



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

**Πρότυπη Εφαρμογή Διαλειτουργικότητας Για Φορητές
Συσκευές**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΩΝΣΤΑΝΤΙΝΟΥ Α. ΧΡΗΣΤΙΔΗ

Επιβλέπων : Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2006

Η σελίδα αυτή είναι σκόπιμα λευκή



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Πρότυπη Εφαρμογή Διαλειτουργικότητας Για Φορητές Συσκευές

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΩΝΣΤΑΝΤΙΝΟΥ Α. ΧΡΗΣΤΙΔΗ

Επιβλέπων : Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 3^η Νοεμβρίου 2006

(Υπογραφή)

.....

Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....

Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....

Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2006

(Υπογραφή)

.....

ΚΩΝΣΤΑΝΤΙΝΟΣ Α. ΧΡΗΣΤΙΔΗΣ

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

© 2006 – All rights reserved

Η σελίδα αυτή είναι σκόπιμα λευκή

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον υποψήφιο διδάκτορα Σπύρο Αθανασίου για την υποστήριξη και την συνεργασία κατά την εκπόνηση της παρούσας διπλωματικής εργασίας. Επίσης θα ήθελα να ευχαριστήσω τους καθηγητές Ιωάννη Βασιλείου και Τιμολέων Σελλή για την ανάληψη της εποπτείας της εργασίας και τους υπευθύνους του Εργαστηρίου Βάσεων Γνώσεων και Δεδομένων για την βοήθειά τους.

Τέλος θα ήθελα να ευχαριστήσω τους γονείς και τους φίλους μου για την συμπαράσταση και την κατανόησή τους.

Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας είναι η σχεδίαση, ανάπτυξη και υλοποίηση ενός λογισμικού εφαρμογών που να βασίζεται στην τεχνολογία των δικτυακών υπηρεσιών και να καθιστά εφικτή την ανταλλαγή πληροφοριών μεταξύ ασύρματων φορητών συσκευών και ενός κεντρικού εξυπηρετητή. Στόχος επομένως είναι ο έλεγχος της εφικτότητας και απόδοσης της επικοινωνίας μεταξύ συστημάτων που βασίζονται σε διαφορετικά λογισμικά περιβάλλοντα και πολλαπλά τηλεπικοινωνιακά μέσα για την παράδοση των υπηρεσιών προς τους χρήστες.

Συγκεκριμένα, το σύστημα που αναπτύχθηκε στο πλαίσιο της διπλωματικής, υλοποιεί με εναλλακτικό τρόπο τις διαδικασίες που απαρτίζουν ένα ολοκληρωμένο σύστημα ελεγχόμενης στάθμευσης. Η προσέγγιση που προτείνεται διαφοροποιείται σημαντικά από τα σημερινά μονολιθικά και άκαμπτα συστήματα, όπου απαιτούν την πλήρη ομοιογένεια σε επίπεδο πληροφοριακών υποδομών, ενώ θέτουν σημαντικούς περιορισμούς στην επιλογή των τηλεπικοινωνιακών διαύλων. Για να πραγματοποιηθεί αυτό το έργο απαιτούνται δυο ξεχωριστά συστήματα λογισμικού, ένα για τον πελάτη (client) και ένα για τον εξυπηρετητή (server). Ο πελάτης είναι υλοποίηση μιας εφαρμογής J2ME (Java 2 Micro Edition) που έχει όσο το δυνατόν μεγαλύτερη ταχύτητα και συμβατότητα με πλήθος μοντέλων κινητών και φορητών ψηφιακών συσκευών. Ο εξυπηρετητής είναι υλοποιημένος σε πλατφόρμα .NET και συνδέεται με μια σχεσιακή βάση δεδομένων.

Η προτεινόμενη μεθοδολογία και τρόπος ολοκλήρωσης των υποσυστημάτων μπορεί να γίνει οδηγός για αντίστοιχα συστήματα στα οποία στόχος είναι η κάλυψη των αναγκών των χρηστών με πλήρη διαλειτουργικότητα, επεκτασιμότητα και ευελιξία, ανεξάρτητα των χαρακτηριστικών της συσκευής των χρηστών (μέγεθος, πόροι, αυτονομία) και του μέσου που επιλέγεται για την επικοινωνία με το κεντρικό σύστημα.

