

    /**
125     * Prints the specified table of Measurements to the system's console.
     *
     * @param measurements The table of Measurements to be printed
     */
    public void printMeasurements (String [][] measurements ) {
130        System.out.println (
            "IP Address \t Provider \t Technology \t Signal Strength \t "      +
            "Available Bandwidth" );
        for (int i = 0; i < measurements .length; i++) {
            System.out.println (measurements [i][0] + "\t" + measurements [i][1] + "\t" +
135                measurements [i][2] + "\t" + measurements [i][3] + "\t" +
                measurements [i][4]);
        }
    }

140 }

```





**ApplicationsPanel.java**

```
package diplomatiki .gui;

import java.awt.*;
import javax.swing.*;
5 import javax.swing.border.*;
import java.io.*;
import java.util.*;
import diplomatiki .Config;

10 /**
 * <p>Title: Term Paper - <code>ApplicationsPanel</code> class</p>
 *
 * <p>Description: This is the applications panel of the GUI. In this panel,
 * details for all running services are presented.</p>
15 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
20 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
public class ApplicationsPanel
    extends JPanel {
25     private final String SERVICES_CONFIG_FILE = Config.SERVICES_CONFIG_FILE ;
    private final String TECHNOLOGIES_FILE = Config.TECHNOLOGIES_FILE ;
    private final String PROVIDERS_FILE = Config.PROVIDERS_FILE ;

    /**
30     * Creates a new <code>ApplicationsPanel</code> instance.
    */
    public ApplicationsPanel () {
    }

35     /**
     * Creates a new <code>ApplicationsPanel</code> instance with the specified
     * parent frame and previous panel.
     *
     * @param myFrame The frame which this panel belongs to
40     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
     * wants to return to the previous panel
     */
    public ApplicationsPanel (JFrame myFrame, JPanel previousPanel ) {
45         super (myFrame, previousPanel );

        this.remove (centerPanel );
        northPanel .setPreferredSize (new Dimension (400, 210));
        northPanel .setLayout (new FlowLayout ());
50         northPanel .setBorder (
            new TitledBorder (new EtchedBorder (), "Running Services" ,
                TitledBorder .DEFAULT_JUSTIFICATION ,
                TitledBorder .DEFAULT_POSITION ,
55                 new java.awt.Font ("Microsoft Sans Serif" , 0, 11),
                new Color (10, 92, 174)));

        Properties serviceConfig = loadPropertiesFile (SERVICES_CONFIG_FILE );
        Properties technologyConfig = loadPropertiesFile (TECHNOLOGIES_FILE );
        Properties providerConfig = loadPropertiesFile (PROVIDERS_FILE );

60         RunningServicesInfoPanel rsip = new RunningServicesInfoPanel ();

        synchronized ( ( (MenuFrame) myFrame).getMobilityManager ().
            getRunningServices ()) {

65             for (int i = 0; i < ( (MenuFrame) myFrame).getMobilityManager ().
                getRunningServices ().size (); i++) {
                diplomatiki .mobmgr.Choice ch = (diplomatiki .mobmgr.Choice)
                    ( (MenuFrame) myFrame).getMobilityManager ().
70                 getRunningServices ().get (i);
```

```

        rsip.getLabel (4 * i + 1).
            setText (serviceConfig .getProperty (" " + ch.getServiceCode (),
                                                "Unknown" ));

        rsip.getLabel (4 * i + 2).
75         setText (technologyConfig .getProperty (" " + ch.getAccessTechnology (),
                                                "Unknown" ));

        rsip.getLabel (4 * i + 3).
            setText (providerConfig .getProperty (" " + ch.getProviderCode (),
                                                " " + ch.getProviderCode ());
80     rsip.getLabel (4 * i + 4).setText (" " + ch.getQos ());
    }

}

85     northPanel .add(rsip);
}

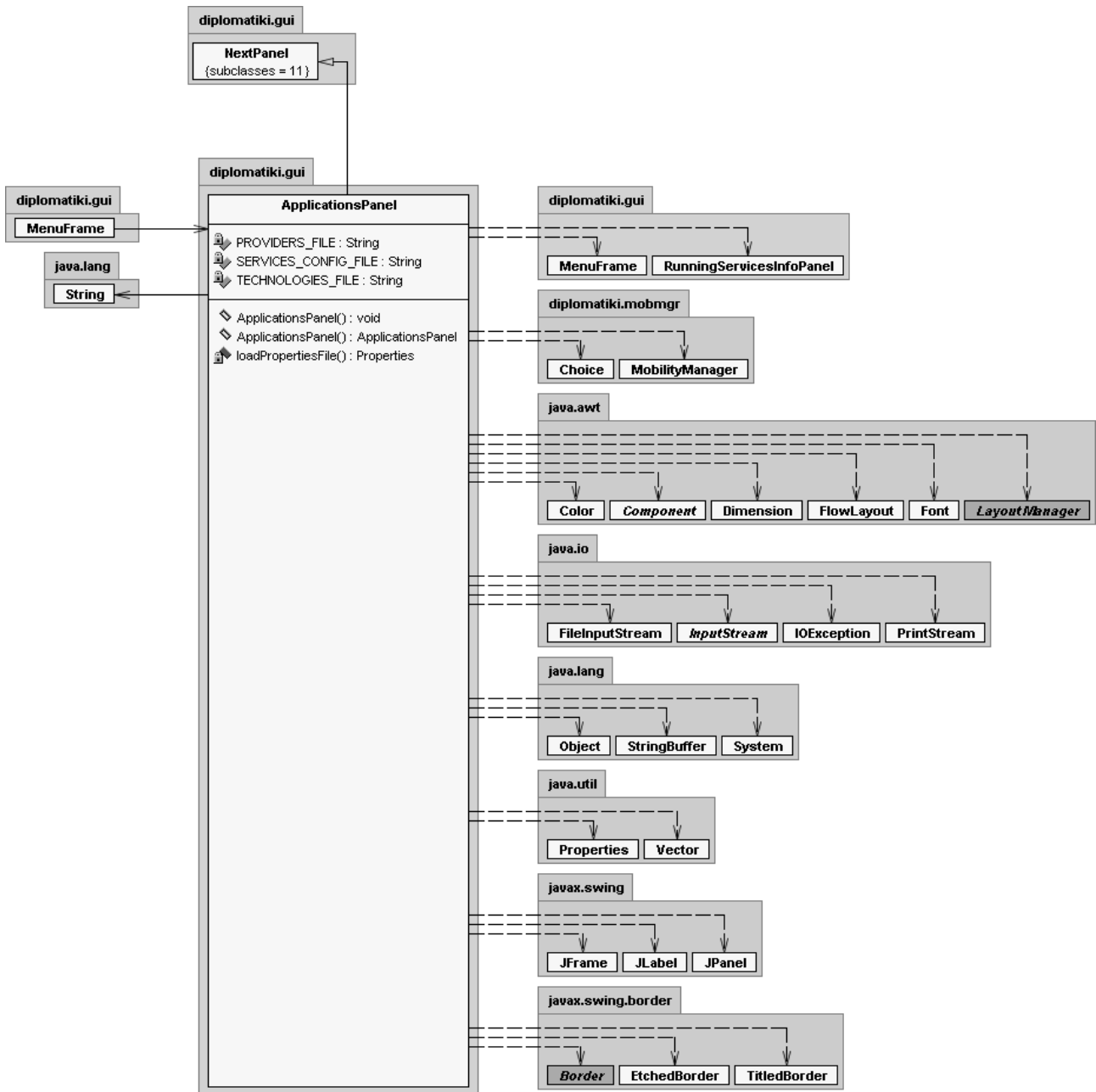
/**
 * Loads a Properties file.
90  *
 * @param filePath The path of the properties file
 * @return A <code>Properties</code> instance that represents the properties
 *         file's contents
 */
95  private Properties loadPropertiesFile (String filePath) {
    Properties config = new Properties ();
    InputStream input = null;
    try {
100     input = new FileInputStream (filePath);
        config.load(input);
    }
    catch (IOException ex) {
        System.err.println ("Cannot open and load properties file." );
    }
105  finally {
        try {
            if (input != null) {
                input.close ();
            }
110     }
        catch (IOException ex) {
            System.err.println ("Cannot close properties file." );
        }
115  }

    return config;
}

}
120

```





```

package diplomatiki .gui;

import java.util.*;
import javax.swing.JFrame;
5 import diplomatiki .*;

/**
 * <p>Title: Term Paper - <code>CalendarTitle</code> class</p>
 *
10 * <p>Description: An instance of this class runs as a separate thread and is
 * responsible for updating the frame's title bar with the current date and
 * time.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class CalendarTitle
    implements Runnable {
    private JFrame myFrame;
    private Thread runner;
25 private boolean okToRun = true;

    //private String greeting = "";
    private String greeting = "HI " + Config.USERNAME.toUpperCase () + "! ";

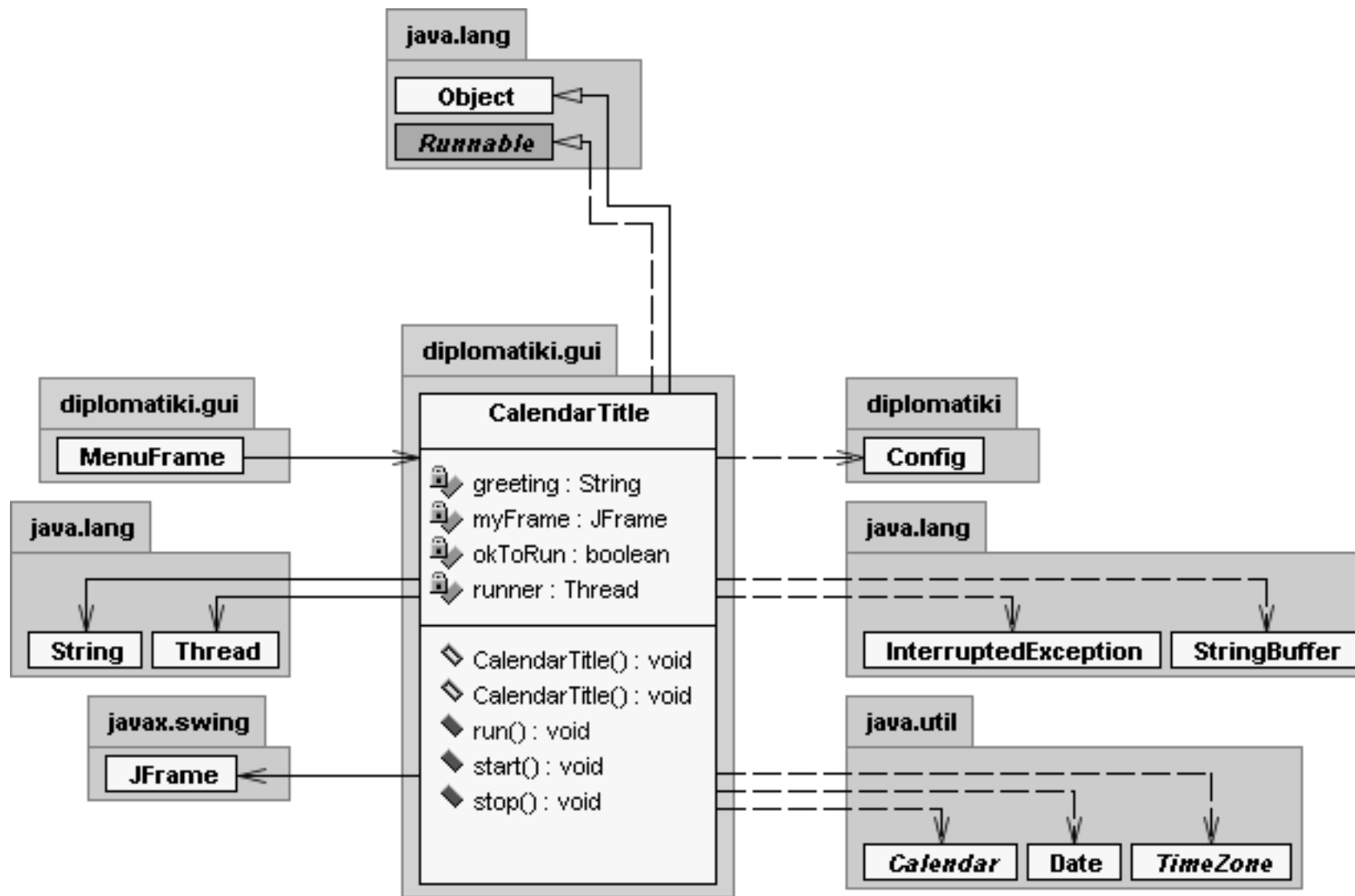
30 /**
 * Creates a new <code>CalendarTitle</code> instance. It also starts the
 * <code>CalendarTitle</code> thread.
 */
    public CalendarTitle () {
35         myFrame = new JFrame ();
        if (runner == null) {
            runner = new Thread (this);
        }
    }

40 /**
 * Creates a new <code>CalendarTitle</code> instance. It also starts the
 * <code>CalendarTitle</code> thread. The time will be shown on
 * <code>myFrame</code>'s title bar.
45 *
 * @param myFrame The title of this <code>JFrame</code> will show the current
 * date and time
 */
    public CalendarTitle (JFrame myFrame) {
50         this.myFrame = myFrame;
        if (runner == null) {
            runner = new Thread (this);
        }
    }

55 /**
 * Entry point of this thread. This thread stays alive until the
 * <code>stop()</code> method is invoked.
 */
60 public void run () {
    while (okToRun) {
        Date rightNow = Calendar.getInstance (TimeZone.getDefault ()).getTime ();
        myFrame.setTitle (greeting + rightNow.toString ());
        try {
65             runner.sleep (1000);
        }
        catch (InterruptedException ex) {
            ex.printStackTrace ();
        }
70     }
}

```

```
    }  
  
    /**  
     * Starts the <code>CalendarTitle</code> thread.  
75     */  
    public void start () {  
        if (runner != null) {  
            runner.start ();  
        }  
80    }  
  
    /**  
     * Stops the <code>CalendarTitle</code> thread.  
     */  
85    public void stop () {  
        okToRun = false;  
    }  
  
90 }
```



```

package diplomatiki .gui;

import java.awt.*;
import javax.swing.*;

5
/**
 * <p>Title: Term Paper - <code>ConnectionInfoPanel</code> class</p>
 *
 * <p>Description: This panel presents the details of an open connection, e.g.
10 * the access point's IP address, access technology and provider name, and the
 * QoS level of the running service. It is used as part of an
 * <code>InfoPanel</code>.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class ConnectionInfoPanel
    extends JPanel {
    JLabel jLabel1 = new JLabel ();
    JLabel jLabel2 = new JLabel ();
25 JLabel jLabel3 = new JLabel ();
    JLabel jLabel4 = new JLabel ();
    JLabel jLabel5 = new JLabel ();
    JLabel jLabel6 = new JLabel ();
    JLabel technologyLabel = new JLabel ();
    JLabel providerLabel = new JLabel ();
    JLabel qosLabel = new JLabel ();
    JLabel ipLabel = new JLabel ();

    /**
35 * Creates a new <code>ConnectionInfoPanel</code> instance.
    */
    public ConnectionInfoPanel () {
    }

40 /**
 * Creates a new <code>ConnectionInfoPanel</code> instance. This panel will
 * present the info specified by the constructor's parameters.
 *
 * @param technology The access technology
45 * @param provider The provider's name
 * @param qos The QoS level
 * @param ip The access point's IP address
    */
    public ConnectionInfoPanel (String technology , String provider , String qos ,
50 String ip) {
        try {
            jbInit ();
        }
        catch (Exception ex) {
55 ex.printStackTrace ();
        }

        technologyLabel .setText (technology);
        providerLabel .setText (provider);
60 qosLabel .setText (qos);
        ipLabel .setText (ip);
    }

    /**
65 * Builds the panel.
 *
 * @throws Exception An exception is thrown if the construction of the panel
 * is not successful
    */
70 void jbInit () throws Exception {

```

```

    this.setBackground (UIManager.getColor ("InternalFrame.activeTitleGradient" ));
    this.setMinimumSize (new Dimension (300, 140));
    this.setMaximumSize (new Dimension (300, 140));
    this.setPreferredSize (new Dimension (300, 140));
75    this.setLayout (null);

    jLabel1.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
    jLabel1.setText ("The requested service has been provided with the " );
    jLabel1.setBounds (new Rectangle (9, 0, 349, 21));
80

    jLabel2.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
    jLabel2.setText ("followingspecifications:" );
    jLabel2.setBounds (new Rectangle (10, 20, 156, 15));

85    jLabel3.setFont (new java.awt.Font ("Palatino Linotype" , 1, 13));
    jLabel3.setText ("Technology:" );
    jLabel3.setBounds (new Rectangle (10, 38, 80, 26));

    jLabel4.setFont (new java.awt.Font ("Palatino Linotype" , 1, 13));
90    jLabel4.setText ("Provider:" );
    jLabel4.setBounds (new Rectangle (11, 67, 61, 15));

    jLabel5.setFont (new java.awt.Font ("Palatino Linotype" , 1, 13));
    jLabel5.setText ("QoS Level:" );
95    jLabel5.setBounds (new Rectangle (11, 91, 73, 15));

    jLabel6.setFont (new java.awt.Font ("Palatino Linotype" , 1, 13));
    jLabel6.setText ("IP Address:" );
100    jLabel6.setBounds (new Rectangle (12, 115, 76, 15));

    technologyLabel .setFont (new java.awt.Font ("Dialog" , 1, 15));
    technologyLabel .setForeground (Color.red);
    technologyLabel .setText ("UMTS");
    technologyLabel .setBounds (new Rectangle (100, 42, 180, 15));
105

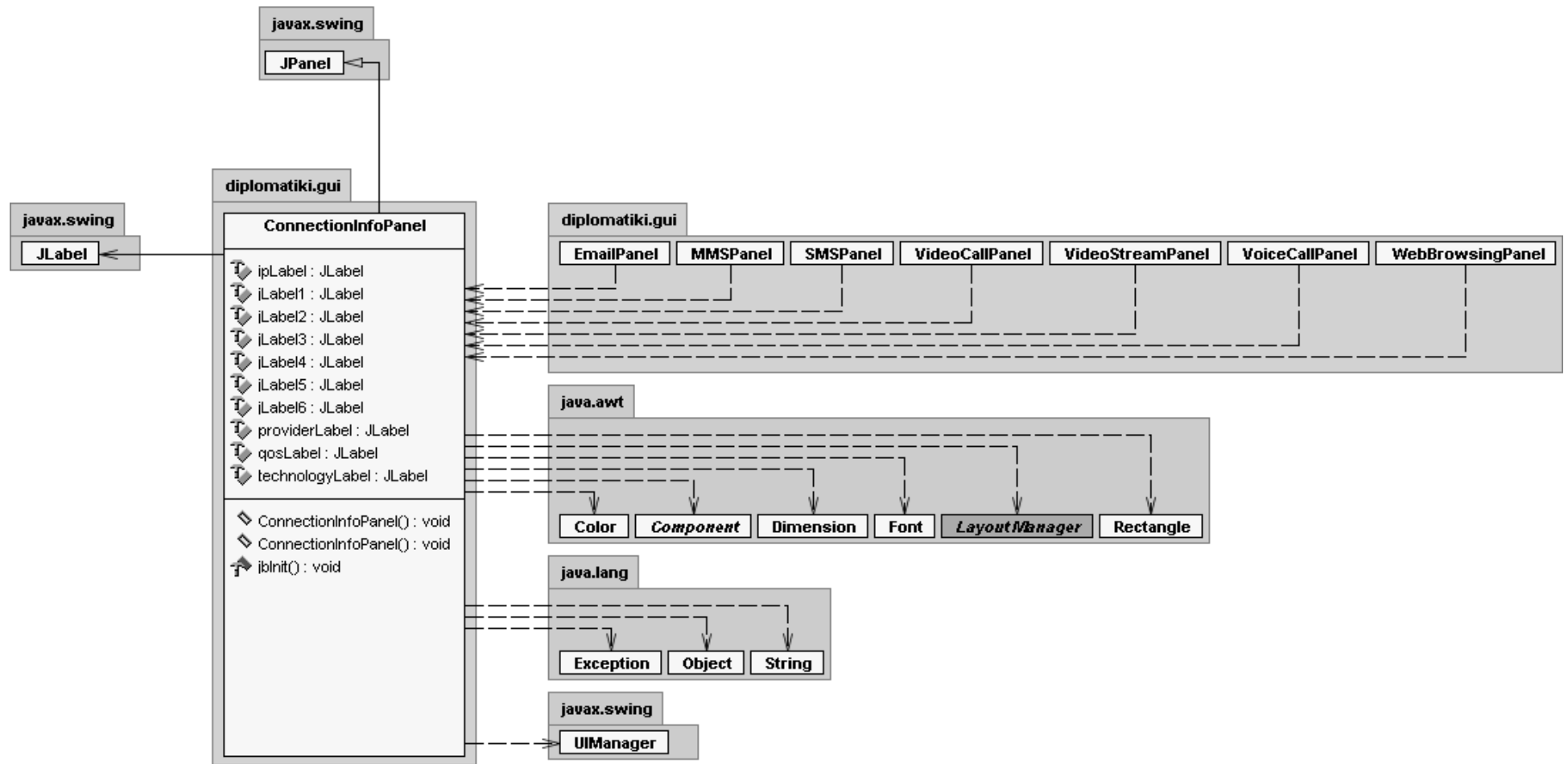
    providerLabel .setFont (new java.awt.Font ("Dialog" , 1, 15));
    providerLabel .setForeground (Color.red);
    providerLabel .setText ("Cosmote" );
    providerLabel .setBounds (new Rectangle (100, 65, 180, 15));
110

    qosLabel .setFont (new java.awt.Font ("Dialog" , 1, 15));
    qosLabel .setForeground (Color.red);
    qosLabel .setText ("2");
    qosLabel .setBounds (new Rectangle (120, 88, 60, 15));
115

    ipLabel .setFont (new java.awt.Font ("Dialog" , 1, 15));
    ipLabel .setForeground (Color.red);
    ipLabel .setText ("147.102.210.13" );
    ipLabel .setBounds (new Rectangle (100, 113, 180, 15));
120

    this.add (jLabel1 , null);
    this.add (jLabel2 , null);
    this.add (jLabel5 , null);
    this.add (jLabel3 , null);
125    this.add (jLabel4 , null);
    this.add (jLabel6 , null);
    this.add (technologyLabel , null);
    this.add (providerLabel , null);
    this.add (qosLabel , null);
130    this.add (ipLabel , null);
}
}

```



```

package diplomatiki .gui;

import java.awt.*;
import java.awt.event.*;
5 import javax.swing.*;
import javax.swing.border.*;
import diplomatiki .Config;

/**
10 * <p>Title: Term Paper - <code>EditPanel</code> class</p>
*
* <p>Description: This panel appears when the Edit Profile button (the center
* button) of the main menu is pressed.</p>
*
15 * <p>Copyright: Copyright (c) 2005</p>
*
* <p>Company: Jadacode</p>
*
* @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
*/
public class EditPanel
    extends JPanel {
    final String CURRENT_ICON_PATH =
25 Config.EDIT_CURRENT_ICON_PATH ;
    final String CURRENT_SELECTED_ICON_PATH =
    Config.EDIT_CURRENT_SELECTED_ICON_PATH ;
    final String LOW_COST_ICON_PATH =
    Config.EDIT_LOW_COST_ICON_PATH ;
30 final String LOW_COST_SELECTED_ICON_PATH =
    Config.EDIT_LOW_COST_SELECTED_ICON_PATH ;
    final String HIGH_QUALITY_ICON_PATH =
    Config.EDIT_HIGH_QUALITY_ICON_PATH ;
    final String HIGH_QUALITY_SELECTED_ICON_PATH =
35 Config.EDIT_HIGH_QUALITY_SELECTED_ICON_PATH ;
    final String BALANCED_ICON_PATH =
    Config.EDIT_BALANCED_ICON_PATH ;
    final String BALANCED_SELECTED_ICON_PATH =
    Config.EDIT_BALANCED_SELECTED_ICON_PATH ;
40
    final int CURRENT_CODE = Config.PROFILE_CURRENT_CODE ;
    final int LOW_COST_CODE = Config.PROFILE_LOW_COST_CODE ;
    final int HIGH_QUALITY_CODE = Config.PROFILE_HIGH_QUALITY_CODE ;
    final int BALANCED_CODE = Config.PROFILE_BALANCED_CODE ;
45
    ExtendedButton currentButton , lowCostButton , highQualityButton ,
        balancedButton ;
    ProfilePanel pp;

50 /**
* Creates a new <code>EditPanel</code> instance.
*/
    public EditPanel () {
    }
55
    /**
* Creates a new <code>EditPanel</code> instance with the specified
* parent frame and previous panel.
*
* @param myFrame The frame which this panel belongs to
* @param previousPanel The previous panel that was visible in the
* <code>myFrame</code> frame. This reference is very useful if the user
* wants to return to the previous panel
*/
65 public EditPanel (JFrame myFrame , JPanel previousPanel ) {
    super (myFrame , previousPanel );

    GridLayout1 .setColumns (1);
    GridLayout1 .setRows (4);
70

```



```

northPanel .setBorder (
    new TitledBorder (new EtchedBorder (), "Choose a profile" ,
        TitledBorder .DEFAULT_JUSTIFICATION ,
        TitledBorder .DEFAULT_POSITION ,
75         new java .awt .Font ("Microsoft Sans Serif" , 0, 11),
        new Color (10, 92, 174)));
centerPanel .setBorder (null);

currentButton = new ExtendedButton (CURRENT_ICON_PATH ,
80     CURRENT_SELECTED_ICON_PATH );
lowCostButton = new ExtendedButton (LOW_COST_ICON_PATH ,
    LOW_COST_SELECTED_ICON_PATH );
highQualityButton = new ExtendedButton (HIGH_QUALITY_ICON_PATH ,
85     HIGH_QUALITY_SELECTED_ICON_PATH );
balancedButton = new ExtendedButton (BALANCED_ICON_PATH ,
    BALANCED_SELECTED_ICON_PATH );

currentButton .
    addActionListener (new EditPanel_currentButton_actionAdapter (this));
90 lowCostButton .
    addActionListener (new EditPanel_lowCostButton_actionAdapter (this));
highQualityButton .
    addActionListener (new EditPanel_highQualityButton_actionAdapter (this));
balancedButton .
95     addActionListener (new EditPanel_balancedButton_actionAdapter (this));

northPanel .add (currentButton );
northPanel .add (lowCostButton );
northPanel .add (highQualityButton );
100 northPanel .add (balancedButton );
}

/**
 * Constructs and shows a <code>ProfilePanel</code>.
105 *
 * @param profileCode The unique code of the profile (e.g. "Current Profile",
 * "Low Cost Profile", etc.) that should be presented.
 */
private void createProfilePanel (int profileCode ) {
110     pp = new ProfilePanel (myFrame , this, profileCode );

    //Show the ProfilePanel
    this .setVisible (false);
    myFrame .setContentPane (pp);
115     myFrame .setVisible (false);
    myFrame .setVisible (true);
    pp .setVisible (true);
}

120 /**
 * Action performed when the "Current" profile button is pressed.
 *
 * @param e ActionEvent
 */
125 void currentButton_actionPerformed (ActionEvent e) {
    createProfilePanel (CURRENT_CODE );
}

130 /**
 * Action performed when the "Low Cost" profile button is pressed.
 *
 * @param e ActionEvent
 */
135 void lowCostButton_actionPerformed (ActionEvent e) {
    createProfilePanel (LOW_COST_CODE );
}

140 /**
 * Action performed when the "High Quality" profile button is pressed.
 *

```

**EditPanel.java**

```
    * @param e ActionEvent
    */
    void highQualityButton_actionPerformed (ActionEvent e) {
145     createProfilePanel (HIGH_QUALITY_CODE );
    }

    /**
    * Action performed when the "Balanced" profile button is pressed.
    *
150    * @param e ActionEvent
    */
    void balancedButton_actionPerformed (ActionEvent e) {
        createProfilePanel (BALANCED_CODE );
155    }
}

class EditPanel_currentButton_actionAdapter
160     implements java.awt.event.ActionListener {
    EditPanel adaptee ;

    EditPanel_currentButton_actionAdapter (EditPanel adaptee) {
        this.adaptee = adaptee ;
165    }

    public void actionPerformed (ActionEvent e) {
        adaptee.currentButton_actionPerformed (e);
    }
}
170

class EditPanel_lowCostButton_actionAdapter
    implements java.awt.event.ActionListener {
    EditPanel adaptee ;

175    EditPanel_lowCostButton_actionAdapter (EditPanel adaptee) {
        this.adaptee = adaptee ;
    }

    public void actionPerformed (ActionEvent e) {
180        adaptee.lowCostButton_actionPerformed (e);
    }
}

class EditPanel_highQualityButton_actionAdapter
185     implements java.awt.event.ActionListener {
    EditPanel adaptee ;

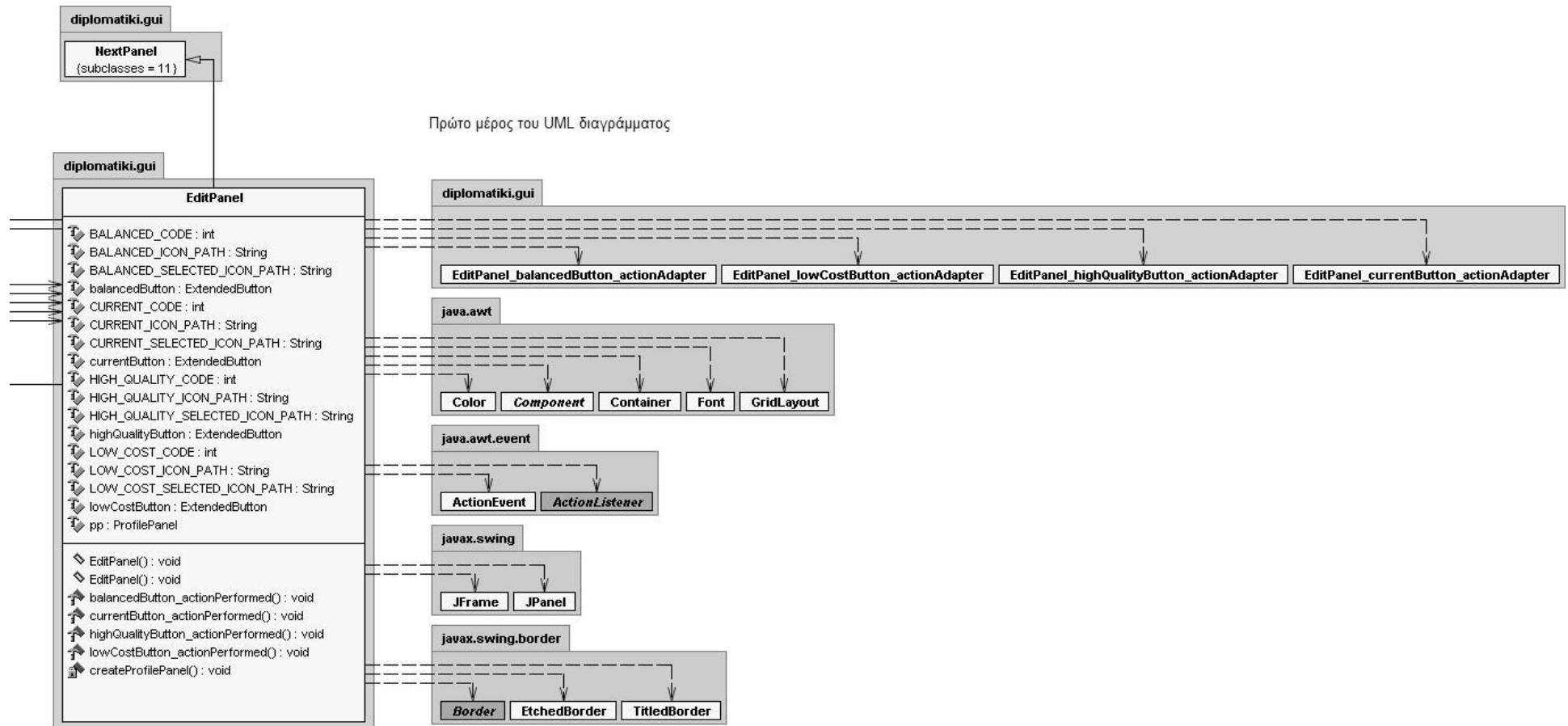
    EditPanel_highQualityButton_actionAdapter (EditPanel adaptee) {
190        this.adaptee = adaptee ;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.highQualityButton_actionPerformed (e);
195    }
}

class EditPanel_balancedButton_actionAdapter
    implements java.awt.event.ActionListener {
    EditPanel adaptee ;

200    EditPanel_balancedButton_actionAdapter (EditPanel adaptee) {
        this.adaptee = adaptee ;
    }

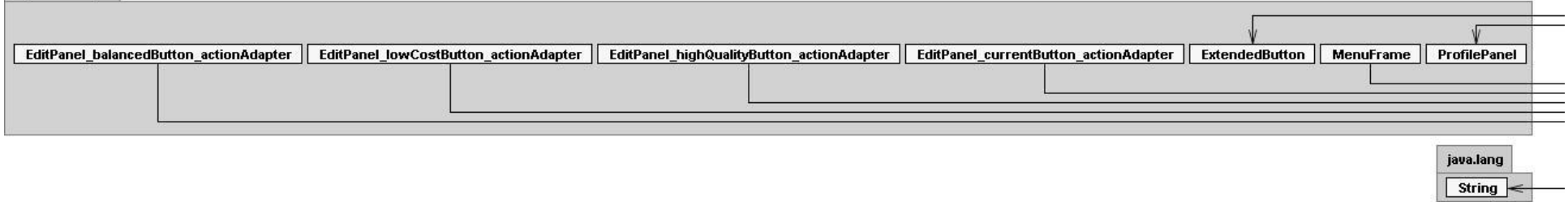
205    public void actionPerformed (ActionEvent e) {
        adaptee.balancedButton_actionPerformed (e);
    }
}
}
```



Πρώτο μέρος του UML διαγράμματος

diplomatiki.gui

Δεύτερο Μέρος του UML διαγράμματος



```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>EmailInputPanel</code> class</p>
10 *
 * <p>Description: This is the center panel of the Email Service panel. It holds
 * the various input fields, where the user should fill in the appropriate
 * information (e.g. the recipient's address).</p>
 *
15 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
 */
public class EmailInputPanel
    extends JPanel {
    final String LOAD_DEFAULT_ICON_PATH = Config.LOAD_DEFAULT_ICON_PATH ;
25    final String LOAD_SELECTED_ICON_PATH = Config.LOAD_SELECTED_ICON_PATH ;

    JLabel addressLabel = new JLabel ();
    JTextField addressField = new JTextField ();
    JLabel messageLabel = new JLabel ();
30    JScrollPane scroll = new JScrollPane ();
    JTextArea message = new JTextArea ();
    JTextField attachmentField = new JTextField ();
    JLabel attachmentLabel = new JLabel ();
    ExtendedButton load1;
35    JFileChooser jFileChooser1 = new JFileChooser ();

    /**
     * Creates a new <code>EmailInputPanel</code> instance.
     */
40    public EmailInputPanel () {
        try {
            jbInit ();
        }
        catch (Exception ex) {
45            ex.printStackTrace ();
        }
    }

    /**
50     * Builds the panel.
     *
     * @throws Exception An exception is thrown if the construction of the panel
     * is not successful
     */
55    void jbInit () throws Exception {
        this.setBackground (UIManager .getColor ("InternalFrame.activeTitleGradient" ));
        this.setMinimumSize (new Dimension (300, 140));
        this.setMaximumSize (new Dimension (300, 140));
        this.setPreferredSize (new Dimension (300, 140));
60        this.setLayout (null);

        addressLabel.setFont (new java .awt .Font ("Microsoft Sans Serif" , 1, 11));
        addressLabel.setText ("Address");
        addressLabel.setBounds (new Rectangle (40, 9, 51, 15));

65        addressField.setText ("");
        addressField.setBounds (new Rectangle (110, 7, 74, 21));

        messageLabel.setFont (new java .awt .Font ("Microsoft Sans Serif" , 1, 11));
70        messageLabel.setText ("Message");

```

```

        messageLabel.setBounds(new Rectangle(40, 34, 55, 15));

        scroll.getViewport().setBackground(UIManager.getColor(
            "InternalFrame.activeTitleGradient"));
75     scroll.setBounds(new Rectangle(110, 32, 168, 80));

        message.setText("");
        message.addKeyListener(new EmailInputPanel_message_keyAdapter (this));

80     scroll.getViewport().add(message, null);

        attachmentLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
        attachmentLabel.setText("Attachment");
        attachmentLabel.setBounds(new Rectangle(40, 118, 72, 15));

85     attachmentField.setText("");
        attachmentField.setBounds(new Rectangle(110, 116, 74, 21));

        load1 = new ExtendedButton(Load_DEFAULT_ICON_PATH, Load_SELECTED_ICON_PATH);
90     load1.setBounds(new Rectangle(189, 116, 65, 20));
        load1.addActionListener(new EmailInputPanel_load1_actionAdapter (this));

        this.add(addressLabel, null);
        this.add(addressField, null);
95     this.add(scroll, null);
        this.add(messageLabel, null);
        this.add(attachmentField, null);
        this.add(attachmentLabel, null);
        this.add(load1, null);
100 }

/*
 * message_keyPressed
 *
105 * @param e KeyEvent
 */
void message_keyPressed(KeyEvent e) {
    if (e.getKeyCode() == KeyEvent.VK_DOWN) {
        message.transferFocus();
110    }
    else if (e.getKeyCode() == KeyEvent.VK_UP) {
        message.transferFocusBackward();
    }
}

115 /**
 * Action performed when the "Load" button is pressed. The
 * <code>loadFile()</code> method is invoked.
 *
120 * @param e ActionEvent
 */
void load1_actionPerformed(ActionEvent e) {
    loadFile(jFileChooser1, attachmentField);
125 }

/**
 * A dialog window pops up, so the user may choose a file (as an attachment to
 * the e-mail).
 *
130 * @param jfc File chooser object
 * @param textField The text field in which the filepath will be written
 */
void loadFile(JFileChooser jfc, JTextField textField) {
    if (JFileChooser.APPROVE_OPTION == jfc.showOpenDialog(this)) {
135        String filename = jfc.getSelectedFile().getPath();
        textField.setText(filename);
    }
}

140 }

```

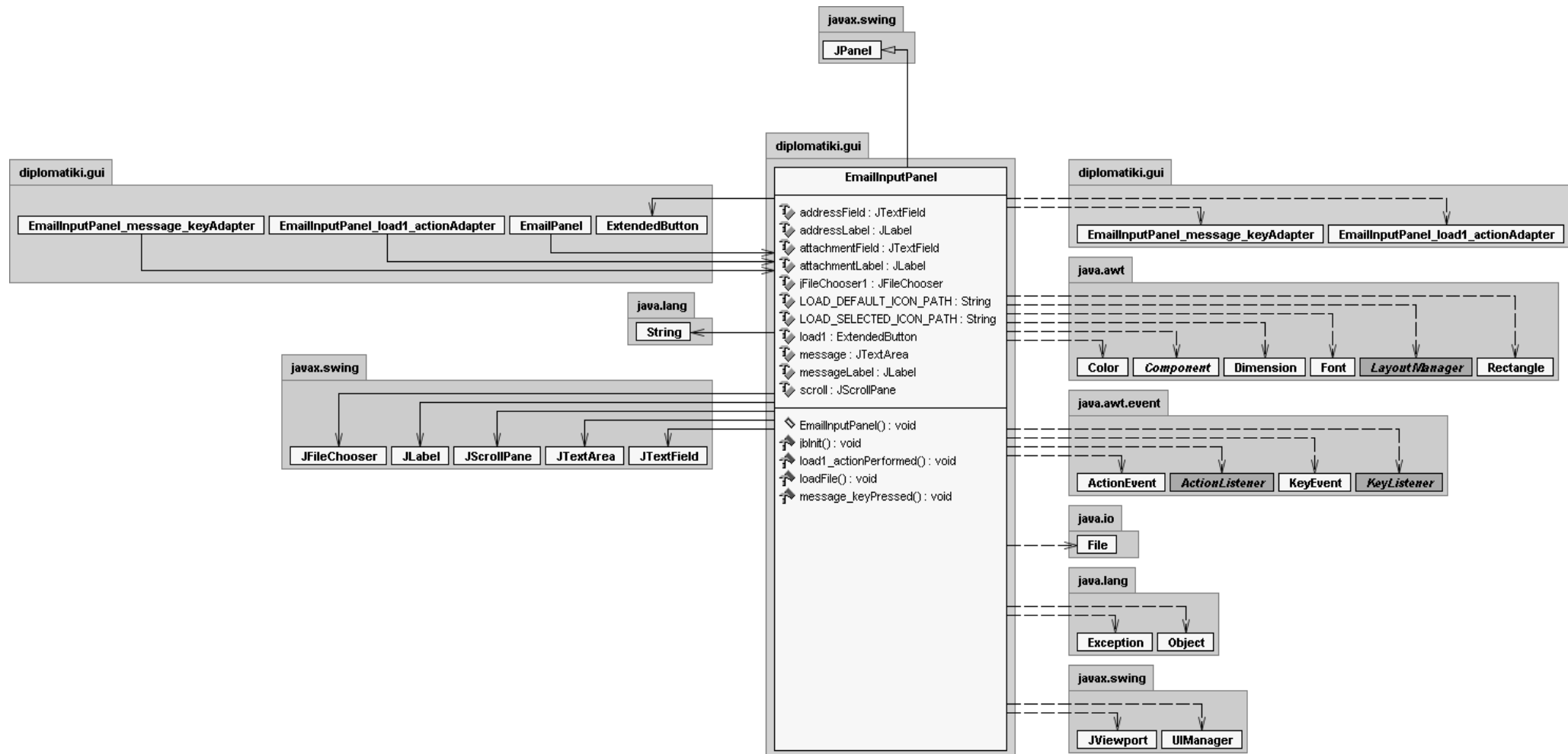
```
class EmailInputPanel_message_keyAdapter
    extends java.awt.event.KeyAdapter {
    EmailInputPanel adaptee;
145     EmailInputPanel_message_keyAdapter (EmailInputPanel adaptee) {
        this.adaptee = adaptee;
    }

150     public void keyPressed (KeyEvent e) {
        adaptee.message_keyPressed (e);
    }
}

155 class EmailInputPanel_load1_actionAdapter
    implements java.awt.event.ActionListener {
    EmailInputPanel adaptee;

    EmailInputPanel_load1_actionAdapter (EmailInputPanel adaptee) {
160         this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
165         adaptee.load1_actionPerformed (e);
    }
}
```





```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;
import diplomatiki .Config;
import diplomatiki .services .*;

/**
10 * <p>Title: Term Paper - <code>EmailPanel</code> class</p>
*
* <p>Description: This is the Email Service panel. This panel is shown when the
* user chooses to send an e-mail.</p>
*
15 * <p>Copyright: Copyright (c) 2005</p>
*
* <p>Company: Jadacode</p>
*
* @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
*/
public class EmailPanel
    extends JPanel {
    final int SERVICE_CODE = Config.EMAIL_SERVICE_CODE ;
25    final int MAX_TIME = Config.BAR_MAX_TIME ;

    final String START_DEFAULT_ICON_PATH = Config.EMAIL_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config.EMAIL_START_SELECTED_ICON_PATH ;

30    InfoPanel ip;
    ExtendedButton startButton;
    EmailInputPanel input;

    /**
35    * Creates a new <code>EmailPanel</code> instance.
    */
    public EmailPanel () {
    }

40    /**
    * Creates a new <code>EmailPanel</code> instance with the specified
    * parent frame and previous panel.
    *
    * @param myFrame The frame which this panel belongs to
45    * @param previousPanel The previous panel that was visible in the
    * <code>myFrame</code> frame. This reference is very useful if the user
    * wants to return to the previous panel
    */
    public EmailPanel (JFrame myFrame, JPanel previousPanel) {
50        super (myFrame, previousPanel);

        GridLayout1 .setColumns (1);
        GridLayout1 .setRows (1);

55        startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
            START_SELECTED_ICON_PATH );
        startButton .
            addActionListener (new EmailPanel_startButton_actionAdapter (this));

60        northPanel .setMinimumSize (new Dimension (400, 42));
        northPanel .setMinimumSize (new Dimension (400, 42));
        northPanel .setPreferredSize (new Dimension (400, 42));
        northPanel .add (startButton);

65        input = new EmailInputPanel ();

        centerPanel .add (input);
    }

70    /**

```

```

    * Action performed when the "Start" service button is pressed.
    */
private void startAction () {
    //Check if the address/number field is filled in
75     if (input.addressField.getText().equals("")) {
        new MessageWindow(myFrame, "Input Required:", "Please type an address" );
        return;
    }

80     //Run IIS and Construct ConnectionInfoPanel
    diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
        executeIIS (SERVICE_CODE, true, -1);

    //Check if myInfo is null
85     if (myInfo == null) {
        new MessageWindow (myFrame);
        return;
    }

90     //Construct ConnectionInfoPanel
    ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
        getAccessTechnology (),
        myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());

95     //Construct InfoPanel
    ip = new InfoPanel (myFrame, this, "Sending e-mail to: " ,
        input.addressField.getText ());

    //Add the ConnectionInfoPanel to the InfoPanel
100    ip.centerPanel.add(cip);

    //Create ProgressBar and Add it to the InfoPanel
    ProgressBar bar = new ProgressBar (MAX_TIME);
    //ip.add("East", bar);
105    ip.add(bar, BorderLayout.EAST);

    //Show the InfoPanel
    this.setVisible (false);
    myFrame.setContentPane (ip);
110    myFrame.setVisible (false);
    myFrame.setVisible (true);
    ip.setVisible (true);

    //Start the ProgressBar
115    bar.start ();

    //Start the actual service
    if (Config.EMAIL_ENABLED) {
        startActualService ();
120    }
}

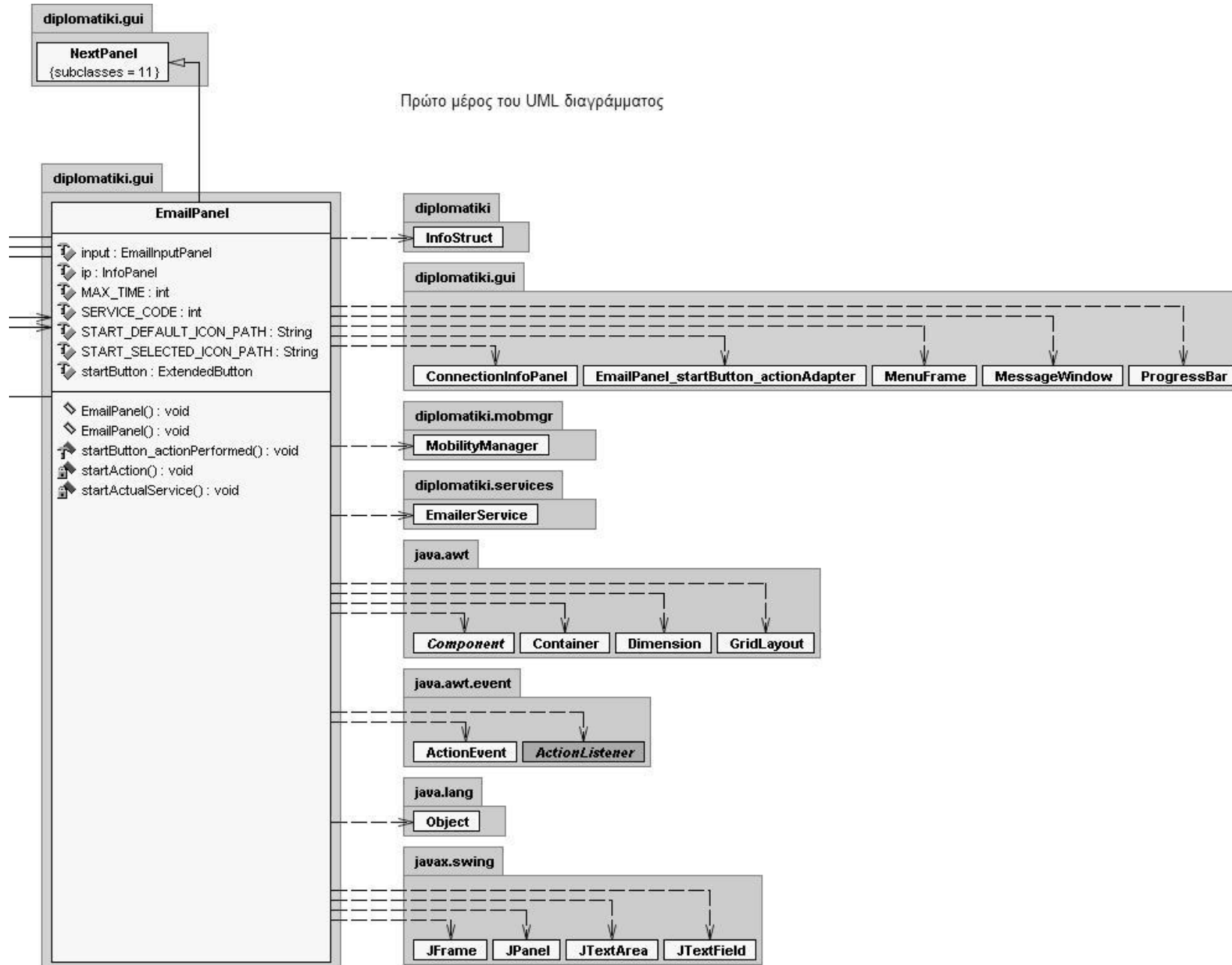
/**
 * The actual service is started. It runs in a separate thread.
 */
125 private void startActualService () {
    String recipient = input.addressField.getText ();
    String subject = "";
    String body = input.message.getText ();
130    EmailerService emailer = new EmailerService (recipient, subject, body);
}

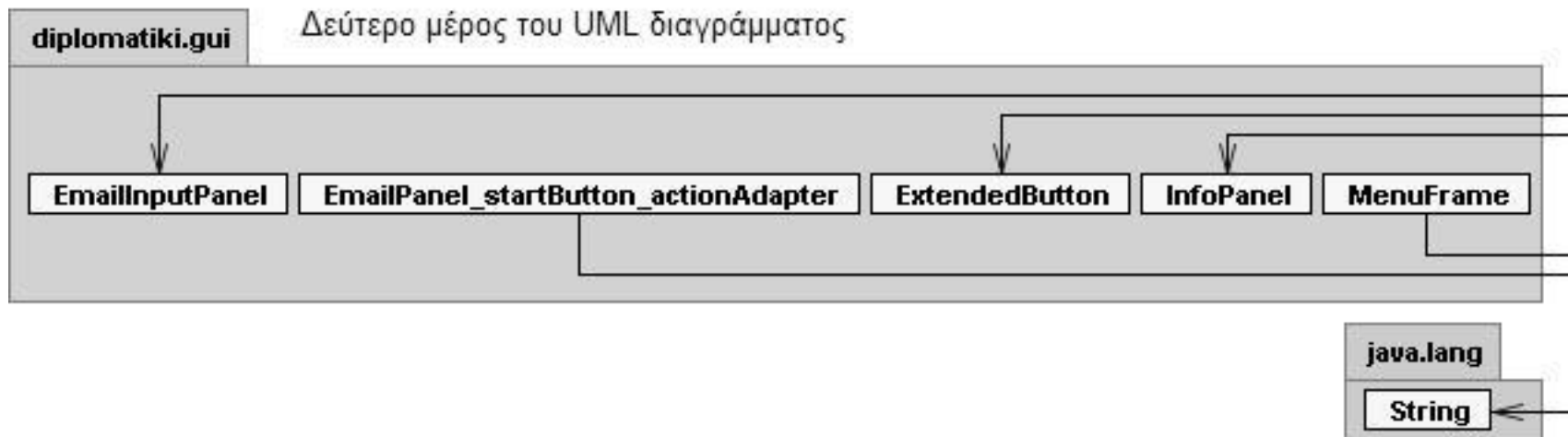
/**
 * Action performed when the "Start" service button is pressed. The
135 * <code>startAction()</code> method is invoked.
 *
 * @param e ActionEvent
 */
140 void startButton_actionPerformed (ActionEvent e) {
    startAction ();
}

```

```
    }  
    }  
145  class EmailPanel_startButton_actionAdapter  
    implements java.awt.event.ActionListener {  
        EmailPanel adaptee;  
  
        EmailPanel_startButton_actionAdapter (EmailPanel adaptee) {  
150    this.adaptee = adaptee;  
        }  
  
        public void actionPerformed (ActionEvent e) {  
155    adaptee.startButton_actionPerformed (e);  
        }  
    }
```

Πρώτο μέρος του UML διαγράμματος





```

package diplomatiki .gui;

import java .awt .event .*;
import javax .swing .*;
5
/**
 * <p>Title: Term Paper - <code>ExtendedButton</code> class</p>
 *
 * <p>Description: This class is a sub-class of <code>JButton</code>. Buttons of
10 * this class have specific characteristics (e.g. color, border, etc). They also
 * have a different icon when they are selected and when they are not.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
20 public class ExtendedButton
    extends JButton {
    ImageIcon defaultIcon;
    ImageIcon selectedIcon;

25 /**
 * Creates a new <code>ExtendedButton</code> instance.
 */
    public ExtendedButton () {
30         super ();
        this .setFocusable (true);
        this .setFocusPainted (false);
        this .setBackground (UIManager .getColor ("InternalFrame .activeTitleGradient" ));
        this .setBorder (null);
        this .setText ("");
35     }

    /**
 * Creates a new <code>ExtendedButton</code> instance with the specified
 * default icon and selected icon.
40 *
 * @param defaultIconPath Icon to be used for the default state
 * @param selectedIconPath Icon to be used for the selected state
 */
    public ExtendedButton (String defaultIconPath , String selectedIconPath ) {
45         super ();
        defaultIcon = new ImageIcon (defaultIconPath );
        selectedIcon = new ImageIcon (selectedIconPath );
        this .setIcon (defaultIcon);
        this .setFocusable (true);
50         this .setFocusPainted (false);
        this .setBackground (UIManager .getColor ("InternalFrame .activeTitleGradient" ));
        this .setBorder (null);
        this .setText ("");

55         try {
            jbInit ();
        }
        catch (Exception e) {
            e.printStackTrace ();
60     }
    }

    /**
 * Completes the construction of the <code>ExtendedButton</code> instance.
65 *
 * @throws Exception An exception is thrown if the construction of the button
 * could not be completed
 */
    private void jbInit () throws Exception {
70         this .addFocusListener (new ExtendedButton_this_focusAdapter (this));

```

```
    }

    /**
     * Action performed when the button gains focus.
75     *
     * @param e FocusEvent
     */
    void this_focusGained (FocusEvent e) {
80         this.setIcon (selectedIcon);
    }

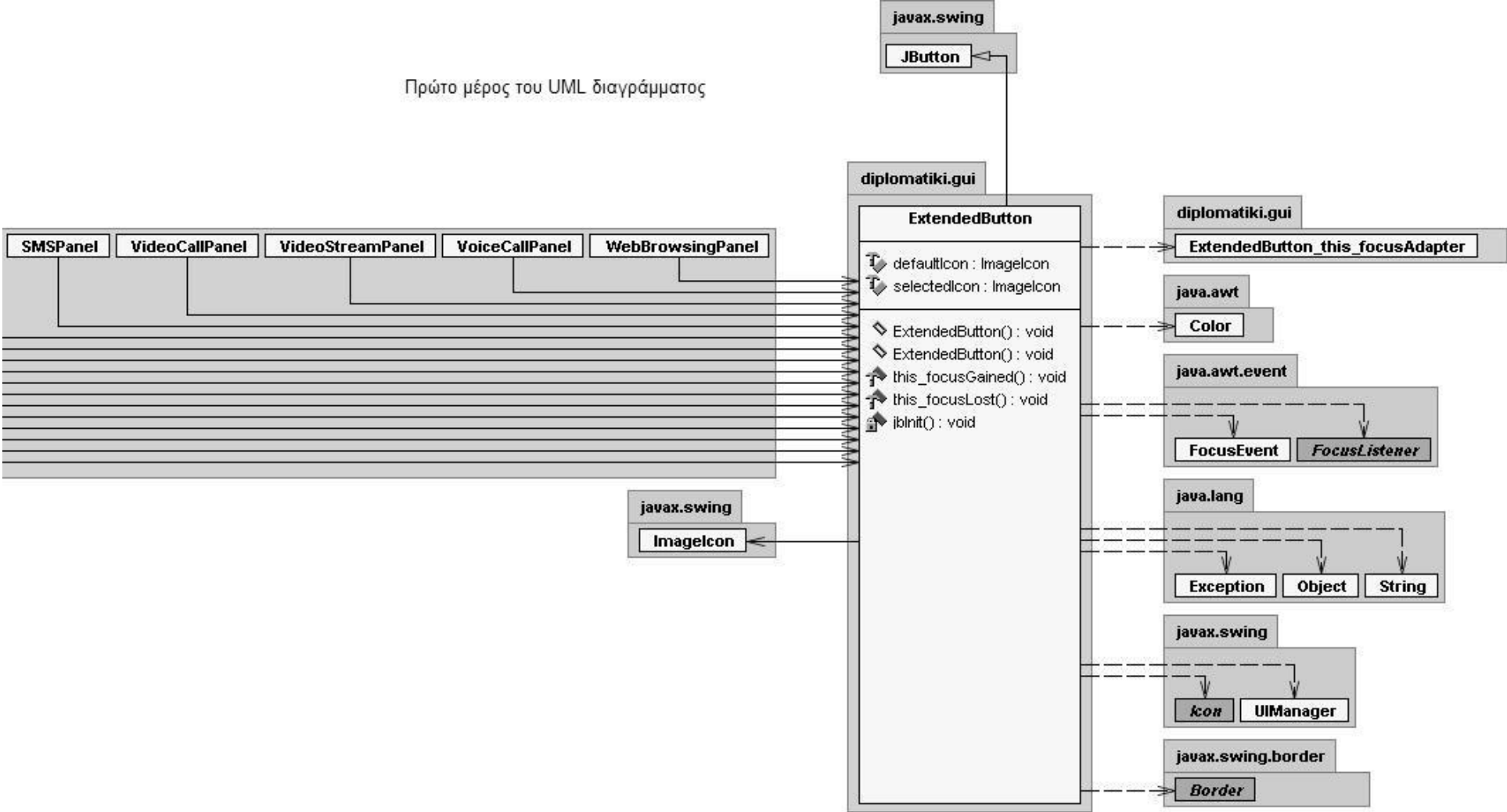
    /**
     * Action performed when the button loses focus.
     *
85     * @param e FocusEvent
     */
    void this_focusLost (FocusEvent e) {
90         this.setIcon (defaultIcon);
    }
}

class ExtendedButton_this_focusAdapter
    extends java.awt.event.FocusAdapter {
95     ExtendedButton adaptee;

    ExtendedButton_this_focusAdapter (ExtendedButton adaptee) {
        this.adaptee = adaptee;
    }
100     public void focusGained (FocusEvent e) {
        adaptee.this_focusGained (e);
    }

