



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

*Σχεδίαση και Ανάπτυξη Συστήματος Νόμιμης και
Ασφαλούς Εξόρυξης Προσωπικών Δεδομένων από
Βάσεις Δεδομένων Τηλεπικοινωνιακών Παρόχων*

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΠΑΠΠΑΣ ΚΩΝΣΤΑΝΤΙΝΟΣ
ΠΟΥΛΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ

Επιβλέπων : Ιάκωβος Βενιέρης
Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Αύγουστος 2007



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

***Σχεδίαση και Ανάπτυξη Συστήματος Νόμιμης και
Ασφαλούς Εξόρυξης Προσωπικών Δεδομένων από
Βάσεις Δεδομένων Τηλεπικοινωνιακών Παρόχων***

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΠΑΠΠΑΣ ΚΩΝΣΤΑΝΤΙΝΟΣ
ΠΟΥΛΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ

Επιβλέπων : Ιάκωβος Βενιέρης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 27^η Αυγούστου 2007.

.....
Ιάκωβος Βενιέρης
Αναπλ. Καθηγητής Ε.Μ.Π.

.....
Δήμητρα Κακλαμάνη
Επικ. Καθηγήτρια Ε.Μ.Π.

.....
Γεώργιος Στασινόπουλος
Καθηγητής Ε.Μ.Π.

Αθήνα, Αύγουστος 2007

.....

ΠΑΠΠΑΣ Χ. ΚΩΝΣΤΑΝΤΙΝΟΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός & Μηχανικός Υπολογιστών Ε.Μ.Π.

.....

ΠΟΥΛΟΠΟΥΛΟΣ Π. ΓΕΩΡΓΙΟΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός & Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © ΠΟΥΛΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ, ΠΑΠΠΑΣ ΚΩΝΣΤΑΝΤΙΝΟΣ, 2007

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 ραγδαία ανάπτυξη στον τομέα των τηλεπικοινωνιών κατά την τελευταία εικοσαετία έδωσε μεγάλη ώθηση στον τρόπο με τον οποίο οι άνθρωποι αντιλαμβάνονται την μεταξύ τους επικοινωνία καθώς παρείχε τεράστιες διευκολύνσεις για την απομακρυσμένη επαφή δυο ή και περισσότερων προσώπων. Κυρίαρχο ρόλο σε αυτές τις εξελίξεις διαδραμάτισε η κινητή τηλεφωνία, παρέχοντας ένα εύκολο και ποιοτικά επαρκή τρόπο για την επικοινωνία ατόμων που βρίσκονται σε απομακρυσμένες μεταξύ τους περιοχές, αρκεί αυτές να καλύπτονταν από το τηλεφωνικό δίκτυο.

Δεν θα ήταν υπερβολή να χαρακτηρίσουμε το σημερινό τρόπο ζωής και επικοινωνίας ως εξαρτημένο από τις διευκολύνσεις της κινητής τηλεφωνίας. Ωστόσο μια τέτοια εξάρτηση δεν περικλείει μόνο θετικά σημεία αλλά εμπεριέχει και αρνητικές πλευρές, οι οποίες σχετίζονται με το γεγονός ότι τα ίδια οφέλη που προκύπτουν από τη χρήση της κινητής τηλεφωνίας έχουν χρησιμοποιηθεί από διάφορα άτομα ή ομάδες με κακόβουλους σκοπούς και για παράνομες δραστηριότητες.

Εγείρεται λοιπόν το θέμα της αποτροπής τέτοιων ενεργειών, κάτι που αποκτά ιδιαίτερες προεκτάσεις όσον αφορά τα ατομικά δικαιώματα και τις προσωπικές πολιτικές ελευθερίες, καθώς η αποκάλυψη των στοιχείων των επικοινωνιών φυσικών προσώπων ενδέχεται εξίσου εύκολα να χρησιμοποιηθεί για παράνομους σκοπούς. Τέθηκε λοιπόν η ανάγκη τόσο της διασφάλισης του απορρήτου των προσωπικών επικοινωνιών όσο και της άρσης αυτού όταν εθνικοί ή άλλοι λόγοι το επιβάλλουν, με τρόπο πάντοτε νόμιμο και αιτιολογημένο.

Η Ευρωπαϊκή Ένωση, μέσω της αρμόδιας επιτροπής της, εξέδωσε κοινοτικές οδηγίες που αφορούν στην υποχρέωση των τηλεπικοινωνιακών παρόχων να διατηρούν για ορισμένο χρονικό διάστημα τα στοιχεία των συνδιαλέξεων των συνδρομητών τους. Αυτές οι οδηγίες ενσωματώθηκαν στο εθνικό δίκαιο με τη μορφή Προεδρικού διατάγματος και ουσιαστικά υποχρεώνουν τους παρόχους να διατηρούν ευαίσθητα προσωπικά δεδομένα των συνδρομητών τους για 2 χρόνια προτού τα διαγράψουν.

Για να είναι δυνατή η εφαρμογή των κείμενων νομοθεσιών πρέπει να αναπτυχθεί ένα σύστημα που θα επιτρέπει την ασφαλή άρση του απορρήτου των επικοινωνιών, διατηρώντας αφενός μεν τα ειδικά δικαιώματα που απολαμβάνουν τα προσωπικά δεδομένα και διασφαλίζοντας αφετέρου ότι η άρση του απορρήτου θα γίνεται υπό συγκεκριμένες προϋποθέσεις και μόνο από τις αρμόδιες αρχές.

Στο κενό αυτό έρχεται να προτείνει μια λύση η παρούσα διπλωματική, περιγράφοντας τόσο τις προϋποθέσεις που οφείλουν να ισχύουν για την άρση του απορρήτου, όσο και περιγράφοντας την ανάλυση, σχεδίαση και υλοποίηση ενός τέτοιου συστήματος. Για τον σκοπό αυτό χρησιμοποιήθηκαν open source τεχνολογίες όπως MySQL, Advanced Encryption Standard (AES), Eclipse IDE, JBoss Application Server κ.α.

Λέξεις κλειδιά: **δικτυακή εφαρμογή, MySQL, Advanced Encryption Standard (AES), J2EE (Java 2 Enterprise Edition), Eclipse IDE, Enterprise Java Bean (EJB), Java Server Pages (JSP), JBoss Application Server, ψηφιακά πιστοποιητικά (digital certificates), πρωτοκόλλου HTTPS, Secure Sockets Layer (SSL), άρση του απορρήτου των προσωπικών δεδομένων και των στοιχείων των κλήσεων ενός συνδρομητή, αρχές που εμπλέκονται στην διαδικασία.**

Abstract

The rapid development in the telecommunications area over the past twenty years contributed greatly on how people perceive communication with each other and provided enormous ease for the remote contact between two or more persons. Outstanding was the contribution of mobile telephony to these changes, providing an easy and quality sufficient manner for two or more persons to contact each other, whilst being in far apart areas, as long as the telecommunications network provided coverage.

We wouldn't exaggerate to comment that today's lifestyle depends vastly on the comforts of mobile telephony. However, this dependence comes with many upsides and, of course with corresponding downsides. Certain aspects of the negative traits are related to the fact that the same benefits that derive from usage of mobile communications can be turned to serve malicious and criminal aspirations of persons.

This raises the issue of obstructing such misuse, however this is an issue that causes great distress concerning civil rights and political freedom, as the disclosure of private communications information can as easily be used for malevolent activities. This results to the imperative demand that the privacy of communication has to be protected, along side with the need to lift that protection when national or other reasons impose, though always under strict legal and vindicated justification.

The corresponding European Union committee gave out several directives concerning the obligatory preservation of subscriber's telecommunication conversations information for a certain period of time, by the telecommunications provider. The national legislation adopted these directives and compels providers to maintain private information of their subscribers for two years before destroying them.

To enforce the current legislative acts an application must be implemented to allow lawful interception of the classified communication information, securing the special privileges of personal information on the one hand and ensuring that lawful interception will be conducted under specific preconditions and only by the authorized authorities, on the other.

This topic is covered by this dissertation, as it not only describes the required preconditions for lawful interception but it provides analysis, design and implementation of such an application. For this purpose open source technologies and tools were used, such as MySQL, Advanced Encryption Standard (AES), Eclipse IDE, JBoss Application Server etc.

Key words: web application, MySQL, Advanced Encryption Standard (AES), J2EE (Java 2 Enterprise Edition), Eclipse IDE, Enterprise Java Bean (EJB), Java Server Pages (JSP), JBoss Application Server, digital certificates, HTTPS protocol, Secure Sockets Layer (SSL), lawful interception, authorized parties.

Πρόλογος

Η εκπόνηση της παρούσης διπλωματικής εργασίας πραγματοποιήθηκε το ακαδημαϊκό έτος 2006 – 2007 στη σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου υπό την επίβλεψη του Καθηγητή κου Ιάκωβου Βενιέρη, στον οποίο θέλουμε να εκφράσουμε τις ευχαριστίες μας για την δυνατότητα που μας έδωσε να ασχοληθούμε με το συγκεκριμένο θέμα, καθώς και για την άριστη συνεργασία μαζί μας.

Επιθυμούμε επίσης να εκφράσουμε τις ευχαριστίες μας προς τον υποψήφιο διδάκτορα της σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, κ. Γεώργιο Λιουδάκη, για την καθοδήγηση, την συνδρομή, τις παρατηρήσεις και το υλικό που μας παρείχε, παράγοντες καθοριστικούς για την περάτωση της διπλωματικής αυτής.

Οφείλουμε όμως να εκφράσουμε ιδιαιτέρως τις ευχαριστίες μας καθώς και την ευγνωμοσύνη μας προς τις οικογένειές μας για την αμέριστη συμπαράσταση και υποστήριξη που μας παρείχαν όλα αυτά τα χρόνια. Ελπίζουμε η παρούσα διπλωματική να αποτελεί και μια μικρή δικαίωση των κόπων και των θυσιών τους προς εμάς. Τέλος, ευχαριστούμε τους φίλους και συμφοιτητές μας, για όλα αυτά που μοιραστήκαμε.

ΠΕΡΙΕΧΟΜΕΝΑ

1 Εισαγωγή	13
1.1 Αντικείμενο της διπλωματικής	13
1.2 Οργάνωση του τόμου	14
2 Άρση Απορρήτου Επικοινωνιών (Lawful Interception).	15
2.1 Υποχρεώσεις Παρόχων Υπηρεσιών Επικοινωνιών.	15
2.2 Αρμόδιες Αρχές και Άρση Απορρήτου Επικοινωνιών.	16
3 Προδιαγραφές και Σχεδίαση Λογισμικού.....	17
3.1 Σκοπός Συστήματος.	17
3.2 Μοντέλο Επιχειρησιακού Σεναρίου.....	17
3.2.1 Δράστες (χρήστες).	17
3.2.1.1 Περίληψη	18
3.2.1.2 Διάγραμμα Δραστών (Actor Diagram).	18
3.2.1.3 Ορισμοί Χρηστών.....	19
3.2.2 Σενάρια Περιπτώσεων Χρήσης.	21
3.2.2.1 Σενάριο Εισαγωγής Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.	21
3.2.2.2 Σενάριο Επιβεβαίωσης Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.	23
3.2.2.3 Σενάριο Επισκόπησης Στοιχείων Συνδιαλέξεων έπειτα από Εισαγωγή Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.	25
3.2.2.4 Σενάριο χρήσης του συστήματος από χρήστη που έχει καθολικό ρόλο.	28
3.2.3 Διαγράμματα Περιπτώσεων Χρήσης.....	29
3.2.3.1 Σενάριο Εισαγωγής Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.	29
3.2.3.2 Σενάριο Επιβεβαίωσης Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.	30
3.2.3.3 Σενάριο Επισκόπησης Στοιχείων Συνδιαλέξεων έπειτα από Εισαγωγή Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.	31
3.3 Μοντέλο Πεδίου.....	31
3.3.1 Διάγραμμα κλάσεων Πεδίου.....	31
3.3.2 Ορισμός κύριας κλάσης πεδίου (Σύστημα).	32
3.4 Διαγράμματα Αλληλεπίδρασης.....	33
3.4.1 Σενάριο Εισαγωγής Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.....	33
3.4.2 Σενάριο Επιβεβαίωσης Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.....	35
3.4.3 Σενάριο Επισκόπησης Στοιχείων Συνδιαλέξεων έπειτα από Εισαγωγή Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.....	37
3.5 Σημαντικές Σχεδιαστικές Αποφάσεις.....	38
3.6 Αρχιτεκτονική Συστήματος – Λεπτομερή διαγράμματα κλάσεων.	40
3.6.1 Παραταξιακό διάγραμμα συστήματος.	40
3.6.2 Διαγράμματα κλάσεων UML.....	40
4 Τεχνολογίες και Εργαλεία που Χρησιμοποιήθηκαν.....	47
4.1 Τεχνικές Απαιτήσεις Συστήματος.....	47
4.2 Προγραμματιστικά Εργαλεία-Εφαρμογές.....	47
4.2.1 Εργαλείο Ανάπτυξης Εφαρμογής.....	47
4.2.2 Εξυπηρετητής Εφαρμογής.....	47
4.2.3 Εξυπηρετητής Διαδικτύου.....	48
4.2.4 Γλώσσα Προγραμματισμού.....	48

4.2.5 Βάση Δεδομένων (Database server)	48
4.2.6 Κρυπτογράφηση Δεδομένων	49
4.3 Τεχνολογίες που Χρησιμοποιήθηκαν.....	49
4.3.1 Java 2 Platform, Enterprise Edition (J2EE)	49
4.3.1.1 Εφαρμογές Πολλών Βαθμίδων.....	50
4.3.1.1.1 Πελάτες (J2EE Clients).....	51
4.3.1.1.2 Συστατικά Ιστού (Web Components).....	53
4.3.1.1.3 Επιχειρησιακά Συστατικά (Business Components)	54
4.3.1.2 J2EE Containers	55
4.3.1.2.1 Υπηρεσίες των Containers	55
4.3.1.2.2 Τύποι Containers	56
4.3.2 JavaServer Pages (JSP).....	57
4.3.2.1 Τρόπος λειτουργίας JSP σελίδων.	58
4.3.3 Enterprise JavaBeans (EJB).....	59
4.3.3.1 Τύποι Enterprise JavaBeans (EJBs)	59
4.3.3.1.1 Session Beans.	60
4.3.3.1.2 Entity Beans.....	61
4.3.3.1.3 Message-Driven Beans.....	62
4.3.3.2 Πρόσβαση Εξυπηρετούμενων με Interfaces.	63
4.3.3.3 Συστατικά των Enterprise JavaBeans (EJBs)	64
4.3.3.4 Κύκλοι Ζωής των Enterprise JavaBeans (EJBs)	65
4.3.4 Συσκευασία (Packaging).....	69
5 Υλοποίηση και Ρύθμιση Παραμέτρων.....	70
5.1 Υλοποίηση Εφαρμογής με JBossIDE Eclipse1.6.....	70
5.1.1 Δημιουργία EJB και Μεθόδων.	71
5.1.2 Δημιουργία των Αρχείων που σχετίζονται με το EJB.	75
5.1.3 Δημιουργία των Αρχείων που σχετίζονται με τα Συστατικά Ιστού (Web Components).....	83
5.1.4 Δημιουργία EAR Περιγραφών Διάταξης (EAR Deployment Descriptors).	89
5.1.5 Δημιουργία των Συστατικών Ιστού (Web Components).....	91
5.1.6 Συσκευασία Εφαρμογής (Packaging).	92
5.1.7 Ρύθμιση Παραμέτρων και Έναρξη Εξυπηρετητή (Launch & Deploy to JBoss).	93
5.2 Δημιουργία και Επεξήγηση των Πινάκων της Βάσης Δεδομένων MySQL.....	95
5.3 Ρύθμιση Παραμέτρων του JBoss για χρήση SSL.	98
5.4 Δημιουργία Ψηφιακών Πιστοποιητικών (Digital Certificates).....	100
6 Υπηρεσίες Εφαρμογής και Σενάρια Χρηστών	101
6.1 Είσοδος Χρηστών και Αποδοχή Ψηφιακού Πιστοποιητικού.....	101
6.2 Κύριο Μενού Εφαρμογής και Δικαιώματα Πρόσβασης Χρηστών.....	104
6.3 Υπηρεσίες Εφαρμογής.....	105
6.3.1 Εισαγωγή Αιτήματος	105
6.3.2 Αιτήματα που Εκκρεμούν και Έγκριση τους.....	106
6.3.3 Αιτήματα που Έχουν Εγκριθεί.....	108
7 Βιβλιογραφία	114

Ευρετήριο Σχημάτων

Σχήμα 3.1 διάγραμμα δραστών.....	19
Σχήμα 3.2 Διάγραμμα Ενεργειών για το σενάριο εισαγωγής αιτήματος 3.2.2.1.....	29
Σχήμα 3.3 Διάγραμμα Ενεργειών για το σενάριο επιβεβαίωσης αιτήματος 3.2.2.2 ...	30
Σχήμα 3.3 Διάγραμμα Ενεργειών για το σενάριο επισκόπησης στοιχείων	31
Σχήμα 3.4 Διάγραμμα Κύριων Κλάσεων Πεδίου	32
Σχήμα 3.5 Ακολουθιακό Διάγραμμα για το σενάριο εισαγωγής αιτήματος 3.2.2.1....	33
Σχήμα 3.6 Συνεργατικό Διάγραμμα για το σενάριο εισαγωγής αιτήματος 3.2.2.1	34
Σχήμα 3.7 Ακολουθιακό Διάγραμμα για το σενάριο επιβεβαίωσης αιτήματος 3.2.2.2	35
Σχήμα 3.8 Συνεργατικό Διάγραμμα για το σενάριο επιβεβαίωσης αιτήματος 3.2.2.2	36
Σχήμα 3.9 Ακολουθιακό Διάγραμμα για το σενάριο επισκόπησης αιτήματος 3.2.2.3	37
Σχήμα 3.10 Συνεργατικό Διάγραμμα για το σενάριο επισκόπησης αιτήματος 3.2.2.3	38
Σχήμα 3.11 Παραταξιακό Διάγραμμα Συστήματος.....	40
Σχήμα 3.12 Διάγραμμα κλάσεων ValidationBean.....	41
Σχήμα 3.13 Διάγραμμα κλάσεων Validation	41
Σχήμα 3.14 Διάγραμμα κλάσεων ValidationHome	41
Σχήμα 3.15 Διάγραμμα κλάσεων CallDataVO (α)	42
Σχήμα 3.16 Διάγραμμα κλάσεων CallDataVO (β)	43
Σχήμα 3.17 Διάγραμμα κλάσεων NameCodeGenericVO	43
Σχήμα 3.18 Διάγραμμα κλάσεων SubscriberDataVO (part 1)	44
Σχήμα 3.19 Διάγραμμα κλάσεων SubscriberDataVO (part 2)	44
Σχήμα 3.20 Διάγραμμα κλάσεων RequestVO (part 1)	45
Σχήμα 3.21 Διάγραμμα κλάσεων RequestVO (part 2)	46
Σχήμα 3.22 Διάγραμμα κλάσεων delfile.....	46
Σχήμα 4.1 Εφαρμογές πολλών βαθμίδων. Συστατικά J2EE.....	50
Σχήμα 4.2 Συστατικά στοιχεία της βαθμίδας του εξυπηρετητή.	53
Σχήμα 4.3 Βαθμίδα ιστού και J2EE εφαρμογές.	54
Σχήμα 4.4 Enterprise JavaBeans.....	54
Σχήμα 4.5 Τύποι Containers σε δομή εξυπηρετητή-εξυπηρετούμενου.....	56
Σχήμα 4.6 Τα interfaces για ένα Enterprise Bean με απομακρυσμένη πρόσβαση. ...	64
Σχήμα 4.7 α .Ο κύκλος ζωής ενός Stateful Session Bean.....	66
Σχήμα 4.8 β .Ο κύκλος ζωής ενός Stateless Session Bean	66
Σχήμα 4.9 γ .Ο κύκλος ζωής ενός Entity Bean.....	68
Σχήμα 4.10 δ .Ο κύκλος ζωής ενός Message-Driven Bean	68
Σχήμα 5.1 Επιλογή output folder	71
Σχήμα 5.2 Package Explorer tree structure.....	71
Σχήμα 5.3 Δημιουργία Session EJB	72
Σχήμα 5.4 Project tree structure after EJB creation	72
Σχήμα 5.5 Add Bussiness Method to EJB (a).....	73
Σχήμα 5.6 Add Bussiness Method to EJB (b).....	73
Σχήμα 5.7 Project Tree Structure after EJB methods have been created.....	74
Σχήμα 5.8 XDoclet Configurations (a)	75
Σχήμα 5.9 XDoclet Configurations (b).....	76
Σχήμα 5.10 XDoclet Configurations (c) – Ejbdoclet Configuration	77
Σχήμα 5.11 XDoclet Configurations (d) – Fileset Configuration.....	78
Σχήμα 5.12 XDoclet Configurations (e) – Deployment Descriptor Configuration	79
Σχήμα 5.13 XDoclet Configurations (f) – JBoss Configurations	80

Σχήμα 5.14 XDoclet Configurations (g) – Package Substitution Configurations	81
Σχήμα 5.15 XDoclet Configurations (h) – Interface Configurations.....	82
Σχήμα 5.16 Console Output for Run XDoclet Command	82
Σχήμα 5.17 Package Tree Structure after XDoclet build.....	83
Σχήμα 5.18 XDoclet Configuration for Webdoclet	84
Σχήμα 5.19 Webdoclet Configuration	85
Σχήμα 5.20 Webdoclet Configuration – fileset.....	86
Σχήμα 5.21 Webdoclet Configuration – deployment descriptor	87
Σχήμα 5.22 Webdoclet Configuration – jboss	88
Σχήμα 5.23 Webdoclet Configuration – Run XDoclet console output.....	88
Σχήμα 5.24 Updated Project Package Tree Structure	89
Σχήμα 5.25 Create application.xml (a)	89
Σχήμα 5.26 Create application.xml (b)	90
Σχήμα 5.27 Project final Package Tree Structure	91
Σχήμα 5.28 Packaging Configuration	92
Σχήμα 5.29 Launch and Deploy JBoss	93
Σχήμα 5.30 Deployment of ValidationApp.ear	94
Σχήμα 5.31 Selection of server for deployment of ValidationApp.ear.....	94
Σχήμα 6.1 Αποδοχή Ψηφιακού Πιστοποιητικού για χρήση SSL	102
Σχήμα 6.2 Login στην εφαρμογή.....	102
Σχήμα 6.3 Upload ψηφιακού πιστοποιητικού χρήστη.....	103
Σχήμα 6.4 Σφάλμα κατά την πιστοποίηση του ψηφιακού πιστοποιητικού	103
Σχήμα 6.5 Επιβεβαιωμένο ψηφιακό πιστοποιητικό.....	103
Σχήμα 6.6 Κύριο Μενού Εφαρμογής για πρόσβαση επιπέδου 4.....	104
Σχήμα 6.7 Κύριο Μενού Εφαρμογής για πρόσβαση επιπέδου 2.....	104
Σχήμα 6.8 Κύριο Μενού Εφαρμογής για πρόσβαση επιπέδου 1 και 2 και 3	105
Σχήμα 6.9 Εισαγωγή αιτήματος για χρήστη με πρόσβαση επιπέδου 4	105
Σχήμα 6.10 Εισαγωγή αιτήματος για χρήστη με πρόσβαση επιπέδου 1 και 2 και 3.....	106
Σχήμα 6.11 Σφάλμα κατά την εισαγωγή αιτήματος	106
Σχήμα 6.12 Εκκρεμή αιτήματα και έγκριση για χρήστη επιπέδου 4.....	107
Σχήμα 6.13 Εκκρεμή αιτήματα και έγκριση για χρήστη επιπέδου 2.....	107
Σχήμα 6.14 Εκκρεμή αιτήματα και έγκριση για χρήστη επιπέδου 1 και 2 και 3.....	107
Σχήμα 6.15 Επισκόπηση λόγου εισαγωγής αιτήματος	108
Σχήμα 6.16 Αιτήματα που έχουν εγκριθεί για χρήστη επιπέδου 4.....	109
Σχήμα 6.17 Αιτήματα που έχουν εγκριθεί για χρήστη επιπέδου 1 και 2 και 3.....	109
Σχήμα 6.18 Επισκόπηση Αιτήματος.....	110
Σχήμα 6.19 Επισκόπηση Αιτήματος.....	111
Σχήμα 6.20 Επισκόπηση Αιτήματος.....	112
Σχήμα 6.21 Επισκόπηση Αιτήματος.....	113

1

1 Εισαγωγή

Στο κεφάλαιο αυτό αναλύεται το αντικείμενο της παρούσας διπλωματικής εργασίας και επίσης περιγράφεται η οργάνωση του τόμου της διπλωματικής αυτής όπως ακολουθεί στα επόμενα κεφάλαια.

1.1 Αντικείμενο της διπλωματικής

Η διπλωματική αυτή έχει ως αντικείμενο τη δημιουργία μιας εφαρμογής η οποία θα προσφέρεται από μία εταιρία πάροχο επικοινωνιών και συγκεκριμένα κινητών επικοινωνιών φωνής. Η εφαρμογή αυτή θα έχει ως στόχο τον έλεγχο για την διασφάλιση ότι η Άρση του Απορρήτου Επικοινωνιών που θα αφορά κάποιον συνδρομητή της θα γίνεται σύμφωνα με έννομα μέσα και με βάση τη νομοθεσία όπως αυτή περιγράφεται από Προεδρικά διατάγματα και άλλα νομοθετικά άρθρα, αυτά περιγράφονται αναλυτικότερα στο 2 κεφάλαιο.

Λέγοντας έννομα μέσα εννοούμε πώς δικαίωμα για άρση του απορρήτου, και πρόσβαση -μέσω του αρχείου που διατηρεί η εταιρία- στις κλήσεις που έχει διεξάγει ή έχει αποδεχθεί ένας συνδρομητής της, έχουν μόνο ορισμένα πρόσωπα που ανήκουν σε κάποιες Αρχές υπεύθυνες και ορισμένες από το κράτος για το σκοπό αυτό. Επίσης για να γίνει Άρση του Απορρήτου δεν αρκεί μόνο το πρόσωπο που την ζητεί να ανήκει σε μία από αυτές τις αρχές αλλά θα πρέπει να δώσουν την έγκρισή τους και μέλη όλων των άλλων Αρχών. Αυτό σημαίνει πως θα πρέπει να έχουν συμφωνήσει, δηλαδή να έχουν δώσει την συγκατάθεσή τους, μέλη από τις όλες αυτές τις ανεξάρτητες αρχές ώστε να γίνει τελικά η Άρση του Απορρήτου Επικοινωνιών, πράγμα που ενισχύει την προστασία των προσωπικών δεδομένων και διασφαλίζει, όσο το δυνατόν, τη χωρίς λόγο παραβίαση των προσωπικών δεδομένων των πολιτών.

Επιπλέον μόνο ένας διαχειριστής θα υφίσταται για την εφαρμογή χωρίς αυτός να είναι άμεσα συνδεδεμένος με την εκάστοτε εταιρία, αποτρέποντας έτσι κατάχρηση από την ίδια την εταιρία και την πρόσβαση από προσωπικό της εταιρίας σε δεδομένα κλήσεων συνδρομητών της.

Ένας άλλος στόχος της διπλωματικής αυτής εργασίας είναι να διασφαλίσει πώς η αποθήκευση των δεδομένων που αφορούν κλήσεις συνδρομητών της καθώς και η διάθεση τους, μόνο σε περίπτωση Άρσης του Απορρήτου των Επικοινωνιών, διέπεται από κανόνες που εξασφαλίζουν τη μέγιστη δυνατή ασφάλεια. Για το σκοπό αυτό έχει χρησιμοποιηθεί κρυπτογραφία για την αποθήκευση των δεδομένων καθώς και χρήση της υπηρεσίας SSL.

1.2 Οργάνωση του τόμου

Ο τόμος της διπλωματικής αυτής εργασίας αποτελείται από 6 κεφάλαια.

Το **πρώτο κεφάλαιο** είναι η εισαγωγή η οποία περιγράφει το αντικείμενο της διπλωματικής, και επίσης αναφέρονται κάποιοι στόχοι της εφαρμογής καθώς και η χρήση για την οποία προορίζεται η εφαρμογή.

Το **δεύτερο κεφάλαιο** αναφέρεται στο τι είναι η Άρση του Απορρήτου Επικοινωνιών με λεπτομέρειες όσο αφορά τις υποχρεώσεις των εταιριών παροχής κινητών επικοινωνιών φωνής και τη συνεργασία τους με τις αρμόδιες Αρχές όπως αυτές προκύπτουν από τη μέχρι τώρα υπάρχουσα νομοθεσία.

Στο **τρίτο κεφάλαιο** δίνονται κάποιες προδιαγραφές της εφαρμογής, έτσι παρουσιάζονται τόσο οι χρήστες του συστήματος όσο και τα βασικά σενάρια περιπτώσεων χρήσης μοντελοποιημένα σύμφωνα με τη γλώσσα UML 2.0 που μοντελοποιούν και ορίζουν τις βασικές προδιαγραφές και παραδοχές του συστήματος.

Στο **τέταρτο κεφάλαιο** παρουσιάζονται οι απαιτήσεις υλικού (hardware) της εφαρμογής, τα προγραμματιστικά εργαλεία και οι εφαρμογές-προγράμματα που χρησιμοποιήθηκαν για την υλοποίηση και την υποστήριξη της, καθώς και οι τεχνολογίες JAVA που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής (J2EE, JSP, EJB κ.α.).

Το **πέμπτο κεφάλαιο** αναφέρεται στην υλοποίηση της εφαρμογής με χρήση κατάλληλων προγραμματιστικών εργαλείων, στο τρόπο δημιουργίας του Enterprise Java Bean που στηρίχθηκε η εφαρμογή, στη δημιουργία των κατάλληλων σχημάτων και πινάκων της βάσης δεδομένων που χρησιμοποιεί η εφαρμογή καθώς και στη παραμετροποίηση και τη ρύθμιση των προγραμμάτων-εφαρμογών που υποστηρίζουν την εφαρμογή. Επίσης αναφέρεται η δημιουργία και η χρήση Ψηφιακών Πιστοποιητικών που θα προσκομίζουν οι χρήστες για την είσοδό τους στις υπηρεσίες της εφαρμογής.

Το **έκτο κεφάλαιο** παρουσιάζει κάποια παραδείγματα-εκτελέσεις της εφαρμογής όπως αυτή παρουσιάζεται από την πλευρά των χρηστών. Παρουσιάζονται οι υπηρεσίες που προσφέρει η εφαρμογή στους χρήστες σε σχέση με τα δικαιώματα πρόσβασης του εκάστοτε χρήστη. Επίσης γίνεται εμφανής ο τρόπος αλληλεπίδρασης των χρηστών και η πρόσβαση των χρηστών στα δεδομένα των κλήσεων των συνδρομητών για τους οποίους έχει γίνει Άρση Απορρήτου Επικοινωνιών.

2

2 Άρση Απορρήτου Επικοινωνιών (Lawful Interception).

2.1 Υποχρεώσεις Παρόχων Υπηρεσιών Επικοινωνιών.

Σύμφωνα με την «ΟΔΗΓΙΑ 2006/24/ΕΚ ΤΟΥ ΕΥΡΩΠΑΪΚΟΥ ΚΟΙΝΟΒΟΥΛΙΟΥ ΚΑΙ ΤΟΥ ΣΥΜΒΟΥΛΙΟΥ της 15ης Μαρτίου 2006» καθώς και το «ΠΡΟΕΔΡΙΚΟ ΔΙΑΤΑΓΜΑ ΥΠ' ΑΡΙΘΜ. 47» οι εταιρίες παροχής υπηρεσιών επικοινωνιών φέρουν κάποιες υποχρεώσεις όσο αφορά τα δεδομένα των κλήσεων και την συνεργασία τους με τις νόμιμα αρμόδιες αρχές για τους σκοπούς της διερεύνησης, διαπίστωσης και δίωξης σοβαρών ποινικών αδικημάτων, όπως ορίζονται βάσει του εθνικού δικαίου της κάθε χώρας.

Οι εταιρίες παροχής υπηρεσιών επικοινωνιών είναι υποχρεωμένες να διαθέτουν αρχείο των κλήσεων που αφορά κλήσεις από και προς τους συνδρομητές της. Αυτό το αρχείο κλήσεων θα πρέπει να περιλαμβάνει δεδομένα κλήσεων σε βάθος χρόνου όχι μεγαλύτερο των δύο ετών και όχι μικρότερο των έξι μηνών (εκτός από τις κλήσεις που έχει ζητηθεί άρση του απορρήτου επικοινωνιών, τα δεδομένα των οποίων διατηρούνται) και θα πρέπει να περιλαμβάνει τα ακόλουθα:

- Καλών και καλούμενος αριθμός κλήσης και στις αναπάντητες κλήσεις.
- Καλών και καλούμενος συνδρομητής και πελάτης και στις αναπάντητες κλήσεις.
- το όνομα και η διεύθυνση του συνδρομητή ή του εγγεγραμμένου χρήστη.
- Ώρα έναρξης και ώρα λήξης της επικοινωνίας.
- Γεωγραφικός εντοπισμός καλούντος και καλούμενου (στις κινητές επικοινωνίες) είτε ομιλούν είτε πρόκειται για SMS , είτε είναι σε θέση stand-by, είτε πραγματοποιούν αναπάντητη κλήση.

- Η διεθνής ταυτότητα συνδρομητή κινητής τηλεφωνίας (IMSI) του καλούντος.
- Η διεθνής ταυτότητα εξοπλισμού κινητής τηλεφωνίας (IMEI) του καλούντος.
- Η διεθνής ταυτότητα συνδρομητή κινητής τηλεφωνίας (IMSI) του καλούμενου.
- Η διεθνής ταυτότητα εξοπλισμού κινητής τηλεφωνίας (IMEI) του καλούμενου.
- Οι αλφαριθμητικοί χαρακτήρες που εισάγει ο χρήστης για την πραγματοποίηση της σύνδεσης ή για την ενεργοποίηση ειδικών υπηρεσιών ή λειτουργιών.
- Η σηματοδότηση της ετοιμότητας για την πρόσβαση (προειδοποιητικό σήμα επικοινωνίας).
- Ο πραγματικός προορισμός και οι ενδιάμεσοι αριθμοί κλήσεως, σε περίπτωση εκτροπής της κλήσης.

2.2 Αρμόδιες Αρχές και Άρση Απορρήτου Επικοινωνιών.

Με τον όρο αρμόδια αρχή θεωρούμε ή Δικαστική ή άλλη Πολιτική, στρατιωτική ή αστυνομική δημόσια αρχή η οποία δικαιούται να υποβάλλει αίτηση για άρση του απορρήτου και να λάβει τα στοιχεία της επικοινωνίας. Για την άρση του απορρήτου επικοινωνιών θα πρέπει να υπάρχει επικύρωση των αιτημάτων μεταξύ των αρχών, επιβεβαίωση της αίτησης άρσης απορρήτου δηλαδή μίας αρχής από κάποια άλλη ή από περισσότερες της μίας, ώστε να διασφαλιστεί ότι τα δεδομένα καθίστανται διαθέσιμα μόνο για τους σκοπούς της διερεύνησης, διαπίστωσης και δίωξης σοβαρών ποινικών αδικημάτων, όπως ορίζονται βάσει του εθνικού δικαίου της κάθε χώρας.

Ο πάροχος πρέπει να παρέχει την δυνατότητα πολλαπλής και ταυτόχρονης διάθεσης των στοιχείων επικοινωνίας σε περισσότερες από μία αρμόδιες αρχές. Η εκτέλεση της διάταξης από τον πάροχο και η διαβίβαση των στοιχείων στην αρμόδια αρχή γίνεται με τρόπο που θα εξασφαλίζεται αξιοπιστία, εγκυρότητα, ακρίβεια, ταχύτητα και ασφάλεια. Στην διαδικασία αυτή εμπλέκεται και ο ελάχιστος αναγκαίος αριθμός εξουσιοδοτημένων προσώπων. Οι πάροχοι οφείλουν να λαμβάνουν όλα τα μέτρα για την διασφάλιση του απορρήτου κατά την εκτέλεση των διατάξεων άρσης απορρήτου που τους κοινοποιούνται.

Επίσης σε περίπτωση που ένας πάροχος υπηρεσίας χρησιμοποιεί μεθόδους κωδικοποίησης, συμπίεσης ή κρυπτογράφησης, υποχρεούται κατά την εκτέλεση μιας διάταξης να παραδίδει ή να διαβιβάζει τα ζητούμενα στοιχεία σε αποκωδικοποιημένη μορφή.

3

3 Προδιαγραφές και Σχεδίαση Λογισμικού

Σκοπός αυτού του κεφαλαίου είναι η παρουσίαση των προδιαγραφών κάτω από τις οποίες υλοποιήθηκε το λογισμικό σύστημα για την άρση του απορρήτου των επικοινωνιών από ένα τηλεπικοινωνιακό πάροχο. Παρουσιάζονται τόσο οι δράστες (χρήστες) του συστήματος όσο και τα βασικά σενάρια περιπτώσεων χρήσης μοντελοποιημένα σύμφωνα με τη γλώσσα UML 2.0 που μοντελοποιούν και ορίζουν τις βασικές προδιαγραφές και παραδοχές του συστήματος. Επίσης γίνεται εκτενής αναφορά στις κύριες σχεδιαστικές αποφάσεις που λήφθηκαν και παρατίθενται λεπτομερή διαγράμματα των κλάσεων που υλοποιήθηκαν και των μεθόδων που περιέχουν.

3.1 Σκοπός Συστήματος.

Ο στόχος που εξυπηρετεί το σύστημα λογισμικού για την υλοποίηση της ασφαλούς άρσης του απορρήτου των επικοινωνιών είναι να προσφέρει τόσο ένα ασφαλές δικτυακό περιβάλλον όσο και τα απαραίτητα εκείνα εχέγγυα που θα επιτρέψουν την άρση των προσωπικών δεδομένων ενός πελάτη ενός τηλεπικοινωνιακού παρόχου μόνο κάτω από τις απαραίτητες προϋποθέσεις όπως αυτές ορίζονται στους αντίστοιχους κανονισμούς της ευρωπαϊκής ένωσης, καθώς και στο σχετικό προεδρικό διάταγμα και αναφέρονται εκτενέστερα στο κεφάλαιο 2.

3.2 Μοντέλο Επιχειρησιακού Σεναρίου.

3.2.1 Δράστες (χρήστες).

3.2.1.1 Περίληψη

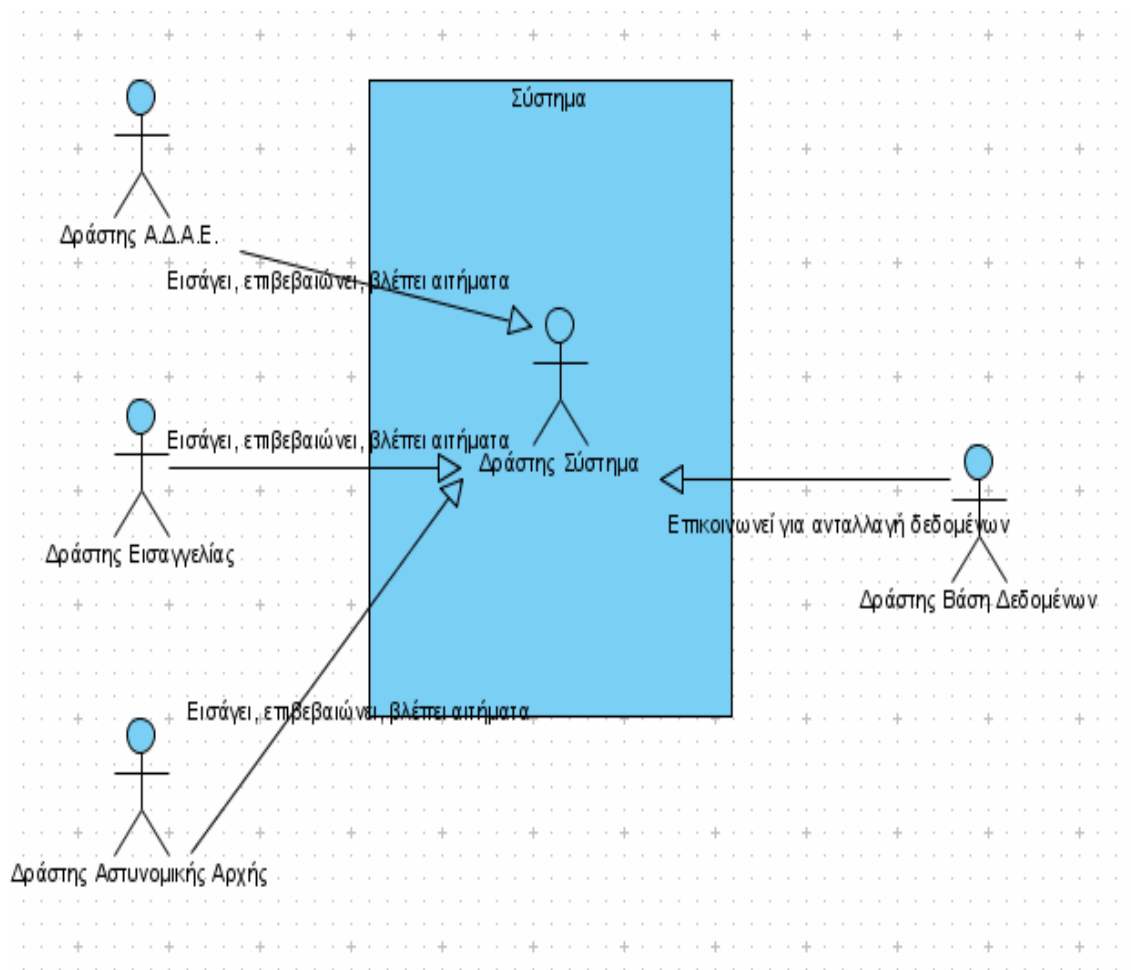
Ως δράστες της εφαρμογής ορίζουμε τόσο τα φυσικά πρόσωπα που χρησιμοποιούν τους διάφορους ρόλους για πρόσβαση στις λειτουργίες της παραγωγής όσο και τα ίδια τα συστήματα (όπως π.χ. την βάση δεδομένων) ή τα διάφορα υποσυστήματα που τα απαρτίζουν. Με δεδομένη λοιπόν την παραπάνω σύμβαση μπορούμε να χωρίσουμε το σύστημα σε διάφορες κατηγορίες δραστών ώστε να επιτύχουμε μια σαφή διάταξη που διευκολύνει την κατανόηση της εφαρμογής.

Ως δράστες λοιπόν θα πρέπει να θεωρηθούν κατ' αρχήν τα φυσικά πρόσωπα που θα «υποδύονται» διάφορους ρόλους στην εφαρμογή. Έτσι μπορούμε να διακρίνουμε ως δράστες την Αρχή Διασφάλισης Απορρήτου Επικοινωνιών (Α.Δ.Α.Ε.), την Εισαγγελία και την Αστυνομία. Οι συγκεκριμένες αρχές είναι αυτές που σε μια πρώτη θεώρηση θα πρέπει να συνεργάζονται ούτως ώστε να είναι δυνατή η άρση του απορρήτου των επικοινωνιών για κάποιο άτομο. Στους ρόλους δραστών που είναι υποδιδόμενοι από φυσικά πρόσωπα θα πρέπει να προστεθεί φυσικά και ο ρόλος του Διαχειριστή Βάσεως Δεδομένων και ο ρόλος του Διαχειριστή Συστήματος που αποτελούν τα πρόσωπα που φροντίζουν για την συνοχή και συνεχή λειτουργία της βάσης δεδομένων και του συστήματος αντίστοιχα.