Λέξεις Κλειδιά: <<δικτυακές υπηρεσίες, διαλειτουργικότητα, υπολογιστής τσέπης, XML, .NET, JAVA, midlet, ελεγχόμενη στάθμευση>>

Η σελίδα αυτή είναι σκόπιμα λευκή

Abstract

The scope of this thesis is the design, development and implementation of a software system based on web services technology, enabling the exchange of information between wireless portable devices and a central server, while also examining the feasibility and performance of the communication between systems based on different software environments and using multiple telecommunication links for the delivery of services.

Specifically, the system developed during this thesis is an alternative implementation of the procedures involved in a full managed parking system. The approach proposed is essentially differentiated from today's monolithic and rigid systems, which require strict coherence in informational infrastructures and substantially constrain the choice of communication links. For the actualization of this project two different software systems are required; a client and a server. The client software is an application developed in J2ME (Java 2 Micro Edition) and aims to maximize speed and compatibility with the existing range of mobile phones and portable digital devices. The server is implemented on the .NET framework and connected to a relational database.

The proposed methodology and subsystem integration can be a guideline for similar systems attempting to cover similar user needs, while also featuring interoperability, extendibility and flexibility, independently of the user's device specifications (size, resources, battery run-time), and the medium chosen that provides connectivity to the central system.

Keywords: <<web services, interoperability, Personal Digital Assistant, XML, .NET, JAVA, midlet, managed parking >>

Η σελίδα αυτή είναι σκόπιμα λευκή.

Πίνακας περιεχομένων

1	Εισαγωγή.....	1
1.1	Αντικείμενο της διπλωματικής	1
1.2	Οργάνωση του τόμου	2
2	Περιγραφή Θέματος	5
2.1	Τεχνολογικά Πρότυπα	5
2.1.1	Service-oriented (υπηρεσιοστραφής) αρχιτεκτονική.....	5
2.1.2	Δικτυακές Υπηρεσίες (Web Services).....	6
2.1.2.1	XML.....	6
2.1.2.2	SOAP	7
2.1.2.3	Λειτουργία Δικτυακών Υπηρεσιών	7
2.1.3	Java 2 Micro Edition.....	8
2.1.4	Σχεσιακές Βάσεις -Γλώσσα SQL.....	9
2.1.4.1	Σχεσιακές Βάσεις.....	9
2.1.4.2	MS SQL Server.....	10
2.1.5	.NET Framework	10
2.1.5.1	Windows Services.....	11
2.1.5.2	ASP .NET	12
2.2	Στόχος του Συστήματος	12
2.2.1	Παρούσα κατάσταση συστημάτων στάθμευσης.....	12
2.2.2	Γενική περιγραφή του συστήματος	13
3	Ανάλυση και Σχεδίαση	15
3.1	Έγγραφο Προσδιορισμού Απαιτήσεων από το Λογισμικό	15
3.1.1	Εισαγωγή	15
3.1.1.1	Σκοπός.....	15
3.1.1.2	Εμβέλεια	16
3.1.1.3	Ορισμοί, Ακρωνύμια, Συντομογραφίες	16
3.1.1.4	Βιβλιογραφία	16