105     public void focusLost (FocusEvent e) {
        adaptee.this_focusLost (e);
    }
}
```

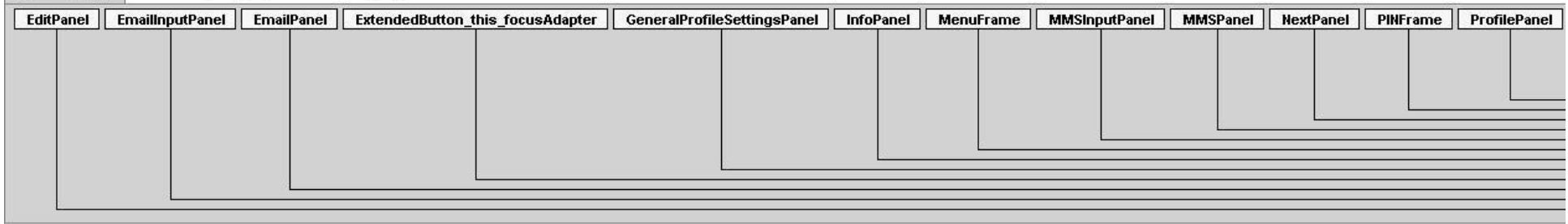
Πρώτο μέρος του UML διαγράμματος





diplomatiki.gui

Δεύτερο μέρος του UML διαγράμματος



```

package diplomatiki .gui;

import java .awt .Dimension ;
import javax .swing .JSlider ;
5 import javax .swing .UIManager ;

/**
 * <p>Title: Term Paper - <code>ExtendedSlider</code> class</p>
 *
10 * <p>Description: This class is a sub-class of <code>JSlider</code>. Sliders of
 * this class have specific characteristics (e.g. color, size, etc). They can
 * also have a specific maximum and default value.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class ExtendedSlider
    extends JSlider {
    int maxValue = 5;
    int defaultValue = 1;

25
    /**
     * Sets the slider's default value.
     *
     * @param defaultValue The default value
    */
30 public void setDefaultValue (int defaultValue) {
    this.defaultValue = defaultValue ;
    this.setValue (defaultValue);
}

35
    /**
     * Returns the slider's default value.
     *
     * @return The slider's default value
40 */
    public int getDefaultValue () {
        return defaultValue ;
    }

45
    /**
     * Creates a new <code>ExtendedSlider</code> instance.
     */
    public ExtendedSlider () {
        try {
50         jbInit ();
        }
        catch (Exception e) {
            e.printStackTrace ();
        }
55 }

    /**
     * Creates a new <code>ExtendedSlider</code> instance with the specified
     * maximum value and name.
     *
     * @param maxValue The slider's maximum value
     * @param sliderName The slider's name
     */
60
    public ExtendedSlider (int maxValue, String sliderName) {
        super ();
        this.maxValue = maxValue ;
        this.setName (sliderName);

65
        try {
70         jbInit ();
        }
    }

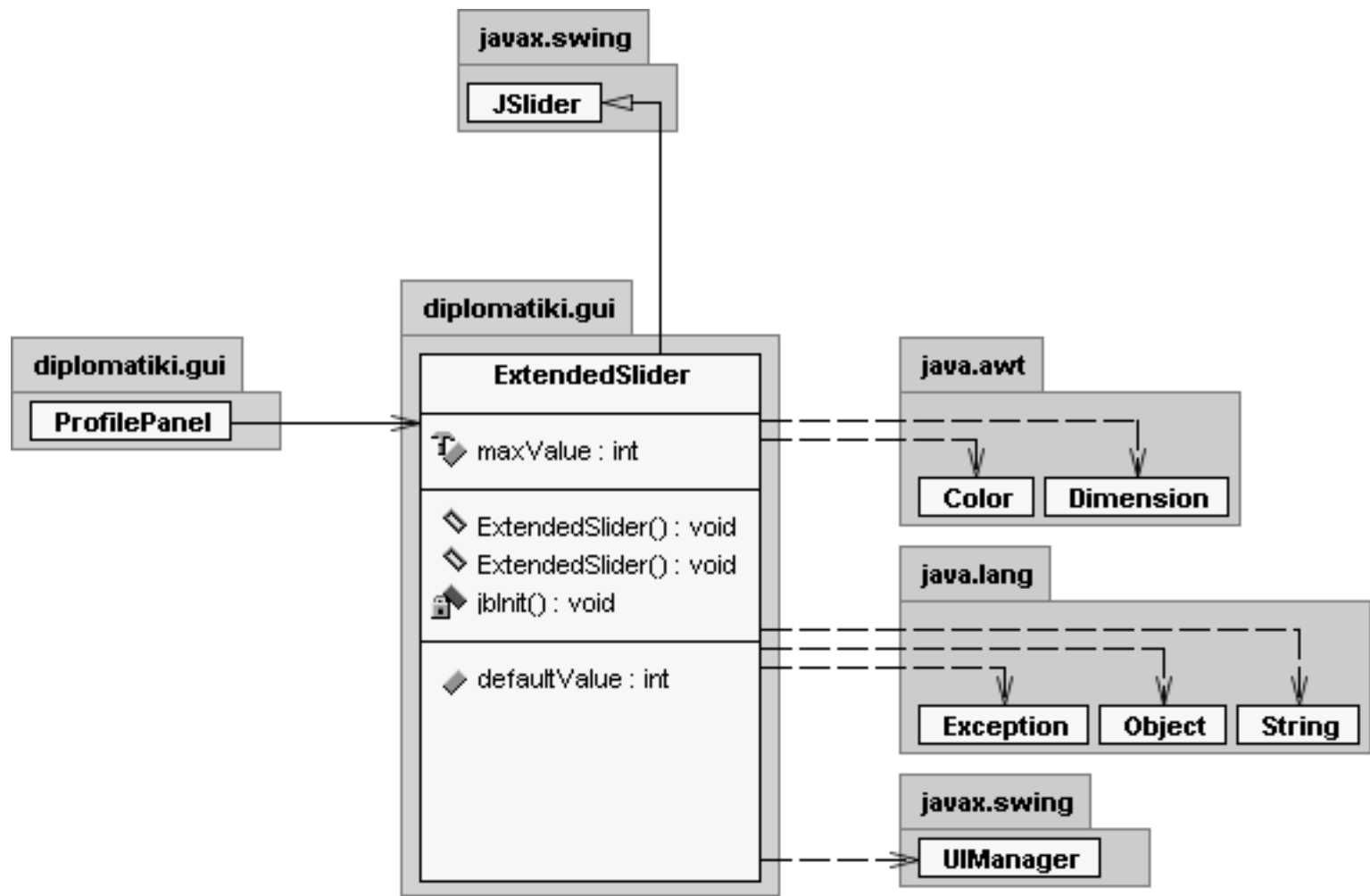
```

```
    }
    catch (Exception e) {
        e.printStackTrace ();
    }
75 }

/**
 * Builds the slider.
 *
80 * @throws Exception An exception is thrown if the construction of the slider
 * is not successful
 */
private void jbInit() throws Exception {
85     this.setBackground (UIManager.getColor ("InternalFrame.activeTitleGradient" ));
    this.setMinimumSize (new Dimension (100, 40));
    this.setMaximumSize (new Dimension (100, 40));
    this.setPreferredSize (new Dimension (100, 40));

90     this.setMinimum (1);
    this.setMaximum (maxValue);
    this.setValue (defaultValue);

    this.setMinorTickSpacing (0);
    this.setMajorTickSpacing (1);
95     this.setPaintTicks (true);
    this.setPaintLabels (true);
}
}
100 }
```



```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>GeneralProfileSettingsPanel</code> class</p>
10 *
 * <p>Description: This is part (the north panel) of the
 * <code>ProfilePanel</code>. It holds and presents the user's general
 * settings.</p>
 *
15 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
 */
public class GeneralProfileSettingsPanel
    extends JPanel {
    final String UP_DEFAULT_ICON_PATH = Config.UP_DEFAULT_ICON_PATH ;
25 final String UP_SELECTED_ICON_PATH = Config.UP_SELECTED_ICON_PATH ;
    final String DOWN_DEFAULT_ICON_PATH = Config.DOWN_DEFAULT_ICON_PATH ;
    final String DOWN_SELECTED_ICON_PATH = Config.DOWN_SELECTED_ICON_PATH ;

    DefaultListModel dlm = new DefaultListModel ();
30 JList factorList = new JList (dlm);
    ExtendedButton up = new ExtendedButton (UP_DEFAULT_ICON_PATH ,
        UP_SELECTED_ICON_PATH );
    ExtendedButton down = new ExtendedButton (DOWN_DEFAULT_ICON_PATH ,
        DOWN_SELECTED_ICON_PATH );
35 String factor1 = "", factor2 = "", factor3 = "";
    String [] providers ;
    int selectedIndex = -1, otherProviderCode = 0;

    JLabel providerLabel = new JLabel ();
40 JComboBox providerBox = new JComboBox ();
    JLabel specifyLabel = new JLabel ();
    JTextField otherProviderCodeField = new JTextField ();

    /**
45 * Creates a new <code>GeneralProfileSettingsPanel</code> instance.
    */
    public GeneralProfileSettingsPanel () {
    }

50 /**
 * Creates a new <code>GeneralProfileSettingsPanel</code> instance. The first
 * three parameters represent the ordered list of factors according to the
 * user preferences. They can take the following values: "Quality", "Cost" or
 * "Provider". The fourth argument is a list of providers. The fifth argument
55 * represents the position of the selected (preferred) provider in the
 * aforementioned list. The final argument is the code of the preferred
 * provider, in case it is not one of the predefined providers (and as a
 * result it is not in the aforementioned list). These pieces of information
 * are presented in the GUI, and are also editable.
60 *
 * @param factor1 The first (most important) factor according the user's
 * current preferences
 * @param factor2 The second factor according the user's current preferences
 * @param factor3 The third (least important) factor according the user's
65 * current preferences
 * @param providers A list of predefined provider names
 * @param selectedIndex The position (in the previous list) of the preferred
 * provider
 * @param otherProviderCode The preferred provider's code (in case it is not
70 * included in the <code>providers</code> list)

```

```

    */
    public GeneralProfileSettingsPanel (String factor1, String factor2,
                                       String factor3, String[] providers,
                                       int selectedIndex, int otherProviderCode) {
75     super ();
        this.factor1 = factor1;
        this.factor2 = factor2;
        this.factor3 = factor3;
        this.providers = providers;
80     this.selectedIndex = selectedIndex;
        this.otherProviderCode = otherProviderCode ;

        try {
            jbInit ();
85     }
        catch (Exception ex) {
            ex.printStackTrace ();
        }
    }
90
    /**
     * Builds the panel.
     *
     * @throws Exception An exception is thrown if the construction of the panel
     * is not successful
    */
    void jbInit () throws Exception {
        this.setBackground (UIManager.getColor ("InternalFrame.activeTitleGradient" ));
        this.setMinimumSize (new Dimension (300, 90));
100     this.setPreferredSize (new Dimension (300, 90));
        this.setLayout (null);

        factorList.setBackground (UIManager.getColor (
            "InternalFrame.activeTitleGradient" ));
105     factorList.setBorder (BorderFactory.createEtchedBorder ());
        factorList.setSelectionMode (ListSelectionModel.SINGLE_SELECTION);
        factorList.setVisibleRowCount (3);
        factorList.setBounds (new Rectangle (12, 10, 75, 55));

110     dlm.addElement (factor1);
        dlm.addElement (factor2);
        dlm.addElement (factor3);

        up.setBounds (new Rectangle (106, 13, 73, 20));
115     up.addActionListener (
        new GeneralProfileSettingsPanel_up_actionAdapter (this));
        down.setBounds (new Rectangle (106, 43, 73, 20));
        down.addActionListener (
        new GeneralProfileSettingsPanel_down_actionAdapter (this));
120

        providerBox.setBackground (UIManager.getColor (
            "InternalFrame.activeTitleGradient" ));
        providerBox.setBounds (new Rectangle (277, 12, 73, 21));
        providerBox.addItemListener (new
125     GeneralProfileSettingsPanel_providerBox_itemAdapter (this)
);

        for (int i = 0; i < providers.length; i++) {
            providerBox.addItem (providers [i]);
130     }
        providerBox.setSelectedIndex (selectedIndex);

        providerLabel.setText ("Choose provider" );
        providerLabel.setBounds (new Rectangle (190, 15, 80, 15));

135     specifyLabel.setText ("Please specify" );
        specifyLabel.setBounds (new Rectangle (190, 45, 72, 15));

        otherProviderCodeField.setText ("");
        otherProviderCodeField.setBounds (new Rectangle (277, 41, 73, 21));

```

```

140     if (otherProviderCode == 0) {
        hideExtraFields ();
    }
    else {
145     showExtraFields ();
        otherProviderCodeField .setText (" " + otherProviderCode );
    }

    this.add(factorList , null);
150    this.add(up, null);
    this.add(down, null);
    this.add(providerLabel , null);
    this.add(providerBox , null);
    this.add(specifyLabel , null);
155    this.add(otherProviderCodeField , null);
    }

    /**
    * Hides some extra fields in the panel. This happens in case the preferred
160    * provider is included in the list of predefined providers.
    */
    void hideExtraFields () {
        specifyLabel .setVisible (false);
        otherProviderCodeField .setEnabled (false);
165    otherProviderCodeField .setVisible (false);
    }

    /**
    * Shows some extra fields in the panel. This happens in case the preferred
170    * provider is not included in the list of predefined providers.
    */
    void showExtraFields () {
        specifyLabel .setVisible (true);
        otherProviderCodeField .setEnabled (true);
175    otherProviderCodeField .setVisible (true);
    }

    /**
    * Action performed when the state of the drop-down box with the provider
180    * names is changed. It invokes either the <code>showExtraFields()</code> or
    * the <code>hideExtraFields()</code> method.
    *
    * @param e ItemEvent
    */
185    void providerBox_itemStateChanged (ItemEvent e) {
        if (providerBox .getSelectedIndex () == providers .length - 1) {
            showExtraFields ();
        }
        else {
190            hideExtraFields ();
        }
    }

    /**
195    * Action performed if the "Up" button is pressed.
    *
    * @param e ActionEvent
    */
    void up_actionPerformed (ActionEvent e) {
200    int selectedFactorIndex = factorList .getSelectedIndex ();
        if (selectedFactorIndex == -1) {
            return;
        }

205    String selectedFactorName = (String) dlm.getElementAt (selectedFactorIndex );
        if ( (selectedFactorName != null) && (selectedFactorIndex != 0)) {
            dlm.setElementAt (dlm.getElementAt ( (selectedFactorIndex - 1)),
                selectedFactorIndex );
            dlm.setElementAt (selectedFactorName , selectedFactorIndex - 1);
        }
    }

```

```

210     factorList.setSelectedIndex (selectedFactorIndex - 1);
        }
    }

    /**
215     * Action performed if the "Down" button is pressed.
     *
     * @param e ActionEvent
     */
    void down_actionPerformed (ActionEvent e) {
220     int selectedFactorIndex = factorList.getSelectedIndex ();
        if (selectedFactorIndex == -1) {
            return;
        }

225     String selectedFactorName = (String) dlm.getElementAt (selectedFactorIndex );
        if ( (selectedFactorName != null) &&
            (selectedFactorIndex != dlm.getSize () - 1)) {
            dlm.setElementAt (dlm.getElementAt ( (selectedFactorIndex + 1)),
230             selectedFactorIndex );
            dlm.setElementAt (selectedFactorName , selectedFactorIndex + 1);
            factorList.setSelectedIndex (selectedFactorIndex + 1);
        }
    }

235 }

    class GeneralProfileSettingsPanel_providerBox_itemAdapter
        implements java.awt.event.ItemListener {
        GeneralProfileSettingsPanel adaptee;

240     GeneralProfileSettingsPanel_providerBox_itemAdapter (
            GeneralProfileSettingsPanel adaptee) {
            this.adaptee = adaptee;
        }

245     public void itemStateChanged (ItemEvent e) {
            adaptee.providerBox_itemStateChanged (e);
        }
    }

250     class GeneralProfileSettingsPanel_up_actionAdapter
        implements java.awt.event.ActionListener {
        GeneralProfileSettingsPanel adaptee;

255     GeneralProfileSettingsPanel_up_actionAdapter (GeneralProfileSettingsPanel
            adaptee) {
            this.adaptee = adaptee;
        }

260     public void actionPerformed (ActionEvent e) {
            adaptee.up_actionPerformed (e);
        }
    }

265     class GeneralProfileSettingsPanel_down_actionAdapter
        implements java.awt.event.ActionListener {
        GeneralProfileSettingsPanel adaptee;

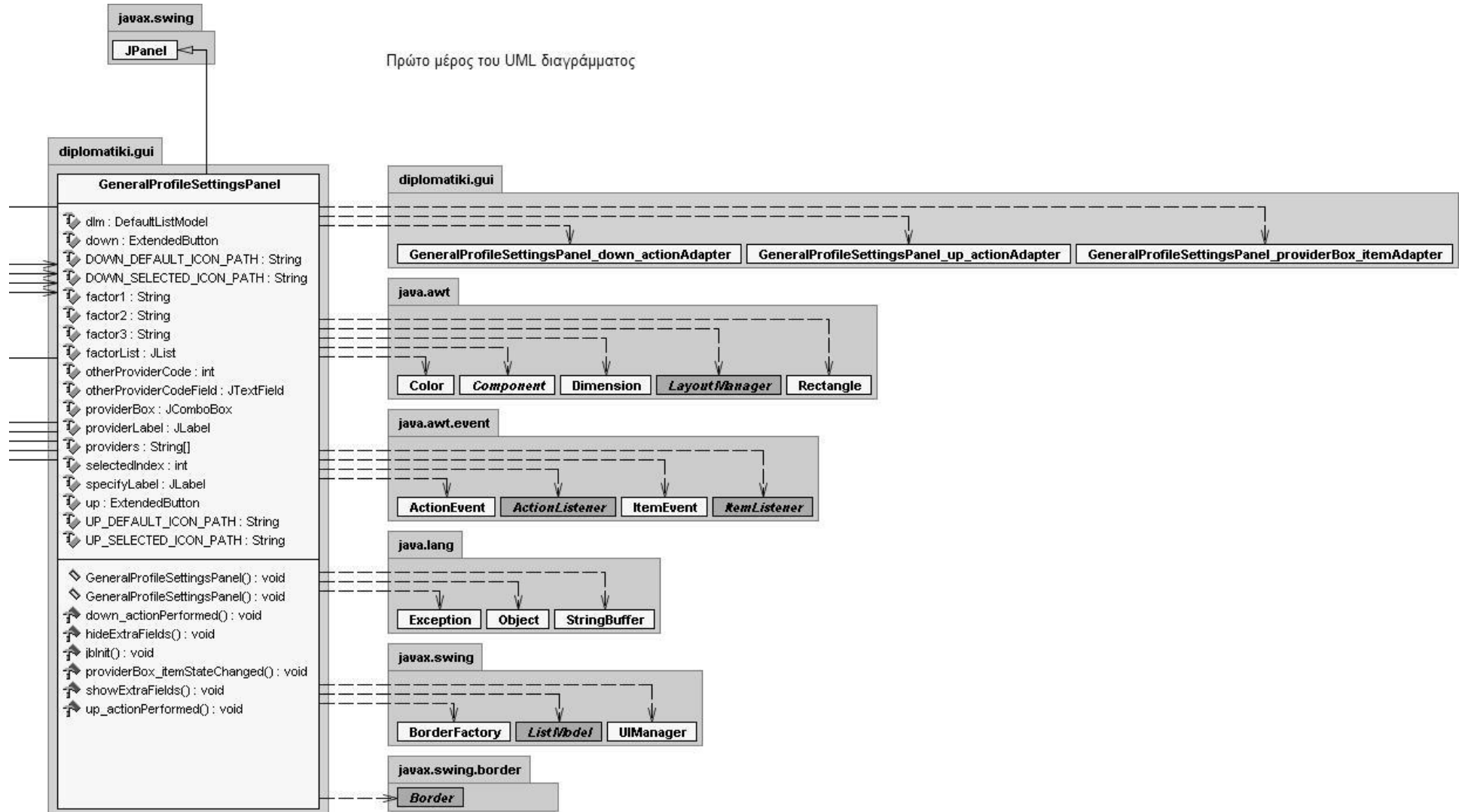
270     GeneralProfileSettingsPanel_down_actionAdapter (GeneralProfileSettingsPanel
            adaptee) {
            this.adaptee = adaptee;
        }

275     public void actionPerformed (ActionEvent e) {
            adaptee.down_actionPerformed (e);
        }
    }

```



Πρώτο μέρος του UML διαγράμματος



diplomatiqi.gui

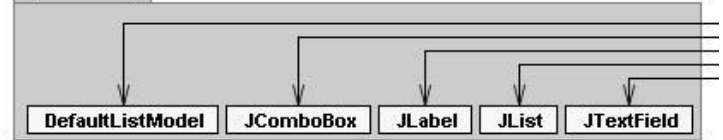
Δεύτερο μέρος του UML διαγράμματος



java.lang

String

javax.swing



```

package diplomatiki .gui;

import java .awt .*;
import javax .swing .*;
5 import javax .swing .border .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>InfoPanel</code> class</p>
10 *
 * <p>Description: This panel presents the details of an open connection. A
 * <code>ConnectionInfoPanel</code> is used as its center panel. It also
 * presents information about the running service.</p>
 *
15 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
 */
public class InfoPanel
    extends JPanel {
    final String DEFAULT_ANIMATED_ICON_PATH = Config.DEFAULT_ANIMATED_ICON_PATH ;
25
    JLabel messageLabel ;
    JTextField textInfo ;
    ImageIcon animatedIcon ;
    ExtendedButton animatedButton ;
30
    /**
     * Creates a new <code>InfoPanel</code> instance.
     */
    public InfoPanel () {
35
    }

    /**
     * Creates a new <code>InfoPanel</code> instance with the specified parent
     * frame and previous panel. The last two arguments are pieces of information
40 * that should be printed in the panel.
     *
     * @param myFrame The frame which this panel belongs to
     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
45 * wants to return to the previous panel
     * @param messageLabelInput A message that should be printed in the panel.
     * Usually, it's service-specific
     * @param textInfoInput Textual information that should also be presented
     */
50 public InfoPanel (JFrame myFrame , JPanel previousPanel ,
                    String messageLabelInput , String textInfoInput ) {
    super (myFrame , previousPanel );

    northPanel .setLayout (new FlowLayout ());
55 northPanel .setPreferredSize (new Dimension (400 , 60));

    northPanel .setBorder (
        new TitledBorder (new EtchedBorder (), "Service Info" ,
            TitledBorder .DEFAULT_JUSTIFICATION ,
60 TitledBorder .DEFAULT_POSITION ,
            new java .awt .Font ("Microsoft Sans Serif" , 0 , 11),
            new Color (10 , 92 , 174));

    centerPanel .setBorder (
        new TitledBorder (new EtchedBorder (), "Connection Details" ,
65 TitledBorder .DEFAULT_JUSTIFICATION ,
            TitledBorder .DEFAULT_POSITION ,
            new java .awt .Font ("Microsoft Sans Serif" , 0 , 11),
            new Color (10 , 92 , 174));

70 animatedIcon = new ImageIcon (DEFAULT_ANIMATED_ICON_PATH );

```

```
        animatedButton = new ExtendedButton ();
        animatedButton.setEnabled (false);
        animatedButton.setFocusable (false);
        animatedButton.setIcon (animatedIcon);
75    animatedButton.setDisabledIcon (animatedIcon);
        animatedIcon.setImageObserver (animatedButton);
        northPanel.add (animatedButton);

        messageLabel = new JLabel (messageLabelInput );
80    messageLabel.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));

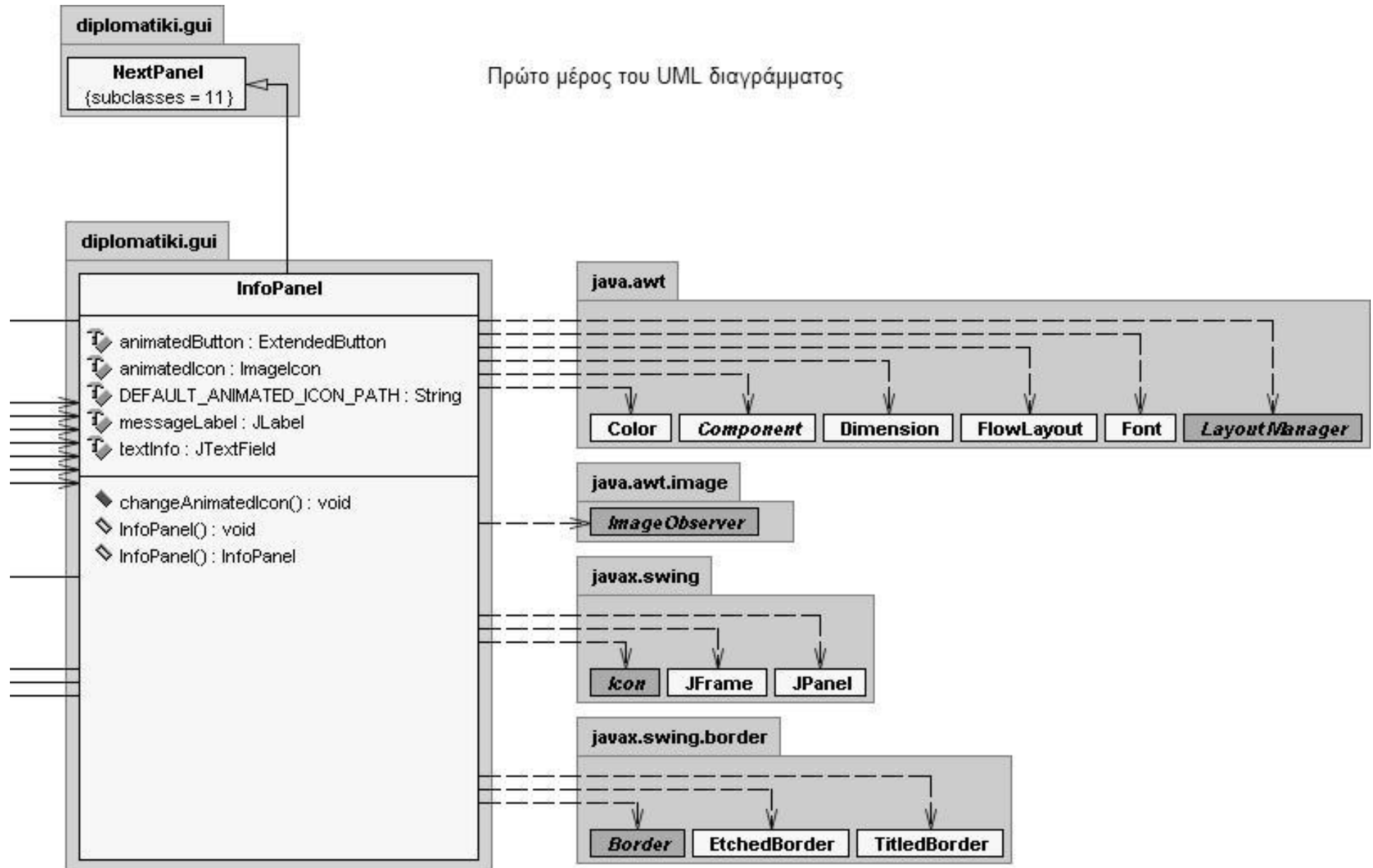
        textInfo = new JTextField (textInfoInput );
        textInfo.setBackground (new Color (166, 202, 240));
        textInfo.setFont (new Font ("Microsoft Sans Serif" , 1, 12));
85    textInfo.setForeground (Color.RED);
        textInfo.setBorder (null);
        textInfo.setEditable (false);
        textInfo.setFocusable (false);

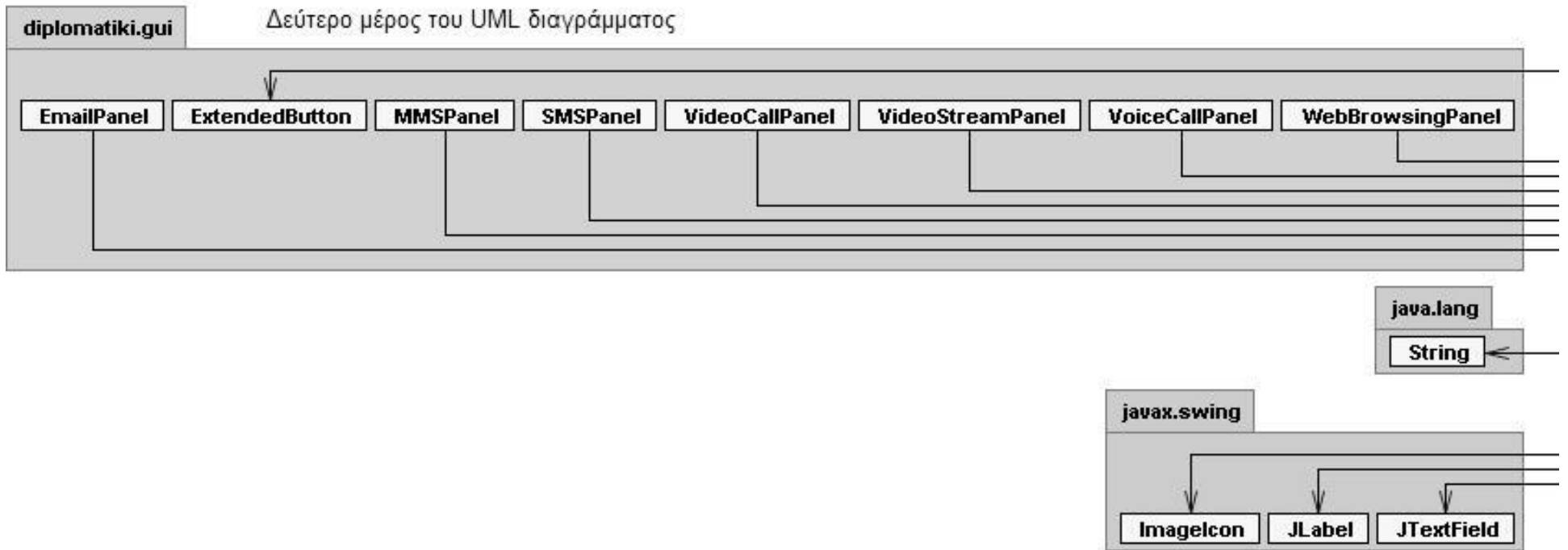
90    northPanel.add (messageLabel );
        northPanel.add (textInfo);
    }

    /**
95    * Changes the default animated icon in the panel with the one specified.
    *
    * @param newIconPath The path of the new animated icon
    */
    public void changeAnimatedIcon (String newIconPath) {
100    animatedIcon.setImageObserver (null);

        animatedIcon = new ImageIcon (newIconPath);
        animatedButton.setIcon (animatedIcon);
        animatedButton.setDisabledIcon (animatedIcon);
105    animatedIcon.setImageObserver (animatedButton);
    }
}
```

Πρώτο μέρος του UML διαγράμματος





```

package diplomatiki .gui;

import java .awt .*;
import javax .swing .*;
5
/**
 * <p>Title: Term Paper - <code>InterfaceAdaptationLogFrame</code> class</p>
 *
 * <p>Description: This is a frame used as a log output area by the Interface
10 * Adaptation (IA) module.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
15 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
public class InterfaceAdaptationLogFrame
20     extends JFrame {
    FlowLayout flowLayout1 = new FlowLayout ();
    JScrollPane scroller = new JScrollPane ();
    JTextArea output = new JTextArea ();

25     /**
     * Creates a new <code>InterfaceAdaptationLogFrame</code> instance.
     */
    public InterfaceAdaptationLogFrame () {
30         try {
            jbInit ();
        }
        catch (Exception ex) {
            ex.printStackTrace ();
        }
35     }

    /**
     * Builds the frame.
     *
     * @throws Exception An exception is thrown if the construction of the frame
40     * is not successful
     */
    void jbInit () throws Exception {
45         this .getContentPane ().setBackground (UIManager .getColor (
            "InternalFrame.activeTitleGradient" ));
        this .setDefaultCloseOperation (HIDE_ON_CLOSE );
        this .setTitle ("IA Module Log Window" );
        this .getContentPane ().setLayout (flowLayout1 );

50         output .setBackground (UIManager .getColor (
            "InternalFrame.activeTitleGradient" ));
        output .setEditable (false );
        output .setText ("");
        output .setColumns (40);
55         output .setRows (8);
        scroller .getViewport ().add (output, null );

        this .getContentPane ().add (scroller, null );

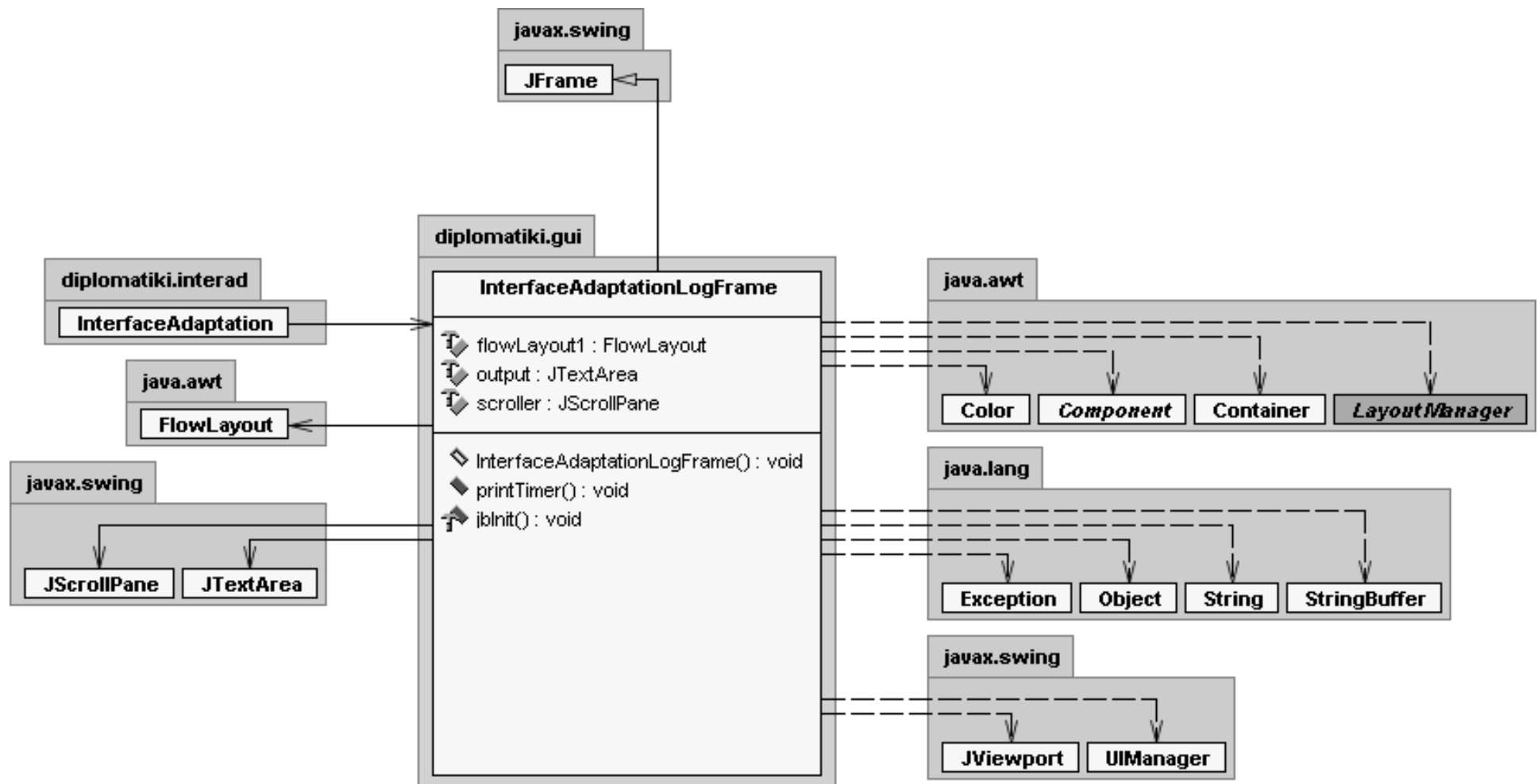
60         this .pack ();
        this .setVisible (true );
    }

    /**
65     * Prints the value of the <code>timer</code> parameter in the log area.
     *
     * @param timer The value that should be printed in the log area
     */
    public void printTimer (int timer) {
70         output .append ("New measurements retrieved, timer = " + timer + ".\n");

```

```
}  
}
```





```

package diplomatiki.gui;

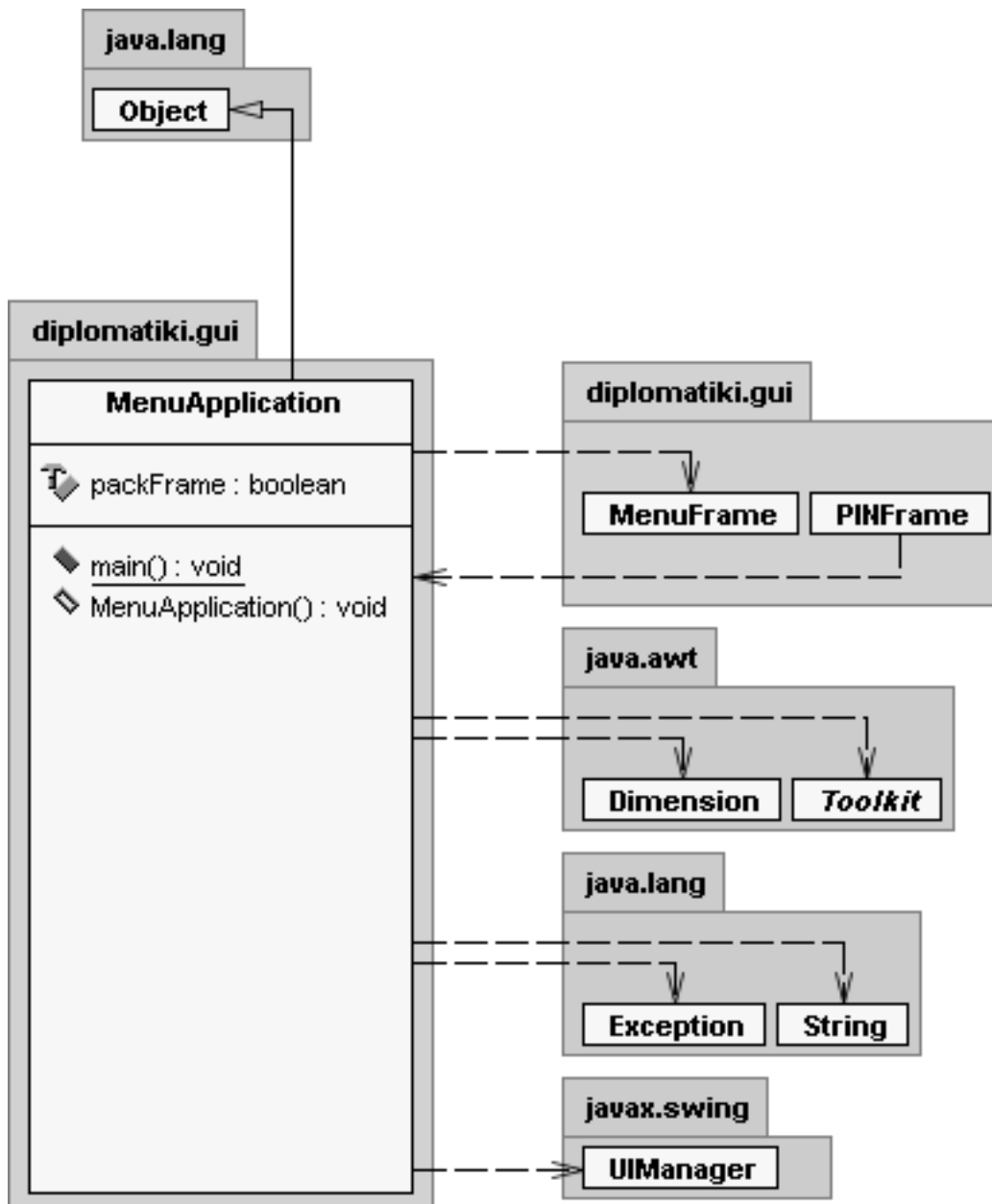
import java.awt.*;
import javax.swing.*;

5
/**
 * <p>Title: Term Paper - <code>MenuApplication</code> class</p>
 *
 * <p>Description: An instance of this class constructs and centers on screen a
10 * <code>MenuFrame</code> instance.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
15 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
public class MenuApplication {
20     boolean packFrame = false;

    //Construct the application
    /**
     * Apart from creating a new <code>MenuApplication</code> instance, this
25 * constructor creates a new <code>MenuFrame</code> instance and centers it on
     * screen.
     */
    public MenuApplication () {
30         MenuFrame frame = new MenuFrame ();
        //Validate frames that have preset sizes
        //Pack frames that have useful preferred size info, e.g. from their layout
        if (packFrame) {
            frame.pack();
        }
35         else {
            frame.validate ();
        }
        //Center the window
        Dimension screenSize = Toolkit.getDefaultToolkit ().getScreenSize ();
40         Dimension frameSize = frame.getSize ();
        if (frameSize.height > screenSize.height) {
            frameSize.height = screenSize.height;
        }
45         if (frameSize.width > screenSize.width) {
            frameSize.width = screenSize.width;
        }
        frame.setLocation ( (screenSize.width - frameSize.width) / 2,
                             (screenSize.height - frameSize.height) / 2);
        frame.setVisible (true);
50     }

    //Main method
    /**
     * This method is useful in case one wants to construct and launch directly
55 * the <code>MenuApplication</code>.
     *
     * @param args No command line arguments are necessary
     */
    public static void main(String[] args) {
60         try {
            UIManager.setLookAndFeel (UIManager.getSystemLookAndFeelClassName ());
        }
        catch (Exception e) {
            e.printStackTrace ();
65         }
        new MenuApplication ();
    }
}
70

```



```

package diplomatiki .gui;

import java.util.*;

5 import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

import diplomatiki .*;
10 import diplomatiki .mobmgr.*;

/**
 * <p>Title: Term Paper - <code>MenuFrame</code> class</p>
 *
15 * <p>Description: This is the GUI's main frame. All panels that are shown or
 * hidden belong to this frame. When first launched, the content panel of this
 * frame holds the application's main menu. This menu consists of nine (9)
 * buttons.</p>
 *
20 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
25 * @version 1.0
 */
public class MenuFrame
    extends JFrame {
    JPanel contentPane;
30 GridLayout gridLayout1 = new GridLayout ();
    ExtendedButton jButton1 =
        new ExtendedButton (Config.JBUTTON1_DEFAULT_ICON_PATH ,
            Config.JBUTTON1_SELECTED_ICON_PATH );
    ExtendedButton jButton2 =
35     new ExtendedButton (Config.JBUTTON2_DEFAULT_ICON_PATH ,
        Config.JBUTTON2_SELECTED_ICON_PATH );
    ExtendedButton jButton3 =
        new ExtendedButton (Config.JBUTTON3_DEFAULT_ICON_PATH ,
            Config.JBUTTON3_SELECTED_ICON_PATH );
40     ExtendedButton jButton4 =
        new ExtendedButton (Config.JBUTTON4_DEFAULT_ICON_PATH ,
            Config.JBUTTON4_SELECTED_ICON_PATH );
    ExtendedButton jButton5 =
45     new ExtendedButton (Config.JBUTTON5_DEFAULT_ICON_PATH ,
        Config.JBUTTON5_SELECTED_ICON_PATH );
    ExtendedButton jButton6 =
        new ExtendedButton (Config.JBUTTON6_DEFAULT_ICON_PATH ,
            Config.JBUTTON6_SELECTED_ICON_PATH );
    ExtendedButton jButton7 =
50     new ExtendedButton (Config.JBUTTON7_DEFAULT_ICON_PATH ,
        Config.JBUTTON7_SELECTED_ICON_PATH );
    ExtendedButton jButton8 =
        new ExtendedButton (Config.JBUTTON8_DEFAULT_ICON_PATH ,
            Config.JBUTTON8_SELECTED_ICON_PATH );
55     ExtendedButton jButton9 =
        new ExtendedButton (Config.JBUTTON9_DEFAULT_ICON_PATH ,
            Config.JBUTTON9_SELECTED_ICON_PATH );
    JButton [] jButton = new JButton [10];
    CalendarTitle showTimeInTitleBar ;
60     VoiceCallPanel vcp;
    SMSPanel smp;
    MMSPanel mmp;
    EmailPanel emailp;
    EditPanel ep;
65     VideoCallPanel vdc;
    VideoStreamPanel vsp;
    WebBrowsingPanel wbp;
    ApplicationsPanel ap;

70     MobilityManager mob = new MobilityManager (this);

```

```
/**
 * Returns a reference to the Mobility Manager module of the TMS.
 *
75  * @return A reference to the Mobility Manager module (that is a reference to
 *      a <code>MobilityManager</code> object)
 */
public MobilityManager getMobilityManager () {
80  return mob;
}

/**
 * Returns a reference to the <code>VoiceCallPanel</code>.
 *
85  * @return A reference to a <code>VoiceCallPanel</code> object
 */
public VoiceCallPanel getVoiceCallPanel () {
  return vcp;
90 }

/**
 * Returns a reference to the <code>VideoCallPanel</code>.
 *
 * @return A reference to a <code>VideoCallPanel</code> object
95 */
public VideoCallPanel getVideoCallPanel () {
  return vdc;
}

100 /**
 * Returns a reference to the <code>VideoStreamPanel</code>.
 *
 * @return A reference to a <code>VideoStreamPanel</code> object
 */
105 public VideoStreamPanel getVideoStreamPanel () {
  return vsp;
}

/**
110 * Returns a reference to the <code>WebBrowsingPanel</code>.
 *
 * @return A reference to a <code>WebBrowsingPanel</code> object
 */
115 public WebBrowsingPanel getWebBrowsingPanel () {
  return wbp;
}

/**
120 * Refreshes (updates) the connectivity information shown by the GUI. The
 * <code>serviceCode</code> defines which service's connection info should be
 * updated. The <code>myInfo</code> object holds the new information.
 *
 * @param serviceCode The code of the service for which the info should be
 *      updated
125 * @param myInfo The new info
 */
public void refreshAppropriateConnectionInfo (int serviceCode ,
                                             InfoStruct myInfo) {
  switch (serviceCode) {
130  case Config.VOICECALL_SERVICE_CODE :
    getVoiceCallPanel ().refreshConnectionInfo (myInfo);
    break;
    case Config.VIDEOCALL_SERVICE_CODE :
    getVideoCallPanel ().refreshConnectionInfo (myInfo);
135  break;
    case Config.VIDEOSTREAM_SERVICE_CODE :
    getVideoStreamPanel ().refreshConnectionInfo (myInfo);
    break;
    case Config.WEBBROWSING_SERVICE_CODE :
140  getWebBrowsingPanel ().refreshConnectionInfo (myInfo);
  }
}
```

```
        break;
    } //end of switch
}

145 //Construct the frame
/**
 * Creates a new <code>MenuFrame</code> instance.
 */
150 public MenuFrame () {
    enableEvents (AWTEvent.WINDOW_EVENT_MASK);
    try {
        jbInit ();
    }
    catch (Exception e) {
155     e.printStackTrace ();
    }
}

//Component initialization
160 /**
 * Builds the frame.
 *
 * @throws Exception An exception is thrown if the construction of the frame
 * is not successful
 */
165 private void jbInit () throws Exception {
    this.setResizable (false);
    this.setSize (new Dimension (398, 300));
    this.setTitle ("Menu");

170     gridLayout1.setColumns (3);
    gridLayout1.setHgap (5);
    gridLayout1.setRows (3);
    gridLayout1.setVgap (5);

175     contentPane = (JPanel) this.getContentPane ();
    contentPane.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    contentPane.setLayout (gridLayout1);

180     jButton1.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    jButton1.setName ("jButton1");
    jButton1.addKeyListener (new MenuFrame_jButton1_keyAdapter (this));
185     jButton1.addActionListener (new MenuFrame_jButton1_actionAdapter (this));

    jButton2.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    jButton2.setName ("jButton2");
190     jButton2.addActionListener (new MenuFrame_jButton2_actionAdapter (this));
    jButton2.addKeyListener (new MenuFrame_jButton2_keyAdapter (this));

    jButton3.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    jButton3.setName ("jButton3");
195     jButton3.addActionListener (new MenuFrame_jButton3_actionAdapter (this));
    jButton3.addKeyListener (new MenuFrame_jButton3_keyAdapter (this));

    jButton4.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    jButton4.setName ("jButton4");
200     jButton4.addActionListener (new MenuFrame_jButton4_actionAdapter (this));
    jButton4.addKeyListener (new MenuFrame_jButton4_keyAdapter (this));

    jButton5.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    jButton5.setName ("jButton5");
205     jButton5.addActionListener (new MenuFrame_jButton5_actionAdapter (this));
    jButton5.addKeyListener (new MenuFrame_jButton5_keyAdapter (this));

210
```

```

jButton6.setBackground (UIManager .getColor (
    "InternalFrame.activeTitleGradient" ));
jButton6.setName ("jButton6");
jButton6.addActionListener (new MenuFrame_jButton6_actionAdapter (this));
215 jButton6.addKeyListener (new MenuFrame_jButton6_keyAdapter (this));

jButton7.setBackground (UIManager .getColor (
    "InternalFrame.activeTitleGradient" ));
jButton7.setName ("jButton7");
220 jButton7.addActionListener (new MenuFrame_jButton7_actionAdapter (this));
jButton7.addKeyListener (new MenuFrame_jButton7_keyAdapter (this));

jButton8.setBackground (UIManager .getColor (
    "InternalFrame.activeTitleGradient" ));
225 jButton8.setName ("jButton8");
jButton8.addActionListener (new MenuFrame_jButton8_actionAdapter (this));
jButton8.addKeyListener (new MenuFrame_jButton8_keyAdapter (this));

jButton9.setBackground (UIManager .getColor (
230 "InternalFrame.activeTitleGradient" ));
jButton9.setName ("jButton9");
jButton9.addActionListener (new MenuFrame_jButton9_actionAdapter (this));
jButton9.addKeyListener (new MenuFrame_jButton9_keyAdapter (this));

235 Set mySet = new HashSet ();
mySet.add(AWTKeyStroke .getAWTKeyStroke (KeyEvent.VK_RIGHT, 0, false));
mySet.add(AWTKeyStroke .getAWTKeyStroke (KeyEvent.VK_TAB, 0, false));
this.setFocusTraversalKeys (KeyboardFocusManager .FORWARD_TRAVERSAL_KEYS ,
    mySet);

240 mySet.clear ();
mySet.add(AWTKeyStroke .getAWTKeyStroke (KeyEvent.VK_LEFT, 0, false));
mySet.add(AWTKeyStroke .getAWTKeyStroke (KeyEvent.VK_TAB,
    java.awt.event.InputEvent .
    SHIFT_DOWN_MASK , false));

245 this.setFocusTraversalKeys (KeyboardFocusManager .BACKWARD_TRAVERSAL_KEYS ,
    mySet);
mySet.clear ();

//Add the buttons to the contentPane
250 //Be careful not to change the order
contentPane.add(jButton1, null);
contentPane.add(jButton4, null);
contentPane.add(jButton7, null);
contentPane.add(jButton2, null);
255 contentPane.add(jButton5, null);
contentPane.add(jButton8, null);
contentPane.add(jButton3, null);
contentPane.add(jButton6, null);
contentPane.add(jButton9, null);

260
jButton [0] = null;
jButton [1] = jButton1;
jButton [2] = jButton2;
jButton [3] = jButton3;
265 jButton [4] = jButton4;
jButton [5] = jButton5;
jButton [6] = jButton6;
jButton [7] = jButton7;
jButton [8] = jButton8;
270 jButton [9] = jButton9;

vcp = new VoiceCallPanel (this, contentPane);
vcp.setVisible (false);
smsp = new SMSPanel (this, contentPane);
275 smsp.setVisible (false);
mmsp = new MMSPanel (this, contentPane);
mmsp.setVisible (false);
emailp = new EmailPanel (this, contentPane);
emailp.setVisible (false);
280 ep = new EditPanel (this, contentPane);

```

```

    ep.setVisible (false);
    vdcp = new VideoCallPanel (this, contentPane);
    vdcp.setVisible (false);
    vsp = new VideoStreamPanel (this, contentPane);
285    vsp.setVisible (false);
    wbp = new WebBrowsingPanel (this, contentPane);
    wbp.setVisible (false);
    ap = new ApplicationsPanel (this, contentPane);
    ap.setVisible (false);

290
    showTimeInTitleBar = new CalendarTitle (this);
    showTimeInTitleBar .start ();
}

295 //Overridden so we can exit when window is closed
/**
 * This method is overridden so we can exit when the window is closed.
 *
 * @param e WindowEvent
300 */
protected void processWindowEvent (WindowEvent e) {
    super.processWindowEvent (e);
    if (e.getID () == WindowEvent .WINDOW_CLOSING ) {
305        showTimeInTitleBar .stop ();
        System.exit (0);
    }
}

/**
310 * Hides the current panel and makes visible the <code>np</code> panel.
 *
 * @param np The new panel that will be shown
 */
private void showNextPanel (JPanel np) {
315    contentPane .setVisible (false);
    this.setContentPane (np);
    this.setVisible (false);
    this.setVisible (true);
    np.setVisible (true);
320 }

/**
 * Re-constructs the applications panel. In other words, it creates a new
 * <code>ApplicationsPanel</code> instance. Special care is taken for whether
325 * or not the new applications panel should be made visible or not.
 */
public void reConstructApplicationsPanel () {
    if (this.getContentPane () == ap) {
330        ap = new ApplicationsPanel (this, contentPane);
        ap.setVisible (false);
        showNextPanel (ap);
        ap.backButton .grabFocus ();
    }
    else {
335        ap = new ApplicationsPanel (this, contentPane);
    }
}

/**
340 * jButtonon1_actionPerformed
 *
 * @param e ActionEvent
 */
void jButtonon1_actionPerformed (ActionEvent e) {
345    showNextPanel (vcp);
}

/**
350 * jButtonon2_actionPerformed
 *
```



```
    * @param e ActionEvent
    */
    void jButton2_actionPerformed (ActionEvent e) {
355     showNextPanel (emailp);
    }

    /*
    * jButton3_actionPerformed
    *
    360    * @param e ActionEvent
    */
    void jButton3_actionPerformed (ActionEvent e) {
        showNextPanel (vsp);
    }
365

    /*
    * jButton4_actionPerformed
    *
    370    * @param e ActionEvent
    */
    void jButton4_actionPerformed (ActionEvent e) {
        showNextPanel (smsp);
    }

375    /*
    * jButton5_actionPerformed
    *
    * @param e ActionEvent
    */
380    void jButton5_actionPerformed (ActionEvent e) {
        showNextPanel (ep);
    }

    /*
385    * jButton6_actionPerformed
    *
    * @param e ActionEvent
    */
390    void jButton6_actionPerformed (ActionEvent e) {
        showNextPanel (wbp);
    }

    /*
395    * jButton7_actionPerformed
    *
    * @param e ActionEvent
    */
    void jButton7_actionPerformed (ActionEvent e) {
400     showNextPanel (mmsp);
    }

    /*
    * jButton8_actionPerformed
    *
    405    * @param e ActionEvent
    */
    void jButton8_actionPerformed (ActionEvent e) {
        showNextPanel (vdcp);
    }
410

    /*
    * jButton9_actionPerformed
    *
    * @param e ActionEvent
    */
415    void jButton9_actionPerformed (ActionEvent e) {
        ap = new ApplicationsPanel (this, contentPane);
        ap.setVisible (false);
        showNextPanel (ap);
420    }
```

```
    /*
    * jButton1_keyPressed
    *
425  * @param e KeyEvent
    */
    void jButton1_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }
430
    /*
    * jButton2_keyPressed
    *
435  * @param e KeyEvent
    */
    void jButton2_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }

440  /*
    * jButton3_keyPressed
    *
    * @param e KeyEvent
    */
445  void jButton3_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }

450  /*
    * jButton4_keyPressed
    *
    * @param e KeyEvent
    */
455  void jButton4_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }

460  /*
    * jButton5_keyPressed
    *
    * @param e KeyEvent
    */
465  void jButton5_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }

470  /*
    * jButton6_keyPressed
    *
    * @param e KeyEvent
    */
475  void jButton6_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }

480  /*
    * jButton7_keyPressed
    *
    * @param e KeyEvent
    */
485  void jButton7_keyPressed (KeyEvent e) {
        keyNavigation (e);
    }

490  /*
    * jButton8_keyPressed
    *
    * @param e KeyEvent
    */
    void jButton8_keyPressed (KeyEvent e) {
```

```

        keyNavigation (e);
    }

    /*
495     * jButton9_keyPressed
    *
    * @param e KeyEvent
    */
    void jButton9_keyPressed (KeyEvent e) {
500     keyNavigation (e);
    }

    /*
505     * keyNavigation
    *
    * @param e KeyEvent
    */
    /**
510     * This method is used to assist in navigating the main menu with the keyboard
    * arrows.
    *
    * @param e KeyEvent
    */
    void keyNavigation (KeyEvent e) {
515     JButton myButton = (JButton) e.getSource ();
        String buttonName = myButton.getName ();
        int but = new Integer (" " + buttonName.charAt (7)).intValue ();

        if (e.getKeyCode () == KeyEvent.VK_UP) {
520             but = but - 1;
                if (but == 0) {
                    but = 9;
                }
                jButton [but].grabFocus ();
525         }
        else if (e.getKeyCode () == KeyEvent.VK_DOWN) {
            but = (but % 9) + 1;
            jButton [but].grabFocus ();
        }
530     }
    }

    class MenuFrame_jButtonon1_keyAdapter
535     extends java.awt.event.KeyAdapter {
        MenuFrame adaptee;

        MenuFrame_jButtonon1_keyAdapter (MenuFrame adaptee) {
540             this.adaptee = adaptee;
        }

        public void keyPressed (KeyEvent e) {
            adaptee.jButton1_keyPressed (e);
        }
545     }

    class MenuFrame_jButtonon2_keyAdapter
        extends java.awt.event.KeyAdapter {
        MenuFrame adaptee;

550     MenuFrame_jButtonon2_keyAdapter (MenuFrame adaptee) {
            this.adaptee = adaptee;
        }

555     public void keyPressed (KeyEvent e) {
            adaptee.jButton2_keyPressed (e);
        }
    }

560     class MenuFrame_jButtonon3_keyAdapter

```

```
        extends java.awt.event.KeyAdapter {
    MenuFrame adaptee;

    MenuFrame_jButton3_keyAdapter (MenuFrame adaptee) {
565         this.adaptee = adaptee;
    }

    public void keyPressed (KeyEvent e) {
        adaptee.jButton3_keyPressed (e);
570     }
    }

    class MenuFrame_jButton4_keyAdapter
        extends java.awt.event.KeyAdapter {
575     MenuFrame adaptee;

    MenuFrame_jButton4_keyAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }
580     public void keyPressed (KeyEvent e) {
        adaptee.jButton4_keyPressed (e);
    }
    }
585     class MenuFrame_jButton5_keyAdapter
        extends java.awt.event.KeyAdapter {
    MenuFrame adaptee;

590     MenuFrame_jButton5_keyAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }

    public void keyPressed (KeyEvent e) {
595         adaptee.jButton5_keyPressed (e);
    }
    }

    class MenuFrame_jButton6_keyAdapter
600     extends java.awt.event.KeyAdapter {
    MenuFrame adaptee;

    MenuFrame_jButton6_keyAdapter (MenuFrame adaptee) {
605         this.adaptee = adaptee;
    }

    public void keyPressed (KeyEvent e) {
        adaptee.jButton6_keyPressed (e);
    }
610 }

    class MenuFrame_jButton7_keyAdapter
        extends java.awt.event.KeyAdapter {
    MenuFrame adaptee;
615     MenuFrame_jButton7_keyAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }

620     public void keyPressed (KeyEvent e) {
        adaptee.jButton7_keyPressed (e);
    }
    }

625     class MenuFrame_jButton8_keyAdapter
        extends java.awt.event.KeyAdapter {
    MenuFrame adaptee;