Σαν συστήματα (ή υποσυστήματα) δράστες θα θεωρήσουμε δυο, την βάση δεδομένων και το σύστημα αυτό καθ' εαυτό, χωρίς να το χωρίσουμε σε υποσυστήματα. Ο λόγος που γίνεται αυτό δεν είναι άλλος από την ίδια την φύση της εφαρμογής που επιτρέπει διαφορετικά επίπεδα πρόσβασης σε άτομα των κατηγοριών που απαρτίζουν τους δράστες – φυσικά πρόσωπα. Η απαίτηση αυτή για την εφαρμογή κάνει δύσκολο το χώρισμα της εφαρμογής σε υποσυστήματα, καθώς δημιουργούνται πολλοί και διάφοροι συνδυασμοί που μπορούν να αλλάξουν δυναμικά. Επομένως ένας τέτοιος διαχωρισμός απλά θα προκαλούσε σύγχυση και θα δυσχέραινε αισθητά την κατανόηση της εφαρμογής και του ρόλου που επιτελεί.

3.2.1.2 Διάγραμμα Δραστών (Actor Diagram).

Παρακάτω παρουσιάζεται το διάγραμμα δραστών που μας δίνει μια συνολική εικόνα του συστήματος, καθώς και τις γενικές δράσεις που ακολουθούν οι διάφοροι δράστες σε αυτό:



Σχήμα 3.1 διάγραμμα δραστήων

3.2.1.3 Ορισμοί Χρηστών.

Χρήστης Αρχής Διασφάλισης Απορρήτου Επικοινωνιών:

Description	Χρήστης συστήματος που ανάλογα με τον ρόλο που του έχει ανατεθεί μπορεί να εισάγει αίτημα για άρση του απορρήτου των συνδιαλέξεων ενός συνδρομητή του τηλεπικοινωνιακού παρόχου, να δίνει την επιβεβαίωση εκ μέρους της Α.Δ.Α.Ε. σε ένα εκκρεμές αίτημα για την άρση του απορρήτου, να πραγματοποιεί την άρση απορρήτου σε ένα αίτημα που ο ίδιος ο χρήστης έχει εισάγει (δηλαδή να επιλέγει την εμφάνιση των στοιχείων των συνδιαλέξεων) είτε ακόμη να έχει την δυνατότητα να πραγματοποιεί την άρση του απορρήτου σε οποιοδήποτε αίτημα έχει εισαχθεί στο σύστημα.
Aliases	None
Inherits	None
Actor Type	Ενεργός δράστης συστήματος
Contact Person	None
Contact Details	None

Χρήστης Εισαγγελίας:

Description	Χρήστης συστήματος που ανάλογα με τον ρόλο που του έχει ανατεθεί μπορεί να εισάγει αίτημα για άρση του απορρήτου των συνδιαλέξεων ενός συνδρομητή του τηλεπικοινωνιακού παρόχου, να δίνει την επιβεβαίωση εκ μέρους της εισαγγελίας σε ένα εκκρεμές αίτημα για την άρση του απορρήτου, να πραγματοποιεί την άρση απορρήτου σε ένα αίτημα που ο ίδιος ο χρήστης έχει εισάγει (δηλαδή να επιλέγει την εμφάνιση των στοιχείων των συνδιαλέξεων) είτε ακόμη να έχει την δυνατότητα να πραγματοποιεί την άρση του απορρήτου σε οποιοδήποτε αίτημα έχει εισαχθεί στο σύστημα.
Aliases	None
Inherits	None
Actor Type	Ενεργός δράστης συστήματος
Contact Person	None
Contact Details	None

Χρήστης Αστυνομικής Αρχής:

Description	Χρήστης συστήματος που ανάλογα με τον ρόλο που του έχει ανατεθεί μπορεί να εισάγει αίτημα για άρση του απορρήτου των συνδιαλέξεων ενός συνδρομητή του τηλεπικοινωνιακού παρόχου, να δίνει την επιβεβαίωση εκ μέρους της αστυνομικής αρχής σε ένα εκκρεμές αίτημα για την άρση του απορρήτου, να πραγματοποιεί την άρση απορρήτου σε ένα αίτημα που ο ίδιος ο χρήστης έχει εισάγει (δηλαδή να επιλέγει την εμφάνιση των στοιχείων των συνδιαλέξεων) είτε ακόμη να έχει την δυνατότητα να πραγματοποιεί την άρση του απορρήτου σε οποιοδήποτε αίτημα έχει εισαχθεί στο σύστημα.
Aliases	None
Inherits	None
Actor Type	Ενεργός δράστης συστήματος
Contact Person	None
Contact Details	None

Σύστημα για την ασφαλή άρση του απορρήτου των επικοινωνιών σε συνδρομητή/ές παρόχου τηλεπικοινωνιών:

Description	Το σύστημα άρσης απορρήτου επικοινωνιών που με τις δομές του επιτρέπει σε εξουσιοδοτημένους και πιστοποιημένους χρήστες να εισάγουν αιτήματα για την άρση του απορρήτου σε φυσικά πρόσωπα, να επιβεβαιώσουν εκκρεμή αιτήματα, καθώς και να προχωρήσουν ουσιαστικά σε άρση του απορρήτου, εάν το αίτημα έχει δικαιολογηθεί και πιστοποιηθεί (εγκριθεί) από όλες τις συμμετέχουσες αρχές.
Aliases	Σύστημα, εφαρμογή
Inherits	None
Actor Type	Το ίδιο το σύστημα
Contact Person	None
Contact Details	None

Βάση Δεδομένων που περιέχει τα στοιχεία των κλήσεων των πελατών του παρόχου:

Description	Βάση δεδομένων που περιέχει σε κρυπτογραφημένη μορφή τα δεδομένα των κλήσεων των συνδρομητών του τηλεπικοινωνιακού παρόχου και που σε επικοινωνία με το σύστημα επιτρέπει την εισαγωγή αιτημάτων και την εξαγωγή των στοιχείων που προκύπτουν από αυτά σε κρυπτογραφημένη πάντα μορφή. Περιέχει επίσης πληροφορίες για τους χρήστες του συστήματος όπως τους κωδικούς και τα ειδικά ονόματα που οι χρήστες χρησιμοποιούν για την είσοδό τους στην εφαρμογή καθώς και τους ρόλους που έχουν ανατεθεί στον καθένα από αυτούς. Παρέχει επίσης τη δυνατότητα να γίνεται αλλαγή της ανάθεσης των ρόλων των χρηστών δυναμικά.
Aliases	Βάση, Βάση Δεδομένων
Inherits	None
Actor Type	Αποθήκη δεδομένων
Contact Person	None
Contact Details	None

3.2.2 Σενάρια Περιπτώσεων Χρήσης.

3.2.2.1 Σενάριο Εισαγωγής Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

Περιγραφή:

Ένας χρήστης – είτε μέλος της Α.Δ.Α.Ε., είτε της Εισαγγελίας, είτε της Αστυνομικής αρχής – επιθυμεί να συνδεθεί στο σύστημα για να εισάγει ένα αίτημα σχετικά με την άρση του απορρήτου των συνδιαλέξεων ενός φυσικού προσώπου, συνδρομητή του συγκεκριμένου παρόχου τηλεπικοινωνιών που αφορά το σύστημα.

Δράστες:

- Χρήστης αρχής
- Σύστημα
- Βάση δεδομένων

Προϋποθέσεις:

- i. Ο χρήστης θα πρέπει να είναι εξουσιοδοτημένος να χρησιμοποιεί το σύστημα, δηλαδή θα πρέπει να έχει λάβει το απαραίτητο ψηφιακό πιστοποιητικό που θα πιστοποιεί ότι είναι εγκεκριμένος να επιτελεί μια από τις εργασίες του συστήματος.
- ii. Ο χρήστης θα πρέπει να υπάρχει στη βάση δεδομένων του συστήματος στον αντίστοιχο πίνακα χρηστών με συγκεκριμένο όνομα και κωδικό πρόσβασης που θα επιτρέψει και την ταυτοποίησή του από το σύστημα με την αποκωδικοποίηση του ψηφιακού πιστοποιητικού.

- iii. Στον χρήστη θα πρέπει να έχει ανατεθεί επίσης ο συγκεκριμένος ρόλος της εισαγωγής αιτημάτων για την άρση του απορρήτου των επικοινωνιών για φυσικά πρόσωπα στον αντίστοιχο πίνακα ρόλων της βάσης δεδομένων, ούτως ώστε να είναι δυνατή η πρόσβασή του στο συγκεκριμένο κομμάτι της εφαρμογής.

Κείμενο Σεναρίου:

1. Ο χρήστης συνδέεται στην εφαρμογή εισάγοντας στον browser του τερματικού του το url της εφαρμογής το οποίο για λόγους ασφαλείας απαιτεί https σύνδεση.
2. Στην αρχική σελίδα απαιτείται να εισάγει το αναγνωριστικό του όνομα για το σύστημα και τον ειδικό αντίστοιχο κωδικό πρόσβασης.
3. Μετέπειτα καλείται να εισάγει το ψηφιακό πιστοποιητικό που του έχει δοθεί έτσι ώστε να είναι δυνατή η επιβεβαίωση ότι είναι ο συγκεκριμένος εγκεκριμένος χρήστης του συστήματος και ότι δεν αποτελεί κάποιον παρείσακτο που υπέκλεψε τα στοιχεία εισόδου στην εφαρμογή του κανονικού χρήστη.
4. Εάν έχει ολοκληρωθεί και αυτός ο έλεγχος τότε ο χρήστης μπορεί να επιλέξει να εισάγει ένα αίτημα για την άρση του απορρήτου ενός προσώπου επιλέγοντας το αντίστοιχο link που βρίσκεται στα αριστερά της σελίδας.
5. Δίνεται η δυνατότητα στον χρήστη να εισάγει στοιχεία για το φυσικό πρόσωπο που επιθυμεί να διατυπώσει αίτημα για να αρθεί το απόρρητο των επικοινωνιών. Μεταξύ των στοιχείων αυτών είναι το ονοματεπώνυμο του φυσικού προσώπου, το πατρώνυμο, ο αριθμός κινητού τηλεφώνου (υποχρεωτικά πρέπει να ανήκει στον συγκεκριμένο πάροχο που αφορά η εφαρμογή), διεύθυνση και συγκεκριμένο διάστημα για το οποίο θέλει να δει τηλεφωνικές συνδιαλέξεις. Το διάστημα αυτό μπορεί να είναι είτε κενό (αφορά δηλαδή όλο το εύρος των στοιχείων των συνδιαλέξεων που υπάρχουν στη βάση δεδομένων), είτε ανοιχτό ως προς το άνω ή κάτω άκρο. Ο χρήστης μπορεί να συμπληρώσει επίσης και το λόγο για τον οποίο εισάγει το εν λόγω αίτημα σε ειδική περιοχή αιτιολόγησης όπου μπορεί να γράψει ελεύθερα τις προϋποθέσεις που οδήγησαν στην εισαγωγή του αιτήματος.
6. Μετά την επιτυχή εισαγωγή του αιτήματος δίνεται η επιλογή στον χρήστη να εισάγει είτε ένα άλλο αίτημα είτε να επιλέξει την έξοδό του από την εφαρμογή.

Εναλλακτικές Περιπτώσεις:

1. Ο server της εφαρμογής αντιμετωπίζει κάποιο τεχνικό πρόβλημα και δεν είναι δυνατή η πρόσβαση στην εφαρμογή από τον χρήστη. Εμφανίζεται στη σελίδα του browser μήνυμα: The page cannot be displayed.
2. Ο χρήστης είτε δίνει λανθασμένο κάποιο στοιχείο για την είσοδο στην εφαρμογή είτε δεν είναι χρήστης του συστήματος και προσπαθεί να εισέρθει στην εφαρμογή με λανθασμένα στοιχεία. Σε αυτή την περίπτωση εμφανίζεται η σελίδα που περιέχει το μήνυμα για λανθασμένα στοιχεία εισόδου και δίνεται το link για να επιχειρηθεί ξανά είσοδος στην εφαρμογή.
3. Ο χρήστης δεν είναι εγκεκριμένος χρήστης του συστήματος και δεν έχει το απαραίτητο ψηφιακό πιστοποιητικό, είτε δεν δίνει ψηφιακό πιστοποιητικό. Απαγορεύεται η συνέχεια στην εφαρμογή με την εμφάνιση σελίδας που

- περιέχει το μήνυμα σφάλματος και το link για να δοθεί το σωστό πιστοποιητικό.
4. Ο χρήστης κάνει κάποιο λάθος στα στοιχεία που εισάγει και τα οποία είναι απαραίτητα για την διατύπωση αιτήματος. Δεν εισάγεται το αίτημα στη βάση δεδομένων και ο χρήστης ειδοποιείται με μήνυμα σφάλματος ότι πρέπει να επιχειρήσει ξανά την εισαγωγή του αιτήματος.
 5. Ο χρήστης επιχειρεί να εισάγει αίτημα με αριθμό τηλεφώνου που δεν αντιστοιχεί σε συνδρομητή της εταιρείας τηλεπικοινωνιών που αφορά το σύστημα. Δίνεται μήνυμα σφάλματος και ο χρήστης προτρέπεται να απευθυνθεί στην αντίστοιχη εταιρεία.
 6. Ο χρήστης παραμένει ανενεργός για μεγάλο χρονικό διάστημα στην εφαρμογή. Το σύστημα αυτόματα τον αποσυνδέει από την εφαρμογή.

Σενάρια Χρήσης που επεκτείνει:

Κανένα

Διαπροσωπείες:

Κλασσική διαπροσωπεία συστήματος.

Περιορισμοί:

- i. Ο χρήστης θα πρέπει να είναι εγκεκριμένος από το σύστημα, δηλαδή να βρίσκεται και στη βάση δεδομένων, να κατέχει το απαραίτητο ψηφιακό πιστοποιητικό και να του έχει ανατεθεί ο συγκεκριμένος ρόλος.
- ii. Θα πρέπει η επικοινωνία του συστήματος με την βάση δεδομένων να γίνεται με τα στοιχεία σε κρυπτογραφημένη μορφή.
- iii. Τα στοιχεία που «διακινούνται» στο σύστημα από κομμάτι σε κομμάτι της εφαρμογής θα πρέπει να είναι κρυπτογραφημένα και σε ασφαλή σύνδεση https.

3.2.2.2 Σενάριο Επιβεβαίωσης Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

Περιγραφή:

Ένας χρήστης – είτε μέλος της Α.Δ.Α.Ε., είτε της Εισαγγελίας, είτε της Αστυνομικής αρχής – επιθυμεί να συνδεθεί στο σύστημα για να επιβεβαιώσει ένα αίτημα σχετικά με την άρση του απορρήτου των συνδιαλέξεων ενός φυσικού προσώπου, συνδρομητή του συγκεκριμένου παρόχου τηλεπικοινωνιών που αφορά το σύστημα.

Δράστες:

- Χρήστης αρχής
- Σύστημα
- Βάση δεδομένων

Προϋποθέσεις:

- i. Ο χρήστης θα πρέπει να είναι εξουσιοδοτημένος να χρησιμοποιεί το σύστημα, δηλαδή θα πρέπει να έχει λάβει το απαραίτητο ψηφιακό πιστοποιητικό που θα πιστοποιεί ότι είναι εγκεκριμένος να επιτελεί μια από τις εργασίες του συστήματος.
- ii. Ο χρήστης θα πρέπει να υπάρχει στη βάση δεδομένων του συστήματος στον αντίστοιχο πίνακα χρηστών με συγκεκριμένο όνομα και κωδικό πρόσβασης που θα επιτρέψει και την ταυτοποίησή του από το σύστημα με την αποκωδικοποίηση του ψηφιακού πιστοποιητικού.
- iii. Στον χρήστη θα πρέπει να έχει ανατεθεί επίσης ο συγκεκριμένος ρόλος της επιβεβαίωσης αιτημάτων για την άρση του απορρήτου των επικοινωνιών για φυσικά πρόσωπα στον αντίστοιχο πίνακα ρόλων της βάσης δεδομένων, ούτως ώστε να είναι δυνατή η πρόσβασή του στο συγκεκριμένο κομμάτι της εφαρμογής.

Κείμενο Σεναρίου:

- 1 Ο χρήστης συνδέεται στην εφαρμογή εισάγοντας στον browser του τερματικού του το url της εφαρμογής το οποίο για λόγους ασφαλείας απαιτεί https σύνδεση.
- 2 Στην αρχική σελίδα απαιτείται να εισάγει το αναγνωριστικό του όνομα για το σύστημα και τον ειδικό αντίστοιχο κωδικό πρόσβασης.
- 3 Μετέπειτα καλείται να εισάγει το ψηφιακό πιστοποιητικό που του έχει δοθεί έτσι ώστε να είναι δυνατή η επιβεβαίωση ότι είναι ο συγκεκριμένος εγκεκριμένος χρήστης του συστήματος και ότι δεν αποτελεί κάποιον παρείσακτο που υπέκλεψε τα στοιχεία εισόδου στην εφαρμογή του κανονικού χρήστη.
- 4 Εάν έχει ολοκληρωθεί και αυτός ο έλεγχος τότε ο χρήστης μπορεί να επιλέξει να επιβεβαιώσει ένα αίτημα για την άρση του απορρήτου ενός προσώπου επιλέγοντας το αντίστοιχο link που βρίσκεται στα αριστερά της σελίδας.
- 5 Δίνεται η δυνατότητα στον χρήστη να επιλέξει το αίτημα για το οποίο επιθυμεί να δώσει έγκριση. Παρέχεται επίσης η δυνατότητα να δει την αιτιολόγηση που έχει δώσει ο χρήστης που εισήγαγε το αίτημα. Στον χρήστη παρέχεται η δυνατότητα να επιβεβαιώσει αιτήματα που δεν είναι ήδη εγκεκριμένα από άλλο μέλος της ίδιας αρχής ή από τον ίδιο τον χρήστη.
- 6 Μετά την επιτυχή εισαγωγή του αιτήματος δίνεται η επιλογή στον χρήστη να εισάγει είτε ένα άλλο αίτημα είτε να επιλέξει την έξοδό του από την εφαρμογή.

Εναλλακτικές Περιπτώσεις:

1. Ο server της εφαρμογής αντιμετωπίζει κάποιο τεχνικό πρόβλημα και δεν είναι δυνατή η πρόσβαση στην εφαρμογή από τον χρήστη. Εμφανίζεται στη σελίδα του browser μήνυμα: The page cannot be displayed.
2. Ο χρήστης είτε δίνει λανθασμένο κάποιο στοιχείο για την είσοδο στην εφαρμογή είτε δεν είναι χρήστης του συστήματος και προσπαθεί να εισέρθει στην εφαρμογή με λανθασμένα στοιχεία. Σε αυτή την περίπτωση εμφανίζεται η σελίδα που περιέχει το μήνυμα για λανθασμένα στοιχεία εισόδου και δίνεται το link για να επιχειρηθεί ξανά είσοδος στην εφαρμογή.
3. Ο χρήστης δεν είναι εγκεκριμένος χρήστης του συστήματος και δεν έχει το απαραίτητο ψηφιακό πιστοποιητικό, είτε δεν δίνει ψηφιακό πιστοποιητικό. Απαγορεύεται η συνέχεια στην εφαρμογή με την εμφάνιση σελίδας που

- περιέχει το μήνυμα σφάλματος και το link για να δοθεί το σωστό πιστοποιητικό.
4. Ο χρήστης δεν επιλέγει κάποιο αίτημα για να επιβεβαιώσει, αλλά παρ' όλα αυτά πατά το κουμπί της επιβεβαίωσης. Το σύστημα προτρέπει τον χρήστη να επιλέξει συγκεκριμένο αίτημα.
 5. Ο χρήστης παραμένει ανενεργός για μεγάλο χρονικό διάστημα στην εφαρμογή. Το σύστημα αυτόματα τον αποσυνδέει από την εφαρμογή.

Σενάρια Χρήσης που επεκτείνει:

Κανένα

Διαπροσωπείες:

Κλασσική διαπροσωπεία συστήματος.

Περιορισμοί:

- i. Ο χρήστης θα πρέπει να είναι εγκεκριμένος από το σύστημα, δηλαδή να βρίσκεται και στη βάση δεδομένων, να κατέχει το απαραίτητο ψηφιακό πιστοποιητικό και να του έχει ανατεθεί ο συγκεκριμένος ρόλος.
- ii. Θα πρέπει η επικοινωνία του συστήματος με την βάση δεδομένων να γίνεται με τα στοιχεία σε κρυπτογραφημένη μορφή.
- iii. Τα στοιχεία που «διακινούνται» στο σύστημα από κομμάτι σε κομμάτι της εφαρμογής θα πρέπει να είναι κρυπτογραφημένα και σε ασφαλή σύνδεση https.
- iv. Είναι απαραίτητο ο χρήστης να μπορεί να επιβεβαιώσει τα αιτήματα εκείνα που δεν έχουν ακόμη έγκριση από τον ίδιο ή κάποιο άλλο εξουσιοδοτημένο μέλος της ίδιας αρχής στην οποία ανήκει ο χρήστης.

3.2.2.3 Σενάριο Επισκόπησης Στοιχείων Συνδιαλέξεων έπειτα από Εισαγωγή Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

Περιγραφή:

Ένας χρήστης – είτε μέλος της Α.Δ.Α.Ε., είτε της Εισαγγελίας, είτε της Αστυνομικής αρχής – επιθυμεί να συνδεθεί στο σύστημα για να δει τα στοιχεία που προκύπτουν από την εκτέλεση ενός ολοκληρωμένου αιτήματος σχετικά με την άρση του απορρήτου των συνδιαλέξεων ενός φυσικού προσώπου, συνδρομητή του συγκεκριμένου παρόχου τηλεπικοινωνιών που αφορά το σύστημα.

Δράστες:

- Χρήστης αρχής
- Σύστημα
- Βάση δεδομένων

Προϋποθέσεις:

- i. Ο χρήστης θα πρέπει να είναι εξουσιοδοτημένος να χρησιμοποιεί το σύστημα, δηλαδή θα πρέπει να έχει λάβει το απαραίτητο ψηφιακό πιστοποιητικό που θα πιστοποιεί ότι είναι εγκεκριμένος να επιτελεί μια από τις εργασίες του συστήματος.
- ii. Ο χρήστης θα πρέπει να υπάρχει στη βάση δεδομένων του συστήματος στον αντίστοιχο πίνακα χρηστών με συγκεκριμένο όνομα και κωδικό πρόσβασης που θα επιτρέψει και την ταυτοποίησή του από το σύστημα με την αποκωδικοποίηση του ψηφιακού πιστοποιητικού.
- iii. Στον χρήστη θα πρέπει να έχει ανατεθεί επίσης ο συγκεκριμένος ρόλος της επισκόπησης στοιχείων συνδιαλέξεων ολοκληρωμένων αιτημάτων για φυσικά πρόσωπα στον αντίστοιχο πίνακα ρόλων της βάσης δεδομένων, ούτως ώστε να είναι δυνατή η πρόσβασή του στο συγκεκριμένο κομμάτι της εφαρμογής.
- iv. Θα πρέπει να υπάρχουν τουλάχιστον ένα ή περισσότερα αιτήματα που έχουν λάβει έγκριση από όλες τις αρχές και είναι έτοιμα προς εκτέλεση.

Κείμενο Σεναρίου:

1. Ο χρήστης συνδέεται στην εφαρμογή εισάγοντας στον browser του τερματικού του το url της εφαρμογής το οποίο για λόγους ασφαλείας απαιτεί https σύνδεση.
2. Στην αρχική σελίδα απαιτείται να εισάγει το αναγνωριστικό του όνομα για το σύστημα και τον ειδικό αντίστοιχο κωδικό πρόσβασης.
3. Μετέπειτα καλείται να εισάγει το ψηφιακό πιστοποιητικό που του έχει δοθεί έτσι ώστε να είναι δυνατή η επιβεβαίωση ότι είναι ο συγκεκριμένος εγκεκριμένος χρήστης του συστήματος και ότι δεν αποτελεί κάποιον παρείσακτο που υπέκλεψε τα στοιχεία εισόδου στην εφαρμογή του κανονικού χρήστη.
4. Εάν έχει ολοκληρωθεί και αυτός ο έλεγχος τότε ο χρήστης μπορεί να επιλέξει να δει τα στοιχεία των συνδιαλέξεων ενός προσώπου, όπως αυτά διατυπώνονται σε ένα αίτημα για την άρση του απορρήτου του προσώπου, επιλέγοντας το αντίστοιχο link που βρίσκεται στα αριστερά της σελίδας.
5. Ο χρήστης μπορεί να επιλέξει ένα από τα αιτήματα που του παρουσιάζονται και να επιλέξει να δει τα στοιχεία των συνδιαλέξεων του προσώπου στο οποίο αφορά το αίτημα. Έχει επίσης τη δυνατότητα να δει την αιτιολόγηση που δόθηκε κατά την εισαγωγή του δεδομένου αιτήματος.
6. Μετά την επιλογή του αιτήματος ο χρήστης προωθείται σε άλλη σελίδα του συστήματος που παρουσιάζει αναλυτικά τα στοιχεία τόσο του προσώπου στο οποίο αφορά το αίτημα όσο και των τηλεφωνικών του συνδιαλέξεων. Συγκεκριμένα για τις τηλεφωνικές συνδιαλέξεις περιλαμβάνονται στοιχεία όπως ο αριθμός καλών και καλούμενου, οι τοποθεσίες τους, οι σταθμοί βάσης που εξυπηρέτησαν τον καθένα από αυτούς, η ημερομηνία και ακριβής ώρα της κλήσης (τόσο ώρα έναρξης όσο και λήξης) καθώς και στοιχεία προσδιοριστικά των φυσικών προσώπων που μετείχαν στην επικοινωνία (IMSI) και των συσκευών που χρησιμοποίησαν (IMEI). Φυσικά αν ένας από τους συμμετέχοντες στην επικοινωνία δεν είναι συνδρομητής του παρόχου τότε κάποια στοιχεία είναι άγνωστα.
7. Έπειτα ο χρήστης έχει τη δυνατότητα να δει ένα άλλο ολοκληρωμένο αίτημα είτε να επιλέξει την έξοδό του από την εφαρμογή.

Εναλλακτικές Περιπτώσεις:

1. Ο server της εφαρμογής αντιμετωπίζει κάποιο τεχνικό πρόβλημα και δεν είναι δυνατή η πρόσβαση στην εφαρμογή από τον χρήστη. Εμφανίζεται στη σελίδα του browser μήνυμα: The page cannot be displayed.
2. Ο χρήστης είτε δίνει λανθασμένο κάποιο στοιχείο για την είσοδο στην εφαρμογή είτε δεν είναι χρήστης του συστήματος και προσπαθεί να εισέρθει στην εφαρμογή με λανθασμένα στοιχεία. Σε αυτή την περίπτωση εμφανίζεται η σελίδα που περιέχει το μήνυμα για λανθασμένα στοιχεία εισόδου και δίνεται το link για να επιχειρηθεί ξανά είσοδος στην εφαρμογή.
3. Ο χρήστης δεν είναι εγκεκριμένος χρήστης του συστήματος και δεν έχει το απαραίτητο ψηφιακό πιστοποιητικό, είτε δεν δίνει ψηφιακό πιστοποιητικό. Απαγορεύεται η συνέχεια στην εφαρμογή με την εμφάνιση σελίδας που περιέχει το μήνυμα σφάλματος και το link για να δοθεί το σωστό πιστοποιητικό.
4. Ο χρήστης δεν επιλέγει κάποιο αίτημα για να εκτελεστεί, αλλά παρ' όλα αυτά πατά το κουμπί της εκτέλεσης. Το σύστημα προτρέπει τον χρήστη να επιλέξει συγκεκριμένο αίτημα.
5. Ο χρήστης παραμένει ανενεργός για μεγάλο χρονικό διάστημα στην εφαρμογή. Το σύστημα αυτόματα τον αποσυνδέει από την εφαρμογή.

Σενάρια Χρήσης που επεκτείνει:

Κανένα

Διαπροσωπείες:

Κλασσική διαπροσωπεία συστήματος.

Περιορισμοί:

- i. Ο χρήστης θα πρέπει να είναι εγκεκριμένος από το σύστημα, δηλαδή να βρίσκεται και στη βάση δεδομένων, να κατέχει το απαραίτητο ψηφιακό πιστοποιητικό και να του έχει ανατεθεί ο συγκεκριμένος ρόλος.
- ii. Θα πρέπει η επικοινωνία του συστήματος με την βάση δεδομένων να γίνεται με τα στοιχεία σε κρυπτογραφημένη μορφή.
- iii. Τα στοιχεία που «διακινούνται» στο σύστημα από κομμάτι σε κομμάτι της εφαρμογής θα πρέπει να είναι κρυπτογραφημένα και σε ασφαλή σύνδεση https.
- iv. Ο χρήστης θα πρέπει να μπορεί να δει μόνο τα αιτήματα εκείνα που είναι ολοκληρωμένα και που έχουν έγκριση από όλες τις αρμόδιες αρχές.
- v. Στον χρήστη θα παρουσιάζονται μόνο τα αιτήματα εκείνα που έχει εισάγει ο ίδιος, εκτός κι αν του έχει ανατεθεί ο υπερ – ρόλος που επιτρέπει την πρόσβαση σε όλα τα ολοκληρωμένα αιτήματα (καθολικός ρόλος), ανεξαρτήτως του προσώπου που τα εισήγαγε.

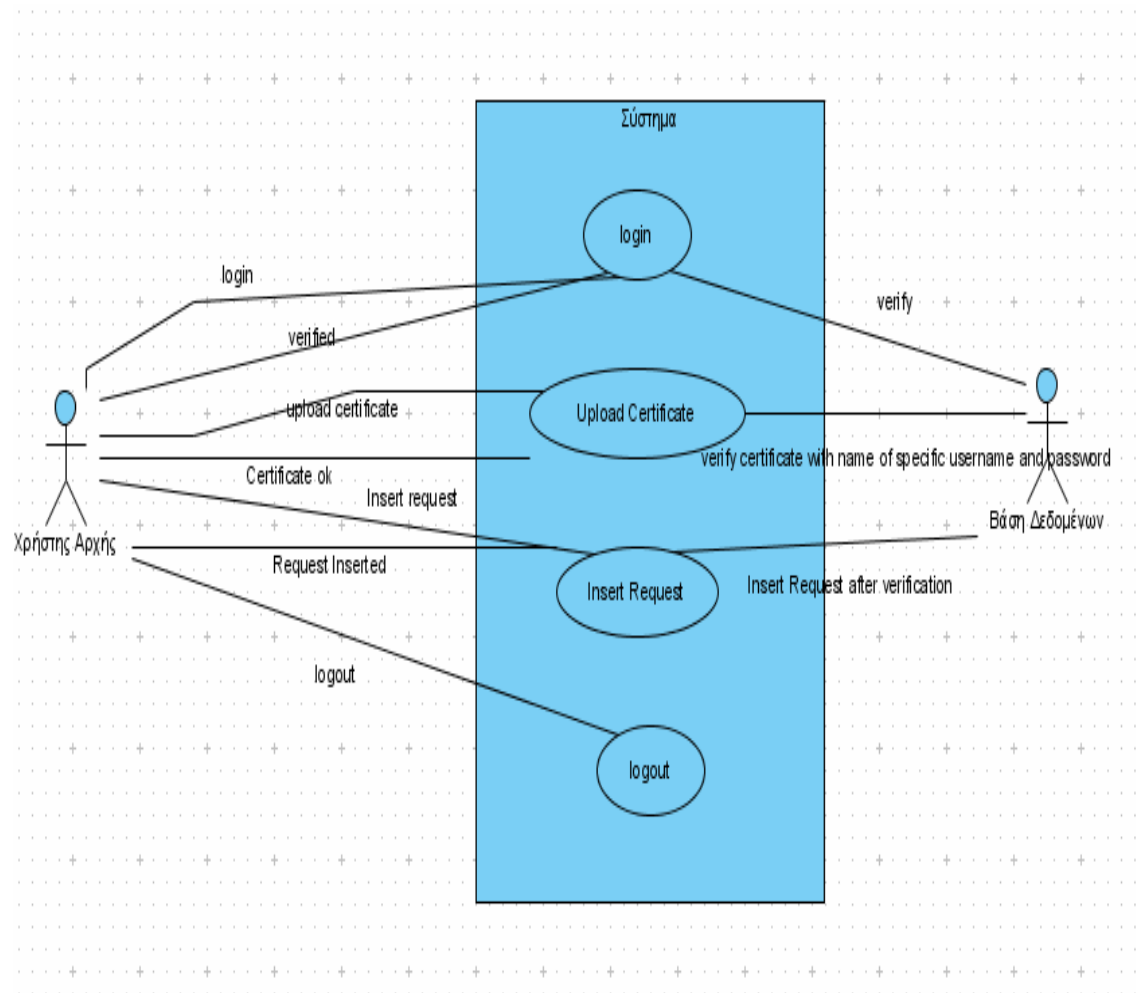
3.2.2.4 Σενάριο χρήσης του συστήματος από χρήστη που έχει καθολικό ρόλο.

Περιγραφή:

Ένας χρήστης – είτε μέλος της Α.Δ.Α.Ε., είτε της Εισαγγελίας, είτε της Αστυνομικής αρχής – έχει καθολικό ρόλο όταν είναι εγκεκριμένος να επιτελέσει όλες τις λειτουργίες που παρέχει το σύστημα. Ουσιαστικά το δεδομένο σενάριο χρήσης μπορεί να είναι οποιοδήποτε από αυτά που προηγήθηκαν, είτε και όλα μαζί, εκτελεσμένα σε οποιαδήποτε σειρά. Για τους περιορισμούς/προϋποθέσεις που επιβάλλει το δεδομένο σενάριο ισχύει η ένωση των περιορισμών/προϋποθέσεων των προηγούμενων σεναρίων.

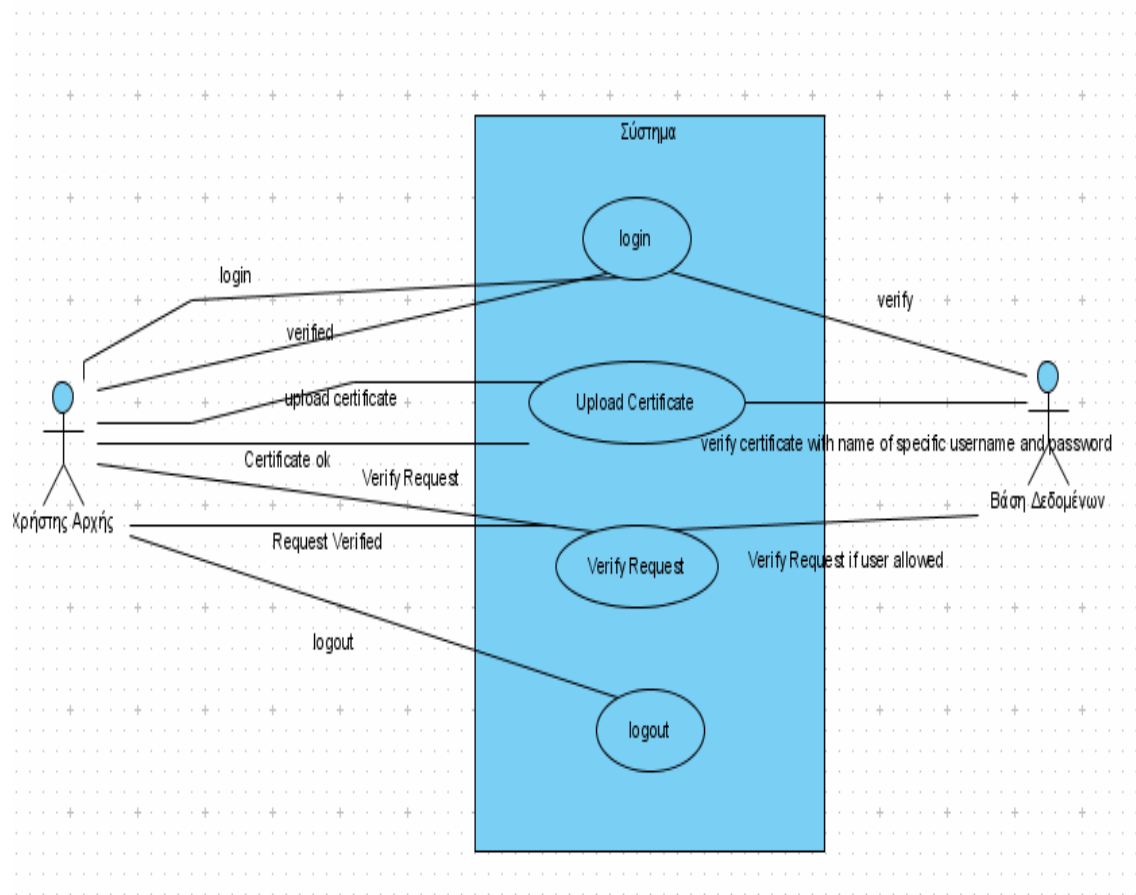
3.2.3 Διαγράμματα Περιπτώσεων Χρήσης.

3.2.3.1 Σενάριο Εισαγωγής Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.



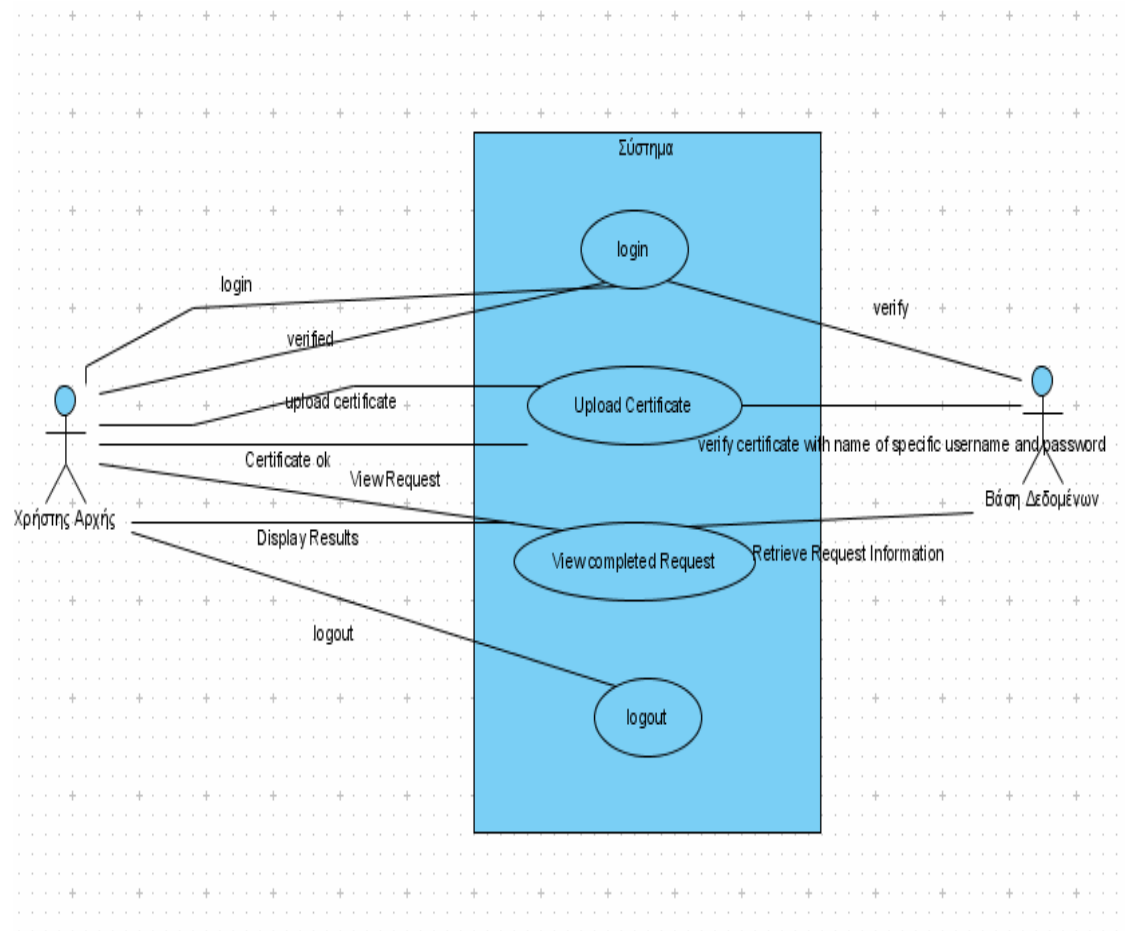
Σχήμα 3.2 Διάγραμμα Ενεργειών για το σενάριο εισαγωγής αιτήματος 3.2.2.1

3.2.3.2 Σενάριο Επιβεβαίωσης Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.



Σχήμα 3.3 Διάγραμμα Ενεργειών για το σενάριο επιβεβαίωσης αιτήματος 3.2.2.2

3.2.3.3 Σενάριο Επισκόπησης Στοιχείων Συνδιαλέξεων έπειτα από Εισαγωγή Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

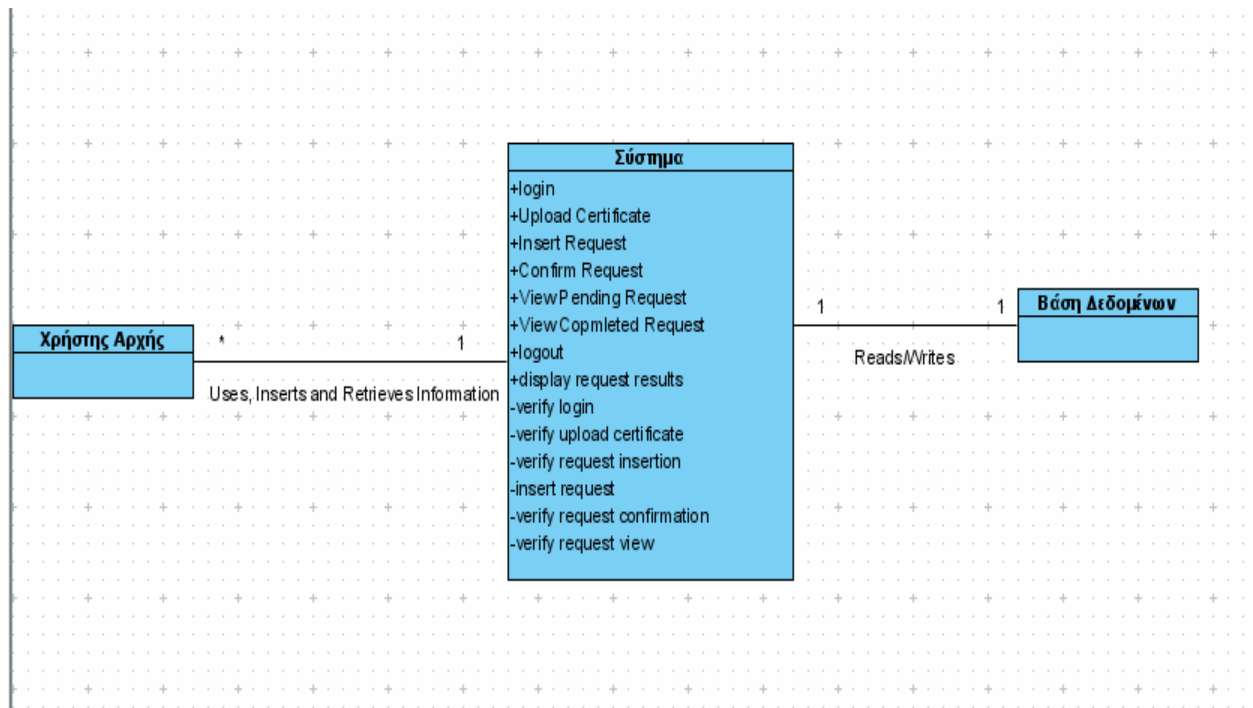


Σχήμα 3.3 Διάγραμμα Ενεργειών για το σενάριο επισκόπησης στοιχείων

3.3 Μοντέλο Πεδίου.