3.1.2	Απαιτήσεις.....	16
3.1.2.1	Προδιαγραφές περιπτώσεων χρήσης	17
3.1.2.2	Λειτουργικότητα	17
3.1.2.3	Απαιτήσεις χρήσης.....	17
3.1.2.4	Απαιτήσεις αξιοπιστίας.....	18
3.1.2.5	Απαιτήσεις ασφάλειας	18
3.1.2.6	Απαιτήσεις επιδόσεων	18
3.1.2.7	Πρότυπα	18
3.1.3	Εργασίες προς αυτοματοποίηση	18
3.1.3.1	Σύνοψη.....	18
3.1.4	Περιπτώσεις χρήσης	22
3.1.4.1	Εισαγωγή	22
3.1.4.2	Έναρξη και λήξη στάθμευσης για εγγεγραμμένο χρήστη.....	22
3.1.4.3	Στάθμευση σταθερού χρόνου για εγγεγραμμένο χρήστη.....	25
3.1.4.4	Έναρξη και λήξη στάθμευσης για μη εγγεγραμμένο χρήστη	27
3.1.4.5	Στάθμευση σταθερού χρόνου για μη εγγεγραμμένο χρήστη	30
3.1.4.6	Έλεγχος οχήματος, καταγραφή και εκτύπωση παράβασης	31
3.1.4.7	Εγγραφή χρήστη	34
3.1.4.8	Διαχείριση οχημάτων – χρεώσεων από το χρήστη	35
3.1.4.9	Έλεγχος συστήματος.....	38
3.1.4.10	Παρακολούθηση στατιστικών.....	40
3.2	Έγγραφο Περιγραφής Αρχιτεκτονικού Σχεδίου	40
3.2.1	Εισαγωγή	40
3.2.1.1	Σκοπός.....	40
3.2.1.2	Εμβέλεια	40
3.2.1.3	Ορισμοί, ακρωνύμια, συντομογραφίες	41
3.2.1.4	Βιβλιογραφία	41
3.2.2	Αρχιτεκτονικές Αποφάσεις.....	41
3.2.2.1	Πλατφόρμα	41
3.2.2.2	Γλώσσες – σχολή προγραμματισμού	41
3.2.2.3	Κωδική γλώσσα	42

3.2.3	Αρχιτεκτονικές Όψεις.....	42
3.2.3.1	Περιβαλλοντική / Υπηρεσιακή	42
3.2.3.2	Πλατφορμική	44
3.2.3.3	Χρηστική.....	44
3.2.3.4	Σύστημα Πελάτη Ελεγκτή	58
3.2.3.5	Σύστημα Δικτυακού Τόπου	61
3.2.3.6	Δομική.....	66
3.3	Έγγραφο Περιγραφής Λεπτομερούς Σχεδίου	68
3.3.1	Εισαγωγή	68
3.3.1.1	Σκοπός.....	68
3.3.1.2	Εμβέλεια	68
3.3.1.3	Ορισμοί, ακρωνύμια, συντομογραφίες	68
3.3.1.4	Βιβλιογραφία	69
3.3.2	Σχεδιαστικές Αποφάσεις.....	69
3.3.3	Σχεδιαστικά Μορφήματα.....	69
3.3.4	Σχεδιαστικές Όψεις.....	69
3.3.4.1	Αποσυνθετική Όψη.....	69
3.3.4.2	Δεδομενική Όψη	73
3.3.4.3	Συνεργατική Όψη.....	83
3.3.4.4	Διαπροσωπική Όψη	91
4	Υλοποίηση	93
4.1	Πλατφόρμες και Προγραμματιστικά Εργαλεία	93
4.1.1	Microsoft Visual Studio .NET	93
4.1.2	NetBeans IDE	94
4.1.3	Microsoft SQL Server.....	94
4.2	Λεπτομέρειες Υλοποίησης.....	95
4.2.1	Βιβλιοθήκες.....	95
4.2.1.1	kXML.....	95
4.2.1.2	SMSLib.....	95
4.2.2	Υλοποίηση των υποσυστημάτων.....	96

4.2.2.1	Εξυπηρετητής.....	96
4.2.2.2	Εφαρμογή Σύντομων Μηνυμάτων.....	101
4.2.2.3	Πελάτης.....	102
5	Έλεγχος.....	107
5.1	Οδηγός Εγκατάστασης.....	107
5.1.1	Εγκατάσταση του εξυπηρετητή βάσης δεδομένων.....	107
5.1.2	Εγκατάσταση του τμήματος των δικτυακών υπηρεσιών και των δικτυακών τόπων....	108
5.1.3	Εγκατάσταση του διαχειριστή σύντομων μηνυμάτων.....	108
5.1.4	Εγκατάσταση της εφαρμογής πελάτη (χρήστη/ ελεγκτή).....	109
5.2	Οδηγός Χρήσης.....	109
5.2.1	Χρήση της ιστοσελίδας του συστήματος.....	109
5.2.2	Στάθμευση μέσω MIDlet.....	110
5.2.2.1	Εγγεγραμμένος χρήστης	110
5.2.2.2	Μη εγγεγραμμένος χρήστης.....	112
5.2.3	Στάθμευση μέσω σύντομων μηνυμάτων	113
5.2.3.1	Εγγεγραμμένος χρήστης	113
5.2.3.2	Μη εγγεγραμμένος χρήστης.....	114
6	Επίλογος	115
6.1	Σύνοψη και συμπεράσματα.....	115
6.1.1	Σύνοψη	115
6.1.2	Συμπεράσματα	116
6.2	Μελλοντικές επεκτάσεις	117
7	Βιβλιογραφία.....	119