    MenuFrame_jButton8_keyAdapter (MenuFrame adaptee) {
630         this.adaptee = adaptee;
```

```
    }

    public void keyPressed (KeyEvent e) {
        adaptee.jButton8_keyPressed (e);
635    }
    }

    class MenuFrame_jButton9_keyAdapter
        extends java.awt.event.KeyAdapter {
640    MenuFrame adaptee;

    MenuFrame_jButton9_keyAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }

645    public void keyPressed (KeyEvent e) {
        adaptee.jButton9_keyPressed (e);
    }
    }

650    class MenuFrame_jButton1_actionAdapter
        implements java.awt.event.ActionListener {
        MenuFrame adaptee;

655    MenuFrame_jButton1_actionAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
660    adaptee.jButton1_actionPerformed (e);
    }
    }

    class MenuFrame_jButton2_actionAdapter
665    implements java.awt.event.ActionListener {
        MenuFrame adaptee;

    MenuFrame_jButton2_actionAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
670    }

    public void actionPerformed (ActionEvent e) {
        adaptee.jButton2_actionPerformed (e);
    }
675    }

    class MenuFrame_jButton3_actionAdapter
        implements java.awt.event.ActionListener {
        MenuFrame adaptee;

680    MenuFrame_jButton3_actionAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }

685    public void actionPerformed (ActionEvent e) {
        adaptee.jButton3_actionPerformed (e);
    }
    }

690    class MenuFrame_jButton4_actionAdapter
        implements java.awt.event.ActionListener {
        MenuFrame adaptee;

    MenuFrame_jButton4_actionAdapter (MenuFrame adaptee) {
695    this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.jButton4_actionPerformed (e);
700    }
    }
```

```
    }

    class MenuFrame_jButton5_actionAdapter
        implements java.awt.event.ActionListener {
705     MenuFrame adaptee;

        MenuFrame_jButton5_actionAdapter (MenuFrame adaptee) {
            this.adaptee = adaptee;
        }

710     public void actionPerformed (ActionEvent e) {
        adaptee.jButton5_actionPerformed (e);
    }
}

715 class MenuFrame_jButton6_actionAdapter
    implements java.awt.event.ActionListener {
        MenuFrame adaptee;

720     MenuFrame_jButton6_actionAdapter (MenuFrame adaptee) {
        this.adaptee = adaptee;
    }

        public void actionPerformed (ActionEvent e) {
725     adaptee.jButton6_actionPerformed (e);
    }
}

    class MenuFrame_jButton7_actionAdapter
730     implements java.awt.event.ActionListener {
        MenuFrame adaptee;

        MenuFrame_jButton7_actionAdapter (MenuFrame adaptee) {
735     this.adaptee = adaptee;
    }

        public void actionPerformed (ActionEvent e) {
        adaptee.jButton7_actionPerformed (e);
    }
740 }

    class MenuFrame_jButton8_actionAdapter
        implements java.awt.event.ActionListener {
745     MenuFrame adaptee;

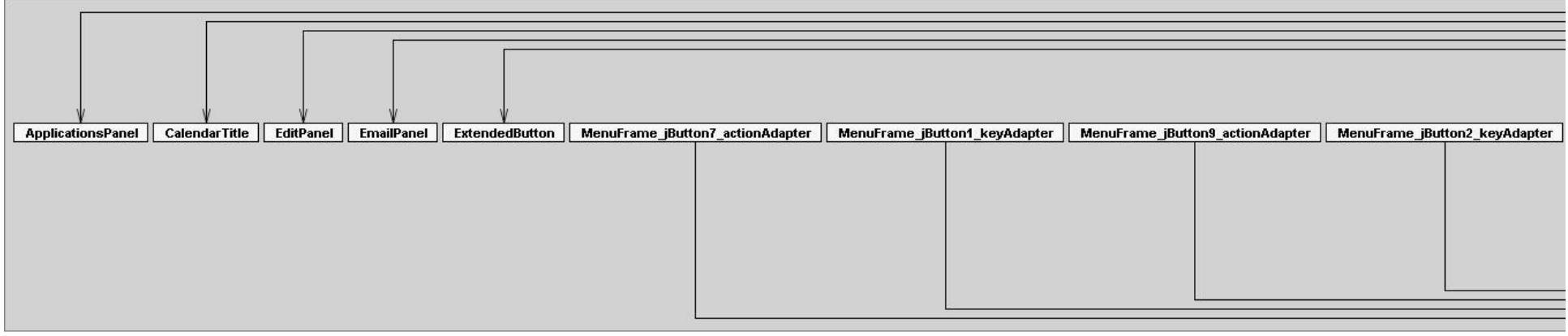
        MenuFrame_jButton8_actionAdapter (MenuFrame adaptee) {
            this.adaptee = adaptee;
        }

750     public void actionPerformed (ActionEvent e) {
        adaptee.jButton8_actionPerformed (e);
    }
}

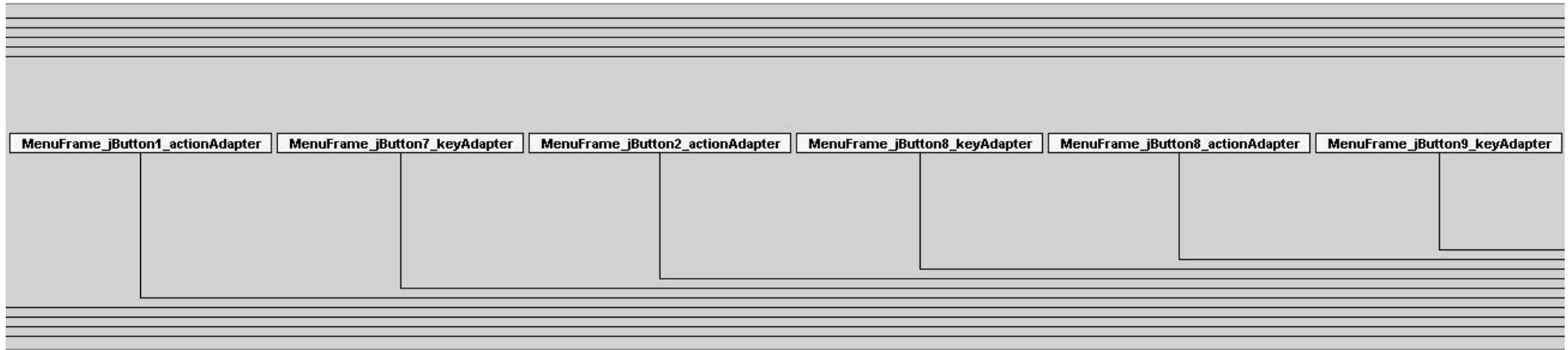
755 class MenuFrame_jButton9_actionAdapter
    implements java.awt.event.ActionListener {
        MenuFrame adaptee;

        MenuFrame_jButton9_actionAdapter (MenuFrame adaptee) {
760     this.adaptee = adaptee;
    }

        public void actionPerformed (ActionEvent e) {
765     adaptee.jButton9_actionPerformed (e);
    }
}
```

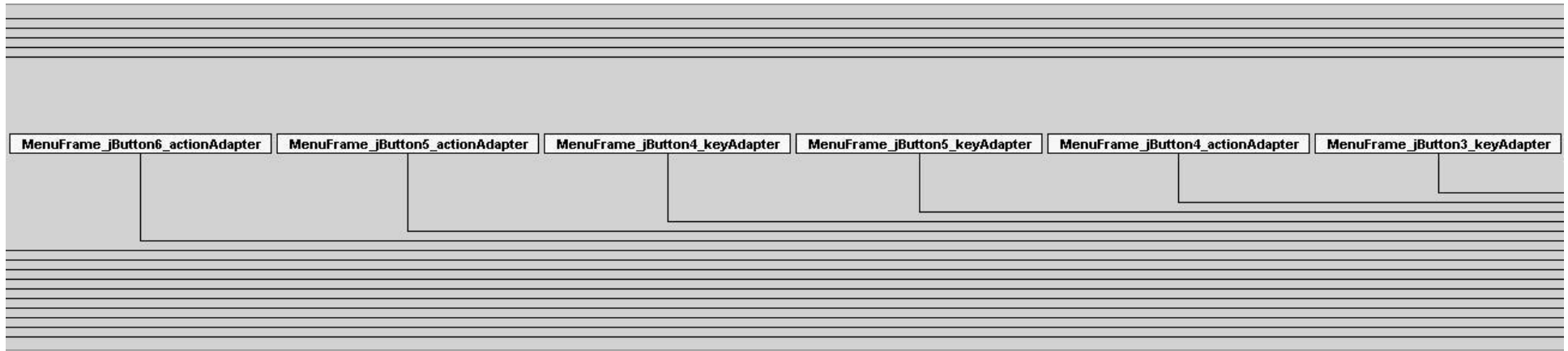


UML Διάγραμμα - Τμήμα [2]

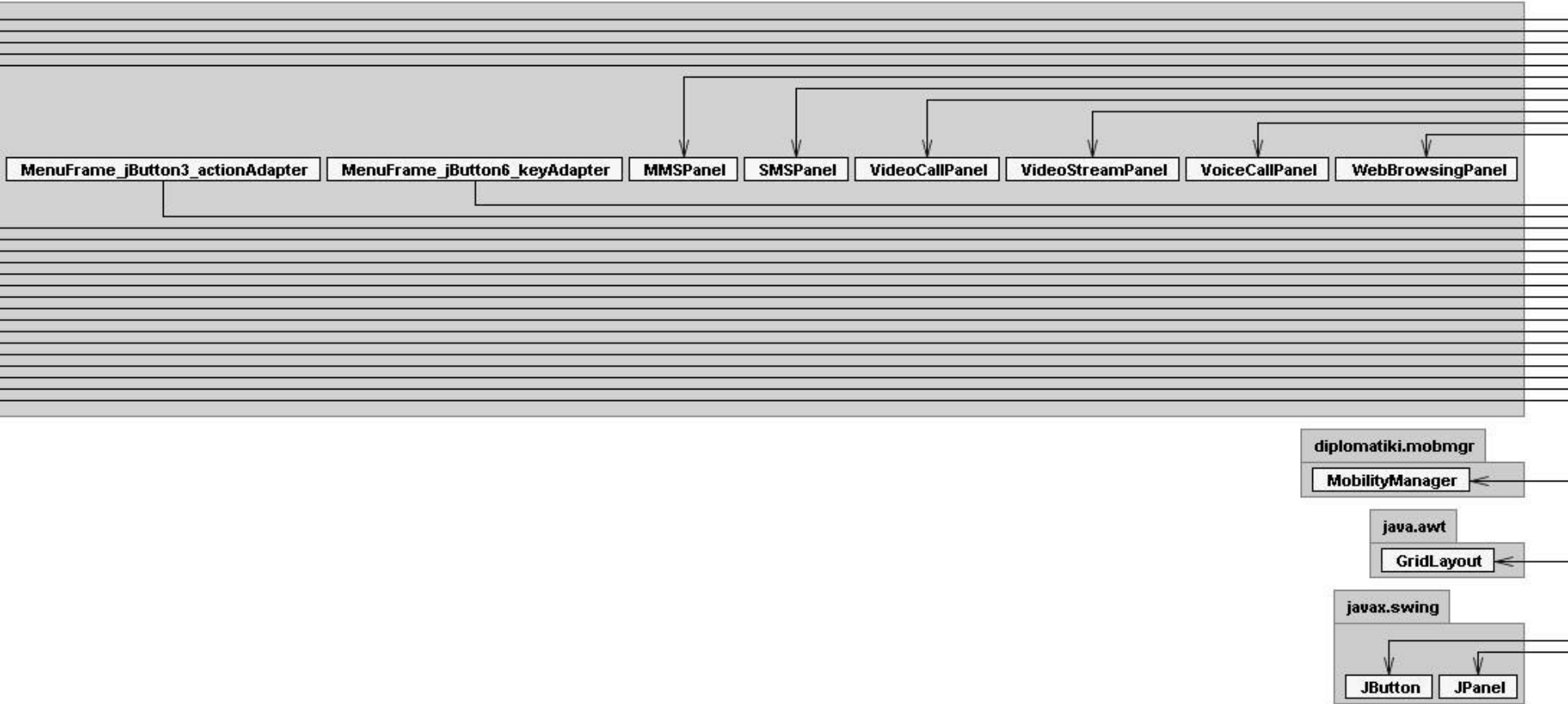


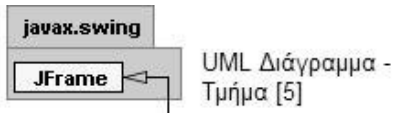


UML Διάγραμμα - Τμήμα [3]

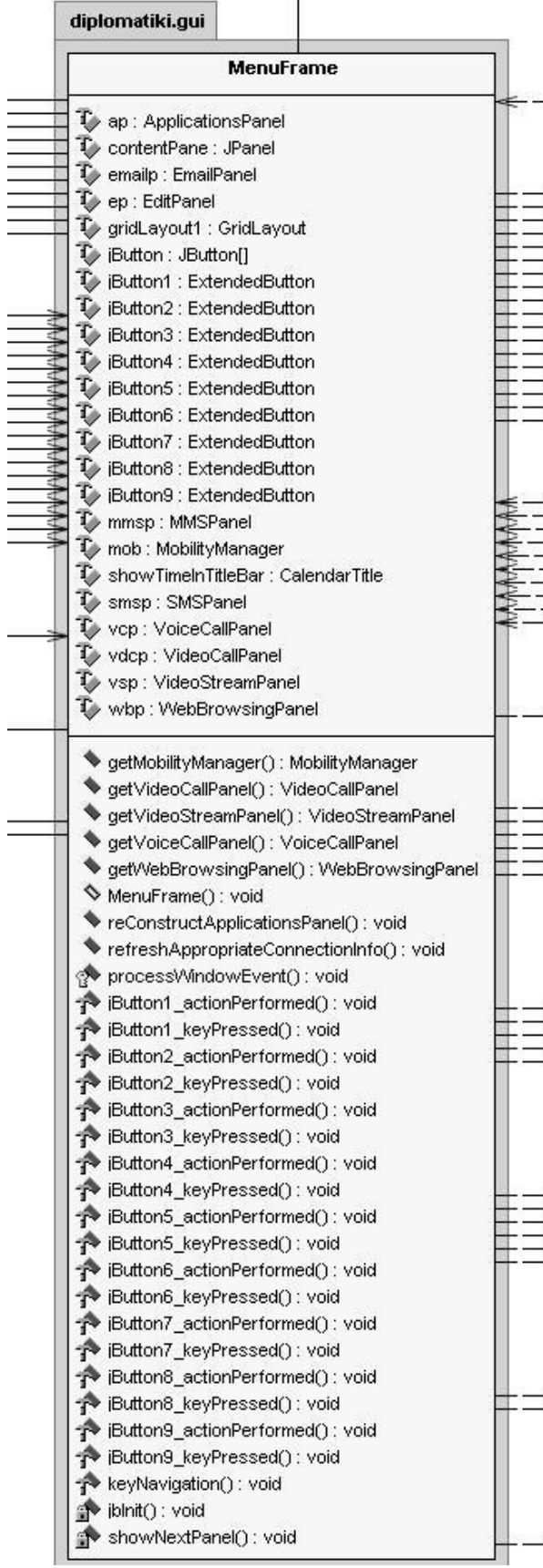


UML Διάγραμμα - Τμήμα [4]

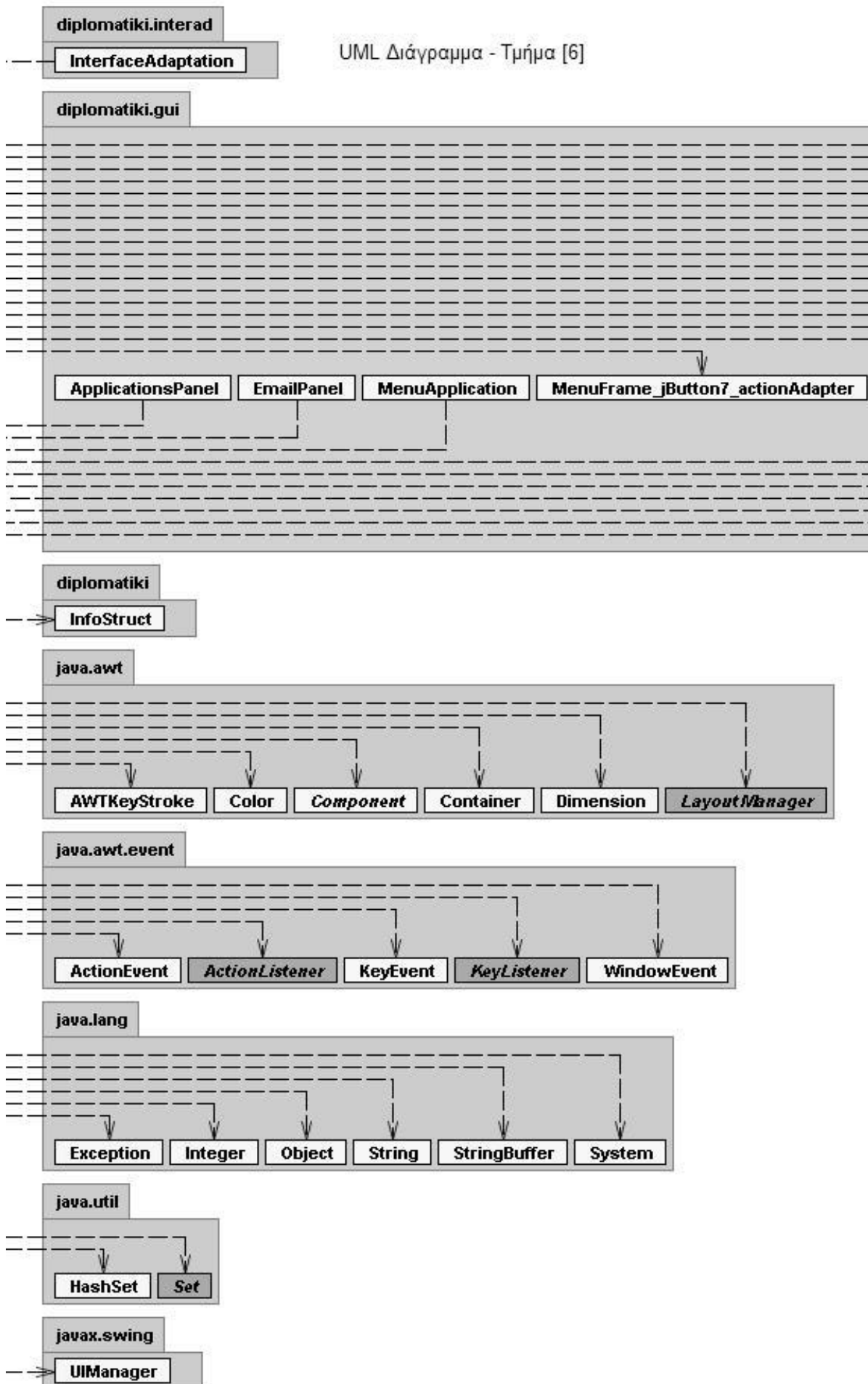




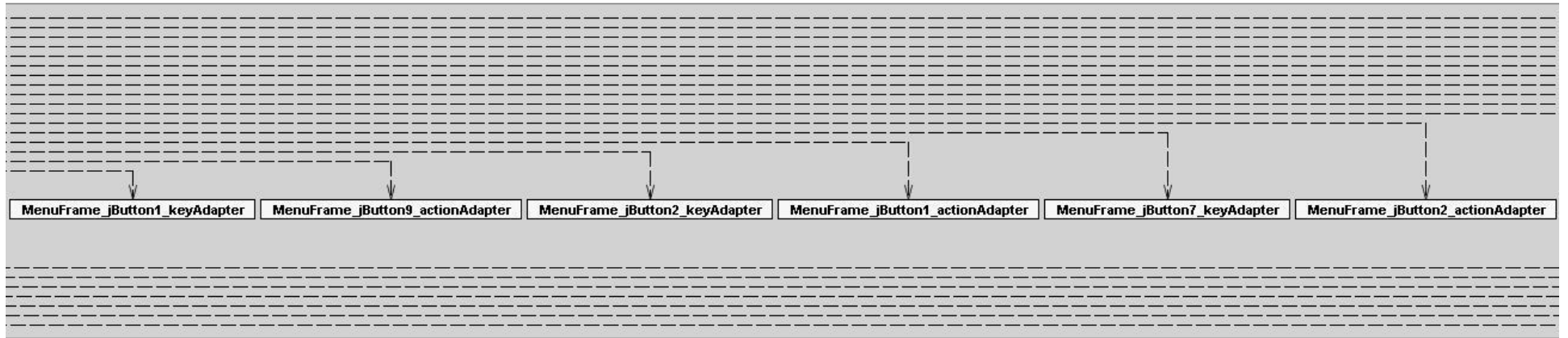
UML Διάγραμμα - Τμήμα [5]



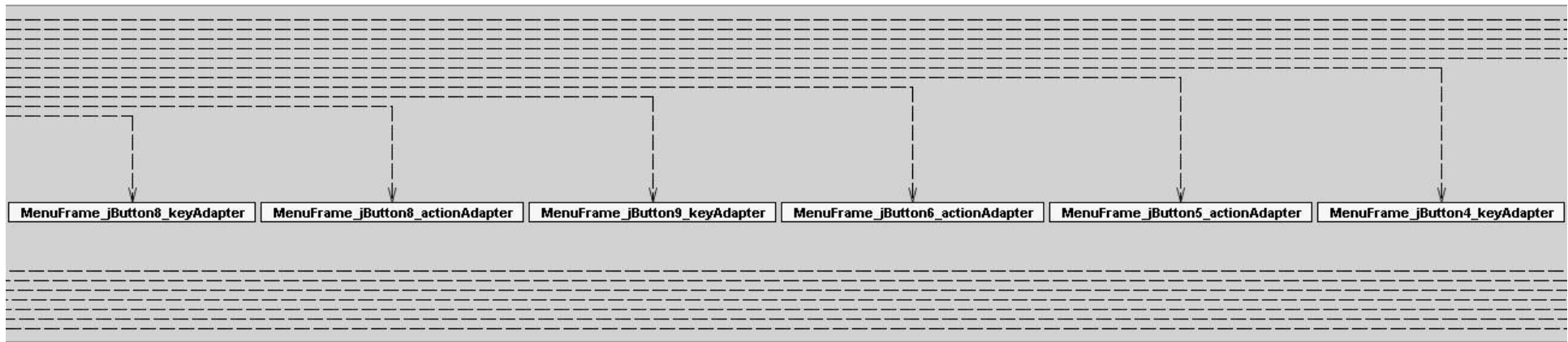
UML Διάγραμμα - Τμήμα [6]



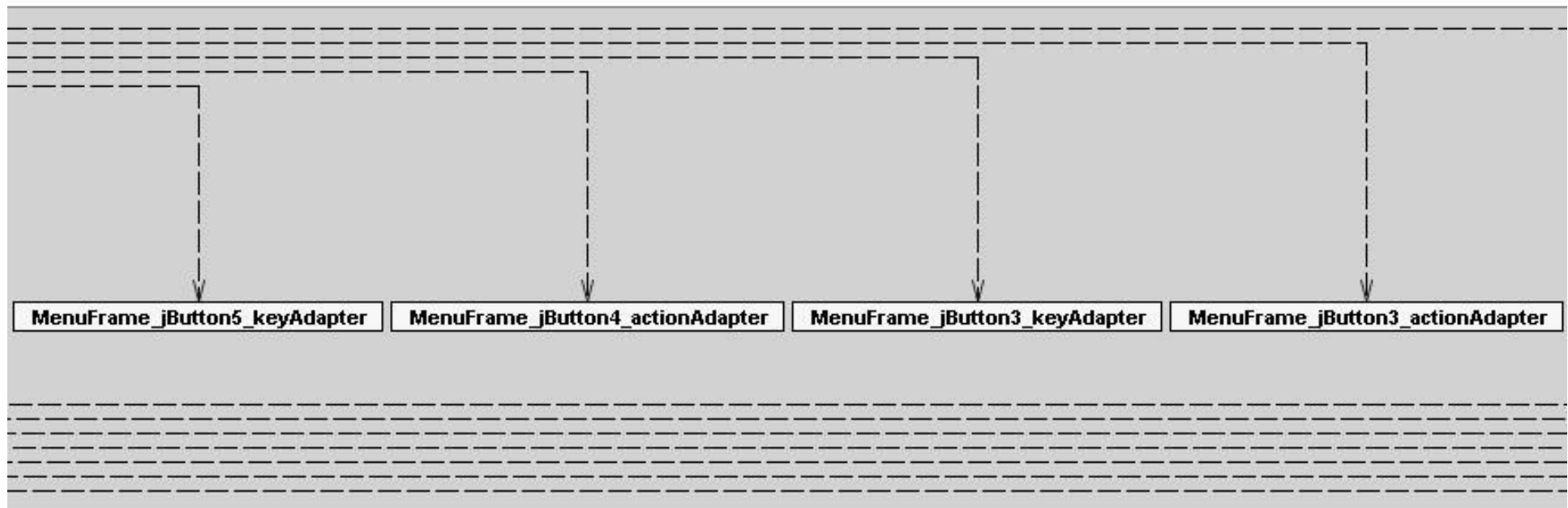
UML Διάγραμμα - Τμήμα [7]



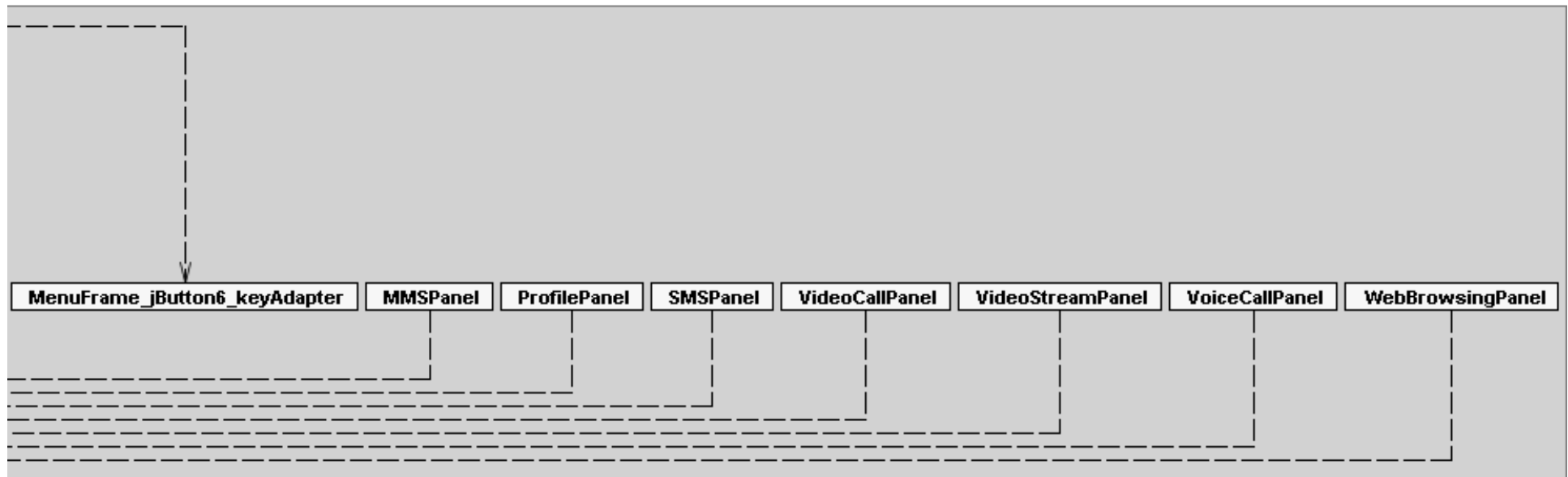
UML Διάγραμμα - Τμήμα [8]



UML Διάγραμμα - Τμήμα [9]



UML Διάγραμμα - Τμήμα [10]





```

package diplomatiki .gui;

import java .awt .*;
import javax .swing .*;
5 import javax .swing .border .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>MessageWindow</code> class</p>
10 *
 * <p>Description: Instances of this class are windows that appear for only a
 * few seconds and show a message to the user (e.g. a "Service Unavailable"
 * message). They run as separate threads.</p>
 *
15 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
 */
public class MessageWindow
    extends JWindow
    implements Runnable {
25     final long SHOW_TIME = Config.SHOW_TIME ;
    final String WEST_ICON_PATH = Config.WARNING_ICON_PATH ;

    Thread runner = null;
    JPanel winPanel = new JPanel ();
30    BorderLayout borderLayout1 = new BorderLayout ();
    JLabel northLabel = new JLabel ();
    JLabel westLabel = new JLabel ();
    JLabel centerLabel = new JLabel ();
    //JLabel againLabel = new JLabel ();
35    String northString = "Connection Error:" ;
    String centerString = "Service Unavailable" ;

    /**
     * Creates a new <code>MessageWindow</code> instance.
40     */
    public MessageWindow () {
        super ();
        try {
            jbInit ();
45        }
        catch (Exception e) {
            e.printStackTrace ();
        }
    }
50

    /**
     * Creates a new <code>MessageWindow</code> instance with the specified parent
     * frame.
     *
55     * @param parentFrame The window's parent frame
     */
    public MessageWindow (JFrame parentFrame) {
        super (parentFrame);
        try {
60            jbInit ();
        }
        catch (Exception e) {
            e.printStackTrace ();
        }
65    }

    /**
     * Creates a new <code>MessageWindow</code> instance with the specified parent
     * frame and message.
70     */

```

```

    * @param parentFrame The window's parent frame
    * @param northString The window's title message
    * @param centerString The window's main message
    */
75  public MessageWindow (JFrame parentFrame , String northString ,
        String centerString) {
    super (parentFrame);
    this.northString = northString;
    this.centerString = centerString;
80  try {
    jbInit ();
    }
    catch (Exception e) {
    e.printStackTrace ();
85  }
    }

    /**
    * Builds the window.
    *
    * @throws Exception An exception is thrown if the construction of the window
    * is not successful
    */
95  private void jbInit () throws Exception {
    winPanel.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    winPanel.setBorder (new EtchedBorder ());
    winPanel.setLayout (borderLayout1);
    winPanel.setMaximumSize (new Dimension (200, 100));
100
    northLabel.setFont (new java.awt.Font ("Dialog", 1, 12));
    northLabel.setHorizontalAlignment (SwingConstants.CENTER);
    northLabel.setText (northString);

105
    westLabel.setHorizontalAlignment (SwingConstants.CENTER);
    westLabel.setText ("");
    westLabel.setIcon (new ImageIcon (WEST_ICON_PATH));

    centerLabel.setFont (new java.awt.Font ("Dialog", 1, 12));
110
    centerLabel.setHorizontalAlignment (SwingConstants.CENTER);
    centerLabel.setText (centerString);

    /*
    againLabel.setFont (new java.awt.Font ("Dialog", 2, 11));
115
    againLabel.setHorizontalAlignment (SwingConstants.CENTER);
    againLabel.setText ("Please try again later");
    */

    winPanel.add (northLabel , BorderLayout.NORTH);
120
    winPanel.add (westLabel , BorderLayout.WEST);
    winPanel.add (centerLabel , BorderLayout.CENTER);
    //winPanel.add (againLabel , BorderLayout.SOUTH);

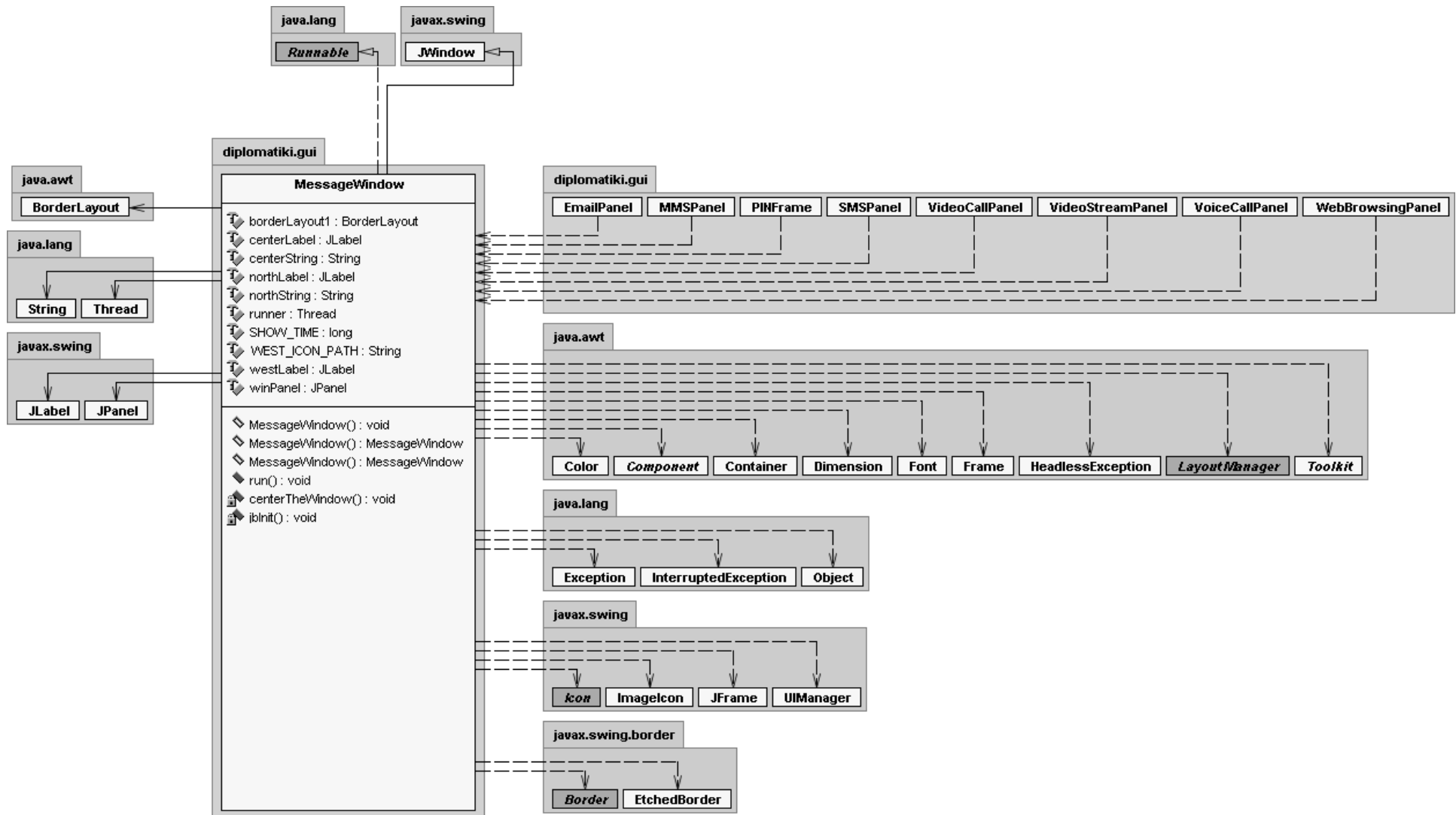
    this.getContentPane ().add (winPanel);
125
    this.setSize (new Dimension (200, 100));
    this.validate ();

    //centerTheWindow ();

130
    if (runner == null) {
    runner = new Thread (this);
    runner.start ();
    }
    }
135
    /**
    * Centers the window on screen.
    *
    * @throws HeadlessException Exception that could be thrown
    */
140

```

```
private void centerTheWindow () throws HeadlessException {
    //Center the window
    Dimension screenSize = Toolkit.getDefaultToolkit ().getScreenSize ();
    Dimension frameSize = this.getSize ();
145    //Dimension frameSize = new Dimension(200, 100);
    if (frameSize.height > screenSize.height) {
        frameSize.height = screenSize.height;
    }
    if (frameSize.width > screenSize.width) {
150        frameSize.width = screenSize.width;
    }
    this.setLocation ( (screenSize.width - frameSize.width) / 2,
                       (screenSize.height - frameSize.height) / 2);
}
155
/**
 * Entry point for the <code>MessageWindow</code> thread. It makes the window
 * visible, waits for a few seconds, and then hides the window.
 */
160 public void run() {
    centerTheWindow ();
    this.setVisible (true);
    try {
        runner.sleep (SHOW_TIME);
165    }
    catch (InterruptedException ex) {
        ex.printStackTrace ();
    }
    this.setVisible (false);
170 }
}
```



```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>MMSInputPanel</code> class</p>
10 *
 * <p>Description: This is the center panel of the MMS Service panel. It holds
 * the various input fields, where the user should fill in the appropriate
 * information (e.g. the recipient's address).</p>
 *
15 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
 */
public class MMSInputPanel
    extends JPanel {
    final String LOAD_DEFAULT_ICON_PATH = Config.LOAD_DEFAULT_ICON_PATH ;
25    final String LOAD_SELECTED_ICON_PATH = Config.LOAD_SELECTED_ICON_PATH ;

    JLabel numberLabel = new JLabel ();
    JTextField numberField = new JTextField ();
    JLabel messageLabel = new JLabel ();
30    JScrollPane scroll = new JScrollPane ();
    JTextArea message = new JTextArea ();
    JTextField pictureField = new JTextField ();
    JTextField soundField = new JTextField ();
    JLabel pictureLabel = new JLabel ();
35    JLabel soundLabel = new JLabel ();
    ExtendedButton load1;
    ExtendedButton load2;
    JFileChooser jFileChooser1 = new JFileChooser ();
    JFileChooser jFileChooser2 = new JFileChooser ();
40

    /**
     * Creates a new <code>MMSInputPanel</code> instance.
     */
    public MMSInputPanel () {
45        try {
            jbInit ();
        }
        catch (Exception ex) {
            ex.printStackTrace ();
50        }
    }

    /**
     * Builds the panel.
55     *
     * @throws Exception An exception is thrown if the construction of the panel
     * is not successful
     */
    void jbInit () throws Exception {
60        this.setBackground (UIManager.getColor ("InternalFrame.activeTitleGradient" ));
        this.setMinimumSize (new Dimension (300, 140));
        this.setMaximumSize (new Dimension (300, 140));
        this.setPreferredSize (new Dimension (300, 140));
        this.setLayout (null);
65

        numberLabel.setFont (new java .awt .Font ("Microsoft Sans Serif" , 1, 11));
        numberLabel.setText ("Number");
        numberLabel.setBounds (new Rectangle (40, 9, 51, 15));

70        numberField.setText ("");

```

```

        numberField.setBounds(new Rectangle(103, 7, 74, 21));

        messageLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
        messageLabel.setText("Message");
75    messageLabel.setBounds(new Rectangle(40, 34, 55, 15));

        scroll.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_NEVER);
        scroll.getViewport().setBackground(UIManager.getColor(
            "InternalFrame.activeTitleGradient"));
80    scroll.setBounds(new Rectangle(103, 32, 144, 43));

        message.setText("");
        message.setLineWrap(true);
        message.addKeyListener(new MMSInputPanel_message_keyAdapter(this));
85    scroll.getViewport().add(message, null);

        pictureLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
        pictureLabel.setText("Picture");
90    pictureLabel.setBounds(new Rectangle(40, 83, 40, 15));

        pictureField.setText("");
        pictureField.setBounds(new Rectangle(103, 79, 74, 21));

95    load1 = new ExtendedButton(Load.DEFAULT_ICON_PATH, Load.SELECTED_ICON_PATH);
        load1.setBounds(new Rectangle(182, 80, 65, 20));
        load1.addActionListener(new MMSInputPanel_load1_actionAdapter(this));

        soundLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
100    soundLabel.setText("Sound");
        soundLabel.setBounds(new Rectangle(40, 107, 39, 15));

        soundField.setText("");
        soundField.setBounds(new Rectangle(103, 104, 74, 21));
105

        load2 = new ExtendedButton(Load.DEFAULT_ICON_PATH, Load.SELECTED_ICON_PATH);
        load2.setBounds(new Rectangle(182, 105, 65, 20));
        load2.addActionListener(new MMSInputPanel_load2_actionAdapter(this));

110    this.add(numberLabel, null);
        this.add(numberField, null);
        this.add(scroll, null);
        this.add(messageLabel, null);
        this.add(pictureLabel, null);
115    this.add(pictureField, null);
        this.add(load1, null);
        this.add(soundLabel, null);
        this.add(soundField, null);
        this.add(load2, null);
120    }

    /**
     * message_keyPressed
     *
125    * @param e KeyEvent
     */
    void message_keyPressed(KeyEvent e) {
        if (e.getKeyCode() == KeyEvent.VK_DOWN) {
130            message.transferFocus();
        }
        else if (e.getKeyCode() == KeyEvent.VK_UP) {
            message.transferFocusBackward();
        }
    }
135

    /**
     * Action performed when the first "Load" button is pressed. The
     * <code>loadFile()</code> method is invoked.
     *
140    * @param e ActionEvent

```

```

    */
    void load1_actionPerformed (ActionEvent e) {
        loadFile(jFileChooser1 , pictureField);
    }
145
    /**
     * Action performed when the second "Load" button is pressed. The
     * <code>loadFile()</code> method is invoked.
     *
     * @param e ActionEvent
     */
150
    void load2_actionPerformed (ActionEvent e) {
        loadFile(jFileChooser2 , soundField);
    }
155
    /**
     * A dialog window pops up, so the user may choose a file (as an attachment to
     * the e-mail).
     *
     * @param jfc File chooser object
     * @param textField The text field in which the filepath will be written
     */
160
    void loadFile (JFileChooser jfc, JTextField textField) {
        if (JFileChooser.APPROVE_OPTION == jfc.showOpenDialog (this)) {
165
            String filename = jfc.getSelectedFile ().getPath ();
            textField.setText (filename);
        }
    }
170
}

class MMSInputPanel_message_keyAdapter
    extends java.awt.event.KeyAdapter {
    MMSInputPanel adaptee;
175

    MMSInputPanel_message_keyAdapter (MMSInputPanel adaptee) {
        this.adaptee = adaptee;
    }

180
    public void keyPressed (KeyEvent e) {
        adaptee.message_keyPressed (e);
    }
}

185
class MMSInputPanel_load1_actionAdapter
    implements java.awt.event.ActionListener {
    MMSInputPanel adaptee;

    MMSInputPanel_load1_actionAdapter (MMSInputPanel adaptee) {
190
        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.load1_actionPerformed (e);
195
    }
}

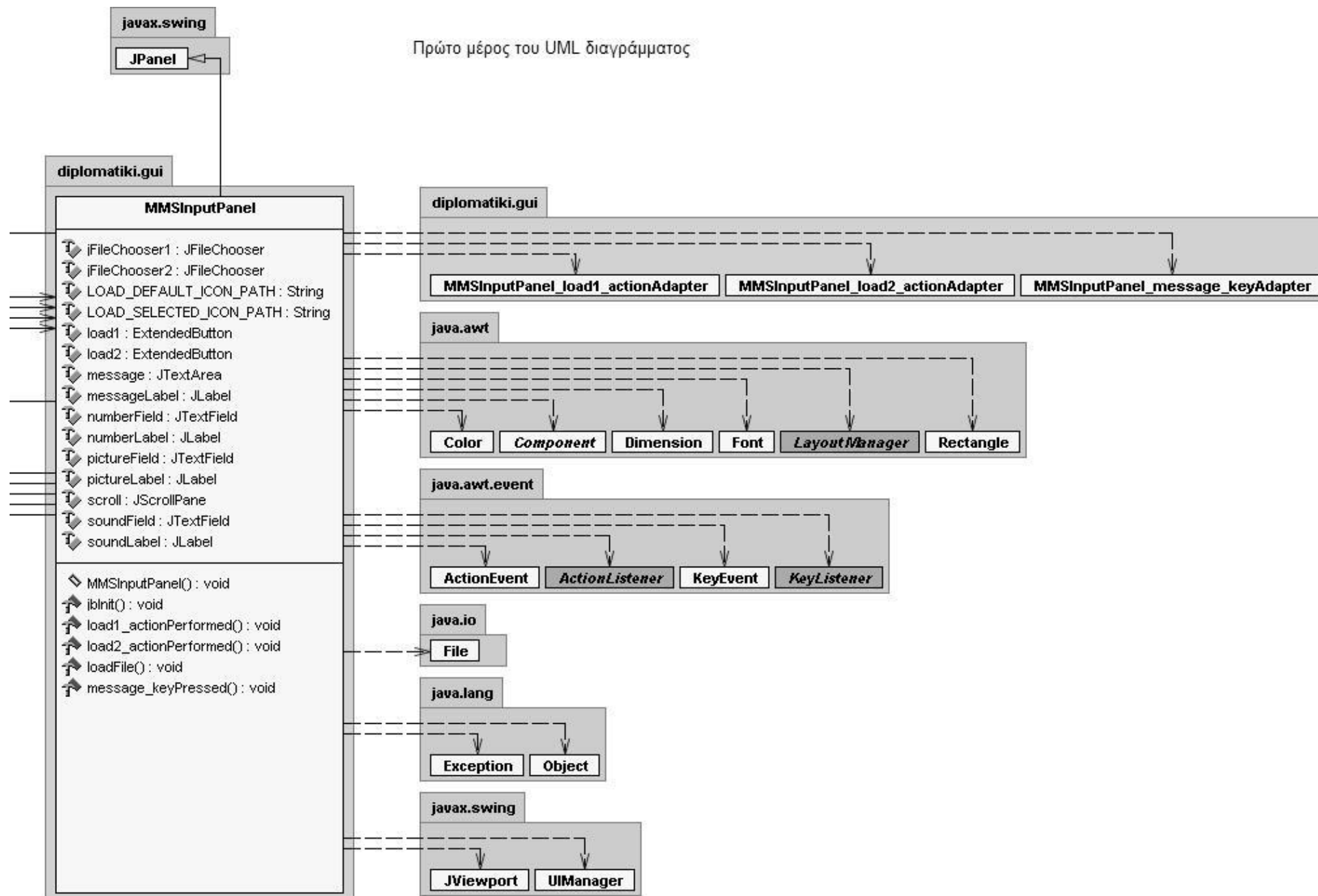
class MMSInputPanel_load2_actionAdapter
    implements java.awt.event.ActionListener {
200
    MMSInputPanel adaptee;

    MMSInputPanel_load2_actionAdapter (MMSInputPanel adaptee) {
        this.adaptee = adaptee;
    }

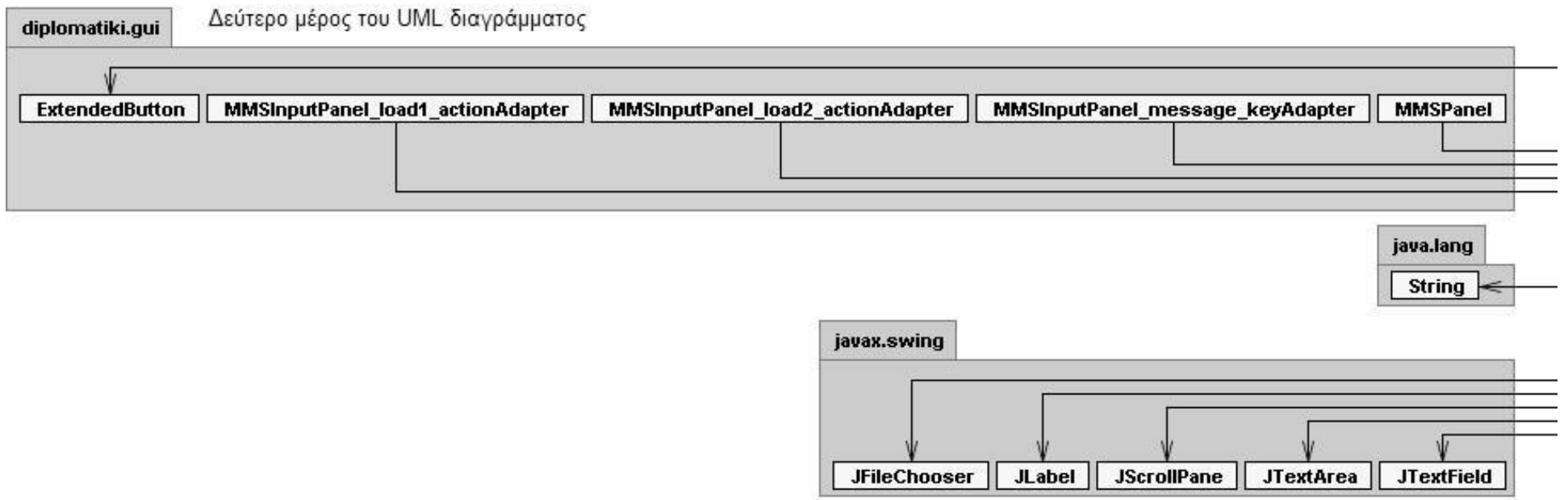
205
    public void actionPerformed (ActionEvent e) {
        adaptee.load2_actionPerformed (e);
    }
}
210

```

Πρώτο μέρος του UML διαγράμματος







```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>MMSPanel</code> class</p>
10 *
 * <p>Description: This is the MMS Service panel. This panel is shown when the
 * user chooses to send an MMS.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class MMSPanel
    extends JPanel {
    final int SERVICE_CODE = Config.MMS_SERVICE_CODE ;
    final int MAX_TIME = Config.BAR_MAX_TIME ;

25
    final String START_DEFAULT_ICON_PATH = Config.MMS_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config.MMS_START_SELECTED_ICON_PATH ;
    final String EDIT_DEFAULT_ICON_PATH = Config.EDIT_DEFAULT_ICON_PATH ;
    final String EDIT_SELECTED_ICON_PATH = Config.EDIT_SELECTED_ICON_PATH ;

30
    InfoPanel ip;
    ExtendedButton startButton , settingsButton ;
    MMSInputPanel input;

35
    /**
     * Creates a new <code>MMSPanel</code> instance.
     */
    public MMSPanel () {
40
    }

    /**
     * Creates a new <code>MMSPanel</code> instance with the specified
     * parent frame and previous panel.
     *
45 * @param myFrame The frame which this panel belongs to
     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
     * wants to return to the previous panel
     */
50 public MMSPanel (JFrame myFrame , JPanel previousPanel ) {
    super (myFrame , previousPanel );

    GridLayout1 .setColumns (1);
    GridLayout1 .setRows (2);

55
    startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
        START_SELECTED_ICON_PATH );
    settingsButton = new ExtendedButton (EDIT_DEFAULT_ICON_PATH ,
        EDIT_SELECTED_ICON_PATH );

60
    startButton .addActionListener (new MMSPanel_startButton_actionAdapter (this));
    settingsButton .
        addActionListener (new MMSPanel_settingsButton_actionAdapter (this));

65
    northPanel .setMinimumSize (new Dimension (400 , 60));
    northPanel .setMaximumSize (new Dimension (400 , 60));
    northPanel .setPreferredSize (new Dimension (400 , 60));
    northPanel .add (startButton );
    northPanel .add (settingsButton );

70

```

```

        input = new MMSInputPanel ();

        centerPanel.add(input);
    }
75
    /**
     * Action performed when the "Start" service button is pressed.
     */
    private void startAction () {
80
        //Check if the address/number field is filled in
        if (input.numberField.getText().equals("")) {
            new MessageWindow (myFrame, "Input Required:", "Please type a number" );
            return;
        }
85

        //Run IIS
        diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
            executeIIS (SERVICE_CODE, true, -1);

90
        //Check if myInfo is null
        if (myInfo == null) {
            new MessageWindow (myFrame);
            return;
        }
95

        //Construct ConnectionInfoPanel
        ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
            getAccessTechnology (),
            myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());
100

        //Construct InfoPanel
        ip = new InfoPanel (myFrame, this, "Sending MMS to: ",
            input.numberField.getText ());

105
        //Add the ConnectionInfoPanel to the InfoPanel
        ip.centerPanel.add(cip);

        //Create ProgressBar and Add it to the InfoPanel
        ProgressBar bar = new ProgressBar (MAX_TIME);
110
        //ip.add("East", bar);
        ip.add(bar, BorderLayout.EAST);

        //Show the InfoPanel
        this.setVisible (false);
115
        myFrame.setContentPane (ip);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
        ip.setVisible (true);

120
        //Start the ProgressBar
        bar.start ();
    }

    /**
125
     * Action performed when the "Edit Settings" button is pressed.
     */
    private void settingsAction () {
        //Construct ProfilePanel
        ProfilePanel pp = new ProfilePanel (myFrame, this, 1);
130
        //Disable all other sliders in the ProfilePanel
        pp.disableAllSlidersExceptOne (1);

        //Show the ProfilePanel
        this.setVisible (false);
135
        myFrame.setContentPane (pp);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
        pp.setVisible (true);
    }
140

```

```
/**
 * Action performed when the "Start" service button is pressed. The
 * <code>startAction()</code> method is invoked.
 *
145  * @param e ActionEvent
 */
void startButton_actionPerformed (ActionEvent e) {
    startAction ();
}

150 /**
 * Action performed when the "Edit Settings" button is pressed. The
 * <code>settingsAction()</code> method is invoked.
 *
155  * @param e ActionEvent
 */
void settingsButton_actionPerformed (ActionEvent e) {
    settingsAction ();
}

160 }

class MMSPanel_startButton_actionAdapter
    implements java.awt.event.ActionListener {
165     MMSPanel adaptee;

    MMSPanel_startButton_actionAdapter (MMSPanel adaptee) {
        this.adaptee = adaptee;
    }

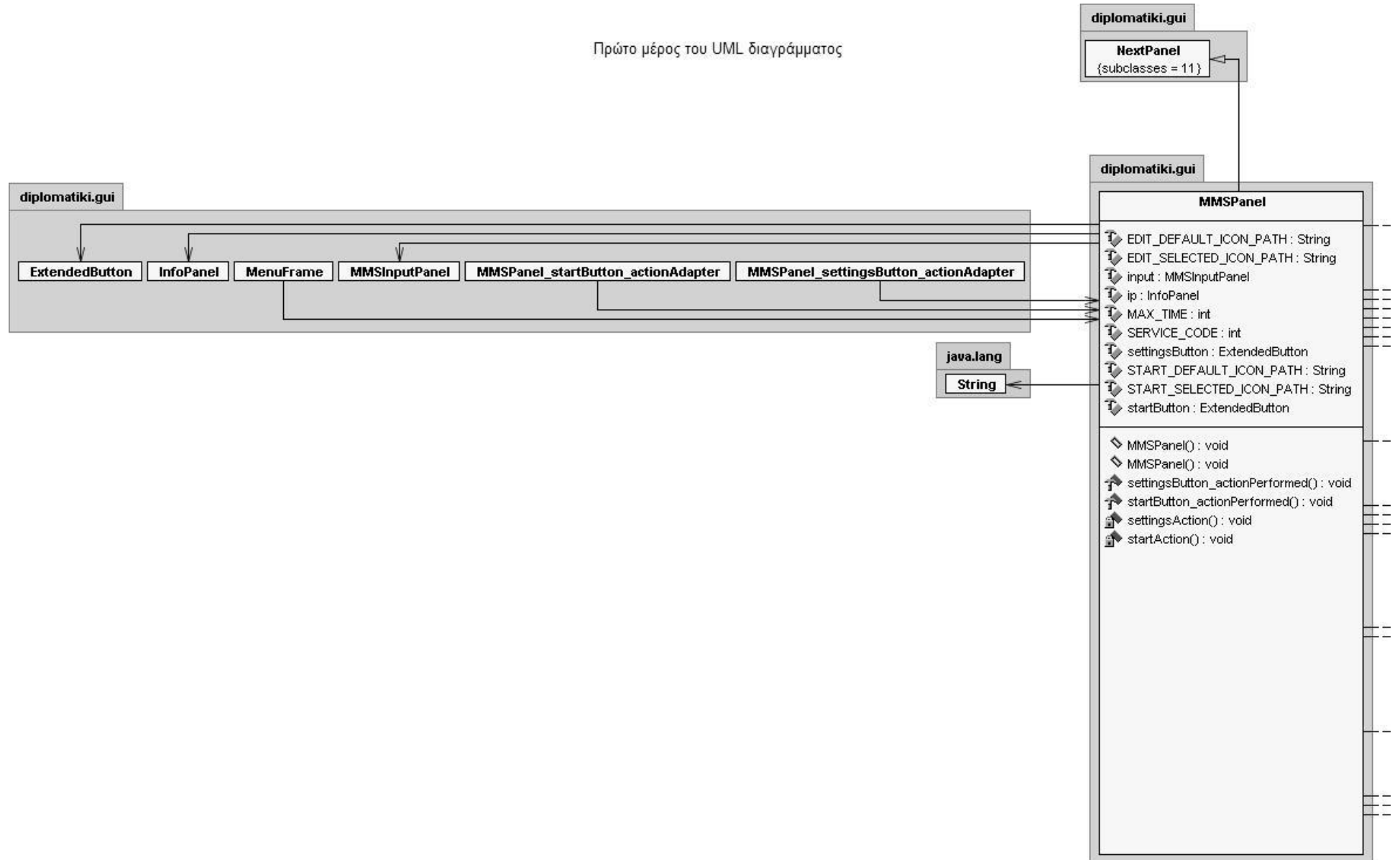
170     public void actionPerformed (ActionEvent e) {
        adaptee.startButton_actionPerformed (e);
    }
}

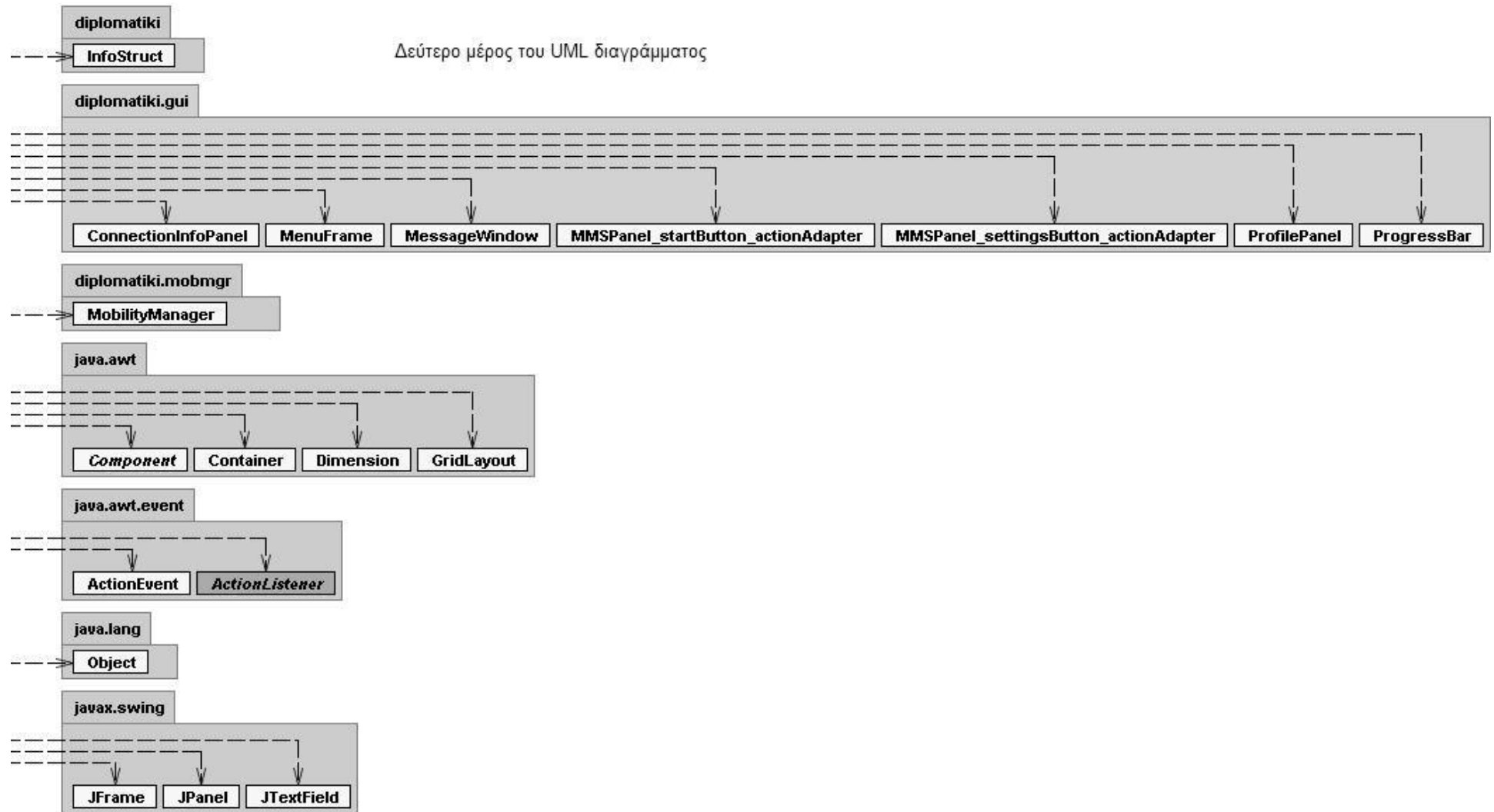
175 class MMSPanel_settingsButton_actionAdapter
    implements java.awt.event.ActionListener {
    MMSPanel adaptee;