3.3.1 Διάγραμμα κλάσεων Πεδίου.

Στο κάτωθι διάγραμμα κλάσεων πεδίου παρουσιάζονται οι βασικές λειτουργίες των κύριων κλάσεων του πεδίου σε αφαιρετική μορφή για την ομαδοποιημένη παρουσίαση και καλύτερη κατανόηση του συστήματος:



Σχήμα 3.4 Διάγραμμα Κύριων Κλάσεων Πεδίου

3.3.2 Ορισμός κύριας κλάσης πεδίου (Σύστημα).

<Σύστημα>

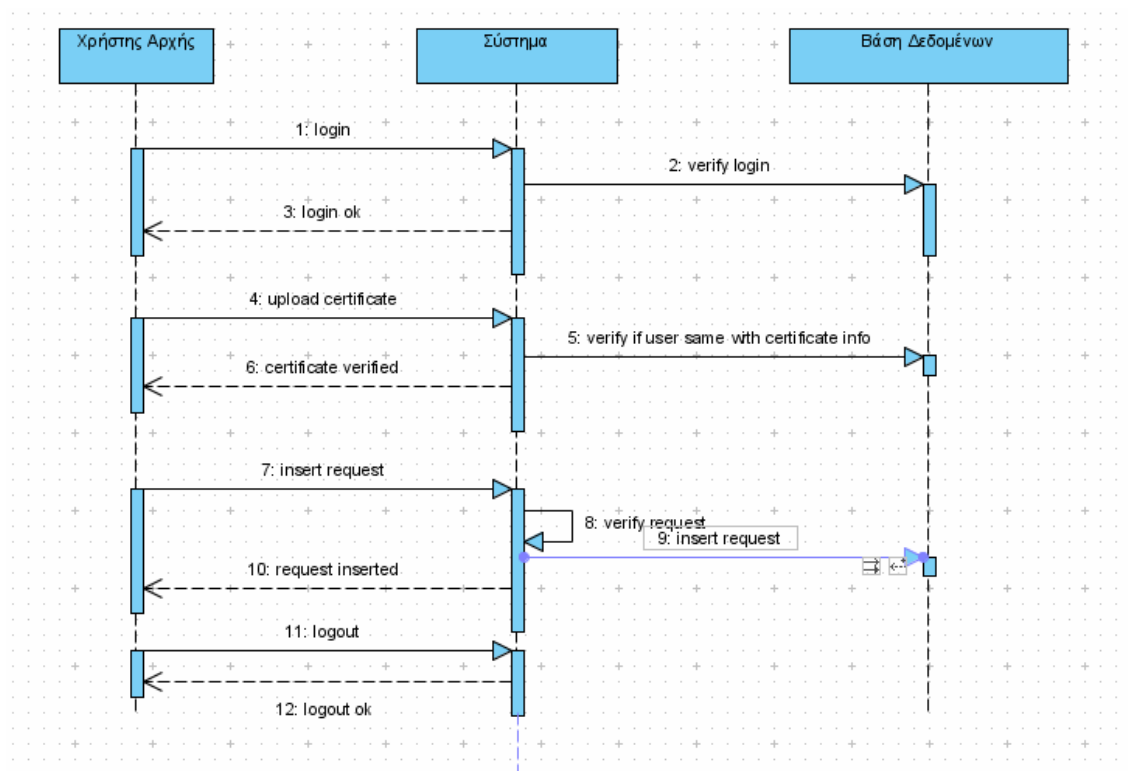
Description	Σύστημα, εφαρμογή
Attributes	+login +upload certificate -verify login -verify upload certificate +insert request +confirm request +view pending request +view completed request +logout -verify request insertion -verify request confirmation -verify request view -insert request +display request results
Responsibilities	<i>Επιβεβαίωση γνησιότητας πιστοποιητικού, επιβεβαίωση αυθεντικότητας χρήστη. Επιβεβαίωση ορθότητας για εισαγωγή αιτήματος, επιβεβαίωση αιτήματος, επισκόπηση αποτελεσμάτων εκτέλεσης αιτήματος. Εισαγωγή αιτήματος και επιβεβαίωσης αιτήματος στη βάση δεδομένων. Επικοινωνία με βάση δεδομένων. Χειρισμός επιχειρηματικής λογικής κ.τ.λ.</i>

3.4 Διαγράμματα Αλληλεπίδρασης.

Για κάθε περίπτωση σεναρίου χρήσης (εκτός από την τελευταία – περίπτωση 2.2.4) παρατίθενται διαγράμματα αλληλεπίδρασης, δηλαδή ακολουθιακά διαγράμματα και συνεργατικά διαγράμματα για να τονίσουν τα μεν ακολουθιακά τη χρονική αλληλουχία των ενεργειών, τα δε συνεργατικά την γενικότερη θέση των δρώντων μονάδων στο σύστημα και τον ρόλο που επιτελούν.

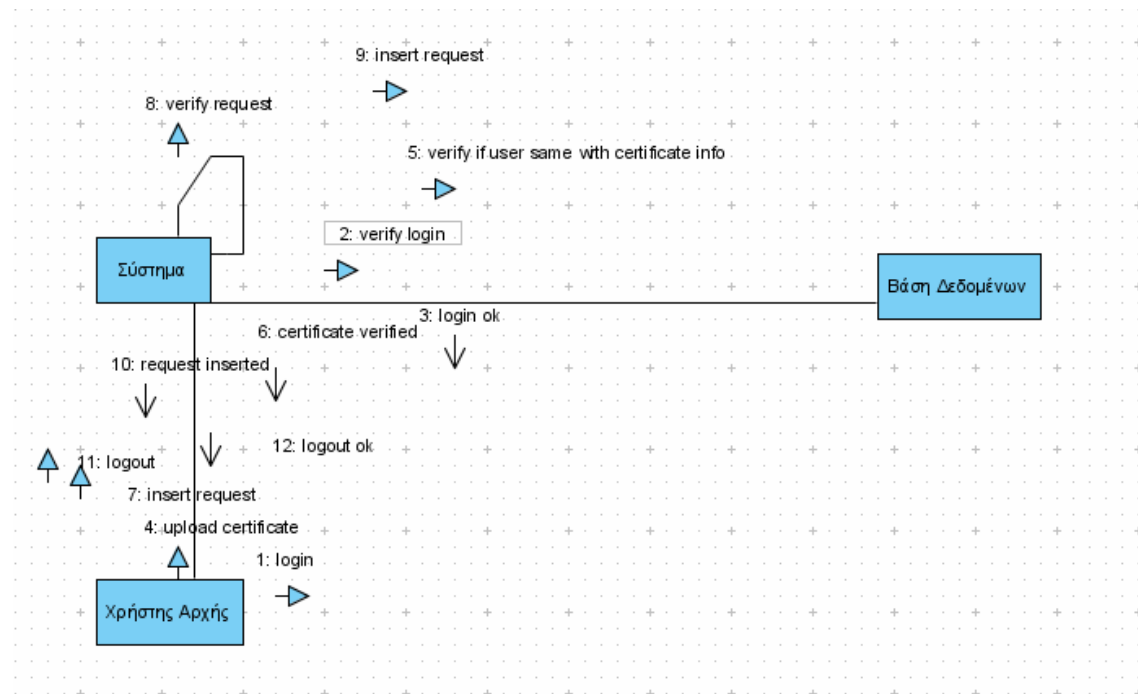
3.4.1 Σενάριο Εισαγωγής Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

Το ακολουθιακό διάγραμμα που παρατίθεται παρακάτω, καλύπτει το σενάριο εισαγωγής αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου, με ανταλλαγή μηνυμάτων αριθμημένων κατά χρονολογική σειρά, για την εμπέδωση της χρονικής αλληλουχίας που λαμβάνει χώρα κατά την εκτέλεση του αντίστοιχου σεναρίου.



Σχήμα 3.5 Ακολουθιακό Διάγραμμα για το σενάριο εισαγωγής αιτήματος 3.2.2.1

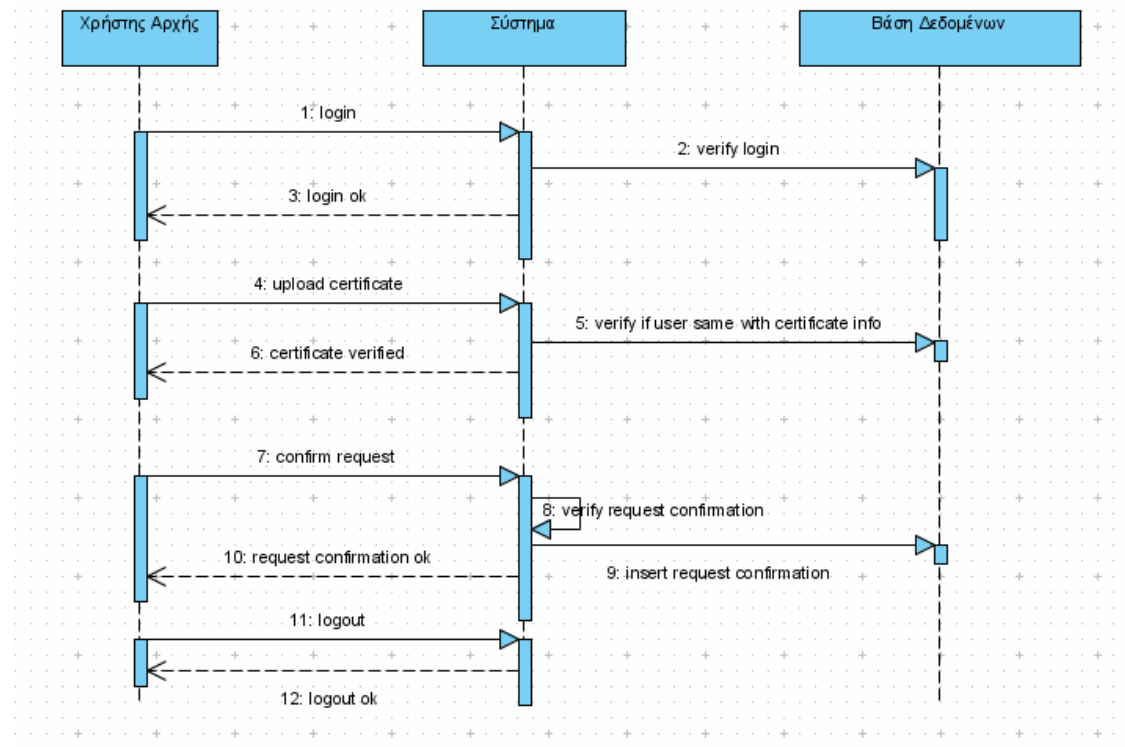
Το κάτωθι συνεργατικό διάγραμμα καλύπτει το σενάριο εισαγωγής αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου, και εξυπηρετεί την ανάγκη για την κατάλληλη αναπαράσταση των μηνυμάτων – αλληλεπιδράσεων μεταξύ των οντοτήτων του σεναρίου, έτσι ώστε να δίνεται βαρύτητα στον τρόπο επικοινωνίας – αλληλεπίδρασης.



Σχήμα 3.6 Συνεργατικό Διάγραμμα για το σενάριο εισαγωγής αιτήματος 3.2.2.1

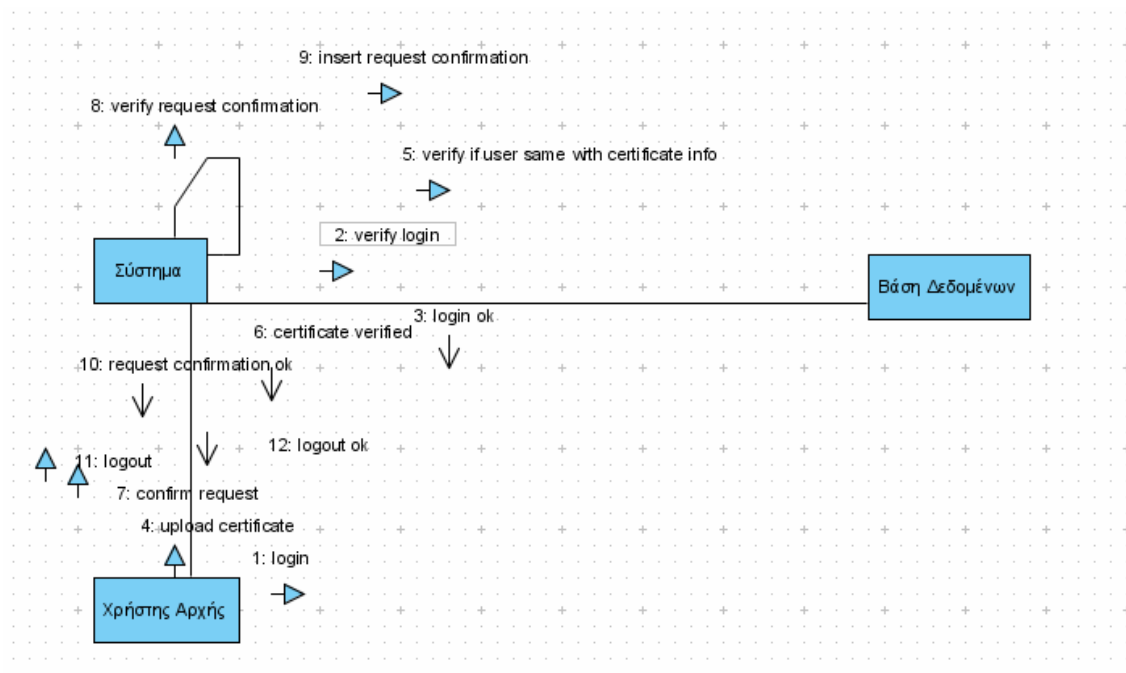
3.4.2 Σενάριο Επιβεβαίωσης Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

Ακολουθεί το ακολουθιακό διάγραμμα της περίπτωσης χρήσης που αφορά στην επιβεβαίωση αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου, με τέτοια αναπαράσταση που να καταδεικνύει την χρονική και λογική αλληλουχία των ενεργειών και διεργασιών που εκτελούνται για το δεδομένο σενάριο.



Σχήμα 3.7 Ακολουθιακό Διάγραμμα για το σενάριο επιβεβαίωσης αιτήματος 3.2.2.2

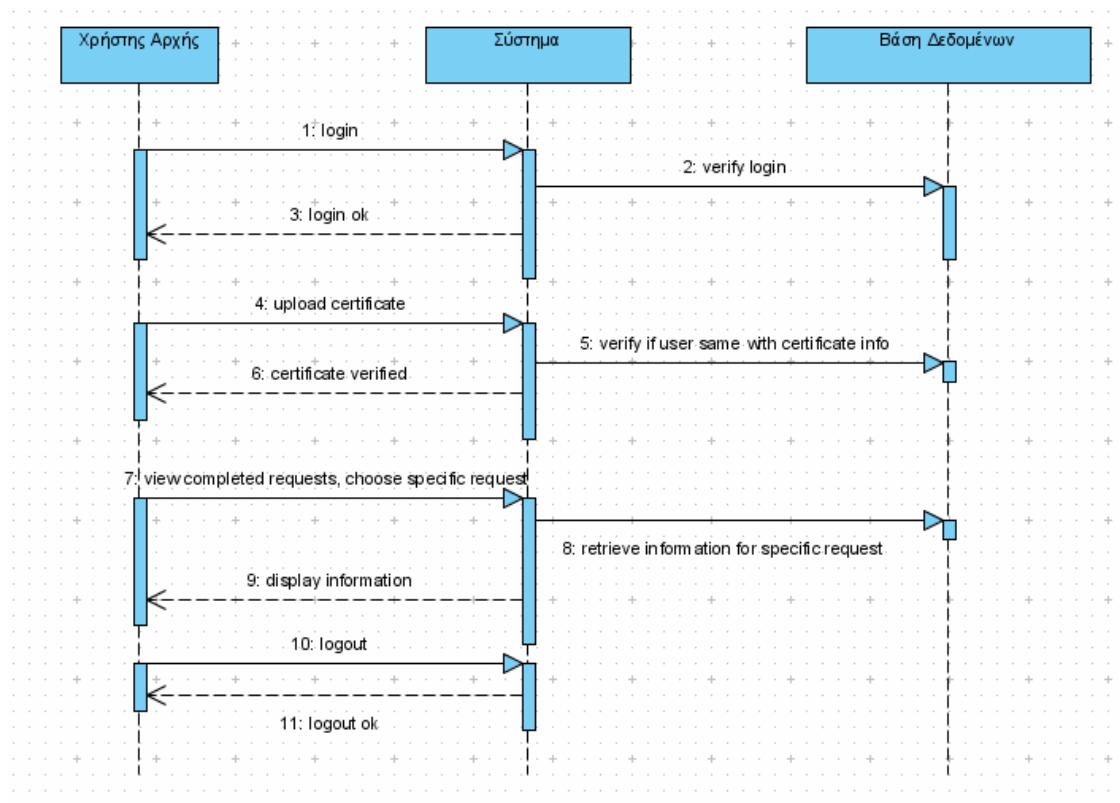
Το αντίστοιχο συνεργατικό διάγραμμα για το σενάριο επιβεβαίωσης αιτήματος δίνει βαρύνουσα σημασία στην αναπαράσταση των οντοτήτων και στο πως επηρεάζουν η μια την άλλη με τα διάφορα μηνύματα που ανταλλάσσουν και τις διαδικασίες που εκκινούνται από αυτά.



**Σχήμα 3.8 Συνεργατικό Διάγραμμα για το σενάριο επιβεβαίωσης αιτήματος
3.2.2.2**

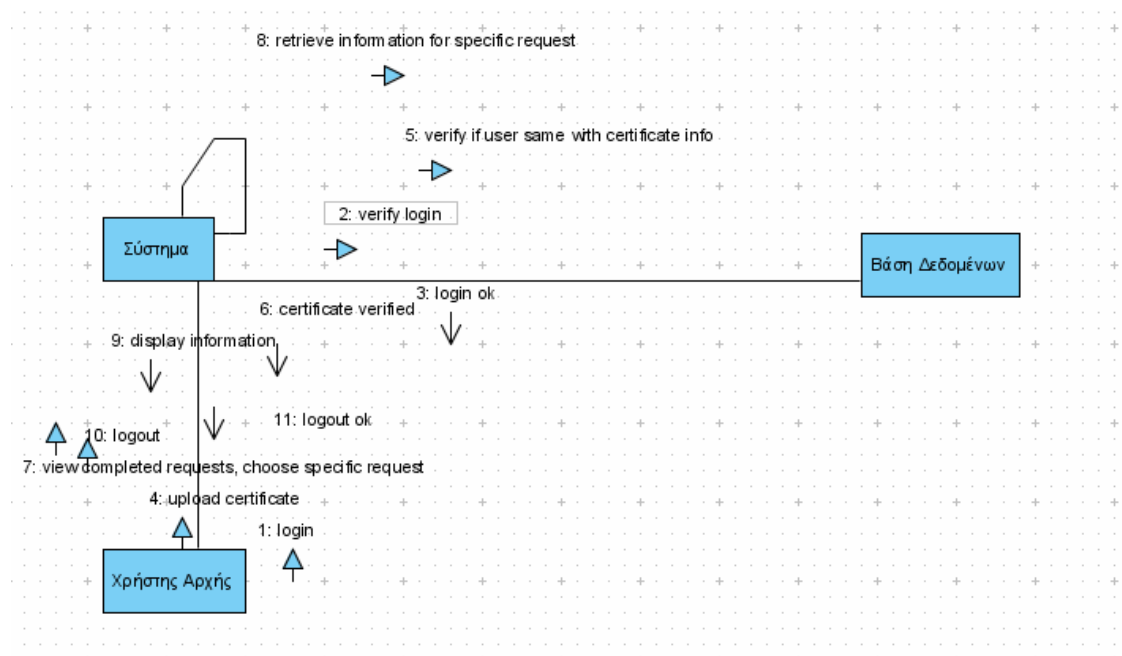
3.4.3 Σενάριο Επισκόπησης Στοιχείων Συνδιαλέξεων έπειτα από Εισαγωγή Αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου.

Για το σενάριο επισκόπησης στοιχείων συνδιαλέξεων έπειτα από εισαγωγή και φυσικά επιβεβαίωση αιτήματος για την άρση του απορρήτου συνδρομητή τηλεπικοινωνιακού παρόχου δίνεται το παρακάτω ακολουθιακό διάγραμμα που διευκρινίζει τις απαραίτητες ενέργειες ώστε ένας χρήστης να κάνει χρήση των δεδομένων δυνατοτήτων της εφαρμογής που αφορούν στο συγκεκριμένο σενάριο.



Σχήμα 3.9 Ακολουθιακό Διάγραμμα για το σενάριο επισκόπησης αιτήματος
3.2.2.3

Αντίστοιχα με το ακολουθιακό διάγραμμα, το συνεργατικό διάγραμμα παρουσιάζει πως ο χρήστης αλληλεπιδρά με τις λοιπές οντότητες του σεναρίου ώστε να επιτευχθεί η υλοποίηση της λειτουργικότητας που περιγράφεται από αυτή την περίπτωση χρήσης.



Σχήμα 3.10 Συνεργατικό Διάγραμμα για το σενάριο επισκόπησης αιτήματος
3.2.2.3

3.5 Σημαντικές Σχεδιαστικές Αποφάσεις.

Στο τμήμα αυτό είναι σκόπιμο να αναφερθούμε στις πιο σημαντικές σχεδιαστικές αποφάσεις που πάρθηκαν κατά την ανάπτυξη της εφαρμογής και οδήγησαν στη δεδομένη υλοποίηση του έργου. Στις αποφάσεις αυτές κυρίαρχο ρόλο έπαιξαν τόσο οι απαιτήσεις – προδιαγραφές του συστήματος όσο και η διαθέσιμη τεχνολογία με την οποία υλοποιήθηκε το σύστημα, καθώς και οι κύριες κατευθύνσεις που επέβαλαν οι Ευρωπαϊκοί Κανονισμοί και το ελληνικό Προεδρικό Διάταγμα που καθορίζουν την φύση και τις προϋποθέσεις της άρσης του τηλεπικοινωνιακού απορρήτου.

Κρίσιμο ζήτημα στην περίπτωση της άρσης των τηλεπικοινωνιακών δεδομένων φυσικού προσώπου και συνακόλουθα και ευαίσθητων προσωπικών δεδομένων αποτελεί η ασφάλεια με την οποία θα υλοποιηθεί το όλο σύστημα. Είναι προφανές ότι δεν επιτρέπεται σε καμία περίπτωση η άρση των προσωπικών δεδομένων να γίνεται από μη εξουσιοδοτημένους εκπροσώπους φορέων είτε να υπάρχει δυνατότητα υποκλοπής των στοιχείων που παρουσιάζονται από το σύστημα. Επομένως, η εφαρμογή θα έπρεπε να υλοποιηθεί με την μεγαλύτερη δυνατή προστασία.

Δεδομένου ότι η αποθήκευση όλων των δεδομένων γίνεται στη βάση δεδομένων του συστήματος ο πρωταρχικός τομέας όπου θα έπρεπε να εφαρμοστεί η ασφάλεια ήταν αυτός. Επιλέχθηκε τα δεδομένα να διατηρούνται σε κωδικοποιημένη μορφή στη βάση με βάση το πρότυπο AES (Advanced Encryption Standard). Αυτό το πρότυπο διασφαλίζει ότι τα κρυπτογραφημένα δεδομένα δεν θα μπορούν να αναγνωστούν από κάποιο παρείσακτο ή κάποιο πρόσωπο που παρείσφρησε σε κάποια θέση στην ομάδα διαχείρισης της βάσης δεδομένων και έχει κακόβουλο σκοπό. Για τους ίδιους λόγους επιλέχθηκε τα δεδομένα που ανταλλάσσονται μεταξύ της βάσης και της εφαρμογής να είναι σε κρυπτογραφημένη μορφή.

Ως δηλωτικό της ασφάλειας που απαιτεί η δεδομένη εφαρμογή αποτελεί ο τρόπος με τον οποίο γίνεται πιστοποίηση των προσώπων που δικαιούνται να έχουν πρόσβαση σε αυτή. Συγκεκριμένα για να έχει ένα πρόσωπο πρόσβαση στην εφαρμογή θα πρέπει όχι μόνο να είναι στα πρόσωπα που δικαιούνται πρόσβαση στον αντίστοιχο πίνακα της βάσης δεδομένων αλλά και να διαθέτει ένα μοναδικό ψηφιακό πιστοποιητικό που του παρέχει το σύστημα και μέσω του οποίου γίνεται ταυτοποίηση του χρήστη. Συγκεκριμένα ελέγχεται όχι μόνο η εγκυρότητα του πιστοποιητικού αλλά και η αντιστοιχία με το πρόσωπο που εισήλθε στην εφαρμογή βάσει των στοιχείων της βάσης δεδομένων.

Τέλος για να καλυφθεί το θέμα της ασφάλειας από όλες τις απόψεις επιλέχθηκε η χρήση της εφαρμογής να είναι δυνατή μόνο μέσω του πρωτοκόλλου https που χρησιμοποιεί το SSL (Secure Sockets Layer) ώστε τα δεδομένα να είναι πάντοτε κρυπτογραφημένα κατά την χρήση τους στην εφαρμογή.

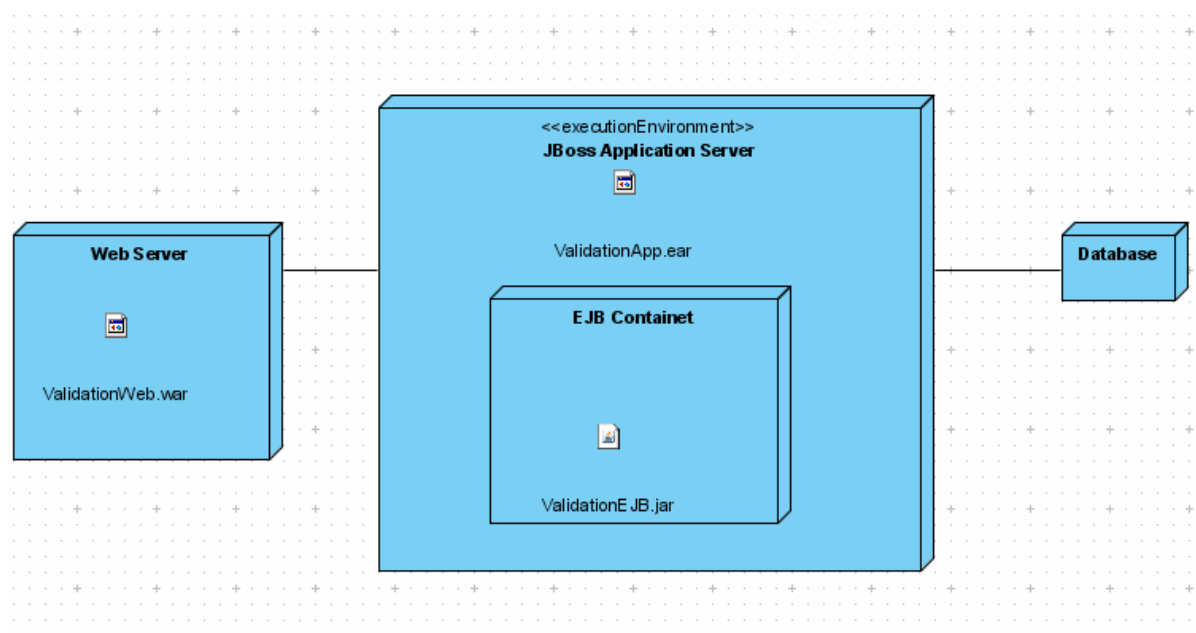
Ωστόσο σε ένα τόσο ευαίσθητο θέμα όπως η άρση του απορρήτου προσωπικών δεδομένων δεν αρκεί μόνο η σκοπιά της ασφάλειας του συστήματος από κακόβουλο λογισμικό ή κακόβουλα πρόσωπα με στόχο να υποκλέψουν σημαντικές πληροφορίες. Είναι απαραίτητο να ελέγχεται με τον πιο αυστηρό τρόπο η εξουσιοδότηση πρόσβασης στα δεδομένα. Για τον λόγο αυτό δημιουργήθηκαν κατηγορίες χρηστών στην εφαρμογή με διαφορετικά δικαιώματα στην κάθε κατηγορία ώστε να διασφαλιστεί ότι δεν είναι δυνατόν για ένα μέλος κάποιας αρχής να αποκτήσει πρόσβαση σε όλες τις λειτουργικότητες της εφαρμογής και άρα να μπορεί να έχει πρόσβαση σε ευαίσθητα δεδομένα παρακάμπτοντας κάθε άλλη αρχή. Αυτό άλλωστε διασφαλίζεται και από την απαίτηση για την εφαρμογή να είναι απαραίτητη η επιβεβαίωση ενός αιτήματος άρσης απορρήτου από όλες τις συμμετέχουσες αρχές.

Λόγω λοιπόν των ανωτέρω περιορισμών επιλέχθηκε να είναι δυνατή για ένα χρήστη μόνο η εισαγωγή αιτιολογημένων αιτημάτων είτε η επιβεβαίωση αιτημάτων (που φυσικά δεν έχουν ήδη δεχθεί επιβεβαίωση από κάποιο άλλο μέλος της ίδιας αρχής) είτε η επισκόπηση αιτημάτων που έχει υποβάλλει μέλος της ίδιας αρχής ή ο ίδιος ο χρήστης, εάν έχει το ρόλο της επισκόπησης των προσωπικών στοιχείων και των στοιχείων των κλήσεων που προκύπτουν από την εκτέλεση αιτήματος. Έτσι δημιουργήθηκαν στεγανά ανάμεσα στις λειτουργίες της εφαρμογής και δεν είναι δυνατόν να γίνει κάποια προσπάθεια που θα έθετε σε κίνδυνο την εφαρμογή είτε τα ευαίσθητα δεδομένα. Ωστόσο, δημιουργήθηκε και ένας υπέρ-ρόλος στο σύστημα που επιτρέπει την χρήση όλων των λειτουργιών, ωστόσο και αυτός περιορίζεται από την ανάγκη επιβεβαίωσης από όλες τις διαφορετικές αρχές και έτσι δεν τίθεται σε κίνδυνο η ασφάλεια των δεδομένων.

3.6 Αρχιτεκτονική Συστήματος – Λεπτομερή διαγράμματα κλάσεων.

3.6.1 Παραταξιακό διάγραμμα συστήματος.

Το σύστημα που αναπτύχθηκε ακολουθεί την κλασσική παραταξιακή διάταξη όλων των βασισμένων στο διαδίκτυο εφαρμογών (web – based applications) και για λόγους συνοχής και καλύτερης απόδοσης του συστήματος, καθώς και για την δημιουργία ενός προτύπου για τις jsp (java server pages) σελίδες που υλοποιούν την διαπροσωπεία (interface) της εφαρμογής επιλέχθηκε όλη η λογική να εισαχθεί σε ένα EJB (Enterprise Java Bean). Παρακάτω ακολουθεί το παραταξιακό διάγραμμα της εφαρμογής για μια επισκόπηση του τρόπου με τον οποίο η εφαρμογή είναι ανεπτυγμένη:

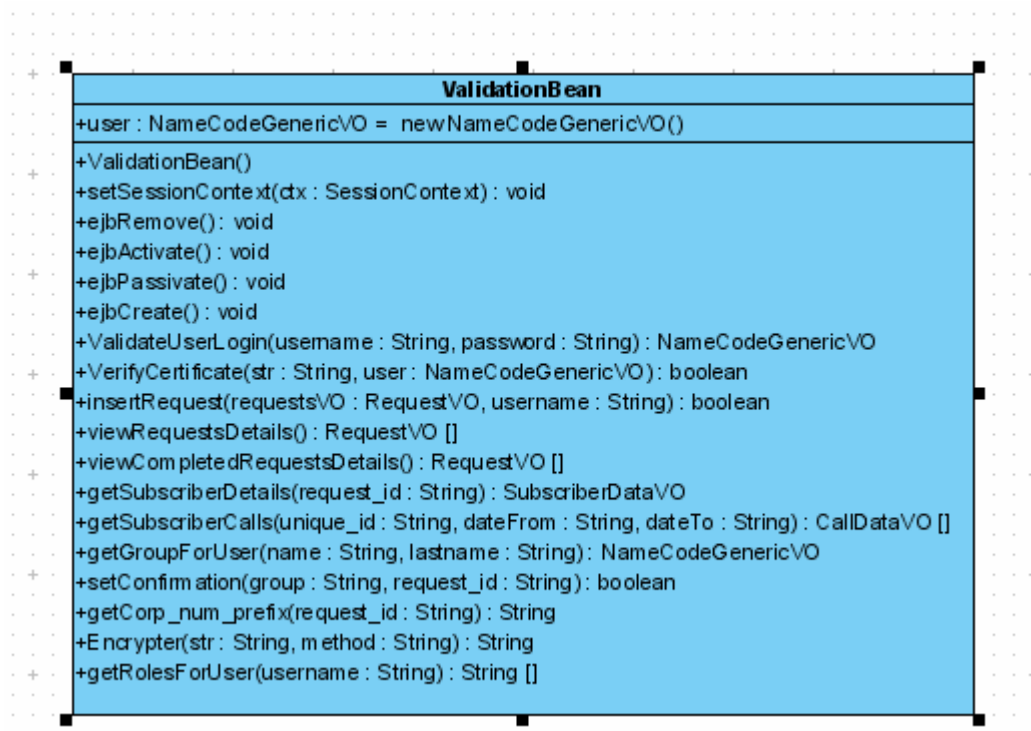


Σχήμα 3.11 Παραταξιακό Διάγραμμα Συστήματος

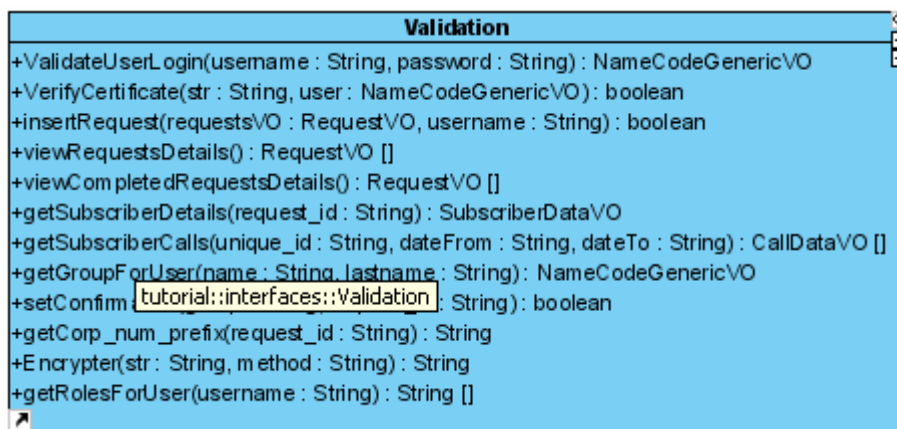
3.6.2 Διαγράμματα κλάσεων UML.

Λόγω της πολυπλοκότητας της εφαρμογής δεν είναι δυνατή η παράθεση του ίδιου του κώδικα της εφαρμογής γιατί κάτι τέτοιο όχι μόνο αποπροσανατολίζει την προσπάθεια απόδοσης της λογικής του συστήματος αλλά ξεφεύγει και από τους σκοπούς του παρόντος κειμένου. Ωστόσο είναι σκόπιμη μια συνοπτική παρουσίαση των κλάσεων του συστήματος και των μεθόδων τους για να είναι εύκολη η κατανόηση της λειτουργίας του συστήματος καθώς και να εξυπηρετούν ένα είδος API (Application

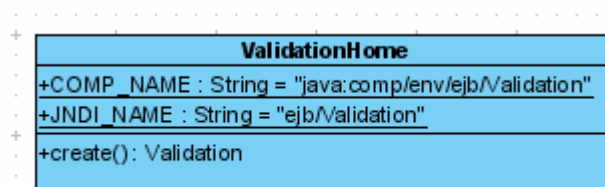
Programming Interface). Για τους σκοπούς της παρουσίασης θα χρησιμοποιηθεί το πρότυπο UML.



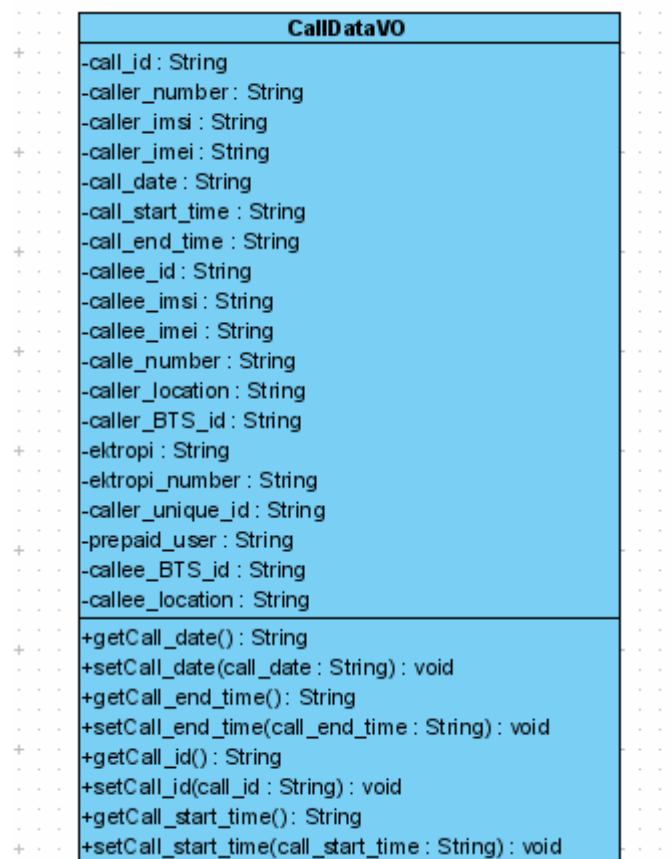
Σχήμα 3.12 Διάγραμμα κλάσεων ValidationBean



Σχήμα 3.13 Διάγραμμα κλάσεων Validation



Σχήμα 3.14 Διάγραμμα κλάσεων ValidationHome



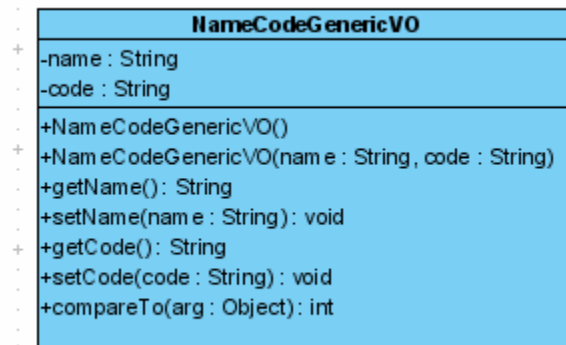
Σχήμα 3.15 Διάγραμμα κλάσεων CallDataVO (α)

```

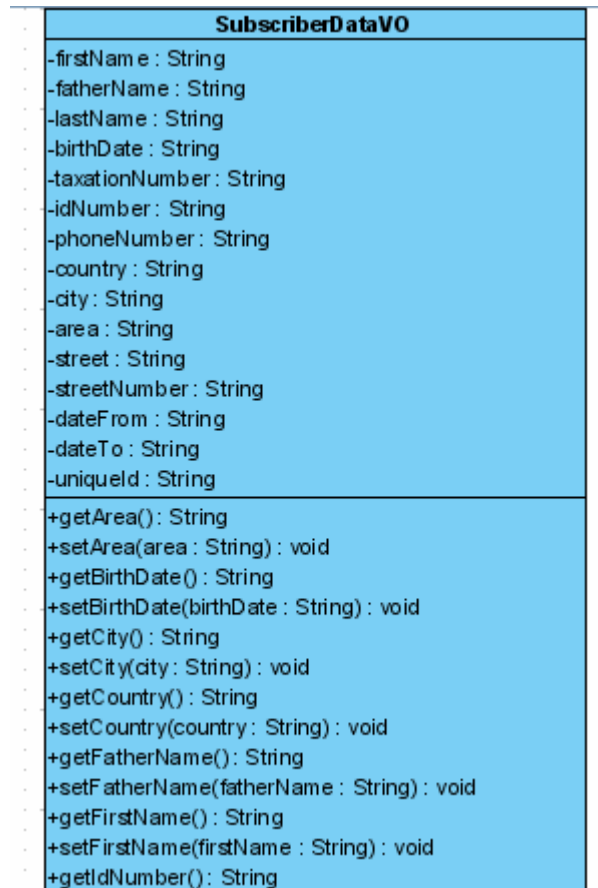
+getCallee_number(): String
+setCallee_number(callee_number: String): void
+getCaller_number(): String
+setCaller_number(callee_number: String): void
+getCallee_BTS_id(): String
+setCallee_BTS_id(callee_BTS_id: String): void
+getCallee_id(): String
+setCallee_id(callee_id: String): void
+getCallee_imei(): String
+setCallee_imei(callee_imei: String): void
+getCallee_imsi(): String
+setCallee_imsi(callee_imsi: String): void
+getCallee_location(): String
+setCallee_location(callee_location: String): void
+getCaller_BTS_id(): String
+setCaller_BTS_id(caller_BTS_id: String): void
+getCaller_imei(): String
+setCaller_imei(caller_imei: String): void
+getCaller_imsi(): String
+setCaller_imsi(caller_imsi: String): void
+getCaller_location(): String
+setCaller_location(caller_location: String): void
+getCaller_unique_id(): String
+setCaller_unique_id(caller_unique_id: String): void
+getEktropi(): String
+setEktropi(ektropi: String): void
+getEktropi_number(): String
+setEktropi_number(ektropi_number: String): void
+getPrepaid_user(): String
+setPrepaid_user(prepaid_user: String): void

```

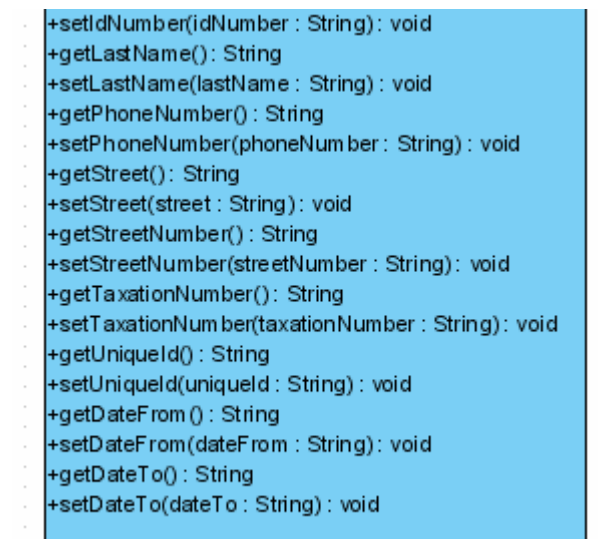
Σχήμα 3.16 Διάγραμμα κλάσεων CallDataVO (β)



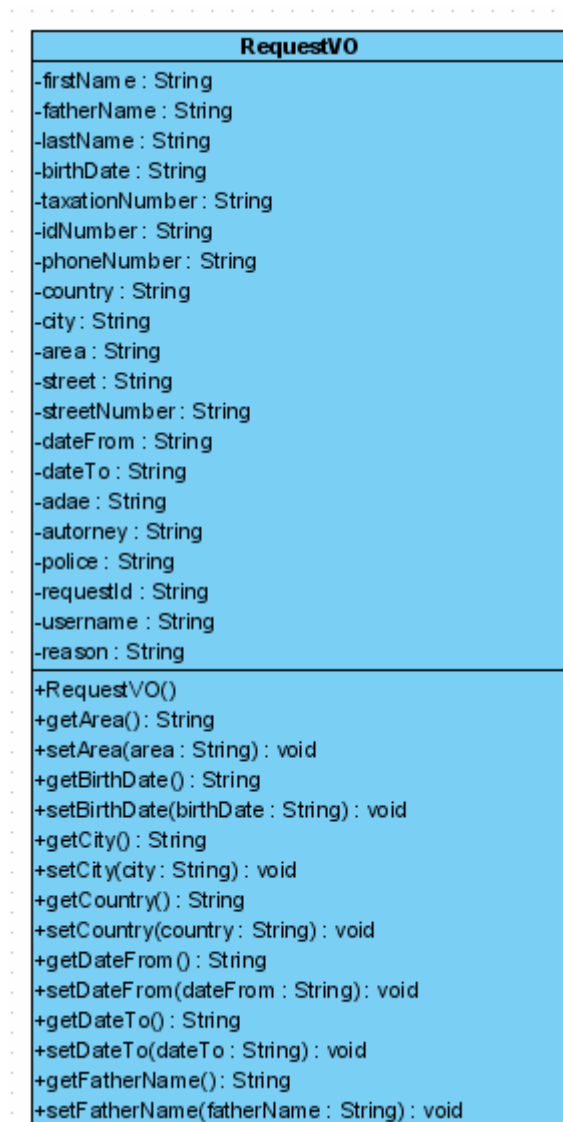
Σχήμα 3.17 Διάγραμμα κλάσεων NameCodeGenericVO



Σχήμα 3.18 Διάγραμμα κλάσεων SubscriberDataVO (part 1)



Σχήμα 3.19 Διάγραμμα κλάσεων SubscriberDataVO (part 2)



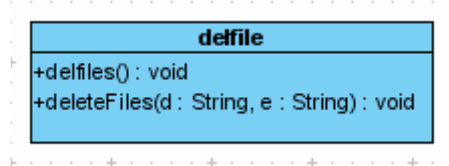
Σχήμα 3.20 Διάγραμμα κλάσεων RequestVO (part 1)

```

+getFirstName() : String
+setFirstName(firstName : String) : void
+getIdNumber() : String
+setIdNumber(idNumber : String) : void
+getLastName() : String
+setLastName(lastName : String) : void
+getPhoneNumber() : String
+setPhoneNumber(phoneNumber : String) : void
+getStreet() : String
+setStreet(street : String) : void
+getStreetNumber() : String
+setStreetNumber(streetNumber : String) : void
+getTaxationNumber() : String
+setTaxationNumber(taxationNumber : String) : void
+getAdae() : String
+setAdae(adae : String) : void
+getAutorney() : String
+setAutorney(autorney : String) : void
+getPolice() : String
+setPolice(police : String) : void
+getRequestId() : String
+setRequestId(requestId : String) : void
+getUsername() : String
+setUsername(username : String) : void
+getReason() : String
+setReason(reason : String) : void

```

Σχήμα 3.21 Διάγραμμα κλάσεων RequestVO (part 2)



Σχήμα 3.22 Διάγραμμα κλάσεων delfile

4

4 Τεχνολογίες και Εργαλεία που Χρησιμοποιήθηκαν

Στο κεφάλαιο αυτό περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν από πλευράς πλατφόρμας και εργαλείων ανάπτυξης του λογισμικού, όσο και αναφορικά με τη γλώσσα προγραμματισμού που χρησιμοποιήθηκε, τα πρωτόκολλα καθώς και οι τεχνικές απαιτήσεις της εφαρμογής σε hardware και software στην πλευρά του εξυπηρετητή, επίσης περιγράφεται αναλυτικά και ο τρόπος υλοποίησης της όλης εφαρμογής.