1

Εισαγωγή

Τα τελευταία χρόνια, η εφαρμογή στην καθημερινότητα σύγχρονων τεχνολογιών πληροφορικής και τηλεπικοινωνιών, συναντά μεγάλες προκλήσεις και ειδικότερα στο πεδίο της εφαρμογής τους σε ασύρματες, φορητές συσκευές περιορισμένων δυνατοτήτων, οι οποίες όμως είναι εξαιρετικά δημοφιλείς μεταξύ των χρηστών. Συγκεκριμένα, οι ασύρματες φορητές συσκευές αντιμετωπίζουν μεγάλες δυσκολίες που αφορούν όχι τόσο την πολυπλοκότητα των προβλημάτων που καλούνται να λύσουν, αλλά την αδυναμία διαλειτουργικότητας και τη δυσκολία επέκτασης και συντήρησης του λογισμικού εφαρμογών που έχει αναπτυχθεί για αυτές.

Σημαντική βοήθεια σε αυτή την κατεύθυνση αποτελεί η χρήση δικτυακών υπηρεσιών, η υιοθέτηση γλωσσών προγραμματισμού που παρέχουν ανεξαρτησία από την υποκείμενη πληροφοριακή υποδομή, καθώς και οποιαδήποτε άλλη τεχνολογία παρέχει ένα επίπεδο αφαίρεσης από τα ειδικά στοιχεία της υλοποίησης εφαρμογών σε συσκευές διαφορετικών κατασκευαστών.

1.1 Αντικείμενο της διπλωματικής

Αντικείμενο της παρούσας διπλωματικής είναι η σχεδίαση και ανάπτυξη ενός συστήματος ελεγχόμενης στάθμευσης για τους δήμους της χώρας που θα χρησιμοποιεί τις δικτυακές υπηρεσίες και θα επιτρέπει την επικοινωνία μεταξύ ασύρματων φορητών συσκευών και ενός κεντρικού εξυπηρετητή.

Στα βασικά χαρακτηριστικά του προτεινόμενου συστήματος, συγκαταλέγονται τα ακόλουθα:

- **Αγορά χρόνου στάθμευσης μέσω απλών κινητών τηλεφώνων.** Ο χρήστης θα μπορεί χρησιμοποιώντας απλά και κατανοητά βήματα για να δεσμεύσει χρόνο στάθμευσης στην περιοχή ή και στη θέση που επιθυμεί. Θα υπάρχει μια πληθώρα διαφορετικών μεθόδων πληρωμής οι οποίες θα καλύπτουν πλήρως τις ετερόκλητες απαιτήσεις των διαφόρων χρηστών. Ακόμη θα δίνεται δυνατότητα για πλήρη έλεγχο στο λογαριασμό του, των οχημάτων που σταθμεύει, των χρεώσεων που υφίσταται καθώς και των πιθανών προστίμων. Με τον τρόπο αυτό εξασφαλίζεται η διαφάνεια των χρεώσεων και τελικά η εμπιστοσύνη των χρηστών στο σύστημα.
- **Ολοκλήρωση των ελέγχων από την Δημοτική Αστυνομία.** Η αρχή επίβλεψης του συγκεκριμένου συστήματος, θα έχει τη δυνατότητα να ελέγχει εύκολα και γρήγορα τα σταθμευμένα οχήματα. Ακόμη, θα δίνεται η δυνατότητα όχι μόνο για εύκολη συντήρηση του συστήματος, αλλά και για εμφάνιση παρατηρήσεων και εξαγωγή συμπερασμάτων που θα μπορούσαν να βελτιώσουν τις παραμέτρους του συστήματος.
- **Χαμηλό κόστος κτήσης και λειτουργίας.** Στα θετικά του συστήματος περιλαμβάνεται το χαμηλό κόστος κτήσης και λειτουργίας και η άμεση εισροή εσόδων στον ιδιοκτήτη του συστήματος. Το σύστημα θα παρέχει ευελιξία στους τρόπους χρήσης που θα υποστηρίζει, καθώς και σε πιθανές επεκτάσεις ή τροποποιήσεις τους.
- **Ευκολία στη χρήση και την εκμάθηση από όλους τους χρήστες του.** Για το λόγο αυτό δίνεται ιδιαίτερη έμφαση στην ομοιογένεια, την εικαστική επιμέλεια και την αποδοτικότητα της διεπαφής των χρηστών, ώστε να μπορούν να ολοκληρώσουν τις συχνότερες συναλλαγές τους με το μικρότερο δυνατό πλήθος βημάτων.
- **Το σύστημα θα πρέπει να είναι σε θέση να καλύπτει την ανάγκη των αστικών κέντρων της χώρας για ένα ολοκληρωμένο, φτηνό και άρτιο τεχνολογικά, σύστημα ελεγχόμενης στάθμευσης.**

1.2 Οργάνωση του τόμου

Στο **πρώτο κεφάλαιο** γίνεται μια εισαγωγή και περιγράφεται το αντικείμενο της διπλωματικής καθώς και η οργάνωση του τόμου.

Στο **δεύτερο κεφάλαιο** γίνεται ιδιαίτερη μνεία στις τεχνολογίες οι οποίες χρησιμοποιούνται στη συνέχεια. Δίνεται με αυτόν τον τρόπο μια εισαγωγή στο περιβάλλον και τα δομικά συστατικά της

διπλωματικής εργασίας. Ακόμη, γίνεται σαφής ο στόχος του συστήματος που αναπτύχθηκε και με ποιον τρόπο επιχειρήθηκε να καταλήξουμε σε αυτόν τον στόχο.

Το **τρίτο κεφάλαιο** περιλαμβάνει τη διαδικασία σύλληψης και σχεδίασης, όπως αυτή προβλέπεται από τις σύγχρονες μεθόδους ανάπτυξης λογισμικού. Αναλύονται οι απαιτήσεις από το σύστημα και ακολουθεί μια πλήρης περιγραφή τόσο του αρχιτεκτονικού όσο και του λεπτομερούς σχεδίου.

Στο **τέταρτο κεφάλαιο** αναλύεται η υλοποίηση του συστήματος όπως αυτή γίνεται σε διαφορετικές πλατφόρμες. Επιπροσθέτως, περιλαμβάνεται σύντομη ανάλυση των διαφόρων βιβλιοθηκών που χρησιμοποιήθηκαν.

Στο **πέμπτο κεφάλαιο** δίνεται ένας σύντομος οδηγός εγκατάστασης του συστήματος καθώς και μια αναλυτική παρουσίαση όλων των διαδικασιών ελέγχου που εφαρμόζονται στο σύστημα.

Το **έκτο κεφάλαιο** περιέχει μια σύνοψη της διπλωματικής εργασίας και τα συμπεράσματα που εξάχθηκαν με την ολοκλήρωση της. Επίσης προτείνονται μελλοντικές επεκτάσεις του συστήματος.

Στο **έβδομο κεφάλαιο** περιέχεται η βιβλιογραφία που χρησιμοποιήθηκε.