180     MMSPanel_settingsButton_actionAdapter (MMSPanel adaptee) {
        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
185         adaptee.settingsButton_actionPerformed (e);
    }
}
```

Πρώτο μέρος του UML διαγράμματος





```

package diplomatiki .gui;

import java.awt.*;
import java.awt.event.*;
5 import javax.swing.*;
import javax.swing.border.*;
import java.util.*;
import diplomatiki .Config;

10 /**
 * <p>Title: Term Paper - <code>NextPanel</code> class</p>
 *
 * <p>Description: Most panels inherit from this class. Panels of this class
 * have specific characteristics (e.g. color, sections, etc). They have a north
15 * panel, a center panel and a south panel. In the south panel, there is a
 * "Back" button, which allows the user to return to the previous panel.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
20 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
25 public class NextPanel
    extends JPanel {
    JFrame myFrame;
    JPanel previousPanel;

30    JPanel centerPanel = new JPanel ();
    JPanel southPanel = new JPanel ();
    JPanel northPanel = new JPanel ();

    BorderLayout borderLayout1 = new BorderLayout ();
35    BorderLayout borderLayoutSouth = new BorderLayout ();
    GridLayout gridLayout1 = new GridLayout ();
    FlowLayout flowLayout1 = new FlowLayout ();

    ExtendedButton backButton =
40        new ExtendedButton (Config.BACK_BUTTON_DEFAULT_ICON_PATH ,
                            Config.BACK_BUTTON_SELECTED_ICON_PATH );

    /**
     * Creates a new <code>NextPanel</code> instance.
45     */
    public NextPanel () {
    }

    /**
     * Creates a new <code>NextPanel</code> instance with the specified
     * parent frame and previous panel.
     *
     * @param myFrame The frame which this panel belongs to
     * @param previousPanel The previous panel that was visible in the
55     * <code>myFrame</code> frame. This reference is very useful if the user
     * wants to return to the previous panel
     */
    public NextPanel (JFrame myFrame , JPanel previousPanel ) {
        super ();
60        this.myFrame = myFrame;
        this.previousPanel = previousPanel;
        try {
            jbInit ();
        }
65        catch (Exception ex) {
            ex.printStackTrace ();
        }
    }
70    /**

```

```

    * Builds the panel.
    *
    * @throws Exception An exception is thrown if the construction of the panel
    *   is not successful
75  */
    void jbInit() throws Exception {
        //Define the NextPanel's layout
        this.setLayout (borderLayout1);

80    //Define the layout for each sub-panel
        northPanel.setLayout (gridLayout1);
        centerPanel.setLayout (flowLayout1);
        southPanel.setLayout (borderLayoutSouth);

85    //Set the northPanel's properties
        northPanel.setBackground (UIManager.getColor (
            "InternalFrame.activeTitleGradient" ));
        northPanel.setBorder (
90        new TitledBorder (new EtchedBorder (), "Actions",
            TitledBorder.DEFAULT_JUSTIFICATION ,
            TitledBorder.DEFAULT_POSITION ,
            new java.awt.Font ("Microsoft Sans Serif" , 0, 11),
            new Color (10, 92, 174));
        northPanel.setMinimumSize (new Dimension (400, 42));
95        northPanel.setMaximumSize (new Dimension (400, 100));
        northPanel.setPreferredSize (new Dimension (400, 100));

        //Set the centerPanel's properties
        centerPanel.setBackground (UIManager.getColor (
100        "InternalFrame.activeTitleGradient" ));
        centerPanel.setBorder (
            new TitledBorder (new EtchedBorder (), "Input",
105            TitledBorder.DEFAULT_JUSTIFICATION ,
            TitledBorder.DEFAULT_POSITION ,
            new java.awt.Font ("Microsoft Sans Serif" , 0, 11),
            new Color (10, 92, 174));

        //Set the southPanel's properties
        southPanel.setBackground (UIManager.getColor (
110        "InternalFrame.activeTitleGradient" ));
        southPanel.setPreferredSize (new Dimension (400, 40));

        //Set the backButton's properties
        //The following is only used in case of a null layout for the southPanel
115        //backButton.setBounds(new Rectangle(307, 3, 77, 31));
        backButton.setMaximumSize (new Dimension (74, 35));
        backButton.setMinimumSize (new Dimension (74, 35));
        backButton.setPreferredSize (new Dimension (74, 35));
        backButton.addActionListener (new NextPanel_backButton_actionAdapter (this));
120

        //Add the backButton to the southPanel
        southPanel.add (backButton, BorderLayout.LINE_END);

        //Set the properties of NextPanel
125        this.setBackground (UIManager.getColor ("InternalFrame.activeTitleGradient" ));
        this.setMinimumSize (new Dimension (400, 290));
        this.setPreferredSize (new Dimension (400, 290));

        //Add the sub-panels to the appropriate positions
130        this.add (northPanel, BorderLayout.NORTH);
        this.add (centerPanel, BorderLayout.CENTER);
        this.add (southPanel, BorderLayout.SOUTH);

        //Define the keyboard keys used for navigation
135        Set mySet = new HashSet ();
        mySet.add (AWTKeyStroke.getAWTKeyStroke (KeyEvent.VK_DOWN, 0, false));
        mySet.add (AWTKeyStroke.getAWTKeyStroke (KeyEvent.VK_TAB, 0, false));
        this.setFocusTraversalKeys (KeyboardFocusManager.FORWARD_TRAVERSAL_KEYS ,
            mySet);
140        mySet.clear ();

```



```
mySet.add(AWTKeyStroke.getAWTKeyStroke(KeyEvent.VK_UP, 0, false));
mySet.add(AWTKeyStroke.getAWTKeyStroke(KeyEvent.VK_TAB,
    java.awt.event.InputEvent.
    SHIFT_DOWN_MASK, false));
145     this.setFocusTraversalKeys (KeyboardFocusManager.BACKWARD_TRAVERSAL_KEYS ,
        mySet);
        mySet.clear ();
    }

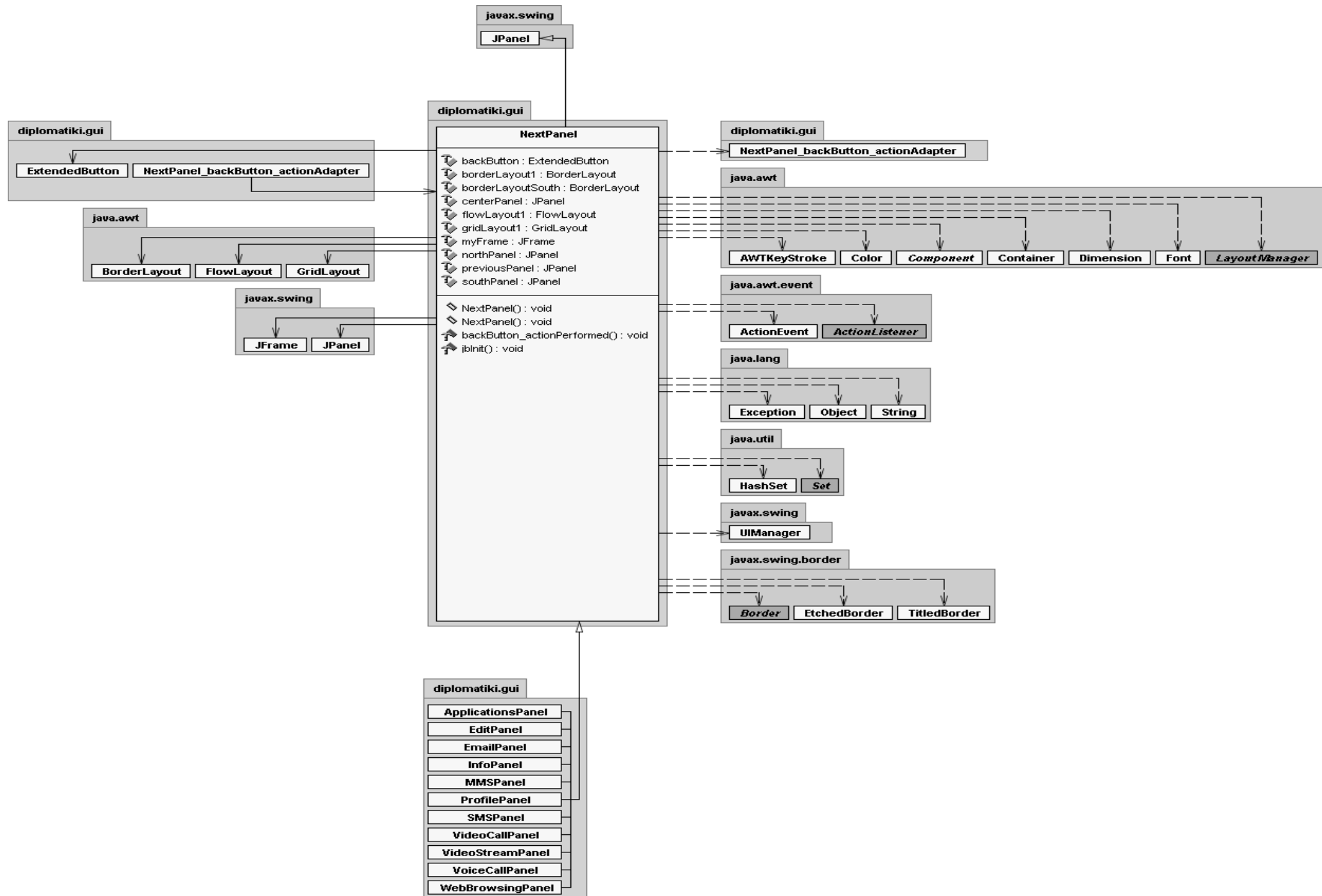
150     /**
    * Action performed when the "Back" button is pressed. The current panel is
    * hidden and the previous panel is shown.
    *
    * @param e ActionEvent
155     */
    void backButton_actionPerformed (ActionEvent e) {
        this.setVisible (false);
        myFrame.setContentPane (previousPanel);
        myFrame.setVisible (false);
160     myFrame.setVisible (true);
        previousPanel.setVisible (true);
    }

}

165     class NextPanel_backButton_actionAdapter
        implements java.awt.event.ActionListener {
            NextPanel adaptee;

170     NextPanel_backButton_actionAdapter (NextPanel adaptee) {
                this.adaptee = adaptee;
            }

            public void actionPerformed (ActionEvent e) {
175     adaptee.backButton_actionPerformed (e);
            }
        }
    }
```



```

package diplomatiki.gui;

import javax.swing.UIManager;
import java.awt.*;

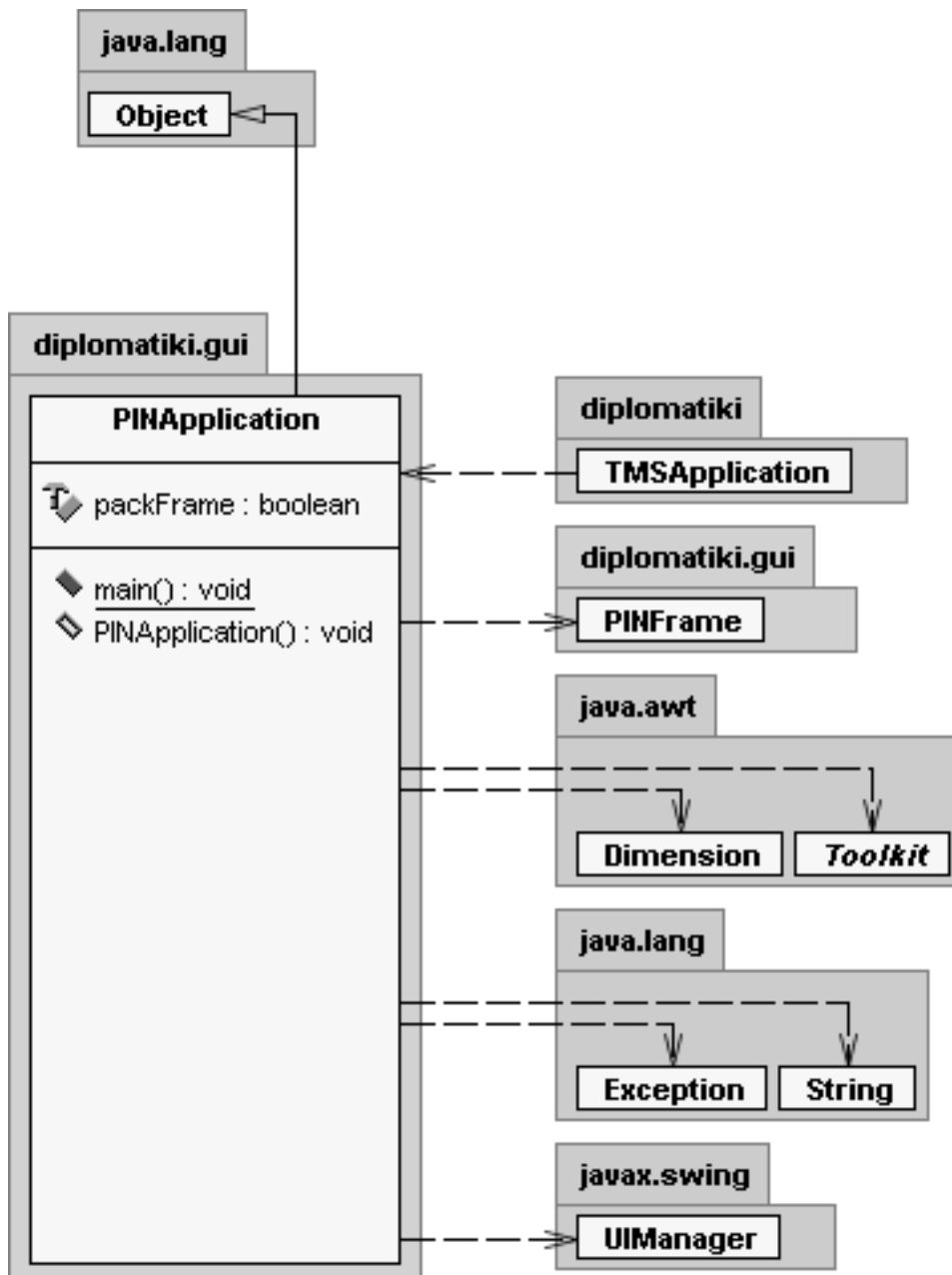
5
/**
 * <p>Title: Term Paper - <code>PINApplication</code> class</p>
 *
 * <p>Description: An instance of this class constructs and centers on screen a
10 * <code>PINFrame</code> instance.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
15 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
public class PINApplication {
20     boolean packFrame = false;

    //Construct the application
    /**
     * Apart from creating a new <code>PINApplication</code> instance, this
25 * constructor creates a new <code>PINFrame</code> instance and centers it on
     * screen.
     */
    public PINApplication () {
30         PINFrame frame = new PINFrame ();
        //Validate frames that have preset sizes
        //Pack frames that have useful preferred size info, e.g. from their layout
        if (packFrame) {
            frame.pack();
        }
35         else {
            frame.validate ();
        }
        //Center the window
        Dimension screenSize = Toolkit.getDefaultToolkit ().getScreenSize ();
40         Dimension frameSize = frame.getSize ();
        if (frameSize.height > screenSize.height) {
            frameSize.height = screenSize.height;
        }
45         if (frameSize.width > screenSize.width) {
            frameSize.width = screenSize.width;
        }
        frame.setLocation ( (screenSize.width - frameSize.width) / 2,
                             (screenSize.height - frameSize.height) / 2);
        frame.setVisible (true);
50     }

    //Main method
    /**
     * This method is useful in case one wants to construct and launch directly
55 * the <code>PINApplication</code>.
     *
     * @param args No command line arguments are necessary
     */
    public static void main(String[] args) {
60         try {
            UIManager.setLookAndFeel (UIManager.getSystemLookAndFeelClassName ());
        }
        catch (Exception e) {
            e.printStackTrace ();
65         }
        new PINApplication ();
    }
}

70

```



```

package diplomatiki .gui;

import java.io.*;
import java.util.*;
5
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

10 import diplomatiki .*;

/**
 * <p>Title: Term Paper - <code>PINFrame</code> class</p>
 *
15 * <p>Description: This is the frame that allows the user to login. The user
 * fills in their name and password. The password can also be changed.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
20 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
25 public class PINFrame
    extends JFrame {
    JPanel contentPane ;

    JLabel usernameLabel = new JLabel ();
    JLabel passwordLabel = new JLabel ();
    JLabel newPasswordLabel = new JLabel ();
    JLabel retypeNewPasswordLabel = new JLabel ();

    JTextField usernameField = new JTextField ();
35
    JPasswordField passwordField = new JPasswordField ();
    JPasswordField newPasswordField = new JPasswordField ();
    JPasswordField retypeNewPasswordField = new JPasswordField ();

    JCheckBox changeCheckBox = new JCheckBox ();

    ExtendedButton okButton = new ExtendedButton (Config.OK_DEFAULT_ICON_PATH ,
                                                    Config.OK_SELECTED_ICON_PATH );
    ExtendedButton cancelButton = new ExtendedButton (Config.
45     CANCEL_DEFAULT_ICON_PATH , Config.CANCEL_SELECTED_ICON_PATH );

    boolean visible = false;
    Properties pswConfig = loadPropertiesFile (Config.PSW_FILE);

50 //Construct the frame
/**
 * Creates a new <code>PINFrame</code> instance.
 */
    public PINFrame () {
55     enableEvents (AWTEvent.WINDOW_EVENT_MASK );
        try {
            jbInit ();
        }
        catch (Exception e) {
60             e.printStackTrace ();
        }
    }

    //Component initialization
65 /**
 * Builds the frame.
 *
 * @throws Exception An exception is thrown if the construction of the frame
 * is not successful
70 */

```

```

private void jbInit() throws Exception {
    contentPane = (JPanel) this.getContentPane ();
    contentPane.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
75    contentPane.setLayout (null);

    this.getContentPane ().setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
80    this.setSize (new Dimension (400, 300));
    this.setTitle ("Please Log In");

    usernameLabel.setFont (new java.awt.Font ("Dialog", 1, 11));
    usernameLabel.setText ("Username");
    usernameLabel.setBounds (new Rectangle (50, 40, 60, 15));
85

    usernameField.setText ("");
    usernameField.setBounds (new Rectangle (130, 37, 120, 23));

    passwordLabel.setFont (new java.awt.Font ("Dialog", 1, 11));
90    passwordLabel.setText ("Password");
    passwordLabel.setBounds (new Rectangle (50, 67, 60, 15));

    passwordField.setText ("");
    passwordField.setBounds (new Rectangle (130, 64, 120, 23));
95

    changeCheckBox.setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    changeCheckBox.setText ("I also want to change my password" );
    changeCheckBox.setBounds (new Rectangle (50, 110, 210, 23));
100    changeCheckBox .
        addActionListener (new PINFrame_changeCheckBox_actionAdapter (this));

    newPasswordLabel.setEnabled (false);
    newPasswordLabel.setVisible (false);
105    newPasswordLabel.setFont (new java.awt.Font ("Dialog", 1, 11));
    newPasswordLabel.setText ("New Password");
    newPasswordLabel.setBounds (new Rectangle (50, 160, 95, 15));

    newPasswordField.setEnabled (false);
110    newPasswordField.setVisible (false);
    newPasswordField.setText ("");
    newPasswordField.setBounds (new Rectangle (180, 157, 110, 23));

    retypeNewPasswordLabel.setEnabled (false);
115    retypeNewPasswordLabel.setVisible (false);
    retypeNewPasswordLabel.setFont (new java.awt.Font ("Dialog", 1, 11));
    retypeNewPasswordLabel.setText ("Retype New Password" );
    retypeNewPasswordLabel.setBounds (new Rectangle (50, 187, 128, 15));

    retypeNewPasswordField.setEnabled (false);
120    retypeNewPasswordField.setVisible (false);
    retypeNewPasswordField.setBounds (new Rectangle (180, 184, 110, 23));

    okButton.setBounds (new Rectangle (0, 240, 66, 30));
125    okButton.setText ("");
    okButton.addActionListener (new PINFrame_okButton_actionAdapter (this));

    cancelButton.setBounds (new Rectangle (288, 240, 100, 25));
    cancelButton.setText ("");
130    cancelButton .
        addActionListener (new PINFrame_cancelButton_actionAdapter (this));

    Set mySet = new HashSet ();
    mySet.add (AWTKeyStroke.getAWTKeyStroke (KeyEvent.VK_DOWN, 0, false));
135    mySet.add (AWTKeyStroke.getAWTKeyStroke (KeyEvent.VK_TAB, 0, false));
    this.setFocusTraversalKeys (KeyboardFocusManager.FORWARD_TRAVERSAL_KEYS ,
        mySet);
    mySet.clear ();
    mySet.add (AWTKeyStroke.getAWTKeyStroke (KeyEvent.VK_UP, 0, false));
140    mySet.add (AWTKeyStroke.getAWTKeyStroke (KeyEvent.VK_TAB,

```

```

        java.awt.event.InputEvent .
            SHIFT_DOWN_MASK , false));
    this.setFocusTraversalKeys (KeyboardFocusManager .BACKWARD_TRAVERSAL_KEYS ,
        mySet);
145    mySet.clear ();

        contentPane .add(usernameLabel , null);
        contentPane .add(usernameField , null);
        contentPane .add(retypeNewPasswordLabel , null);
150    contentPane .add(retypeNewPasswordField , null);
        contentPane .add(newPasswordField , null);
        contentPane .add(newPasswordLabel , null);
        contentPane .add(changeCheckBox , null);
        contentPane .add(passwordLabel , null);
155    contentPane .add(passwordField , null);
        contentPane .add(okButton , null);
        contentPane .add(cancelButton , null);
    }

160    //Overridden so we can exit when window is closed
    /**
     * This method is overridden so we can exit when the window is closed.
     *
     * @param e WindowEvent
    165    */
    protected void processWindowEvent (WindowEvent e) {
        super.processWindowEvent (e);
        if (e.getID () == WindowEvent .WINDOW_CLOSING ) {
170            System.exit (0);
        }
    }

    /**
     * Loads a Properties file.
    175    *
     * @param filePath The path of the properties file
     * @return A <code>Properties</code> instance that represents the properties
     *         file's contents
     */
180    private Properties loadPropertiesFile (String filePath) {
        Properties config = new Properties ();
        InputStream input = null;
        try {
185            input = new FileInputStream (filePath);
            config.load(input);
        }
        catch (IOException ex) {
            System.err.println ("Cannot open and load properties file." );
        }
190    finally {
        try {
            if (input != null) {
                input.close ();
195            }
        }
        catch (IOException ex) {
            System.err.println ("Cannot close properties file." );
        }
    }

200    return config;
}

    /**
    205    * Writes the property list (key and element pairs) in the <code>config
     * Properties</code> table to the file specified.
     *
     * @param config A <code>Properties</code> object
     * @param filePath The path of the file in which the property list should be
    210    *         stored

```

```

    */
    private void storePropertiesFile (Properties config, String filePath) {
        OutputStream output = null;
        try {
215         output = new FileOutputStream (filePath);
            config.store (output, null);
        }
        catch (IOException ex) {
220         System.err.println ("Cannot open and store properties file." );
        }
        finally {
            try {
                if (output != null) {
225                 output.close ();
                }
            }
            catch (IOException ex) {
                System.err.println ("Cannot close properties file." );
            }
230        }
    }

    /**
235     * Action performed when the state of the panel's check-box is changed. The
     * fields where the user should type the new password are either hidden or
     * shown.
     *
     * @param e ActionEvent
     */
240    void changeCheckBox_actionPerformed (ActionEvent e) {
        //System.out.println ("BOX STATUS CHANGED");

        visible = !visible;

245        newPasswordLabel.setEnabled (visible);
        retypeNewPasswordLabel.setEnabled (visible);
        newPasswordField.setEnabled (visible);
        retypeNewPasswordField.setEnabled (visible);

250        newPasswordLabel.setVisible (visible);
        retypeNewPasswordLabel.setVisible (visible);
        newPasswordField.setVisible (visible);
        retypeNewPasswordField.setVisible (visible);
    }
255    /**
     * Action performed when the "OK" button is pressed.
     *
     * @param e ActionEvent
260     */
    void okButton_actionPerformed (ActionEvent e) {
        if (!correct ()) {
            return;
265        }

        if (!checkNewPassword ()) {
            return;
        }

270        launchMenuApplication ();
    }

    /**
275     * Returns <code>>true</code> if the username/password typed is correct.
     *
     * @return <code>True</code> if the username/password typed is correct
     */
    private boolean correct () {
280        String usernameFilled = usernameField.getText ();
        String passwordFilled = new String (passwordField.getPassword ());
    }

```



```

        if ( (pswConfig.getProperty (usernameFilled) == null) ||
            (!pswConfig.getProperty (usernameFilled).equals (passwordFilled))) {
285         new MessageWindow (this, "Wrong Username/Password" ,
                                "Wrong username/password" );
            clear ();
            return false;
        }

290     Config.init (usernameFilled);

        return true;
    }

295     /**
     * Clears all the text and password fields.
     */
    private void clear () {
300         usernameField.setText ("");
        passwordField.setText ("");
        newPasswordField.setText ("");
        retypeNewPasswordField.setText ("");
    }

305     /**
     * Clears only the optional password fields.
     */
    private void clearNew () {
310         newPasswordField.setText ("");
        retypeNewPasswordField.setText ("");
    }

    /**
315     * Checks if the user wants to change their password. Also, checks if the new
     * password has been retyped correctly.
     *
     * @return <code>True</code> if the new password has been retyped correctly
     *         or if the user doesn't want to change their password.
     */
320     private boolean checkNewPassword () {
        if (changeCheckBox.isSelected ()) {
            String pass1 = new String (newPasswordField.getPassword ());
            String pass2 = new String (retypeNewPasswordField.getPassword ());
            if (pass1.equals (pass2)) {
325                 pswConfig.setProperty (usernameField.getText (), pass1);
                storePropertiesFile (pswConfig, Config.PSW_FILE);
                return true;
            }
            else {
330                 new MessageWindow (this, "Retyped password not correct" ,
                                        "New password not valid" );
                    clearNew ();
                    return false;
                }
            }
335     }

        return true;
    }

340     /**
     * Launches the <code>MenuApplication</code>.
     */
    public void launchMenuApplication () {
345         //This is the appropriate place to optionally show an animation

        this.setVisible (false);
        this.dispose ();

        new MenuApplication ();
350     }

```

```
    /**
     * Action performed when the "Cancel" button is pressed. The program is
     * terminated.
355     *
     * @param e ActionEvent
     */
    void cancelButton_actionPerformed (ActionEvent e) {
360         System.exit(0);
    }

}

class PINFrame_changeCheckBox_actionAdapter
365     implements java.awt.event.ActionListener {
    PINFrame adaptee;

    PINFrame_changeCheckBox_actionAdapter (PINFrame adaptee) {
370         this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.changeCheckBox_actionPerformed (e);
    }
375 }

class PINFrame_okButton_actionAdapter
    implements java.awt.event.ActionListener {
380     PINFrame adaptee;

    PINFrame_okButton_actionAdapter (PINFrame adaptee) {
        this.adaptee = adaptee;
    }

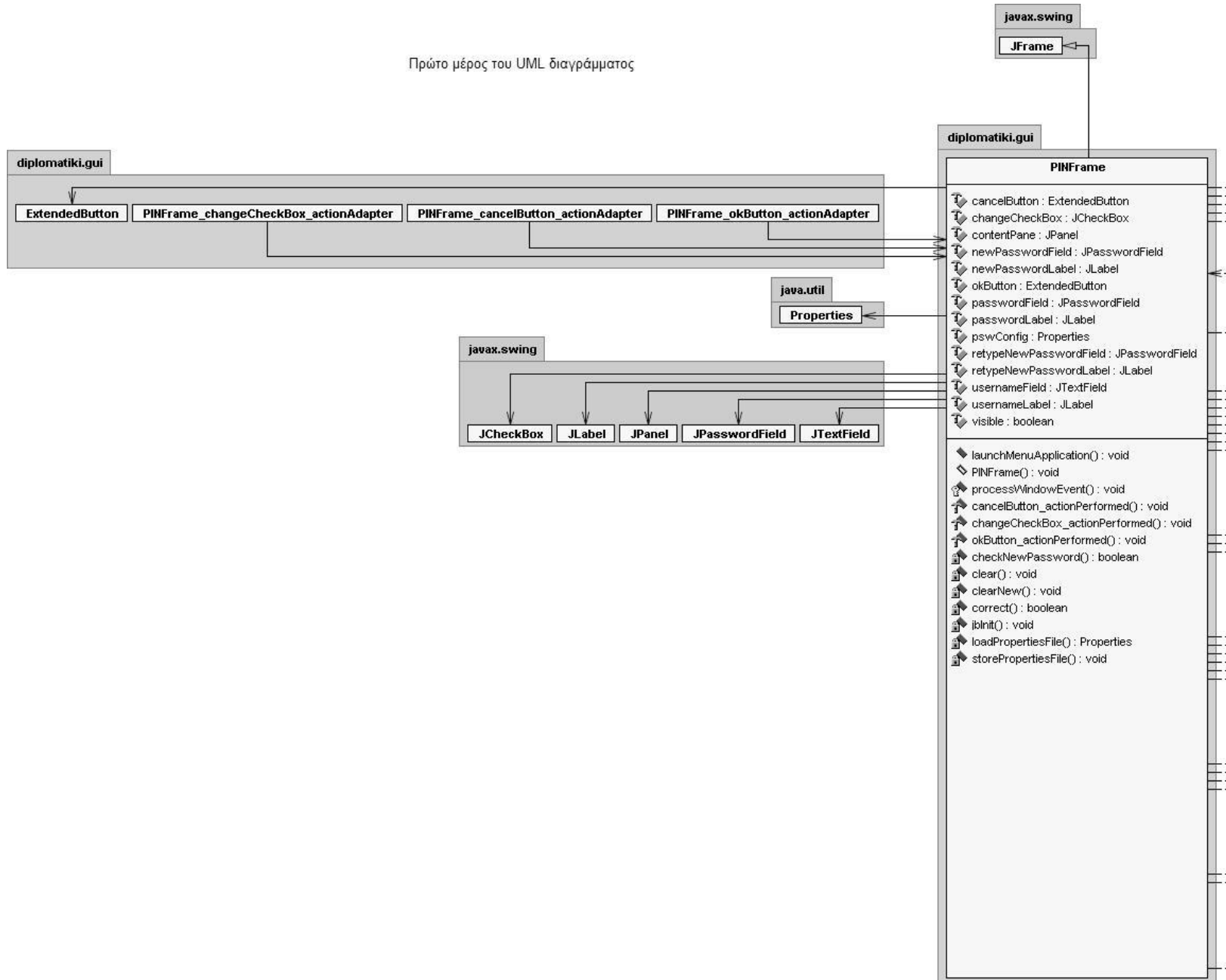
385     public void actionPerformed (ActionEvent e) {
        adaptee.okButton_actionPerformed (e);
    }
}

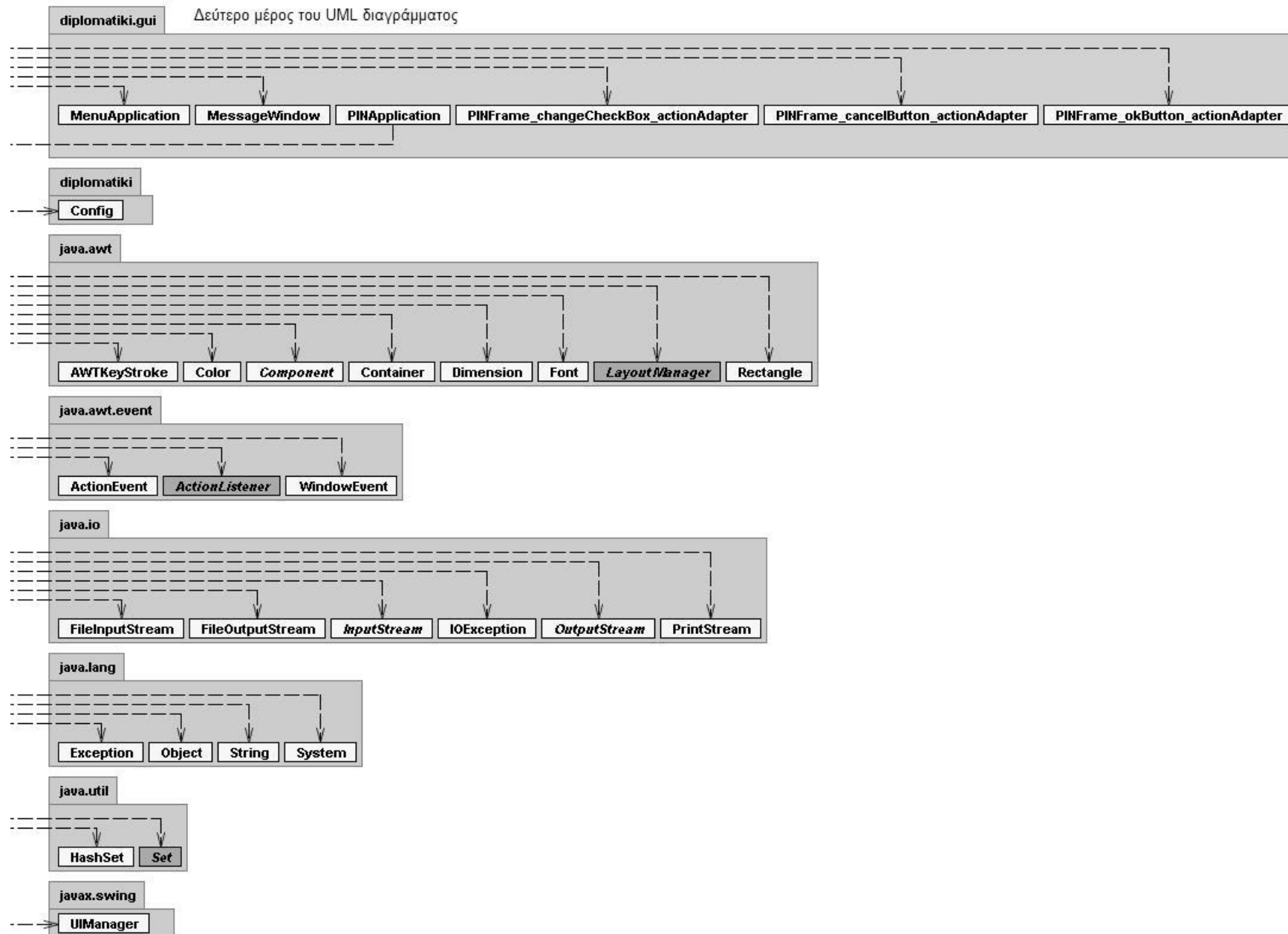
390 class PINFrame_cancelButton_actionAdapter
    implements java.awt.event.ActionListener {
    PINFrame adaptee;

    PINFrame_cancelButton_actionAdapter (PINFrame adaptee) {
395         this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.cancelButton_actionPerformed (e);
400     }
}
}
```

Πρώτο μέρος του UML διαγράμματος





```

package diplomatiki .gui;

import java.awt.*;
import java.awt.event.*;
5 import javax.swing.*;
import javax.swing.border.*;
import java.io.*;
import java.util.*;
import diplomatiki .Config;
10
/**
 * <p>Title: Term Paper - <code>ProfilePanel</code> class</p>
 *
 * <p>Description: This panel shows the user settings. It also allows the user
15 * to edit these settings. A panel of this type contains a
 * <code>GeneralProfileSettingsPanel</code> in its north section.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
20 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
25 public class ProfilePanel
    extends JPanel {
    final String CURRENT_PROFILE_PATH = Config.PROFILE_PATH_CURRENT ;
    final String LOW_COST_PROFILE_PATH = Config.PROFILE_PATH_LOW_COST ;
    final String HIGH_QUALITY_PROFILE_PATH = Config.PROFILE_PATH_HIGH_QUALITY ;
30 final String BALANCED_PROFILE_PATH = Config.PROFILE_PATH_BALANCED ;

    final String PROVIDERS_FILE = Config.PROVIDERS_FILE ;
    final String PROVIDERS_INVERSED_FILE = Config.PROVIDERS_INVERSED_FILE ;

35 final String ACTIVATE_DEFAULT_ICON_PATH = Config.ACTIVATE_DEFAULT_ICON_PATH ;
    final String ACTIVATE_SELECTED_ICON_PATH = Config.ACTIVATE_SELECTED_ICON_PATH ;
    final String STORE_DEFAULT_ICON_PATH = Config.STORE_DEFAULT_ICON_PATH ;
    final String STORE_SELECTED_ICON_PATH = Config.STORE_SELECTED_ICON_PATH ;

40 String[] profilePath = {
    null, CURRENT_PROFILE_PATH , LOW_COST_PROFILE_PATH ,
    HIGH_QUALITY_PROFILE_PATH , BALANCED_PROFILE_PATH } ;
    int profileCode ;

45 GeneralProfileSettingsPanel gps;
    JLabel
        voiceCallLabel = new JLabel ("Voice Call:"),
        mmsLabel = new JLabel ("MMS:"),
        videoCallLabel = new JLabel ("Video Call:"),
50 videoStreamLabel = new JLabel ("Video Stream:"),
        webBrowsingLabel = new JLabel ("Web Browsing:");
    ExtendedSlider voiceCallSlider , mmsSlider , videoCallSlider , videoStreamSlider ,
        webBrowsingSlider ;
    ExtendedSlider [] sliders = new ExtendedSlider [5];
55 JScrollPane centerScroll = new JScrollPane ();
    JPanel centerViewPort = new JPanel ();
    ExtendedButton activateButton , storeButton ;

    /**
60 * Creates a new <code>ProfilePanel</code> instance.
    */
    public ProfilePanel () {
    }

65 /**
    * Creates a new <code>ProfilePanel</code> instance with the specified parent
    * frame and previous panel.
    *
    * @param myFrame The frame which this panel belongs to
70 * @param previousPanel The previous panel that was visible in the

```

```

* <code>myFrame</code> frame. This reference is very useful if the user
* wants to return to the previous panel
* @param profileCode The unique profile code that defines which of the
* predefined profiles (e.g. the "Current" profile, the "Low Cost" profile,
75 * etc.) has been selected and should be shown on screen
*/
public ProfilePanel (JFrame myFrame, JPanel previousPanel, int profileCode) {
    super (myFrame, previousPanel);

80    //Store the profile code
    this.profileCode = profileCode;

    //Reset traversal keys to default
    this.setFocusTraversalKeys (KeyboardFocusManager .FORWARD_TRAVERSAL_KEYS ,
85     null);
    this.setFocusTraversalKeys (KeyboardFocusManager .BACKWARD_TRAVERSAL_KEYS ,
     null);

    //Set the border of each sub-panel
90    northPanel.setBorder (
        new TitledBorder (new EtchedBorder (), "General Settings" ,
            TitledBorder .DEFAULT_JUSTIFICATION ,
            TitledBorder .DEFAULT_POSITION ,
            new java.awt.Font ("Microsoft Sans Serif" , 0, 11),
95     new Color (10, 92, 174));
    centerPanel.setBorder (
        new TitledBorder (new EtchedBorder (), "Choose QoS Levels" ,
            TitledBorder .DEFAULT_JUSTIFICATION ,
            TitledBorder .DEFAULT_POSITION ,
100     new java.awt.Font ("Microsoft Sans Serif" , 0, 11),
            new Color (10, 92, 174));

    //Load the profile properties file
    Properties profileConfig = loadPropertiesFile (profilePath [profileCode]);
105

    //Find the 3 main factors
    String f1 = profileConfig.getProperty ("factor1");
    String f2 = profileConfig.getProperty ("factor2");
    String f3 = profileConfig.getProperty ("factor3");
110    String providerName = profileConfig.getProperty ("providerName");
    int providerCode = new Integer (profileConfig.getProperty ("providerCode")).
        intValue ();
    int otherProviderCode = 0;
    if (providerName.equals ("other")) {
115     otherProviderCode = providerCode;

        //Load the providers properties file
    }
    Properties providerConfig = loadPropertiesFile (PROVIDERS_FILE);
120

    //Find all known providers and also the preferred one
    String[] providersArray = findAllKnownProviders (providerConfig);
    int selectedIndex = findSelectedIndex (providersArray, providerName);

125    //Ready to create GeneralProfileSettingsPanel and add it to the northPanel
    gps = new GeneralProfileSettingsPanel (f1, f2, f3, providersArray,
        selectedIndex, otherProviderCode);
    northPanel.add (gps);

130    //Set the properties of the centerScroll panel
    centerScroll.getViewport ().setBackground (UIManager.getColor (
        "InternalFrame.activeTitleGradient" ));
    centerScroll.setBorder (null);
    centerScroll.setMinimumSize (new Dimension (250, 100));
135    centerScroll.setMaximumSize (new Dimension (250, 100));
    centerScroll.setPreferredSize (new Dimension (250, 100));

    //Set the properties of the centerViewPort panel
    centerViewPort.setBackground (UIManager.getColor (
140     "InternalFrame.activeTitleGradient" ));

```

```

        centerViewPort .setLayout (new GridLayout (5, 2));

        //Set the font for the centerPanel's labels
        voiceCallLabel .setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
145    mmsLabel .setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
        videoCallLabel .setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
        videoStreamLabel .setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
        webBrowsingLabel .setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));

150    //Create the sliders
        voiceCallSlider = new ExtendedSlider (5, "1");
        mmsSlider = new ExtendedSlider (2, "3");
        videoCallSlider = new ExtendedSlider (5, "5");
        videoStreamSlider = new ExtendedSlider (5, "6");
155    webBrowsingSlider = new ExtendedSlider (5, "7");

        //Additionally, store them to an array
        sliders [0] = voiceCallSlider ;
        sliders [1] = mmsSlider ;
160    sliders [2] = videoCallSlider ;
        sliders [3] = videoStreamSlider ;
        sliders [4] = webBrowsingSlider ;

        //Find (from the profile properties file) the (maximum) preferred QoS levels
165    getDefaultSliderValues (profileConfig);

        //Add a focus listener to each slider
        voiceCallSlider .
            addFocusListener (new ProfilePanel_voiceCallSlider_focusAdapter (this));
170    mmsSlider .
            addFocusListener (new ProfilePanel_mmsSlider_focusAdapter (this));
        videoCallSlider .
            addFocusListener (new ProfilePanel_videoCallSlider_focusAdapter (this));
        videoStreamSlider .
175    addFocusListener (new ProfilePanel_videoStreamSlider_focusAdapter (this));
        webBrowsingSlider .
            addFocusListener (new ProfilePanel_webBrowsingSlider_focusAdapter (this));

        //Add the labels and sliders to the centerViewPort
180    centerViewPort .add (voiceCallLabel );
        centerViewPort .add (voiceCallSlider );
        centerViewPort .add (mmsLabel );
        centerViewPort .add (mmsSlider );
        centerViewPort .add (videoCallLabel );
185    centerViewPort .add (videoCallSlider );
        centerViewPort .add (videoStreamLabel );
        centerViewPort .add (videoStreamSlider );
        centerViewPort .add (webBrowsingLabel );
        centerViewPort .add (webBrowsingSlider );

190    //Add the centerViewPort to the centerScroll panel
        //and the centerScroll to the centerPanel
        centerScroll .getViewport ().add (centerViewPort );
        centerPanel .add (centerScroll );

195    //Create two new buttons, activateButton and storeButton
        activateButton = new ExtendedButton (ACTIVATE_DEFAULT_ICON_PATH ,
            ACTIVATE_SELECTED_ICON_PATH );
        storeButton = new ExtendedButton (STORE_DEFAULT_ICON_PATH ,
200    STORE_SELECTED_ICON_PATH );

        activateButton .
            addActionListener (new ProfilePanel_activateButton_actionAdapter (this));
        storeButton .
205    addActionListener (new ProfilePanel_storeButton_actionAdapter (this));

        //Add them to the southPanel
        southPanel .add (activateButton , BorderLayout .LINE_START);
        southPanel .add (storeButton , BorderLayout .CENTER);
210    }

```

```
/**
 * Loads a Properties file.
 *
215 * @param filePath The path of the properties file
 * @return A <code>Properties</code> instance that represents the properties
 *       file's contents
 */
private Properties loadPropertiesFile (String filePath) {
220 Properties config = new Properties ();
    InputStream input = null;
    try {
        input = new FileInputStream (filePath);
        config.load(input);
225    }
    catch (IOException ex) {
        System.err.println("Cannot open and load properties file." );
    }
    finally {
230        try {
            if (input != null) {
                input.close ();
            }
        }
        catch (IOException ex) {
235            System.err.println("Cannot close properties file." );
        }
    }

240    return config;
}

/**
 * Writes the property list (key and element pairs) in the <code>config
245 * Properties</code> table to the file specified.
 *
 * @param config A <code>Properties</code> object
 * @param filePath The path of the file in which the property list should be
 *       stored
250 */
private void storePropertiesFile (Properties config, String filePath) {
    OutputStream output = null;
    try {
        output = new FileOutputStream (filePath);
255        config.store(output, null);
    }
    catch (IOException ex) {
        System.err.println("Cannot open and store properties file." );
    }
    finally {
260        try {
            if (output != null) {
                output.close ();
            }
265        }
        catch (IOException ex) {
            System.err.println("Cannot close properties file." );
        }
    }
270 }

/**
 * Action performed when the user wants to save the changes made to their
275 * settings. It is invoked when the "Store" or the "Activate" button is
 * pressed.
 *
 * @param config A <code>Properties</code> object which encapsulates the
 *       property list that should be saved
 */
280 private void saveNewSettings (Properties config) {
```



```

config.setProperty ("factor1", (String) gps.dlm.getElementAt (0));
config.setProperty ("factor2", (String) gps.dlm.getElementAt (1));
config.setProperty ("factor3", (String) gps.dlm.getElementAt (2));

285 String selectedProviderName = (String) gps.providerBox.getSelectedItem ();
config.setProperty ("providerName", selectedProviderName );
if (selectedProviderName .equals ("other")) {
    config.setProperty ("providerCode", gps.otherProviderCodeField .getText ());
}
290 else if (selectedProviderName .equals ("none")) {
    config.setProperty ("providerCode", "-1");
}
else {
    Properties inversedProviderConfig = loadPropertiesFile (
295 PROVIDERS_INVERSED_FILE );
    config.
        setProperty ("providerCode",
                    inversedProviderConfig .getProperty (selectedProviderName ));
}
300
for (int i = 0; i < sliders.length; i++) {
    config.setProperty (sliders [i].getName (), "" + sliders [i].getValue ());
}
305 }

/**
 * Constructs a list with all known providers, so the user can select their
 * favorite one.
310 *
 * @param config The <code>Properties</code> object that contains the list
 * with all known providers
 * @return A table with all known providers
 */
315 private String [] findAllKnownProviders (Properties config) {
    Vector lst = new Vector ();
    Enumeration enum = config.propertyNames ();
    while (enum.hasMoreElements ()) {
        lst.addElement (config.getProperty ( (String) enum.nextElement ()));
320 }

    String [] allKnownProviders = new String [lst.size () + 2];
    allKnownProviders [0] = "none";
    for (int i = 0; i < lst.size (); i++) {
325 allKnownProviders [i + 1] = (String) lst.get (i);
    }
    allKnownProviders [lst.size () + 1] = "other";

    return allKnownProviders ;
330 }

/**
 * Finds the provider that the user prefers. It searches through the table
 * <code>myArray</code> until it finds a match with <code>name</code>, and
335 * then returns its position. If there is no match, this method returns
 * <code>0</code>. This is the index of the choice "none" (the user doesn't
 * have a favorite provider).
 *
 * @param myArray The table of known providers
340 * @param name The name of the favorite provider
 * @return The position of the favorite provider in the table
 */
private int findSelectedIndex (String [] myArray, String name) {
    for (int i = 0; i < myArray.length; i++) {
345 if (myArray [i].equals (name)) {
        return i;
    }
}

350 return 0;

```

```
    }

    /**
     * Finds the sliders' default values. The sliders' default values are their
355     * initial values. They can be found in the user profile.
     *
     * @param config The <code>Properties</code> that contains the default values
     */
    private void getDefaultSliderValues (Properties config) {
360     for (int i = 0; i < sliders.length; i++) {
        sliders[i].setDefaultValue (new Integer (config.getProperty (sliders[i].
            getName ())).intValue ());
    }
365 }

    /**
     * Sets the new value of the scroll bar. A scroll bar's value is reflected by
     * its position.
     *
370     * @param newValue The new value
     */
    private void setScrollBarValue (int newValue) {
        centerScroll.setVerticalScrollBar ().setValue (newValue);
375 }

    /**
     * Action performend when the Voice Call slider gains focus.
     *
     * @param e FocusEvent
380     */
    void voiceCallSlider_focusGained (FocusEvent e) {
        setScrollBarValue (centerScroll.setVerticalScrollBar ().getMinimum ());
    }

385 /**
     * Action performend when the MMS slider gains focus.
     *
     * @param e FocusEvent
     */
390 void mmsSlider_focusGained (FocusEvent e) {
        setScrollBarValue (centerScroll.setVerticalScrollBar ().getMinimum ());
    }

    /**
395     * Action performend when the Video Call slider gains focus.
     *
     * @param e FocusEvent
     */
400 void videoCallSlider_focusGained (FocusEvent e) {
        setScrollBarValue (45);
    }

    /**
405     * Action performend when the Video Stream slider gains focus.
     *
     * @param e FocusEvent
     */
    void videoStreamSlider_focusGained (FocusEvent e) {
410     setScrollBarValue (centerScroll.setVerticalScrollBar ().getMaximum ());
    }

    /**
     * Action performend when the Web Browsing slider gains focus.
     *
415     * @param e FocusEvent
     */
    void webBrowsingSlider_focusGained (FocusEvent e) {
        setScrollBarValue (centerScroll.setVerticalScrollBar ().getMaximum ());
420 }
    }
```

```

/**
 * Action performed when the "Activate" button is pressed.
 *
 * @param e ActionEvent
425 */
void activateButton_actionPerformed (ActionEvent e) {
    Properties outConfig = loadPropertiesFile (profilePath [1]);
    saveNewSettings (outConfig);
    storePropertiesFile (outConfig, profilePath [1]);
430 ( (MenuFrame) myFrame).getMobilityManager ().reExamineAllRunningServices ();
}

/**
 * Action performed when the "Store" button is pressed.
435 *
 * @param e ActionEvent
 */
void storeButton_actionPerformed (ActionEvent e) {
440     if (profileCode == 1) {
        activateButton_actionPerformed (e);
    }
    Properties outConfig = loadPropertiesFile (profilePath [profileCode]);
    saveNewSettings (outConfig);
    storePropertiesFile (outConfig, profilePath [profileCode]);
445 }

/**
 * Disables all sliders except the one specified by <code>sliderCode</code>.
 * Disabled sliders cannot be edited.
450 *
 * @param sliderCode The code of the slider that should remain enabled
 */
void disableAllSlidersExceptOne (int sliderCode) {
455     JLabel [] labels = new JLabel [5];
    labels [0] = voiceCallLabel ;
    labels [1] = mmsLabel ;
    labels [2] = videoCallLabel ;
    labels [3] = videoStreamLabel ;
    labels [4] = webBrowsingLabel ;
460
    for (int i = 0; i < sliders.length; i++) {
        labels [i].setEnabled (false);
        sliders [i].setEnabled (false);
    }
465     labels [sliderCode].setEnabled (true);
    sliders [sliderCode].setEnabled (true);
}

}
470
class ProfilePanel_voiceCallSlider_focusAdapter
    extends java.awt.event.FocusAdapter {
    ProfilePanel adaptee;
475
    ProfilePanel_voiceCallSlider_focusAdapter (ProfilePanel adaptee) {
        this.adaptee = adaptee;
    }

    public void focusGained (FocusEvent e) {
480         adaptee.voiceCallSlider_focusGained (e);
    }

    public void focusLost (FocusEvent e) {
    }
485 }

class ProfilePanel_mmsSlider_focusAdapter
    extends java.awt.event.FocusAdapter {
    ProfilePanel adaptee;
490

```

```
ProfilePanel_mmsSlider_focusAdapter (ProfilePanel adaptee) {
    this.adaptee = adaptee;
}

495 public void focusGained(FocusEvent e) {
    adaptee.mmsSlider_focusGained (e);
}

public void focusLost(FocusEvent e) {
500 }
}

class ProfilePanel_videoCallSlider_focusAdapter
    extends java.awt.event.FocusAdapter {
505 ProfilePanel adaptee;

ProfilePanel_videoCallSlider_focusAdapter (ProfilePanel adaptee) {
    this.adaptee = adaptee;
}

510 public void focusGained(FocusEvent e) {
    adaptee.videoCallSlider_focusGained (e);
}

515 public void focusLost(FocusEvent e) {
}
}

class ProfilePanel_videoStreamSlider_focusAdapter
    extends java.awt.event.FocusAdapter {
520 ProfilePanel adaptee;

ProfilePanel_videoStreamSlider_focusAdapter (ProfilePanel adaptee) {
525     this.adaptee = adaptee;
}

public void focusGained(FocusEvent e) {
    adaptee.videoStreamSlider_focusGained (e);
}

530 public void focusLost(FocusEvent e) {
}
}

535 class ProfilePanel_webBrowsingSlider_focusAdapter
    extends java.awt.event.FocusAdapter {
    ProfilePanel adaptee;

ProfilePanel_webBrowsingSlider_focusAdapter (ProfilePanel adaptee) {
540     this.adaptee = adaptee;
}

public void focusGained(FocusEvent e) {
    adaptee.webBrowsingSlider_focusGained (e);
545 }

public void focusLost(FocusEvent e) {
}
}

550 class ProfilePanel_activateButton_actionAdapter
    implements java.awt.event.ActionListener {
    ProfilePanel adaptee;

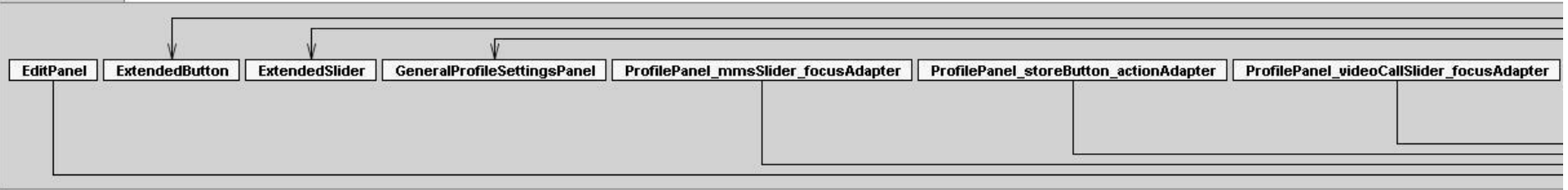
555 ProfilePanel_activateButton_actionAdapter (ProfilePanel adaptee) {
    this.adaptee = adaptee;
}

public void actionPerformed(ActionEvent e) {
560     adaptee.activateButton_actionPerformed (e);
}
```

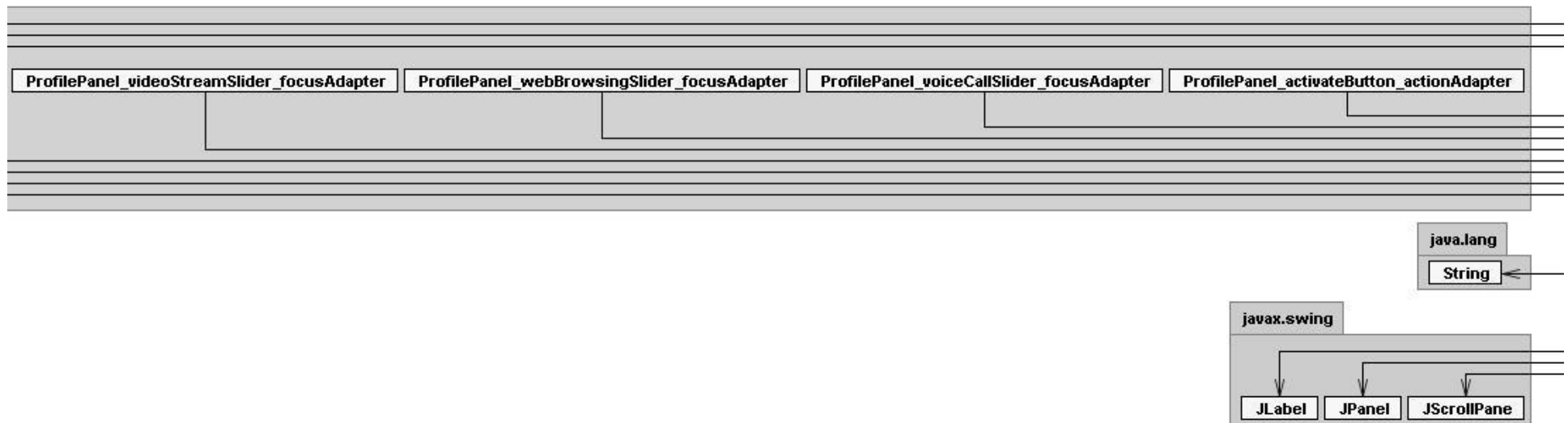
```
    }  
  }  
  
  class ProfilePanel_storeButton_actionAdapter  
565    implements java.awt.event.ActionListener {  
      ProfilePanel adaptee;  
  