4.1 Τεχνικές Απαιτήσεις Συστήματος.

Για την εγκατάσταση και τον έλεγχο της εφαρμογής χρησιμοποιήθηκε Προσωπικός Ηλεκτρονικός Υπολογιστής (H/Y) με λειτουργικό σύστημα Windows XP Home Edition Service Pack 2, ωστόσο οι απαιτήσεις της εφαρμογής δεν απαιτούν ιδιαίτερα υψηλή υπολογιστική ισχύ καθιστώντας την ευέλικτη και υλοποιήσιμη σε έναν μέσης υπολογιστικής ισχύς H/Y.

4.2 Προγραμματιστικά Εργαλεία-Εφαρμογές

4.2.1 Εργαλείο Ανάπτυξης Εφαρμογής

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε η εφαρμογή Eclipse που είναι μια ελεύθερη εφαρμογή για τη δημιουργία εφαρμογών σε γλώσσα Java που αποτέλεσε και τη γλώσσα προγραμματισμού της εφαρμογής μας. Συγκεκριμένα χρησιμοποιήθηκε η έκδοση JBossIDE Eclipse 1.6 στην οποία έχουν προστεθεί διάφορες βιβλιοθήκες (JARs) οι οποίες θεωρήθηκαν απαραίτητες καθώς και Επιπρόσθετα Στοιχεία (Plugins).

4.2.2 Εξυπηρετητής Εφαρμογής

Ως Εξυπηρετητής Εφαρμογής (Application Server) χρησιμοποιήθηκε ο JBoss. Μία ιδιαίτερα εύχρηστη εφαρμογή που είναι διαθέσιμη ελεύθερα στο διαδίκτυο, για μη εμπορικές εφαρμογές. Συγκεκριμένα χρησιμοποιήθηκε η τελευταία «σταθερή»

(Stable) έκδοση η οποία αυτή τη στιγμή είναι η JBoss-4.2.0.GA. Η εφαρμογή αυτή θα αποτελέσει και τον Εξυπηρετητή της Εφαρμογής μας, ρυθμίζοντας κάποιες παραμέτρους του όπως περιγράφονται πιο κάτω.

4.2.3 Εξυπηρετητής Διαδικτύου

Ως Εξυπηρετητής Διαδικτύου (Web Server) χρησιμοποιήθηκε ο JBossWS ο οποίος είναι ενσωματωμένος στην προαναφερθείσα έκδοση η JBoss-4.2.0.GA που χρησιμοποιήσαμε .

4.2.4 Γλώσσα Προγραμματισμού

Η γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την Εφαρμογή είναι η java, και συγκεκριμένα η έκδοση jdk1.5.0_09. Τα πλεονεκτήματα της γλώσσας αυτής μπορούν συνοπτικά να αναφερθούν ότι είναι :

- ✓ **Απλότητα**
- ✓ **Μικρό Μέγεθος**
- ✓ **Αντικειμενοστρεφής Γλώσσα**
- ✓ **Κατανεμημένη**
- ✓ **Αξιόπιστη**
- ✓ **Ασφαλής**
- ✓ **Ανεξάρτητη Αρχιτεκτονικής**
- ✓ **Μεταφερσιμότητα**
- ✓ **Υψηλή απόδοση**
- ✓ **Δυναμική**

4.2.5 Βάση Δεδομένων (Database server)

Για την υλοποίηση της βάσης δεδομένων που χρειάζεται η εφαρμογή χρησιμοποιήθηκε η MySQL, και συγκεκριμένα η έκδοση 5.0.11 η οποία διατίθεται δωρεάν και μπορεί να βρεθεί εύκολα στο διαδίκτυο. Η βάση δεδομένων MySQL μπορεί να επικοινωνήσει με κατάλληλο JDBC driver με την εφαρμογή που έχουμε αναπτύξει σε Java. Ένα επιπλέον πλεονέκτημα της MySQL είναι ότι δεν είναι ιδιαίτερα απαιτητική σε πόρους (μνήμη, επεξεργαστική ισχύς, χώρος αποθήκευσης) και γι' αυτό το λόγο χρησιμοποιήθηκε κατά τη διαδικασία ανάπτυξης του συστήματος. Για την πραγματική υλοποίηση του συστήματος, θα μπορούσε να χρησιμοποιηθεί οποιαδήποτε άλλη βάση δεδομένων του εμπορίου η οποία θα πρέπει να έχει συμμόρφωση με το πρότυπο SQL 93 καθώς και τον κατάλληλο JDBC driver για την επικοινωνία με την εφαρμογή.

4.2.6 Κρυπτογράφηση Δεδομένων

Τα δεδομένα στη βάση βρίσκονται κρυπτογραφημένα. Πιο συγκεκριμένα χρησιμοποιήθηκε ο αλγόριθμος AES της Java. Πρόκειται για μια συμμετρική κρυπτογράφηση με το «κλειδί» της κωδικοποίησης να φυλάσσεται από το σύστημα. Με τη χρήση κρυπτογραφίας σε περίπτωση παράνομης πρόσβασης στη βάση από τρίτους αποφεύγεται η κτήση των δεδομένων από μη εξουσιοδοτημένα πρόσωπα.

Έτσι, για την ασφάλεια των δεδομένων στη βάση σε περίπτωση παράνομης πρόσβασης από τρίτους ή σε περίπτωση κακόβουλης διαχείρισής της επιβάλλεται η διαφύλαξη των δεδομένων σε κρυπτογραφημένη μορφή. Με τον τρόπο αυτό επιτυγχάνεται η διασφάλιση του απορρήτου των δεδομένων.

Για την κρυπτογράφηση των δεδομένων στη βάση χρησιμοποιήθηκε ο Advanced Encryption Standard αλγόριθμος. Δεδομένου ότι πρόκειται για έναν αλγόριθμο συμμετρικής κρυπτογράφησης αυτό απαιτεί την διαφύλαξη του κλειδιού κωδικοποίησης/αποκωδικοποίησης. Για τον σκοπό αυτό στο φυσικό μηχάνημα που φιλοξενεί την εφαρμογή πρέπει να εγκατασταθεί και ένα κλειδί που βρίσκεται αποθηκευμένο σε ένα .txt αρχείο. Για την διασφάλιση του κωδικού το αρχείο πρέπει να αποθηκευτεί σε ένα path στο οποίο μετέπειτα δεν θα παρέχεται read/write πρόσβαση σε κανέναν άλλον εκτός της εφαρμογής.

Η χρήση του AES αλγορίθμου αποφασίστηκε λόγω του γεγονότος ότι αποτελεί τον πιο δημοφιλή αλγόριθμο για κρυπτογραφία συμμετρικού κλειδιού χρησιμοποιούμενου σε ευρεία κλίμακα, καθώς και για την ταχύτητά του, τόσο σε software και hardware, όσο και για την ευκολία που προσφέρει στην υλοποίηση και τις μικρές απαιτήσεις σε μνήμη που έχει.

4.3 Τεχνολογίες που Χρησιμοποιήθηκαν

4.3.1 Java 2 Platform, Enterprise Edition (J2EE)

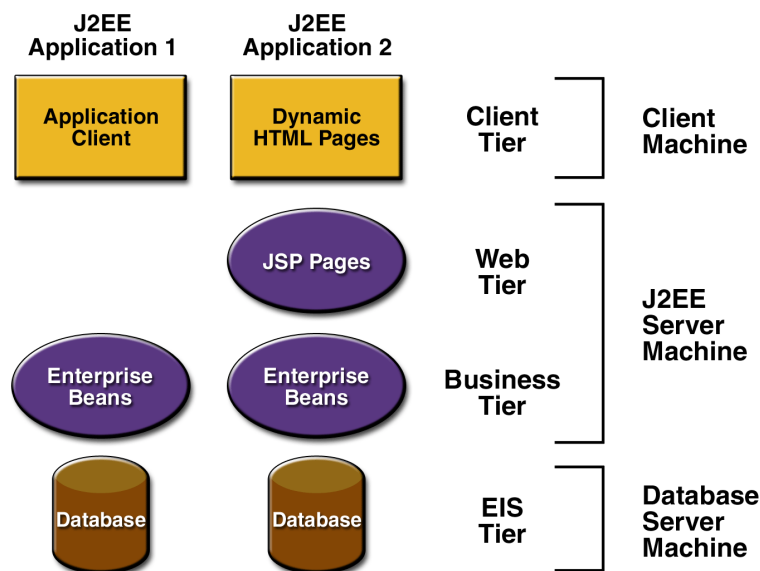
Η J2EE καθορίζει κάποια στάνταρ για την ανάπτυξη πολύ-επίπεδου επιχειρησιακού λογισμικού. Αυτό που ουσιαστικά γίνεται στην J2EE πλατφόρμα είναι να απλοποιούνται οι διαδικασίες ανάπτυξης των εφαρμογών, βασίζοντας την σε συγκεκριμένα στάνταρ και χρησιμοποιώντας την έννοια των επαναχρησιμοποιούμενων τμημάτων κώδικα, τα οποία υποστηρίζονται από ένα σύνολο υπηρεσιών. Τελικά η ανάπτυξη απλοποιείται αρκετά αφού πολλές λεπτομέρειες της συμπεριφοράς των εφαρμογών ρυθμίζονται αυτόματα, χωρίς ανάγκη για πολύπλοκο προγραμματισμό. Η J2EE χρησιμοποιεί πολλά χαρακτηριστικά της πλατφόρμας Java 2 Standard Edition.

4.3.1.1 Εφαρμογές Πολλών Βαθμίδων

Η πλατφόρμα J2EE χρησιμοποιεί ένα μοντέλο κατανεμημένων εφαρμογών πολλών βαθμίδων για τις εταιρικές εφαρμογές. Η λογική της εφαρμογής διαιρείται σε συστατικά τμήματα σύμφωνα με την λειτουργία τους και τα διάφορα συστατικά που συνθέτουν την εφαρμογή εγκαθίστανται σε διαφορετικές μηχανές ανάλογα με την βαθμίδα του διαβαθμισμένου J2EE περιβάλλοντος στο οποίο ανήκουν. Στο σχήμα 4.1 φαίνονται δύο διαβαθμισμένες J2EE εφαρμογές διηρημένες στις βαθμίδες που περιγράφονται ακολούθως.

- Βαθμίδα-πελάτη : συστατικά που τρέχουν στο μηχάνημα του πελάτη
- Βαθμίδα - ιστού : συστατικά που τρέχουν στο J2EE server
- Βαθμίδα - επιχείρησης : συστατικά που τρέχουν στο J2EE εξυπηρετητή
- Βαθμίδα συστήματος πληροφοριών επιχείρησης (Enterprise information system EIS) : λογισμικό που τρέχει στον EIS εξυπηρετητή.

Παρά το γεγονός ότι μια J2EE εφαρμογή μπορεί να αποτελείται από τρεις ή τέσσερις βαθμίδες, όπως φαίνεται και στο σχήμα 4.1 , οι εφαρμογές J2EE γενικά θεωρούνται ως τριών βαθμίδων εφαρμογές, αφού κατανέμονται κυρίως σε τρία μέρη : το μηχάνημα του πελάτη, το μηχάνημα του J2EE server και τη βάση δεδομένων ή τα legacy machine στο back end. Οι εφαρμογές τριών βαθμίδων που τρέχουν κατά αυτό το τρόπο, επεκτείνουν το κλασικό μοντέλο δύο βαθμίδων του πελάτη-εξυπηρετητή, τοποθετώντας έναν πολυνηματικό εξυπηρετητή εφαρμογών ανάμεσα στην εφαρμογή του πελάτη και στις μονάδες αποθήκευσης.



Σχήμα 4.1 Εφαρμογές πολλών βαθμίδων. Συστατικά J2EE

Οι J2EE εφαρμογές αποτελούνται από συστατικά. Ένα J2EE συστατικό είναι μια ανεξάρτητη και λειτουργική μονάδα λογισμικού, η οποία έχει συμπεριληφθεί σε μια J2EE εφαρμογή με τις σχετιζόμενες σε αυτή κλάσεις και φακέλους και επικοινωνεί με άλλα συστατικά. Η J2EE προδιαγραφή καθορίζει τα ακόλουθα J2EE συστατικά :

- Εφαρμογές πελατών και μικροεφαρμογές (applets), είναι συστατικά που τρέχουν στον πελάτη.
- Java Servlet και JavaServer Pages (JSP), είναι συστατικά που τρέχουν στο εξυπηρετητή.
- Enterprise JavaBeans (EJB) συστατικά, είναι επιχειρησιακά συστατικά που τρέχουν στον εξυπηρετητή.

Τα συστατικά J2EE γράφονται σε γλώσσα προγραμματισμού Java και διερμηνεύονται όπως κάθε πρόγραμμα, στην γλώσσα αυτή. Η διαφορά μεταξύ των J2EE συστατικών και των κλασσικών Java κλάσεων είναι ότι τα συστατικά J2EE «συναρμολογούνται» σε μια J2EE εφαρμογή, επαληθεύεται ότι είναι κατάλληλα δομημένα και σύμφωνα με την J2EE προδιαγραφή και αναπτύσσονται στην παραγωγή, όπου τα διαχειρίζονται J2EE εξυπηρετητές.

4.3.1.1.1 Πελάτες (J2EE Clients)

1 Εφαρμογές Εξυπηρετούμενων (Application Clients)

Μια εφαρμογή εξυπηρετούμενου, τρέχει στο μηχάνημα του εξυπηρετούμενου και παρέχει στους χρήστες έναν τρόπο να έχουν στην διάθεση τους λειτουργίες οι οποίες απαιτούν πλουσιότερο περιβάλλον αλληλεπίδρασης απ' ότι μπορούν να προσφέρουν οι markup γλώσσες. Τυπικά μια εφαρμογή έχει ένα γραφικό περιβάλλον αλληλεπίδρασης (GUI), δημιουργούμενο από το Swing ή από το Abstract Window Toolkit (AWT) API. Αλλά και ένα περιβάλλον που να αποτελείται από γραμμή εντολών είναι πιθανό.

Οι εφαρμογές των εξυπηρετούμενων έχουν απευθείας πρόσβαση στα enterprise beans που τρέχουν στη βαθμίδα επιχείρησης. Όμως εάν οι απαιτήσεις της εφαρμογής το επιβάλλουν, μια εφαρμογή εξυπηρετούμενου μπορεί να ανοίξει μια HTTP σύνδεση για να εδραιώσει επικοινωνία με ένα servlet το οποίο τρέχει στη βαθμίδα του ιστού.

2 Εξυπηρετούμενοι ιστού (Web Client)

Ένας εξυπηρετούμενος ιστού (Web Client) αποτελείται από δύο τμήματα :

- α) Δυναμικές σελίδες ιστού, οι οποίες περιέχουν διάφορους τύπους markup γλωσσών (HTML, XML και άλλες), που δημιουργούνται από τα συστατικά ιστού τα οποία με τη σειρά τους τρέχουν στη βαθμίδα του ιστού.
- β) Ένας φυλλομετρητής κατάλληλος για να αποδίδει τις σελίδες που λαμβάνει από τον εξυπηρετητή.

Ένας εξυπηρετούμενος ιστού μερικές φορές αποκαλείται και *thin client*. Οι thin clients συνήθως δεν επικοινωνούν με βάσεις δεδομένων και δεν εκτελούν σύνθετες επιχειρησιακές λειτουργίες. Όταν χρησιμοποιείς έναν thin client, τέτοιες βαριές λειτουργίες αναλαμβάνονται από τα enterprise beans τα οποία τις εκτελούν στους J2EE εξυπηρετητές, απ' όπου μπορούν να έχουν πρόσβαση σε ανώτερα επίπεδα ασφάλειας, αυξημένη ταχύτητα, καθώς και αξιοπιστία των τεχνολογιών J2EE για τους εξυπηρετητές.

3 Εφαρμογές (Applets)

Μια σελίδα ιστού μπορεί να εμπεριέχει μια εφαρμογή ή αλλιώς μια applet. Μια τέτοια εφαρμογή είναι μια μικρή εφαρμογή χρήστη γραμμένη σε Java γλώσσα προγραμματισμού, η οποία εκτελείται στην εικονική μηχανή Java που βρίσκεται εγκατεστημένη στον φυλλομετρητή του εξυπηρετούμενου. Συνεπώς τα συστήματα των εξυπηρετούμενων θα χρειαστούν την εγκαταστημένη εικονική μηχανή Java καθώς και μια πολιτική ασφάλειας για τους φακέλους, έτσι ώστε η εφαρμογή να μπορέσει επιτυχώς να εκτελεστεί στον φυλλομετρητή τους.

Τα συστατικά του ιστού (Web components) αποτελούν το πιο χρησιμοποιούμενο API (πρωτόκολλο αλληλεπίδρασης εφαρμογών) για την δημιουργία ενός προγράμματος για τον εξυπηρετούμενο, αφού προσφέρουν ένα τρόπο να διαχωρίζεται ο προγραμματισμός εφαρμογών από την δημιουργία σελίδων ιστού. Το προσωπικό που εμπλέκεται στην δημιουργία των σελίδων του ιστού, δεν χρειάζεται να καταλαβαίνει την σύνταξη της γλώσσας Java για να κάνουν τη δουλειά τους.

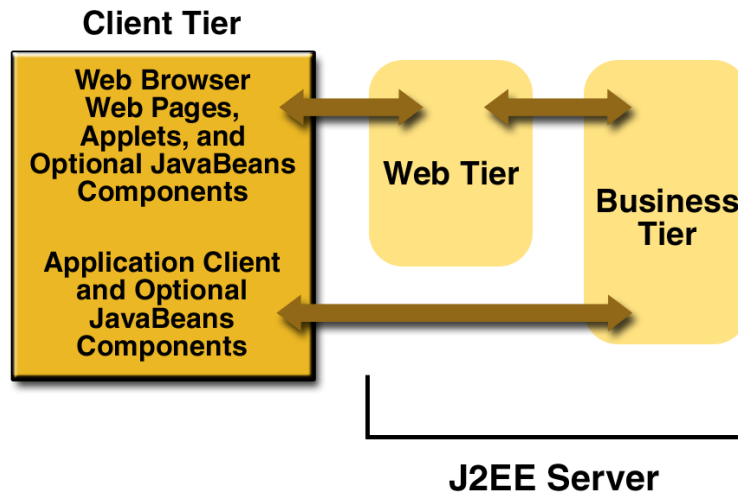
4 Η αρχιτεκτονική των JavaBeans συστατικών

Οι βαθμίδες του εξυπηρετητή και του εξυπηρετούμενου μπορεί επιπλέον να περιέχουν συστατικά βασισμένα στην αρχιτεκτονική των JavaBeans, για να διαχειρίζονται την ροή των δεδομένων μεταξύ των εφαρμογών των εξυπηρετούμενων και των συστατικών που τρέχουν στον J2EE εξυπηρετητή ή μεταξύ των συστατικών του εξυπηρετητή και μιας βάσης δεδομένων.

Τα JavaBeans είναι συστατικά τα οποία έχουν κάποιες ιδιότητες στις οποίες μπορούμε να έχουμε πρόσβαση με τις μεθόδους get και set. Τα συστατικά JavaBeans που χρησιμοποιούνται κατά αυτόν τον τρόπο είναι ουσιαστικά απλά στο σχεδιασμό και την υλοποίηση, αλλά πρέπει να συμμορφώνονται με τις συμβάσεις για τα ονόματα και τον σχεδιασμό, που καθορίζονται από την αρχιτεκτονική των JavaBeans συστατικών.

5 Επικοινωνίες των J2EE εξυπηρετητών

Το σχήμα 4.2 δείχνει τα διάφορα συστατικά στοιχεία που αποτελούν τη βαθμίδα του εξυπηρετητή. Ο εξυπηρετητής επικοινωνεί με τη βαθμίδα επιχείρησης, η οποία τρέχει στον J2EE εξυπηρετητή, είτε απευθείας, είτε, όπως στην περίπτωση μιας εφαρμογής που τρέχει στον φυλλομετρητή του εξυπηρετούμενου, δια μέσου των JSP σελίδων ή των servlets που τρέχουν στη βαθμίδα του ιστού (Web Tier).



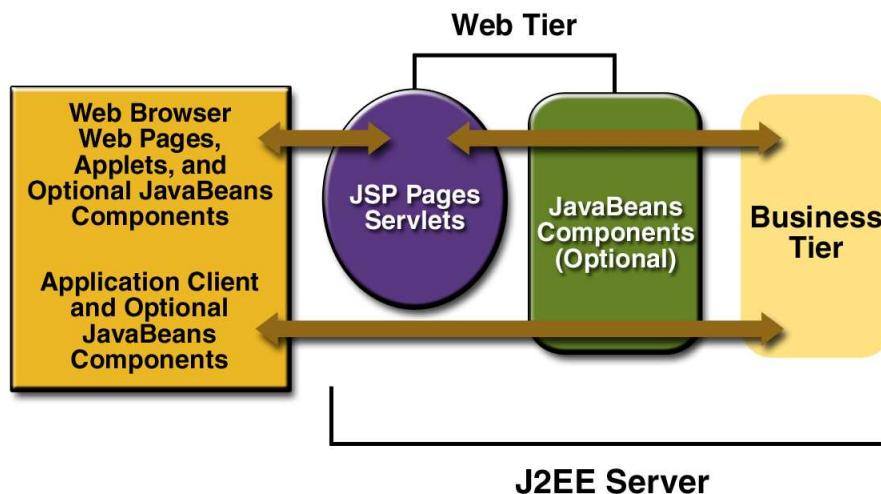
Σχήμα 4.2 Συστατικά στοιχεία της βαθμίδας του εξυπηρετητή.

4.3.1.1.2 Συστατικά Ιστού (Web Components)

Τα J2EE συστατικά του ιστού είναι είτε servlets είτε σελίδες που έχουν δημιουργηθεί από την JSP τεχνολογία (JSP pages). Τα servlets είναι κλάσεις της γλώσσας προγραμματισμού Java, οι οποίες δυναμικά επεξεργάζονται αιτήσεις και δημιουργούν απαντήσεις. Οι σελίδες JSP είναι έγγραφα βασισμένα σε κείμενο που λειτουργούν σαν τα servlets, αλλά επιτρέπουν πιο εύκολη δημιουργία του στατικού περιεχομένου της σελίδας.

Οι στατικές HTML σελίδες και οι εφαρμογές πακετάρονται με τα συστατικά του ιστού κατά την διάρκεια της συναρμολόγησης της εφαρμογής, αλλά δεν θεωρούνται συστατικά του ιστού από την J2EE προδιαγραφή. Τυποποιημένες κλάσεις από την πλευρά του εξυπηρετητή μπορούν επίσης να πακετάρονται με συστατικά ιστού και όπως και οι HTML σελίδες, δεν θεωρούνται συστατικά ιστού.

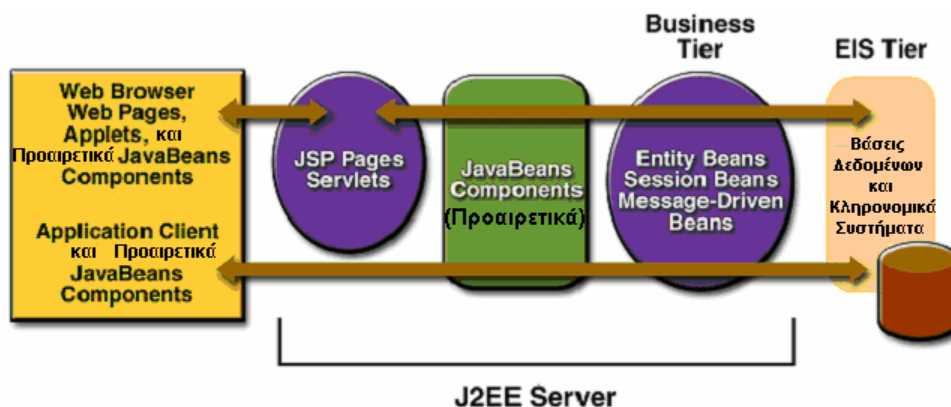
Όπως φαίνεται και στο σχήμα 4.3, η βαθμίδα του ιστού, όπως και η βαθμίδα του εξυπηρετούμενου, ίσως να περιέχει κάποιο συστατικό JavaBean για να χειρίζεται την είσοδο του χρήστη και να στέλνει αυτή την είσοδο στα Enterprise Beans που τρέχουν στη βαθμίδα της επιχείρησης προς επεξεργασία.



Σχήμα 4.3 Βαθμίδα ιστού και J2EE εφαρμογές.

4.3.1.1.3 Επιχειρησιακά Συστατικά (Business Components)

Ο κώδικας για την βαθμίδα των επιχειρησιακών συστατικών, όπως είναι λογικό αντιμετωπίζει τις ανάγκες μιας συγκεκριμένης περιοχής επιχειρήσεων κάθε φορά, όπως τραπεζικές, οικονομικές και πωλήσεων, οι οποίες διαχειρίζονται από enterprise beans που τρέχουν στη βαθμίδα επιχείρησης. Το σχήμα 4.4 δείχνει πως ένα enterprise bean λαμβάνει δεδομένα από ένα πρόγραμμα εξυπηρετούμενου, τα επεξεργάζεται και τα στέλνει στη βαθμίδα του συστήματος πληροφοριών για αποθήκευση. Ένα enterprise bean επιπλέον ανακτά δεδομένα από αποθήκες δεδομένων, τα επεξεργάζεται και (αν είναι απαραίτητο), τα στέλνει πίσω στο πρόγραμμα του εξυπηρετούμενου.



Σχήμα 4.4 Enterprise JavaBeans

Υπάρχουν τρία είδη enterprise beans: session beans, entity beans και message-driven beans. Ένα session bean αναπαριστά μια προσωρινή συνομιλία με τον εξυπηρετούμενο. Όταν ο εξυπηρετούμενος τελειώσει την εκτέλεση, το session bean και τα δεδομένα του χάνονται. Αντίθετα, το entity bean αντιπροσωπεύει σταθερά δεδομένα τα οποία αποθηκεύονται σε ένα πίνακα μιας βάσης δεδομένων. Αν ο εξυπηρετούμενος τερματίσει ή αν ο εξυπηρετητής «πέσει», οι υποκείμενες υπηρεσίες

διασφαλίζουν ότι τα δεδομένα του entity bean έχουν αποθηκευτεί. Ένα message-driven bean συνδυάζει χαρακτηριστικά των session beans και των message listeners του Java Message Service, επιτρέποντας ένα συστατικό της επιχείρησης να δέχεται JMS μηνύματα ασύγχρονα. Στα Enterprise JavaBeans θα αναφερθούμε αναλυτικότερα σε επόμενη παράγραφο.

4.3.1.2 J2EE Containers

Κανονικά, η συγγραφή διαβαθμισμένων εφαρμογών είναι δύσκολη, αφού εμπεριέχει πολλές γραμμές περίπλοκου κώδικα για τον χειρισμό των συναλλαγών, των καταστάσεων, του πολυνηματισμού (multithread), των πόρων και άλλων χαμηλού επιπέδου λεπτομερειών. Η αρχιτεκτονική J2EE που βασίζεται σε συστατικά και στην ανεξαρτησία πλατφόρμας κάνει τις εφαρμογές J2EE πιο εύκολες στην δημιουργία τους, γιατί η επιχειρηματική λογική οργανώνεται σε επαναχρησιμοποιήσιμα συστατικά. Επιπρόσθετα κάθε J2EE εξυπηρετητής προσφέρει ως υπόβαθρο υπηρεσίες με την μορφή ενός container για κάθε είδος συστατικού. Επειδή ο προγραμματιστής δεν χρειάζεται να δημιουργήσει αυτές τις υπηρεσίες ο ίδιος, είναι ελεύθερος να συγκεντρωθεί στην επίλυση των επιχειρηματικών προβλημάτων τα οποία αντιμετωπίζει.

4.3.1.2.1 Υπηρεσίες των Containers

Οι containers αποτελούν το interface ανάμεσα στα συστατικά και στη χαμηλού επιπέδου λειτουργικότητα ανάλογα με την πλατφόρμα η οποία υποστηρίζει τα συστατικά. Πριν μπορέσουν να εκτελεστούν συστατικά του ιστού ή enterprise beans ή εφαρμογές εξυπηρετούμενου, πρέπει αυτά τα συστατικά να έχουν συναρμολογηθεί σε μια J2EE υπομονάδα και να έχουν αναπτυχθεί μέσα στον container τους.

Η διαδικασία συναρμολόγησης εμπεριέχει τον καθορισμό των ρυθμίσεων του container για κάθε συστατικό που ανήκει στην J2EE εφαρμογή, καθώς επίσης και για την J2EE εφαρμογή. Οι ρυθμίσεις του container προσαρμόζουν την υποστήριξη που προσφέρει ο J2EE εξυπηρετητής, όπως υπηρεσίες σαν την ασφάλεια, διαχείριση συναλλαγών, Java Naming & Directory Interface (JNDI) lookup και την απομακρυσμένη συνδετικότητα. Εδώ παρουσιάζουμε μερικές από τις σημαντικότερες:

- Το μοντέλο ασφάλειας J2EE μας επιτρέπει να ρυθμίσουμε ένα συστατικό ιστού ή ένα enterprise bean, έτσι ώστε στους πόρους του συστήματος να έχουν πρόσβαση μόνο εγκεκριμένοι χρήστες.
- Το μοντέλο συναλλαγών J2EE μας επιτρέπει να καθορίσουμε σχέσεις μεταξύ μεθόδων που συνεισφέρουν σε μια συναλλαγή, έτσι ώστε όλες οι μέθοδοι σε μια συναλλαγή να μπορούν να θεωρηθούν ως μια μονάδα.
- Η υπηρεσία JNDI lookup παρέχει ένα ομογενοποιημένο περιβάλλον αλληλεπίδρασης σε πολλαπλές υπηρεσίες ονομασίας και καταλόγων στην

επιχείρηση, έτσι ώστε τα συστατικά των εφαρμογών να μπορούν να έχουν πρόσβαση σε αυτές τις υπηρεσίες ονομάτων και καταλόγων.

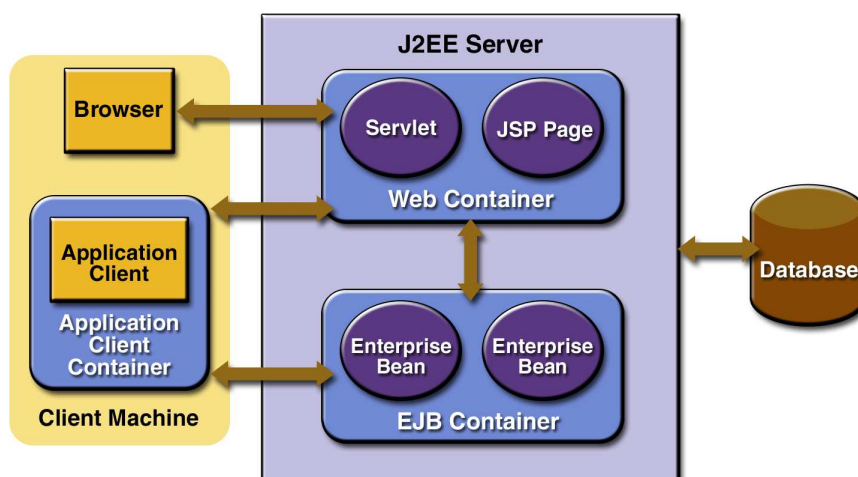
- Το J2EE μοντέλο απομακρυσμένης συνδετικότητας διαχειρίζεται χαμηλού επιπέδου επικοινωνιακά θέματα μεταξύ των εξυπηρετούμενων και των enterprise beans. Αφού δημιουργηθεί ένα enterprise bean, ένας πελάτης καλεί τις μεθόδους του σαν να βρίσκονταν στην ίδια εικονική μηχανή.

Επειδή η αρχιτεκτονική J2EE παρέχει προσαρμοζόμενες υπηρεσίες, τα συστατικά των εφαρμογών που βρίσκονται στην ίδια J2EE εφαρμογή μπορούν να συμπεριφέρονται διαφορετικά με βάση το που έχουν αναπτυχθεί. Για παράδειγμα ένα enterprise bean μπορεί να έχει ρυθμίσεις ασφαλείας που να του επιτρέπουν ένα συγκεκριμένο επίπεδο πρόσβασης στα δεδομένα μιας βάσης δεδομένων σε ένα περιβάλλον παραγωγής και ένα άλλο επίπεδο πρόσβασης στα δεδομένα της βάσης δεδομένων σε ένα άλλο περιβάλλον παραγωγής.

Ο container επίσης διαχειρίζεται και μη προσαρμοζόμενες υπηρεσίες όπως τους κύκλους ζωής των enterprise beans και των servlets, την επαναχρησιμοποίηση των πόρων σύνδεσης με τις βάσεις δεδομένων, την διατήρηση δεδομένων και άλλων. Αν και η διατήρηση των δεδομένων είναι μη προσαρμοζόμενη υπηρεσία, η J2EE αρχιτεκτονική μας επιτρέπει να παραμερίσουμε την διαχείριση της διατήρησης των δεδομένων από τον container συμπεριλαμβάνοντας τον απαραίτητο κώδικα στα δικά μας enterprise beans όταν θέλουμε περισσότερο έλεγχο από αυτόν που προσφέρουν οι προκαθορισμένες ρυθμίσεις του container για την διατήρηση των δεδομένων. Για παράδειγμα μπορούμε να χρησιμοποιήσουμε διαχείριση διατήρησης δεδομένων από τα beans για να υλοποιήσουμε τις δικές μας μεθόδους ψαξίματος ή για να δημιουργήσουμε εξατομικευμένη cache για την βάση δεδομένων.

4.3.1.2.2 Τύποι Containers

Τα είδη των containers παρουσιάζονται σχηματικά στο Σχήμα 4.5.



Σχήμα 4.5 Τύποι Containers σε δομή εξυπηρετητή-εξυπηρετούμενου.

Οι τύποι των Containers είναι οι εξής:

- **J2EE εξυπηρετητής (Server Container)**

Ένας J2EE εξυπηρετητής παρέχει τους containers για τα Enterprise Java Beans και για τις εφαρμογές ιστού.

- **Enterprise Java Beans Container (EJB Container)**

Διαχειρίζεται την εκτέλεση των enterprise java beans για τις J2EE εφαρμογές. Τα enterprise beans και οι container τους τρέχουν σε J2EE εξυπηρετητές.

- **Container Ιστού (Web Container)**

Διαχειρίζεται την εκτέλεση των JSP σελίδων και των servlet συστατικών για τις J2EE εφαρμογές. Τα συστατικά του ιστού και οι container τους τρέχουν σε J2EE εξυπηρετητές.

- **Container εφαρμογών εξυπηρετούμενου (Client Application Container)**

Διαχειρίζεται την εκτέλεση των συστατικών των εφαρμογών του εξυπηρετούμενου. Οι εφαρμογές του εξυπηρετούμενου και οι container τους τρέχουν στον εξυπηρετούμενο.

- **Container εφαρμογών (Application Container)**

Διαχειρίζεται την εκτέλεση των εφαρμογών. Αποτελείται από έναν φυλλομετρητή ιστού και από μια πλατφόρμα Java ενσωματωμένη που τρέχουν στον εξυπηρετούμενο.

4.3.2 JavaServer Pages (JSP)

Με τη χρήση των JavaServer Pages μπορούμε να δημιουργήσουμε εύκολα περιεχόμενο ιστού τέτοιο που να περιέχει και δυναμικά και στατικά συστατικά. Η JSP τεχνολογία δημιουργήθηκε κατά την περίοδο όπου η τεχνολογία servlet κέρδιζε δημοτικότητα σαν μια από τις καλύτερες διαθέσιμες τεχνολογίες ιστού. Η τεχνολογία JSP δεν έχει ως σκοπό να αντικαταστήσει την τεχνολογία servlet, αφού ουσιαστικά αποτελεί μια προέκταση αυτής της τεχνολογίας, και αποτελεί κοινή πρακτική η παράλληλη χρήση τόσο των servlets, όσο και σελίδων JSP στην ίδια εφαρμογή ιστού.

Τα βασικά χαρακτηριστικά της JSP τεχνολογίας είναι ότι είναι μια γλώσσα για την ανάπτυξη JSP σελίδων, η οποία βασίζεται σε έγγραφα βασισμένα σε κείμενο, τα οποία περιγράφουν πως να επεξεργαστεί μια αίτηση και δημιουργηθεί μια απάντηση. Είναι μια γλώσσα εκφράσεων για πρόσβαση σε αντικείμενα στην πλευρά του εξυπηρετητή.

4.3.2.1 Τρόπος λειτουργίας JSP σελίδων.

Μια JSP σελίδα είναι ένα έγγραφο κειμένου που περιέχει δύο είδη κειμένου: στατικά δεδομένα, τα οποία μπορούν να εκφραστούν σε τυποποίηση κειμένου, όπως HTML, SVG, WML, και XML, και τα JSP στοιχεία που αποτελούν το δυναμικό περιεχόμενο.

Η κατάληξη των αρχείων με τον πηγαίο κώδικα για μια JSP σελίδα είναι .jsp. Η σελίδα μπορεί να αποτελείται από ένα αρχείο, το οποίο μπορεί να περιέχει άλλα αρχεία τα οποία περιέχουν είτε ολόκληρες JSP σελίδες ή τμήματα JSP σελίδων.

Μια JSP σελίδα εξυπηρετεί αιτήσεις ακριβώς όπως ένα servlet. Αυτό συμβαίνει αφού μέσα στο JSP container υπάρχει ένα ειδικό servlet που λέγεται servlet compiler. Ο servlet container είναι κατασκευασμένος έτσι ώστε να διαβιβάζει σε αυτόν τον μεταγλωττιστή σελίδας κάθε HTTP αίτηση με URL που να ταιριάζουν στην κατάληξη .jsp. Αυτός ο μεταγλωττιστής της σελίδας μετατρέπει έναν servlet container σε JSP container. Όταν μια σελίδα με κατάληξη .jsp κληθεί για πρώτη φορά ο μεταγλωττιστής της σελίδας αναλύει και μεταγλωττίζει την .jsp σελίδα σε μια κλάση servlet. Αν η μεταγλώττιση είναι επιτυχής, η jsp servlet κλάση φορτώνεται στην μνήμη, ή απλώς ανανεώνεται, αν ήδη υπάρχει στην μνήμη. Για αυτό το λόγο το servlet της μεταγλωττισμένης σελίδας συγκρίνει πάντα την χρονοσφραγίδα του jsp servlet με αυτή της jsp σελίδας. Αν η .jsp σελίδα είναι πιο πρόσφατη, τότε η επαναμεταγλώττιση είναι απαραίτητη. Με αυτή τη διαδικασία, όταν έχουν διαταχθεί, οι jsp σελίδες περνούν από την χρονοβόρα διαδικασία της μεταγλώττισης μόνο μια φορά.