2

Περιγραφή Θέματος

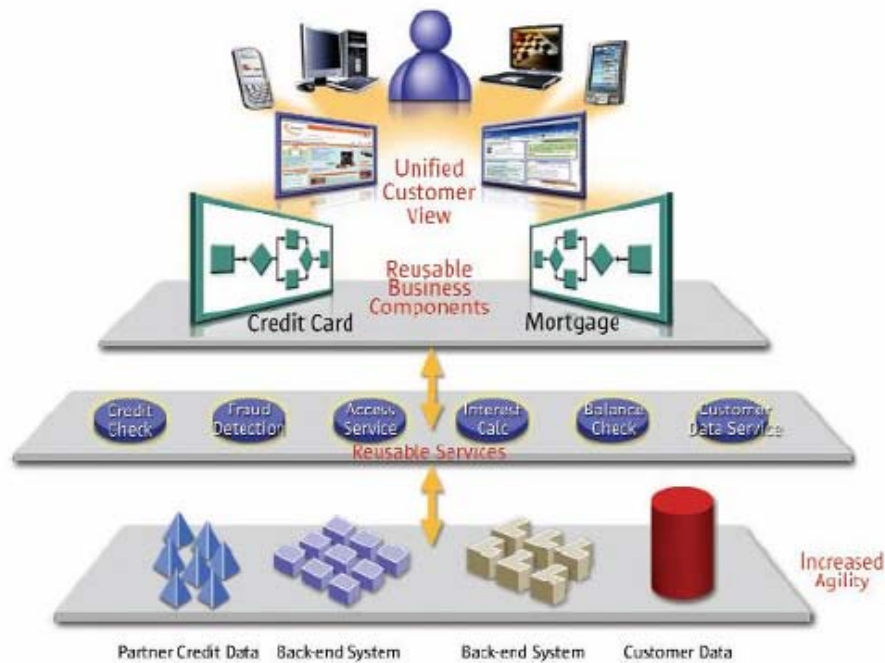
2.1 Τεχνολογικά Πρότυπα

2.1.1 Service-oriented (υπηρεσιοστραφής) αρχιτεκτονική

Η υπηρεσιοστραφής αρχιτεκτονική (Service-Oriented Architecture, SOA) είναι μια σχεδιαστική προσέγγιση συστημάτων η οποία προάγει την ασύνδετη (“loose”) επικοινωνία μεταξύ δύο τμημάτων λογισμικού. Αναλυτικά εκ των δυο επικοινωνούντων τμημάτων, το ένα παράγει μια υπηρεσία ενώ το δεύτερο την καταναλώνει.

Η συγκεκριμένη τεχνολογία εφαρμόζεται σε μια πληθώρα περιπτώσεων καθώς προσφέρει ανεξαρτησία από την υλοποίηση σε ένα δικτυακό περιβάλλον. Για να προωθηθεί η μέγιστη διαλειτουργικότητα μεταξύ των τμημάτων λογισμικού, έχουν αναπτυχθεί ορισμένα standards για αυτή την επικοινωνία. Παραδείγματα είναι τα RPC, DCOM, ORB και WSDL.

Στα θετικά της συγκεκριμένης αρχιτεκτονικής συγκαταλέγεται η αυξημένη διαλειτουργικότητα μεταξύ εργαλείων που έχουν αναπτυχθεί σε διαφορετικές πλατφόρμες. Ακόμη, σημαντική είναι η δυνατότητα για επαναχρησιμοποίηση και αφαιρετική σχεδίαση των εμπλεκόμενων μερών. Ουσιαστικά η συγκεκριμένη προσέγγιση εφαρμόζει τις πλέον σύγχρονες απαιτήσεις του λογισμικού για την υλοποίηση κατακευκτωμένων συστημάτων με μεταβαλλόμενες απαιτήσεις.



Σχήμα 1: Παράδειγμα από τη Sun Service-Oriented Architecture for Financial Services που αναπτύχθηκε από κοινού με την SeeBeyond

2.1.2 Δικτυακές Υπηρεσίες (Web Services)

Για να περιγράψουμε την τεχνολογία των δικτυακών υπηρεσιών πρέπει να προηγηθεί μια ανάλυση της γλώσσας XML καθώς και του πρωτοκόλλου SOAP.