      ProfilePanel_storeButton_actionAdapter (ProfilePanel adaptee) {  
570        this.adaptee = adaptee;  
  
        public void actionPerformed (ActionEvent e) {  
            adaptee.storeButton_actionPerformed (e);  
        }  
575    }  
}
```

diplomatiki.gui

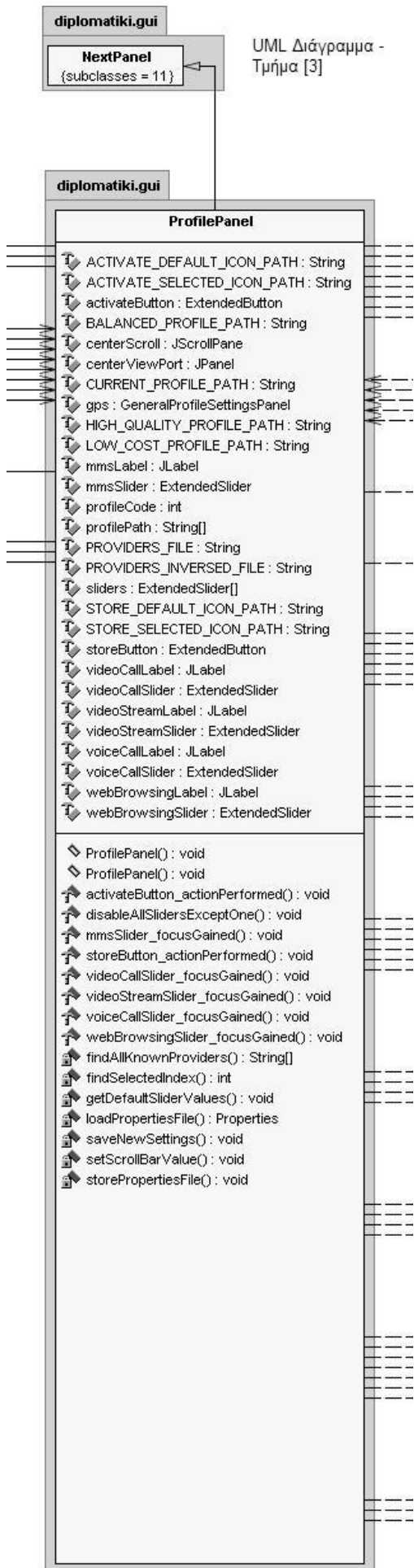
UML Διάγραμμα - Τμήμα [1]



UML Διάγραμμα - Τμήμα [2]

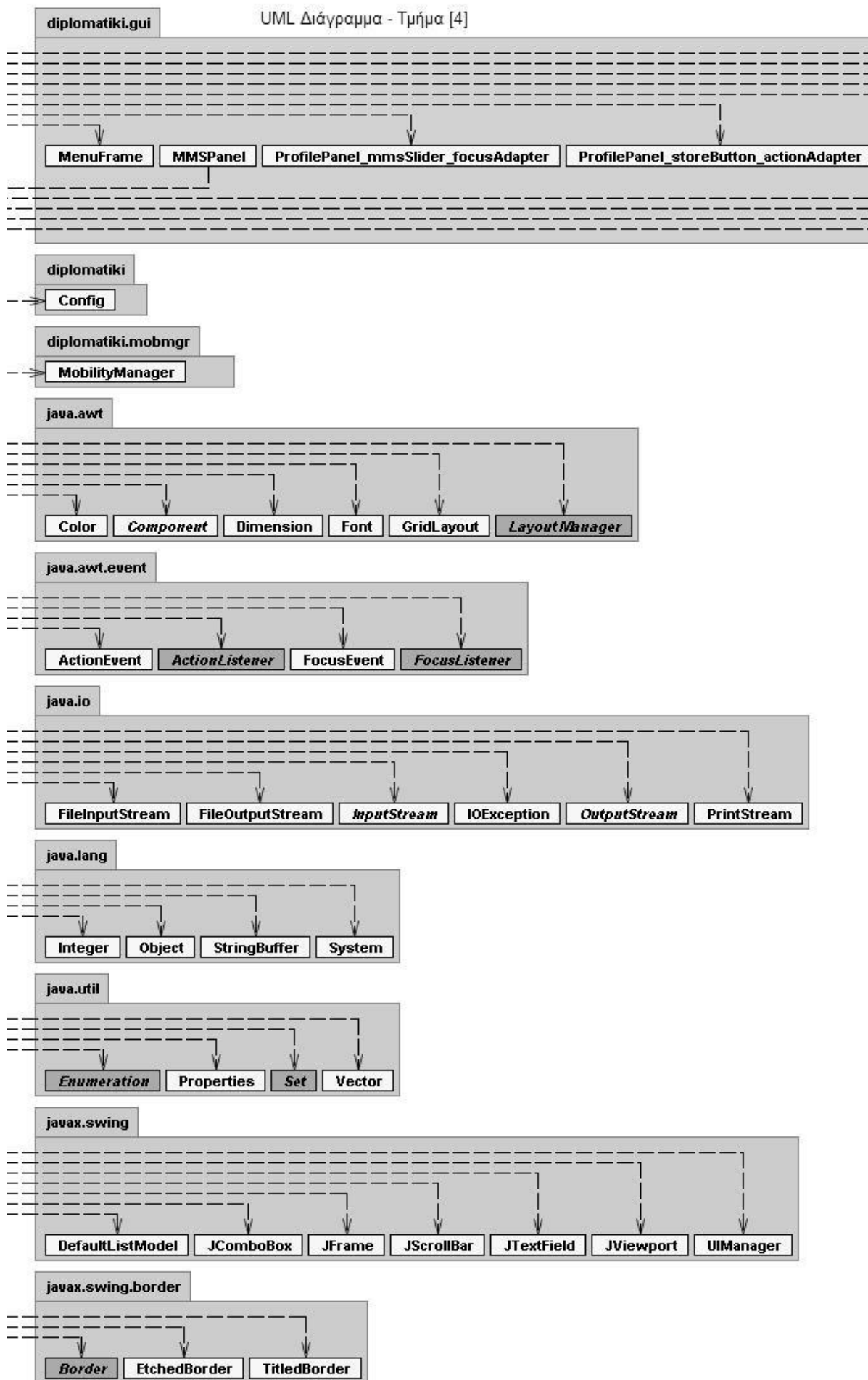


UML Διάγραμμα - Τμήμα [3]

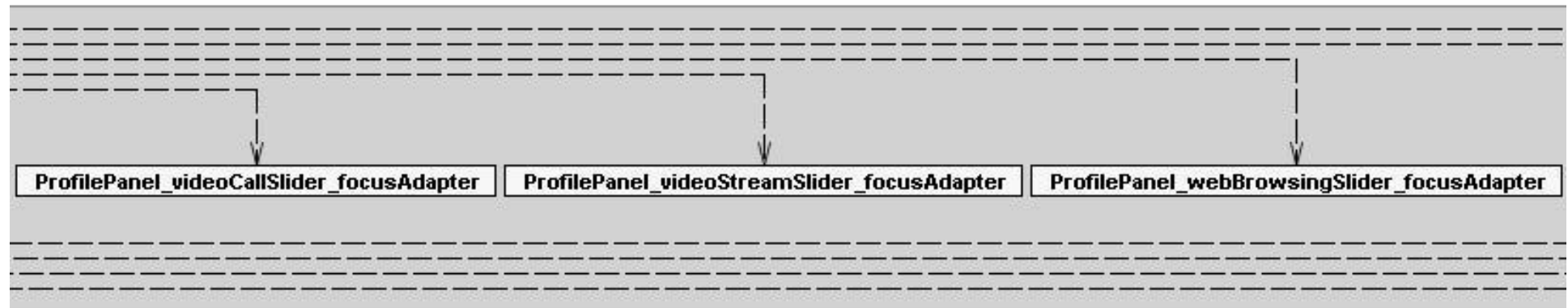




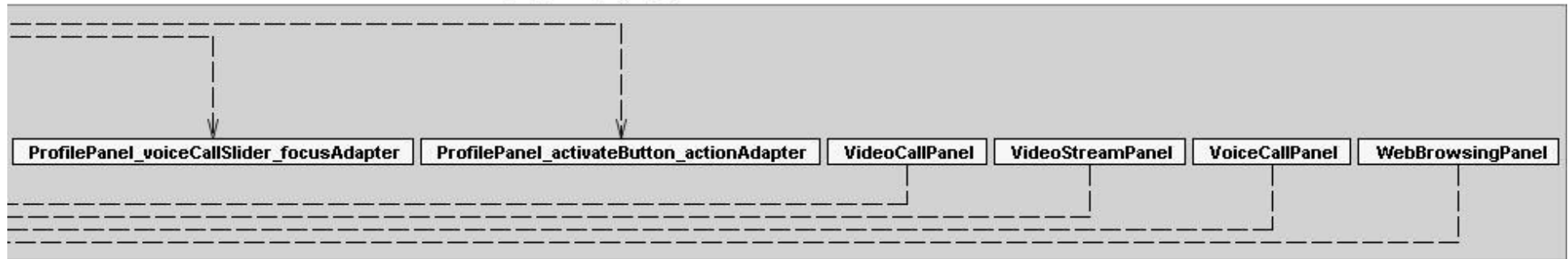
UML Διάγραμμα - Τμήμα [4]



UML Διάγραμμα - Τμήμα [5]



UML Διάγραμμα - Τμήμα [6]



```
package diplomatiki .gui;

import java.awt.Color;
import javax.swing.JProgressBar ;
5 import javax.swing.SwingConstants ;

/**
 * <p>Title: Term Paper - <code>ProgressBar</code> class</p>
 *
10 * <p>Description: This is a sub-clas of <code>JProgressBar</code>. Progress
 * bars of this type have specific characteristics (e.g. color). The progress
 * bar is progressively painted and runs in a separate thread.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class ProgressBar
    extends JProgressBar
    implements Runnable {
    private Thread runner;
25 private int max = 1;

    /**
     * Creates a new <code>ProgressBar</code> instance.
     */
30 public ProgressBar () {
    super (SwingConstants .VERTICAL, 0, 1);
    if (runner == null) {
        runner = new Thread (this);
    }
35 }

    /**
     * Creates a new <code>ProgressBar</code> instance with the specified maximum
     * value. This value represents the amount of time (in seconds) that will pass
40 * until the progress bar reaches 100%.
     *
     * @param max The progress bar's maximum value
     */
45 public ProgressBar (int max) {
    super (SwingConstants .VERTICAL, 0, max);
    this.max = max;
    this.setBackground (new Color (166, 202, 240));
    this.setStringPainted (true);
    if (runner == null) {
50 runner = new Thread (this);
    }
}

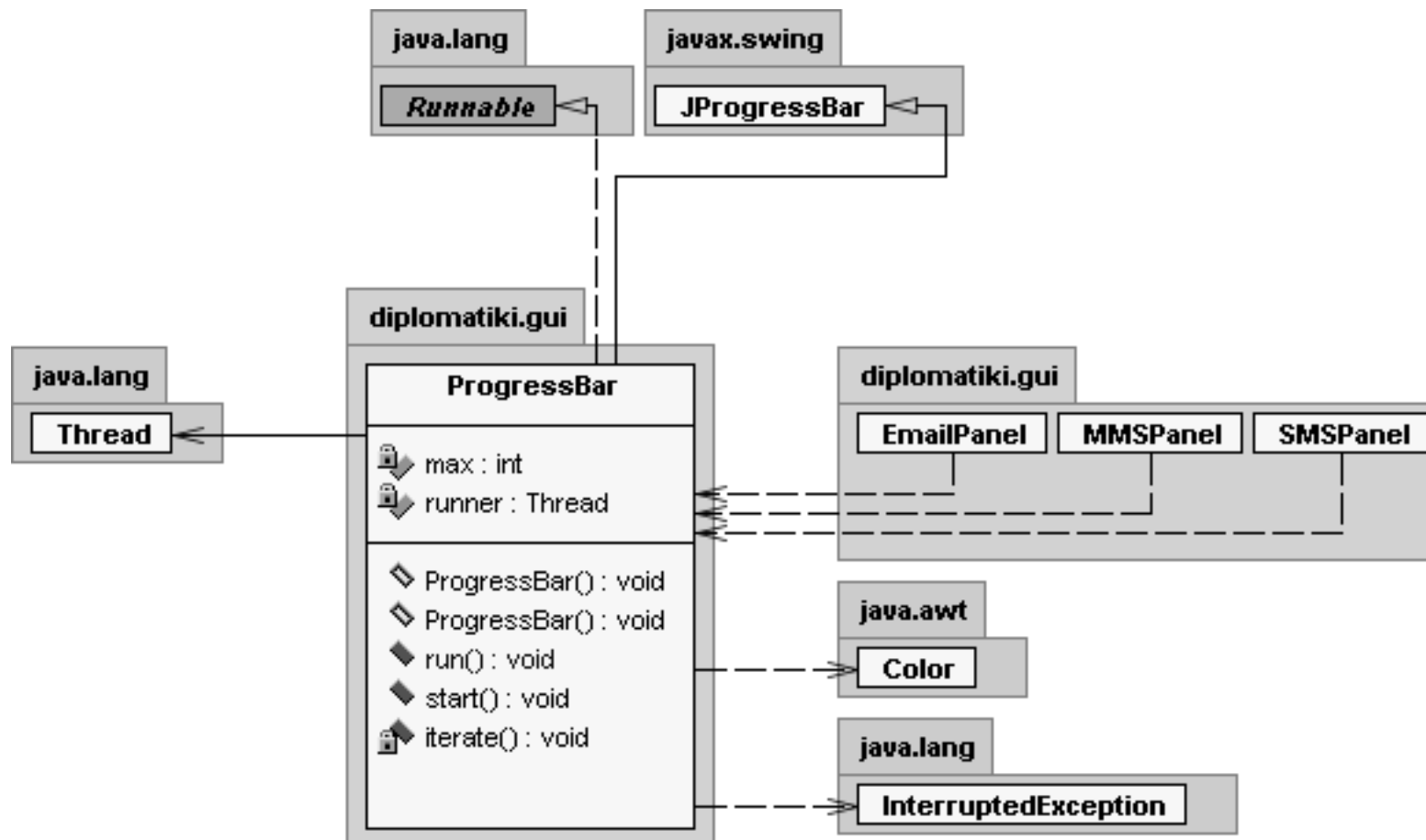
    /**
55 * Entry point for the <code>ProgressBar</code> thread. The
 * </code>iterate()</code> method is called.
     */
    public void run () {
60 iterate ();
}

    /**
65 * Progressively paints the progress bar (e.g. 0%, then 33%, then 66%, then
 * 100%).
     */
    private void iterate () {
        int count = 0;
        while (count <= max) {
            this.setValue (count);
70 try {
```

```
        runner.sleep(1000);
    }
    catch (InterruptedException ex) {
75     ex.printStackTrace();
    }
    count++;
}
}

80 /**
 * Starts the <code>ProgressBar</code> thread.
 */
public void start() {
85     if (runner != null) {
        runner.start();
    }
}

90 }
```



```

package diplomatiki .gui;

import java .awt .*;
import javax .swing .*;

5
/**
 * <p>Title: Term Paper - <code>RunningServicesInfoPanel</code> class</p>
 *
 * <p>Description: This panel is a part of the applications panel
10 * (<code>ApplicationsPanel</code>). It contains a set of editable labels. These
 * labels are used to present connection information.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
20 public class RunningServicesInfoPanel
    extends JPanel {
    JLabel serviceLabel = new JLabel ();
    JLabel technologyLabel = new JLabel ();
    JLabel qualityLabel = new JLabel ();
25 JLabel jLabel1 = new JLabel ();
    JLabel jLabel2 = new JLabel ();
    JLabel jLabel3 = new JLabel ();
    JLabel jLabel4 = new JLabel ();
    JLabel jLabel5 = new JLabel ();
30 JLabel jLabel6 = new JLabel ();
    JLabel jLabel7 = new JLabel ();
    JLabel jLabel8 = new JLabel ();
    JLabel jLabel9 = new JLabel ();
    JLabel jLabel10 = new JLabel ();
35 JLabel jLabel11 = new JLabel ();
    JLabel jLabel12 = new JLabel ();
    JLabel jLabel13 = new JLabel ();
    JLabel jLabel14 = new JLabel ();
    JLabel jLabel15 = new JLabel ();
40 JLabel jLabel16 = new JLabel ();
    JLabel [] labels = new JLabel [17];
    JLabel providerLabel = new JLabel ();

    /**
45 * Creates a new <code>RunningServicesInfoPanel</code> instance.
    */
    public RunningServicesInfoPanel () {
        try {
            jbInit ();
50        }
        catch (Exception ex) {
            ex.printStackTrace ();
        }
    }
55
    /**
     * Builds the panel.
     *
     * @throws Exception An exception is thrown if the construction of the panel
60 * is not successful
     */
    void jbInit () throws Exception {
        this .setLayout (null);
        this .setBackground (UIManager .getColor ("InternalFrame .activeTitleGradient" ));
65        this .setMinimumSize (new Dimension (300, 180));
        this .setMaximumSize (new Dimension (300, 180));
        this .setPreferredSize (new Dimension (300, 180));

        serviceLabel .setFont (new java .awt .Font ("Microsoft Sans Serif" , 1, 11));
70        serviceLabel .setText ("Service");

```

```
        serviceLabel.setBounds(new Rectangle(5, 5, 50, 15));

        technologyLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
        technologyLabel.setText("Technology");
75     technologyLabel.setBounds(new Rectangle(80, 5, 70, 15));

        providerLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
        providerLabel.setText("Provider");
80     providerLabel.setBounds(new Rectangle(180, 5, 50, 15));

        qualityLabel.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 11));
        qualityLabel.setText("Quality");
        qualityLabel.setBounds(new Rectangle(260, 5, 50, 15));

85     jLabel1.setText("");
        jLabel1.setBounds(new Rectangle(5, 40, 75, 15));
        jLabel2.setText("");
        jLabel2.setBounds(new Rectangle(80, 40, 100, 15));
        jLabel3.setText("");
90     jLabel3.setBounds(new Rectangle(180, 40, 80, 15));
        jLabel4.setText("");
        jLabel4.setBounds(new Rectangle(260, 40, 40, 15));
        jLabel5.setText("");
        jLabel5.setBounds(new Rectangle(5, 70, 75, 15));
95     jLabel6.setText("");
        jLabel6.setBounds(new Rectangle(80, 70, 100, 15));
        jLabel7.setText("");
        jLabel7.setBounds(new Rectangle(180, 70, 80, 15));
        jLabel8.setText("");
100    jLabel8.setBounds(new Rectangle(260, 70, 40, 15));
        jLabel9.setText("");
        jLabel9.setBounds(new Rectangle(5, 100, 75, 15));
        jLabel10.setText("");
        jLabel10.setBounds(new Rectangle(80, 100, 100, 15));
105    jLabel11.setText("");
        jLabel11.setBounds(new Rectangle(180, 100, 80, 15));
        jLabel12.setText("");
        jLabel12.setBounds(new Rectangle(260, 100, 40, 15));
        jLabel13.setText("");
110    jLabel13.setBounds(new Rectangle(5, 130, 75, 15));
        jLabel14.setText("");
        jLabel14.setBounds(new Rectangle(80, 130, 100, 15));
        jLabel15.setText("");
        jLabel15.setBounds(new Rectangle(180, 130, 80, 15));
115    jLabel16.setText("");
        jLabel16.setBounds(new Rectangle(260, 130, 40, 15));

        labels[0] = null;
        labels[1] = jLabel1;
120    labels[2] = jLabel2;
        labels[3] = jLabel3;
        labels[4] = jLabel4;
        labels[5] = jLabel5;
        labels[6] = jLabel6;
125    labels[7] = jLabel7;
        labels[8] = jLabel8;
        labels[9] = jLabel9;
        labels[10] = jLabel10;
        labels[11] = jLabel11;
130    labels[12] = jLabel12;
        labels[13] = jLabel13;
        labels[14] = jLabel14;
        labels[15] = jLabel15;
        labels[16] = jLabel16;

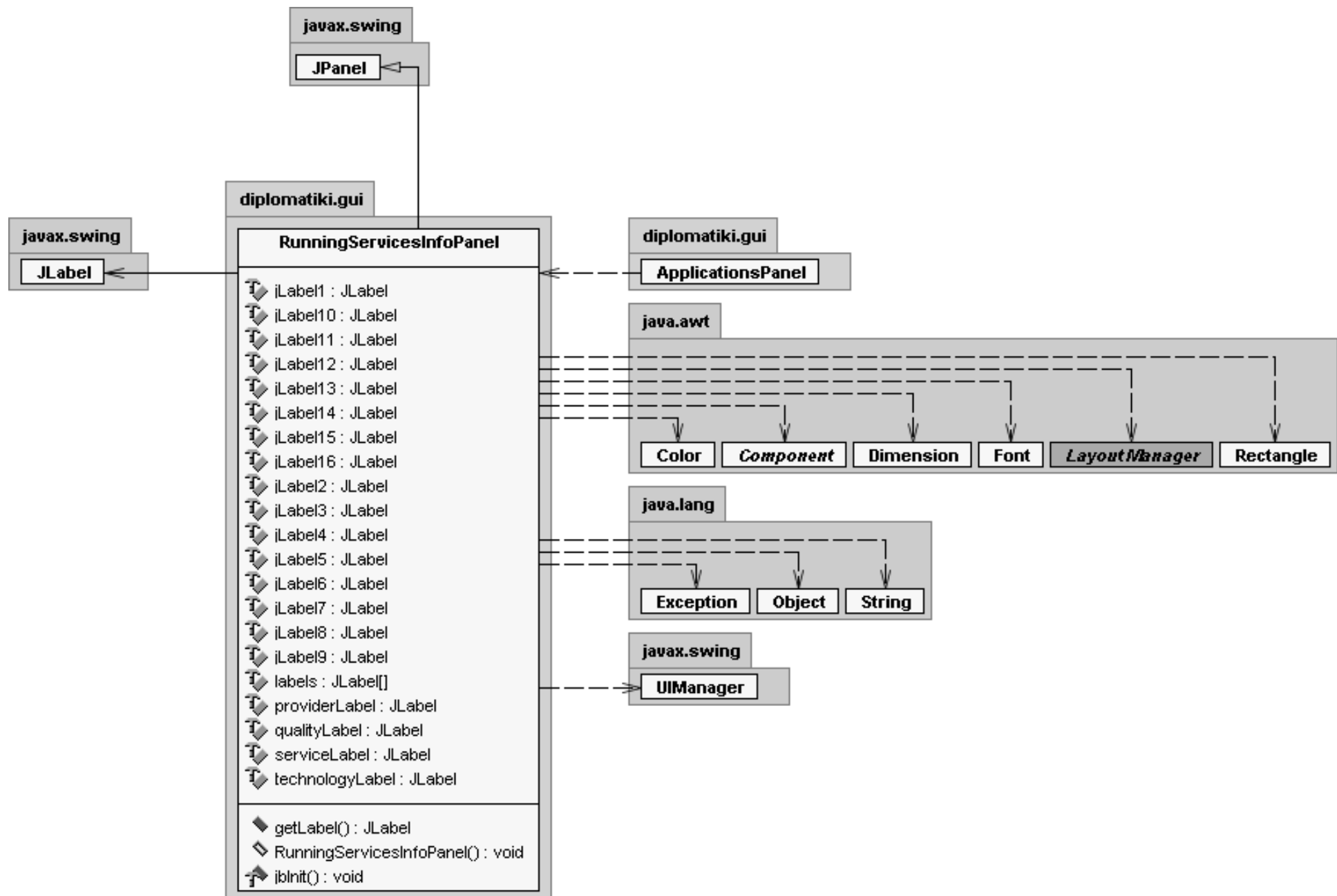
135    this.add(serviceLabel, null);
        this.add(technologyLabel, null);
        this.add(providerLabel, null);
        this.add(qualityLabel, null);

140
```



```
        this.add(jLabel11, null);
        this.add(jLabel12, null);
        this.add(jLabel13, null);
        this.add(jLabel14, null);
145    this.add(jLabel15, null);
        this.add(jLabel16, null);
        this.add(jLabel17, null);
        this.add(jLabel18, null);
        this.add(jLabel19, null);
150    this.add(jLabel110, null);
        this.add(jLabel111, null);
        this.add(jLabel112, null);
        this.add(jLabel113, null);
        this.add(jLabel114, null);
155    this.add(jLabel115, null);
        this.add(jLabel116, null);
    }

    /**
160    * Returns the label specified by the method's argument (<code>index</code>).
    *
    * @param index The index of the label that should be returned
    * @return A reference to the label specified
    */
165    public JLabel getLabel(int index) {
        return labels[index];
    }
}
170
```



```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;

/**
 * <p>Title: Term Paper - <code>SMSInputPanel</code> class</p>
 *
10 * <p>Description: This is the center panel of the SMS Service panel. It holds
 * the various input fields, where the user should fill in the appropriate
 * information (e.g. the recipient's address).</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class SMSInputPanel
    extends JPanel {
    JLabel numberLabel = new JLabel ();
    JTextField numberField = new JTextField ();
25 JLabel messageLabel = new JLabel ();
    JScrollPane scroll = new JScrollPane ();
    JTextArea message = new JTextArea ();

    /**
30 * Creates a new <code>SMSInputPanel</code> instance.
    */
    public SMSInputPanel () {
        try {
35         jbInit ();
        }
        catch (Exception ex) {
            ex.printStackTrace ();
        }
    }

40    /**
    * Builds the panel.
    *
    * @throws Exception An exception is thrown if the construction of the panel
45 * is not successful
    */
    void jbInit () throws Exception {
        this.setBackground (UIManager .getColor ("InternalFrame.activeTitleGradient" ));
        this.setMinimumSize (new Dimension (300, 140));
50        this.setMaximumSize (new Dimension (300, 140));
        this.setPreferredSize (new Dimension (300, 140));
        this.setLayout (null);

        numberLabel.setFont (new java .awt .Font ("Microsoft Sans Serif" , 1, 11));
55        numberLabel.setText ("Number");
        numberLabel.setBounds (new Rectangle (43, 21, 51, 15));

        numberField.setText ("");
        numberField.setBounds (new Rectangle (103, 19, 74, 21));

60        messageLabel.setFont (new java .awt .Font ("Microsoft Sans Serif" , 1, 11));
        messageLabel.setText ("Message");
        messageLabel.setBounds (new Rectangle (43, 59, 55, 15));

65        scroll.setHorizontalScrollBarPolicy (JScrollPane .HORIZONTAL_SCROLLBAR_NEVER );
        scroll.getViewPort ().setBackground (UIManager .getColor (
            "InternalFrame.activeTitleGradient" ));
        scroll.setBounds (new Rectangle (103, 58, 144, 72));

70        message.setText ("");

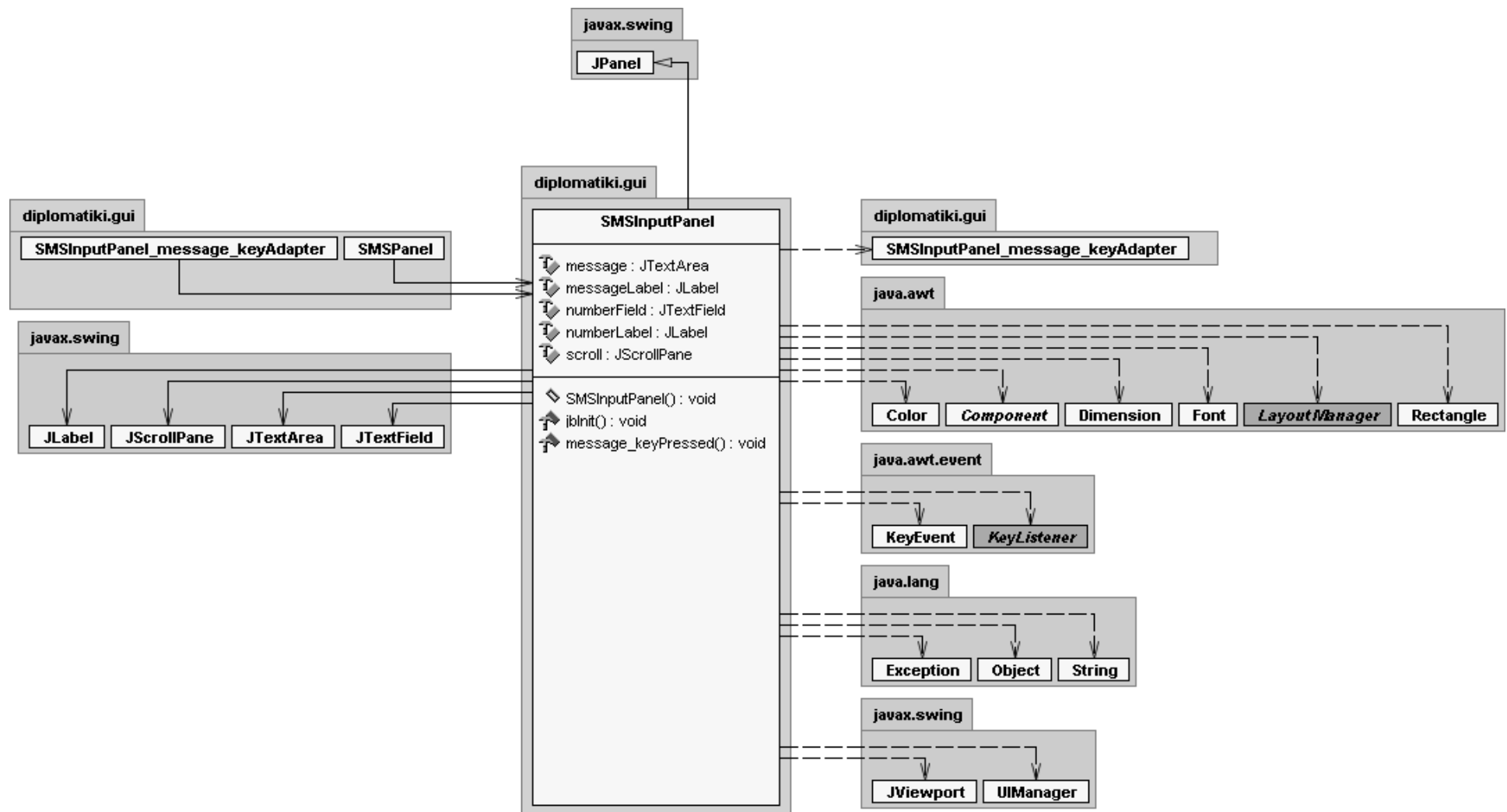
```

```
message.setLineWrap (true);
message.addKeyListener (new SMSInputPanel_message_keyAdapter (this));

    this.add(numberLabel , null);
75    this.add(scroll , null);
    this.add(messageLabel , null);
    this.add(numberField , null);
    scroll.getViewPort ().add(message , null);
}
80
/*
 * message_keyPressed
 *
 * @param e KeyEvent
85 */
void message_keyPressed (KeyEvent e) {
    if (e.getKeyCode () == KeyEvent.VK_DOWN) {
        message.transferFocus ();
    }
90    else if (e.getKeyCode () == KeyEvent.VK_UP) {
        message.transferFocusBackward ();
    }
}
95 }

class SMSInputPanel_message_keyAdapter
    extends java.awt.event.KeyAdapter {
    SMSInputPanel adaptee;
100
    SMSInputPanel_message_keyAdapter (SMSInputPanel adaptee) {
        this.adaptee = adaptee;
    }

105    public void keyPressed (KeyEvent e) {
        adaptee.message_keyPressed (e);
    }
}
```



```

package diplomatiki .gui;

import java .awt .*;
import java .awt .event .*;
5 import javax .swing .*;
import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>SMSPanel</code> class</p>
10 *
 * <p>Description: This is the SMS Service panel. This panel is shown when the
 * user chooses to send an SMS.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
15 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
20 */
public class SMSPanel
    extends JPanel {
    final int SERVICE_CODE = Config .SMS_SERVICE_CODE ;
    final int MAX_TIME = Config .BAR_MAX_TIME ;

25
    final String START_DEFAULT_ICON_PATH = Config .SMS_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config .SMS_START_SELECTED_ICON_PATH ;

    InfoPanel ip;
    ExtendedButton startButton;
    SMSInputPanel input;

    /**
    * Creates a new <code>SMSPanel</code> instance.
35 */
    public SMSPanel () {
    }

    /**
    * Creates a new <code>SMSPanel</code> instance with the specified
    * parent frame and previous panel.
    *
    * @param myFrame The frame which this panel belongs to
    * @param previousPanel The previous panel that was visible in the
45 * <code>myFrame</code> frame. This reference is very useful if the user
    * wants to return to the previous panel
    */
    public SMSPanel (JFrame myFrame , JPanel previousPanel ) {
50
        super (myFrame , previousPanel );

        GridLayout1 .setColumns (1);
        GridLayout1 .setRows (1);

        startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
55 START_SELECTED_ICON_PATH );
        startButton .addActionListener (new SMSPanel_startButton_actionAdapter (this));

        northPanel .setMinimumSize (new Dimension (400 , 42));
        northPanel .setMinimumSize (new Dimension (400 , 42));
60 northPanel .setPreferredSize (new Dimension (400 , 42));
        northPanel .add (startButton );

        input = new SMSInputPanel ();

65 centerPanel .add (input);
    }

    /**
    * Action performed when the "Start" service button is pressed.
70 */

```

```

private void startAction () {
    //Check if the address/number field is filled in
    if (input.numberField.getText ().equals ("")) {
        new MessageWindow (myFrame, "Input Required:", "Please type a number" );
75     return;
    }

    //Run IIS
    diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
80     executeIIS (SERVICE_CODE, true, -1);

    //Check if myInfo is null
    if (myInfo == null) {
        new MessageWindow (myFrame);
85     return;
    }

    //Construct ConnectionInfoPanel
    ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
90     getAccessTechnology (),
        myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());

    //Construct InfoPanel
    ip = new InfoPanel (myFrame, this, "Sending SMS to: ",
95     input.numberField.getText ());

    //Add the ConnectionInfoPanel to the InfoPanel
    ip.centerPanel .add (cip);

100    //Create ProgressBar and Add it to the InfoPanel
    ProgressBar bar = new ProgressBar (MAX_TIME);
    //ip.add ("East", bar);
    ip.add (bar, BorderLayout .EAST);

105    //Show the InfoPanel
    this.setVisible (false);
    myFrame.setContentPane (ip);
    myFrame.setVisible (false);
    myFrame.setVisible (true);
110    ip.setVisible (true);

    //Start the ProgressBar
    bar.start ();
}

115 /**
 * Action performed when the "Start" service button is pressed. The
 * <code>startAction ()</code> method is invoked.
 *
120 * @param e ActionEvent
 */
void startButton_actionPerformed (ActionEvent e) {
    startAction ();
}

125 }

class SMSPanel_startButton_actionAdapter
    implements java.awt.event.ActionListener {
130     SMSPanel adaptee;

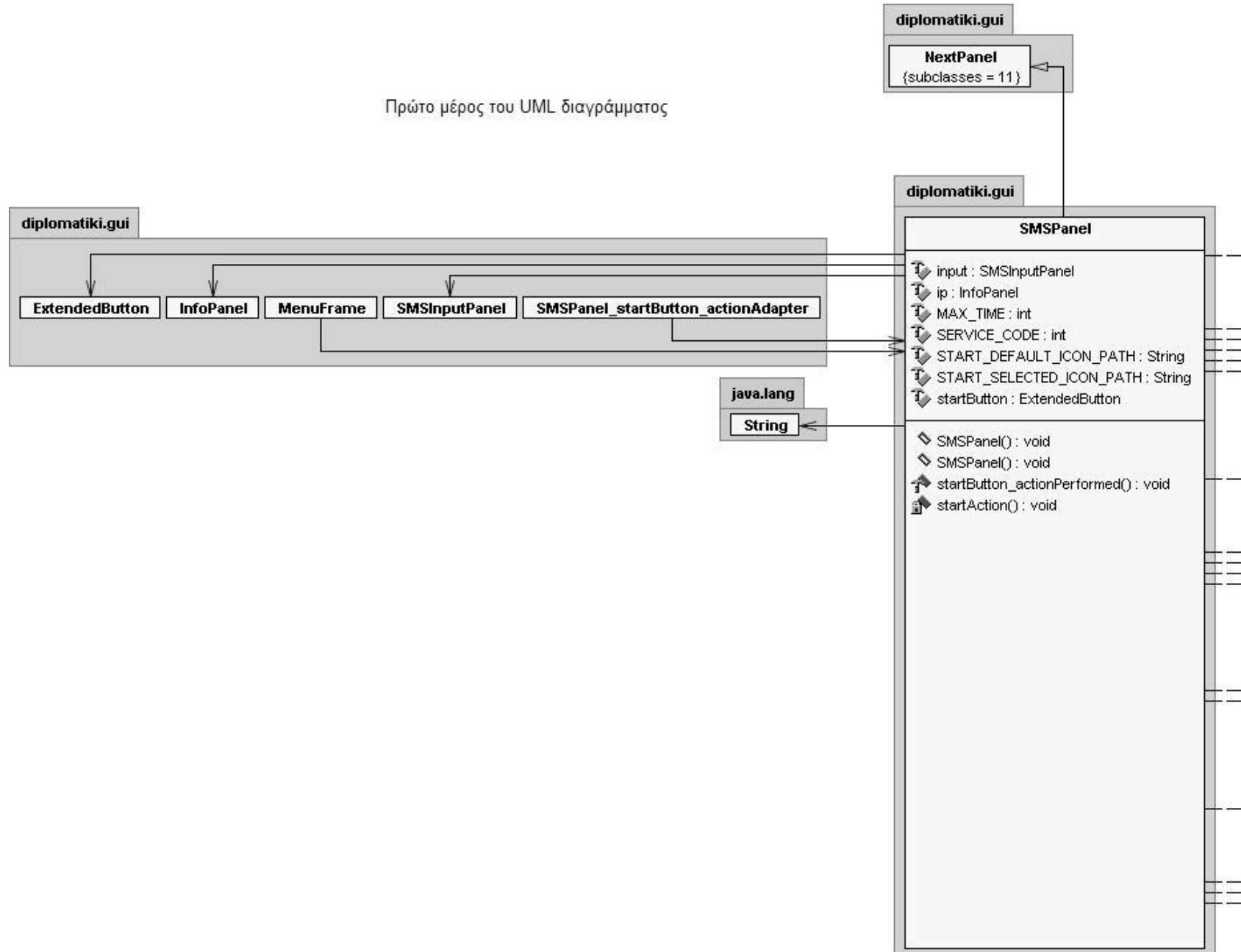
    SMSPanel_startButton_actionAdapter (SMSPanel adaptee) {
        this.adaptee = adaptee;
    }

135     public void actionPerformed (ActionEvent e) {
        adaptee.startButton_actionPerformed (e);
    }
}

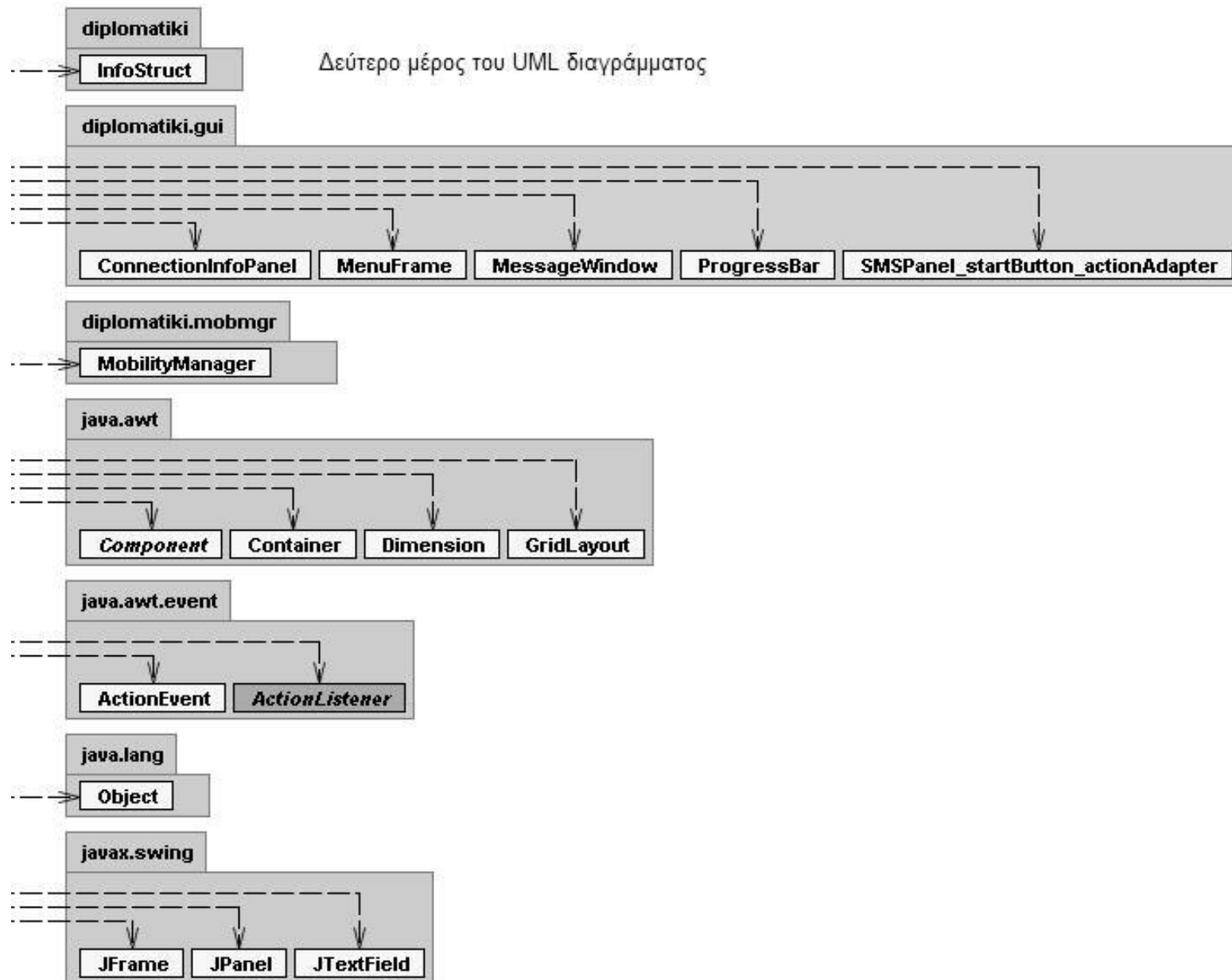
140

```

Πρώτο μέρος του UML διαγράμματος







```

package diplomatiki .gui;

import java.awt.event.*;
import javax.swing.*;
5 import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>VideoCallPanel</code> class</p>
 *
10 * <p>Description: This is the Video Call Service panel. This panel is shown
 * when the user chooses to make a video call.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: JadaCode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
20 public class VideoCallPanel
    extends JPanel {
    final int SERVICE_CODE = Config.VIDEOCALL_SERVICE_CODE ;

    final String START_DEFAULT_ICON_PATH = Config.
25     VIDEOCALL_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config.
        VIDEOCALL_START_SELECTED_ICON_PATH ;
    final String START_DISABLED_ICON_PATH = Config.
        VIDEOCALL_START_DISABLED_ICON_PATH ;
30    final String END_DEFAULT_ICON_PATH = Config.VIDEOCALL_END_DEFAULT_ICON_PATH ;
    final String END_SELECTED_ICON_PATH = Config.VIDEOCALL_END_SELECTED_ICON_PATH ;
    final String END_DISABLED_ICON_PATH = Config.VIDEOCALL_END_DISABLED_ICON_PATH ;
    final String INFO_DEFAULT_ICON_PATH = Config.INFO_DEFAULT_ICON_PATH ;
    final String INFO_SELECTED_ICON_PATH = Config.INFO_SELECTED_ICON_PATH ;
35    final String INFO_DISABLED_ICON_PATH = Config.INFO_DISABLED_ICON_PATH ;
    final String EDIT_DEFAULT_ICON_PATH = Config.EDIT_DEFAULT_ICON_PATH ;
    final String EDIT_SELECTED_ICON_PATH = Config.EDIT_SELECTED_ICON_PATH ;

    InfoPanel ip;
40    ExtendedButton startButton, endButton, infoButton, settingsButton;
    JLabel numberLabel = new JLabel ();
    JTextField numberField = new JTextField ();

    /**
45     * Creates a new <code>VideoCallPanel</code> instance.
     */
    public VideoCallPanel () {
    }

50    /**
     * Creates a new <code>VideoCallPanel</code> instance with the specified
     * parent frame and previous panel.
     *
     * @param myFrame The frame which this panel belongs to
55     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
     * wants to return to the previous panel
     */
    public VideoCallPanel (JFrame myFrame, JPanel previousPanel) {
60         super (myFrame, previousPanel);

        GridLayout1.setColumns (1);
        GridLayout1.setRows (4);

65         startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
            START_SELECTED_ICON_PATH );
        startButton.setDisabledIcon (new ImageIcon (START_DISABLED_ICON_PATH ));
        endButton = new ExtendedButton (END_DEFAULT_ICON_PATH ,
            END_SELECTED_ICON_PATH );
70         endButton.setDisabledIcon (new ImageIcon (END_DISABLED_ICON_PATH ));

```

```

infoButton = new ExtendedButton (INFO_DEFAULT_ICON_PATH ,
                                INFO_SELECTED_ICON_PATH );
infoButton.setDisabledIcon (new ImageIcon (INFO_DISABLED_ICON_PATH ));
settingsButton = new ExtendedButton (EDIT_DEFAULT_ICON_PATH ,
75     EDIT_SELECTED_ICON_PATH );

startButton .
    addActionListener (new VideoCallPanel_startButton_actionAdapter (this));
endButton .
80     addActionListener (new VideoCallPanel_endButton_actionAdapter (this));
infoButton .
    addActionListener (new VideoCallPanel_infoButton_actionAdapter (this));
settingsButton .
85     addActionListener (new
        VideoCallPanel_settingsButton_actionAdapter (this));

endButton.setEnabled (false);
infoButton.setEnabled (false);

90 northPanel.add (startButton );
northPanel.add (endButton );
northPanel.add (infoButton );
northPanel.add (settingsButton );

95 numberLabel.setText ("Number");
numberLabel.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
numberField.setColumns (12);

centerPanel.add (numberLabel );
100 centerPanel.add (numberField );
}

/**
 * Refreshes (updates) the presented connection information concerning this
105 * service. The new information is contained in </code>myInfo</code>.
 *
 * @param myInfo The new information that should be presented
 */
public void refreshConnectionInfo (diplomatiki.InfoStruct myInfo) {
110     ConnectionInfoPanel cip = null;

    //Check if myInfo is null
    if (myInfo == null) {
        new MessageWindow (myFrame);
115     endAction ();
        ( (MenuFrame) myFrame).reConstructApplicationsPanel ();
        if ( ( (MenuFrame) myFrame).getContentPane () == ip) {
            cip = new ConnectionInfoPanel ("", "", "", "");
        }
120     else {
        return;
    }
}

125 //If not, construct a new ConnectionInfoPanel
if (cip == null) {
    cip = new ConnectionInfoPanel (myInfo .
                                getAccessTechnology (),
                                myInfo.getProviderName (), myInfo.getQos (),
130                                myInfo.getIpAddress ());
}

//Add the ConnectionInfoPanel to the InfoPanel
ip.centerPanel.removeAll ();
135 ip.centerPanel.add (cip);

//Repaint the InfoPanel
ip.setVisible (false);
ip.setVisible (true);
140 ip.backButton.grabFocus ();

```

```

    }

    /**
     * Action performed when the "Start" service button is pressed.
145    */
    private void startAction () {
        //Check if the address/number field is filled in
        if (numberField.getText().equals("")) {
150            new MessageWindow (myFrame, "Input Required:", "Please type a number" );
            return;
        }

        //Run IIS
        diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
155         executeIIS (SERVICE_CODE, true, -1);

        //Check if myInfo is null
        if (myInfo == null) {
160            new MessageWindow (myFrame);
            endAction ();
            return;
        }

        //Construct ConnectionInfoPanel
165        ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
            getAccessTechnology (),
            myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());

        //Construct InfoPanel
170        ip = new InfoPanel (myFrame, this, "Video Call to: ", numberField.getText ());

        //Add the ConnectionInfoPanel to the InfoPanel
        ip.centerPanel.add(cip);

175        //Disable the Start Service button
        startButton.setEnabled (false);
        //Enable the End Service and View Info buttons
        endButton.setEnabled (true);
        infoButton.setEnabled (true);

180        //Show the InfoPanel
        this.setVisible (false);
        myFrame.setContentPane (ip);
        myFrame.setVisible (false);
185        myFrame.setVisible (true);
        ip.setVisible (true);
    }

    /**
190    * Action performed when the "End" service button is pressed.
     */
    private void endAction () {
        ( (MenuFrame) myFrame).getMobilityManager ().removeFromRunningServices (
            SERVICE_CODE );
195        //Enable the Start Service button
        startButton.setEnabled (true);
        //Disable the End Service and View Info buttons
        endButton.setEnabled (false);
        infoButton.setEnabled (false);

200        //Give focus to the Start Service button
        startButton.grabFocus ();
    }

205    /**
     * Action performed when the "View Info" button is pressed.
     */
    private void infoAction () {
        //Show the InfoPanel
210        if (ip != null) {

```

```

        this.setVisible (false);
        myFrame.setContentPane (ip);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
215     ip.setVisible (true);
    }
}

/**
220  * Action performed when the "Edit Settings" button is pressed.
 */
private void settingsAction () {
    //Construct ProfilePanel
    ProfilePanel pp = new ProfilePanel (myFrame, this, 1);
225     //Disable all other sliders in the ProfilePanel
    pp.disableAllSlidersExceptOne (2);

    //Show the ProfilePanel
    this.setVisible (false);
230     myFrame.setContentPane (pp);
    myFrame.setVisible (false);
    myFrame.setVisible (true);
    pp.setVisible (true);
}

235 /**
 * Action performed when the "Start" service button is pressed. The
 * <code>startAction()</code> method is invoked.
 *
 * @param e ActionEvent
 */
240 void startButton_actionPerformed (ActionEvent e) {
    startAction ();
}

245 /**
 * Action performed when the "End" service button is pressed. The
 * <code>endAction()</code> method is invoked.
 *
 * @param e ActionEvent
 */
250 void endButton_actionPerformed (ActionEvent e) {
    endAction ();
}

255 /**
 * Action performed when the "View Info" button is pressed. The
 * <code>infoAction()</code> method is invoked.
 *
 * @param e ActionEvent
 */
260 void infoButton_actionPerformed (ActionEvent e) {
    infoAction ();
}

265 /**
 * Action performed when the "Edit Settings" button is pressed. The
 * <code>settingsAction()</code> method is invoked.
 *
 * @param e ActionEvent
 */
270 void settingsButton_actionPerformed (ActionEvent e) {
    settingsAction ();
}

275 }

class VideoCallPanel_startButton_actionAdapter
    implements java.awt.event.ActionListener {
280     VideoCallPanel adaptee;

```

```
VideoCallPanel_startButton_actionAdapter (VideoCallPanel adaptee) {
    this.adaptee = adaptee;
}
285
public void actionPerformed (ActionEvent e) {
    adaptee.startButton_actionPerformed (e);
}
}
290
class VideoCallPanel_endButton_actionAdapter
    implements java.awt.event.ActionListener {
    VideoCallPanel adaptee;

295
    VideoCallPanel_endButton_actionAdapter (VideoCallPanel adaptee) {
        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
300
        adaptee.endButton_actionPerformed (e);
    }
}

class VideoCallPanel_infoButton_actionAdapter
305
    implements java.awt.event.ActionListener {
    VideoCallPanel adaptee;

    VideoCallPanel_infoButton_actionAdapter (VideoCallPanel adaptee) {
310
        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.infoButton_actionPerformed (e);
    }
}
315

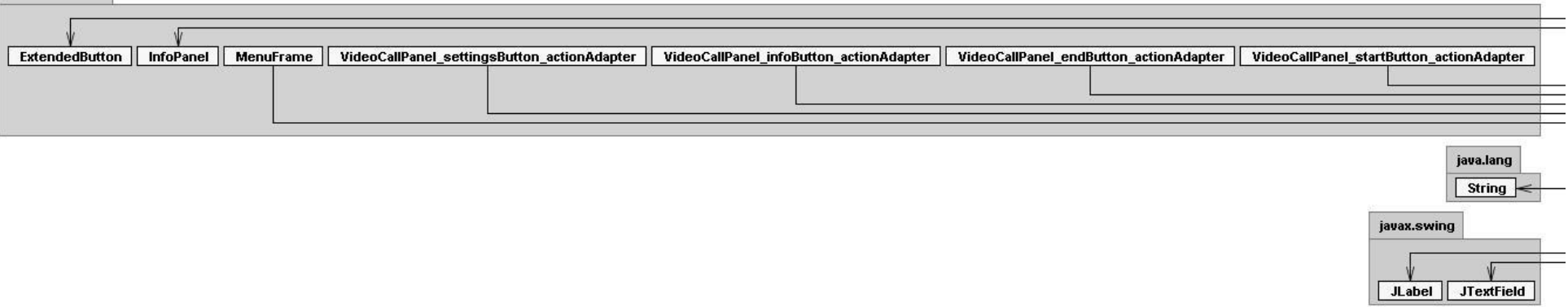
class VideoCallPanel_settingsButton_actionAdapter
    implements java.awt.event.ActionListener {
    VideoCallPanel adaptee;

320
    VideoCallPanel_settingsButton_actionAdapter (VideoCallPanel adaptee) {
        this.adaptee = adaptee;
    }

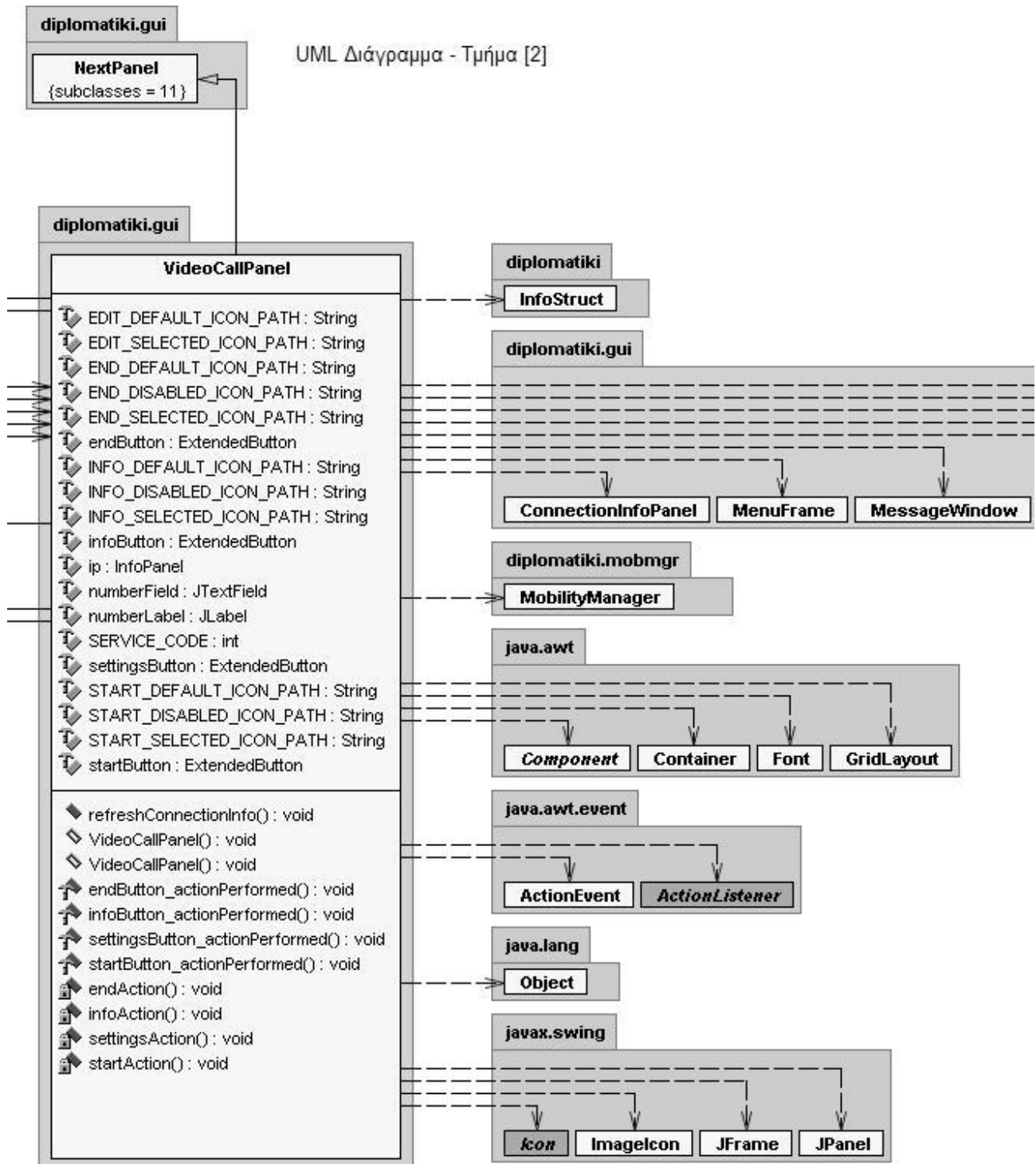
325
    public void actionPerformed (ActionEvent e) {
        adaptee.settingsButton_actionPerformed (e);
    }
}
```

diplomatiki.gui

UML Διάγραμμα - Τμήμα [1]



UML Διάγραμμα - Τμήμα [2]





UML Διάγραμμα - Τμήμα [3]



```

package diplomatiki .gui;

import java.awt.event.*;
import javax.swing.*;
5 import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>VideoStreamPanel</code> class</p>
 *
10 * <p>Description: This is the Video Stream Service panel. This panel is shown
 * when the user chooses to receive a video stream.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
20 public class VideoStreamPanel
    extends JPanel {
    final int SERVICE_CODE = Config.VIDEOSTREAM_SERVICE_CODE ;

    final String START_DEFAULT_ICON_PATH = Config.
25     VIDEOSTREAM_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config.
    VIDEOSTREAM_START_SELECTED_ICON_PATH ;
    final String START_DISABLED_ICON_PATH = Config.
    VIDEOSTREAM_START_DISABLED_ICON_PATH ;
30    final String END_DEFAULT_ICON_PATH = Config.
    VIDEOSTREAM_END_DEFAULT_ICON_PATH ;
    final String END_SELECTED_ICON_PATH = Config.
    VIDEOSTREAM_END_SELECTED_ICON_PATH ;
    final String END_DISABLED_ICON_PATH = Config.
35     VIDEOSTREAM_END_DISABLED_ICON_PATH ;
    final String INFO_DEFAULT_ICON_PATH = Config.INFO_DEFAULT_ICON_PATH ;
    final String INFO_SELECTED_ICON_PATH = Config.INFO_SELECTED_ICON_PATH ;
    final String INFO_DISABLED_ICON_PATH = Config.INFO_DISABLED_ICON_PATH ;
    final String EDIT_DEFAULT_ICON_PATH = Config.EDIT_DEFAULT_ICON_PATH ;
40    final String EDIT_SELECTED_ICON_PATH = Config.EDIT_SELECTED_ICON_PATH ;

    InfoPanel ip;
    ExtendedButton startButton, endButton, infoButton, settingsButton;
    JLabel urlLabel = new JLabel ();
45    JTextField urlField = new JTextField ();

    /**
     * Creates a new <code>VideoStreamPanel</code> instance.
     */
50    public VideoStreamPanel () {
    }

    /**
     * Creates a new <code>VideoStreamPanel</code> instance with the specified
55     * parent frame and previous panel.
     *
     * @param myFrame The frame which this panel belongs to
     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
60     * wants to return to the previous panel
     */
    public VideoStreamPanel (JFrame myFrame, JPanel previousPanel) {
        super (myFrame, previousPanel);

65        GridLayout1.setColumns (1);
        GridLayout1.setRows (4);

        startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
70         START_SELECTED_ICON_PATH );
        startButton.setDisabledIcon (new ImageIcon (START_DISABLED_ICON_PATH ));

```

```

        endButton = new ExtendedButton (END_DEFAULT_ICON_PATH ,
                                        END_SELECTED_ICON_PATH );
        endButton.setDisabledIcon (new ImageIcon (END_DISABLED_ICON_PATH ));
        infoButton = new ExtendedButton (INFO_DEFAULT_ICON_PATH ,
75         INFO_SELECTED_ICON_PATH );
        infoButton.setDisabledIcon (new ImageIcon (INFO_DISABLED_ICON_PATH ));
        settingsButton = new ExtendedButton (EDIT_DEFAULT_ICON_PATH ,
                                        EDIT_SELECTED_ICON_PATH );

80         startButton .
            addActionListener (new VideoStreamPanel_startButton_actionAdapter (this));
        endButton .
            addActionListener (new VideoStreamPanel_endButton_actionAdapter (this));
        infoButton .
85         addActionListener (new VideoStreamPanel_infoButton_actionAdapter (this));
        settingsButton .
            addActionListener (new
                                VideoStreamPanel_settingsButton_actionAdapter (this));