Επειδή μετά την διάταξη μιας JSP σελίδας θα εμφανιζόταν μεγάλη καθυστέρηση λόγω του χρόνου που ξοδεύεται για την μεταγλώττιση του .jsp αρχείου σε jsp servlet, εφαρμόζεται ένας μηχανισμός που επιτρέπει σε μια .jsp σελίδα να προ-μεταγλωττίζεται πριν εμφανιστεί οποιαδήποτε αίτηση χρήστη για αυτήν.

Κατά τη διάρκεια της φάσης μετάφρασης μεταχειριζόμαστε διαφορετικά κάθε τύπο δεδομένων σε μια JSP σελίδα. Τα στατικά δεδομένα μετατρέπονται σε κώδικα, ο οποίος θα μεταδώσει τα δεδομένα στο stream απάντησης. Ειδικότερα, μεταχειριζόμαστε τα JSP στοιχεία ως εξής :

- Κατευθυντήριες οδηγίες χρησιμοποιούνται για να ελέγξουν πως ο container του ιστού μεταφράζει και εκτελεί την JSP σελίδα.
- Στοιχεία scripting εισάγονται στην servlet κλάση της JSP σελίδας.
- Εκφράσεις γλώσσας έκφρασης (Expression language) μεταφέρονται σαν παράμετροι στις κλήσεις στον αποτιμητή των JSP εκφράσεων.
- jsp: plugin στοιχεία μετατρέπονται σε κατάλληλες εντολές markup γλώσσας (όπως την HTML) για να ενεργοποιήσουν ένα applet.
- Μη καθορισμένα tags μετατρέπονται σε κλήσεις στον κατάλληλο χειριστή tags.

4.3.3 Enterprise JavaBeans (EJB)

Η τεχνολογία Enterprise JavaBeans αποτελεί μια αρχιτεκτονική συστατικών από την πλευρά του εξυπηρετητή που απλοποιεί τη διαδικασία της κατασκευής επιχειρησιακών κλάσεων διανεμημένων συστατικών εφαρμογών σε Java. Χρησιμοποιώντας EJB μπορούμε να γράψουμε κλιμακωτές, αξιόπιστες και ασφαλείς εφαρμογές χωρίς να χρειάζεται να γράψουμε το δικό μας περίπλοκο και κατανεμημένο πλαίσιο εργασίας. Η τεχνολογία EJB ουσιαστικά λοιπόν προσφέρει γρήγορη ανάπτυξη εφαρμογών για λειτουργία στην πλευρά του εξυπηρετητή.

Τα Enterprise Beans τρέχουν σε έναν EJB application server, ο οποίος αποτελεί ένα runtime περιβάλλον στον εξυπηρετητή εφαρμογών. Αν και είναι διαφανής ως προς τον προγραμματιστή της εφαρμογής, ο EJB container παρέχει υπηρεσίες σε επίπεδο συστήματος, όπως συναλλαγές και ασφάλεια στα enterprise beans που χειρίζεται. Αυτές οι υπηρεσίες μας δίνουν τη δυνατότητα να χτίσουμε και να κάνουμε deploy enterprise beans γρήγορα, που αποτελεί τον κορμό των συναλλακτικών J2EE εφαρμογών.

Η EJB τεχνολογία βασίζεται σε άλλες Java τεχνολογίες για να λειτουργήσει κατάλληλα. Καταρχήν χρησιμοποιεί το Java Remote Method Invocation γνωστό και ως RMI σαν το πρωτόκολλο επικοινωνίας ανάμεσα σε δύο enterprise beans και ανάμεσα σε ένα enterprise bean και σε έναν εξυπηρετούμενο. Μια άλλη τεχνολογία που χρησιμοποιείται στην EJB είναι η Remote Method Invocation over Internet Inter-ORB ή αλλιώς RMI-IIOP, όπου το ORB προέρχεται από το Object Request Broker καθώς και η Java Naming Directory Interface ή JNDI σαν την υπηρεσία ονομασίας που συνδέει ένα όνομα με ένα enterprise bean.

4.3.3.1 Τύποι Enterprise JavaBeans (EJBs) .

Τα EJBs αποτελούνται ουσιαστικά από τις εξής τρεις κατηγορίες :

- **Session beans.** Τα session beans μοντελοποιούν τις επιχειρηματικές διαδικασίες. Είναι σαν ρήματα, αφού κάνουν ενέργειες. Μια ενέργεια μπορεί να είναι οτιδήποτε, όπως η πρόσθεση αριθμών, πρόσβαση σε βάσεις δεδομένων ή η κλήση ενός enterprise bean. Αποτελούνται από δύο κατηγορίες:

- ο **Stateless Session Beans**

Ένα stateless session bean δεν διατηρεί μια συμβατική κατάσταση για τον εξυπηρετούμενο. Όταν ένας εξυπηρετούμενος καλεί την μέθοδο ενός stateless bean, οι μεταβλητές του στιγμιότυπου του bean μπορεί να διατηρούν μια κατάσταση, όμως μόνο κατά την διάρκεια της εκτέλεσης αυτής της μεθόδου. Όταν η μέθοδος τελειώσει, η κατάσταση δεν διατηρείται πια.

- ο **Stateful Session Beans**

Η κατάσταση ενός αντικειμένου αποτελείται από τις τιμές των μεταβλητών του στιγμιότυπού του. Σε ένα stateful session bean οι

μεταβλητές του στιγμιότυπού αντιπροσωπεύουν την κατάσταση μιας συνεδρίας εξυπηρετητή-bean. Επειδή ο πελάτης αλληλεπιδρά με το bean του, αυτή η κατάσταση καλείται συχνά και κατάσταση συνομιλίας.

- **Entity beans.** Τα entity beans μοντελοποιούν τα επιχειρηματικά δεδομένα. Είναι σαν αντικείμενα δεδομένων, που σημαίνει αντικείμενα Java που περιέχουν πληροφορίες από βάσεις δεδομένων. Παραδείγματα τέτοιων αποτελεί ένα προϊόν, μια παραγγελία, ένας εργαζόμενος ή μια πιστωτική κάρτα.
- **Message-driven beans.** Τα Message-driven beans είναι παρόμοια με τα session beans στο ότι αντιπροσωπεύουν ενέργειες. Η διαφορά είναι ότι μπορείς να καλέσεις message-driven beans μόνο στέλνοντας μηνύματα σε αυτά τα beans. Παραδείγματα από message-driven beans μπορεί να είναι beans που λαμβάνουν μηνύματα για συναλλαγές μετοχών, μηνύματα για εξουσιοδότηση πιστωτικών καρτών, ή μηνύματα φόρτου εργασίας. Τέλος, τα message-driven beans μπορούν να καλούν και άλλα enterprise beans.

4.3.3.1.1 Session Beans.

Ένα session Bean αντιπροσωπεύει έναν απλό εξυπηρετούμενο μέσα σε έναν Εξυπηρετητή Εφαρμογών. Για να έχουμε πρόσβαση σε μια εφαρμογή που έχει γίνει deployed σε έναν εξυπηρετητή, ο εξυπηρετούμενος καλεί τις μεθόδους του session bean. Το session bean εκτελεί δουλειές για τον εξυπηρετούμενο, προστατεύοντας τον εξυπηρετούμενο από την πολυπλοκότητα εκτελώντας επιχειρησιακές εργασίες μέσα στον εξυπηρετητή.

Όπως υπονοεί και το όνομά του, ένα session bean λειτουργεί παρόμοια με μια αλληλεπιδραστική συνεδρία. Ένα session bean δεν μοιράζεται ανάμεσα σε πολλούς εξυπηρετούμενους, όπως ακριβώς και μια αλληλεπιδραστική συνεδρία μπορεί να έχει μόνο έναν χρήστη. Επιπλέον ακριβώς όπως μια αλληλεπιδραστική συνεδρία και ένα session bean δεν είναι persistent, δηλαδή δεν αποθηκεύονται δεδομένα του σε μια βάση δεδομένων. Όταν ένας εξυπηρετούμενος τερματίσει, το session bean που του αντιστοιχεί τερματίζει και δεν συσχετίζεται πλέον με αυτόν τον εξυπηρετούμενο.

- **Stateless Session Beans**

Εκτός από ότι συμβαίνει κατά την κλήση μεθόδων, όλα τα στιγμιότυπα ενός stateless bean είναι ισοδύναμα, επιτρέποντας στον container να αναθέτει οποιοδήποτε στιγμιότυπο ενός stateless bean σε οποιοδήποτε πελάτη.

Επειδή τα stateless session beans μπορούν να υποστηρίξουν πολλαπλούς πελάτες, μπορούν να προσφέρουν καλύτερη κλιμάκωση για εφαρμογές που απαιτούν την

συμμετοχή μεγάλου αριθμού πελατών. Ουσιαστικά μια εφαρμογή απαιτεί περισσότερα stateless session bean απ' ότι stateful session beans για τον ίδιο αριθμό εξυπηρετούμενων.

Κατά περιόδους, ένας EJB container μπορεί να αποθηκεύει ένα stateful session bean σε δευτερεύουσα αποθηκευτική μονάδα. Αντίθετα τα stateless session beans δεν αποθηκεύονται ποτέ σε δευτερεύουσα αποθηκευτική μονάδα. Αυτό έχει ως συνέπεια τα stateless session beans να προσφέρουν καλύτερη απόδοση απ' ότι τα stateful session beans.

Ένα stateless session bean μπορεί να υλοποιήσει μια υπηρεσία ιστού, αλλά άλλοι τύποι enterprise beans δε μπορούν.

- **Stateful Session Beans**

Όπως είπαμε και πιο πάνω στα *Stateful Session Beans* η κατάσταση ενός αντικειμένου αποτελείται από τις τιμές των μεταβλητών του στιγμιότυπου του δηλαδή οι μεταβλητές του στιγμιότυπου αντιπροσωπεύουν την κατάσταση μιας συνεδρίας εξυπηρετητή-bean. Η κατάσταση διατηρείται κατά την διάρκεια της συνεδρίας εξυπηρετούμενου-bean. Αν ο εξυπηρετούμενος καταργήσει ή τερματίσει το bean, η συνεδρία τελειώνει και η κατάσταση χάνεται. Αυτή η εφήμερη φύση της κατάστασης δεν αποτελεί πρόβλημα, αφού όταν η αλληλεπίδραση μεταξύ εξυπηρετούμενου και bean τελειώσει δεν υπάρχει ανάγκη να διατηρηθεί η κατάσταση.

4.3.3.1.2 Entity Beans.

Ένα entity bean είναι ένα συστατικό δεδομένων, το οποίο διατηρεί τα δεδομένα μόνιμα σε μια δευτερεύουσα αποθηκευτική μονάδα, όπως μια βάση δεδομένων. Όπως άλλες Java κλάσεις ένα entity bean έχει πεδία και μεθόδους. Χρησιμοποιεί τα πεδία για να αποθηκεύει τα δεδομένα και μεθόδους για να εκτελέσει λειτουργίες.

Σε μια EJB εφαρμογή, ένα entity bean αποτελεί μια ματιά στις εγγραφές ενός πίνακα μιας βάσης δεδομένων. Αντί να διαχειριζόμαστε λοιπόν δεδομένα όπως θα κάναμε σε μια εφαρμογή που δεν θα ανήκει στην τεχνολογία EJB, τώρα διαχειριζόμαστε entity beans που αντιπροσωπεύουν τα δεδομένα. Ο EJB container αναλαμβάνει έπειτα τον συγχρονισμό και άλλες εργασίες για τη διατήρηση των δεδομένων. Ένα entity bean ζει όσο και τα δεδομένα που αναπαριστά και γι' αυτό ένα entity bean μπορεί να υπάρχει για βδομάδες, μήνες ή και χρόνια.

Τα entity beans προσφέρουν αρκετά πλεονεκτήματα σε σχέση με την άμεση πρόσβαση στην βάση δεδομένων. Αρχικά λοιπόν μπορούμε να αναφέρουμε ότι είναι ευκολότερος ο χειρισμός των δεδομένων σαν αντικείμενα παρά σαν σκέτα δεδομένα και δεύτερον ότι ο container προσφέρει υπηρεσίες που κάνουν πιο εύκολο τον χειρισμό των δεδομένων που αντιπροσωπεύονται από entity beans.

Υπάρχουν δύο τύποι entity beans: entity beans με διαχείριση διατήρησης δεδομένων από το bean (bean manage persistence BMP) και entity beans με διατήρηση δεδομένων από τον container (container manage persistence CMP).

4.3.3.1.3 Message-Driven Beans.

Ένα message-driven bean είναι ένα enterprise bean που επιτρέπει σε J2EE εφαρμογές να επεξεργάζονται μηνύματα ασύγχρονα. Βασίζονται στο Java Messaging Service (JMS), ένα API της J2EE πλατφόρμας που υποστηρίζει την ανταλλαγή μηνυμάτων. Φυσιολογικά δρα σαν ένας JMS ακροατής μηνυμάτων, ο οποίος είναι παρόμοιος με τον ακροατή γεγονότων εκτός του ότι δέχεται JMS μηνύματα αντί για γεγονότα. Τα μηνύματα μπορούν να σταλούν από ένα συστατικό ιστού ή από ένα σύστημα που δεν χρησιμοποιεί J2EE τεχνολογία. Τα message-driven beans μπορούν να επεξεργαστούν είτε JMS μηνύματα είτε άλλων ειδών μηνύματα.

Για κάποιον εξυπηρετούμενο, ένα message-driven bean είναι ένας ασύγχρονος καταναλωτής μηνυμάτων. Η συμπεριφορά ενός message-driven bean μοιάζει με αυτή ενός stateless session bean στο ότι και το message-driven bean δεν έχει κατάσταση συνομιλίας. Κάθε message-driven bean δημιουργείται και ελέγχεται από έναν EJB container. Αντίθετα με τα session και entity beans τα message-driven bean δεν έχουν home ή remote interface. Οι εφαρμογές του εξυπηρετητή προσπελαίνουν τα message-driven beans στέλνοντας μηνύματα μέσω JMS σε ένα JMS προορισμό.

Η κλάση ενός message-driven bean πρέπει να είναι μια δημόσια κλάση και πρέπει να υλοποιεί τα ακόλουθα δύο interfaces:

- javax.ejb.MessageDrivenBean
- javax.jms.MessageListener

Πέρα από αυτά τα δύο interfaces, ο προγραμματιστής πρέπει να γνωρίζει για το interface MessageDrivenContext το οποίο παρέχεται από το EJB container για κάθε στιγμιότυπο message-driven bean.

Από πολλές απόψεις ένα message-driven bean μοιάζει με ένα stateless session bean.

- Τα στιγμιότυπα ενός message-driven bean δε διατηρούν καθόλου δεδομένα και καμία διαλογική κατάσταση για συγκεκριμένο πελάτη.
- Όλα τα στιγμιότυπα ενός message-driven bean είναι ισοδύναμα, επιτρέποντας στο EJB container να αναθέσει ένα μήνυμα σε οποιοδήποτε στιγμιότυπο ενός message-driven bean. Ο container μπορεί να διατηρεί pool από αυτά τα στιγμιότυπα για να επιτρέψει σε streams από μηνύματα να επεξεργάζονται ταυτόχρονα.
- Ένα απλό message-driven bean μπορεί να επεξεργαστεί μηνύματα από πολλούς πελάτες.

Τα message-driven beans έχουν τα παρακάτω χαρακτηριστικά:

- Εκτελούνται μετά από την παραλαβή ενός απλού μηνύματος ενός εξυπηρετούμενου.
- Καλούνται ασύγχρονα.

- Ζουν σχετικά λίγο.
- Δε διατηρούν καταστάσεις
- Μπορούν να είναι ενήμερα συναλλαγών.

4.3.3.2 Πρόσβαση Εξυπηρετούμενων με Interfaces.

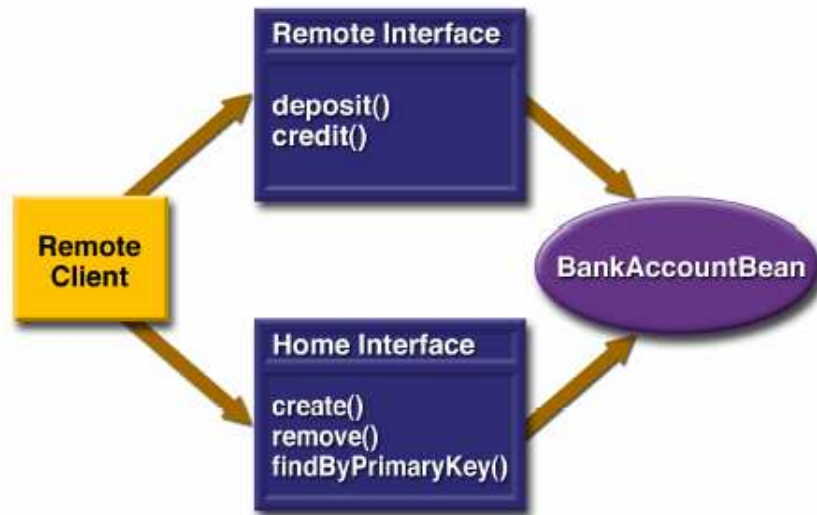
Ένας εξυπηρετούμενος μπορεί να έχει πρόσβαση σε ένα session ή σε ένα entity bean μόνο μέσω των μεθόδων που καθορίζονται στα interface του bean. Αυτά τα interfaces καθορίζουν ουσιαστικά την εικόνα που έχει ο εξυπηρετούμενος για το bean. Όλες οι άλλες πλευρές του bean, όπως υλοποίηση μεθόδων, ρυθμίσεις περιγραφικών διάταξης, αφηρημένα σχήματα, και κλήσεις βάσεων δεδομένων, είναι κρυμμένα από τον εξυπηρετούμενο.

- **Απομακρυσμένοι Εξυπηρετούμενοι (Remote Clients)**

Ένας απομακρυσμένος εξυπηρετούμενος (remote client) ενός enterprise bean έχει τα ακόλουθα χαρακτηριστικά :

- Μπορεί να τρέξει σε διαφορετικά μηχανήματα και σε διαφορετικές Java εικονικές μηχανές από το enterprise bean στο οποίο έχει πρόσβαση.
- Μπορεί να είναι ένα συστατικό ιστού, μια εφαρμογή εξυπηρετητή ή ένα άλλο enterprise bean.
- Για έναν απομακρυσμένο εξυπηρετητή, η τοποθεσία του enterprise bean είναι ξεκάθαρη.

Για να δημιουργήσουμε ένα enterprise bean, το οποίο θα έχει απομακρυσμένη πρόσβαση, πρέπει να δημιουργήσουμε ένα remote interface και ένα home interface. Το remote interface καθορίζει τις επιχειρησιακές μεθόδους που υπάρχουν σε αυτό το bean. Για παράδειγμα το remote interface ενός bean που ονομάζεται BankAccountBean μπορεί να έχει επιχειρησιακές μεθόδους που να ονομάζονται deposit και credit. Το home interface καθορίζει τις μεθόδους του κύκλου ζωής του bean, δηλαδή τις create και remove. Για τα entity beans, το home interface καθορίζει επιπλέον τις μεθόδους finder και home. Οι μέθοδοι finder είναι αυτές που χρησιμοποιούνται για να εντοπίζουν entity beans. Οι home μέθοδοι είναι επιχειρησιακές μέθοδοι που καλούνται από όλα τα στιγμιότυπα της κλάσης ενός entity bean. Το Σχήμα 4.6 απεικονίζει πως τα interfaces ελέγχουν την εικόνα του πελάτη για ένα enterprise bean.



Σχήμα 4.6 Τα interfaces για ένα Enterprise Bean με απομακρυσμένη πρόσβαση.

- **Τοπικοί Εξυπηρετούμενοι (Local Clients)**

Ένας τοπικός εξυπηρετούμενος (local client) έχει τα ακόλουθα χαρακτηριστικά:

- Πρέπει να τρέχει στην ίδια εικονική μηχανή Java με το enterprise bean στο οποίο έχει πρόσβαση.
- Μπορεί να είναι συστατικό ιστού ή ένα άλλο enterprise bean.
- Για τον τοπικό εξυπηρετούμενο, η τοποθεσία του enterprise bean στο οποίο έχει πρόσβαση δεν είναι προφανής.
- Είναι συχνό ένα entity bean να έχει σχέση διαχειριζόμενη από τον container με ένα άλλο entity bean.

Για αν δημιουργήσουμε ένα enterprise bean που να επιτρέπει την τοπική πρόσβαση, πρέπει να δημιουργήσουμε ένα τοπικό interface. Το local interface καθορίζει τις επιχειρησιακές μεθόδους του bean, και το local home interface καθορίζει τις μεθόδους του κύκλου ζωής και τις finder μεθόδους του.

4.3.3.3 Συστατικά των Enterprise JavaBeans (EJBs) .

Για να δημιουργήσουμε ένα enterprise bean, πρέπει να δημιουργήσουμε τα παρακάτω αρχεία.

- **Περιγραφέας διάταξης (Deployment Descriptor):** Ένα XML αρχείο που καθορίζει τις πληροφορίες για το bean, όπως τον τύπο διατήρησης δεδομένων του (CMP ή BMP) και τα χαρακτηριστικά συναλλαγών του.
- **Enterprise bean κλάση:** Υλοποιεί τις μεθόδους που αναφέρονται στα παρακάτω interfaces

- **Interfaces:** Το απομακρυσμένο και το home interface χρειάζονται για απομακρυσμένη πρόσβαση. Για τοπική πρόσβαση, το τοπικό και το τοπικό home απαιτούνται. Για πρόσβαση από εξυπηρετούμενους υπηρεσιών ιστού, το interface endpoint υπηρεσίας ιστού είναι απαραίτητο.
- **Βοηθητικές κλάσεις:** Άλλες κλάσεις που χρειάζονται από την κλάση των enterprise beans, όπως κλάσεις εξαιρέσεων και γενικής χρήσης.

Πακετάρουμε τα αρχεία της προηγούμενης λίστας σε ένα EJB JAR αρχείο, το οποίο είναι μεταφέρσιμο και μπορεί να χρησιμοποιηθεί για διαφορετικές εφαρμογές. Για να συναρμολογήσει κάποιος μια J2EE εφαρμογή, πακετάρουμε μία ή περισσότερες υπομονάδες, όπως τα EJB JAR αρχεία σε ένα EAR αρχείο, το οποίο είναι το αρχείο που τελικά περιέχει την εφαρμογή. Όταν γίνει deploy το EAR αρχείο που περιέχει το EJB JAR αρχείο, ταυτόχρονα κάνουμε deploy και το enterprise bean στον εξυπηρετητή εφαρμογών. Μπορούμε επίσης να κάνουμε deploy ένα EJB JAR το οποίο δεν περιέχεται σε ένα EAR αρχείο. Η συσκευασία αποτελεί το αντικείμενο επόμενης παραγράφου.

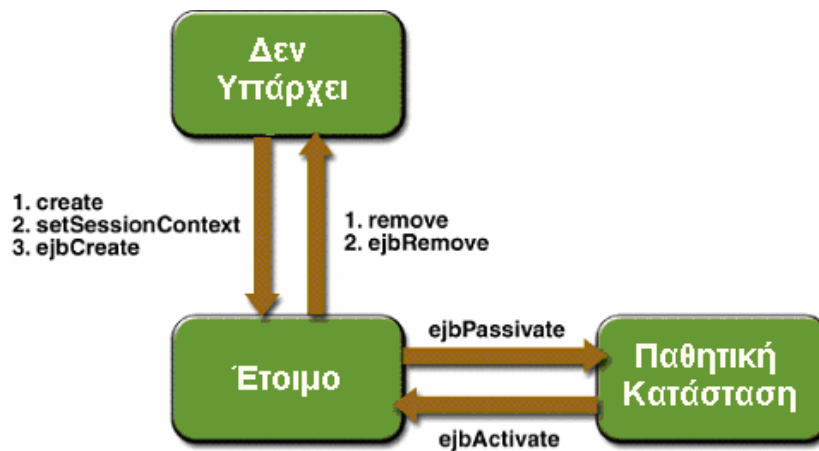
4.3.3.4 Κύκλοι Ζωής των Enterprise JavaBeans (EJBs) .

- **Stateful Enterprise Beans**

Ο εξυπηρετούμενος ξεκινάει τον κύκλο ζωής καλώντας την μέθοδο create. Το EJB container δημιουργεί ένα στιγμιότυπο του bean και μετά καλεί τις μεθόδους setSessionContext και ejbCreate στο session bean. Έπειτα το bean είναι έτοιμο ώστε να καλεστεί η business μεθόδός του από τον εξυπηρετούμενο.

Ενώ βρίσκεται σε κατάσταση αναμονής, ο EJB container μπορεί να αποφασίσει να απενεργοποιήσει το bean μεταφέροντάς το από τη μνήμη σε δευτερεύουσα μονάδα αποθήκευσης. Ο EJB container καλεί την μέθοδο ejbPassivate του bean ακριβώς πριν το απενεργοποιήσει. Αν ο εξυπηρετούμενος καλέσει μια επιχειρησιακή μέθοδο από το bean, καθώς αυτό είναι σε απενεργοποιημένη κατάσταση, ο EJB container ενεργοποιεί το bean, καλεί την μέθοδο ejbActivate και μετά το θέτει στην κατάσταση ετοιμότητας.

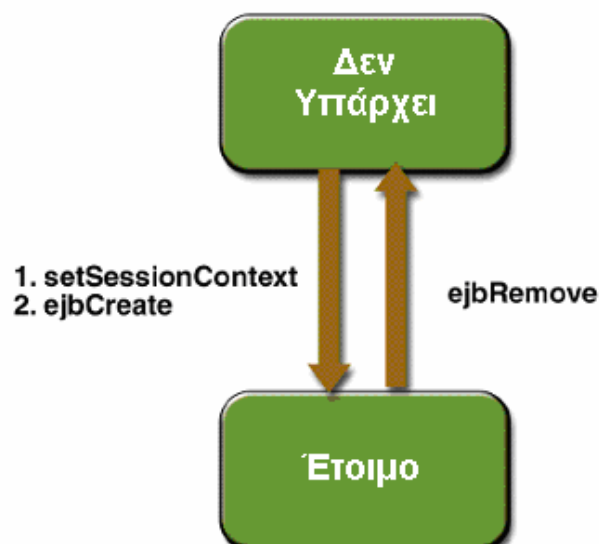
Στο τέλος του κύκλου ζωής, ο εξυπηρετούμενος καλεί την μέθοδο remove, και ο EJB container καλεί την ejbRemove μέθοδο του bean. Ουσιαστικά ο κώδικας που γράφουμε ελέγχει την κλήση μόνο δύο μεθόδων του κύκλου ζωής ενός bean: την create και την remove μέθοδο του εξυπηρετούμενου. Η μέθοδος ejbCreate για παράδειγμα, είναι μέσα στην κλάση του bean και μας επιτρέπει να κάνουμε κάποιες λειτουργίες ακριβώς μετά τη δημιουργία ενός στιγμιότυπου bean. Στο σχήμα 4.7 φαίνονται τα στάδια ενός stateful session bean.



Σχήμα 4.7 α .Ο κύκλος ζωής ενός Stateful Session Bean

- ***Stateless Enterprise Beans***

Επειδή ένα stateless session bean δεν απενεργοποιείται ποτέ, ο κύκλος ζωής του έχει μόνο δύο καταστάσεις: την μη ύπαρξη και την κατάσταση έτοιμο για κλήση των επιχειρηματικών του μεθόδων. Στο σχήμα 4.8 φαίνονται τα στάδια ενός stateless session bean.



Σχήμα 4.8 β .Ο κύκλος ζωής ενός Stateless Session Bean

- **Entity Beans**

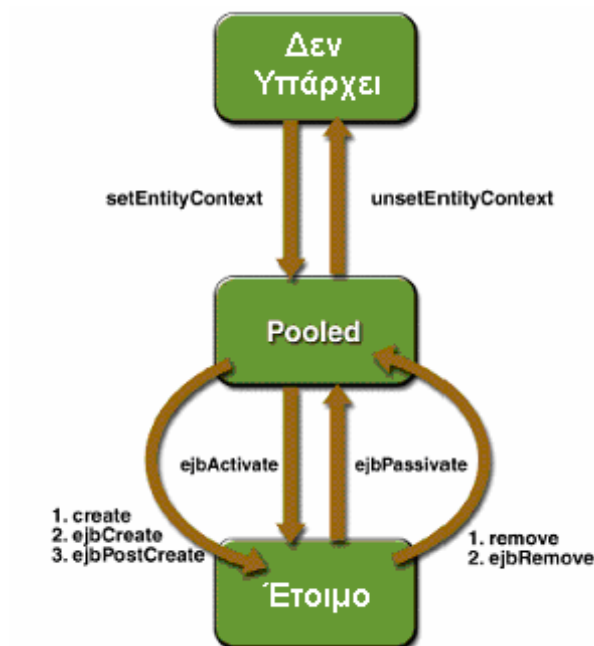
Αφού ο EJB container δημιουργήσει ένα στιγμιότυπο, καλεί την μέθοδο `setEntityContext` της κλάσης του entity bean. Η μέθοδος `setEntityContext` περνάει τις παραμέτρους περιβάλλοντος στο bean. Αφού δημιουργηθεί το στιγμιότυπο, το entity bean πηγαίνει στο pool με τα διαθέσιμα στιγμιότυπα. Όταν βρίσκεται στην κατάσταση που είναι στο pool, το στιγμιότυπο δε συσχετίζεται με καμία συγκεκριμένη ταυτότητα αντικείμενου. Όλα τα στιγμιότυπα στο pool είναι πανομοιότυπα. Ο EJB container αναθέτει μια ταυτότητα, όταν το μετακινεί στην κατάσταση ετοιμότητας.

Υπάρχουν δύο μονοπάτια από την κατάσταση pooled προς την κατάσταση ετοιμότητας. Στο πρώτο μονοπάτι ο εξυπηρετούμενος καλεί την `create` μέθοδο, προκαλώντας την κλήση της `ejbCreate` και της `ejbPostCreate` μεθόδου από τον EJB container. Στο δεύτερο μονοπάτι ο EJB container καλεί την `ejbActivate` μέθοδο. Ενώ, όταν ένα entity bean είναι στην κατάσταση ετοιμότητας, οι επιχειρησιακές του μέθοδοι μπορούν να κληθούν.

Υπάρχουν επιπλέον δύο μονοπάτια από την κατάσταση ετοιμότητας προς την pooled κατάσταση. Πρώτον, ο εξυπηρετούμενος μπορεί να καλέσει την `remove` μέθοδο, η οποία προκαλεί την κλήση της `ejbRemove` μεθόδου από τον EJB container. Δεύτερον, ο EJB container μπορεί να καλέσει την `ejbPassivate` μέθοδο.

Στο τέλος του κύκλου ζωής, ο EJB container αφαιρεί τα στιγμιότυπα από το pool και καλεί την `unsetEntityContext` μέθοδο.

Στην κατάσταση pooled, ένα στιγμιότυπο δε συσχετίζεται με κανένα συγκεκριμένο EJB αντικείμενο. Για BMP entity beans, όταν ο EJB container μεταφέρει ένα στιγμιότυπο από την pooled κατάσταση στην κατάσταση ετοιμότητας, δε θέτει αυτόματα το πρωτεύον κλειδί. Για αυτό το λόγο οι μέθοδοι `ejbCreate` και `ejbActivate` πρέπει να αναθέσουν μια αξία στο πρωτεύον κλειδί. Αν το πρωτεύον κλειδί είναι λάθος, τότε οι μέθοδοι `ejbLoad` και `ejbStore` δεν μπορούν να συγχρονίσουν τις μεταβλητές των στιγμιότυπων με την βάση δεδομένων. Στο σχήμα 4.9 φαίνονται τα στάδια ενός Entity Bean.



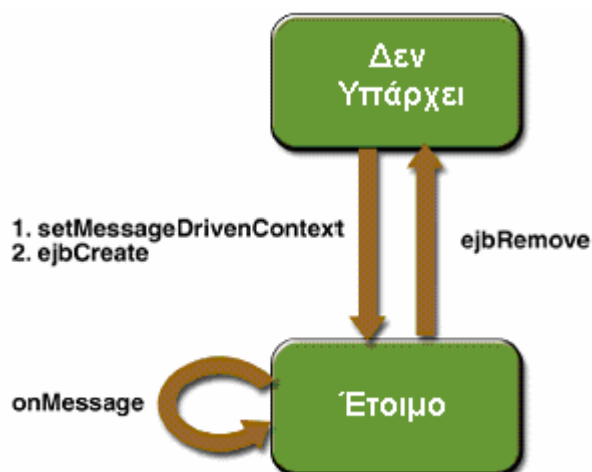
Σχήμα 4.9 γ .Ο κύκλος ζωής ενός Entity Bean

- **Message-Driven Beans**

Ο EJB container συνήθως δημιουργεί ένα pool από στιγμιότυπα message-driven beans. Για κάθε στιγμιότυπο, ο EJB container δημιουργεί στιγμιότυπα και κάνει τα ακόλουθα:

1. Καλεί τη μέθοδο setMessageDrivenContext για να περάσει το αντικείμενο περιεχομένου στο στιγμιότυπο.
2. Καλεί τη μέθοδο ejbCreate του στιγμιότυπου.

Όπως και τα stateless session beans, τα message-driven beans ποτέ δεν απενεργοποιούνται και έχουν μόνο δύο καταστάσεις: μη ύπαρξη και ετοιμότητα για λήψη μηνυμάτων. Στο τέλος του κύκλου ζωής, ο container καλεί την μέθοδο ejbRemove. Στο σχήμα 4.10 φαίνονται τα στάδια ενός Message-Driven Bean.



Σχήμα 4.10 δ .Ο κύκλος ζωής ενός Message-Driven Bean

4.3.4 Συσκευασία (Packaging)

Μια εφαρμογή J2EE παραδίδεται ουσιαστικά με τη μορφή ενός αρχείου EAR (Enterprise Archive), το οποίο είναι ουσιαστικά ένα JAR (Java Archive) αρχείο με κατάληξη .ear. Χρησιμοποιώντας EAR αρχεία και υπομονάδες, μπορούμε να συναρμολογήσουμε ένα πλήθος από διαφορετικές J2EE εφαρμογές χρησιμοποιώντας εν μέρη τα ίδια συστατικά. Ένα EAR αρχείο περιέχει J2EE υπομονάδες και περιγραφείς ανάπτυξης. Ένας περιγραφέας διάταξης είναι ένα έγγραφο XML με μια .xml κατάληξη που περιγράφει τις ρυθμίσεις της διάταξης μιας εφαρμογής, μιας υπομονάδας, ή ενός συστατικού.

Υπάρχουν δύο τύποι περιγραφέων διάταξης: φορητοί και runtime. Ένας φορητός περιγραφέας διάταξης καθορίζεται από την J2EE προδιαγραφή και μπορεί να χρησιμοποιηθεί για να καθορίσει ρυθμίσεις της ανάπτυξης σε οποιοδήποτε J2EE συμβατή υλοποίηση. Ένας runtime περιγραφέας διάταξης χρησιμοποιείται για να καθορίσει J2EE εξειδικευμένες στην υλοποίηση παραμέτρους. Για παράδειγμα, ο runtime περιγραφέας διάταξης του Sun Java System Application Server Platform Edition 8 περιέχει πληροφορίες, όπως την αντιστοίχιση μεταξύ φορητών ονομάτων των πόρων μιας εφαρμογής και των πόρων ενός εξυπηρετητή. Οι runtime περιγραφείς διάταξης ενός εξυπηρετητή εφαρμογών ιστού ονομάζονται sun-moduleType.xml και βρίσκεται στον ίδιο κατάλογο με τον φορητό περιγραφέα διάταξης.

Μια J2EE υπομονάδα αποτελείται από ένα ή περισσότερα J2EE συστατικά για τον ίδιο τύπο container, καθώς και έναν περιγραφέα διάταξης αυτού του τύπου. Ο περιγραφέας διάταξης ενός enterprise bean, για παράδειγμα, δηλώνει τα χαρακτηριστικά συναλλαγών και εξουσιοδοτήσεις ασφάλειας. Μια J2EE υπομονάδα χωρίς έναν περιγραφέα διάταξης εφαρμογής μπορεί να διαταχθεί σαν stand-alone υπομονάδα.

Οι τέσσερις τύποι J2EE υπομονάδων είναι οι παρακάτω:

1. EJB υπομονάδες, οι οποίες περιέχουν αρχεία κλάσεων για τα Enterprise Beans και έναν EJB περιγραφέα διάταξης. Οι EJB υπομονάδες πακετάρονται σαν JAR φάκελοι με .jar κατάληξη.
2. Υπομονάδες ιστού, οι οποίες περιέχουν αρχεία κλάσεων για servlets, JSP αρχεία, GIF και HTML αρχεία, και έναν περιγραφέα διάταξης ιστού. Οι υπομονάδες ιστού, πακετάρονται σαν JAR αρχεία με .war κατάληξη.
3. Υπομονάδες εφαρμογών εξυπηρετούμενων, οι οποίες περιέχουν αρχεία κλάσεων και περιγραφείς διάταξης εφαρμογών εξυπηρετούμενων. Οι υπομονάδες εφαρμογών εξυπηρετούμενων πακετάρονται σαν JAR αρχεία με κατάληξη .jar .
4. Υπομονάδες προσαρμογέων πόρων, οι οποίες περιέχουν όλες τις Java διασυνδέσεις, κλάσεις, τοπικές βιβλιοθήκες, καθώς και άλλα έγγραφα τεκμηρίωσης, μαζί με περιγραφείς διάταξης προσαρμογέων πόρων. Μαζί όλα τα παραπάνω υλοποιούν την αρχιτεκτονική του συνδέσμου (Connector Architecture) για κάποιο συγκεκριμένο EIS. Οι υπομονάδες προσαρμογέων πόρων πακετάρονται σαν JAR αρχεία με κατάληξη .rar (resource adapter archive).

5

5 Υλοποίηση και Ρύθμιση Παραμέτρων

Στο κεφάλαιο αυτό περιγράφεται ο τρόπος δημιουργίας της εφαρμογής με χρήση της τεχνολογίας Enterprise JavaBean (EJB) χρησιμοποιώντας το προγραμματιστικό εργαλείο JBossIDE Eclipse 1.6. Επίσης περιγράφεται και η δημιουργία των σχημάτων και των πινάκων στη βάση δεδομένων MySQL και τέλος ορισμένες ρυθμίσεις στον JBoss-4.2.0.GA.

5.1 Υλοποίηση Εφαρμογής με JBossIDE Eclipse1.6.

Το εργαλείο JBossIDE Eclipse 1.6 μας προσφέρει πολλές δυνατότητες όσο αφορά τη δημιουργία EJB μερικές από αυτές είναι :

- Υποστήριξη για δημιουργία XDoclet.
- Debugging και παρακολούθηση του JBoss server και έλεγχο του κύκλου ζωής του.
- Εύκολο τρόπο διαμόρφωσης του τρόπου συσκευασίας των πακέτων.
- Εύκολο τρόπο για διάθεση ή όχι στον server των πακέτων (deploy/undeploy of packages)
- Ποικίλες εντολές δημιουργίας συστατικών J2EE που διευκολύνουν και απλοποιούν την ανάπτυξη J2EE εφαρμογών.
- Δυνατότητα δημιουργίας διάφορων άλλων συστατικών όπως JSP, HTML, και XML.

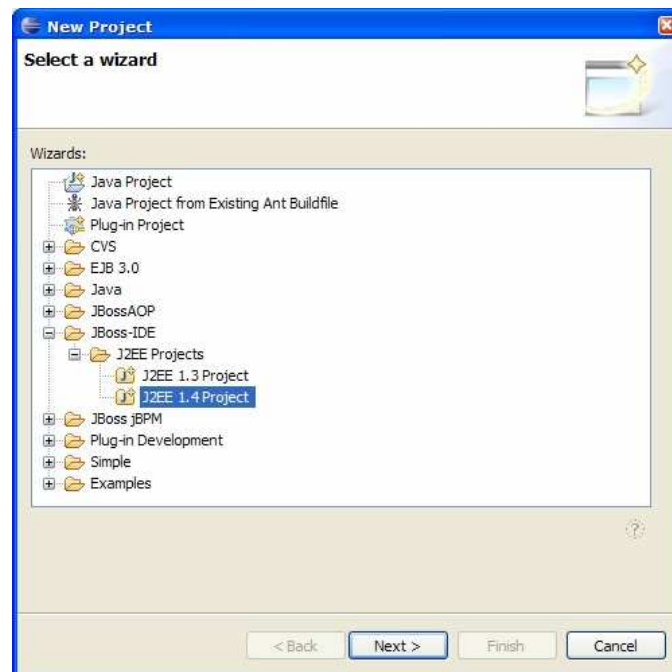
Για τη δημιουργία του project είναι απαραίτητα τα ακόλουθα :

- ✓ Java Development Kit 1.3.0 ή νεότερο (ένα JDK χρειάζεται για την έναρξη του JBoss 3.x, στην περίπτωση αυτή χρησιμοποιήσαμε το jdk1.5.0_09).
- ✓ Έναν JBoss Application Server 4.x (στην περίπτωση αυτή τον JBoss-4.2.0.GA).

5.1.1 Δημιουργία EJB και Μεθόδων.

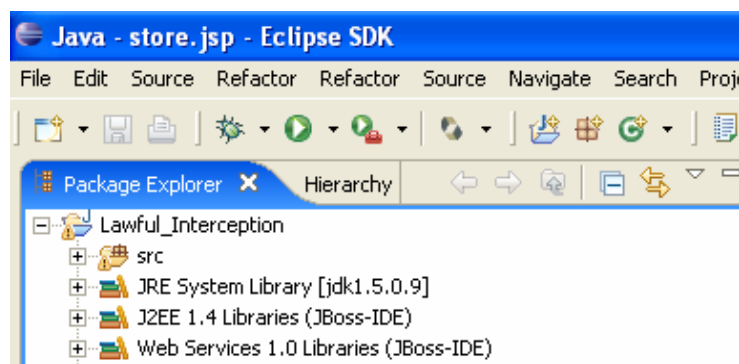
Θα δημιουργήσουμε ένα κατάλογο κώδικα (source folder), θα εισάγουμε τις απαραίτητες βιβλιοθήκες και θα καθορίσουμε το κατάλληλο μονοπάτι δημιουργίας (build path).

Δημιουργούμε ένα νέο J2EE 1.4 Project. Επιλέξτε File > New > Project... και επιλέξτε JBoss-IDE > J2EE Projects > J2EE 1.4 Project. Πληκτρολογήστε Lawful_Interception για το όνομα του project και επιβεβαιώστε ότι ο προεπιλεγμένος φάκελος εξόδου (default output folder) είναι ο bin..



Σχήμα 5.1 Επιλογή output folder

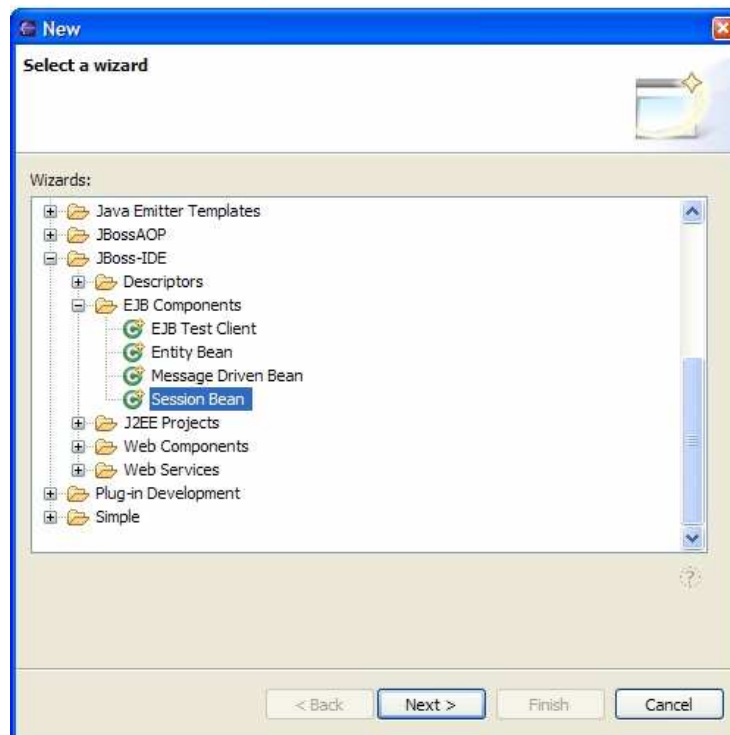
Έτσι από το package explorer, το νέο project θα πρέπει να δείχνει κάπως έτσι. Ας σημειώσουμε ότι οι J2EE 1.4 core classes έχουν άμεσα προστεθεί. Είναι διαθέσιμες σαν προκαθορισμένη βιβλιοθήκη με τον κώδικα πηγής (source code).



Σχήμα 5.2 Package Explorer tree structure

Το επόμενο βήμα θα είναι η δημιουργία ενός EJB, στην περίπτωση μας πρόκειται για ένα Stateless Session EJB.

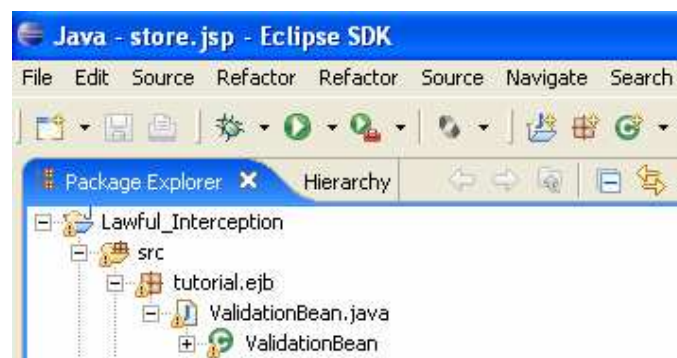
Δημιουργία ενός νέου Session EJB. Επιλέξτε File > New > Other... και επιλέξτε JBoss-IDE > EJB Components > Session Bean.



Σχήμα 5.3 Δημιουργία Session EJB

Το «package» θα είναι tutorial.ejb και το όνομα της κλάσης “ValidationBean”.

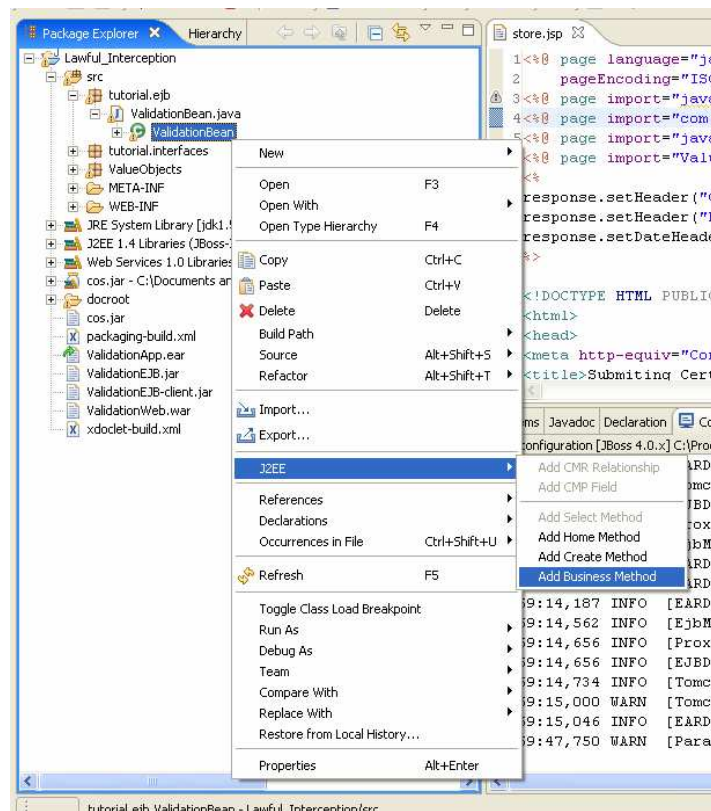
Αφήστε τις προεπιλεγμένες επιλογές και επιβεβαιώστε ότι η ejbCreate() method είναι επιλεγμένη. Αφού πατήσετε «Finish» από το package explorer, το νέο project θα πρέπει να δείχνει κάπως έτσι.



Σχήμα 5.4 Project tree structure after EJB creation

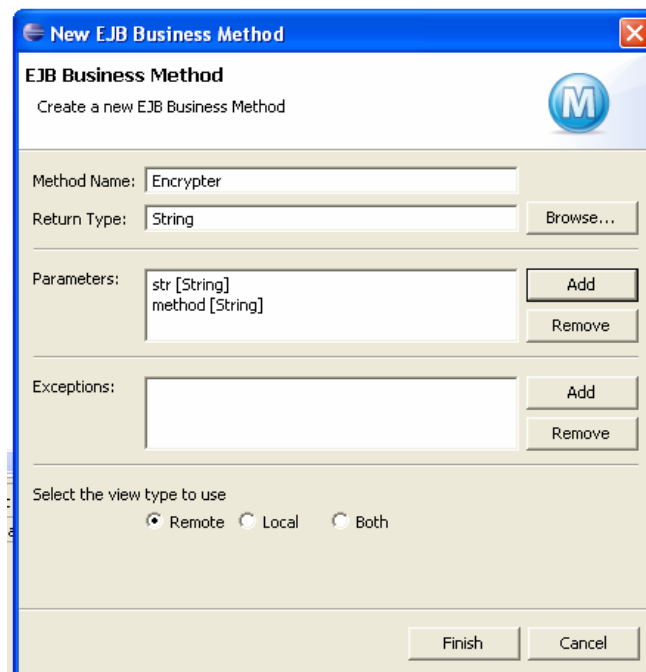
Στη συνέχεια θα δημιουργήσουμε τις επιχειρησιακές μεθόδους (business methods) που χρειάζονται για το την εφαρμογή μας.

Δεξί κλικ στη ValidationBean class, «κάτω» από το ValidationBean Java αρχείο. Θα πρέπει να βλέπετε ένα J2EE μενού. Επιλέξτε J2EE > Add Business Method.



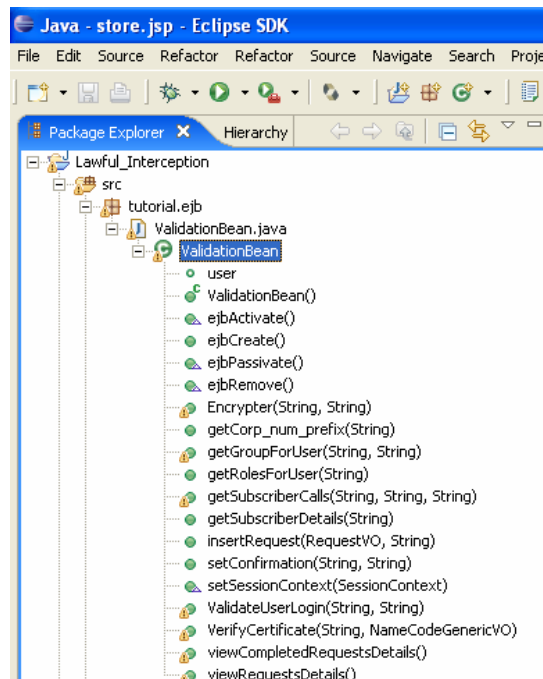
Σχήμα 5.5 Add Bussiness Method to EJB (a)

Μία από τις μεθόδους της εφαρμογής είναι η Encrypter που «επιστρέφει» String και δέχεται ως παραμέτρους το str (που είναι String) method (επίσης String).



Σχήμα 5.6 Add Bussiness Method to EJB (b)

Ακολουθώντας ακριβώς την ίδια μεθοδολογία και για τις υπόλοιπες μεθόδους της εφαρμογής από το package explorer, το νέο project θα πρέπει να δείχνει κάπως έτσι.

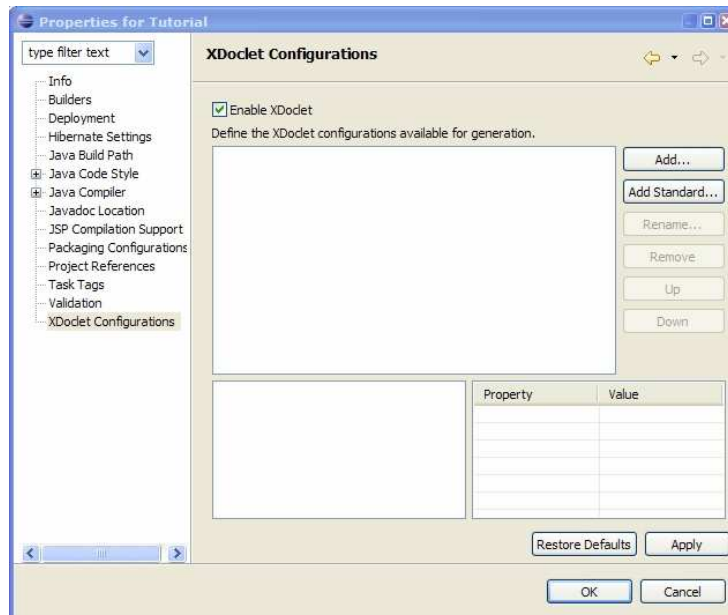


Σχήμα 5.7 Project Tree Structure after EJB methods have been created

5.1.2 Δημιουργία των Αρχείων που σχετίζονται με το EJB.

Βήμα 1.Ενεργοποίηση XDoclet (Enable XDoclet).

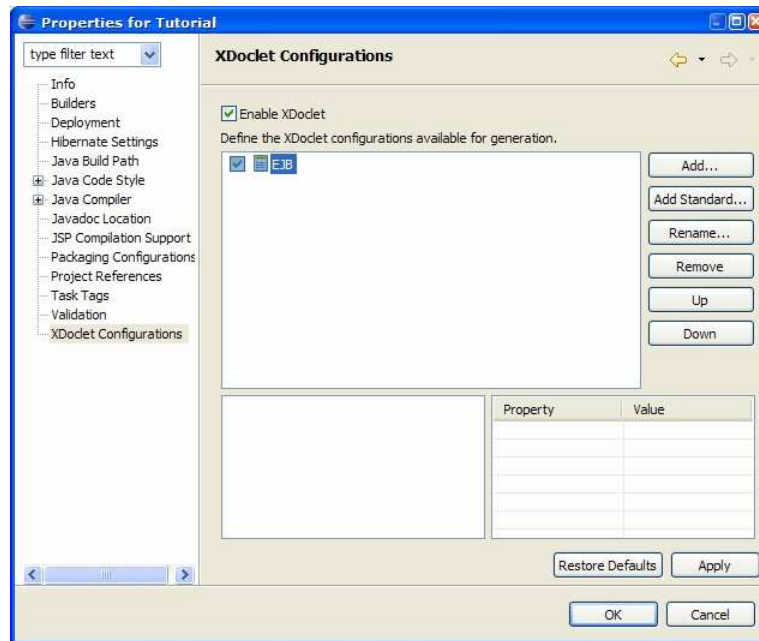
1. Δεξί κλικ στο project and επιλέξτε Properties.
2. Στη σελίδα των properties , επιλέξτε XDoclet configurations.
3. Στην αρχή της σελίδας υπάρχει ένα κουτί επιλογής (check-box) με ιδιότητα Enable XDoclet. Επιλέξτε το.



Σχήμα 5.8 XDoclet Configurations (a)

Βήμα 2. XDoclet EJB Δημιουργία και ρύθμιση παραμέτρων.

- Δεξί κλικ στην πάνω περιοχή ώστε ν εμφανιστεί το pop-up μενού και επιλέξτε Add. Πληκτρολογήστε EJB στο dialog box και κάντε κλικ στο OK.

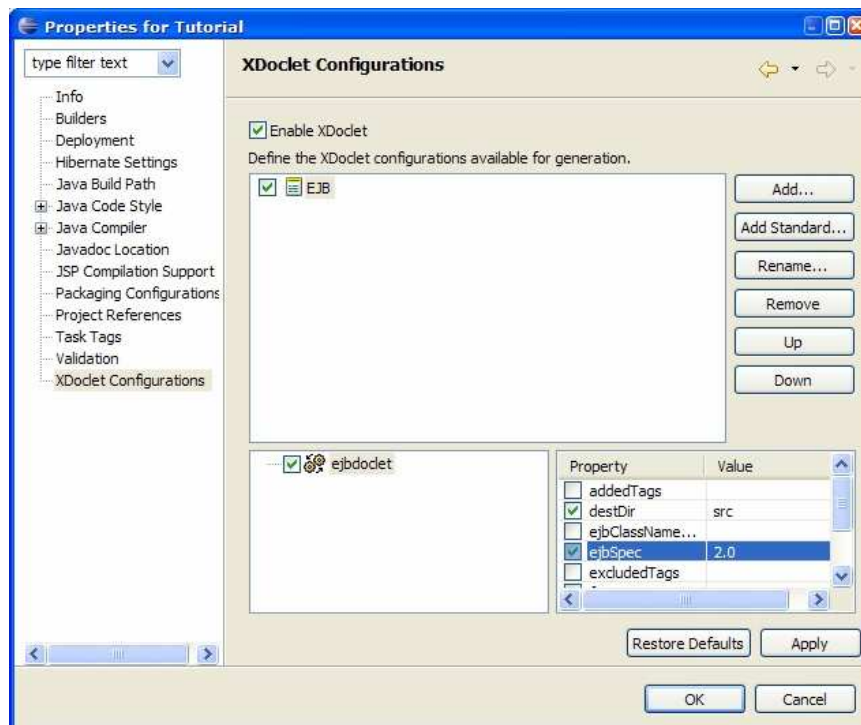


Σχήμα 5.9 XDoclet Configurations (b)

Βήμα 3. Ejbdoclet Παραμετροποίηση (Ejbdoclet Configuration).

1. Επιλέξτε EJB configuration.
2. Στην κάτω-αριστερά περιοχή ,δεξί κλικ ώστε να εμφανιστεί το popur μενού και επιλέξτε Add Doclet.
3. Μία λίστα από διαθέσιμα doclets θα εμφανιστεί. Επιλέξτε ejbdoclet και κάντε κλικ στο OK.
4. Στην κάτω-δεξιά περιοχή, θα δείτε τις ιδιότητες του ejbdoclet.
 - a. θέστε την destDir ιδιότητα σε src.
 - b. θέστε την ejbSpec ιδιότητα σε 2.0.

Η παραμετροποίηση μας τώρα περιέχει ένα ejbdoclet που θα παράγει αρχεία στον src κατάλογο και για τις EJB 2.0 προδιαγραφές.

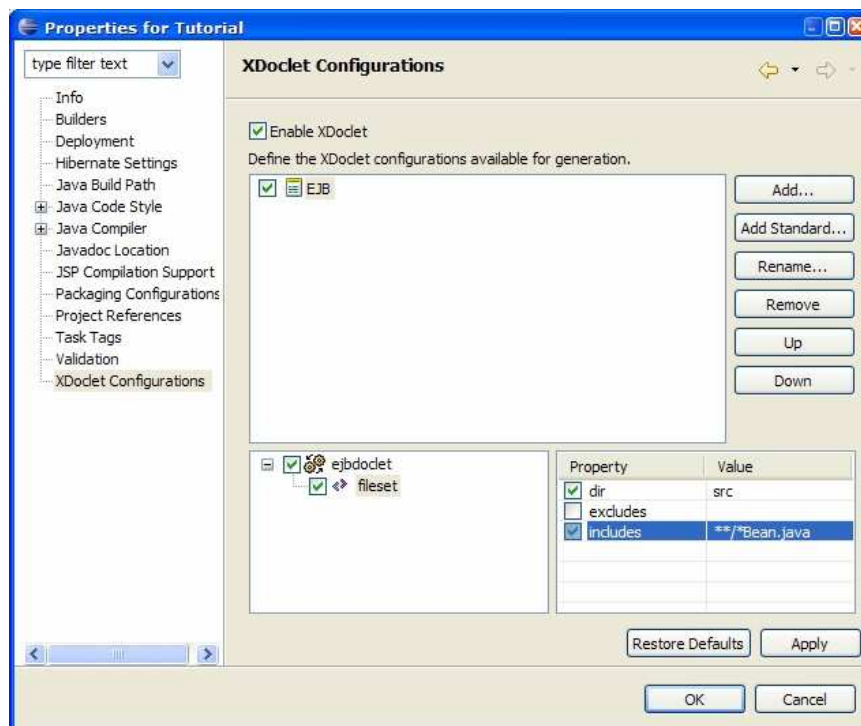


Σχήμα 5.10 XDoclet Configurations (c) – Ejbdoclet Configuration

Βήμα 4. Fileset Παραμετροποίηση (Fileset Configuration).

1. Στην κάτω-αριστερή περιοχή, κάντε δεξί κλικ στο ejbdoclet ώστε να εμφανιστεί το popup μενού και επιλέξτε Add.
2. Μία λίστα από διαθέσιμες υποεργασίες(subtasks) θα εμφανιστεί. Επιλέξτε fileset και κάντε κλικ στο Ok.
3. Στην κάτω-δεξιά περιοχή, βλέπετε τις ιδιότητες του fileset.
 - a. θέστε την dir ιδιότητα σε src.
 - b. αφαιρέστε την επιλογή excludes
 - c. θέστε την includes ιδιότητα σε `**/*Bean.java`.

Η παραμετροποίηση μας τώρα περιέχει ένα ejbdoclet με ένα fileset που περιέχει την src διεύθυνση, και όλα τα αρχεία «κάτω» από αυτήν που τελειώνουν σε Bean.java.

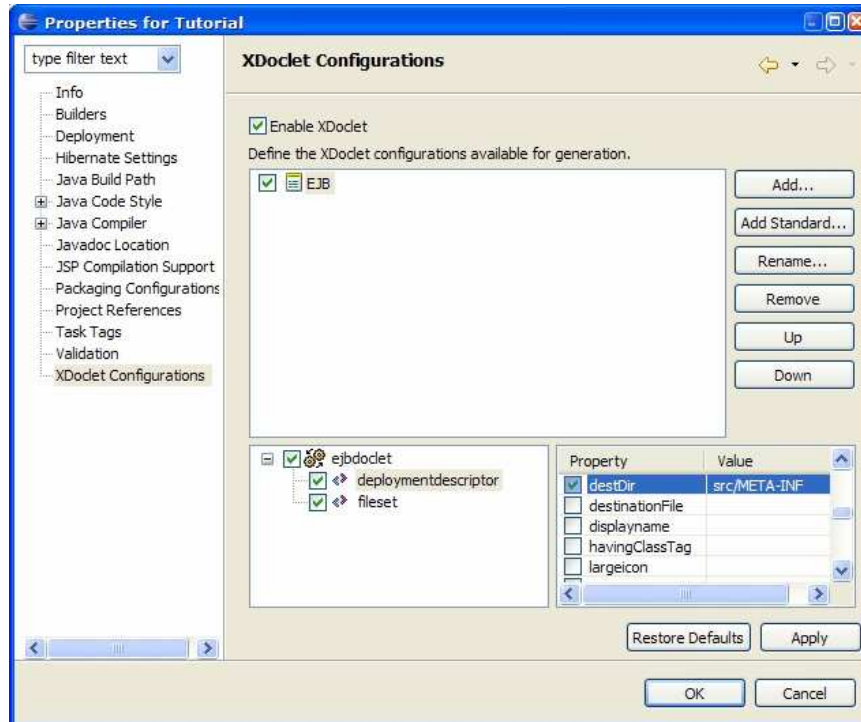


Σχήμα 5.11 XDoclet Configurations (d) – Fileset Configuration

Βήμα 5.Παραμετροποίηση Περιγραφέα Διάταξης (Deployment Descriptor Configuration)

- Προσθέστε ένα νέο deploymentdescriptor subtask στο ejbdoclet .
 - a. θέστε την destDir ιδιότητα σε src/META-INF.

Όλοι οι προκαθορισμένοι του EJB περιγραφείς διάταξης θα τοποθετηθούν τώρα στην src/META-INF διεύθυνση.



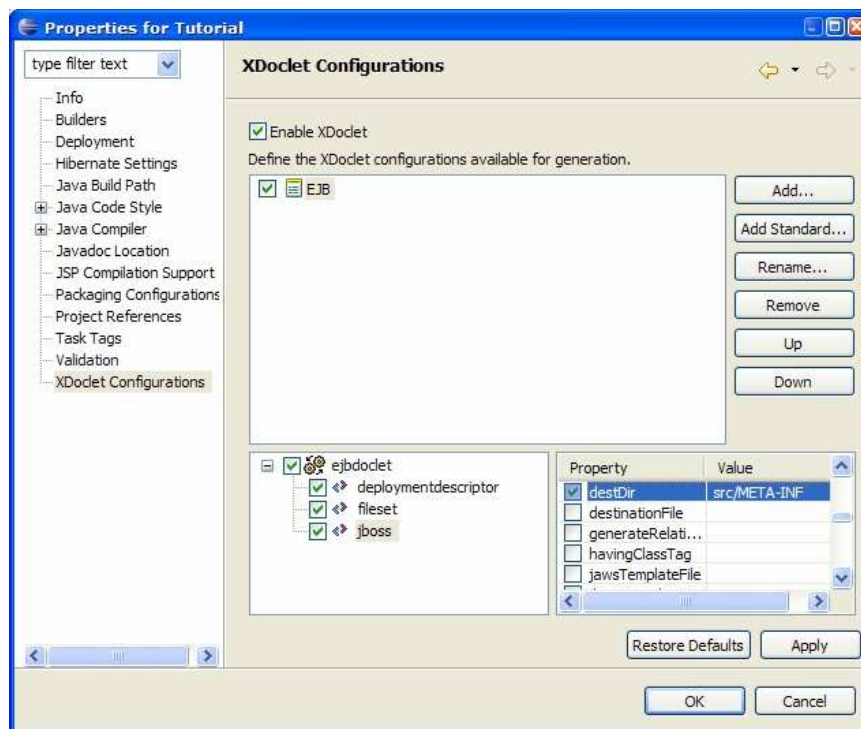
Σχήμα 5.12 XDoclet Configurations (e) – Deployment Descriptor Configuration

Βήμα 6. Παραμετροποίηση του JBoss (JBoss Configuration).

Προσθέστε ένα νέο jboss subtask στο ejbdoclet.

- a. θέστε την destDir ιδιότητα σε src/META-INF.
- b. θέστε την Version ιδιότητα σε το 3.0.

Όλοι οι παραπάνω περιγραφείς διατάξεις του JBoss θα τοποθετηθούν τώρα στην src/META-INF διεύθυνση.



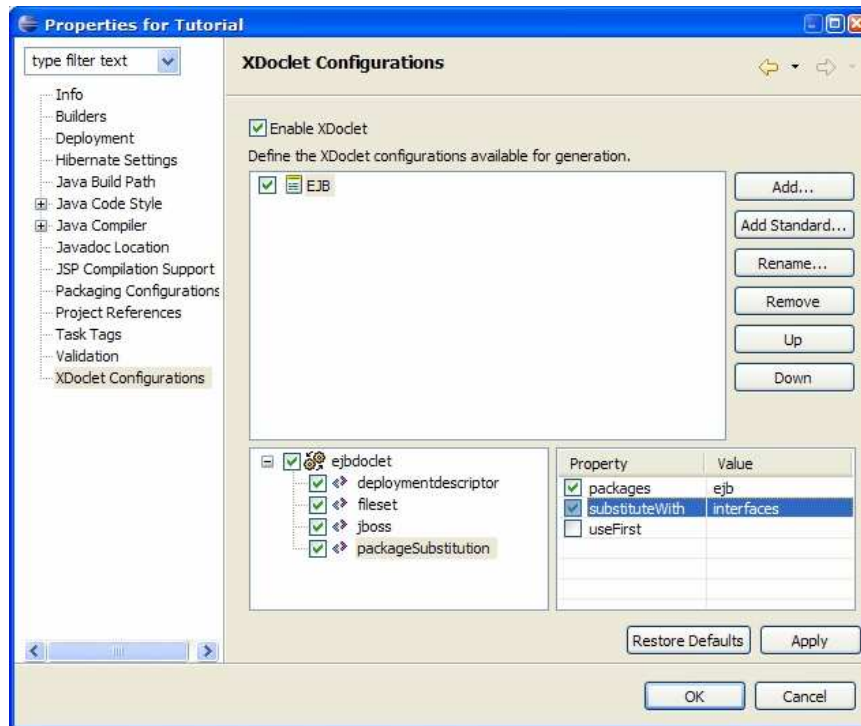
Σχήμα 5.13 XDoclet Configurations (f) – JBoss Configurations

Βήμα 7. Παραμετροποίηση Υποκατάστασης Πακέτων (Package Substitution Configuration).

Προσθέστε ένα νέο jboss subtask στο ejbdoclet.

- Προσθέστε ένα νέο packageSubstitution subtask στο ejbdoclet.
 - a. θέστε την packages ιδιότητα σε ejb.
 - b. θέστε την substituteWith ιδιότητα σε interfaces.

Με αυτό θα τοποθετηθούν τα δημιουργηθέντα EJB interfaces στο tutorial.interfaces java πακέτο.

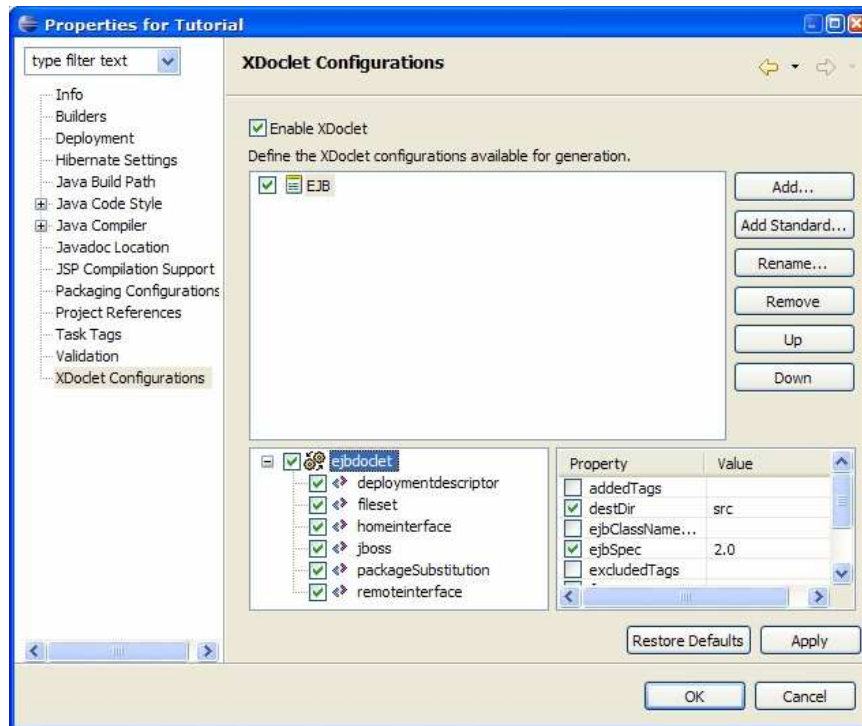


Σχήμα 5.14 XDoclet Configurations (g) – Package Substitution Configurations

Βήμα 8. Παραμετροποίηση Interface (Interface Configuration).

1. Προσθέστε ένα νέο remoteInterface subtask στο ejbdoclet.
2. Προσθέστε ένα νέο homeInterface subtask στο ejbdoclet.

Αυτά τα subtasks θα δημιουργήσουν τα EJB τοπικά (home) και απομακρυσμένα (remote) interfaces.



Σχήμα 5.15 XDoclet Configurations (h) – Interface Configurations

Κάντε κλικ στο OK και η XDoclet παραμετροποίηση για το Lawful_Interception θα αποθηκευτεί. Αφού η παραμετροποίηση αποθηκευτεί, κάντε δεξί κλικ στο Validation project και επιλέξτε Run XDoclet. Η δημιουργία XDoclet θα εμφανιστεί στην έξοδο (console). Αυτή θα είναι η εξής:

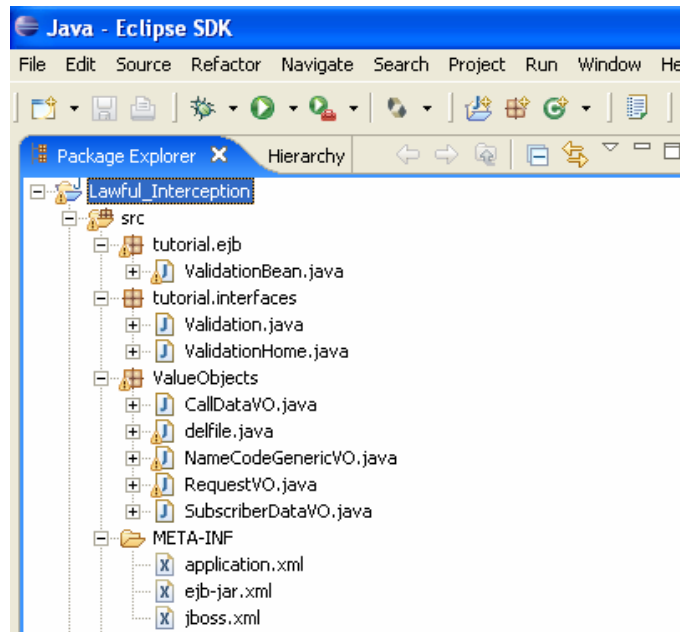
```

Problems Javadoc Declaration Console Debug
<terminated> C:\Program Files\Java\jdk1.5.0_09\bin\javaw.exe (2 Ιουλ 2007 7:24:39 μμ)
Buildfile: C:\Documents and Settings\george.poulopoulos\Τα έγγραφά μου\σχολι\privacy_dissertation\eclipse
N10004:
[ejbdoclet] (XDocletMain.start 47 ) Running <jboss/>
[ejbdoclet] (XDocletMain.start 47 ) Running <remoteinterface/>
[ejbdoclet] (XDocletMain.start 47 ) Running <homeinterface/>
[ejbdoclet] (XDocletMain.start 47 ) Running <deploymentdescriptor/>
_xdoclet_generation_:
BUILD SUCCESSFUL
Total time: 3 seconds

```

Σχήμα 5.16 Console Output for Run XDoclet Command

Τώρα project θα δείχνει κάπως έτσι. Ας σημειωθεί ότι ένα tutorial.interfaces package δημιουργήθηκε με νέες κλάσεις. Υπάρχει επίσης ένας META-INF φάκελος με τους περιγραφείς διατάξεις (τον standard και του jboss).



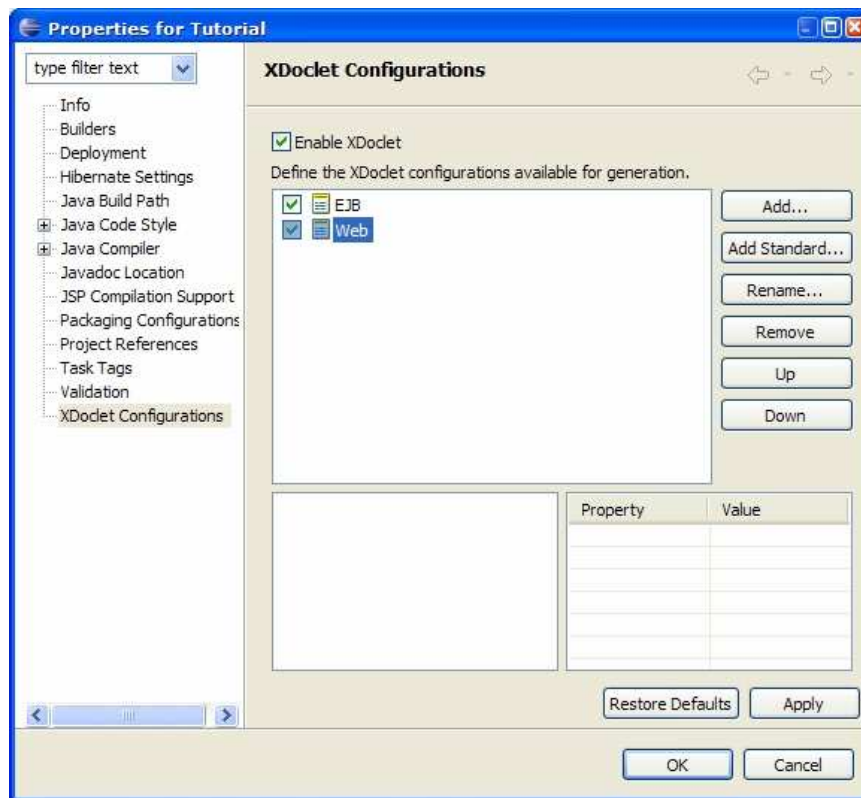
Σχήμα 5.17 Package Tree Structure after XDoclet build

Ας σημειωθεί ότι το package ValueObjects περιέχει κάποιες βοηθητικές κλάσεις που χρησιμοποιούνται από το EJB. Επίσης το cos.jar είναι ένα εξωτερικό jar (Add External JARs από το build path) το οποίο επίσης περιέχει βοηθητικές κλάσεις που χρησιμοποιούνται από το EJB.

5.1.3 Δημιουργία των Αρχείων που σχετίζονται με τα Συστατικά Ιστού (Web Components).

Βήμα 1. XDoclet EJB Δημιουργία και ρύθμιση παραμέτρων.

- Δεξί κλικ στην πάνω περιοχή ώστε να εμφανιστεί το pop-up μενού και επιλέξτε Add. Πληκτρολογήστε WEB στο dialog box και κάντε κλικ στο OK.

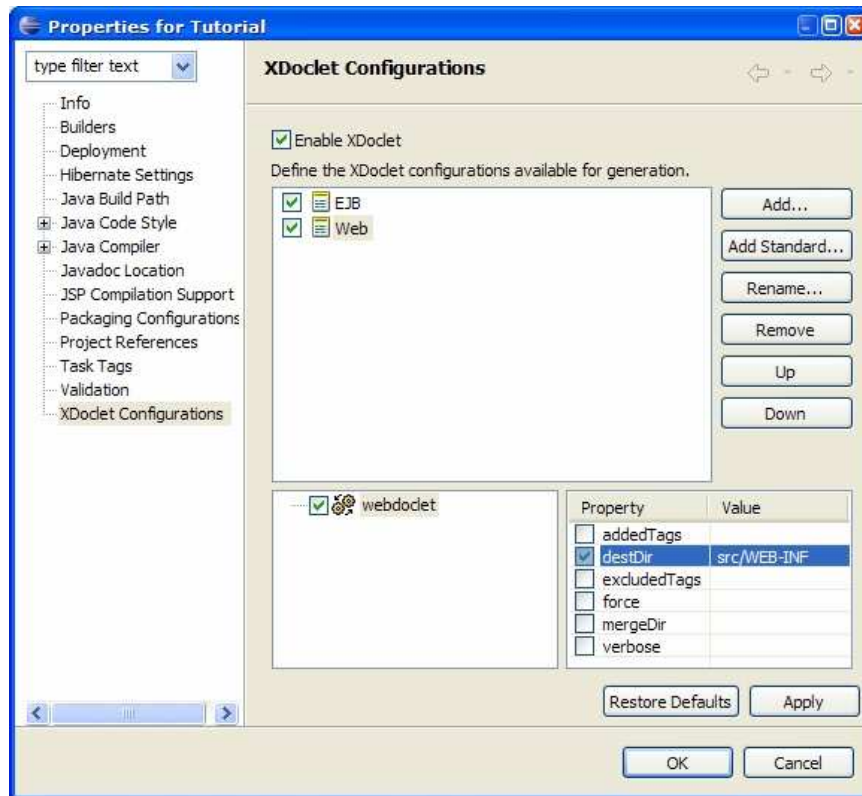


Σχήμα 5.18 XDodlet Configuration for Webdoclet

Βήμα 2. Παραμετροποίηση Webdoclet (Webdoclet Configuration).

1. Επιλέξτε το Web configuration.
2. Στην κάτω-αριστερά περιοχή, κάντε δεξί κλικ ώστε να εμφανιστεί το popup μενού και επιλέξτε Add Doclet.
3. Μία λίστα από διαθέσιμα doclets θα εμφανιστεί. Επιλέξτε webdoclet και κάντε κλικ στο OK.
4. Στην κάτω-δεξιά περιοχή, βλέπουμε τις ιδιότητες του ejbdodlet.
 - a. θέστε την destDir επιλογή σε src/WEB-INF.

Η παραμετροποίηση μας τώρα περιέχει ένα webdoclet που θα δημιουργήσει αρχεία στον src/WEB-INF φάκελο.

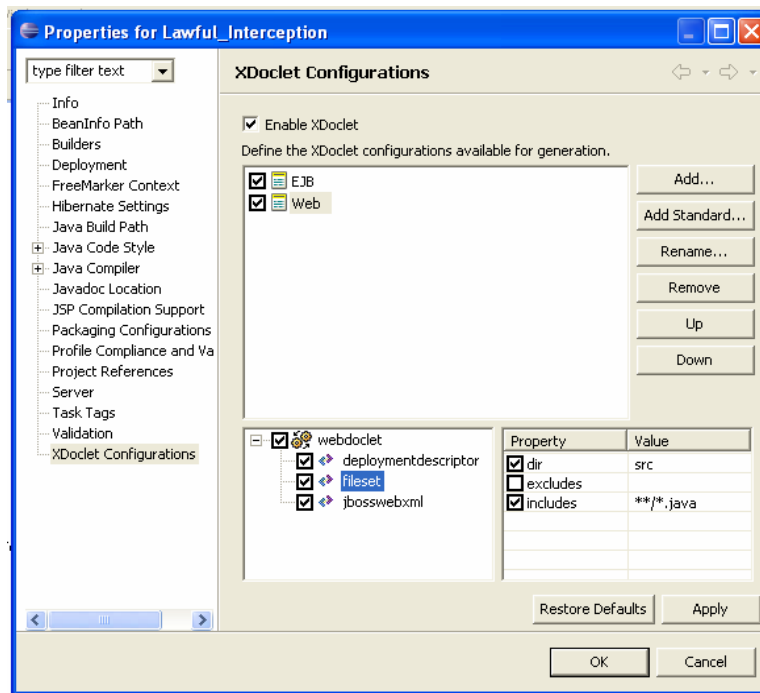


Σχήμα 5.19 Webdoclet Configuration

Βήμα 3. Παραμετροποίηση Fileset (Fileset Configuration)

1. Στην κάτω-αριστερά περιοχή, κάντε δεξί κλικ στο webdoclet ώστε να εμφανιστεί το popup μενού και επιλέξτε Add.
2. Μία λίστα από διαθέσιμα subtasks θα εμφανιστεί. Επιλέξτε fileset και κάντε κλικ στο Ok.
3. Στην κάτω-δεξιά περιοχή, βλέπουμε τις ιδιότητες του fileset.
 - a. θέστε την dir ιδιότητα σε src.
 - b. αφαιρέστε την επιλογή την ιδιότητα excludes
 - c. θέστε την includes ιδιότητα σε **/*.java.

Η παραμετροποίηση μας τώρα περιέχει ένα webdoclet με ένα fileset που περιέχει την src διεύθυνση, και όλα τα αρχεία «κάτω» από αυτήν που τελειώνουν σε *.java

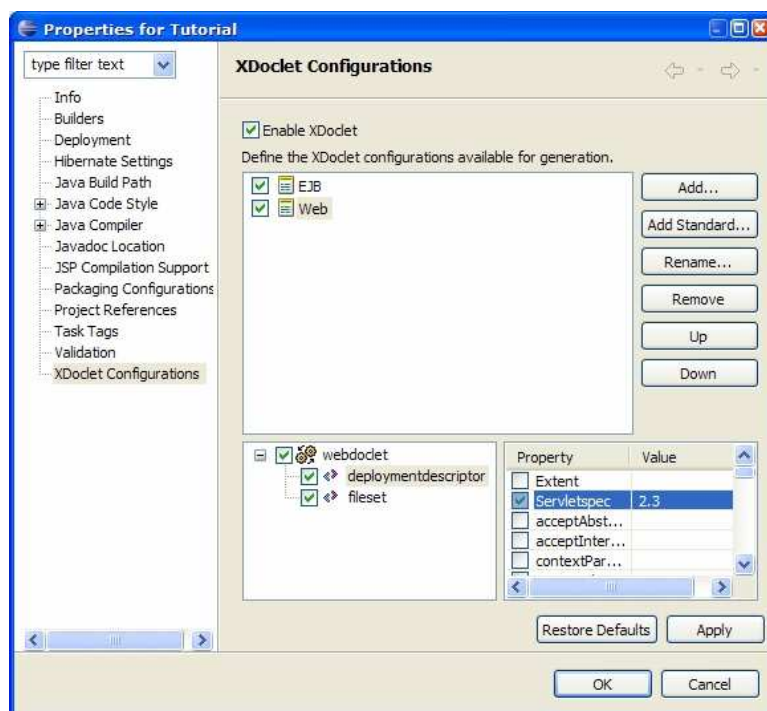


Σχήμα 5.20 Webdoclet Configuration – fileset

Βήμα 4.Παραμετροποίηση Περιγραφέα Διάταξης (Deployment Descriptor Configuration)

- Προσθέστε ένα νέο deploymentdescriptor subtask στο webdoclet.
 - a. θέστε την Servletspec ιδιότητα σε 2.3.

Όλοι οι προκαθορισμένοι του EJB περιγραφείς διάταξης θα τοποθετηθούν τώρα στην src/META-INF διεύθυνση.