90         endButton.setEnabled (false);
        infoButton.setEnabled (false);

        northPanel.add (startButton);
        northPanel.add (endButton);
95         northPanel.add (infoButton);
        northPanel.add (settingsButton);

        urlLabel.setText ("URL");
        urlLabel.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
100         urlField.setColumns (20);

        centerPanel.add (urlLabel);
        centerPanel.add (urlField);
    }

105 /**
 * Refreshes (updates) the presented connection information concerning this
 * service. The new information is contained in </code>myInfo</code>.
 *
110 * @param myInfo The new information that should be presented
 */
public void refreshConnectionInfo (diplomatiiki.InfoStruct myInfo) {
    ConnectionInfoPanel cip = null;

115     //Check if myInfo is null
    if (myInfo == null) {
        new MessageWindow (myFrame);
        endAction ();
        ( (MenuFrame) myFrame).reConstructApplicationsPanel ();
120         if ( ( (MenuFrame) myFrame).getContentPane () == ip) {
            cip = new ConnectionInfoPanel ("", "", "", "");
        }
        else {
125             return;
        }
    }

    //If not, construct a new ConnectionInfoPanel
    if (cip == null) {
130         cip = new ConnectionInfoPanel (myInfo .
                                        getAccessTechnology (),
                                        myInfo.getProviderName (), myInfo.getQos (),
                                        myInfo.getIpAddress ());
    }

135     //Add the ConnectionInfoPanel to the InfoPanel
    ip.centerPanel.removeAll ();
    ip.centerPanel.add (cip);

140     //Repaint the InfoPanel

```

```

        ip.setVisible (false);
        ip.setVisible (true);
        ip.backButton.grabFocus ();
    }
145
    /**
     * Action performed when the "Start" service button is pressed.
     */
    private void startAction () {
150        //Check if the address/number field is filled in
        if (urlField.getText ().equals ("")) {
            new MessageWindow (myFrame, "Input Required:", "Please type a url" );
            return;
        }
155
        //Run IIS
        diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
            executeIIS (SERVICE_CODE, true, -1);

160        //Check if myInfo is null
        if (myInfo == null) {
            new MessageWindow (myFrame);
            endAction ();
            return;
165        }

        //Construct ConnectionInfoPanel
        ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
            getAccessTechnology (),
170            myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());

        //Construct InfoPanel
        ip = new InfoPanel (myFrame, this,
            "Video Stream from: ", urlField.getText ());
175

        //Add the ConnectionInfoPanel to the InfoPanel
        ip.centerPanel.add (cip);

        //Disable the Start Service button
180        startButton.setEnabled (false);
        //Enable the End Service and View Info buttons
        endButton.setEnabled (true);
        infoButton.setEnabled (true);

185        //Show the InfoPanel
        this.setVisible (false);
        myFrame.setContentPane (ip);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
190        ip.setVisible (true);
    }

    /**
     * Action performed when the "End" service button is pressed.
     */
195    private void endAction () {
        ( (MenuFrame) myFrame).getMobilityManager ().removeFromRunningServices (
            SERVICE_CODE );
        //Enable the Start Service button
        startButton.setEnabled (true);
200        //Disable the End Service and View Info buttons
        endButton.setEnabled (false);
        infoButton.setEnabled (false);

205        //Give focus to the Start Service button
        startButton.grabFocus ();
    }

    /**
210    * Action performed when the "View Info" button is pressed.

```

```
    */
    private void infoAction () {
        //Show the InfoPanel
        if (ip != null) {
215         this.setVisible (false);
            myFrame.setContentPane (ip);
            myFrame.setVisible (false);
            myFrame.setVisible (true);
            ip.setVisible (true);
220     }
    }

    /**
    * Action performed when the "Edit Settings" button is pressed.
225     */
    private void settingsAction () {
        //Construct ProfilePanel
        ProfilePanel pp = new ProfilePanel (myFrame, this, 1);
        //Disable all other sliders in the ProfilePanel
230     pp.disableAllSlidersExceptOne (3);

        //Show the ProfilePanel
        this.setVisible (false);
        myFrame.setContentPane (pp);
235     myFrame.setVisible (false);
        myFrame.setVisible (true);
        pp.setVisible (true);
    }

    /**
240     * Action performed when the "Start" service button is pressed. The
    * <code>startAction()</code> method is invoked.
    *
    * @param e ActionEvent
245     */
    void startButton_actionPerformed (ActionEvent e) {
        startAction ();
    }

    /**
250     * Action performed when the "End" service button is pressed. The
    * <code>endAction()</code> method is invoked.
    *
    * @param e ActionEvent
255     */
    void endButton_actionPerformed (ActionEvent e) {
        endAction ();
    }

    /**
260     * Action performed when the "View Info" button is pressed. The
    * <code>infoAction()</code> method is invoked.
    *
    * @param e ActionEvent
265     */
    void infoButton_actionPerformed (ActionEvent e) {
        infoAction ();
    }

    /**
270     * Action performed when the "Edit Settings" button is pressed. The
    * <code>settingsAction()</code> method is invoked.
    *
    * @param e ActionEvent
275     */
    void settingsButton_actionPerformed (ActionEvent e) {
        settingsAction ();
    }

280 }
```

```
class VideoStreamPanel_startButton_actionAdapter
    implements java.awt.event.ActionListener {
    VideoStreamPanel adaptee;
285
    VideoStreamPanel_startButton_actionAdapter (VideoStreamPanel adaptee) {
        this.adaptee = adaptee;
    }

290    public void actionPerformed (ActionEvent e) {
        adaptee.startButton_actionPerformed (e);
    }
}

295 class VideoStreamPanel_endButton_actionAdapter
    implements java.awt.event.ActionListener {
    VideoStreamPanel adaptee;

    VideoStreamPanel_endButton_actionAdapter (VideoStreamPanel adaptee) {
300        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.endButton_actionPerformed (e);
305    }
}

class VideoStreamPanel_infoButton_actionAdapter
    implements java.awt.event.ActionListener {
310    VideoStreamPanel adaptee;

    VideoStreamPanel_infoButton_actionAdapter (VideoStreamPanel adaptee) {
        this.adaptee = adaptee;
    }
315
    public void actionPerformed (ActionEvent e) {
        adaptee.infoButton_actionPerformed (e);
    }
}
320

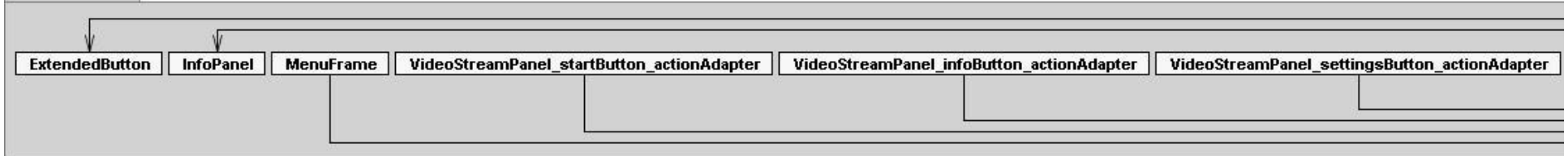
class VideoStreamPanel_settingsButton_actionAdapter
    implements java.awt.event.ActionListener {
    VideoStreamPanel adaptee;

325    VideoStreamPanel_settingsButton_actionAdapter (VideoStreamPanel adaptee) {
        this.adaptee = adaptee;
    }

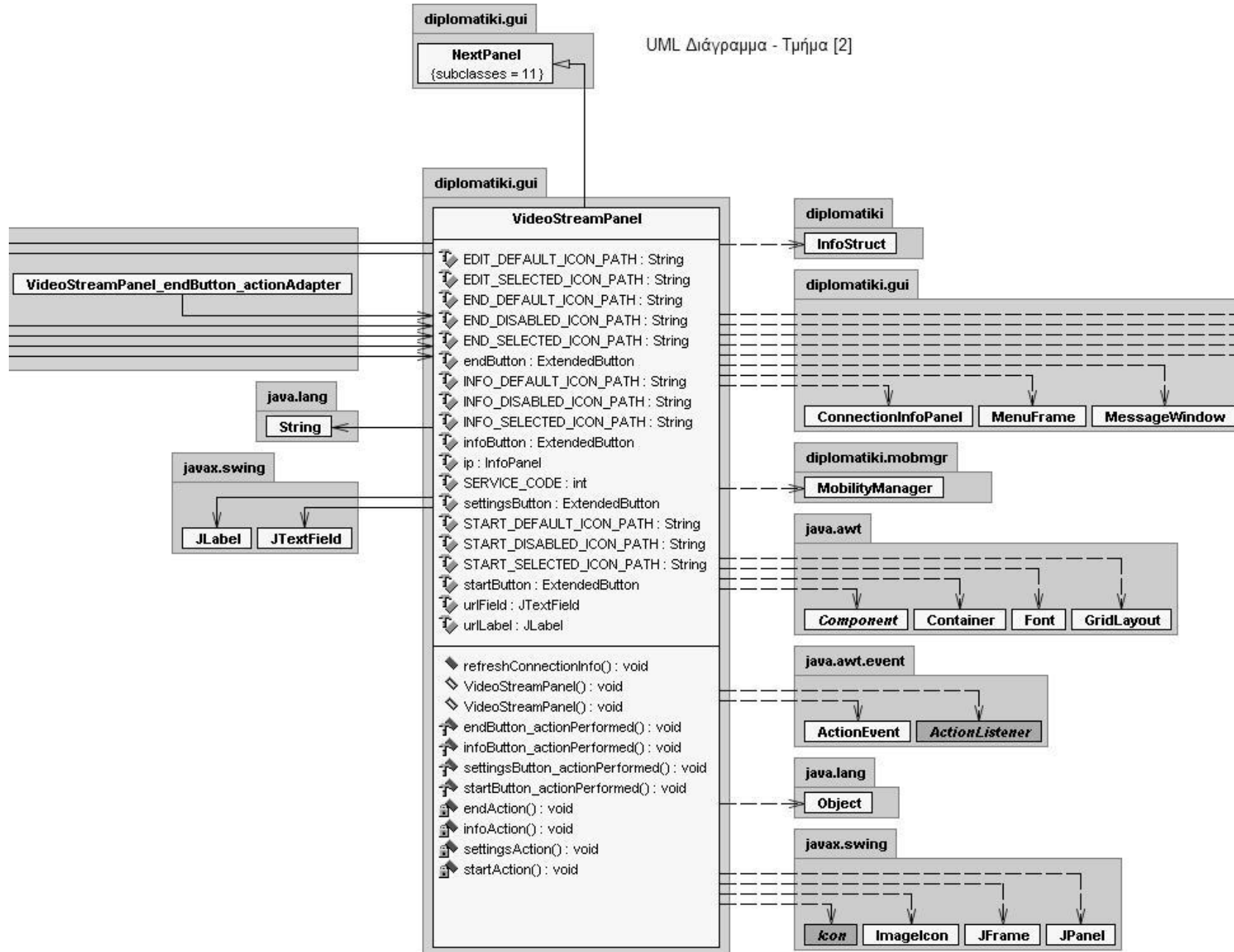
    public void actionPerformed (ActionEvent e) {
330        adaptee.settingsButton_actionPerformed (e);
    }
}
```

diplomatiki.gui

UML Διάγραμμα - Τμήμα [1]

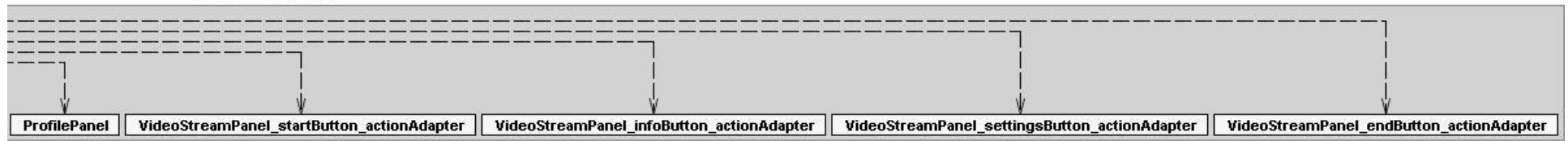


UML Διάγραμμα - Τμήμα [2]





UML Διάγραμμα - Τμήμα [3]



```

package diplomatiki .gui;

import java.awt.event.*;
import javax.swing.*;
5 import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>VoiceCallPanel</code> class</p>
 *
10 * <p>Description: This is the Voice Call Service panel. This panel is shown
 * when the user chooses to make a voice call.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
20 public class VoiceCallPanel
    extends JPanel {
    final int SERVICE_CODE = Config.VOICECALL_SERVICE_CODE ;

    final String START_DEFAULT_ICON_PATH = Config.
25     VOICECALL_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config.
        VOICECALL_START_SELECTED_ICON_PATH ;
    final String START_DISABLED_ICON_PATH = Config.
        VOICECALL_START_DISABLED_ICON_PATH ;
30    final String END_DEFAULT_ICON_PATH = Config.VOICECALL_END_DEFAULT_ICON_PATH ;
    final String END_SELECTED_ICON_PATH = Config.VOICECALL_END_SELECTED_ICON_PATH ;
    final String END_DISABLED_ICON_PATH = Config.VOICECALL_END_DISABLED_ICON_PATH ;
    final String INFO_DEFAULT_ICON_PATH = Config.INFO_DEFAULT_ICON_PATH ;
    final String INFO_SELECTED_ICON_PATH = Config.INFO_SELECTED_ICON_PATH ;
35    final String INFO_DISABLED_ICON_PATH = Config.INFO_DISABLED_ICON_PATH ;
    final String EDIT_DEFAULT_ICON_PATH = Config.EDIT_DEFAULT_ICON_PATH ;
    final String EDIT_SELECTED_ICON_PATH = Config.EDIT_SELECTED_ICON_PATH ;
    final String ALTERNATIVE_ANIMATED_ICON_PATH = Config.
        VOICECALL_ALTERNATIVE_ANIMATED_ICON_PATH ;
40

    InfoPanel ip;
    ExtendedButton startButton, endButton, infoButton, settingsButton;
    JLabel numberLabel = new JLabel ();
    JTextField numberField = new JTextField ();
45

    /**
     * Creates a new <code>VoiceCallPanel</code> instance.
     */
    public VoiceCallPanel () {
50    }

    /**
     * Creates a new <code>VoiceCallPanel</code> instance with the specified
     * parent frame and previous panel.
55     *
     * @param myFrame The frame which this panel belongs to
     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
     * wants to return to the previous panel
60     */
    public VoiceCallPanel (JFrame myFrame, JPanel previousPanel) {
        super (myFrame, previousPanel );

        GridLayout1.setColumns (1);
65        GridLayout1.setRows (4);

        startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
            START_SELECTED_ICON_PATH );
        startButton.setDisabledIcon (new ImageIcon (START_DISABLED_ICON_PATH ));
70        endButton = new ExtendedButton (END_DEFAULT_ICON_PATH ,

```

```

                                END_SELECTED_ICON_PATH );
endButton.setDisabledIcon (new ImageIcon (END_DISABLED_ICON_PATH ));
infoButton = new ExtendedButton (INFO_DEFAULT_ICON_PATH ,
                                INFO_SELECTED_ICON_PATH );
75  infoButton.setDisabledIcon (new ImageIcon (INFO_DISABLED_ICON_PATH ));
settingsButton = new ExtendedButton (EDIT_DEFAULT_ICON_PATH ,
                                    EDIT_SELECTED_ICON_PATH );

startButton .
80  addActionListener (new VoiceCallPanel_startButton_actionAdapter (this));
endButton .
    addActionListener (new VoiceCallPanel_endButton_actionAdapter (this));
infoButton .
    addActionListener (new VoiceCallPanel_infoButton_actionAdapter (this));
85  settingsButton .
    addActionListener (new
                        VoiceCallPanel_settingsButton_actionAdapter (this));

endButton.setEnabled (false);
90  infoButton.setEnabled (false);

northPanel.add (startButton );
northPanel.add (endButton );
northPanel.add (infoButton );
95  northPanel.add (settingsButton );

numberLabel.setText ("Number");
numberLabel.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
numberField.setColumns (12);
100

centerPanel.add (numberLabel );
centerPanel.add (numberField );
}

105  /**
 * Refreshes (updates) the presented connection information concerning this
 * service. The new information is contained in </code>myInfo</code>.
 *
 * @param myInfo The new information that should be presented
 */
110  public void refreshConnectionInfo (diplomatiki.InfoStruct myInfo) {
    ConnectionInfoPanel cip = null;

    //Check if myInfo is null
115  if (myInfo == null) {
        new MessageWindow (myFrame);
        endAction ();
        ( (MenuFrame) myFrame).reConstructApplicationsPanel ();
        if ( ( (MenuFrame) myFrame).getContentPane () == ip) {
120          cip = new ConnectionInfoPanel ("", "", "", "");
        }
        else {
            return;
        }
125  }

    //If not, construct a new ConnectionInfoPanel
    if (cip == null) {
        cip = new ConnectionInfoPanel (myInfo.
130          getAccessTechnology (),
          myInfo.getProviderName (), myInfo.getQos (),
          myInfo.getIpAddress ());
    }

135  //Add the ConnectionInfoPanel to the InfoPanel
    ip.centerPanel.removeAll ();
    ip.centerPanel.add (cip);

    //Repaint the InfoPanel
140  ip.setVisible (false);

```

```

        ip.setVisible (true);
        ip.backButton.grabFocus ();
    }

145    /**
        * Action performed when the "Start" service button is pressed.
        */
    private void startAction () {
        //Check if the address/number field is filled in
150    if (numberField.getText ().equals ("")) {
            new MessageWindow (myFrame, "Input Required:", "Please type a number" );
            return;
        }

155    //Run IIS
        diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
            executeIIS (SERVICE_CODE, true, -1);

        //Check if myInfo is null
160    if (myInfo == null) {
            new MessageWindow (myFrame);
            endAction ();
            return;
        }

165    //Construct ConnectionInfoPanel
        ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
            getAccessTechnology (),
            myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());

170    //Construct InfoPanel
        ip = new InfoPanel (myFrame, this, "Call to: ", numberField.getText ());

        //Optionally, change the animated icon in the InfoPanel
175    ip.changeAnimatedIcon (ALTERNATIVE_ANIMATED_ICON_PATH );

        //Add the ConnectionInfoPanel to the InfoPanel
        ip.centerPanel.add (cip);

180    //Disable the Start Service button
        startButton.setEnabled (false);
        //Enable the End Service and View Info buttons
        endButton.setEnabled (true);
        infoButton.setEnabled (true);

185    //Show the InfoPanel
        this.setVisible (false);
        myFrame.setContentPane (ip);
        myFrame.setVisible (false);
190    myFrame.setVisible (true);
        ip.setVisible (true);
    }

    /**
195    * Action performed when the "End" service button is pressed.
        */
    private void endAction () {
        ( (MenuFrame) myFrame).getMobilityManager ().removeFromRunningServices (
            SERVICE_CODE );
200    //Enable the Start Service button
        startButton.setEnabled (true);
        //Disable the End Service and View Info buttons
        endButton.setEnabled (false);
        infoButton.setEnabled (false);

205    //Give focus to the Start Service button
        startButton.grabFocus ();
    }

210    /**

```

```
    * Action performed when the "View Info" button is pressed.
    */
    private void infoAction () {
        //Show the InfoPanel
215     if (ip != null) {
            this.setVisible (false);
            myFrame.setContentPane (ip);
            myFrame.setVisible (false);
            myFrame.setVisible (true);
220     ip.setVisible (true);
        }
    }

    /**
225     * Action performed when the "Edit Settings" button is pressed.
    */
    private void settingsAction () {
        //Construct ProfilePanel
        ProfilePanel pp = new ProfilePanel (myFrame, this, 1);
230     //Disable all other sliders in the ProfilePanel
        pp.disableAllSlidersExceptOne (0);

        //Show the ProfilePanel
        this.setVisible (false);
235     myFrame.setContentPane (pp);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
        pp.setVisible (true);
    }

240     /**
    * Action performed when the "Start" service button is pressed. The
    * <code>startAction()</code> method is invoked.
    *
245     * @param e ActionEvent
    */
    void startButton_actionPerformed (ActionEvent e) {
        startAction ();
    }

250     /**
    * Action performed when the "End" service button is pressed. The
    * <code>endAction()</code> method is invoked.
    *
255     * @param e ActionEvent
    */
    void endButton_actionPerformed (ActionEvent e) {
        endAction ();
    }

260     /**
    * Action performed when the "View Info" button is pressed. The
    * <code>infoAction()</code> method is invoked.
    *
265     * @param e ActionEvent
    */
    void infoButton_actionPerformed (ActionEvent e) {
        infoAction ();
    }

270     /**
    * Action performed when the "Edit Settings" button is pressed. The
    * <code>settingsAction()</code> method is invoked.
    *
275     * @param e ActionEvent
    */
    void settingsButton_actionPerformed (ActionEvent e) {
        settingsAction ();
    }

280
```

```
    }

    class VoiceCallPanel_startButton_actionAdapter
        implements java.awt.event.ActionListener {
285     VoiceCallPanel adaptee;

        VoiceCallPanel_startButton_actionAdapter (VoiceCallPanel adaptee) {
            this.adaptee = adaptee;
        }
290     public void actionPerformed (ActionEvent e) {
        adaptee.startButton_actionPerformed (e);
    }
}

295 class VoiceCallPanel_endButton_actionAdapter
    implements java.awt.event.ActionListener {
        VoiceCallPanel adaptee;

300     VoiceCallPanel_endButton_actionAdapter (VoiceCallPanel adaptee) {
        this.adaptee = adaptee;
    }

        public void actionPerformed (ActionEvent e) {
305     adaptee.endButton_actionPerformed (e);
    }
}

class VoiceCallPanel_infoButton_actionAdapter
310     implements java.awt.event.ActionListener {
        VoiceCallPanel adaptee;

        VoiceCallPanel_infoButton_actionAdapter (VoiceCallPanel adaptee) {
            this.adaptee = adaptee;
315     }

        public void actionPerformed (ActionEvent e) {
            adaptee.infoButton_actionPerformed (e);
        }
320 }

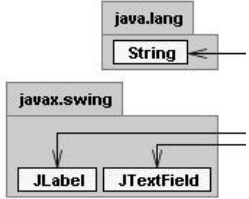
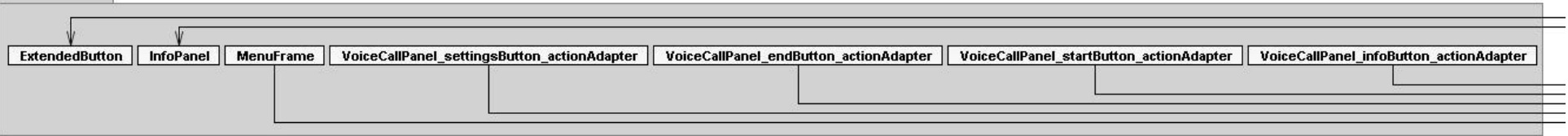
class VoiceCallPanel_settingsButton_actionAdapter
    implements java.awt.event.ActionListener {
        VoiceCallPanel adaptee;

325     VoiceCallPanel_settingsButton_actionAdapter (VoiceCallPanel adaptee) {
        this.adaptee = adaptee;
    }

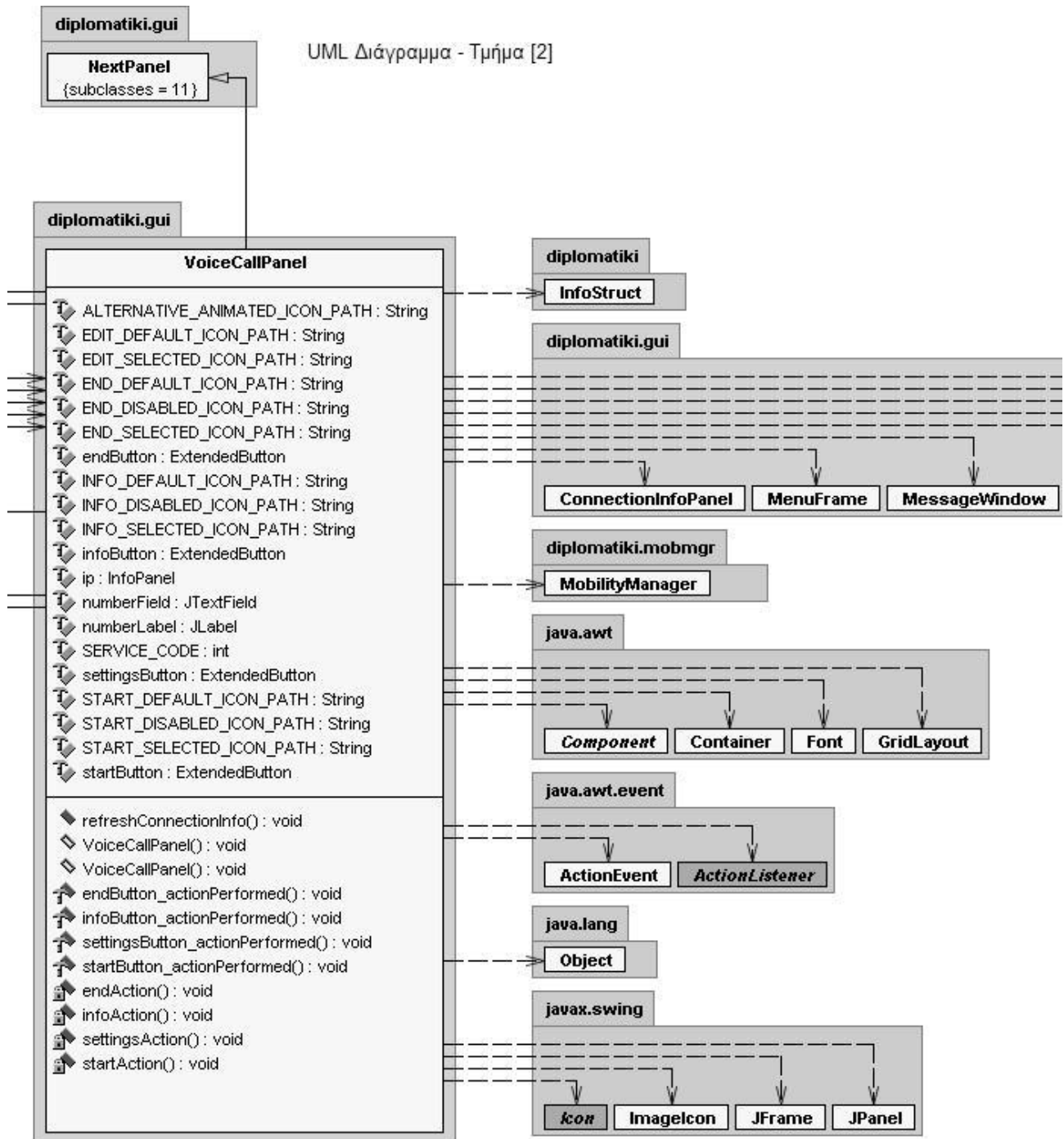
330     public void actionPerformed (ActionEvent e) {
        adaptee.settingsButton_actionPerformed (e);
    }
}
```

diplomatiki.gui

UML Διάγραμμα - Τμήμα [1]

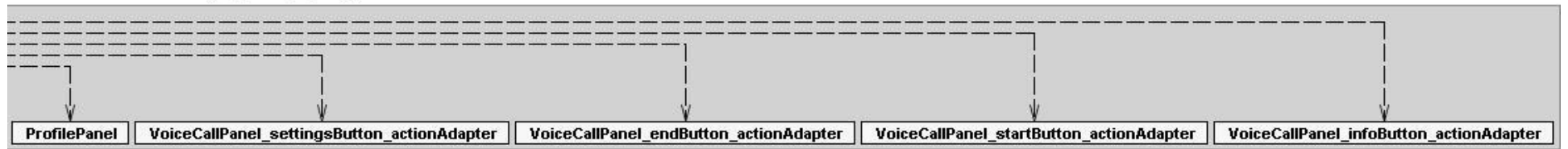


UML Διάγραμμα - Τμήμα [2]





UML Διάγραμμα - Τμήμα [3]



```

package diplomatiki .gui;

import java.awt.event.*;
import javax.swing.*;
5 import diplomatiki .Config;

/**
 * <p>Title: Term Paper - <code>WebBrowsingPanel</code> class</p>
 *
10 * <p>Description: This is the Web Browsing Service panel. This panel is shown
 * when the user chooses to browse the web.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 */
20 public class WebBrowsingPanel
    extends JPanel {
    final int SERVICE_CODE = Config.WEBBROWSING_SERVICE_CODE ;

    final String START_DEFAULT_ICON_PATH = Config.
25     WEBBROWSING_START_DEFAULT_ICON_PATH ;
    final String START_SELECTED_ICON_PATH = Config.
        WEBBROWSING_START_SELECTED_ICON_PATH ;
    final String START_DISABLED_ICON_PATH = Config.
        WEBBROWSING_START_DISABLED_ICON_PATH ;
30    final String END_DEFAULT_ICON_PATH = Config.
        WEBBROWSING_END_DEFAULT_ICON_PATH ;
    final String END_SELECTED_ICON_PATH = Config.
        WEBBROWSING_END_SELECTED_ICON_PATH ;
    final String END_DISABLED_ICON_PATH = Config.
35     WEBBROWSING_END_DISABLED_ICON_PATH ;
    final String INFO_DEFAULT_ICON_PATH = Config.INFO_DEFAULT_ICON_PATH ;
    final String INFO_SELECTED_ICON_PATH = Config.INFO_SELECTED_ICON_PATH ;
    final String INFO_DISABLED_ICON_PATH = Config.INFO_DISABLED_ICON_PATH ;
    final String EDIT_DEFAULT_ICON_PATH = Config.EDIT_DEFAULT_ICON_PATH ;
40    final String EDIT_SELECTED_ICON_PATH = Config.EDIT_SELECTED_ICON_PATH ;

    InfoPanel ip;
    ExtendedButton startButton, endButton, infoButton, settingsButton;
    JLabel urlLabel = new JLabel ();
45    JTextField urlField = new JTextField ();

    /**
     * Creates a new <code>WebBrowsingPanel</code> instance.
     */
50    public WebBrowsingPanel () {
    }

    /**
     * Creates a new <code>WebBrowsingPanel</code> instance with the specified
55     * parent frame and previous panel.
     *
     * @param myFrame The frame which this panel belongs to
     * @param previousPanel The previous panel that was visible in the
     * <code>myFrame</code> frame. This reference is very useful if the user
60     * wants to return to the previous panel
     */
    public WebBrowsingPanel (JFrame myFrame, JPanel previousPanel) {
        super (myFrame, previousPanel);

65        GridLayout1.setColumns (1);
        GridLayout1.setRows (4);

        startButton = new ExtendedButton (START_DEFAULT_ICON_PATH ,
70         START_SELECTED_ICON_PATH );
        startButton.setDisabledIcon (new ImageIcon (START_DISABLED_ICON_PATH ));

```

```

        endButton = new ExtendedButton (END_DEFAULT_ICON_PATH ,
                                        END_SELECTED_ICON_PATH );
        endButton.setDisabledIcon (new ImageIcon (END_DISABLED_ICON_PATH ));
        infoButton = new ExtendedButton (INFO_DEFAULT_ICON_PATH ,
75         INFO_SELECTED_ICON_PATH );
        infoButton.setDisabledIcon (new ImageIcon (INFO_DISABLED_ICON_PATH ));
        settingsButton = new ExtendedButton (EDIT_DEFAULT_ICON_PATH ,
                                        EDIT_SELECTED_ICON_PATH );

80         startButton .
            addActionListener (new WebBrowsingPanel_startButton_actionAdapter (this));
        endButton .
            addActionListener (new WebBrowsingPanel_endButton_actionAdapter (this));
        infoButton .
85         addActionListener (new WebBrowsingPanel_infoButton_actionAdapter (this));
        settingsButton .
            addActionListener (new
                                WebBrowsingPanel_settingsButton_actionAdapter (this));

90         endButton.setEnabled (false);
        infoButton.setEnabled (false);

        northPanel.add (startButton);
        northPanel.add (endButton);
95         northPanel.add (infoButton);
        northPanel.add (settingsButton);

        urlLabel.setText ("URL");
        urlLabel.setFont (new java.awt.Font ("Microsoft Sans Serif" , 1, 11));
100         urlField.setColumns (20);

        centerPanel.add (urlLabel);
        centerPanel.add (urlField);
    }

105 /**
 * Refreshes (updates) the presented connection information concerning this
 * service. The new information is contained in </code>myInfo</code>.
 *
110 * @param myInfo The new information that should be presented
 */
public void refreshConnectionInfo (diplomatiiki.InfoStruct myInfo) {
    ConnectionInfoPanel cip = null;

115     //Check if myInfo is null
    if (myInfo == null) {
        new MessageWindow (myFrame);
        endAction ();
        ( (MenuFrame) myFrame).reConstructApplicationsPanel ();
120         if ( ( (MenuFrame) myFrame).getContentPane () == ip) {
            cip = new ConnectionInfoPanel ("", "", "", "");
        }
        else {
125             return;
        }
    }

    //If not, construct a new ConnectionInfoPanel
    if (cip == null) {
130         cip = new ConnectionInfoPanel (myInfo .
                                        getAccessTechnology (),
                                        myInfo.getProviderName (), myInfo.getQos (),
                                        myInfo.getIpAddress ());
    }

135     //Add the ConnectionInfoPanel to the InfoPanel
    ip.centerPanel.removeAll ();
    ip.centerPanel.add (cip);

140     //Repaint the InfoPanel

```

```

        ip.setVisible (false);
        ip.setVisible (true);
        ip.backButton.grabFocus ();
    }
145
    /**
     * Action performed when the "Start" service button is pressed.
     */
    private void startAction () {
150        //Check if the address/number field is filled in
        if (urlField.getText ().equals ("")) {
            new MessageWindow (myFrame, "Input Required:", "Please type a url" );
            return;
        }
155
        //Run IIS
        diplomatiki.InfoStruct myInfo = ( (MenuFrame) myFrame).getMobilityManager ().
            executeIIS (SERVICE_CODE, true, -1);

160        //Check if myInfo is null
        if (myInfo == null) {
            new MessageWindow (myFrame);
            endAction ();
            return;
165        }

        //Construct ConnectionInfoPanel
        ConnectionInfoPanel cip = new ConnectionInfoPanel (myInfo.
            getAccessTechnology (),
170            myInfo.getProviderName (), myInfo.getQos (), myInfo.getIpAddress ());

        //Construct InfoPanel
        ip = new InfoPanel (myFrame, this, "Browsing URL: ", urlField.getText ());

175        //Add the ConnectionInfoPanel to the InfoPanel
        ip.centerPanel.add (cip);

        //Disable the Start Service button
        startButton.setEnabled (false);
180        //Enable the End Service and View Info buttons
        endButton.setEnabled (true);
        infoButton.setEnabled (true);

        //Show the InfoPanel
185        this.setVisible (false);
        myFrame.setContentPane (ip);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
        ip.setVisible (true);
190    }

    /**
     * Action performed when the "End" service button is pressed.
     */
195    private void endAction () {
        ( (MenuFrame) myFrame).getMobilityManager ().removeFromRunningServices (
            SERVICE_CODE);
        //Enable the Start Service button
        startButton.setEnabled (true);
200        //Disable the End Service and View Info buttons
        endButton.setEnabled (false);
        infoButton.setEnabled (false);

        //Give focus to the Start Service button
205        startButton.grabFocus ();
    }

    /**
     * Action performed when the "View Info" button is pressed.
210    */

```

```
private void infoAction () {
    //Show the InfoPanel
    if (ip != null) {
        this.setVisible (false);
215     myFrame.setContentPane (ip);
        myFrame.setVisible (false);
        myFrame.setVisible (true);
        ip.setVisible (true);
    }
220 }

/**
 * Action performed when the "Edit Settings" button is pressed.
 */
225 private void settingsAction () {
    //Construct ProfilePanel
    ProfilePanel pp = new ProfilePanel (myFrame, this, 1);
    //Disable all other sliders in the ProfilePanel
    pp.disableAllSlidersExceptOne (4);
230
    //Show the ProfilePanel
    this.setVisible (false);
    myFrame.setContentPane (pp);
    myFrame.setVisible (false);
235     myFrame.setVisible (true);
    pp.setVisible (true);
}

/**
240  * Action performed when the "Start" service button is pressed. The
    * <code>startAction()</code> method is invoked.
    *
    * @param e ActionEvent
    */
245 void startButton_actionPerformed (ActionEvent e) {
    startAction ();
}

/**
250  * Action performed when the "End" service button is pressed. The
    * <code>endAction()</code> method is invoked.
    *
    * @param e ActionEvent
    */
255 void endButton_actionPerformed (ActionEvent e) {
    endAction ();
}

/**
260  * Action performed when the "View Info" button is pressed. The
    * <code>infoAction()</code> method is invoked.
    *
    * @param e ActionEvent
    */
265 void infoButton_actionPerformed (ActionEvent e) {
    infoAction ();
}

/**
270  * Action performed when the "Edit Settings" button is pressed. The
    * <code>settingsAction()</code> method is invoked.
    *
    * @param e ActionEvent
    */
275 void settingsButton_actionPerformed (ActionEvent e) {
    settingsAction ();
}
}
280
```

```
class WebBrowsingPanel_startButton_actionAdapter
    implements java.awt.event.ActionListener {
    WebBrowsingPanel adaptee;

285     WebBrowsingPanel_startButton_actionAdapter (WebBrowsingPanel adaptee) {
        this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
290         adaptee.startButton_actionPerformed (e);
    }
}

class WebBrowsingPanel_endButton_actionAdapter
295     implements java.awt.event.ActionListener {
    WebBrowsingPanel adaptee;

    WebBrowsingPanel_endButton_actionAdapter (WebBrowsingPanel adaptee) {
300         this.adaptee = adaptee;
    }

    public void actionPerformed (ActionEvent e) {
        adaptee.endButton_actionPerformed (e);
    }
305 }

class WebBrowsingPanel_infoButton_actionAdapter
    implements java.awt.event.ActionListener {
    WebBrowsingPanel adaptee;

310     WebBrowsingPanel_infoButton_actionAdapter (WebBrowsingPanel adaptee) {
        this.adaptee = adaptee;
    }

315     public void actionPerformed (ActionEvent e) {
        adaptee.infoButton_actionPerformed (e);
    }
}

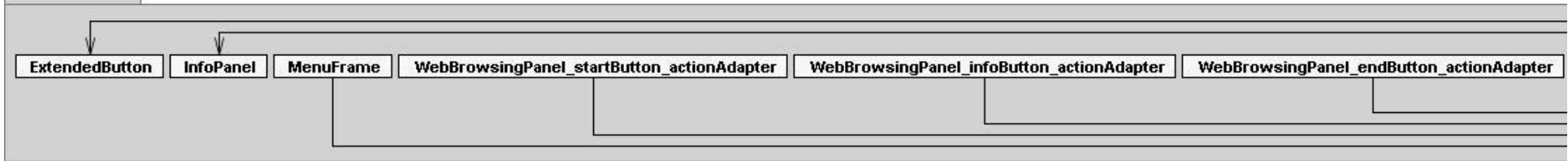
320 class WebBrowsingPanel_settingsButton_actionAdapter
    implements java.awt.event.ActionListener {
    WebBrowsingPanel adaptee;

    WebBrowsingPanel_settingsButton_actionAdapter (WebBrowsingPanel adaptee) {
325         this.adaptee = adaptee;
    }

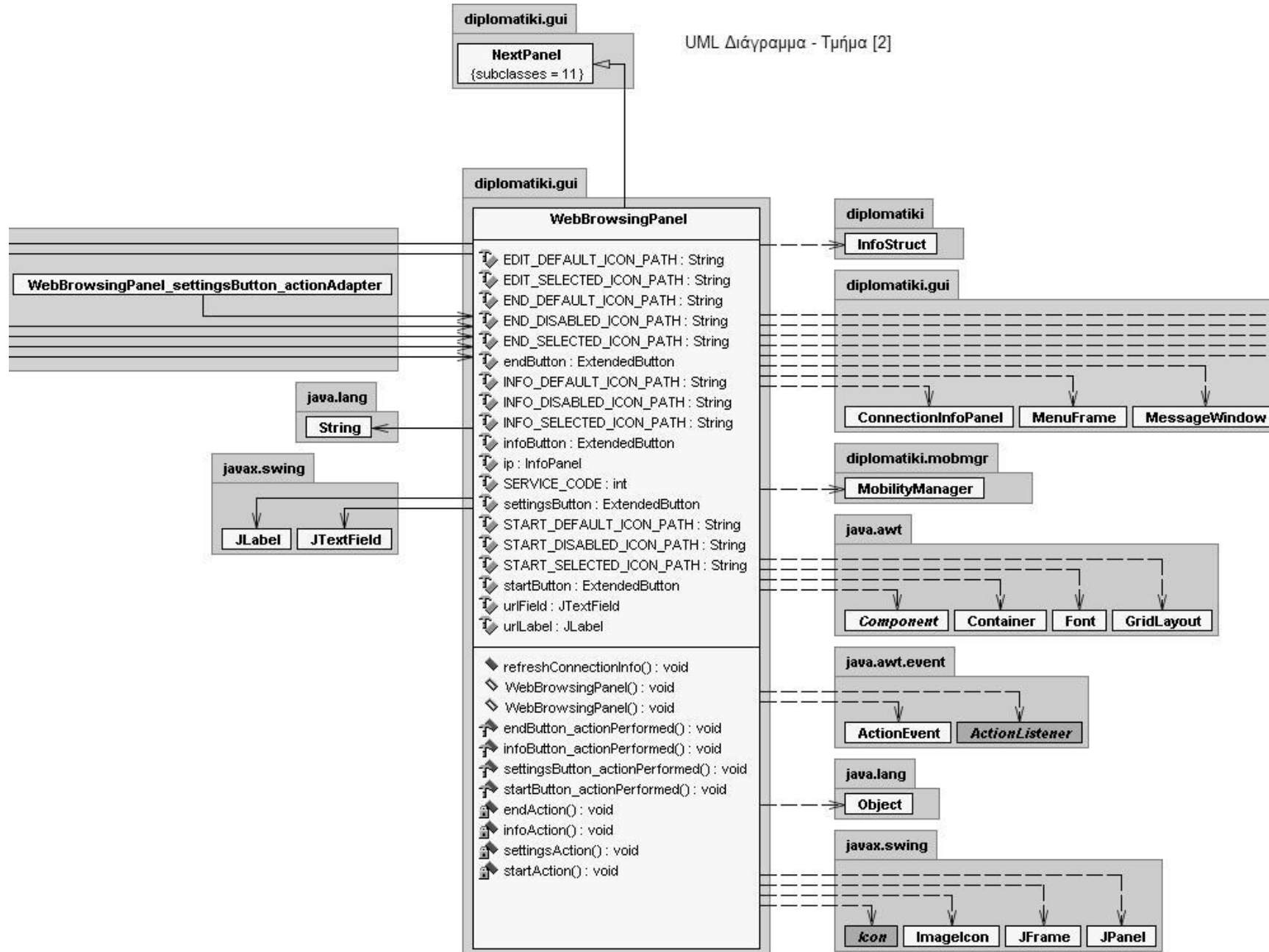
    public void actionPerformed (ActionEvent e) {
        adaptee.settingsButton_actionPerformed (e);
330     }
}
```

diplomatiki.gui

UML Διάγραμμα - Τμήμα [1]

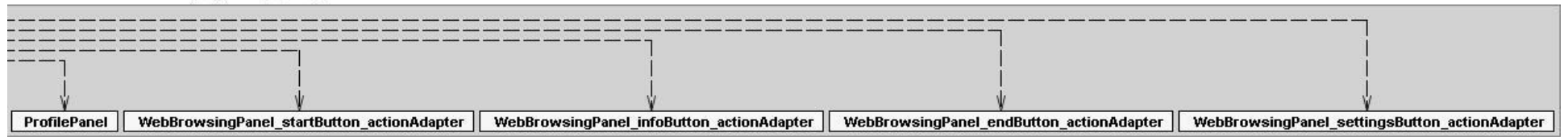


UML Διάγραμμα - Τμήμα [2]





UML Διάγραμμα - Τμήμα [3]



```
package diplomatiki.services;

import java.io.*;
import java.util.*;
5 import javax.mail.*;
import javax.mail.internet.*;

import diplomatiki.*;

10 /**
 * <p>Title: Term Paper - <code>Emailer</code> class</p>
 *
 * <p>Description: This class implements the Email service.</p>
 *
15 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
20 * @version 1.0
 * @see <code>EmailerService</code>
 */
public class Emailer {
    private final String EMAIL_CONFIG_FILE = Config.EMAIL_CONFIG_FILE ;

25     private Properties fMailServerConfig = null;

    protected String recipient = "";
    protected String subject = "";
30     protected String body = "";

    /**
     * Creates a new <code>Emailer</code> instance.
     */
35     public Emailer () {
    }

    /**
     * Creates a new <code>Emailer</code> instance with the specified Recipient,
40     * Subject and Body as attributes.
     *
     * @param recipient The recipient of the e-mail
     * @param subject The e-mail's subject
     * @param body The e-mail's message body
45     */
    public Emailer (String recipient, String subject, String body) {
        this.recipient = recipient;
        this.subject = subject;
        this.body = body;
50     }

    /**
     * Sends a simple e-mail to the specified recipient. The message sent has the
55     * specified subject and body.
     *
     * @param aToEmailAddr The recipient's e-mail address
     * @param aSubject The e-mail's subject
     * @param aBody The message's body
     */
60     public void sendEmail (String aToEmailAddr,
                             String aSubject,
                             String aBody) {
        fMailServerConfig = loadPropertiesFile (EMAIL_CONFIG_FILE );

65         Session session = Session.getDefaultInstance (fMailServerConfig, null);

        MimeMessage message = new MimeMessage (session);

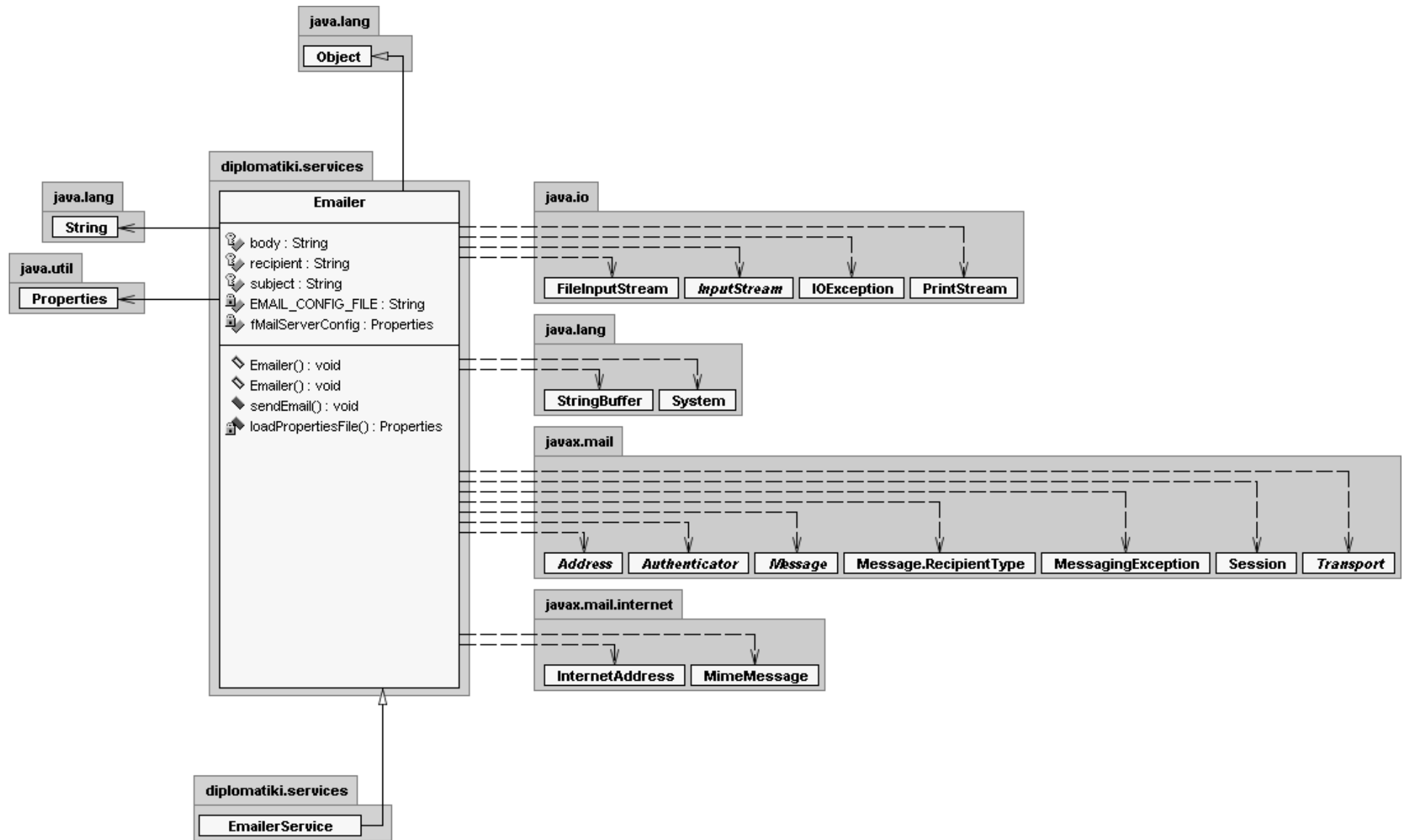
        try {
70             //the "from" address may be set in code, or set in the
```

```
//config file under "mail.from" ; here, the latter style is used
//message.setFrom( new InternetAddress(aFromEmailAddr) );
message.addRecipient (Message.RecipientType.TO,
    new InternetAddress (aToEmailAddr));
75 message.setSubject (aSubject);
message.setText (aBody);

    Transport.send(message);
}
80 catch (MessagingException ex) {
    System.err.println("Cannot send email. " + ex);
}
}

85 /**
 * Loads a Properties file.
 *
 * @param filePath The path of the properties file
 * @return A <code>Properties</code> instance that represents the properties
90 * file's contents
 */
private Properties loadPropertiesFile (String filePath) {
    Properties config = new Properties ();
    InputStream input = null;
95    try {
        input = new FileInputStream (filePath);
        config.load(input);
    }
    catch (IOException ex) {
100        System.err.println("Cannot open and load properties file." );
    }
    finally {
        try {
            try {
105                if (input != null) {
                    input.close();
                }
            }
            catch (IOException ex) {
110                System.err.println("Cannot close properties file." );
            }
        }
    }

    return config;
}
115 }
```



```
package diplomatiki.services;

/**
 * <p>Title: Term Paper - <code>EmailerService</code> class</p>
5  *
 * <p>Description: This is a subclass of the <code>Emailer</code> class. An
 * instance of this class runs as a separate thread and handles the process of
 * sending an e-mail.</p>
 *
10 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
15 * @version 1.0
 * @see <code>Emailer</code>
 */
public class EmailerService
    extends Emailer
20     implements Runnable {
    Thread runner = null;

    /**
 * Creates a new <code>EmailerService</code> instance.
 */
25     public EmailerService () {
    }

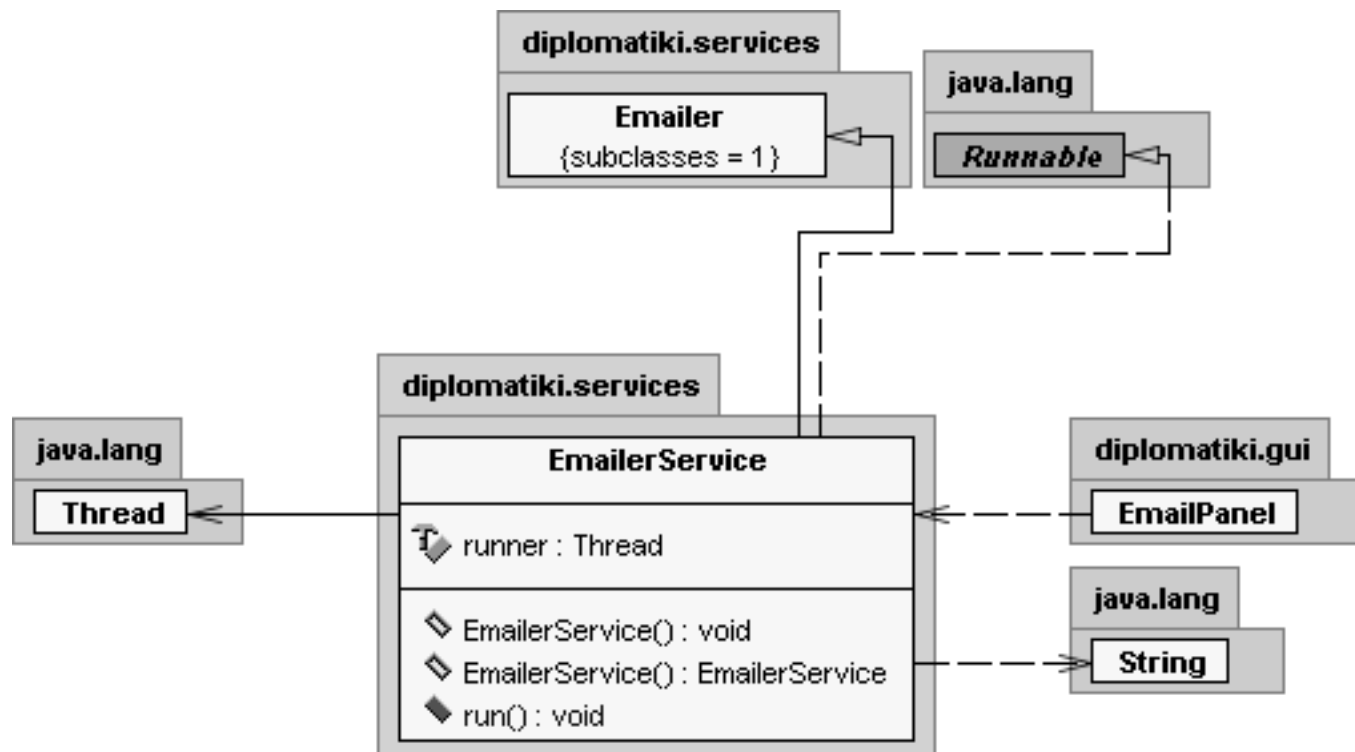
    /**
30 * Creates a new <code>EmailerService</code> instance with the specified
 * Recipient, Subject and Body as attributes. The thread is also started.
 *
 * @param recipient The recipient of the e-mail
 * @param subject The e-mail's subject
35 * @param body The e-mail's message body
 */
    public EmailerService (String recipient, String subject, String body) {
        super (recipient, subject, body);

40         if (runner == null) {
            runner = new Thread (this);
            runner.start ();
        }
    }

45     /**
 * This is the entry point for the <code>EmailerService</code> thread. It
 * sends an e-mail to each of the recipients defined in the
 * <code>Recipient</code> inherited instance variable, by using the
50 * <code>sendEmail()</code> method.
 */
    public void run () {
        String [] recs = recipient.split ("");

55         //Send the email to each recipient
        for (int i = 0; i < recs.length; i++) {
            sendEmail (recs[i], subject, body);
        }
    }

60 }
}
```



```

package networkside .network ;

import java.io.*;
import java.net.*;

5
/**
 * <p>Title: Term Paper - <code>NetworkEntity</code> class</p>
 *
 * <p>Description: This is the main class implementing the Network Management
10 * System.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
 * <p>Company: Jadacode</p>
15 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 * @see <code>NetworkEntityService</code>
 */
20 public class NetworkEntity
    implements Runnable {
    private final int NETWORK_ENTITY_PORT = 4413;
    private ServerSocket ss;
    private Thread runner;

25
    /**
     * Creates a new <code>NetworkEntity</code> instance. This instance runs in a
     * separate thread and listens to a specific port for terminal requests.
     */
30 public NetworkEntity () {
    try {
        ss = new ServerSocket (NETWORK_ENTITY_PORT );
    }
    catch (IOException ex) {
35 ex.printStackTrace ();
    }
    if (runner == null) {
        runner = new Thread (this);
    }
40 runner.start ();
}

/**
 * Entry point for the <code>NetworkEntity</code> thread. When a new terminal
45 * request is received, a new <code>NetworkEntityService</code>
 * instance/thread is constructed.
 */
public void run () {
    Socket client = null;
50 while (true) {
    if (ss == null) {
        return;
    }

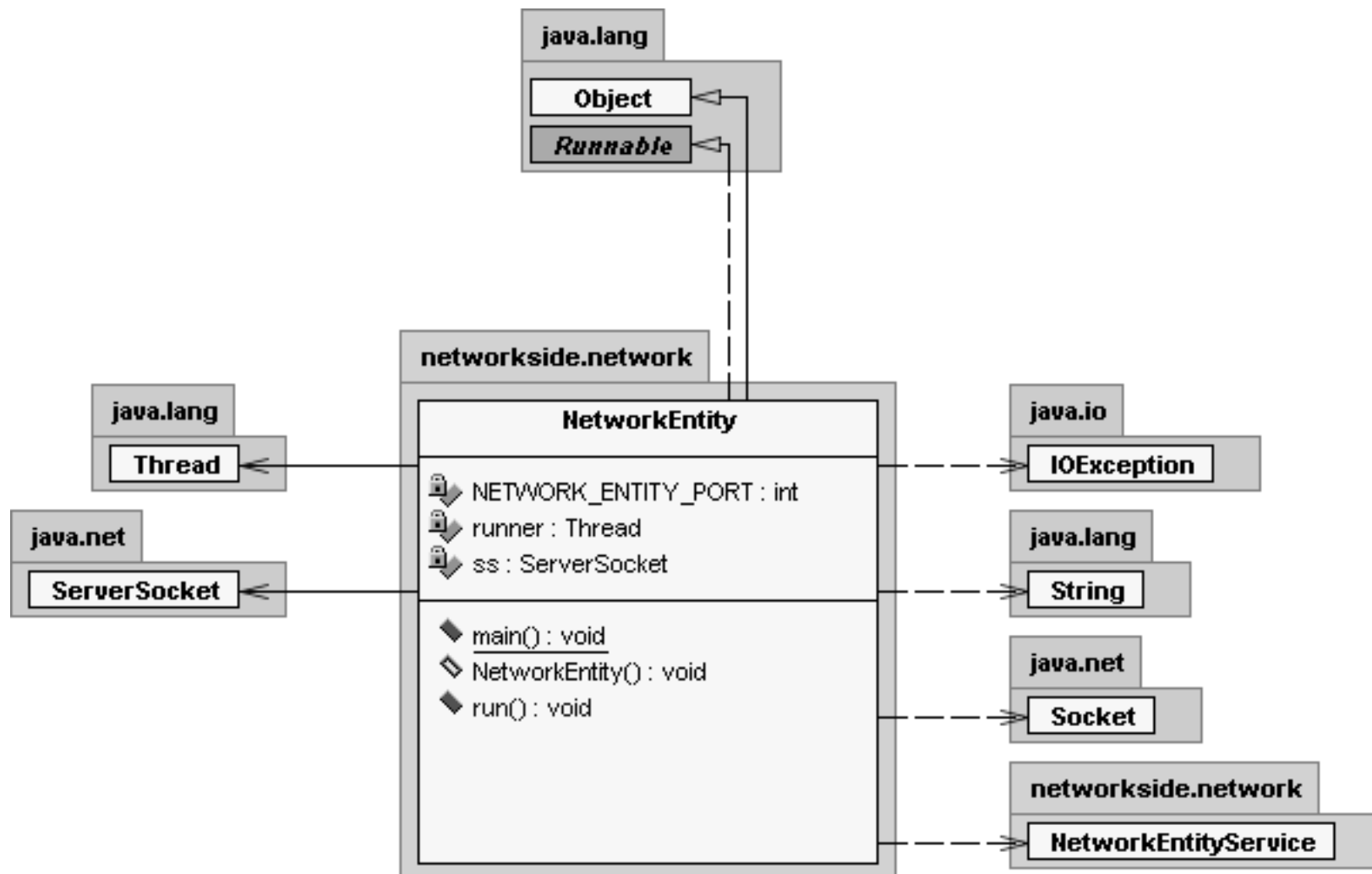
55 try {
        client = ss.accept ();
        NetworkEntityService nes = new NetworkEntityService (client);
        nes.start ();
    }
    catch (IOException ex) {
60 ex.printStackTrace ();
    }
}