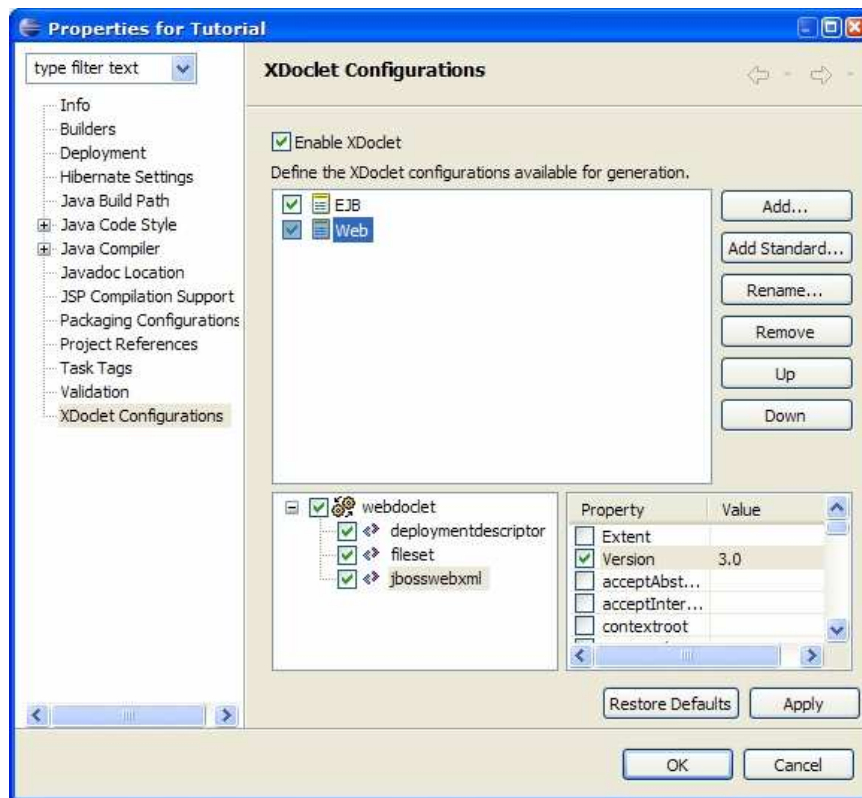


Σχήμα 5.21 Webdoclet Configuration – deployment descriptor

Βήμα 5. Παραμετροποίηση του JBoss (JBoss Configuration).

- Προσθέστε ένα νέο jbosswebxml subtask στο webdoclet.
 - α. θέστε την Version ιδιότητα σε to 3.0.

Όλοι οι παραπάνω JBoss-συγκεκριμένοι περιγραφείς διάταξης ιστού(JBoss-specific Web deployment descriptors) θα τοποθετηθούν τώρα στην src/WEB-INF διεύθυνση.



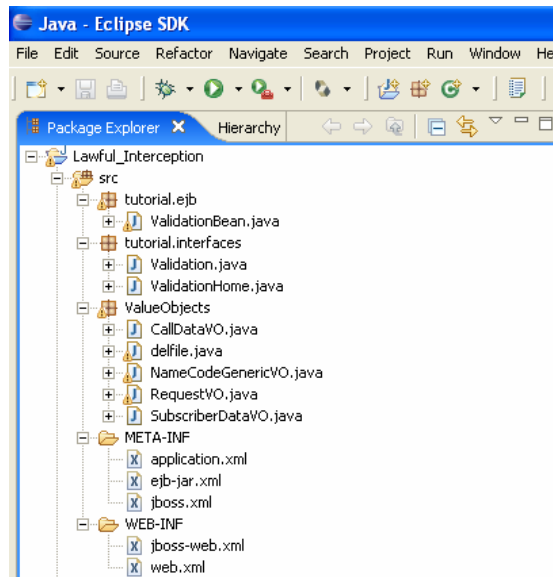
Σχήμα 5.22 Webdoclet Configuration – jboss

Κάντε κλικ στο OK και η XDoclet παραμετροποίηση για το Lawful_Interception θα αποθηκευτεί. Αφού η παραμετροποίηση αποθηκευτεί, κάντε δεξί κλικ στο Validation project και επιλέξτε Run XDoclet. Η δημιουργία XDoclet θα εμφανιστεί στην έξοδο (console). Αυτή θα είναι η εξής:

```
N102D0:
[webdoclet] (XDocletMain.start           47 ) Running <deploymentdescriptor/>
[webdoclet] Generating web.xml.
[webdoclet] (XDocletMain.start           47 ) Running <jbosswebxml/>
[webdoclet] (TemplateSubTask.engineStarted 806 ) Generating output 'jboss-web.xml' using template f
_xdoclet_generation_:
BUILD SUCCESSFUL
Total time: 3 seconds
```

Σχήμα 5.23 Webdoclet Configuration – Run XDoclet console output

Τώρα project θα δείχνει κάπως έτσι. Ας σημειωθεί ότι δημιουργηθεί ένας φάκελος με το όνομα WEB-INF με τους περιγραφείς διάταξης Ιστού (τον standard και του jboss Web deployment descriptors).

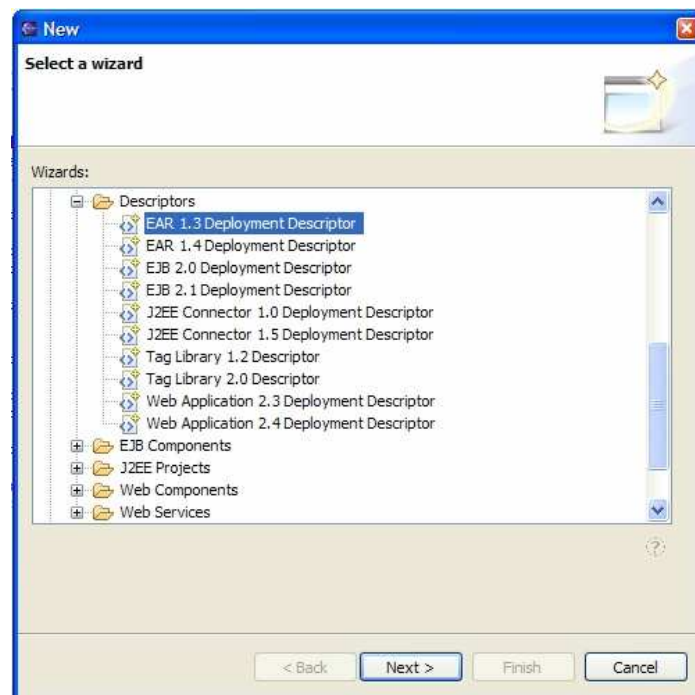


Σχήμα 5.24 Updated Project Package Tree Structure

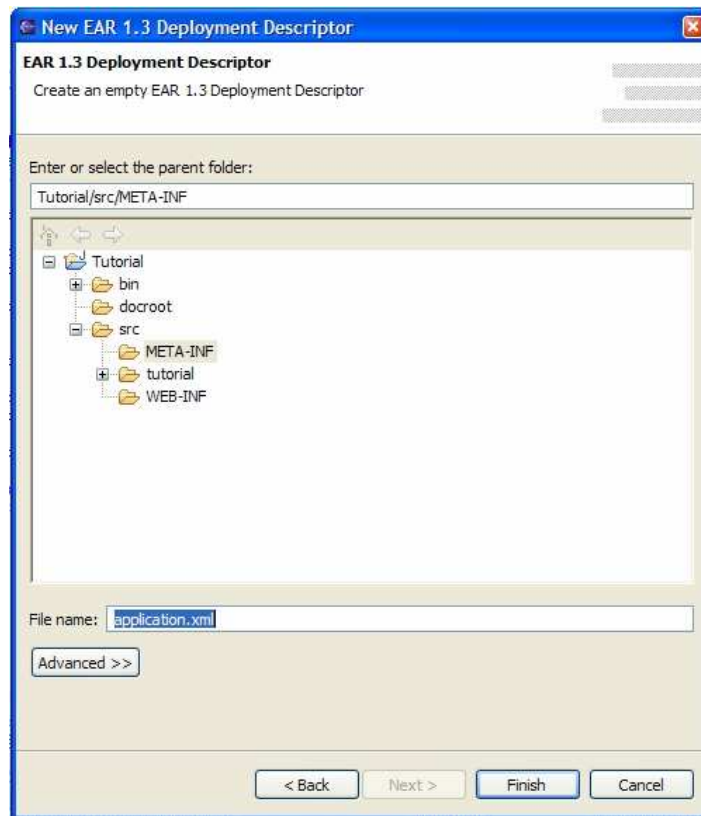
5.1.4 Δημιουργία EAR Περιγραφέων Διάταξης (EAR Deployment Descriptors).

Βήμα 1. Δημιουργία του application.xml

1. Με δεξί-κλικ στον src/META-INF φάκελο, επιλέγουμε New > Other....
2. Επιλέγουμε JBoss-IDE > Descriptors > EAR 1.3 Deployment Descriptor, και μετά Next.



Σχήμα 5.25 Create application.xml (a)



Σχήμα 5.26 Create application.xml (b)

Το application.xml πρέπει να είναι το εξής :

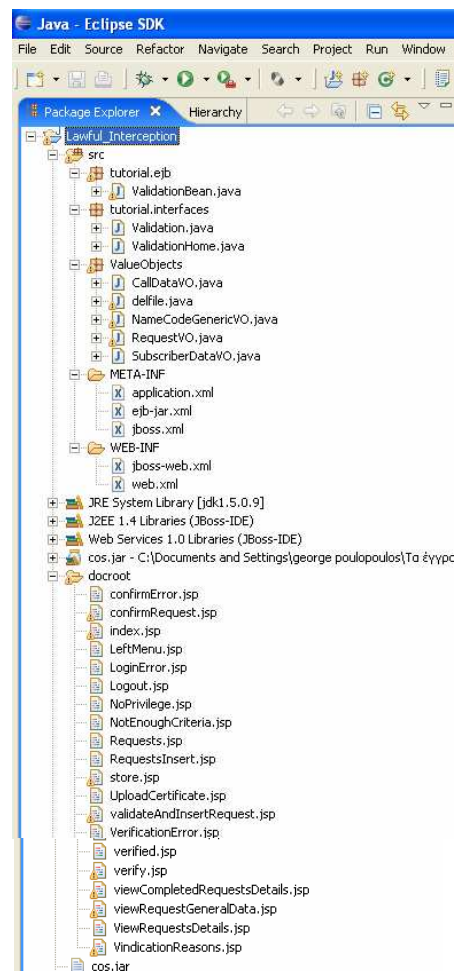
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE application PUBLIC
    "-//Sun Microsystems, Inc.//DTD J2EE Application 1.3//EN"
    "http://java.sun.com/dtd/application_1_3.dtd">
<application>
    <display-name>Sum Application</display-name>
    <module>
        <ejb>ValidationEJB.jar</ejb>
    </module>
    <module>
        <web>
            <web-uri>ValidationWeb.war</web-uri>
            <context-root>validation</context-root>
        </web>
    </module>
</application>
```

5.1.5 Δημιουργία των Συστατικών Ιστού (Web Components).

Για την εφαρμογή αυτή τα J2EE συστατικά του ιστού είναι σελίδες που έχουν δημιουργηθεί από την JSP τεχνολογία (JSP pages).

Αφού δημιουργήσουμε έναν φάκελο με το όνομα docroot τοποθετούμε όλες τις JSP σελίδες της εφαρμογής εκεί.

Έτσι το project θα έχει την εξής τελική μορφή:



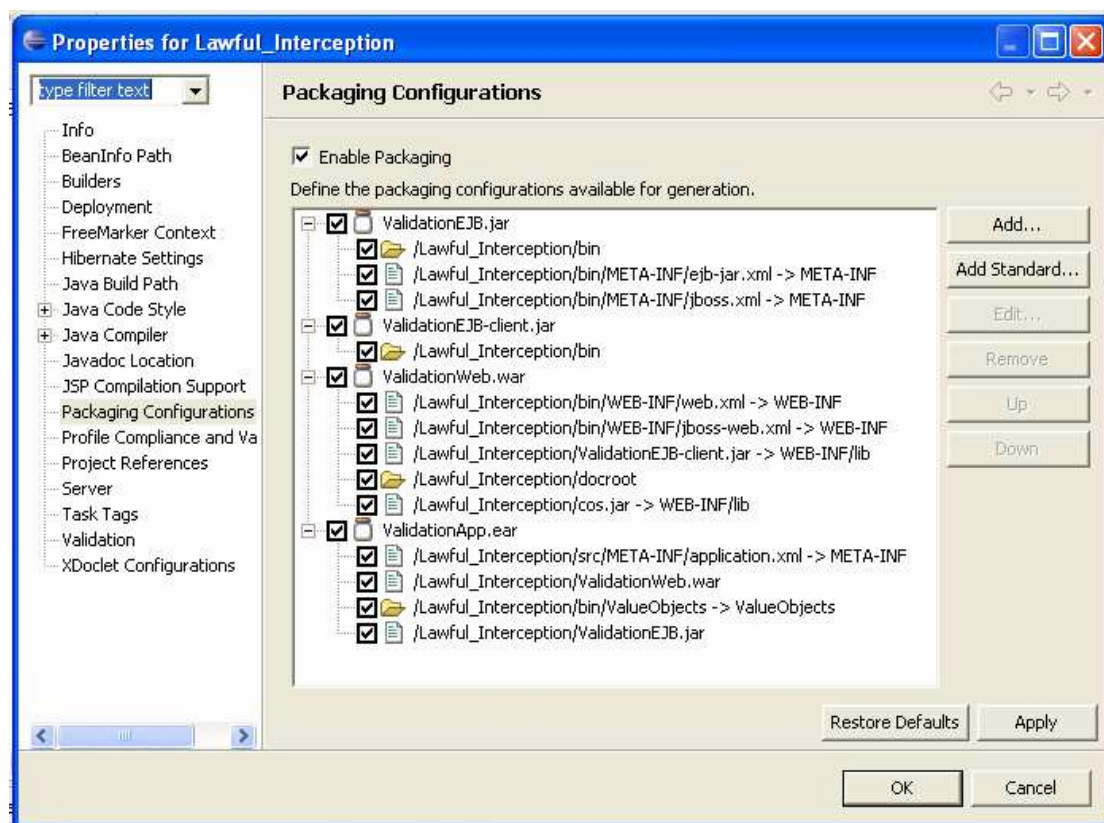
Σχήμα 5.27 Project final Package Tree Structure

5.1.6 Συσκευασία Εφαρμογής (Packaging).

Το επόμενο βήμα είναι η Συσκευασία, η δημιουργία δηλαδή (όπως έχει αναφερθεί και στο κεφάλαιο 5 παράγραφος 3.4) των αρχείων-πακέτων που θα χρησιμοποιεί ο server (JBoss) για την διάθεση της εφαρμογής.

Ενεργοποίηση Συσκευασίας (Enable Packaging).

1. Δεξί κλικ στο project and επιλέξτε Properties.
2. Στη σελίδα των properties , επιλέξτε Packaging configurations.
3. Στην αρχή της σελίδας υπάρχει ένα κουτί επιλογής (check-box) με ιδιότητα Enable Packaging. Επιλέξτε το.
4. Στη συνέχεια με τα πλήκτρα Add... και ύστερα με δεξί κλικ πάνω στο όνομα του πακέτου που κάναμε Add... και επιλέγοντας Add File... ή Add Folder... και το αντίστοιχο αρχείο ή φάκελο έχουμε την ακόλουθη εικόνα.

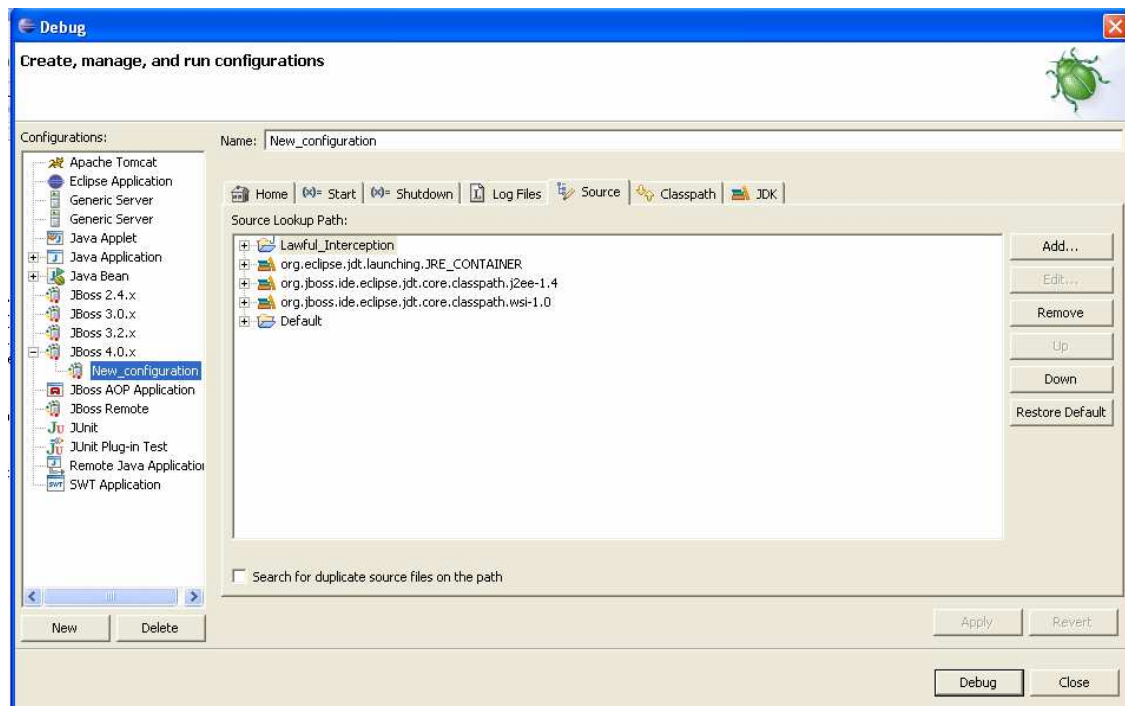


Σχήμα 5.28 Packaging Configuration

5.1.7 Ρύθμιση Παραμέτρων και Έναρξη Εξυπηρετητή (Launch & Deploy to JBoss).

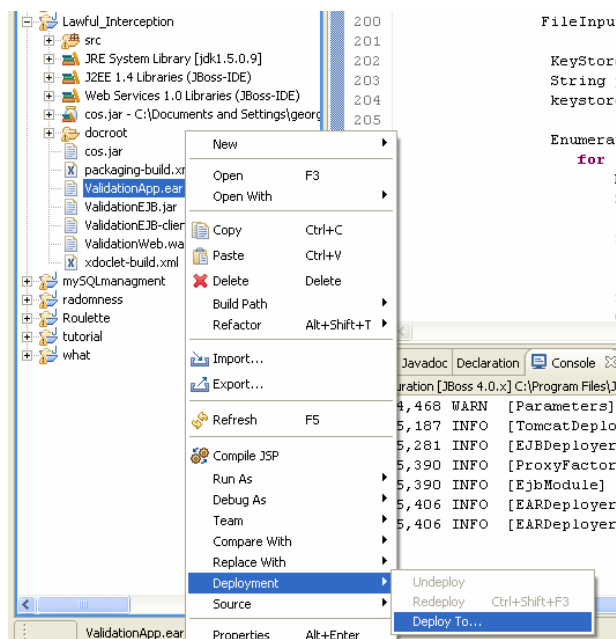
Τέλος αυτό που μένει είναι η έναρξη λειτουργίας του Εξυπηρετητή και η διάθεση της εφαρμογής σε αυτόν.

Επιλέγοντας με δεξί κλικ πάνω στο project «Debug as...-> debug» επιλέγουμε στο μενού που θα εμφανιστεί την διεύθυνση που βρίσκεται ο εξυπηρετητής και στο «source» κάνουμε Add... το project περιλαμβάνοντας όλες τις βιβλιοθήκες και αφαιρώντας με «remove» το cos.jar.



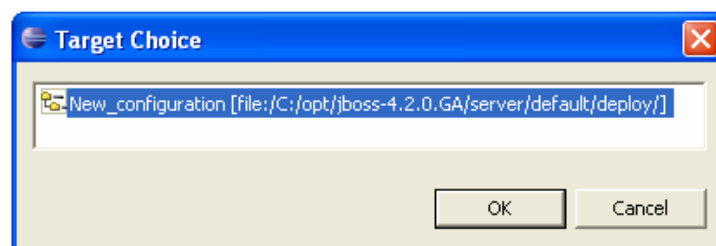
Σχήμα 5.29 Launch and Deploy JBoss

Στη συνέχεια αφού ο JBoss έχει ξεκινήσει (παρατηρώντας το Console) για να διαθέσουμε την εφαρμογή στον εξυπηρετητή κάνουμε δεξί κλικ στο ValidationApp.ear και από το μενού που εμφανίζεται : Deployment -> Deploy To...



Σχήμα 5.30 Deployment of ValidationApp.ear

Στη συνέχεια επιλέγουμε OK στο πιο κάτω μενού που θα εμφανιστεί.



Σχήμα 5.31 Selection of server for deployment of ValidationApp.ear

Η εφαρμογή είναι τώρα διαθέσιμη στον εξυπηρετητή (JBoss) και έτοιμη για χρήση.

5.2 Δημιουργία και Επεξήγηση των Πινάκων της Βάσης Δεδομένων **mysql**.

Το «Σχήμα» της βάσης με όνομα *diplwmatiki* θα αποτελείται από τους εξής πίνακες :

1. Πίνακας **subscriber_data**

Ο πίνακας αυτός θα περιέχει τα στοιχεία των συνδρομητών της εταιρίας τηλεπικοινωνιών που διαθέτει την εφαρμογή αυτή. Αυτά θα είναι:

- *unique_id* (Ένας μοναδικός αριθμός για κάθε εγγεγραμμένο συνδρομητή.)
- *firstname*
- *fathurname*
- *lastname*
- *afm* (Ο αριθμός φορολογικού μητρώου του εκάστοτε συνδρομητή.)
- *city*
- *country*
- *street*
- *area*
- *street_number*
- *phone_number*
- *id_number* (Ο αριθμός ταυτότητας του εκάστοτε συνδρομητή.)
- *birth_date*
- *IMEI*
- *IMSI*

Κάθε πεδίο μπορεί να έχει την τιμή *null* σε περίπτωση που για κάποιον λόγο (π.χ. προπληρωμένος χρήστης) δεν έχουμε όλα τα στοιχεία του συνδρομητή. Τουλάχιστον ένα θα πρέπει να είναι διάφορο του *null* ώστε να είναι καταχωρημένο σε αυτόν τον πίνακα.

2. Πίνακας **corp_num_prefix**

Ο πίνακας αυτός θα περιέχει το όνομα της εταιρίας τηλεπικοινωνιών που ανήκει ο συνδρομητής σε περίπτωση που ο αριθμός τηλεφώνου δεν αντιστοιχεί σε συνδρομητή της εταιρίας που διαθέτει την εφαρμογή αυτή. Ως στοιχεία θα έχει:

- *Corporate_name* (Όνομα της εταιρίας.)
- *Number_prefix* (Τα 3 πρώτα ψηφία του τηλεφωνικού αριθμού)

3. Πίνακας call_data

Ο πίνακας αυτός θα περιέχει τις κλήσεις που έχουν πραγματοποιηθεί και περιλαμβάνουν όπως είναι λογικό κλήσεις που τουλάχιστον ο καλών ή ο καλούμενος είναι συνδρομητής της εταιρίας. Ο πίνακας θα έχει τα εξής στοιχεία:

- call_id (Ένας μοναδικός αριθμός αντιστοιχίζεται σε κάθε κλήση.)
- caller_unique_id (Ο αριθμός αυτός είναι ο unique_id του συνδρομητή από τον πίνακα subscriber_data ή null αν δεν είναι συνδρομητής της εν λόγω εταιρίας.)
- caller_number
- caller_imsi
- caller_imei
- caller_BTS_id
- caller_location
- call_date
- call_start_time
- call_end_time
- callee_id (Ο αριθμός αυτός είναι ο unique_id του συνδρομητή από τον πίνακα subscriber_data ή null αν δεν είναι συνδρομητής της εν λόγω εταιρίας.)
- callee_number
- callee_imsi
- callee_imei
- callee_BTS_id
- callee_location
- ektrapi (yes/null αν ο καλούμενος έχει ενεργοποιήσει την υπηρεσία εκτροπής κλήσεων η όχι αντίστοιχα)
- ektrapi_number (Ο αριθμός που έχει γίνει εκτροπή στον καλούμενο.)
- prepaid_user (yes/null αν ο καλών είναι προπληρωμένος χρήστης/κάρτα η όχι αντίστοιχα)

4. Πίνακας gen_user

Ο πίνακας αυτός θα περιέχει τα στοιχεία των ατόμων που θα έχουν δικαίωμα πρόσβασης στο αρχείο κλήσεων της εταιρίας καθώς και τα στοιχεία για την είσοδο τους στην υπηρεσία (username,password). Τα στοιχεία του πίνακα θα είναι:

- username
- password
- first_name
- last_name
- group (Η υπηρεσία που ανήκει το εκάστοτε μέλος- autorney/ adae/ police)

5. Πίνακας requests

Ο πίνακας αυτός θα περιέχει τα στοιχεία των αιτήσεων που εισάγει ο κάθε χρήστης που έχει την ανάλογη αίτηση και στοιχεία σχετικά με το αν έχει εγκριθεί από όλες τις υπηρεσίες ή όχι. Τα στοιχεία του πίνακα είναι:

- username (Με βάση το username του κάθε εγκεκριμένου χρήστη θα εισάγεται το αίτημά του.)
- request_id (Σε κάθε νέο αίτημα εκχωρείται αυτόματα ένας αύξων αριθμός, μοναδικός για κάθε αίτημα.)
- req_name(Το μικρό όνομα του συνδρομητή για τον οποίο ζητείται άρση του απορρήτου κλήσεων.)
- req_fname (Το όνομα πατέρα.)
- req_lname(Το επίθετο)
- req_birthDate
- req_afm
- req_idNum (Ο αριθμός ταυτότητας.)
- req_phoneNum
- req_country
- req_city
- req_area
- req_street
- req_streetNum
- req_dateFrom (Η ημερομηνία από την οποία και μετά ζητούνται οι κλήσεις για τον συνδρομητή.)
- req_dateTo (Η ημερομηνία μέχρι την οποία ζητούνται οι κλήσεις για τον συνδρομητή.)
- automey /adae / police (Ανάλογα με την υπηρεσία που ανήκει ο κάθε χρήστης το αντίστοιχο πεδίο τίθεται ίσο με 1 αν έχει εισάγει ή εγκρίνει το εν λόγω αίτημα.)
- reason (Το πεδίο αυτό περιέχει μια σύντομη περιγραφή του λόγου για τον οποίο ζητείται άρση του απορρήτου κλήσεων για τον εκάστοτε συνδρομητή που βρίσκεται στον πίνακα αυτόν.)

6. Πίνακας gen_user_roles

Ο πίνακας αυτός χρησιμεύει για τον καθορισμό των δικαιωμάτων πρόσβασης του κάθε χρήστη στην εφαρμογή. Αντιστοιχίζει το κάθε δικαίωμα σε έναν ακέραιο θετικό αριθμό. Το δικαίωμα InsertRequest σε 1 για την εισαγωγή αιτήματος, το ConfirmRequest σε 2 για να έχει πρόσβασή στα αιτήματα που εκκρεμούν και να προβεί στην έγκριση κάποιου αιτήματος, το ViewRequest σε 3 για να έχει πρόσβαση στα αιτήματα που έχει εισάγει αυτός και έχουν ολοκληρωθεί και τέλος το All σε 4 που στην περίπτωση αυτή έχει το δικαίωμα για εισαγωγή αιτήματος, για να δει τα όσα εκκρεμούν και να προβεί εάν επιθυμεί στην έγκριση κάποιου από αυτά και τέλος να μπορεί να έχει πρόσβαση σε όλα τα αιτήματα που έχουν ολοκληρωθεί, ανεξάρτητα αν τα έχει εισάγει αυτός ή όχι. Πρέπει να σημειωθεί ότι ο πίνακας αυτός δεν

συμμετέχει ενεργά στην εφαρμογή και πρέπει να έχει αυτές ακριβώς τις αντιστοιχίσεις που περιγράφηκαν πιο πάνω. Τα στοιχεία του πίνακα είναι:

- role (Η περιγραφή του δικαιώματος πρόσβασης.)
- role_id (Ο αέριος που αντιστοιχίζεται, 1, 2, 3, ή 4.)

7. Πίνακας user_roles_management

Ο πίνακας αυτός είναι υπεύθυνος για την απόδοση σε κάθε χρήστη του δικαιώματος που καθορίζει ο διαχειριστής της εφαρμογής. Ο πίνακας έχει τρία πεδία:

- username (Το username του χρήστη, με βάση αυτό γίνεται η απόδοση του εκάστοτε δικαιώματος πρόσβασης στους χρήστες.)
- role_id (Ο αέριος αριθμός που αντιστοιχεί στο κάθε δικαίωμα πρόσβασης στην εφαρμογή.)
- index (Ένας αύξων αριθμός που αποδίδεται από την εφαρμογή σε κάθε εγγραφή του πίνακα αυτού, μοναδικός για κάθε αντιστοίχιση χρήστη-δικαιώματος πρόσβασης.)

5.3 Ρύθμιση Παραμέτρων του JBoss για χρήση SSL.

Προκειμένου να κάνουμε χρήση της δυνατότητας SSL του JBoss θα πρέπει να τροποποιήσουμε κάποιες παραμέτρους του JBoss. Στον κατάλογο `..\server\default\deploy\jboss-web.deployer` υπάρχει το αρχείο `server.xml`, τροποποιούμε το αρχείο αυτό ως εξής:

server.xml:

```
<Server>

  <!--APR library loader. Documentation at /docs/apr.html -->
  <Listener className="org.apache.catalina.core.AprLifecycleListener"
    SSLEngine="on" />
  <!--Initialize Jasper prior to webapps are loaded. Documentation at /docs/jasper-
    howto.html -->
  <Listener className="org.apache.catalina.core.JasperListener" />

  <!-- Use a custom version of StandardService that allows the
    connectors to be started independent of the normal lifecycle
    start to allow web apps to be deployed before starting the
    connectors.
    -->
  <Service name="jboss.web">
```

```

<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL HTTP/1.1 Connector on port 8080
-->
<!--
<Connector port="8080" address="{jboss.bind.address}"
    maxThreads="250" maxHttpHeaderSize="8192"
    emptySessionPath="true" protocol="HTTP/1.1"
    enableLookups="false" redirectPort="8443" acceptCount="100"
    connectionTimeout="20000" disableUploadTimeout="true" />
-->
<!-- Define a SSL HTTP/1.1 Connector on port 8443
This connector uses the JSSE configuration, when using APR, the
connector should be using the OpenSSL style configuration
described in the APR documentation -->

<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    keystoreFile="thedirectoryofkeystore"
    keystorePass="thekeystorepassword"/>

```

Αφήνουμε το υπόλοιπο αρχείο ως έχει.

Για την δημιουργία του keystore που θα χρησιμοποιηθεί το keytool της java και συγκεκριμένα του JDK1.5.0_09. Από την διεύθυνση bin του JDK δίνουμε την εντολή:

```
keytool -genkey -alias <alias name> -keyalg RSA -keystore ./name.keystore -validity
<the days to be valid from now on>
```

Στο keystoreFile="thedirectoryofkeystore" όπου thedirectoryofkeystore θέτουμε τη διεύθυνση που βρίσκεται το αρχείο που δημιουργήσαμε με το keytool και στο keystorePass="thekeystorepassword" όπου thekeystorepassword τον κωδικό του keystore.

5.4 Δημιουργία Ψηφιακών Πιστοποιητικών (Digital Certificates).

Τα ψηφιακά πιστοποιητικά είναι απαραίτητα για την πιστοποίηση της αυθεντικότητας των χρηστών της εφαρμογής, και αυτό για να διασφαλίσουμε μέγιστη ασφάλεια του συστήματος όσο αφορά την πρόσβαση των χρηστών.

Η δημιουργία των πιστοποιητικών αυτών γίνεται με χρήση του εργαλείου keytool της java και συγκεκριμένα της JDK1.5.0_09. Από την διεύθυνση bin του JDK δίνουμε την εντολή (από Command prompt):

```
keytool -genkey -alias <aliasname> -keypass <keypass> -storepass <storepass> -  
validity <validitydays >
```

Όπου aliasname το όνομα που θα έχει η εγγραφή στο keystore, keypass και storepass ο κωδικός της εγγραφής και του keystore αντίστοιχα και validitydays ο χρόνος από τη ημερομηνία που δημιουργήθηκε η εγγραφή για τον οποίο θα ισχύει το πιστοποιητικό, μετρούμενος σε μέρες.

Στη συνέχεια θα μας ζητηθεί το όνομα και κάποια άλλα στοιχεία του ατόμου που θα ανήκει η εγγραφή και για τον οποίο θα προορίζεται το συγκεκριμένο πιστοποιητικό. Το ονοματεπώνυμο θα χρησιμοποιηθεί κατά την επαλήθευση του χρήστη κατά την είσοδο του στην υπηρεσία που προσφέρει η εφαρμογή, πρέπει να τονίσουμε ότι το aliasname θα πρέπει να είναι το ίδιο με το username του εκάστοτε χρήστη αφού με βάση το username θα αναζητείται εγγραφή στο keystore του συστήματος από την οποία (η εφαρμογή) θα παίρνει το public key προς επαλήθευση του πιστοποιητικού που θα εισάγει ο χρήστης. Αφού ολοκληρωθεί η διαδικασία αυτή δημιουργίας νέας εγγραφής και πιστοποιητικού στο keystore ακολουθεί η επόμενη εντολή η οποία εξάγει το Ψηφιακό Πιστοποιητικό (self-signed Digital Certificate) σε μορφή αρχείου με κατάληξη .der.

```
keytool -export -rfc -alias <aliasname> -file <any_name>.der
```

Όπου aliasname το όνομα με το οποίο δημιουργήσαμε πιο πάνω το ψηφιακό πιστοποιητικό και any_name ένα οποιοδήποτε όνομα επιθυμούμε για το ψηφιακό πιστοποιητικό που θα διανεμηθεί σε έναν χρήστη με username ίδιο με το aliasname.

Από την «home» διεύθυνση του χρήστη αντιγράφουμε το Αρχείο .KEYSTORE στον φάκελο bin του JBoss.

Τέλος δημιουργούμε έναν φάκελο με το όνομα LI (C:\LI) ο οποίος θα χρησιμεύσει για την προσωρινή αποθήκευση των ψηφιακών πιστοποιητικών που θα προσκομίζει κάθε φορά ο χρήστης. Αξίζει να σημειωθεί ότι μετά την επαλήθευσή τους αυτά θα διαγράφονται από τον εν λόγω φάκελο. Επίσης στον φάκελο bin του JBoss θα πρέπει να τοποθετήσουμε το αρχείο που περιέχει το κλειδί για την κρυπτογράφηση και αποκρυπτογράφηση, προς και από τη βάση των δεδομένων (code.txt για την εφαρμογή αυτή).

6

6 Υπηρεσίες Εφαρμογής και Σενάρια Χρηστών

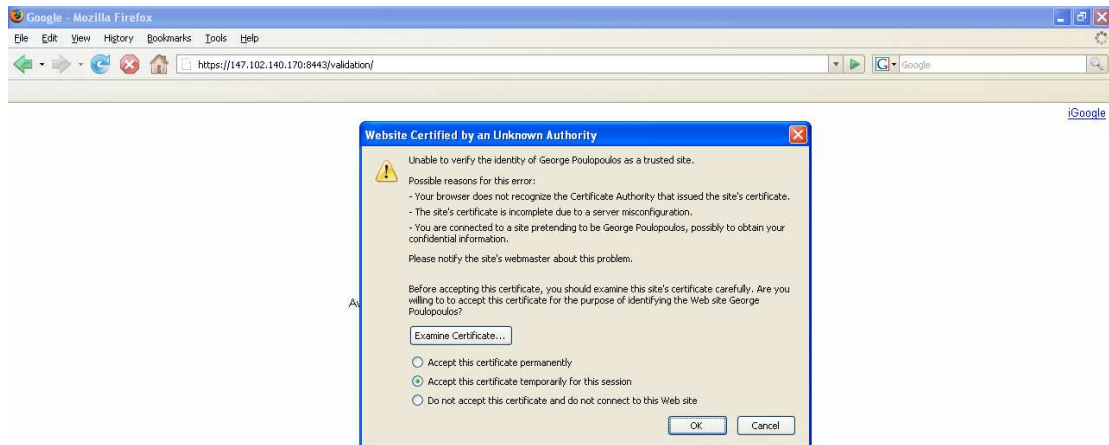
Στο κεφάλαιο αυτό θα εξετάσουμε την εφαρμογή από την πλευρά των πελατών (clients) και θα παρουσιάσουμε κάποια παραδείγματα χρηστών και αντίστοιχες υπηρεσίες της εφαρμογής για τον κάθε ένα.

6.1 Είσοδος Χρηστών και Αποδοχή Ψηφιακού Πιστοποιητικού.

Αφού ο χρήστης μέσω μιας εφαρμογής πλοήγησης στο διαδίκτυο επισκεφτεί το δικτυακό τόπο της εφαρμογής θα αντιμετωπίσει τα επόμενα βήματα προκειμένου να εισέλθει στο κύριο μενού της εφαρμογής και γενικότερα στις υπηρεσίες της εφαρμογής.

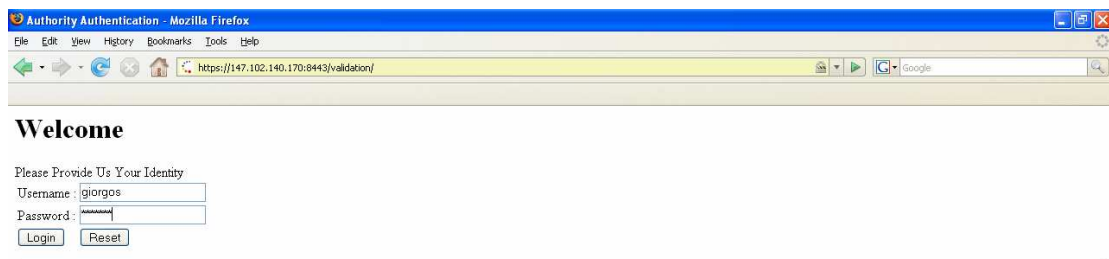
Αυτή η διαδικασία ακολουθείται από όλους τους χρήστες, όπως είναι λογικό, ανεξάρτητα των δικαιωμάτων πρόσβασης (όσο αφορά την κάθε προσφερόμενη υπηρεσία της εφαρμογής, θα εξηγηθεί στα πιο κάτω βήματα αναλυτικότερα).

Στο πρώτο αυτό βήμα θα ζητηθεί από τον χρήστη να αποδεχθεί το Ψηφιακό Πιστοποιητικό της εφαρμογής αφού θα γίνει χρήση της υπηρεσίας SSL.



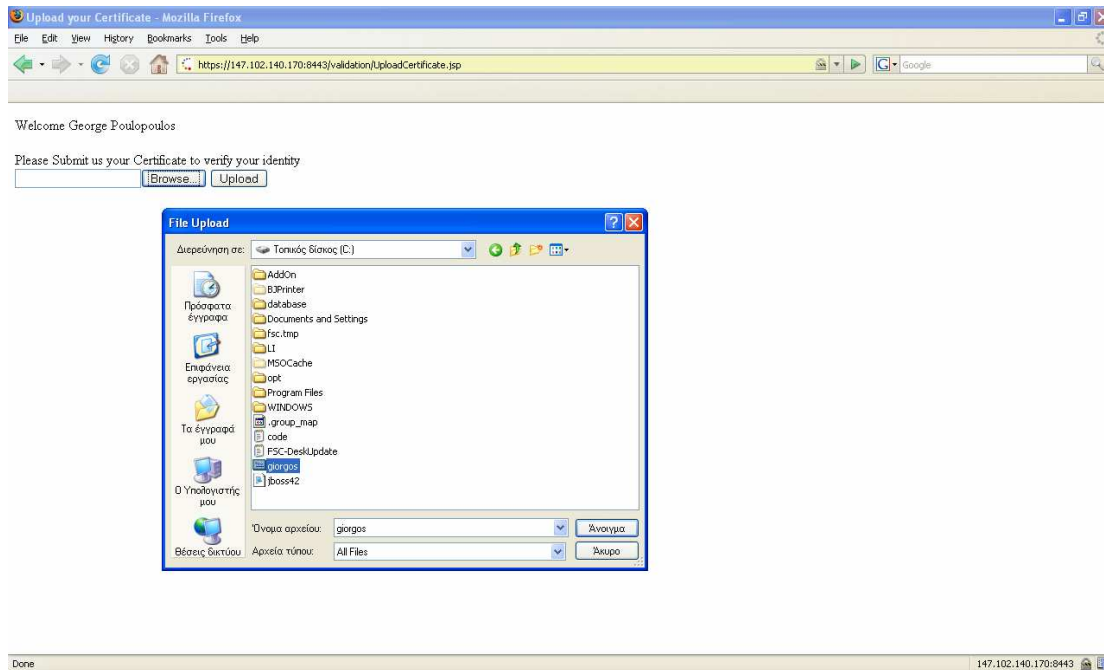
Σχήμα 6.1 Αποδοχή Ψηφιακού Πιστοποιητικού για χρήση SSL

Στη συνέχεια θα ζητηθούν τα στοιχεία του εκάστοτε χρήστη.



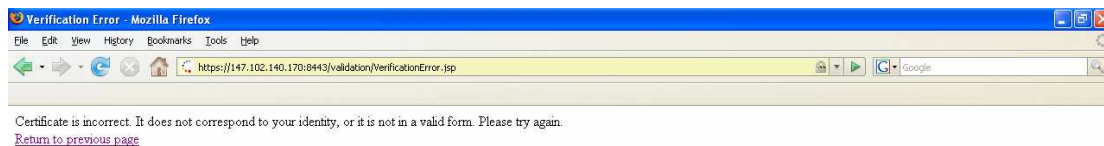
Σχήμα 6.2 Login στην εφαρμογή

Κατόπιν θα ζητηθεί το Ψηφιακό Πιστοποιητικό προς επιβεβαίωση της ταυτότητας του χρήστη.



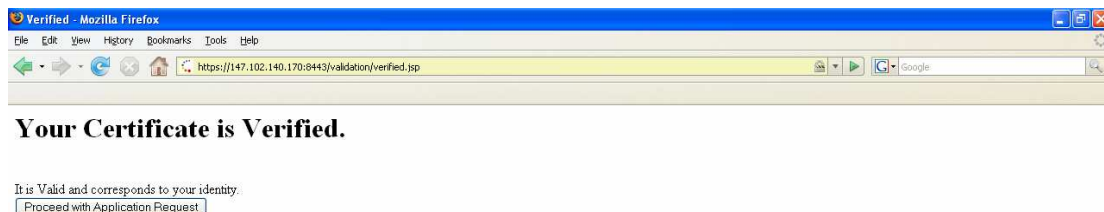
Σχήμα 6.3 Upload ψηφιακού πιστοποιητικού χρήστη

Σε περίπτωση που το Ψηφιακό Πιστοποιητικό είναι εσφαλμένο ή δεν πρόκειται για αρχείο Πιστοποιητικού εμφανίζεται αντίστοιχο μήνυμα λάθους, η ακόλουθη σελίδα.



Σχήμα 6.4 Σφάλμα κατά την πιστοποίηση του ψηφιακού πιστοποιητικού

Αλλιώς αν επιβεβαιωθεί η ταυτότητα του χρήστη μέσω του Πιστοποιητικού αυτού εμφανίζεται η ακόλουθη σελίδα.

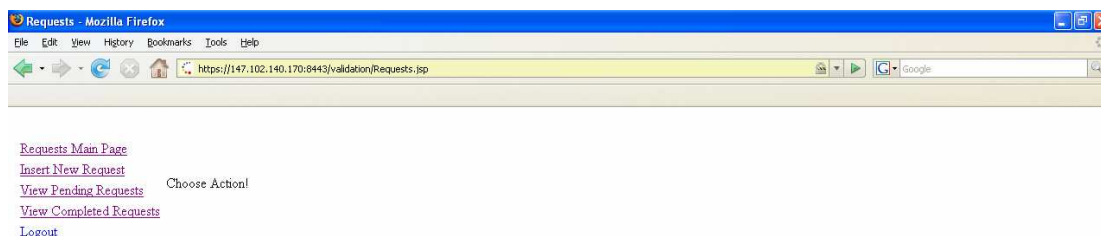


Σχήμα 6.5 Επιβεβαιωμένο ψηφιακό πιστοποιητικό

6.2 Κύριο Μενού Εφαρμογής και Δικαιώματα Πρόσβασης Χρηστών.

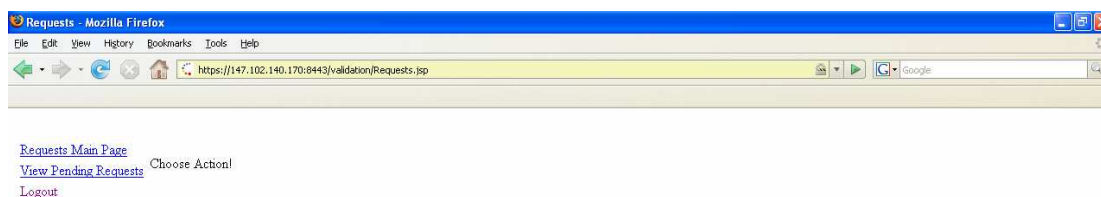
Ανάλογα με τα δικαιώματα πρόσβασης των χρηστών το κύριο μενού της εφαρμογής ενδέχεται να ποικίλει ως προς τις υπηρεσίες που θα παρέχει στον κάθε χρήστη. Τα δικαιώματα πρόσβασης καθορίζονται από τον πίνακα **user_roles_management** όπως αναλύθηκε και στο κεφάλαιο 5. Για το συγκεκριμένο παράδειγμα έχουμε τρεις χρήστες, με usernames : **giorgos** (Αρχή «**autorney**»), **john** (Αρχή «**adae**»), **kostas**(Αρχή «**police**») και αντίστοιχα passwords όπως αυτά έχουν καθοριστεί από τον διαχειριστή της εφαρμογής. Επίσης στον χρήστη giorgos έχουν δοθεί δικαιώματα πρόσβασης επιπέδου 4, στον χρήστη john επιπέδου 2 και στον χρήστη kostas δικαιώματα πρόσβασης επιπέδου 1, 2 και 3 (βλέπε κεφάλαιο 5).

Έτσι για τον χρήστη giorgos θα εμφανιστεί η παρακάτω σελίδα.



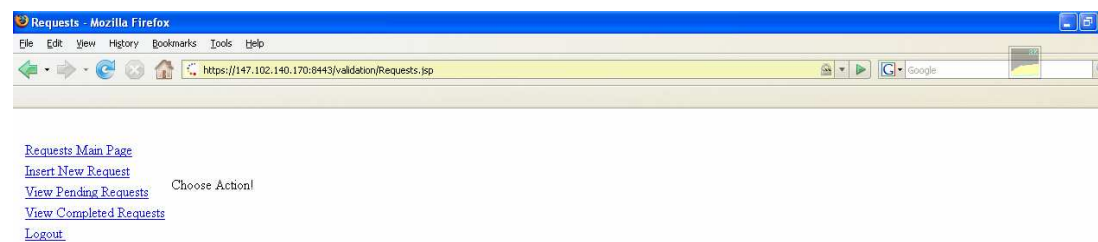
Σχήμα 6.6 Κύριο Μενού Εφαρμογής για πρόσβαση επιπέδου 4

Για τον χρήστη john θα εμφανιστεί η παρακάτω σελίδα.



Σχήμα 6.7 Κύριο Μενού Εφαρμογής για πρόσβαση επιπέδου 2

Τέλος η σελίδα που θα εμφανιστεί στον χρήστη kostas θα είναι η εξής.



Σχήμα 6.8 Κύριο Μενού Εφαρμογής για πρόσβαση επιπέδου 1 και 2 και 3

Παρά το γεγονός ότι το μενού είναι το ίδιο για τους χρήστες giorgos και kostas η διαφοροποίηση τους λόγω δικαιωμάτων πρόσβασης θα γίνει ορατή στα επόμενα βήματα.

6.3 Υπηρεσίες Εφαρμογής

6.3.1 Εισαγωγή Αιτήματος

Για την Εισαγωγή Αιτήματος ([Insert New Request](#)) ο χρήστη θα βρεθεί στην ακόλουθη σελίδα.

1. Χρήστης giorgos :

*Combinations that at least must be provided for a request are:
At any case Vindication of Request is necessary.
-First Name and Last Name
-Last Name and Birth Date
-Last Name and Country and City and Area or Street
-Taxation Number or Id Number or Phone Number

Σχήμα 6.9 Εισαγωγή αιτήματος για χρήστη με πρόσβαση επιπέδου 4

2. Χρήστης john :

Σύμφωνα με τα δικαιώματα πρόσβασης ο χρήστης αυτός δεν έχει δικαίωμα εισαγωγής νέου αιτήματος, η επιλογή αυτή δεν εμφανίζεται καθόλου στο κύριο μενού της εφαρμογής, όπως παρουσιάστηκε και πιο πάνω.

3. Χρήστης kostas :

Insert Request - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://147.102.140.170:8443/validation/RequestsInsert.jsp?username=hMIS/OUQCnWE1020n1viewA==

Requests Main Page
Insert New Request
View Pending Requests
View Completed Requests
Logout

First Name :		Father Name :		Last Name :		Birth Date :		
Taxation Number :		Id Number :		Mobile Phone Number :	6948654345			
Country :		City :						
Area :		Street :		Street Number :				
Period of phone data request								
Date from (dd/mm/yyyy) :				23/3/2006	Date to (dd/mm/yyyy) :			14/4/2006
Vindicate Request				Suspect of an armed robbery at National Bank of Greece.				

Insert Request Reset

*Combinations that at least must be provided for a request are:

At any case **Vindication of Request is necessary.**

-First Name and Last Name

-Last Name and Birth Date

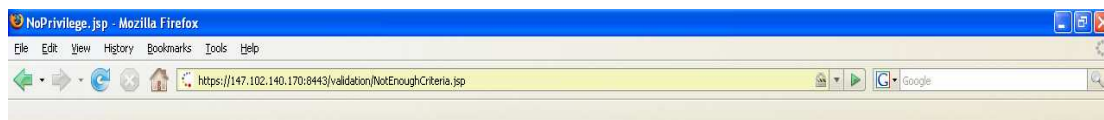
-Last Name and Country and City and Area or Street

-Taxation Number or Id Number or Phone Number



Σχήμα 6.10 Εισαγωγή αιτήματος για χρήστη με πρόσβαση επιπέδου 1 και 2 και 3

Με την επιτυχή κατάθεση του αιτήματος η εφαρμογή επιστρέφει στο κύριο μενού. Σε περίπτωση λάθους (για παράδειγμα αν δεν έχουν συμπληρωθεί τα απαραίτητα πεδία κάθε συνδυασμού) η σελίδα που εμφανίζεται στον χρήστη είναι η ακόλουθη.



You Must Give More Details About Your Query.

More than one person corresponds to that personal details.

[Return to Requests Insert Page](#)

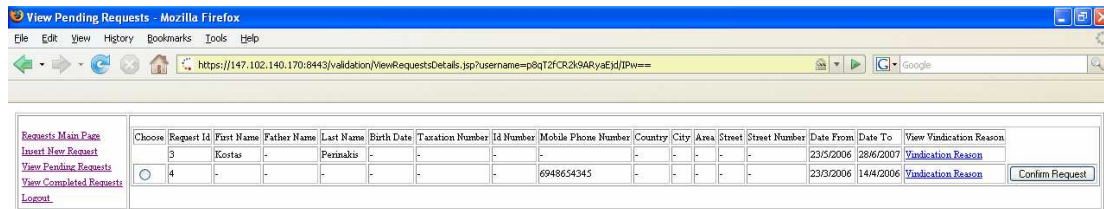
Σχήμα 6.11 Σφάλμα κατά την εισαγωγή αιτήματος

6.3.2 Αιτήματα που Εκκρεμούν και Έγκριση τους.

Για να δει ο κάθε χρήστης (που διαθέτει το ανάλογο δικαίωμα πρόσβασης) τα Αιτήματα που εκκρεμούν και για να προβεί σε έγκριση των όσων κρίνει ότι πρέπει να γίνει Άρση Απορρήτου Επικοινωνιών (Lawful Interception) θα πρέπει να μεταβεί στη

σελίδα της τρίτης επιλογής ([View Pending Requests](#)). Ο κάθε χρήστης τότε θα μεταβεί στην ακόλουθη σελίδα.

1. Χρήστης giorgos :

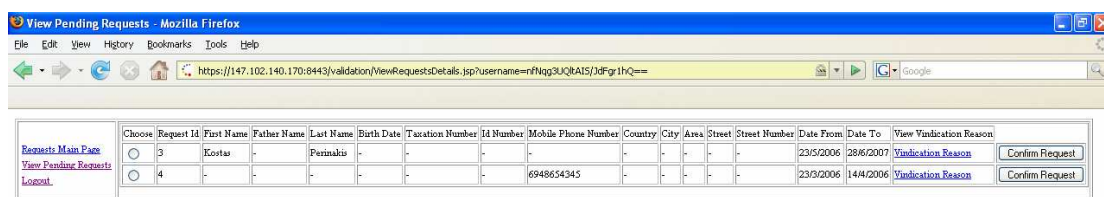


Choose	Request Id	First Name	Father Name	Last Name	Birth Date	Taxation Number	Id Number	Mobile Phone Number	Country	City	Area	Street	Street Number	Date From	Date To	View Vindication Reason
☒	3	Kostas	-	Perinakis	-	-	-	-	-	-	-	-	-	23/5/2006	28/6/2007	Vindication Reason
☐	4	-	-	-	-	-	-	6948654345	-	-	-	-	-	23/3/2006	14/4/2006	Vindication Reason

Σχήμα 6.12 Εκκρεμή αιτήματα και έγκριση για χρήστη επιπέδου 4

Παρατηρούμε ότι έχει το δικαίωμα να εγκρίνει το αίτημα του χρήστη kostas ενώ το αίτημα που είχε εισάγει ο ίδιος δεν μπορεί να εγκριθεί από τον ίδιο η από άλλον χρήστη που ανήκει στην ίδια Αρχή .

2. Χρήστης john :

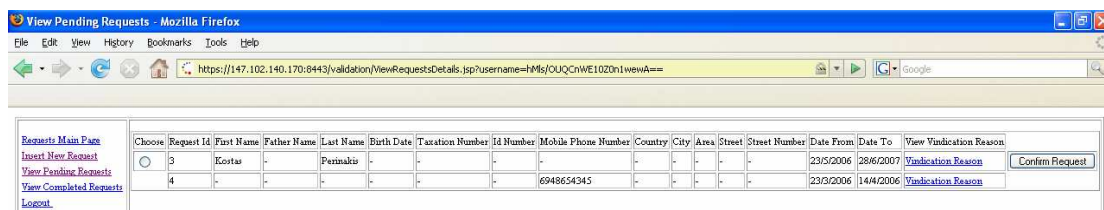


Choose	Request Id	First Name	Father Name	Last Name	Birth Date	Taxation Number	Id Number	Mobile Phone Number	Country	City	Area	Street	Street Number	Date From	Date To	View Vindication Reason
☒	3	Kostas	-	Perinakis	-	-	-	-	-	-	-	-	-	23/5/2006	28/6/2007	Vindication Reason
☐	4	-	-	-	-	-	-	6948654345	-	-	-	-	-	23/3/2006	14/4/2006	Vindication Reason

Σχήμα 6.13 Εκκρεμή αιτήματα και έγκριση για χρήστη επιπέδου 2

Παρατηρούμε ότι έχει το δικαίωμα να δει και να εγκρίνει τόσο το αίτημα του χρήστη kostas όσο και το αίτημα του χρήστη giorgos .

3. Χρήστης kostas :



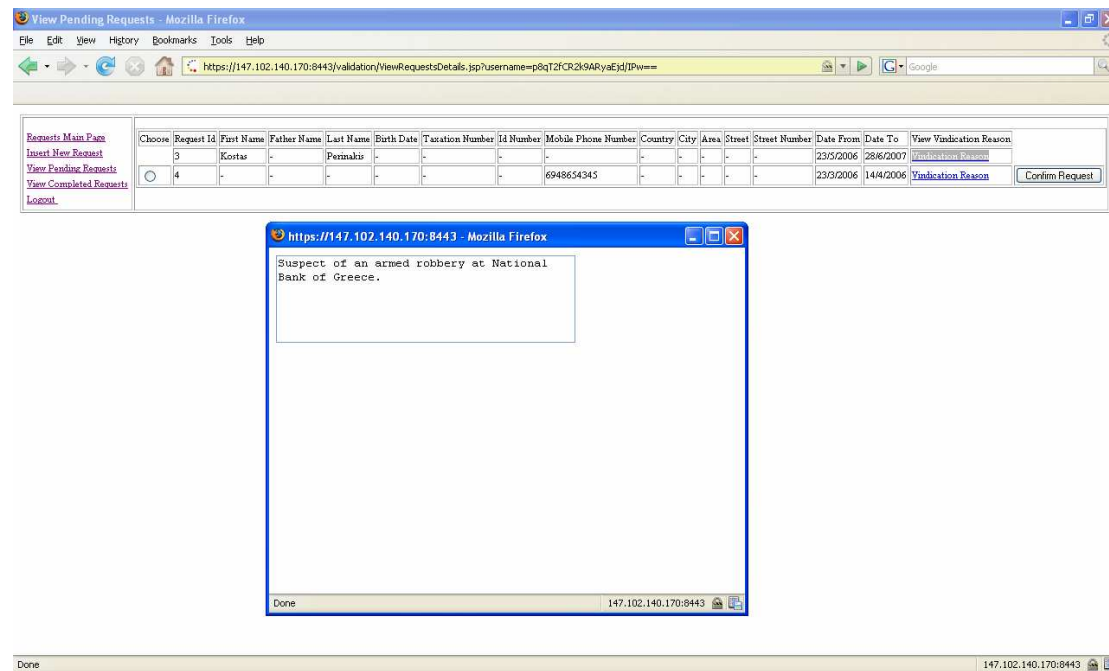
Choose	Request Id	First Name	Father Name	Last Name	Birth Date	Taxation Number	Id Number	Mobile Phone Number	Country	City	Area	Street	Street Number	Date From	Date To	View Vindication Reason
☒	3	Kostas	-	Perinakis	-	-	-	-	-	-	-	-	-	23/5/2006	28/6/2007	Vindication Reason
☐	4	-	-	-	-	-	-	6948654345	-	-	-	-	-	23/3/2006	14/4/2006	Vindication Reason

Σχήμα 6.14 Εκκρεμή αιτήματα και έγκριση για χρήστη επιπέδου 1 και 2 και 3

Παρατηρούμε ότι έχει το δικαίωμα να εγκρίνει το αίτημα του χρήστη giorgos ενώ το αίτημα που είχε εισάγει ο ίδιος δεν μπορεί να εγκριθεί από τον ίδιο η από άλλον χρήστη που ανήκει στην ίδια Αρχή .

Όταν κάποιος χρήστης θελήσει να κάνει έγκριση κάποιου αιτήματος τότε απλά επιλέγει το κύκλο που αντιστοιχεί στο ανάλογο αίτημα και μετά κάνει κλικ στο κουμπί «Confirm Request». Όπως είναι φυσικό μπορεί να δει το λόγο για τον οποίο έχει εισαχθεί το κάθε αίτημα από την επιλογή [Vindication Reason](#).

Για παράδειγμα για τον χρήστη giorgos έχουμε :



Σχήμα 6.15 Επισκόπηση λόγου εισαγωγής αιτήματος

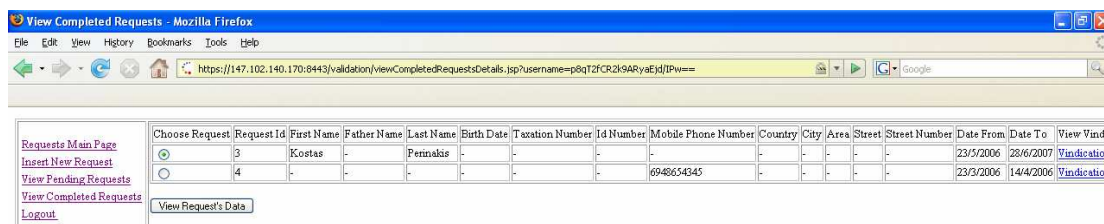
Και ας υποθέσουμε ότι προχωρεί στην έγκριση του εν λόγω αιτήματος. Αν δεν συμβεί κάποιο σφάλμα τότε θα μεταφερθεί στο κύριο μενού της εφαρμογής.

Στη συνέχεια και για το συγκεκριμένο παράδειγμα υποθέτουμε πως και οι άλλοι δύο χρήστες δίνουν έγκριση στα αιτήματα που εκκρεμούν. Οπότε τα δύο αυτά αιτήματα που εισήχθησαν από τους χρήστες giorgos και kostas δεν θα εμφανίζονται πλέον στη σελίδα [View Pending Requests](#) μιάς και δεν θα εκκρεμούν πλέον (θα έχουν πάρει έγκριση από μέλη από τις άλλες δύο Αρχές).

6.3.3 Αιτήματα που Έχουν Εγκριθεί.

Για να δει ο κάθε χρήστης (που διαθέτει το ανάλογο δικαίωμα πρόσβασης) τα Αιτήματα που έχουν εγκριθεί και για να προβεί στην εμφάνιση των στοιχείων των κλήσεων που συνδέονται με το άτομο που αφορά το εκάστοτε αίτημα θα πρέπει να μεταβεί στη σελίδα της τέταρτης επιλογής ([View Completed Requests](#)). Ο κάθε χρήστης τότε θα μεταβεί στην ακόλουθη σελίδα.

1. Χρήστης giorgos :



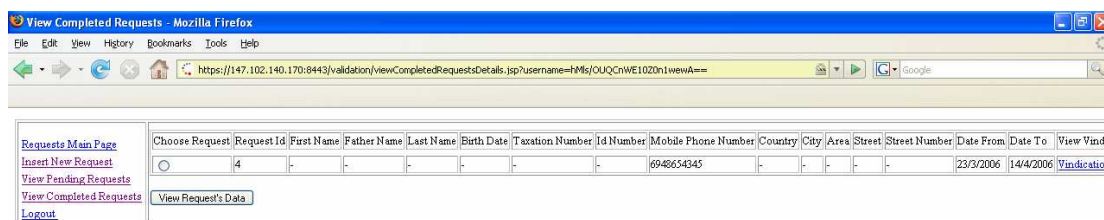
Σχήμα 6.16 Αιτήματα που έχουν εγκριθεί για χρήστη επιπέδου 4

Παρατηρούμε ότι έχει το δικαίωμα να δει κλήσεις που αφορούν Αιτήματα που έχουν ολοκληρωθεί ανεξάρτητα αν είναι Αίτημα που έχει εισάγει ο ίδιος αυτός χρήστης (*giorgos*) ή κάποιος άλλος που ανήκει σε άλλη Αρχή (*kostas*). Ο λόγος που συμβαίνει αυτό είναι γιατί όπως έχουμε αναφερθεί και πιο πάνω πώς ο χρήστης *giorgos* έχει δικαιώματα πρόσβασης επιπέδου 4, το οποίο τον καθιστά ικανό να βλέπει όλα τα ολοκληρωμένα Αιτήματα ανεξάρτητα χρήστη και Αρχή στην οποία ανήκει ο χρήστης που τα έχει αρχικά εισάγει.

2. Χρήστης john :

Σύμφωνα με τα δικαιώματα πρόσβασης ο χρήστης αυτός δεν έχει δικαίωμα να έχει πρόσβαση στα ολοκληρωμένα Αιτήματα, η επιλογή αυτή δεν εμφανίζεται καθόλου στο κύριο μενού της εφαρμογής, όπως παρουσιάστηκε και πιο πάνω.

3. Χρήστης kostas :



Σχήμα 6.17 Αιτήματα που έχουν εγκριθεί για χρήστη επιπέδου 1 και 2 και 3

Παρατηρούμε ότι έχει το δικαίωμα να δει κλήσεις που αφορούν Αιτήματα που έχουν ολοκληρωθεί και τα οποία έχει εισάγει ο ίδιος αυτός χρήστης (*kostas*). Ο λόγος που συμβαίνει αυτό είναι γιατί όπως έχουμε αναφερθεί και πιο πάνω πώς ο χρήστης *kostas* έχει δικαιώματα πρόσβασης επιπέδου 1, 2 & 3, το οποίο τον καθιστά ικανό να βλέπει όλα τα ολοκληρωμένα Αιτήματα που έχει εισάγει ο ίδιος.

Όταν κάποιος χρήστης θελήσει να έχει πρόσβαση στις κλήσεις που αφορούν κάποιο Αίτημα που έχει εγκριθεί (από δύο χρήστες που ανήκουν στις άλλες δύο αρχές) τότε απλά επιλέγει το κύκλο που αντιστοιχεί στο ανάλογο αίτημα και μετά κάνει κλικ στο κουμπί «View Request's Data». Όπως είναι φυσικό μπορεί να δει το λόγο για τον οποίο έχει εισαχθεί το κάθε αίτημα από την επιλογή [Vindication Reason](#).

Για παράδειγμα για τον χρήστη *giorgos* έχουμε για το πρώτο κατά σειρά Αίτημα :

Request # 3 : General Data - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://147.102.140.170:8443/validation/viewRequestGeneralData.jsp

Subscriber Data :

First Name :	Kostas	Father Name :	Efthimios	Last Name :	Perinakis
Birth Date :	-	Taxation Number :	275847362564	Id Number :	S389574
Country :	Greece	City :	Athens	Area :	Zografou
Street :	Esties NTUA	Street Number :	-		

Call Data for requested Period :

Call Id :	Caller number :	Callee Phone Number :	Call Date :	Call Start Time :	Call End Time :	Caller Location :	Callee Location :	Caller IMEI :	Caller IMSI :	Caller BTS Id :	Callee IMEI :
10	6973245323	6944444444	22/6/2006	8.26.04	8.31.35	-	Kalamata	-	-	-	56-456345-563556-S
24	6973454323	6944444444	30/6/2006	11.31.41	11.32.13	-	Volos	-	-	-	56-456345-563556-S
49	6944444444	6934563456	21/5/2007	4.21.36	4.27.30	Halkida	-	56-456345-563556-S	765837647562874	13245	-
52	6945234542	6944444444	10/3/2007	5.44.30	5.53.19	Preveza	Volos	83-453234-563344-2	234556778657789	40112	56-456345-563556-S
73	6944444444	6948654345	13/7/2006	18.16.11	18.18.14	Halkida	Ampelokipi	56-456345-563556-S	765837647562874	36089	34-674746-452344-8
108	6945234542	6944444444	23/5/2006	7.09.50	7.13.00	Halkida	Glyfada	83-453234-563344-2	234556778657789	8907	56-456345-563556-S
134	6944444444	6934563456	18/8/2006	11.43.04	11.50.30	Kerkira	-	56-456345-563556-S	765837647562874	4582	-
137	6946464321	6944444444	3/11/2006	4.05.28	4.07.00	Ampelokipi	Ekaterini	23-564676-123537-6	647826377567143	26963	56-456345-563556-S
179	6973454323	6944444444	22/6/2006	4.02.35	4.07.28	-	Tripoli	-	-	-	56-456345-563556-S
193	6944432532	6944444444	13/8/2006	16.51.20	16.54.17	Argos	Xanthi	35-345563-457543-1	345634245345356	40906	56-456345-563556-S
205	6944761245	6944444444	24/2/2007	23.28.25	23.34.32	Tripoli	Patra	65-434563-452345-2	234545667657856	51150	56-456345-563556-S
230	6944444444	6944761245	3/6/2006	19.46.16	19.55.55	Rodos	Athens	56-456345-563556-S	765837647562874	52221	65-434563-452345-2
247	6972345453	6944444444	27/9/2006	18.40.21	18.48.17	-	Patra	-	-	-	56-456345-563556-S
273	6948654345	6944444444	6/12/2006	22.14.18	22.23.02	Neos Kosmos	Halkidiki	34-674746-452344-8	134243565675889	10251	56-456345-563556-S
283	6973454323	6944444444	5/12/2006	4.18.27	4.22.15	-	Athens	-	-	-	56-456345-563556-S
294	6944444444	6973245323	17/12/2006	19.36.26	19.40.44	Halkida	-	56-456345-563556-S	765837647562874	4886	-
301	6944444444	6972345453	27/1/2007	5.54.37	6.04.01	Xanthi	-	56-456345-563556-S	765837647562874	16769	-
302	6944444444	6934523455	19/8/2006	21.04.03	21.06.33	Glyfada	-	56-456345-563556-S	765837647562874	59025	-
316	6972345453	6944444444	18/7/2006	7.56.08	7.58.40	-	Thessaloniki	-	-	-	56-456345-563556-S
351	6934523455	6944444444	9/6/2007	16.52.10	16.54.44	-	Krete	-	-	-	56-456345-563556-S
352	6945123767	6944444444	13/10/2006	8.35.40	8.40.24	Ekaterini	Volos	23-564573-456653-7	346576893231430	14587	56-456345-563556-S
384	6945341234	6944444444	11/1/2007	11.11.48	11.13.53	Argos	Glyfada	34-564534-563445-3	348745345879838	22052	56-456345-563556-S
393	6945341234	6944444444	7/8/2006	5.03.21	5.09.50	Ekaterini	Tripoli	34-564534-563445-3	348745345879838	56056	56-456345-563556-S

Requests Main Page
Insert New Request
View Pending Requests
View Completed Requests
Logout

Done 147.102.140.170:8443

Σχήμα 6.18 Επισκόπηση Αιτήματος

Request # 3 : General Data - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://147.102.140.170:8443/validation/viewRequestGeneralData.jsp

Perinakis
S389574 Phone Number : 6944444444
Zografou

	Call Start Time :	Call End Time :	Caller Location :	Callee Location :	Caller IMEI :	Caller IMSI :	Caller BTS Id :	Callee IMEI :	Callee IMSI :	Callee BTS Id :	Caller: Prepaid user:	Callee: Call Tansfer:	Callee: Call Tansfer to Number :
6	8.26.04	8.31.35	-	Kalamata	-	-	-	56-456345-563556-5	765837647562874	54728	-	-	-
6	11.31.41	11.32.13	-	Volos	-	-	-	56-456345-563556-5	765837647562874	50196	-	-	-
7	4.21.36	4.27.30	Halkida	-	56-456345-563556-5	765837647562874	13245	-	-	-	-	-	-
7	5.44.30	5.53.19	Preveza	Volos	83-453234-563344-2	234556778657789	40112	56-456345-563556-5	765837647562874	17369	-	-	-
6	18.16.11	18.18.14	Halkida	Ampelokipi	56-456345-563556-5	765837647562874	36089	34-674746-452344-8	134243565675889	2078	-	-	-
6	7.09.50	7.13.00	Halkida	Glyfada	83-453234-563344-2	234556778657789	8907	56-456345-563556-5	765837647562874	45525	-	-	-
6	11.43.04	11.50.30	Kerkira	-	56-456345-563556-5	765837647562874	4582	-	-	-	-	-	-
6	4.05.28	4.07.00	Ampelokipi	Ekaterini	23-564676-123537-6	647826377567143	26963	56-456345-563556-5	765837647562874	27357	-	-	-
6	4.02.35	4.07.28	-	Tripoli	-	-	-	56-456345-563556-5	765837647562874	51756	-	-	-
6	16.51.20	16.54.17	Argos	Xanthi	35-345563-457543-1	345634245345356	40906	56-456345-563556-5	765837647562874	46095	-	-	-
7	23.28.25	23.34.32	Tripoli	Patra	65-434563-452345-2	234545667657856	51150	56-456345-563556-5	765837647562874	660	-	-	-
	19.46.16	19.55.55	Rodos	Athens	56-456345-563556-5	765837647562874	52221	65-434563-452345-2	234545667657856	51242	-	-	-
6	18.40.21	18.48.17	-	Patra	-	-	-	56-456345-563556-5	765837647562874	12685	-	-	-
6	22.14.18	22.23.02	Neos Kosmos	Halkidiki	34-674746-452344-8	134243565675889	10251	56-456345-563556-5	765837647562874	57162	-	-	-
6	4.18.27	4.22.15	-	Athens	-	-	-	56-456345-563556-5	765837647562874	15680	-	-	-
06	19.36.26	19.40.44	Halkida	-	56-456345-563556-5	765837647562874	4886	-	-	-	-	-	-
7	5.54.37	6.04.01	Xanthi	-	56-456345-563556-5	765837647562874	16769	-	-	-	-	-	-
6	21.04.03	21.06.33	Glyfada	-	56-456345-563556-5	765837647562874	59025	-	-	-	-	-	-
6	7.56.08	7.58.40	-	Thessaloniki	-	-	-	56-456345-563556-5	765837647562874	49514	-	-	-
	16.52.10	16.54.44	-	Krete	-	-	-	56-456345-563556-5	765837647562874	23360	-	-	-
06	8.35.40	8.40.24	Ekaterini	Volos	23-564573-456653-7	346576893231430	14587	56-456345-563556-5	765837647562874	53661	-	-	-
7	11.11.48	11.13.53	Argos	Glyfada	34-564534-563445-3	348745345879838	22052	56-456345-563556-5	765837647562874	38629	-	-	-
	5.03.21	5.09.50	Ekaterini	Tripoli	34-564534-563445-3	348745345879838	56056	56-456345-563556-5	765837647562874	23669	-	-	-

Done 147.102.140.170:8443

Σχήμα 6.19 Επισκόπηση Αιτήματος

Request # 3 : General Data - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://147.102.140.170:8443/validation/viewRequestGeneralData.jsp

Google

247	6972345453	6944444444	27/9/2006	18.40.21	18.48.17	-	Patra	-	-	-	56-456345-563556-5
273	6948654345	6944444444	6/12/2006	22.14.18	22.23.02	Neos Kosmos	Halkidiki	34-674746-452344-8	134243565675889	10251	56-456345-563556-5
283	6973454323	6944444444	5/12/2006	4.18.27	4.22.15	-	Athens	-	-	-	56-456345-563556-5
294	6944444444	6973245323	17/12/2006	19.36.26	19.40.44	Halkida	-	56-456345-563556-5	765837647562874	4886	-
301	6944444444	6972345453	27/1/2007	5.54.37	6.04.01	Xanthi	-	56-456345-563556-5	765837647562874	16769	-
302	6944444444	6934523455	19/8/2006	21.04.03	21.06.33	Glyfada	-	56-456345-563556-5	765837647562874	59025	-
316	6972345453	6944444444	18/7/2006	7.56.08	7.58.40	-	Thessaloniki	-	-	-	56-456345-563556-5
351	6934523455	6944444444	9/6/2007	16.52.10	16.54.44	-	Krete	-	-	-	56-456345-563556-5
352	6945123767	6944444444	13/10/2006	8.35.40	8.40.24	Ekaterini	Volos	23-564573-456653-7	346576893231430	14587	56-456345-563556-5
384	6945341234	6944444444	11/1/2007	11.11.48	11.13.53	Argos	Glyfada	34-564534-563445-3	348745345879838	22052	56-456345-563556-5
393	6945341234	6944444444	7/8/2006	5.03.21	5.09.50	Ekaterini	Tripoli	34-564534-563445-3	348745345879838	56056	56-456345-563556-5
399	6947235421	6944444444	3/4/2007	15.16.53	15.21.15	Ekaterini	Peiraios	45-456743-546747-6	567564743353245	19914	56-456345-563556-5
421	6945341234	6944444444	4/3/2007	19.40.19	19.48.50	Tripoli	Xanthi	34-564534-563445-3	348745345879838	25051	56-456345-563556-5
432	6944444444	6946464321	5/6/2007	7.11.31	7.15.05	Kalamaki	Ioannina	56-456345-563556-5	765837647562874	5712	23-564676-123537-6
455	6944444444	6935732984	11/2/2007	2.02.44	2.03.27	Rodos	-	56-456345-563556-5	765837647562874	27719	-
466	6946464321	6944444444	29/4/2007	23.48.33	23.54.52	Ioannina	Ekaterini	23-564676-123537-6	647826377567143	41174	56-456345-563556-5
488	6948654345	6944444444	18/8/2006	14.45.26	14.47.28	Krete	Larissa	34-674746-452344-8	134243565675889	49236	56-456345-563556-5
530	6944444444	6947235421	27/2/2007	21.01.01	21.08.03	Patra	Halkidiki	56-456345-563556-5	765837647562874	32560	45-456743-546747-6
556	6934523455	6944444444	9/4/2007	6.27.49	6.28.38	-	Larissa	-	-	-	56-456345-563556-5
624	6944444444	6973454323	21/6/2007	4.11.09	4.13.35	Xanthi	-	56-456345-563556-5	765837647562874	402	-
732	6944444444	6935732984	27/1/2007	20.22.19	20.31.31	Krete	-	56-456345-563556-5	765837647562874	43627	-
768	6944124456	6944444444	29/9/2006	2.26.02	2.27.06	Kalamaki	Neos Kosmos	67-345534-245434-7	324564564767566	4643	56-456345-563556-5
784	6944444444	6944761245	17/7/2006	3.54.59	4.01.09	Rodos	Ioannina	56-456345-563556-5	765837647562874	46086	65-434563-452345-2
795	6935732984	6944444444	25/7/2006	10.03.00	10.03.49	-	Thessaloniki	-	-	-	56-456345-563556-5
805	6944444444	6947235421	12/9/2006	12.54.10	12.57.53	Halkida	Halkidiki	56-456345-563556-5	765837647562874	43254	45-456743-546747-6
810	6944444444	6948256732	12/4/2007	3.56.50	4.05.43	Exarheia	Athens	56-456345-563556-5	765837647562874	22559	45-645577-234546-7
907	6945234542	6944444444	13/12/2006	1.31.00	1.33.16	Halkida	Peiraios	83-453234-563344-2	234556778657789	38546	56-456345-563556-5
931	6944444444	6945341234	7/9/2006	2.39.14	2.41.31	Volos	Xanthi	56-456345-563556-5	765837647562874	23535	34-564534-563445-3
975	6947235421	6944444444	1/6/2006	1.19.00	1.19.49	Kalamaki	Ampelokipi	45-456743-546747-6	567564743353245	59608	56-456345-563556-5
998	6972345453	6944444444	13/6/2007	23.18.33	23.25.14	-	Volos	-	-	-	56-456345-563556-5

Done

147.102.140.170:8443

Σχήμα 6.20 Επισκόπηση Αιτήματος

Request # 3 : General Data - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://147.102.140.170:8443/validation/viewRequestGeneralData.jsp

6	18.40.21	18.48.17	-	Patra	-	-	-	56-456345-563556-5	765837647562874	12685	-	-	-
6	22.14.18	22.23.02	Neos Kosmos	Halkidiki	34-674746-452344-8	134243565675889	10251	56-456345-563556-5	765837647562874	57162	-	-	-
6	4.18.27	4.22.15	-	Athens	-	-	-	56-456345-563556-5	765837647562874	15680	-	-	-
06	19.36.26	19.40.44	Halkida	-	56-456345-563556-5	765837647562874	4886	-	-	-	-	-	-
7	5.54.37	6.04.01	Xanthi	-	56-456345-563556-5	765837647562874	16769	-	-	-	-	-	-
6	21.04.03	21.06.33	Glyfada	-	56-456345-563556-5	765837647562874	59025	-	-	-	-	-	-
6	7.56.08	7.58.40	-	Thessaloniki	-	-	-	56-456345-563556-5	765837647562874	49514	-	-	-
	16.52.10	16.54.44	-	Krete	-	-	-	56-456345-563556-5	765837647562874	23360	-	-	-
06	8.35.40	8.40.24	Ekaterini	Volos	23-564573-456653-7	346576893231430	14587	56-456345-563556-5	765837647562874	53661	-	-	-
7	11.11.48	11.13.53	Argos	Glyfada	34-564534-563445-3	348745345879838	22052	56-456345-563556-5	765837647562874	38629	-	-	-
	5.03.21	5.09.50	Ekaterini	Tripoli	34-564534-563445-3	348745345879838	56056	56-456345-563556-5	765837647562874	23669	-	-	-
	15.16.53	15.21.15	Ekaterini	Peiramus	45-456743-546747-6	567564743353245	19914	56-456345-563556-5	765837647562874	35939	-	-	-
	19.40.19	19.48.50	Tripoli	Xanthi	34-564534-563445-3	348745345879838	25051	56-456345-563556-5	765837647562874	11434	-	-	-
	7.11.31	7.15.05	Kalamaki	Ioannina	56-456345-563556-5	765837647562874	5712	23-564676-123537-6	647826377567143	1581	-	-	-
7	2.02.44	2.03.27	Rodos	-	56-456345-563556-5	765837647562874	27719	-	-	-	-	-	-
7	23.48.33	23.54.52	Ioannina	Ekaterini	23-564676-123537-6	647826377567143	41174	56-456345-563556-5	765837647562874	31971	-	-	-
6	14.45.26	14.47.28	Krete	Larissa	34-674746-452344-8	134243565675889	49236	56-456345-563556-5	765837647562874	28588	-	-	-
7	21.01.01	21.08.03	Patra	Halkidiki	56-456345-563556-5	765837647562874	32560	45-456743-546747-6	567564743353245	46664	-	-	-
	6.27.49	6.28.38	-	Larissa	-	-	-	56-456345-563556-5	765837647562874	22802	-	-	-
7	4.11.09	4.13.35	Xanthi	-	56-456345-563556-5	765837647562874	402	-	-	-	-	-	-
7	20.22.19	20.31.31	Krete	-	56-456345-563556-5	765837647562874	43627	-	-	-	-	-	-
6	2.26.02	2.27.06	Kalamaki	Neos Kosmos	67-345534-245434-7	324564564767566	4643	56-456345-563556-5	765837647562874	13604	-	-	-
6	3.54.59	4.01.09	Rodos	Ioannina	56-456345-563556-5	765837647562874	46086	65-434563-452345-2	234545667657856	41209	-	-	-
6	10.03.00	10.03.49	-	Thessaloniki	-	-	-	56-456345-563556-5	765837647562874	47907	-	-	-
6	12.54.10	12.57.53	Halkida	Halkidiki	56-456345-563556-5	765837647562874	43254	45-456743-546747-6	567564743353245	28429	-	-	-
7	3.56.50	4.05.43	Exarheia	Athens	56-456345-563556-5	765837647562874	22559	45-645577-234546-7	234345457676768	3451	-	-	-
06	1.31.00	1.33.16	Halkida	Peiramus	83-453234-563344-2	234556778657789	38546	56-456345-563556-5	765837647562874	40704	-	-	-
	2.39.14	2.41.31	Volos	Xanthi	56-456345-563556-5	765837647562874	23535	34-564534-563445-3	348745345879838	39677	-	-	-
	1.19.00	1.19.49	Kalamaki	Ampelokipi	45-456743-546747-6	567564743353245	59608	56-456345-563556-5	765837647562874	52616	-	-	-
7	23.18.33	23.25.14	-	Volos	-	-	-	56-456345-563556-5	765837647562874	43529	-	-	-

Done 147.102.140.170:8443

Σχήμα 6.21 Επισκόπηση Αιτήματος

7

7 Βιβλιογραφία

Υλοποίηση κώδικα Java

- [1] Sun - <http://java.sun.com>
- [2] Tutorial for building J2EE Applications using Eclipse, Lomboz, Jboss and Xdoclet.- <http://www.tusc.com.au/tutorial/html/>
- [3] O'Reilly – <http://www.oreilly.com>
- [4] Learning Java - <http://www.java2s.com/>
- [5] Code Examples - <http://exampledepot.com/egs/>
- [6] Java EJB Tutorials - <http://www.tutorialized.com/tutorials/Java/EJB/1>
- [7] Introduction to EJB - http://www.techseasaw.com/articles/11111/EJB_part1.htm
- [8] Building your first Enterprise JavaBean - <http://www.stardeveloper.com/articles/display.html?article=2001112801&page=1/>
- [9] Java Tips -- Java, Java, and more Java - <http://www.java-tips.org/>
- [10] Eclipse - <http://eclipse-wiki.info/>
- [11] Mastering Enterprise JavaBeans and the Java 2 Platform, Enterprise Edition, Ed Roman
- [12] Javapedia - <http://wiki.java.net/bin/view/Javapedia/>

Application Server

- [13] JBoss - <http://www.jboss.org>
- [14] JBoss Eclipse IDE Tutorial - <http://docs.jboss.com/>

[15] jboss.org: community driven - <http://labs.jboss.com/jbosside/>

[16] Enabling SSL on JBoss -
http://www.theserverside.com/discussions/thread.tss?thread_id=22243

Ψηφιακά Πιστοποιητικά και Ψηφιακές Υπογραφές

[17] Digital Signatures - <http://www.exampledepot.com/egs/java.security/pkg.html>

[18] The X509Certificate - <http://www.java2s.com/Code/JavaAPI/java.security.cert/>

[19] Part 2 of the Security and Trust Services API -
<http://developers.sun.com/techtopics/mobility/apis/articles/satsa2/index.html#managing>

[20] keytool-Key and Certificate Management Tool -
<http://java.sun.com/j2se/1.3/docs/tooldocs/win32/keytool.html/>

[21] Verisign Repository, Introduction to Digital Certificates -
<http://www.verisign.com.au/repository/tutorial/digital/intro1.shtml>

Βάση Δεδομένων

[22] Database System Concepts, Silberschatz – Korth – Sudarshan

[23] MySQL Documentation - <http://dev.mysql.com/doc/>

[24] MySQL Basics -- A Helpful MySQL Tutorial -
<http://www.analysisandsolutions.com/code/mysql-tutorial.htm>

Unified Modeling Language (UML)

[25] UML Object Oriented Software Engineering – Using Uml, Patterns and Java, B. Bruegge – A. H. Dutoit

[26] The Unified Modeling Language User Guide, Grady Booch – James Rumbaugh – Ivar Jacobson

[27] The Unified Modeling Language Reference Manual, James Rumbaugh – Ivar Jacobson – Grady Booch.

[28] Αντικειμενοστρεφής σχεδιασμός με UML -
<http://www.spinellis.gr/c3/uml/index.htm>

[29] Unified Modeling Language Tutorial -
http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML_tutorial/index.htm