65
/**
 * This method initiates the Network Management System.
 *
 * @param arguments No command line arguments are necessary
70 */

```

```
    public static void main(String[] arguments) {  
        new NetworkEntity ();  
    }  
  
75 }
```





```

package networkside .network ;

import java.io.*;
import java.net.*;

5
/**
 * <p>Title: Term Paper - <code>NetworkEntityService</code> class</p>
 *
 * <p>Description: An instance of this class has the responsibility to determine
10 * if a terminal's choice should be accepted or rejected. It runs in a separate
 * thread.</p>
 *
 * <p>Copyright: Copyright (c) 2005</p>
 *
15 * <p>Company: Jadacode</p>
 *
 * @author Evgenia Adamopoulou - Konstantinos Demestichas
 * @version 1.0
 * @see <code>NetworkEntity</code>
20 */
public class NetworkEntityService
    extends Thread {
    private final int RANDOM_MAX_NUMBER = 5;
    private final int REJECTION_DEGREE = 0;
25 //private final String INIT_PATH = "I:/users/cdemest/diplomatiki/databank/";
    private final String INIT_PATH = "networkside/log/" ;
    private final String NET_LOG_FILE = INIT_PATH + "net_log.txt" ;

    private Socket client = null;
30 private PrintWriter fileout = null;

    /**
     * Creates a new <code>NetworkEntityService</code> instance.
     */
35 public NetworkEntityService () {
    }

    /**
     * Creates a new <code>NetworkEntityService</code> instance attached to the
40 * specified client.
     *
     * @param client The socket that should be used to communicate with the client
     */
    public NetworkEntityService (Socket client) {
45     this.client = client;
    }

    /**
     * Entry point for the <code>NetworkEntityService</code> thread. This method
50 * uses a random number generator in order to determine if the terminal's
     * request should be accepted or rejected. The rejection probability can
     * easily be adjusted (e.g. 0%, 20%, etc.). In case of rejection, the
     * <code>NetworkEntityService</code> thread awaits the terminal's next
     * choice/request, and so on.
55 */
    public void run() {
        if (client == null) {
            return;
        }
60
        java.util.Random randomizer = new java.util.Random ();
        try {
            fileout = new
                PrintWriter (new BufferedWriter (new FileWriter (NET_LOG_FILE , true)));
65 fileout.println ("# " +
                    new java.util.Date (System.currentTimeMillis ()).
                        toString () +
                        ":");
70
            ObjectInputStream input = new ObjectInputStream (client.getInputStream ());

```

```
BufferedReader in = new BufferedReader (new InputStreamReader (client .
    getInputStream ());
PrintStream out = new PrintStream (client.getOutputStream ());

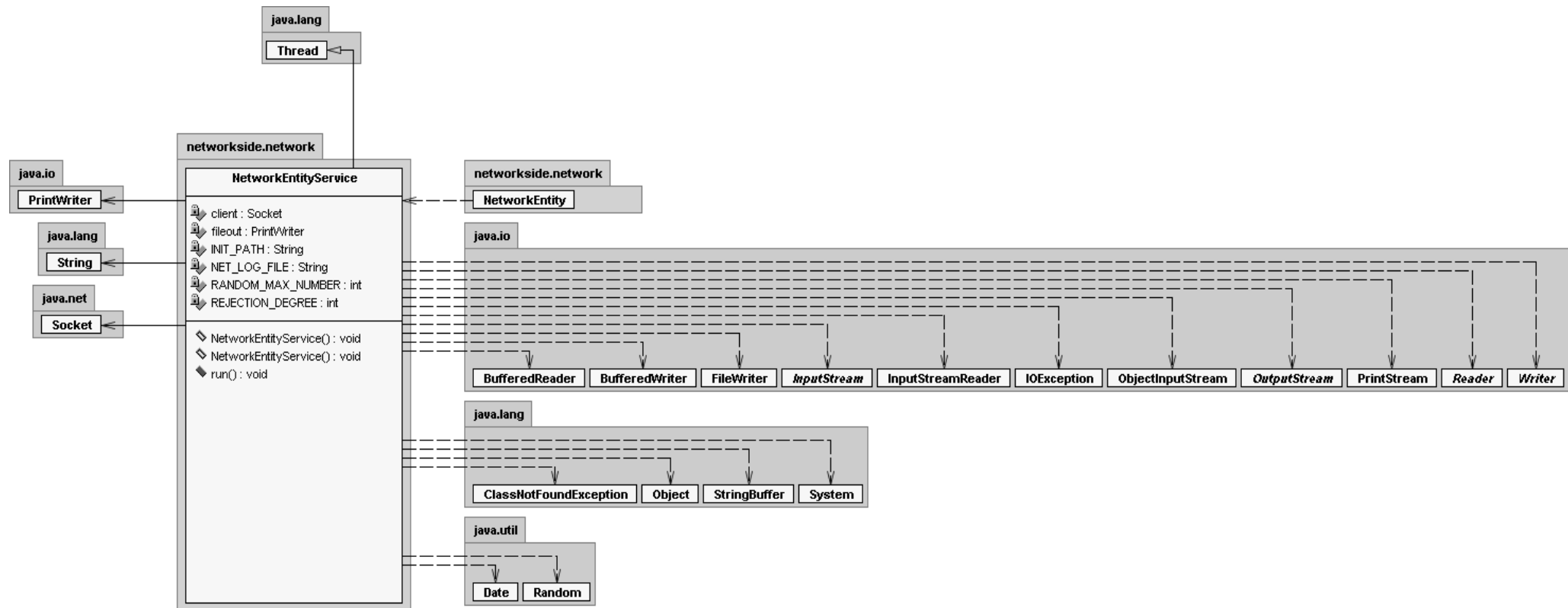
75  while (true) {
    String resultString = (String) input.readObject ();
    int decision = randomizer.nextInt (RANDOM_MAX_NUMBER );
    if (decision > REJECTION_DEGREE ) {
80      System.out.println ("Random Decision Number = " + decision +
        ". Network Entity Service has ACCEPTED the" +
        " terminal's following proposal: " );
        System.out.println (resultString);
        fileout.println ("Random Decision Number = " + decision +
85          ". Network Entity Service has ACCEPTED the" +
            " terminal's following proposal: " );
        fileout.println (resultString);

        out.println ("accepted");
        if (in.readLine ().equals ("end")) {
90            break;
        }
    }
    else {
85      System.out.println ("Random Decision Number = " + decision +
        ". Network Entity Service has REJECTED the" +
        " terminal's following proposal: " );
        System.out.println (resultString);
        fileout.println ("Random Decision Number = " + decision +
100          ". Network Entity Service has REJECTED the" +
            " terminal's following proposal: " );
        fileout.println (resultString);

        out.println ("rejected");
105    }
}
catch (ClassNotFoundException ex) {
    ex.printStackTrace ();
}
110 catch (IOException ex) {
    ex.printStackTrace ();
}

try {
115    fileout.println ();
    fileout.flush ();
    fileout.close ();
    client.close ();
}
120 catch (IOException ex) {
    ex.printStackTrace ();
}
}

125 }
```



## **B. Αρχεία της Τράπεζας Πληροφοριών**

```

<?xml version="1.0"?>
<costs>
  <service code = "1" maxCost="0.032">
    <QoS level = "1">
      <provider code = "1">
        <technology code = "1">0.003</technology>
        <technology code = "2">0.005</technology>
        <technology code = "3">0.002</technology>
        <technology code = "4">0.006</technology>
      </provider>
      <provider code = "2">
        <technology code = "1">0.0032</technology>
        <technology code = "2">0.0055</technology>
        <technology code = "3">0.0016</technology>
        <technology code = "4">0.0057</technology>
      </provider>
      <provider code = "3">
        <technology code = "1">0.0028</technology>
        <technology code = "2">0.0047</technology>
        <technology code = "3">0.0019</technology>
        <technology code = "4">0.0055</technology>
      </provider>
      <provider code = "4">
        <technology code = "1">0.0032</technology>
        <technology code = "2">0.006</technology>
        <technology code = "3">0.0018</technology>
        <technology code = "4">0.0058</technology>
      </provider>
      <provider code = "5">
        <technology code = "1">0.0031</technology>
        <technology code = "2">0.0055</technology>
        <technology code = "3">0.0016</technology>
        <technology code = "4">0.005</technology>
      </provider>
      <provider code = "6">
        <technology code = "1">0.0029</technology>
        <technology code = "2">0.0054</technology>
        <technology code = "3">0.0017</technology>
        <technology code = "4">0.0054</technology>
      </provider>
    </QoS>
    <QoS level = "2">
      <provider code = "1">
        <technology code = "1">0.0055</technology>
        <technology code = "2">0.009</technology>
        <technology code = "3">0.004</technology>
        <technology code = "4">0.01</technology>
      </provider>
      <provider code = "2">
        <technology code = "1">0.006</technology>
        <technology code = "2">0.0098</technology>
        <technology code = "3">0.003</technology>
        <technology code = "4">0.01</technology>
      </provider>
      <provider code = "3">
        <technology code = "1">0.0053</technology>
        <technology code = "2">0.0087</technology>
        <technology code = "3">0.0038</technology>
        <technology code = "4">0.009</technology>
      </provider>
      <provider code = "4">
        <technology code = "1">0.0056</technology>
        <technology code = "2">0.0091</technology>
        <technology code = "3">0.0039</technology>
        <technology code = "4">0.011</technology>
      </provider>
      <provider code = "5">
        <technology code = "1">0.0054</technology>
        <technology code = "2">0.009</technology>
        <technology code = "3">0.0032</technology>
      </provider>
    </QoS>
  </service>
</costs>

```

```
        <technology code = "4">0.0089</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.0053</technology>
        <technology code = "2">0.0086</technology>
        <technology code = "3">0.031</technology>
        <technology code = "4">0.0091</technology>
    </provider>
</QoS>
<QoS level = "3">
    <provider code = "1">
        <technology code = "1">0.01</technology>
        <technology code = "2">0.015</technology>
        <technology code = "3">0.0055</technology>
        <technology code = "4">0.017</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.012</technology>
        <technology code = "2">0.017</technology>
        <technology code = "3">0.0043</technology>
        <technology code = "4">0.018</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.01</technology>
        <technology code = "2">0.011</technology>
        <technology code = "3">0.0055</technology>
        <technology code = "4">0.018</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.011</technology>
        <technology code = "2">0.015</technology>
        <technology code = "3">0.0056</technology>
        <technology code = "4">0.017</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.012</technology>
        <technology code = "2">0.014</technology>
        <technology code = "3">0.0044</technology>
        <technology code = "4">0.017</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.011</technology>
        <technology code = "2">0.013</technology>
        <technology code = "3">0.005</technology>
        <technology code = "4">0.016</technology>
    </provider>
</QoS>
<QoS level = "4">
    <provider code = "1">
        <technology code = "1">0.012</technology>
        <technology code = "2">0.02</technology>
        <technology code = "3">0.0075</technology>
        <technology code = "4">0.024</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.014</technology>
        <technology code = "2">0.02</technology>
        <technology code = "3">0.006</technology>
        <technology code = "4">0.024</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.026</technology>
        <technology code = "3">0.008</technology>
        <technology code = "4">0.027</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.014</technology>
        <technology code = "2">0.019</technology>
        <technology code = "3">0.008</technology>
```

```
        <technology code = "4">0.023</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.016</technology>
        <technology code = "2">0.017</technology>
        <technology code = "3">0.007</technology>
        <technology code = "4">0.023</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.014</technology>
        <technology code = "2">0.02</technology>
        <technology code = "3">0.0065</technology>
        <technology code = "4">0.025</technology>
    </provider>
</QoS>
<QoS level = "5">
    <provider code = "1">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.025</technology>
        <technology code = "3">0.01</technology>
        <technology code = "4">0.03</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.024</technology>
        <technology code = "3">0.009</technology>
        <technology code = "4">0.032</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.019</technology>
        <technology code = "2">0.03</technology>
        <technology code = "3">0.013</technology>
        <technology code = "4">0.032</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.017</technology>
        <technology code = "2">0.023</technology>
        <technology code = "3">0.013</technology>
        <technology code = "4">0.028</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.02</technology>
        <technology code = "2">0.022</technology>
        <technology code = "3">0.011</technology>
        <technology code = "4">0.031</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.018</technology>
        <technology code = "2">0.021</technology>
        <technology code = "3">0.01</technology>
        <technology code = "4">0.029</technology>
    </provider>
</QoS>
</service>
<service code = "2" maxCost="0.13">
    <QoS level = "1">
        <provider code = "1">
            <technology code = "1">0.09</technology>
            <technology code = "2">0.09</technology>
            <technology code = "3">0.013</technology>
            <technology code = "4">0.1</technology>
        </provider>
        <provider code = "2">
            <technology code = "1">0.085</technology>
            <technology code = "2">0.085</technology>
            <technology code = "3">0.015</technology>
            <technology code = "4">0.11</technology>
        </provider>
        <provider code = "3">
            <technology code = "1">0.07</technology>
```



```
<technology code = "2">0.07</technology>
<technology code = "3">0.012</technology>
<technology code = "4">0.12</technology>
</provider>
<provider code = "4">
<technology code = "1">0.075</technology>
<technology code = "2">0.075</technology>
<technology code = "3">0.013</technology>
<technology code = "4">0.1</technology>
</provider>
<provider code = "5">
<technology code = "1">0.08</technology>
<technology code = "2">0.08</technology>
<technology code = "3">0.017</technology>
<technology code = "4">0.11</technology>
</provider>
<provider code = "6">
<technology code = "1">0.085</technology>
<technology code = "2">0.085</technology>
<technology code = "3">0.014</technology>
<technology code = "4">0.13</technology>
</provider>
</QoS>
</service>
<service code = "3" maxCost="0.51">
<QoS level = "1">
<provider code = "1">
<technology code = "1">0.25</technology>
<technology code = "2">0.26</technology>
<technology code = "3">0.1</technology>
<technology code = "4">0.27</technology>
</provider>
<provider code = "2">
<technology code = "1">0.24</technology>
<technology code = "2">0.25</technology>
<technology code = "3">0.1</technology>
<technology code = "4">0.28</technology>
</provider>
<provider code = "3">
<technology code = "1">0.2</technology>
<technology code = "2">0.2</technology>
<technology code = "3">0.11</technology>
<technology code = "4">0.26</technology>
</provider>
<provider code = "4">
<technology code = "1">0.25</technology>
<technology code = "2">0.26</technology>
<technology code = "3">0.12</technology>
<technology code = "4">0.27</technology>
</provider>
<provider code = "5">
<technology code = "1">0.28</technology>
<technology code = "2">0.28</technology>
<technology code = "3">0.12</technology>
<technology code = "4">0.27</technology>
</provider>
<provider code = "6">
<technology code = "1">0.24</technology>
<technology code = "2">0.25</technology>
<technology code = "3">0.12</technology>
<technology code = "4">0.26</technology>
</provider>
</QoS>
<QoS level = "2">
<provider code = "1">
<technology code = "1">0.4</technology>
<technology code = "2">0.4</technology>
<technology code = "3">0.12</technology>
<technology code = "4">0.42</technology>
</provider>
```

```
<provider code = "2">
  <technology code = "1">0.41</technology>
  <technology code = "2">0.41</technology>
  <technology code = "3">0.13</technology>
  <technology code = "4">0.43</technology>
</provider>
<provider code = "3">
  <technology code = "1">0.49</technology>
  <technology code = "2">0.51</technology>
  <technology code = "3">0.13</technology>
  <technology code = "4">0.5</technology>
</provider>
<provider code = "4">
  <technology code = "1">0.4</technology>
  <technology code = "2">0.41</technology>
  <technology code = "3">0.17</technology>
  <technology code = "4">0.42</technology>
</provider>
<provider code = "5">
  <technology code = "1">0.43</technology>
  <technology code = "2">0.43</technology>
  <technology code = "3">0.15</technology>
  <technology code = "4">0.44</technology>
</provider>
<provider code = "6">
  <technology code = "1">0.4</technology>
  <technology code = "2">0.4</technology>
  <technology code = "3">0.16</technology>
  <technology code = "4">0.42</technology>
</provider>
</QoS>
</service>
<service code = "4" maxCost="0.78">
  <QoS level = "1">
    <provider code = "1">
      <technology code = "1">0.58</technology>
      <technology code = "2">0.59</technology>
      <technology code = "3">0.005</technology>
      <technology code = "4">0.7</technology>
    </provider>
    <provider code = "2">
      <technology code = "1">0.59</technology>
      <technology code = "2">0.61</technology>
      <technology code = "3">0.0052</technology>
      <technology code = "4">0.76</technology>
    </provider>
    <provider code = "3">
      <technology code = "1">0.57</technology>
      <technology code = "2">0.59</technology>
      <technology code = "3">0.0048</technology>
      <technology code = "4">0.78</technology>
    </provider>
    <provider code = "4">
      <technology code = "1">0.58</technology>
      <technology code = "2">0.6</technology>
      <technology code = "3">0.005</technology>
      <technology code = "4">0.74</technology>
    </provider>
    <provider code = "5">
      <technology code = "1">0.59</technology>
      <technology code = "2">0.61</technology>
      <technology code = "3">0.0046</technology>
      <technology code = "4">0.69</technology>
    </provider>
    <provider code = "6">
      <technology code = "1">0.6</technology>
      <technology code = "2">0.61</technology>
      <technology code = "3">0.0042</technology>
      <technology code = "4">0.78</technology>
    </provider>
  </QoS>
</service>
</costs>
```

```

</QoS>
</service>
<service code = "5" maxCost="1.19">
  <QoS level = "1">
    <provider code = "1">
      <technology code = "1">0.2</technology>
      <technology code = "2">0.25</technology>
      <technology code = "3">0.15</technology>
      <technology code = "4">0.14</technology>
    </provider>
    <provider code = "2">
      <technology code = "1">0.2</technology>
      <technology code = "2">0.26</technology>
      <technology code = "3">0.16</technology>
      <technology code = "4">0.13</technology>
    </provider>
    <provider code = "3">
      <technology code = "1">0.19</technology>
      <technology code = "2">0.24</technology>
      <technology code = "3">0.16</technology>
      <technology code = "4">0.14</technology>
    </provider>
    <provider code = "4">
      <technology code = "1">0.19</technology>
      <technology code = "2">0.24</technology>
      <technology code = "3">0.14</technology>
      <technology code = "4">0.13</technology>
    </provider>
    <provider code = "5">
      <technology code = "1">0.2</technology>
      <technology code = "2">0.25</technology>
      <technology code = "3">0.15</technology>
      <technology code = "4">0.13</technology>
    </provider>
    <provider code = "6">
      <technology code = "1">0.19</technology>
      <technology code = "2">0.26</technology>
      <technology code = "3">0.16</technology>
      <technology code = "4">0.14</technology>
    </provider>
  </QoS>
  <QoS level = "2">
    <provider code = "1">
      <technology code = "1">0.4</technology>
      <technology code = "2">0.46</technology>
      <technology code = "3">0.3</technology>
      <technology code = "4">0.28</technology>
    </provider>
    <provider code = "2">
      <technology code = "1">0.41</technology>
      <technology code = "2">0.49</technology>
      <technology code = "3">0.31</technology>
      <technology code = "4">0.26</technology>
    </provider>
    <provider code = "3">
      <technology code = "1">0.39</technology>
      <technology code = "2">0.46</technology>
      <technology code = "3">0.3</technology>
      <technology code = "4">0.26</technology>
    </provider>
    <provider code = "4">
      <technology code = "1">0.42</technology>
      <technology code = "2">0.47</technology>
      <technology code = "3">0.32</technology>
      <technology code = "4">0.27</technology>
    </provider>
    <provider code = "5">
      <technology code = "1">0.38</technology>
      <technology code = "2">0.48</technology>
      <technology code = "3">0.28</technology>
    </provider>
  </QoS>
</service>

```

```
        <technology code = "4">0.27</technology >
    </provider >
    <provider code = "6">
        <technology code = "1">0.4</technology >
        <technology code = "2">0.47</technology >
        <technology code = "3">0.29</technology >
        <technology code = "4">0.28</technology >
    </provider >
</QoS>
<QoS level = "3">
    <provider code = "1">
        <technology code = "1">0.6</technology >
        <technology code = "2">0.66</technology >
        <technology code = "3">0.45</technology >
        <technology code = "4">0.44</technology >
    </provider >
    <provider code = "2">
        <technology code = "1">0.63</technology >
        <technology code = "2">0.67</technology >
        <technology code = "3">0.47</technology >
        <technology code = "4">0.42</technology >
    </provider >
    <provider code = "3">
        <technology code = "1">0.6</technology >
        <technology code = "2">0.67</technology >
        <technology code = "3">0.47</technology >
        <technology code = "4">0.44</technology >
    </provider >
    <provider code = "4">
        <technology code = "1">0.59</technology >
        <technology code = "2">0.62</technology >
        <technology code = "3">0.44</technology >
        <technology code = "4">0.43</technology >
    </provider >
    <provider code = "5">
        <technology code = "1">0.59</technology >
        <technology code = "2">0.64</technology >
        <technology code = "3">0.46</technology >
        <technology code = "4">0.45</technology >
    </provider >
    <provider code = "6">
        <technology code = "1">0.61</technology >
        <technology code = "2">0.65</technology >
        <technology code = "3">0.45</technology >
        <technology code = "4">0.43</technology >
    </provider >
</QoS>
<QoS level = "4">
    <provider code = "1">
        <technology code = "1">0.85</technology >
        <technology code = "2">0.92</technology >
        <technology code = "3">0.6</technology >
        <technology code = "4">0.63</technology >
    </provider >
    <provider code = "2">
        <technology code = "1">0.88</technology >
        <technology code = "2">0.93</technology >
        <technology code = "3">0.65</technology >
        <technology code = "4">0.66</technology >
    </provider >
    <provider code = "3">
        <technology code = "1">0.85</technology >
        <technology code = "2">0.92</technology >
        <technology code = "3">0.6</technology >
        <technology code = "4">0.66</technology >
    </provider >
    <provider code = "4">
        <technology code = "1">0.84</technology >
        <technology code = "2">0.91</technology >
        <technology code = "3">0.66</technology >
```

```
                <technology code = "4">0.65</technology >
            </provider >
        <provider code = "5">
            <technology code = "1">0.87</technology >
            <technology code = "2">0.94</technology >
            <technology code = "3">0.59</technology >
            <technology code = "4">0.64</technology >
        </provider >
        <provider code = "6">
            <technology code = "1">0.84</technology >
            <technology code = "2">0.9</technology >
            <technology code = "3">0.62</technology >
            <technology code = "4">0.66</technology >
        </provider >
    </QoS>
    <QoS level = "5">
        <provider code = "1">
            <technology code = "1">1</technology >
            <technology code = "2">1.1</technology >
            <technology code = "3">0.75</technology >
            <technology code = "4">0.9</technology >
        </provider >
        <provider code = "2">
            <technology code = "1">1.14</technology >
            <technology code = "2">1.12</technology >
            <technology code = "3">0.77</technology >
            <technology code = "4">0.85</technology >
        </provider >
        <provider code = "3">
            <technology code = "1">1.16</technology >
            <technology code = "2">1.13</technology >
            <technology code = "3">0.88</technology >
            <technology code = "4">1.07</technology >
        </provider >
        <provider code = "4">
            <technology code = "1">1.02</technology >
            <technology code = "2">1.12</technology >
            <technology code = "3">0.8</technology >
            <technology code = "4">0.95</technology >
        </provider >
        <provider code = "5">
            <technology code = "1">1.1</technology >
            <technology code = "2">1.18</technology >
            <technology code = "3">0.7</technology >
            <technology code = "4">0.96</technology >
        </provider >
        <provider code = "6">
            <technology code = "1">1.15</technology >
            <technology code = "2">1.19</technology >
            <technology code = "3">0.75</technology >
            <technology code = "4">0.99</technology >
        </provider >
    </QoS>
</service >
<service code = "6" maxCost="0.95">
    <QoS level = "1">
        <provider code = "1">
            <technology code = "1">0.15</technology >
            <technology code = "2">0.17</technology >
            <technology code = "3">0.12</technology >
            <technology code = "4">0.09</technology >
        </provider >
        <provider code = "2">
            <technology code = "1">0.14</technology >
            <technology code = "2">0.18</technology >
            <technology code = "3">0.13</technology >
            <technology code = "4">0.1</technology >
        </provider >
        <provider code = "3">
            <technology code = "1">0.16</technology >
```

```
<technology code = "2">0.16</technology>
<technology code = "3">0.11</technology>
<technology code = "4">0.08</technology>
</provider>
<provider code = "4">
  <technology code = "1">0.15</technology>
  <technology code = "2">0.15</technology>
  <technology code = "3">0.12</technology>
  <technology code = "4">0.0085</technology>
</provider>
<provider code = "5">
  <technology code = "1">0.13</technology>
  <technology code = "2">0.16</technology>
  <technology code = "3">0.14</technology>
  <technology code = "4">0.0092</technology>
</provider>
<provider code = "6">
  <technology code = "1">0.14</technology>
  <technology code = "2">0.15</technology>
  <technology code = "3">0.13</technology>
  <technology code = "4">0.0089</technology>
</provider>
</QoS>
<QoS level = "2">
  <provider code = "1">
    <technology code = "1">0.3</technology>
    <technology code = "2">0.33</technology>
    <technology code = "3">0.24</technology>
    <technology code = "4">0.2</technology>
  </provider>
  <provider code = "2">
    <technology code = "1">0.32</technology>
    <technology code = "2">0.32</technology>
    <technology code = "3">0.23</technology>
    <technology code = "4">0.18</technology>
  </provider>
  <provider code = "3">
    <technology code = "1">0.28</technology>
    <technology code = "2">0.35</technology>
    <technology code = "3">0.25</technology>
    <technology code = "4">0.21</technology>
  </provider>
  <provider code = "4">
    <technology code = "1">0.3</technology>
    <technology code = "2">0.35</technology>
    <technology code = "3">0.26</technology>
    <technology code = "4">0.19</technology>
  </provider>
  <provider code = "5">
    <technology code = "1">0.29</technology>
    <technology code = "2">0.34</technology>
    <technology code = "3">0.23</technology>
    <technology code = "4">0.19</technology>
  </provider>
  <provider code = "6">
    <technology code = "1">0.31</technology>
    <technology code = "2">0.33</technology>
    <technology code = "3">0.24</technology>
    <technology code = "4">0.2</technology>
  </provider>
</QoS>
<QoS level = "3">
  <provider code = "1">
    <technology code = "1">0.45</technology>
    <technology code = "2">0.48</technology>
    <technology code = "3">0.36</technology>
    <technology code = "4">0.35</technology>
  </provider>
  <provider code = "2">
    <technology code = "1">0.43</technology>
```

```
<technology code = "2">0.49</technology>
<technology code = "3">0.38</technology>
<technology code = "4">0.36</technology>
</provider>
<provider code = "3">
  <technology code = "1">0.46</technology>
  <technology code = "2">0.47</technology>
  <technology code = "3">0.35</technology>
  <technology code = "4">0.36</technology>
</provider>
<provider code = "4">
  <technology code = "1">0.46</technology>
  <technology code = "2">0.47</technology>
  <technology code = "3">0.37</technology>
  <technology code = "4">0.37</technology>
</provider>
<provider code = "5">
  <technology code = "1">0.45</technology>
  <technology code = "2">0.46</technology>
  <technology code = "3">0.37</technology>
  <technology code = "4">0.35</technology>
</provider>
<provider code = "6">
  <technology code = "1">0.44</technology>
  <technology code = "2">0.47</technology>
  <technology code = "3">0.36</technology>
  <technology code = "4">0.35</technology>
</provider>
</QoS>
<QoS level = "4">
  <provider code = "1">
    <technology code = "1">0.6</technology>
    <technology code = "2">0.64</technology>
    <technology code = "3">0.48</technology>
    <technology code = "4">0.5</technology>
  </provider>
  <provider code = "2">
    <technology code = "1">0.6</technology>
    <technology code = "2">0.63</technology>
    <technology code = "3">0.49</technology>
    <technology code = "4">0.52</technology>
  </provider>
  <provider code = "3">
    <technology code = "1">0.59</technology>
    <technology code = "2">0.65</technology>
    <technology code = "3">0.5</technology>
    <technology code = "4">0.48</technology>
  </provider>
  <provider code = "4">
    <technology code = "1">0.61</technology>
    <technology code = "2">0.66</technology>
    <technology code = "3">0.47</technology>
    <technology code = "4">0.5</technology>
  </provider>
  <provider code = "5">
    <technology code = "1">0.62</technology>
    <technology code = "2">0.64</technology>
    <technology code = "3">0.49</technology>
    <technology code = "4">0.53</technology>
  </provider>
  <provider code = "6">
    <technology code = "1">0.58</technology>
    <technology code = "2">0.62</technology>
    <technology code = "3">0.47</technology>
    <technology code = "4">0.51</technology>
  </provider>
</QoS>
<QoS level = "5">
  <provider code = "1">
    <technology code = "1">0.75</technology>
```

```
<technology code = "2">0.94</technology>
<technology code = "3">0.6</technology>
<technology code = "4">0.7</technology>
</provider>
<provider code = "2">
  <technology code = "1">0.77</technology>
  <technology code = "2">0.94</technology>
  <technology code = "3">0.58</technology>
  <technology code = "4">0.72</technology>
</provider>
<provider code = "3">
  <technology code = "1">0.77</technology>
  <technology code = "2">0.93</technology>
  <technology code = "3">0.61</technology>
  <technology code = "4">0.69</technology>
</provider>
<provider code = "4">
  <technology code = "1">0.76</technology>
  <technology code = "2">0.95</technology>
  <technology code = "3">0.63</technology>
  <technology code = "4">0.71</technology>
</provider>
<provider code = "5">
  <technology code = "1">0.76</technology>
  <technology code = "2">0.92</technology>
  <technology code = "3">0.6</technology>
  <technology code = "4">0.73</technology>
</provider>
<provider code = "6">
  <technology code = "1">0.75</technology>
  <technology code = "2">0.93</technology>
  <technology code = "3">0.62</technology>
  <technology code = "4">0.74</technology>
</provider>
</QoS>
</service>
<service code = "7" maxCost="0.055">
  <QoS level = "1">
    <provider code = "1">
      <technology code = "1">0.008</technology>
      <technology code = "2">0.01</technology>
      <technology code = "3">0.003</technology>
      <technology code = "4">0.01</technology>
    </provider>
    <provider code = "2">
      <technology code = "1">0.009</technology>
      <technology code = "2">0.011</technology>
      <technology code = "3">0.002</technology>
      <technology code = "4">0.0098</technology>
    </provider>
    <provider code = "3">
      <technology code = "1">0.007</technology>
      <technology code = "2">0.009</technology>
      <technology code = "3">0.0025</technology>
      <technology code = "4">0.012</technology>
    </provider>
    <provider code = "4">
      <technology code = "1">0.0085</technology>
      <technology code = "2">0.012</technology>
      <technology code = "3">0.002</technology>
      <technology code = "4">0.008</technology>
    </provider>
    <provider code = "5">
      <technology code = "1">0.009</technology>
      <technology code = "2">0.01</technology>
      <technology code = "3">0.0023</technology>
      <technology code = "4">0.0099</technology>
    </provider>
    <provider code = "6">
      <technology code = "1">0.007</technology>
```



```
        <technology code = "2">0.013</technology>
        <technology code = "3">0.0019</technology>
        <technology code = "4">0.012</technology>
    </provider>
</QoS>
<QoS level = "2">
    <provider code = "1">
        <technology code = "1">0.016</technology>
        <technology code = "2">0.018</technology>
        <technology code = "3">0.006</technology>
        <technology code = "4">0.02</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.017</technology>
        <technology code = "3">0.007</technology>
        <technology code = "4">0.021</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.017</technology>
        <technology code = "2">0.019</technology>
        <technology code = "3">0.0065</technology>
        <technology code = "4">0.021</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.017</technology>
        <technology code = "2">0.02</technology>
        <technology code = "3">0.006</technology>
        <technology code = "4">0.02</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.016</technology>
        <technology code = "2">0.017</technology>
        <technology code = "3">0.0071</technology>
        <technology code = "4">0.022</technology>
    </provider>
    <provider code = "6">
        <technology code = "1">0.015</technology>
        <technology code = "2">0.016</technology>
        <technology code = "3">0.0055</technology>
        <technology code = "4">0.019</technology>
    </provider>
</QoS>
<QoS level = "3">
    <provider code = "1">
        <technology code = "1">0.024</technology>
        <technology code = "2">0.029</technology>
        <technology code = "3">0.009</technology>
        <technology code = "4">0.03</technology>
    </provider>
    <provider code = "2">
        <technology code = "1">0.022</technology>
        <technology code = "2">0.025</technology>
        <technology code = "3">0.009</technology>
        <technology code = "4">0.03</technology>
    </provider>
    <provider code = "3">
        <technology code = "1">0.023</technology>
        <technology code = "2">0.028</technology>
        <technology code = "3">0.0085</technology>
        <technology code = "4">0.032</technology>
    </provider>
    <provider code = "4">
        <technology code = "1">0.025</technology>
        <technology code = "2">0.027</technology>
        <technology code = "3">0.008</technology>
        <technology code = "4">0.029</technology>
    </provider>
    <provider code = "5">
        <technology code = "1">0.025</technology>
```

```
<technology code = "2">0.026</technology>
<technology code = "3">0.009</technology>
<technology code = "4">0.033</technology>
</provider>
<provider code = "6">
  <technology code = "1">0.023</technology>
  <technology code = "2">0.027</technology>
  <technology code = "3">0.0078</technology>
  <technology code = "4">0.031</technology>
</provider>
</QoS>
<QoS level = "4">
  <provider code = "1">
    <technology code = "1">0.032</technology>
    <technology code = "2">0.039</technology>
    <technology code = "3">0.012</technology>
    <technology code = "4">0.04</technology>
  </provider>
  <provider code = "2">
    <technology code = "1">0.034</technology>
    <technology code = "2">0.038</technology>
    <technology code = "3">0.013</technology>
    <technology code = "4">0.042</technology>
  </provider>
  <provider code = "3">
    <technology code = "1">0.032</technology>
    <technology code = "2">0.04</technology>
    <technology code = "3">0.014</technology>
    <technology code = "4">0.043</technology>
  </provider>
  <provider code = "4">
    <technology code = "1">0.031</technology>
    <technology code = "2">0.04</technology>
    <technology code = "3">0.011</technology>
    <technology code = "4">0.039</technology>
  </provider>
  <provider code = "5">
    <technology code = "1">0.033</technology>
    <technology code = "2">0.031</technology>
    <technology code = "3">0.012</technology>
    <technology code = "4">0.044</technology>
  </provider>
  <provider code = "6">
    <technology code = "1">0.032</technology>
    <technology code = "2">0.03</technology>
    <technology code = "3">0.01</technology>
    <technology code = "4">0.039</technology>
  </provider>
</QoS>
<QoS level = "5">
  <provider code = "1">
    <!--<technology code = "1">0.04</technology>-->
    <technology code = "2">0.05</technology>
    <technology code = "3">0.015</technology>
    <technology code = "4">0.05</technology>
  </provider>
  <provider code = "2">
    <!--<technology code = "1">0.038</technology>-->
    <technology code = "2">0.049</technology>
    <technology code = "3">0.016</technology>
    <technology code = "4">0.052</technology>
  </provider>
  <provider code = "3">
    <!--<technology code = "1">0.042</technology>-->
    <technology code = "2">0.051</technology>
    <technology code = "3">0.017</technology>
    <technology code = "4">0.053</technology>
  </provider>
  <provider code = "4">
    <!--<technology code = "1">0.04</technology>-->
```

```
        <technology code = "2">0.048</technology>
        <technology code = "3">0.014</technology>
        <technology code = "4">0.049</technology>
    </provider>
    <provider code = "5">
        <!--<technology code = "1">0.044</technology>-->
        <technology code = "2">0.046</technology>
        <technology code = "3">0.015</technology>
        <technology code = "4">0.055</technology>
    </provider>
    <provider code = "6">
        <!--<technology code = "1">0.04</technology>-->
        <technology code = "2">0.048</technology>
        <technology code = "3">0.014</technology>
        <technology code = "4">0.049</technology>
    </provider>
</QoS>
</service>
</costs>
```

factors.txt

Συντελεστές βαρύτητας των τριών παραγόντων του προφίλ

---

a=0.8

b=0.6

c=0.4

provider\_bonus.txt

Συντελεστής προμοδότησης για τα σημεία access points του προτιμώμενου παρόχου

bonus=0.5

providers.txt

Κατάλογος γνωστών (προκαθορισμένων) παρόχων

---

1=Cosmote

2=Vodafone

3=TIM

4=QTelecom

5=Forthnet

6=Hellas On Line

providers\_inversed.txt

Κατάλογος γνωστών (προκαθορισμένων) παρόχων II

---

Cosmote=1

Vodafone=2

TIM=3

QTelecom=4

Forthnet=5

Hellas On Line=6

services.txt

Κατάλογος Υπηρεσιών (που υποστηρίζει το τερματικό)

---

1=Voice Call  
2=SMS  
3=MMS  
4=e-mail  
5=Video Call  
6=Video Stream  
7=Web Browsing



```
<?xml version="1.0"?>
<services>
  <service code="1" name="Voice Call" availQoSLevels="5">
    <QoS level="1">4</QoS>
    <QoS level="2">8</QoS>
    <QoS level="3">16</QoS>
    <QoS level="4">20</QoS>
    <QoS level="5">25</QoS>
  </service>
  <service code="2" name="SMS" availQoSLevels="1">
    <QoS level="1">1</QoS>
  </service>
  <service code="3" name="MMS" availQoSLevels="2">
    <QoS level="1">8</QoS>
    <QoS level="2">32</QoS>
  </service>
  <service code="4" name="e-mail" availQoSLevels="1">
    <QoS level="1">32</QoS>
  </service>
  <service code="5" name="Video Call" availQoSLevels="5">
    <QoS level="1">32</QoS>
    <QoS level="2">64</QoS>
    <QoS level="3">128</QoS>
    <QoS level="4">256</QoS>
    <QoS level="5">384</QoS>
  </service>
  <service code="6" name="Video Stream" availQoSLevels="5">
    <QoS level="1">64</QoS>
    <QoS level="2">128</QoS>
    <QoS level="3">512</QoS>
    <QoS level="4">1024</QoS>
    <QoS level="5">2000</QoS>
  </service>
  <service code="7" name="Web Browsing" availQoSLevels="5">
    <QoS level="1">32</QoS>
    <QoS level="2">64</QoS>
    <QoS level="3">128</QoS>
    <QoS level="4">384</QoS>
    <QoS level="5">1024</QoS>
  </service>
</services>
```

signal\_factors.txt

Πολλαπλασιαστικοί συντελεστές για κάθε επίπεδο ισχύος σήματος

---

1=0.4

2=0.6

3=0.8

4=0.95

5=1.0

technologies.txt

Κατάλογος τεχνολογιών πρόσβασης (που υποστηρίζει το τερματικό)

---

1=GSM

2=UMTS

3=WLAN

4=DVB

current.txt

Ενδεικτικό αρχείο ρυθμίσεων προφίλ χρήστη

---

```
#The 3 main factors
#factor1 is the most important one, factor3 is the least important
factor1=Quality
factor2=Cost
factor3=Provider
#Specify the provider
providerName=Cosmote
providerCode=1
#QoS Settings
1=3
2=1
3=2
4=1
5=3
6=3
7=3
```

```

<?xml version="1.0"?>
<measurements>
  <measurement timer="0"><!--LAB-->
    <AP ip="147.102.3.1" providerCode="4" technology="3" signal="5"
    availBandwidth="2000"/>
    <AP ip="147.102.3.2" providerCode="5" technology="3" signal="5"
    availBandwidth="3000"/>
    <AP ip="147.102.3.3" providerCode="6" technology="3" signal="5"
    availBandwidth="2000"/>
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="4"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="5"
    availBandwidth="1000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="4"
    availBandwidth="1000"/>
  </measurement>
  <measurement timer="1"><!--LAB => OFFICE, NEAR LAB-->
    <AP ip="147.102.3.1" providerCode="4" technology="3" signal="1"
    availBandwidth="2000"/>
    <AP ip="147.102.3.2" providerCode="5" technology="3" signal="2"
    availBandwidth="3000"/>
    <AP ip="147.102.3.3" providerCode="6" technology="3" signal="1"
    availBandwidth="2000"/>
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="4"
    availBandwidth="1000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="5"
    availBandwidth="1000"/>
  </measurement>
  <measurement timer="2"><!--LAB => OFFICE-->
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="4"
    availBandwidth="1000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="5"
    availBandwidth="1000"/>
  </measurement>
  <measurement timer="3"><!--OFFICE-->
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="4"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="4"
    availBandwidth="1000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="5"
    availBandwidth="1000"/>
    <!--NEW-->
    <AP ip="147.102.3.11" providerCode="6" technology="3" signal="5"
    availBandwidth="2500"/>
  </measurement>
  <measurement timer="4"><!--OFFICE => TREES-->
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="5"
    availBandwidth="1000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="4"
    availBandwidth="1000"/>
  </measurement>
  <measurement timer="5"><!--TREES-->
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="1"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="1"
    availBandwidth="1000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="1"

```

```

    availBandwidth="1000"/>
</measurement>
<measurement timer="6"><!--PARKING-->
  <AP ip="147.102.1.1" providerCode="1" technology="1" signal="1"
  availBandwidth="170"/>
  <AP ip="147.102.2.1" providerCode="1" technology="2" signal="1"
  availBandwidth="1000"/>
</measurement>
<measurement timer="7"><!--PARKING => AMF-->
  <AP ip="147.102.1.1" providerCode="1" technology="1" signal="3"
  availBandwidth="170"/>
  <AP ip="147.102.1.2" providerCode="2" technology="1" signal="4"
  availBandwidth="170"/>
  <AP ip="147.102.2.1" providerCode="1" technology="2" signal="2"
  availBandwidth="1000"/>
  <AP ip="147.102.2.2" providerCode="3" technology="2" signal="4"
  availBandwidth="1000"/>
</measurement>
<measurement timer="8"><!--AMF-->
  <AP ip="147.102.1.1" providerCode="1" technology="1" signal="2"
  availBandwidth="170"/>
  <AP ip="147.102.1.2" providerCode="2" technology="1" signal="3"
  availBandwidth="170"/>
  <AP ip="147.102.2.1" providerCode="1" technology="2" signal="3"
  availBandwidth="1000"/>
  <AP ip="147.102.2.2" providerCode="3" technology="2" signal="3"
  availBandwidth="1000"/>
  <!--NEW-->
  <AP ip="147.102.3.21" providerCode="5" technology="3" signal="5"
  availBandwidth="3000"/>
</measurement>
</measurements>

```

```

<?xml version="1.0"?>
<measurements>
  <measurement timer="0"><!--HOME-->
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="5"
    availBandwidth="2000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="5"
    availBandwidth="2000"/>
    <AP ip="147.102.4.1" providerCode="5" technology="4" signal="5"
    availBandwidth="4000"/>
    <AP ip="147.102.4.2" providerCode="6" technology="4" signal="5"
    availBandwidth="4000"/>
  </measurement>
  <measurement timer="1"><!--HOME => CAR-->
    <AP ip="147.102.1.1" providerCode="1" technology="1" signal="2"
    availBandwidth="170"/>
    <AP ip="147.102.1.2" providerCode="2" technology="1" signal="2"
    availBandwidth="170"/>
    <AP ip="147.102.2.1" providerCode="1" technology="2" signal="3"
    availBandwidth="2000"/>
    <AP ip="147.102.2.2" providerCode="3" technology="2" signal="2"
    availBandwidth="2000"/>
    <AP ip="147.102.4.1" providerCode="5" technology="4" signal="1"
    availBandwidth="4000"/>
    <!--NEW-->
    <AP ip="147.102.1.11" providerCode="1" technology="1" signal="3"
    availBandwidth="170"/>
    <AP ip="147.102.1.12" providerCode="2" technology="1" signal="3"
    availBandwidth="170"/>
    <AP ip="147.102.2.11" providerCode="1" technology="2" signal="3"
    availBandwidth="2000"/>
    <AP ip="147.102.2.12" providerCode="3" technology="2" signal="3"
    availBandwidth="2000"/>
  </measurement>
  <measurement timer="2"><!--CAR-->
    <AP ip="147.102.1.11" providerCode="1" technology="1" signal="4"
    availBandwidth="170"/>
    <AP ip="147.102.1.12" providerCode="2" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.2.11" providerCode="1" technology="2" signal="4"
    availBandwidth="2000"/>
    <AP ip="147.102.2.12" providerCode="3" technology="2" signal="5"
    availBandwidth="2000"/>
  </measurement>
  <measurement timer="3"><!--CAR => SUBWAY-->
    <AP ip="147.102.1.11" providerCode="1" technology="1" signal="1"
    availBandwidth="170"/>
    <AP ip="147.102.1.12" providerCode="2" technology="1" signal="1"
    availBandwidth="170"/>
    <AP ip="147.102.2.11" providerCode="1" technology="2" signal="1"
    availBandwidth="2000"/>
    <AP ip="147.102.2.12" providerCode="3" technology="2" signal="1"
    availBandwidth="2000"/>
    <!--NEW-->
    <AP ip="147.102.1.21" providerCode="1" technology="1" signal="5"
    availBandwidth="170"/>
    <AP ip="147.102.1.22" providerCode="2" technology="1" signal="4"
    availBandwidth="170"/>
    <AP ip="147.102.3.1" providerCode="4" technology="3" signal="4"
    availBandwidth="3000"/>
  </measurement>
  <measurement timer="4"><!--SUBWAY-->
    <AP ip="147.102.1.21" providerCode="1" technology="1" signal="1"
    availBandwidth="170"/>
    <!--NEW-->
    <AP ip="147.102.1.31" providerCode="2" technology="1" signal="4"
    availBandwidth="170"/>
    <AP ip="147.102.1.32" providerCode="4" technology="1" signal="3"
    availBandwidth="170"/>
  </measurement>

```

```
<measurement timer="5"><!--SUBWAY => BUS-->
  <AP ip="147.102.1.31" providerCode="2" technology="1" signal="1"
    availBandwidth="170"/>
  <AP ip="147.102.1.32" providerCode="4" technology="1" signal="1"
    availBandwidth="170"/>
  <!--NEW-->
  <AP ip="147.102.1.41" providerCode="1" technology="1" signal="2"
    availBandwidth="170"/>
  <AP ip="147.102.1.42" providerCode="4" technology="1" signal="3"
    availBandwidth="170"/>
  <AP ip="147.102.2.21" providerCode="1" technology="2" signal="2"
    availBandwidth="2000"/>
  <AP ip="147.102.2.22" providerCode="3" technology="2" signal="3"
    availBandwidth="2000"/>
</measurement>
<measurement timer="6"><!--BUS-->
  <AP ip="147.102.1.41" providerCode="1" technology="1" signal="4"
    availBandwidth="170"/>
  <AP ip="147.102.1.42" providerCode="4" technology="1" signal="5"
    availBandwidth="170"/>
  <AP ip="147.102.2.21" providerCode="1" technology="2" signal="4"
    availBandwidth="2000"/>
  <AP ip="147.102.2.22" providerCode="3" technology="2" signal="5"
    availBandwidth="2000"/>
</measurement>
<measurement timer="7"><!--BUS => OFFICE-->
  <AP ip="147.102.1.41" providerCode="1" technology="1" signal="1"
    availBandwidth="170"/>
  <AP ip="147.102.1.42" providerCode="4" technology="1" signal="1"
    availBandwidth="170"/>
  <AP ip="147.102.2.21" providerCode="3" technology="2" signal="1"
    availBandwidth="2000"/>
  <!--NEW-->
  <AP ip="147.102.1.51" providerCode="1" technology="1" signal="3"
    availBandwidth="170"/>
  <AP ip="147.102.1.52" providerCode="4" technology="1" signal="3"
    availBandwidth="170"/>
  <AP ip="147.102.2.31" providerCode="1" technology="2" signal="3"
    availBandwidth="1500"/>
</measurement>
<measurement timer="8"><!--OFFICE-->
  <AP ip="147.102.1.51" providerCode="1" technology="1" signal="5"
    availBandwidth="170"/>
  <AP ip="147.102.1.52" providerCode="4" technology="1" signal="4"
    availBandwidth="170"/>
  <AP ip="147.102.2.31" providerCode="1" technology="2" signal="5"
    availBandwidth="1500"/>
  <!--NEW-->
  <AP ip="147.102.4.11" providerCode="5" technology="4" signal="5"
    availBandwidth="4000"/>
  <AP ip="147.102.3.11" providerCode="5" technology="3" signal="2"
    availBandwidth="3000"/>
  <AP ip="147.102.3.12" providerCode="6" technology="3" signal="5"
    availBandwidth="3500"/>
</measurement>
</measurements>
```



## Γ. Τεκμηρίωση Κώδικα (Documentation)

diplomatiki

## Class Config

public class **Config**

Title: Term Paper - `Config` class

Description: This class encapsulates configuration information for the TMS. All variables defined here class are class (static) variables and most of them are also final.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### Config

public **Config**()

Creates a new `Config` instance. This constructor is never used, as all variables in this class are static.

### Method Detail

#### init

public static void **init**(String username)

Initiates the values of some static variables, including the name of the current user, and the paths to the current user's stored profiles.

**Parameters:**

`username` - The user's name. In general, the TMS can support multiple users

diplomatiki

## Class InfoStruct

public class **InfoStruct**

Title: Term Paper - InfoStruct class

Description: Instances of this class encapsulate all the information that the GUI needs to know (and present), concerning an open connection of the terminal.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### InfoStruct

```
public InfoStruct()
```

Creates a new InfoStruct instance.

---

#### InfoStruct

```
public InfoStruct(String ipAddress,  
                  String providerName,  
                  String accessTechnology,  
                  String serviceName,  
                  String qos)
```

Creates a new InfoStruct instance with the specified IP Address, Provider Name, Access Technology, Service Name and QoS Level. This means that the specified service has been allocated to the specified QoS level and access point. These pieces of information are presented in the terminal's GUI.

**Parameters:**

`ipAddress` - The access point's IP address  
`providerName` - The access point's provider name  
`accessTechnology` - The point's access technology  
`serviceName` - The name of the service  
`qos` - The QoS level

### Method Detail

## **setIpAddress**

```
public void setIpAddress(String ipAddress)
```

Sets the access point's IP address.

### **Parameters:**

`ipAddress` - The access point's IP address

---

## **setProviderName**

```
public void setProviderName(String providerName)
```

Sets the access point's provider name.

### **Parameters:**

`providerName` - The access point's provider name

---

## **setAccessTechnology**

```
public void setAccessTechnology(String accessTechnology)
```

Sets the point's access technology.

### **Parameters:**

`accessTechnology` - The point's access technology

---

## **setServiceName**

```
public void setServiceName(String serviceName)
```

Sets the name of the service

### **Parameters:**

`serviceName` - The service's name

---

## **setQos**

```
public void setQos(String qos)
```

Sets the QoS level.

### **Parameters:**

`qos` - The QoS level

---

## **getIpAddress**

```
public String getIpAddress()
```

Returns the access point's IP address.

### **Returns:**

The access point's IP address

---

## **getProviderName**

```
public String getProviderName()
```

Returns the access point's provider name.

### **Returns:**

The access point's provider name

---

## **getAccessTechnology**

```
public String getAccessTechnology()
```

Returns the point's access technology.

### **Returns:**

The point's access technology

---

## **getServiceName**

```
public String getServiceName()
```

Returns the name of the service.

### **Returns:**

The service's name

---

## **getQos**

```
public String getQos()
```

Returns the QoS level.

### **Returns:**

The QoS level

---

diplomati

# Class TMSApplication

public class TMSApplication

Title: Term Paper - TMSApplication class

Description: This class holds the main() method. The execution of the Terminal Management System (TMS) begins from this point.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### TMSApplication

public TMSApplication()

Creates a new TMSApplication instance. It is not necessary to use this constructor, as the only method in this class is the main() method.

## Method Detail

### main

public static void main(String[] args)

Starts the TMS application. Sets the GUI's "look and feel" and launches the PINApplication.

**Parameters:**

args - No command line arguments are necessary

## Class Choice

public class **Choice**

Title: Term Paper - *Choice* class

Description: An instance of this class represents an allocation choice

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### Choice

```
public Choice()
```

Creates a new *Choice* instance.

---

#### Choice

```
public Choice(String ipAddress,  
              int providerCode,  
              int accessTechnology,  
              int serviceCode,  
              int qos,  
              double grade)
```

Creates a new *Choice* instance with the specified IP Address, Provider Code, Access Technology, Service Code, QoS Level and Grade.

**Parameters:**

*ipAddress* - The IP address of a candidate access point

*providerCode* - The access point's provider code

*accessTechnology* - The point's access technology

*serviceCode* - The code of the requested service

*qos* - A candidate QoS level

*grade* - The grade (rating) of this allocation choice

### Method Detail



## **setIpAddress**

```
public void setIpAddress(String ipAddress)
```

Sets the candidate access point's IP address

### **Parameters:**

`ipAddress` - The candidate's IP address

---

## **setProviderCode**

```
public void setProviderCode(int providerCode)
```

Sets the candidate access point's provider code

### **Parameters:**

`providerCode` - The candidate's provider code

---

## **setAccessTechnology**

```
public void setAccessTechnology(int accessTechnology)
```

Sets the candidate access point's access technology.

### **Parameters:**

`accessTechnology` - Sets the candidate's access technology

---

## **setServiceCode**

```
public void setServiceCode(int serviceCode)
```

Sets the code of the requested service.

### **Parameters:**

`serviceCode` - The code of the requested service

---

## **setQos**

```
public void setQos(int qos)
```

Sets the QoS level of this allocation choice.

### **Parameters:**

`qos` - The QoS level of this choice

---

## **setGrade**

```
public void setGrade(double grade)
```

Sets the grade (rating) of this allocation choice.

### **Parameters:**

grade - The grade of this choice

---

## **getIpAddress**

```
public String getIpAddress()
```

Returns the candidate access point's IP address.

### **Returns:**

The candidate's IP address

---

## **getProviderCode**

```
public int getProviderCode()
```

Returns the candidate access point's provider code.

### **Returns:**

The candidate's provider code

---

## **getAccessTechnology**

```
public int getAccessTechnology()
```

Returns the candidate access point's access technology.

### **Returns:**

Returns the candidate's access technology

---

## **getServiceCode**

```
public int getServiceCode()
```

Returns the code of the requested service.

### **Returns:**

The code of the requested service

---

### **getQos**

```
public int getQos()
```

Returns the QoS level of this allocation choice

#### **Returns:**

The QoS level of this choice

---

### **getGrade**

```
public double getGrade()
```

Returns the grade (rating) of this allocation choice.

#### **Returns:**

The grade of this choice

---

### **stringRepresentation**

```
public String stringRepresentation()
```

Returns a string with all the details of this allocation choice.

#### **Returns:**

A string fully representing this choice

---

diplomati.mobmgr

## Class DeteriorationController

public class **DeteriorationController**  
extends Thread

**Deprecated.** *This class is no longer used by the Terminal Management System (TMS)*

Title: Term Paper - `DeteriorationController` class

Description: This class can be used to control the deterioration degree of the running services. More specifically, it uses a random number generator, in order to simulate if the signal from a specific access point is severely deteriorated or not. If so, the IIS algorithm should be triggered. This class has been used in the early stages of the implementation, but is no longer supported.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### DeteriorationController

```
public DeteriorationController()
```

Creates a new `DeteriorationController` instance.

---

#### DeteriorationController

```
public DeteriorationController(MobilityManager mm)
```

Creates a new `DeteriorationController` instance that is associated with the specified `MobilityManager` object.

**Parameters:**

`mm` - A reference to the Mobility Manager module

---

### Method Detail

**run**

```
public void run()
```

A `DeteriorationController` instance runs in a separate thread. This is the entry point of this thread. Special care has been taken in order to make the TMS thread-safe.

---

diplomati.mobmgr

## Class **MobilityManager**

public class **MobilityManager**

Title: Term Paper - `MobilityManager` class

Description: This is the main class implementing the Mobility Manager module of a TMS.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### **MobilityManager**

```
public MobilityManager()
```

Creates a new `MobilityManager` instance.

---

#### **MobilityManager**

```
public MobilityManager(MenuFrame myFrame)
```

Creates a new `MobilityManager` instance. The specified `MenuFrame` object is the GUI element that will be used for presenting connectivity information.

**Parameters:**

`myFrame` - The GUI element associated with the Mobility Manager module

### Method Detail

#### **getMeasurements**

```
public String[][] getMeasurements()
```

Acquires the Measurements from the Interface Adaptation module.

**Returns:**

The table of current Measurements

---

## getRunningServices

```
public Vector getRunningServices()
```

Returns the list of running services.

### Returns:

The list of running services

---

## getInterfaceAdaptation

```
public InterfaceAdaptation getInterfaceAdaptation()
```

Returns a reference to the Interface Adaptation module.

### Returns:

A reference to the Interface Adaptation module, that is a reference to the `InterfaceAdaptation` class

---

## getMenuFrame

```
public MenuFrame getMenuFrame()
```

Returns a reference to the associated `MenuFrame` object.

### Returns:

A reference to the associated `MenuFrame` object

---

## executeIIS

```
public InfoStruct executeIIS(int serviceCode,  
                               boolean firstTime,  
                               int replaceIndex)
```

This is the Mobility Manager's main method. It triggers the IIS algorithm for the specified service. It also handles the communication with the Network Management System. Finally, it updates the list of running services and presents the results to the GUI.

### Parameters:

`serviceCode` - The code of the service for which the IIS algorithm should be triggered

`firstTime` - This parameter specifies if the requested service is already running or not. If `false`, then it is already running.

`replaceIndex` - If the requested service is already running, then this argument represents the service's position in the list of running services.

### Returns:

Information about the connection chosen by the IIS sub-module (and approved by the

Network Management System).

---

### **removeFromRunningServices**

```
public void removeFromRunningServices(int serviceCode)
```

Removes the specified service from the list of running services.

**Parameters:**

`serviceCode` - The code of the service that should be removed

---

### **reExamineAllRunningServices**

```
public void reExamineAllRunningServices()
```

Re-examines all running services. In other words, it invokes the `executeIIS()` method for all the services in the list of running services.

---

### **loadPropertiesFile**

```
private Properties loadPropertiesFile(String filePath)
```

Loads a Properties file.

**Parameters:**

`filePath` - The path of the properties file

**Returns:**

A `Properties` instance that represents the properties file's contents

---

### **findNearestNetworkEntityAddress**

```
private String findNearestNetworkEntityAddress()
```

Finds the address of the nearest Network Management System entity. This method offers a trivial implementation of the search process.

**Returns:**

The address of the nearest network entity

---

### **openLogFile**

```
private void openLogFile()
```



Opens a log file in order to append log data.

---

### **writeMeasurementsToLogFile**

```
public void writeMeasurementsToLogFile(PrintWriter fout)
```

Appends the table of Measurements to the log file.

**Parameters:**

`fout` - The output stream that should be used to append the data

---

### **writeRequestToLogFile**

```
private void writeRequestToLogFile(String requestString)
```

Prints the details of the request submitted to the Network Management System to the log file.

**Parameters:**

`requestString` - The details of the request made to the Network Management System

---

### **writeNetworkDecisionToLogFile**

```
private void writeNetworkDecisionToLogFile(boolean accepted)
```

Prints the decision of the Network Management System to the log file. If `accepted` is `true`, then the Network Management System has accepted the terminal's request.

**Parameters:**

`accepted` - The Network Management System's decision

---

### **closeLogFile**

```
private void closeLogFile()
```

Closes the log file.

---

## Class **AccessPoint**

public class **AccessPoint**

Title: Term Paper - `AccessPoint` class

Description: Objects of this class represent access points which the terminal can connect to.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### **AccessPoint**

```
public AccessPoint()
```

Creates a new `AccessPoint` instance.

---

#### **AccessPoint**

```
public AccessPoint(String ipAddress,  
                  int providerCode,  
                  int accessTechnology,  
                  int signalStrength,  
                  int availBandwidth)
```

Creates a new `AccessPoint` instance with the specified IP Address, Provider Code, Access Technology, Signal Strength and Bandwidth.

**Parameters:**

`ipAddress` - The access point's IP address  
`providerCode` - The access point's provider code  
`accessTechnology` - The point's access technology  
`signalStrength` - The strength of the signal currently received by the access point  
`availBandwidth` - The access point's available bandwidth (in Kbps)

### Method Detail

#### **setIpAddress**

```
void setIpAddress (String ipAddress)
```

Sets the access point's IP address.

**Parameters:**

`ipAddress` - The new IP address

---

### **setProvider**

```
void setProvider (int providerCode)
```

Sets the access point's provider code.

**Parameters:**

`providerCode` - The new provider code

---

### **setAccessTechnology**

```
void setAccessTechnology (int accessTechnology)
```

Sets the point's access technology.

**Parameters:**

`accessTechnology` - The new access technology code

---

### **setSignalStrength**

```
void setSignalStrength (int signalStrength)
```

Sets the access point's signal strength.

**Parameters:**

`signalStrength` - The new signal strength

---

### **setAvailBandwidth**

```
void setAvailBandwidth (int availBandwidth)
```

Sets the access point's available bandwidth.

**Parameters:**

`availBandwidth` - The new available bandwidth

---

## createInitialInfo

```
public void createInitialInfo(int noRows)
```

Creates the info table for this access point.

### Parameters:

`noRows` - Defines the size of the table. The actual number of rows created is the value of this argument plus one (`noRows+1`). The number of columns is always equal to 3

---

## setInfo

```
void setInfo(int indexQoS,  
             int column,  
             double value)
```

Sets the value of the table element specified. The two first arguments specify the element's position, while the third specifies the new value.

### Parameters:

`indexQoS` - The table row of the element  
`column` - The table column of the element  
`value` - The new value of the element specified

---

## setToBeRemoved

```
public void setToBeRemoved(boolean toBeRemoved)
```

Sets the `toBeRemoved` property (instance variable) of this object. If `true`, then this access point should be removed from the list of access points maintained by the IIS sub-module.

### Parameters:

`toBeRemoved` - The new value of the `toBeRemoved` property.

---

## getIpAddress

```
String getIpAddress()
```

Returns the access point's IP address.

### Returns:

The access point's IP address

---

## getProviderCode

```
int getProviderCode ()
```

Returns the access point's provider code.

**Returns:**

The access point's provider code

---

### **getAccessTechnology**

```
int getAccessTechnology ()
```

Returns the point's access technology.

**Returns:**

The point's access technology

---

### **getSignalStrength**

```
int getSignalStrength ()
```

Returns the access point's signal strength.

**Returns:**

The access point's signal strength

---

### **getAvailBandwidth**

```
int getAvailBandwidth ()
```

Returns the access point's available bandwidth.

**Returns:**

The access point's available bandwidth

---

### **getInfoSize**

```
int getInfoSize ()
```

Returns the number of rows of the access point's info table. If the table is `null`, it returns 0.

**Returns:**

The number of rows of the info table

---

## **getInfo**

```
double getInfo(int qos,  
               int column)
```

Returns the value of the specified table element. The element's position is specified by the two arguments.

### **Parameters:**

`qos` - The element's table row

`column` - The element's table column

### **Returns:**

The element's value

---

## **isToBeRemoved**

```
public boolean isToBeRemoved()
```

Returns the value of the `toBeRemoved` property (instance variable). If `true`, then this access point should be removed from the list of access points maintained by the IIS sub-module.

### **Returns:**

The value of the `toBeRemoved` property.

---

## Class IIS

public class IIS

Title: Term Paper - IIS class

Description: This is the main class implementing the IIS sub-module of a TMS.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### IIS

```
public IIS ()
```

Creates a new IIS instance.

---

#### IIS

```
public IIS (MobilityManager mm)
```

Creates a new IIS instance that belongs to the specified `MobilityManager` object.

**Parameters:**

`mm` - The `MobilityManager` object that represents the module which contains the IIS sub-module

### Method Detail

#### execute

```
public Stack execute (int serviceCode)
```

Executes the Intelligent Interface Selection (IIS) algorithm.

**Parameters:**

`serviceCode` - The code of the service for which the IIS algorithm will be executed

**Returns:**

An ordered list of allocation choices

---

### **getMeasurements**

```
private String[][] getMeasurements()
```

Acquires the table of Measurements.

**Returns:**

The table of Measurements

---

### **convertToAPList**

```
private void convertToAPList(String[][] measurements)
```

Converts the table of Measurements to a list of `AccessPoint` objects.

**Parameters:**

`measurements` - The table of Measurements

---

### **constructServiceItem**

```
private Service constructServiceItem(int serviceCode)
```

Constructs a `Service` item.

**Parameters:**

`serviceCode` - The code of the service for which the `Service` object should be constructed

**Returns:**

The `Service` object

---

### **findServiceElement**

```
private Element findServiceElement(int serviceCode,  
                                   Element rootElement)
```

Finds the appropriate *service* element from the "services.xml" file of the databank. Returns the *service* element that has the specified attribute value. The name of this attribute is "code".

**Parameters:**

`serviceCode` - The value of the attribute "code". The search performed is based on this value  
`rootElement` - The parent element of the element that is searched for

**Returns:**

The appropriate *service* element



---

## findQuality

```
private void findQuality(AccessPoint myAP)
```

The "Find Quality" stage of the IIS algorithm. This stage finds the QoS levels supported by the given access point. It also calculates the respective *real* QoS levels.

### Parameters:

myAP - The access point for which this stage of the algorithm will be executed

---

## findCost

```
private void findCost(AccessPoint myAP)
```

The "Find Cost" stage of the IIS algorithm. This stage finds the cost of all the possible allocation choices for the given access point.

### Parameters:

myAP - The access point for which this stage of the algorithm will be executed

---

## findChildByAttribute

```
private Element findChildByAttribute(Element parentEl,  
                                     String attrName,  
                                     String attrValue)
```

Finds the appropriate child element of the given element, in an XML document, by using the given attribute name/attribute value pair as a search criteria.

### Parameters:

parentEl - The parent element of the element that we search for

attrName - The name of the attribute

attrValue - The value of the attribute

### Returns:

The appropriate XML element

---

## objectiveFunction

```
private void objectiveFunction(AccessPoint myAP)
```

The "Objective Function" stage of the IIS algorithm. In this stage, we calculate the value of the objective function for all the possible allocation choices with respect to the given access point.

### Parameters:

myAP - The access point for which this stage of the algorithm will be executed

---

## loadPropertiesFile

```
private Properties loadPropertiesFile(String filePath)
```

Loads a Properties file.

### Parameters:

`filePath` - The path of the properties file

### Returns:

A `Properties` instance that represents the properties file's contents

---

## convertToChoicesList

```
private void convertToChoicesList()
```

Converts the list of access points to a list of choices (`Choice` objects).

---

## findBest

```
private void findBest()
```

Sorts all the possible allocation choices according to their grade.

---

## bubbleSort

```
private void bubbleSort(Choice[] choiceArray)
```

Sorts a table using the "Bubble Sort" algorithm.

### Parameters:

`choiceArray` - The table that should be sorted

---

## openLogFile

```
private void openLogFile()
```

Opens a log file in order to append log data.

---

## writeInitialInfoToLogFile

```
private void writeInitialInfoToLogFile()
```

Writes all the appropriate initial info to the log file, including the current date/time, service name, current measurements, user preferences, and more.

---

### **writeOFDataToLogFile**

```
private void writeOFDataToLogFile(String ofData)
```

Prints the grade of an allocation choice and some intermediate calculation data to the log file.

**Parameters:**

ofData - The piece of info that should be printed

---

### **writeSortedListToLogFile**

```
private void writeSortedListToLogFile(Choice[] myArray)
```

Prints the sorted list of allocation choices to the log file.

**Parameters:**

myArray - The sorted table of allocation choices

---

### **closeLogFile**

```
private void closeLogFile()
```

Closes the log file.

---

## Class Service

public class **Service**

Title: Term Paper - *Service* class

Description: Objects of this class encapsulate information about a requested service.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### Service

```
public Service ()
```

Creates a new *Service* instance.

---

#### Service

```
public Service (String serviceName,  
                int serviceCode,  
                int maxUsersQoS,  
                int noQoSLevels)
```

Creates a new *Service* instance with the specified Name, Code, Maximum Preferred QoS Level and Number of QoS Levels.

**Parameters:**

*serviceName* - The service's name

*serviceCode* - The service's code

*maxUsersQoS* - The service's maximum QoS level (this is defined by the user's preferences)

*noQoSLevels* - The total number of QoS levels for this service

### Method Detail

#### setServiceName

```
void setServiceName (String serviceName)
```

Sets the service's name.

**Parameters:**

`serviceName` - The new name

---

**setServiceCode**

```
void setServiceCode(int serviceCode)
```

Sets the service's code.

**Parameters:**

`serviceCode` - The new code

---

**setMaxUsersQoS**

```
void setMaxUsersQoS(int maxUsersQoS)
```

Sets the maximum preferred QoS level.

**Parameters:**

`maxUsersQoS` - The new maximum QoS level

---

**setNoQoSLevels**

```
void setNoQoSLevels(int noQoSLevels)
```

Sets the total number of QoS levels for this service.

**Parameters:**

`noQoSLevels` - The new total number of QoS levels

---

**setQoS**

```
void setQoS(int qosIndex,  
             int bitRate)
```

Sets the bitrate for the specified QoS level. All bitrates are kept in a table. For example, if the bitrate of the QoS level 2 is 8 Kbps, then this method should be called as follows:

```
Service s = new Service();  
s.setQoS(2, 8);
```

**Parameters:**

`qosIndex` - The QoS level

bitRate - The bitrate of the qosIndex QoS level

---

### **getServiceName**

String **getServiceName** ()

Returns the service's name.

**Returns:**

The service's name

---

### **getServiceCode**

int **getServiceCode** ()

Returns the service's code.

**Returns:**

The service's code

---

### **getMaxUsersQoS**

int **getMaxUsersQoS** ()

Returns the maximum QoS level (according to the user's preferences).

**Returns:**

The maximum preferred QoS level

---

### **getNoQoSLevels**

int **getNoQoSLevels** ()

Returns the total number of QoS levels for this service.

**Returns:**

The total number of QoS levels

---

### **getQoS**

int **getQoS** (int qosIndex)

Returns the bitrate of the specified QoS level.

**Parameters:**

qosIndex - The QoS level

**Returns:**

The bitrate of the qosIndex QoS level

---

# Class InterfaceAdaptation

public class **InterfaceAdaptation**  
extends Thread

Title: Term Paper - `InterfaceAdaptation` class

Description: This is the main class implementing the Interface Adaptation (IA) module of a TMS. An `InterfaceAdaptation` instance runs in a separate thred, so special care has been taken in order to make its methods thread-safe.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### InterfaceAdaptation

```
public InterfaceAdaptation()
```

Creates a new `InterfaceAdaptation` instance.

---

### InterfaceAdaptation

```
public InterfaceAdaptation(MobilityManager mm)
```

Creates a new `InterfaceAdaptation` instance, giving it also a reference to the TMS's Mobility Manager module. This constructor also initiates (by invoking the `init()` method) and starts the IA thread.

**Parameters:**

`mm` - A reference to the Mobility Manager module

---

## Method Detail

### setCurrentMeasurements

```
public synchronized void setCurrentMeasurements(String[][] currentMeasurements)
```

Sets the table of current Measurements.



**Parameters:**

`currentMeasurements` - The new table of current Measurements

---

**setPreviousMeasurements**

```
public void setPreviousMeasurements(String[][] previousMeasurements)
```

Sets the table of previous Measurements.

**Parameters:**

`previousMeasurements` - The new table of previous Measurements

---

**getCurrentMeasurements**

```
public synchronized String[][] getCurrentMeasurements()
```

Returns the table of current Measurements.

**Returns:**

The table of current Measurements

---

**getPreviousMeasurements**

```
public String[][] getPreviousMeasurements()
```

Returns the table of previous Measurements.

**Returns:**

The table of previous Measurements.

---

**run**

```
public void run()
```

The entry point of the IA thread. This thread stays alive until the whole TMS is terminated.

---

**measure**

```
public String[][] measure()
```

Retrieves the current Measurements from the measurements XML file.

**Returns:**

A table with the current Measurements

---

### **checkForAPLoss**

```
private void checkForAPLoss()
```

Checks if an access point that is in use by the terminal has disappeared.

---

### **checkForNewAP**

```
private boolean checkForNewAP()
```

Checks if a new access point has appeared. If so, this method returns `true`.

#### **Returns:**

`True` if a new access point has appeared, otherwise `false`

---

### **findChildByAttribute**

```
private Element findChildByAttribute(Element parentEl,  
                                     String attrName,  
                                     String attrValue)
```

Finds the appropriate child element of the given element, in an XML document, by using the given attribute name/attribute value pair as a search criteria.

#### **Parameters:**

`parentEl` - The parent element of the element that we search for  
`attrName` - The name of the attribute  
`attrValue` - The value of the attribute

#### **Returns:**

The appropriate XML element

---

### **printMeasurements**

```
public void printMeasurements(String[][] measurements)
```

Prints the specified table of Measurements to the system's console.

#### **Parameters:**

`measurements` - The table of Measurements to be printed

---

### **init**

```
private int init()
```

Finds out the maximum value of the `timer` instance variable from the measurements XML document, and returns this value.

**Returns:**

The maximum value of the `timer` instance variable

---

## Class InterfaceAdaptationDeprecated

public class **InterfaceAdaptationDeprecated**

**Deprecated.** *This class is no longer used by the Terminal Management System (TMS)*

Title: Term Paper - InterfaceAdaptationDeprecated class

Description: This class has been used in order to implement the Interface Adaptation (IA) module in the early stages of development. It uses a different method to retrieve link-layer measurements and is no longer supported. It doesn't check for access point appearance or disappearance.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### InterfaceAdaptationDeprecated

```
public InterfaceAdaptationDeprecated()
```

Creates a new `InterfaceAdaptation` instance. It also invokes the `init()` method.

### Method Detail

#### measure

```
public String[][] measure(boolean newmsr)
```

Retrieves the current Measurements from the measurements XML file.

**Parameters:**

`newmsr` - If `true` then new Measurements should be retrieved. The `timer` instance variable is increased only if `newmsr` is `true`

**Returns:**

A table with the current Measurements

---

#### increaseTimer

```
public void increaseTimer()
```

Increases the `timer` instance variable by 1.

---

## **init**

```
private int init()
```

Finds out the maximum value of the `timer` instance variable from the measurements XML document, and returns this value.

### **Returns:**

The maximum value of the `timer` instance variable

---

## **findChildByAttribute**

```
private Element findChildByAttribute(Element parentEl,  
                                     String attrName,  
                                     String attrValue)
```

Finds the appropriate child element of the given element, in an XML document, by using the given attribute name/attribute value pair as a search criteria.

### **Parameters:**

`parentEl` - The parent element of the element that we search for  
`attrName` - The name of the attribute  
`attrValue` - The value of the attribute

### **Returns:**

The appropriate XML element

---

## **printMeasurements**

```
public void printMeasurements(String[][] measurements)
```

Prints the specified table of Measurements to the system's console.

### **Parameters:**

`measurements` - The table of Measurements to be printed

---

diplomatiki.gui

## Class ApplicationsPanel

```
public class ApplicationsPanel
extends JPanel
```

Title: Term Paper - ApplicationsPanel class

Description: This is the applications panel of the GUI. In this panel, details for all running services are presented.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### ApplicationsPanel

```
public ApplicationsPanel()
```

Creates a new ApplicationsPanel instance.

---

### ApplicationsPanel

```
public ApplicationsPanel(JFrame myFrame,
                          JPanel previousPanel)
```

Creates a new ApplicationsPanel instance with the specified parent frame and previous panel.

### Parameters:

myFrame - The frame which this panel belongs to

previousPanel - The previous panel that was visible in the myFrame frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### loadPropertiesFile

```
private Properties loadPropertiesFile(String filePath)
```

Loads a Properties file.

**Parameters:**

`filePath` - The path of the properties file

**Returns:**

A `Properties` instance that represents the properties file's contents

---

diplomatiki.gui

## Class CalendarTitle

public class **CalendarTitle**  
implements Runnable

Title: Term Paper - `CalendarTitle` class

Description: An instance of this class runs as a separate thread and is responsible for updating the frame's title bar with the current date and time.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### CalendarTitle

```
public CalendarTitle()
```

Creates a new `CalendarTitle` instance. It also starts the `CalendarTitle` thread.

---

### CalendarTitle

```
public CalendarTitle(JFrame myFrame)
```

Creates a new `CalendarTitle` instance. It also starts the `CalendarTitle` thread. The time will be shown on `myFrame`'s title bar.

### Parameters:

`myFrame` - The title of this `JFrame` will show the current date and time

---

## Method Detail

### run

```
public void run()
```

Entry point of this thread. This thread stays alive until the `stop()` method is invoked.

---



## **start**

```
public void start()
```

Starts the `CalendarTitle` thread.

---

## **stop**

```
public void stop()
```

Stops the `CalendarTitle` thread.

---

diplomatiki.gui

## Class ConnectionInfoPanel

```
public class ConnectionInfoPanel
extends JPanel
```

Title: Term Paper - ConnectionInfoPanel class

Description: This panel presents the details of an open connection, e.g. the access point's IP address, access technology and provider name, and the QoS level of the running service. It is used as part of an InfoPanel.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### ConnectionInfoPanel

```
public ConnectionInfoPanel ()
```

Creates a new ConnectionInfoPanel instance.

---

#### ConnectionInfoPanel

```
public ConnectionInfoPanel (String technology,
                             String provider,
                             String qos,
                             String ip)
```

Creates a new ConnectionInfoPanel instance. This panel will present the info specified by the constructor's parameters.

**Parameters:**

technology - The access technology

provider - The provider's name

qos - The QoS level

ip - The access point's IP address

### Method Detail

## **jbInit**

`void jbInit()`

Builds the panel.

### **Throws:**

`Exception` - An exception is thrown if the construction of the panel is not successful

---

diplomatiki.gui

## Class EditPanel

public class **EditPanel**  
extends NextPanel

Title: Term Paper - `EditPanel` class

Description: This panel appears when the Edit Profile button (the center button) of the main menu is pressed.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### EditPanel

```
public EditPanel()
```

Creates a new `EditPanel` instance.

---

### EditPanel

```
public EditPanel(JFrame myFrame,  
                 JPanel previousPanel)
```

Creates a new `EditPanel` instance with the specified parent frame and previous panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### createProfilePanel

```
private void createProfilePanel(int profileCode)
```

Constructs and shows a `ProfilePanel`.

**Parameters:**

`profileCode` - The unique code of the profile (e.g. "Current Profile", "Low Cost Profile", etc.) that should be presented.

---

**currentButton\_actionPerformed**

```
void currentButton_actionPerformed(ActionEvent e)
```

Action performed when the "Current" profile button is pressed.

**Parameters:**

`e` - `ActionEvent`

---

**lowCostButton\_actionPerformed**

```
void lowCostButton_actionPerformed(ActionEvent e)
```

Action performed when the "Low Cost" profile button is pressed.

**Parameters:**

`e` - `ActionEvent`

---

**highQualityButton\_actionPerformed**

```
void highQualityButton_actionPerformed(ActionEvent e)
```

Action performed when the "High Quality" profile button is pressed.

**Parameters:**

`e` - `ActionEvent`

---

**balancedButton\_actionPerformed**

```
void balancedButton_actionPerformed(ActionEvent e)
```

Action performed when the "Balanced" profile button is pressed.

**Parameters:**

`e` - `ActionEvent`

---

diplomatiki.gui

## Class EmailInputPanel

```
public class EmailInputPanel  
extends JPanel
```

Title: Term Paper - EmailInputPanel class

Description: This is the center panel of the Email Service panel. It holds the various input fields, where the user should fill in the appropriate information (e.g. the recipient's address).

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### EmailInputPanel

```
public EmailInputPanel()
```

Creates a new EmailInputPanel instance.

## Method Detail

### jbInit

```
void jbInit()
```

Builds the panel.

### Throws:

Exception - An exception is thrown if the construction of the panel is not successful

### load1\_actionPerformed

```
void load1_actionPerformed(ActionEvent e)
```

Action performed when the "Load" button is pressed. The loadFile() method is invoked.

### Parameters:

e - ActionEvent

---

## **loadFile**

```
void loadFile(JFileChooser jfc,  
              JTextField textField)
```

A dialog window pops up, so the user may choose a file (as an attachment to the e-mail).

### **Parameters:**

`jfc` - File chooser object

`textField` - The text field in which the filepath will be written

---

diplomatiki.gui

## Class EmailPanel

public class **EmailPanel**  
extends NextPanel

Title: Term Paper - EmailPanel class

Description: This is the Email Service panel. This panel is shown when the user chooses to send an e-mail.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### EmailPanel

```
public EmailPanel()
```

Creates a new `EmailPanel` instance.

---

### EmailPanel

```
public EmailPanel(JFrame myFrame,  
                  JPanel previousPanel)
```

Creates a new `EmailPanel` instance with the specified parent frame and previous panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### startAction

```
private void startAction()
```

Action performed when the "Start" service button is pressed.



---

## **startActualService**

```
private void startActualService()
```

The actual service is started. It runs in a separate thread.

---

## **startButton\_actionPerformed**

```
void startButton_actionPerformed(ActionEvent e)
```

Action performed when the "Start" service button is pressed. The `startAction()` method is invoked.

### **Parameters:**

e - `ActionEvent`

---

diplomatiki.gui

## Class `ExtendedButton`

```
public class ExtendedButton  
extends JButton
```

Title: Term Paper - `ExtendedButton` class

Description: This class is a sub-class of `JButton`. Buttons of this class have specific characteristics (e.g. color, border, etc). They also have a different icon when they are selected and when they are not.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### `ExtendedButton`

```
public ExtendedButton()
```

Creates a new `ExtendedButton` instance.

---

#### `ExtendedButton`

```
public ExtendedButton(String defaultIconPath,  
String selectedIconPath)
```

Creates a new `ExtendedButton` instance with the specified default icon and selected icon.

**Parameters:**

`defaultIconPath` - Icon to be used for the default state  
`selectedIconPath` - Icon to be used for the selected state

### Method Detail

#### `jbInit`

```
private void jbInit()
```

Completes the construction of the `ExtendedButton` instance.

**Throws:**

`Exception` - An exception is thrown if the construction of the button could not be completed

---

**this\_focusGained**

void **this\_focusGained**(FocusEvent e)

Action performed when the button gains focus.

**Parameters:**

e - FocusEvent

---

**this\_focusLost**

void **this\_focusLost**(FocusEvent e)

Action performed when the button loses focus.

**Parameters:**

e - FocusEvent

---

diplomatiki.gui

## Class ExtendedSlider

public class **ExtendedSlider**  
extends JSlider

Title: Term Paper - `ExtendedSlider` class

Description: This class is a sub-class of `JSlider`. Sliders of this class have specific characteristics (e.g. color, size, etc). They can also have a specific maximum and default value.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### ExtendedSlider

```
public ExtendedSlider()
```

Creates a new `ExtendedSlider` instance.

---

#### ExtendedSlider

```
public ExtendedSlider(int maxValue,  
                      String sliderName)
```

Creates a new `ExtendedSlider` instance with the specified maximum value and name.

**Parameters:**

`maxValue` - The slider's maximum value

`sliderName` - The slider's name

### Method Detail

#### setDefaultValue

```
public void setDefaultValue(int defaultValue)
```

Sets the slider's default value.

**Parameters:**

`defaultValue` - The default value

---

**getDefaultValue**

```
public int getDefaultValue()
```

Returns the slider's default value.

**Returns:**

The slider's default value

---

**jbInit**

```
private void jbInit()
```

Builds the slider.

**Throws:**

`Exception` - An exception is thrown if the construction of the slider is not successful

---

diplomatiki.gui

## Class GeneralProfileSettingsPanel

public class **GeneralProfileSettingsPanel**  
extends JPanel

Title: Term Paper - GeneralProfileSettingsPanel class

Description: This is part (the north panel) of the ProfilePanel. It holds and presents the user's general settings.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### GeneralProfileSettingsPanel

```
public GeneralProfileSettingsPanel()
```

Creates a new GeneralProfileSettingsPanel instance.

---

### GeneralProfileSettingsPanel

```
public GeneralProfileSettingsPanel(String factor1,  
                                   String factor2,  
                                   String factor3,  
                                   String[] providers,  
                                   int selectedIndex,  
                                   int otherProviderCode)
```

Creates a new GeneralProfileSettingsPanel instance. The first three parameters represent the ordered list of factors according to the user preferences. They can take the following values: "Quality", "Cost" or "Provider". The fourth argument is a list of providers. The fifth argument represents the position of the selected (preferred) provider in the aforementioned list. The final argument is the code of the preferred provider, in case it is not one of the predefined providers (and as a result it is not in the aforementioned list). These pieces of information are presented in the GUI, and are also editable.

### Parameters:

`factor1` - The first (most important) factor according the user's current preferences

`factor2` - The second factor according the user's current preferences

`factor3` - The third (least important) factor according the user's current preferences

`providers` - A list of predefined provider names

`selectedIndex` - The position (in the previous list) of the preferred provider  
`otherProviderCode` - The preferred provider's code (in case it is not included in the providers list)

---

---

## Method Detail

### **jbInit**

`void jbInit()`

Builds the panel.

#### **Throws:**

`Exception` - An exception is thrown if the construction of the panel is not successful

---

### **hideExtraFields**

`void hideExtraFields()`

Hides some extra fields in the panel. This happens in case the preferred provider is included in the list of predefined providers.

---

### **showExtraFields**

`void showExtraFields()`

Shows some extra fields in the panel. This happens in case the preferred provider is not included in the list of predefined providers.

---

### **providerBox\_itemStateChanged**

`void providerBox_itemStateChanged(ItemEvent e)`

Action performed when the state of the drop-down box with the provider names is changed. It invokes either the `showExtraFields()` or the `hideExtraFields()` method.

#### **Parameters:**

`e` - `ItemEvent`

---

### **up\_actionPerformed**

`void up_actionPerformed(ActionEvent e)`

Action performed if the "Up" button is pressed.

**Parameters:**

e - `ActionEvent`

---

**down\_actionPerformed**

`void down_actionPerformed(ActionEvent e)`

Action performed if the "Down" button is pressed.

**Parameters:**

e - `ActionEvent`

---



diplomatiki.gui

## Class InfoPanel

public class **InfoPanel**  
extends NextPanel

Title: Term Paper - InfoPanel class

Description: This panel presents the details of an open connection. A `ConnectionInfoPanel` is used as its center panel. It also presents information about the running service.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### InfoPanel

```
public InfoPanel()
```

Creates a new `InfoPanel` instance.

---

### InfoPanel

```
public InfoPanel(JFrame myFrame,  
                 JPanel previousPanel,  
                 String messageLabelInput,  
                 String textInfoInput)
```

Creates a new `InfoPanel` instance with the specified parent frame and previous panel. The last two arguments are pieces of information that should be printed in the panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

`messageLabelInput` - A message that should be printed in the panel. Usually, it's service-specific

`textInfoInput` - Textual information that should also be presented

## Method Detail

## **changeAnimatedIcon**

```
public void changeAnimatedIcon(String newIconPath)
```

Changes the default animated icon in the panel with the one specified.

### **Parameters:**

`newIconPath` - The path of the new animated icon

---

diplomatiki.gui

## Class InterfaceAdaptationLogFrame

public class **InterfaceAdaptationLogFrame**  
extends JFrame

Title: Term Paper - `InterfaceAdaptationLogFrame` class

Description: This is a frame used as a log output area by the Interface Adaptation (IA) module.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### InterfaceAdaptationLogFrame

```
public InterfaceAdaptationLogFrame()
```

Creates a new `InterfaceAdaptationLogFrame` instance.

## Method Detail

### **jbInit**

```
void jbInit()
```

Builds the frame.

### Throws:

`Exception` - An exception is thrown if the construction of the frame is not successful

### **printTimer**

```
public void printTimer(int timer)
```

Prints the value of the `timer` parameter in the log area.

### Parameters:

`timer` - The value that should be printed in the log area



diplomatiki.gui

## Class MenuApplication

public class **MenuApplication**

Title: Term Paper - MenuApplication class

Description: An instance of this class constructs and centers on screen a MenuFrame instance.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### MenuApplication

```
public MenuApplication()
```

Apart from creating a new MenuApplication instance, this constructor creates a new MenuFrame instance and centers it on screen.

### Method Detail

#### main

```
public static void main(String[] args)
```

This method is useful in case one wants to construct and launch directly the MenuApplication.

**Parameters:**

args - No command line arguments are necessary

diplomati.gui

## Class MenuFrame

```
public class MenuFrame
extends JFrame
```

Title: Term Paper - MenuFrame class

Description: This is the GUI's main frame. All panels that are shown or hidden belong to this frame. When first launched, the content panel of this frame holds the application's main menu. This menu consists of nine (9) buttons.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### MenuFrame

```
public MenuFrame ()
```

Creates a new MenuFrame instance.

### Method Detail

#### getMobilityManager

```
public MobilityManager getMobilityManager ()
```

Returns a reference to the Mobility Manager module of the TMS.

**Returns:**

A reference to the Mobility Manager module (that is a reference to a MobilityManager object)

#### getVoiceCallPanel

```
public VoiceCallPanel getVoiceCallPanel ()
```

Returns a reference to the VoiceCallPanel.

**Returns:**

A reference to a `VoiceCallPanel` object

---

**getVideoCallPanel**

```
public VideoCallPanel getVideoCallPanel()
```

Returns a reference to the `VideoCallPanel`.

**Returns:**

A reference to a `VideoCallPanel` object

---

**getVideoStreamPanel**

```
public VideoStreamPanel getVideoStreamPanel()
```

Returns a reference to the `VideoStreamPanel`.

**Returns:**

A reference to a `VideoStreamPanel` object

---

**getWebBrowsingPanel**

```
public WebBrowsingPanel getWebBrowsingPanel()
```

Returns a reference to the `WebBrowsingPanel`.

**Returns:**

A reference to a `WebBrowsingPanel` object

---

**refreshAppropriateConnectionInfo**

```
public void refreshAppropriateConnectionInfo(int serviceCode,  
                                             InfoStruct myInfo)
```

Refreshes (updates) the connectivity information shown by the GUI. The `serviceCode` defines which service's connection info should be updated. The `myInfo` object holds the new information.

**Parameters:**

`serviceCode` - The code of the service for which the info should be updated  
`myInfo` - The new info

---

## **jbInit**

```
private void jbInit()
```

Builds the frame.

### **Throws:**

`Exception` - An exception is thrown if the construction of the frame is not successful

---

## **processWindowEvent**

```
protected void processWindowEvent(WindowEvent e)
```

This method is overridden so we can exit when the window is closed.

### **Parameters:**

e - WindowEvent

---

## **showNextPanel**

```
private void showNextPanel(JPanel np)
```

Hides the current panel and makes visible the np panel.

### **Parameters:**

np - The new panel that will be shown

---

## **reConstructApplicationsPanel**

```
public void reConstructApplicationsPanel()
```

Re-constructs the applications panel. In other words, it creates a new `ApplicationsPanel` instance. Special care is taken for whether or not the new applications panel should be made visible or not.

---

## **keyNavigation**

```
void keyNavigation(KeyEvent e)
```

This method is used to assist in navigating the main menu with the keyboard arrows.

### **Parameters:**

e - KeyEvent

---



diplomatiki.gui

## Class MessageWindow

```
public class MessageWindow
extends JWindow
implements Runnable
```

Title: Term Paper - MessageWindow class

Description: Instances of this class are windows that appear for only a few seconds and show a message to the user (e.g. a "Service Unavailable" message). They run as separate threads.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### MessageWindow

```
public MessageWindow()
```

Creates a new MessageWindow instance.

---

#### MessageWindow

```
public MessageWindow(JFrame parentFrame)
```

Creates a new MessageWindow instance with the specified parent frame.

**Parameters:**

parentFrame - The window's parent frame

---

#### MessageWindow

```
public MessageWindow(JFrame parentFrame,
                      String northString,
                      String centerString)
```

Creates a new MessageWindow instance with the specified parent frame and message.

**Parameters:**

`parentFrame` - The window's parent frame  
`northString` - The window's title message  
`centerString` - The window's main message

---

## Method Detail

---

### **jbInit**

```
private void jbInit()
```

Builds the window.

#### **Throws:**

`Exception` - An exception is thrown if the construction of the window is not successful

---

### **centerTheWindow**

```
private void centerTheWindow()
```

Centers the window on screen.

#### **Throws:**

`HeadlessException` - Exception that could be thrown

---

### **run**

```
public void run()
```

Entry point for the `MessageWindow` thread. It makes the window visible, waits for a few seconds, and then hides the window.

---

diplomatiki.gui

## Class MMSInputPanel

```
public class MMSInputPanel  
extends JPanel
```

Title: Term Paper - MMSInputPanel class

Description: This is the center panel of the MMS Service panel. It holds the various input fields, where the user should fill in the appropriate information (e.g. the recipient's address).

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### MMSInputPanel

```
public MMSInputPanel ()
```

Creates a new MMSInputPanel instance.

## Method Detail

### jbInit

```
void jbInit ()
```

Builds the panel.

### Throws:

Exception - An exception is thrown if the construction of the panel is not successful

### load1\_actionPerformed

```
void load1_actionPerformed (ActionEvent e)
```

Action performed when the first "Load" button is pressed. The loadFile () method is invoked.

### Parameters:

e - `ActionEvent`

---

## **load2\_actionPerformed**

```
void load2_actionPerformed(ActionEvent e)
```

Action performed when the second "Load" button is pressed. The `loadFile()` method is invoked.

### **Parameters:**

e - `ActionEvent`

---

## **loadFile**

```
void loadFile(JFileChooser jfc,  
             JTextField textField)
```

A dialog window pops up, so the user may choose a file (as an attachment to the e-mail).

### **Parameters:**

jfc - File chooser object

textField - The text field in which the filepath will be written

---

diplomatiki.gui

## Class MMSPanel

```
public class MMSPanel  
extends JPanel
```

Title: Term Paper - MMSPanel class

Description: This is the MMS Service panel. This panel is shown when the user chooses to send an MMS.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### MMSPanel

```
public MMSPanel ()
```

Creates a new MMSPanel instance.

---

### MMSPanel

```
public MMSPanel (JFrame myFrame,  
                JPanel previousPanel)
```

Creates a new MMSPanel instance with the specified parent frame and previous panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### startAction

```
private void startAction ()
```

Action performed when the "Start" service button is pressed.

---

## **settingsAction**

```
private void settingsAction()
```

Action performed when the "Edit Settings" button is pressed.

---

## **startButton\_actionPerformed**

```
void startButton_actionPerformed(ActionEvent e)
```

Action performed when the "Start" service button is pressed. The `startAction()` method is invoked.

### **Parameters:**

e - `ActionEvent`

---

## **settingsButton\_actionPerformed**

```
void settingsButton_actionPerformed(ActionEvent e)
```

Action performed when the "Edit Settings" button is pressed. The `settingsAction()` method is invoked.

### **Parameters:**

e - `ActionEvent`

---

diplomatiki.gui

## Class NextPanel

```
public class NextPanel
extends JPanel
```

Title: Term Paper - NextPanel class

Description: Most panels inherit from this class. Panels of this class have specific characteristics (e.g. color, sections, etc). They have a north panel, a center panel and a south panel. In the south panel, there is a "Back" button, which allows the user to return to the previous panel.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### NextPanel

```
public NextPanel ()
```

Creates a new NextPanel instance.

---

#### NextPanel

```
public NextPanel (JFrame myFrame,
                  JPanel previousPanel)
```

Creates a new NextPanel instance with the specified parent frame and previous panel.

**Parameters:**

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

### Method Detail

#### jbInit

```
void jbInit ()
```

Builds the panel.

**Throws:**

`Exception` - An exception is thrown if the construction of the panel is not successful

---

**backButton\_actionPerformed**

`void backButton_actionPerformed(ActionEvent e)`

Action performed when the "Back" button is pressed. The current panel is hidden and the previous panel is shown.

**Parameters:**

e - `ActionEvent`

---



diplomatiki.gui

## Class PINApplication

public class **PINApplication**

Title: Term Paper - `PINApplication` class

Description: An instance of this class constructs and centers on screen a `PINFrame` instance.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### **PINApplication**

public **PINApplication**()

Apart from creating a new `PINApplication` instance, this constructor creates a new `PINFrame` instance and centers it on screen.

### Method Detail

#### **main**

public static void **main**(String[] args)

This method is useful in case one wants to construct and launch directly the `PINApplication`.

**Parameters:**

`args` - No command line arguments are necessary

diplomatiki.gui

## Class PINFrame

```
public class PINFrame  
extends JFrame
```

Title: Term Paper - PINFrame class

Description: This is the frame that allows the user to login. The user fills in their name and password. The password can also be changed.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### PINFrame

```
public PINFrame ()
```

Creates a new PINFrame instance.

### Method Detail

#### jbInit

```
private void jbInit ()
```

Builds the frame.

**Throws:**

Exception - An exception is thrown if the construction of the frame is not successful

#### processWindowEvent

```
protected void processWindowEvent (WindowEvent e)
```

This method is overridden so we can exit when the window is closed.

**Parameters:**

e - WindowEvent

---

## loadPropertiesFile

```
private Properties loadPropertiesFile(String filePath)
```

Loads a Properties file.

### Parameters:

`filePath` - The path of the properties file

### Returns:

A `Properties` instance that represents the properties file's contents

---

## storePropertiesFile

```
private void storePropertiesFile(Properties config,  
                                String filePath)
```

Writes the property list (key and element pairs) in the `config` `Properties` table to the file specified.

### Parameters:

`config` - A `Properties` object

`filePath` - The path of the file in which the property list should be stored

---

## changeCheckBox\_actionPerformed

```
void changeCheckBox_actionPerformed(ActionEvent e)
```

Action performed when the state of the panel's check-box is changed. The fields where the user should type the new password are either hidden or shown.

### Parameters:

`e` - `ActionEvent`

---

## okButton\_actionPerformed

```
void okButton_actionPerformed(ActionEvent e)
```

Action performed when the "OK" button is pressed.

### Parameters:

`e` - `ActionEvent`

---

## correct

```
private boolean correct()
```

Returns `true` if the username/password typed is correct.

**Returns:**

`True` if the username/password typed is correct

---

**clear**

```
private void clear()
```

Clears all the text and password fields.

---

**clearNew**

```
private void clearNew()
```

Clears only the optional password fields.

---

**checkNewPassword**

```
private boolean checkNewPassword()
```

Checks if the user wants to change their password. Also, checks if the new password has been retyped correctly.

**Returns:**

`True` if the new password has been retyped correctly or if the user doesn't want to change their password.

---

**launchMenuApplication**

```
public void launchMenuApplication()
```

Launches the `MenuApplication`.

---

**cancelButton\_actionPerformed**

```
void cancelButton_actionPerformed(ActionEvent e)
```

Action performed when the "Cancel" button is pressed. The program is terminated.

**Parameters:**

e - ActionEvent

---

## Class ProfilePanel

public class **ProfilePanel**  
extends `NextPanel`

Title: Term Paper - `ProfilePanel` class

Description: This panel shows the user settings. It also allows the user to edit these settings. A panel of this type contains a `GeneralProfileSettingsPanel` in its north section.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### ProfilePanel

```
public ProfilePanel()
```

Creates a new `ProfilePanel` instance.

---

#### ProfilePanel

```
public ProfilePanel(JFrame myFrame,  
                    JPanel previousPanel,  
                    int profileCode)
```

Creates a new `ProfilePanel` instance with the specified parent frame and previous panel.

**Parameters:**

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

`profileCode` - The unique profile code that defines which of the predefined profiles (e.g. the "Current" profile, the "Low Cost" profile, etc.) has been selected and should be shown on screen

### Method Detail

#### loadPropertiesFile

```
private Properties loadPropertiesFile(String filePath)
```

Loads a Properties file.

**Parameters:**

filePath - The path of the properties file

**Returns:**

A Properties instance that represents the properties file's contents

---

## storePropertiesFile

```
private void storePropertiesFile(Properties config,  
                                String filePath)
```

Writes the property list (key and element pairs) in the config Properties table to the file specified.

**Parameters:**

config - A Properties object

filePath - The path of the file in which the property list should be stored

---

## saveNewSettings

```
private void saveNewSettings(Properties config)
```

Action performed when the user wants to save the changes made to their settings. It is invoked when the "Store" or the "Activate" button is pressed.

**Parameters:**

config - A Properties object which encapsulates the property list that should be saved

---

## findAllKnownProviders

```
private String[] findAllKnownProviders(Properties config)
```

Constructs a list with all known providers, so the user can select their favorite one.

**Parameters:**

config - The Properties object that contains the list with all known providers

**Returns:**

A table with all known providers

---

## findSelectedIndex

```
private int findSelectedIndex(String[] myArray,  
                              String name)
```

Finds the provider that the user prefers. It searches through the table `myArray` until it finds a match with `name`, and then returns its position. If there is no match, this method returns 0. This is the index of the choice "none" (the user doesn't have a favorite provider).

**Parameters:**

`myArray` - The table of known providers  
`name` - The name of the favorite provider

**Returns:**

The position of the favorite provider in the table

---

## **getDefaultSliderValues**

```
private void getDefaultSliderValues(Properties config)
```

Finds the sliders' default values. The sliders' default values are their initial values. They can be found in the user profile.

**Parameters:**

`config` - The `Properties` that contains the default values

---

## **setScrollBarValue**

```
private void setScrollBarValue(int newValue)
```

Sets the new value of the scroll bar. A scroll bar's value is reflected by its position.

**Parameters:**

`newValue` - The new value

---

## **voiceCallSlider\_focusGained**

```
void voiceCallSlider_focusGained(FocusEvent e)
```

Action performed when the Voice Call slider gains focus.

**Parameters:**

`e` - `FocusEvent`

---

## **mmsSlider\_focusGained**

```
void mmsSlider_focusGained(FocusEvent e)
```

Action performed when the MMS slider gains focus.

**Parameters:**



e - FocusEvent

---

### **videoCallSlider\_focusGained**

void **videoCallSlider\_focusGained**(FocusEvent e)

Action performed when the Video Call slider gains focus.

**Parameters:**

e - FocusEvent

---

### **videoStreamSlider\_focusGained**

void **videoStreamSlider\_focusGained**(FocusEvent e)

Action performed when the Video Stream slider gains focus.

**Parameters:**

e - FocusEvent

---

### **webBrowsingSlider\_focusGained**

void **webBrowsingSlider\_focusGained**(FocusEvent e)

Action performed when the Web Browsing slider gains focus.

**Parameters:**

e - FocusEvent

---

### **activateButton\_actionPerformed**

void **activateButton\_actionPerformed**(ActionEvent e)

Action performed when the "Activate" button is pressed.

**Parameters:**

e - ActionEvent

---

### **storeButton\_actionPerformed**

void **storeButton\_actionPerformed**(ActionEvent e)

Action performed when the "Store" button is pressed.

**Parameters:**

e - `ActionEvent`

---

**disableAllSlidersExceptOne**

void **disableAllSlidersExceptOne**(int sliderCode)

Disables all sliders except the one specified by `sliderCode`. Disabled sliders cannot be edited.

**Parameters:**

`sliderCode` - The code of the slider that should remain enabled

---

diplomatiki.gui

## Class ProgressBar

```
public class ProgressBar
extends JProgressBar
implements Runnable
```

Title: Term Paper - `ProgressBar` class

Description: This is a sub-class of `JProgressBar`. Progress bars of this type have specific characteristics (e.g. color). The progress bar is progressively painted and runs in a separate thread.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### ProgressBar

```
public ProgressBar()
```

Creates a new `ProgressBar` instance.

---

#### ProgressBar

```
public ProgressBar(int max)
```

Creates a new `ProgressBar` instance with the specified maximum value. This value represents the amount of time (in seconds) that will pass until the progress bar reaches 100%.

**Parameters:**

`max` - The progress bar's maximum value

---

### Method Detail

#### run

```
public void run()
```

Entry point for the `ProgressBar` thread. The `iterate()` method is called.

---

## **iterate**

```
private void iterate()
```

Progressively paints the progress bar (e.g. 0%, then 33%, then 66%, then 100%).

---

## **start**

```
public void start()
```

Starts the `ProgressBar` thread.

---

diplomatiki.gui

## Class RunningServicesInfoPanel

public class **RunningServicesInfoPanel**  
extends JPanel

Title: Term Paper - `RunningServicesInfoPanel` class

Description: This panel is a part of the applications panel (`ApplicationsPanel`). It contains a set of editable labels. These labels are used to present connection information.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### **RunningServicesInfoPanel**

```
public RunningServicesInfoPanel()
```

Creates a new `RunningServicesInfoPanel` instance.

### Method Detail

#### **jbInit**

```
void jbInit()
```

Builds the panel.

**Throws:**

`Exception` - An exception is thrown if the construction of the panel is not successful

#### **getLabel**

```
public JLabel getLabel(int index)
```

Returns the label specified by the method's argument (`index`).

**Parameters:**

`index` - The index of the label that should be returned

**Returns:**

A reference to the label specified

---

diplomatiki.gui

## Class SMSInputPanel

```
public class SMSInputPanel  
extends JPanel
```

Title: Term Paper - SMSInputPanel class

Description: This is the center panel of the SMS Service panel. It holds the various input fields, where the user should fill in the appropriate information (e.g. the recipient's address).

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### Constructor Detail

#### SMSInputPanel

```
public SMSInputPanel()
```

Creates a new SMSInputPanel instance.

### Method Detail

#### jbInit

```
void jbInit()
```

Builds the panel.

**Throws:**

Exception - An exception is thrown if the construction of the panel is not successful

diplomatiki.gui

## Class SMSPanel

public class **SMSPanel**  
extends JPanel

Title: Term Paper - `SMSPanel` class

Description: This is the SMS Service panel. This panel is shown when the user chooses to send an SMS.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### SMSPanel

```
public SMSPanel()
```

Creates a new `SMSPanel` instance.

---

### SMSPanel

```
public SMSPanel(JFrame myFrame,  
                JPanel previousPanel)
```

Creates a new `SMSPanel` instance with the specified parent frame and previous panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### startAction

```
private void startAction()
```

Action performed when the "Start" service button is pressed.



---

## **startButton\_actionPerformed**

`void startButton_actionPerformed(ActionEvent e)`

Action performed when the "Start" service button is pressed. The `startAction()` method is invoked.

### **Parameters:**

e - `ActionEvent`

---

diplomatiki.gui

## Class VideoCallPanel

```
public class VideoCallPanel
extends NextPanel
```

Title: Term Paper - VideoCallPanel class

Description: This is the Video Call Service panel. This panel is shown when the user chooses to make a video call.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### VideoCallPanel

```
public VideoCallPanel ()
```

Creates a new VideoCallPanel instance.

---

### VideoCallPanel

```
public VideoCallPanel (JFrame myFrame,
                        JPanel previousPanel)
```

Creates a new VideoCallPanel instance with the specified parent frame and previous panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### refreshConnectionInfo

```
public void refreshConnectionInfo (diplomatiki.InfoStruct myInfo)
```

Refreshes (updates) the presented connection information concerning this service. The new

information is contained in myInfo.

**Parameters:**

myInfo - The new information that should be presented

---

**startAction**

```
private void startAction()
```

Action performed when the "Start" service button is pressed.

---

**endAction**

```
private void endAction()
```

Action performed when the "End" service button is pressed.

---

**infoAction**

```
private void infoAction()
```

Action performed when the "View Info" button is pressed.

---

**settingsAction**

```
private void settingsAction()
```

Action performed when the "Edit Settings" button is pressed.

---

**startButton\_actionPerformed**

```
void startButton_actionPerformed(ActionEvent e)
```

Action performed when the "Start" service button is pressed. The startAction() method is invoked.

**Parameters:**

e - ActionEvent

---

**endButton\_actionPerformed**

```
void endButton_actionPerformed(ActionEvent e)
```

Action performed when the "End" service button is pressed. The `endAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **infoButton\_actionPerformed**

```
void infoButton_actionPerformed(ActionEvent e)
```

Action performed when the "View Info" button is pressed. The `infoAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **settingsButton\_actionPerformed**

```
void settingsButton_actionPerformed(ActionEvent e)
```

Action performed when the "Edit Settings" button is pressed. The `settingsAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

diplomatiki.gui

## Class VideoStreamPanel

```
public class VideoStreamPanel
extends JPanel
```

Title: Term Paper - VideoStreamPanel class

Description: This is the Video Stream Service panel. This panel is shown when the user chooses to receive a video stream.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### VideoStreamPanel

```
public VideoStreamPanel ()
```

Creates a new VideoStreamPanel instance.

---

### VideoStreamPanel

```
public VideoStreamPanel (JFrame myFrame,
                          JPanel previousPanel)
```

Creates a new VideoStreamPanel instance with the specified parent frame and previous panel.

### Parameters:

`myFrame` - The frame which this panel belongs to

`previousPanel` - The previous panel that was visible in the `myFrame` frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### refreshConnectionInfo

```
public void refreshConnectionInfo (diplomatiki.InfoStruct myInfo)
```

Refreshes (updates) the presented connection information concerning this service. The new information is contained in myInfo.

**Parameters:**

myInfo - The new information that should be presented

---

**startAction**

```
private void startAction()
```

Action performed when the "Start" service button is pressed.

---

**endAction**

```
private void endAction()
```

Action performed when the "End" service button is pressed.

---

**infoAction**

```
private void infoAction()
```

Action performed when the "View Info" button is pressed.

---

**settingsAction**

```
private void settingsAction()
```

Action performed when the "Edit Settings" button is pressed.

---

**startButton\_actionPerformed**

```
void startButton_actionPerformed(ActionEvent e)
```

Action performed when the "Start" service button is pressed. The startAction() method is invoked.

**Parameters:**

e - ActionEvent

---

**endButton\_actionPerformed**

```
void endButton_actionPerformed(ActionEvent e)
```

Action performed when the "End" service button is pressed. The `endAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **infoButton\_actionPerformed**

```
void infoButton_actionPerformed(ActionEvent e)
```

Action performed when the "View Info" button is pressed. The `infoAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **settingsButton\_actionPerformed**

```
void settingsButton_actionPerformed(ActionEvent e)
```

Action performed when the "Edit Settings" button is pressed. The `settingsAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

diplomatiki.gui

## Class VoiceCallPanel

```
public class VoiceCallPanel
extends NextPanel
```

Title: Term Paper - VoiceCallPanel class

Description: This is the Voice Call Service panel. This panel is shown when the user chooses to make a voice call.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### VoiceCallPanel

```
public VoiceCallPanel()
```

Creates a new VoiceCallPanel instance.

---

### VoiceCallPanel

```
public VoiceCallPanel(JFrame myFrame,
                        JPanel previousPanel)
```

Creates a new VoiceCallPanel instance with the specified parent frame and previous panel.

### Parameters:

myFrame - The frame which this panel belongs to

previousPanel - The previous panel that was visible in the myFrame frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### refreshConnectionInfo

```
public void refreshConnectionInfo(diplomatiki.InfoStruct myInfo)
```

Refreshes (updates) the presented connection information concerning this service. The new



information is contained in myInfo.

**Parameters:**

myInfo - The new information that should be presented

---

**startAction**

```
private void startAction()
```

Action performed when the "Start" service button is pressed.

---

**endAction**

```
private void endAction()
```

Action performed when the "End" service button is pressed.

---

**infoAction**

```
private void infoAction()
```

Action performed when the "View Info" button is pressed.

---

**settingsAction**

```
private void settingsAction()
```

Action performed when the "Edit Settings" button is pressed.

---

**startButton\_actionPerformed**

```
void startButton_actionPerformed(ActionEvent e)
```

Action performed when the "Start" service button is pressed. The startAction() method is invoked.

**Parameters:**

e - ActionEvent

---

**endButton\_actionPerformed**

```
void endButton_actionPerformed(ActionEvent e)
```

Action performed when the "End" service button is pressed. The `endAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **infoButton\_actionPerformed**

```
void infoButton_actionPerformed(ActionEvent e)
```

Action performed when the "View Info" button is pressed. The `infoAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **settingsButton\_actionPerformed**

```
void settingsButton_actionPerformed(ActionEvent e)
```

Action performed when the "Edit Settings" button is pressed. The `settingsAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

diplomatiki.gui

## Class WebBrowsingPanel

```
public class WebBrowsingPanel
extends NextPanel
```

Title: Term Paper - WebBrowsingPanel class

Description: This is the Web Browsing Service panel. This panel is shown when the user chooses to browse the web.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

## Constructor Detail

### WebBrowsingPanel

```
public WebBrowsingPanel ()
```

Creates a new WebBrowsingPanel instance.

---

### WebBrowsingPanel

```
public WebBrowsingPanel (JFrame myFrame,
                          JPanel previousPanel)
```

Creates a new WebBrowsingPanel instance with the specified parent frame and previous panel.

### Parameters:

myFrame - The frame which this panel belongs to

previousPanel - The previous panel that was visible in the myFrame frame. This reference is very useful if the user wants to return to the previous panel

## Method Detail

### refreshConnectionInfo

```
public void refreshConnectionInfo (diplomatiki.InfoStruct myInfo)
```

Refreshes (updates) the presented connection information concerning this service. The new information is contained in myInfo.

**Parameters:**

myInfo - The new information that should be presented

---

**startAction**

```
private void startAction()
```

Action performed when the "Start" service button is pressed.

---

**endAction**

```
private void endAction()
```

Action performed when the "End" service button is pressed.

---

**infoAction**

```
private void infoAction()
```

Action performed when the "View Info" button is pressed.

---

**settingsAction**

```
private void settingsAction()
```

Action performed when the "Edit Settings" button is pressed.

---

**startButton\_actionPerformed**

```
void startButton_actionPerformed(ActionEvent e)
```

Action performed when the "Start" service button is pressed. The startAction() method is invoked.

**Parameters:**

e - ActionEvent

---

**endButton\_actionPerformed**

```
void endButton_actionPerformed(ActionEvent e)
```

Action performed when the "End" service button is pressed. The `endAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **infoButton\_actionPerformed**

```
void infoButton_actionPerformed(ActionEvent e)
```

Action performed when the "View Info" button is pressed. The `infoAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

### **settingsButton\_actionPerformed**

```
void settingsButton_actionPerformed(ActionEvent e)
```

Action performed when the "Edit Settings" button is pressed. The `settingsAction()` method is invoked.

**Parameters:**

e - `ActionEvent`

---

diplomatiki.services

## Class **Mailer**

public class **Mailer**

Title: Term Paper - `Mailer` class

Description: This class implements the Email service.

Copyright: Copyright (c) 2005

Company: Jadacode

### **Version:**

1.0

### **Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

### **See Also:**

`MailerService`

## Constructor Detail

### **Mailer**

```
public Mailer()
```

Creates a new `Mailer` instance.

---

### **Mailer**

```
public Mailer(String recipient,  
              String subject,  
              String body)
```

Creates a new `Mailer` instance with the specified Recipient, Subject and Body as attributes.

### **Parameters:**

`recipient` - The recipient of the e-mail

`subject` - The e-mail's subject

`body` - The e-mail's message body

## Method Detail

### **sendEmail**

```
public void sendEmail(String aToEmailAddr,  
                      String aSubject,  
                      String aBody)
```

Sends a simple e-mail to the specified recipient. The message sent has the specified subject and body.

**Parameters:**

aToEmailAddr - The recipient's e-mail address  
aSubject - The e-mail's subject  
aBody - The message's body

---

**loadPropertiesFile**

```
private Properties loadPropertiesFile(String filePath)
```

Loads a Properties file.

**Parameters:**

filePath - The path of the properties file

**Returns:**

A `Properties` instance that represents the properties file's contents

---

diplomati.services

## Class `MailerService`

public class `MailerService`

extends `Mailer`

implements `Runnable`

Title: Term Paper - `MailerService` class

Description: This is a subclass of the `Mailer` class. An instance of this class runs as a separate thread and handles the process of sending an e-mail.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

### See Also:

`Mailer`

## Constructor Detail

### `MailerService`

```
public MailerService()
```

Creates a new `MailerService` instance.

---

### `MailerService`

```
public MailerService(String recipient,  
                        String subject,  
                        String body)
```

Creates a new `MailerService` instance with the specified Recipient, Subject and Body as attributes. The thread is also started.

### Parameters:

`recipient` - The recipient of the e-mail

`subject` - The e-mail's subject

`body` - The e-mail's message body

---

## Method Detail



## **run**

```
public void run()
```

This is the entry point for the `emailerService` thread. It sends an e-mail to each of the recipients defined in the `recipient` inherited instance variable, by using the `sendEmail()` method.

---

networkside.network

## Class NetworkEntity

public class **NetworkEntity**  
implements Runnable

Title: Term Paper - `NetworkEntity` class

Description: This is the main class implementing the Network Management System.

Copyright: Copyright (c) 2005

Company: Jadacode

**Version:**

1.0

**Author:**

Evgenia Adamopoulou - Konstantinos Demestichas

**See Also:**

`NetworkEntityService`

### Constructor Detail

#### NetworkEntity

```
public NetworkEntity()
```

Creates a new `NetworkEntity` instance. This instance runs in a separate thread and listens to a specific port for terminal requests.

### Method Detail

#### run

```
public void run()
```

Entry point for the `NetworkEntity` thread. When a new terminal request is received, a new `NetworkEntityService` instance/thread is constructed.

#### main

```
public static void main(String[] arguments)
```

This method initiates the Network Management System.

**Parameters:**

`arguments` - No command line arguments are necessary



networkside.network

## Class NetworkEntityService

public class **NetworkEntityService**  
extends Thread

Title: Term Paper - `NetworkEntityService` class

Description: An instance of this class has the responsibility to determine if a terminal's choice should be accepted or rejected. It runs in a separate thread.

Copyright: Copyright (c) 2005

Company: Jadacode

### Version:

1.0

### Author:

Evgenia Adamopoulou - Konstantinos Demestichas

### See Also:

`NetworkEntity`

## Constructor Detail

### NetworkEntityService

```
public NetworkEntityService()
```

Creates a new `NetworkEntityService` instance.

---

### NetworkEntityService

```
public NetworkEntityService(Socket client)
```

Creates a new `NetworkEntityService` instance attached to the specified client.

### Parameters:

`client` - The socket that should be used to communicate with the client

---

## Method Detail

### run

```
public void run()
```

Entry point for the `NetworkEntityService` thread. This method uses a random number generator in order to determine if the terminal's request should be accepted or rejected. The

rejection probability can easily be adjusted (e.g. 0%, 20%, etc.). In case of rejection, the `NetworkEntityService` thread awaits the terminal's next choice/request, and so on.

---