2.1.2.1 XML

Η XML (Extensible Markup Language), είναι το πλέον χρησιμοποιούμενο πρότυπο για την δημιουργία και διαχείριση συνταγμένων εγγράφων. Μπορεί να θεωρηθεί ακόμη ως μια οικογένεια τεχνολογιών που προβλέπει την επεξεργασία και διαχείριση πληροφορίας. Τέλος, αντιπροσωπεύει μια ολόκληρη φιλοσοφία πάνω στην αντιμετώπιση των δεδομένων που δίνει μέγιστη χρησιμότητα και ευελιξία στα δεδομένα.

Η XML είναι μια markup γλώσσα: προσφέρει τη δυνατότητα σύνταξης εγγράφων με δεδομένα ενταγμένα σε ετικέτες, δίνοντας τους έτσι κάποιο νόημα χωρίς να προβλέπει όμως την παρουσίαση της πληροφορίας στο χρήστη. Περιβάλλεται από μια σειρά από συγγενείς τεχνολογίες που όμως η χρήση τους δεν απαιτείται αλλά βοηθάει στην ολοκλήρωση ορισμένων λειτουργιών.

Ένα από τα μεγαλύτερα πλεονεκτήματα της χρήσης της XML είναι η ευελιξία που παρέχει στην ενσωμάτωση πληροφοριών, ανεξάρτητα από το αντικείμενο στο οποίο αναφερόμαστε κάθε φορά. Ακόμη, το γεγονός ότι αποτελεί πρότυπο που ακολουθείται πιστά, δίνει τη δυνατότητα σε διαφορετικά λογισμικά συστήματα να μπορούν να διαβάσουν και να γράψουν σε αυτή τη κοινή γλώσσα. Τέλος η γλώσσα αυτή σε καμία περίπτωση δεν είναι διφορούμενη καθώς διασφαλίζει ότι ένα έγγραφο XML θα διαβαστεί με τον ίδιο τρόπο σε κάθε περίπτωση.

2.1.2.2 SOAP

Το SOAP (Simple Object Access Protocol) είναι ένα πρωτόκολλο που βασίζεται στην γλώσσα XML για την επικοινωνία μεταξύ υπολογιστών. Περιλαμβάνει τρία διακριτά μέρη :

- τον φάκελο (envelope) που περιγράφει το περιεχόμενο του μηνύματος και μερικές οδηγίες για τη μέθοδο επεξεργασίας αυτού.
- τους κανόνες για την κωδικοποίηση ιδιαίτερων τύπων δεδομένων και, τέλος
- τον τρόπο εφαρμογής του φακέλου και τη μεθοδολογία κλήσεων.

Χρησιμοποιείται σε περιπτώσεις όπου μια εργασία λαμβάνει χώρα σε παραπάνω από ένα υπολογιστή. Στα πλεονεκτήματά του συγκαταλέγεται το ότι δεν επιβαρύνει το εύρος ζώνης της σύνδεσης και δεν απαιτεί ιδιαίτερη ισχύ για την κωδικοποίηση και την αποκωδικοποίησή του.

2.1.2.3 Λειτουργία Δικτυακών Υπηρεσιών

Οι δικτυακές υπηρεσίες θεωρούνται από πολλούς μια από τις πλέον ριζοσπαστικές τεχνολογίες αυτή τη στιγμή στο διαδίκτυο, καθώς παρέχουν την δυνατότητα για απρόσκοπτη επικοινωνία μεταξύ διαφορετικών εφαρμογών μέσω του διαδικτύου.

Οι δικτυακές υπηρεσίες είναι εφαρμογές που βασίζονται στη γλώσσα XML και σχετίζονται άμεσα με άλλα προγράμματα, αντικείμενα ή βάσεις δεδομένων. Ένα έγγραφο XML αποστέλλεται από το ένα πρόγραμμα στο άλλο μέσω του δικτύου και προαιρετικά λαμβάνεται μια απάντηση. Οι κανόνες των δικτυακών υπηρεσιών περιγράφουν τη σύνταξη του αρχείου, τη διαπροσωπεία αποστολής, την εισαγωγή αντικειμένων σε XML αλλά και τη διαδικασία δημοσίευσης και αναζήτησης των δικτυακών υπηρεσιών.

Η τεχνολογία αυτή μπορεί να χρησιμοποιηθεί με πολλούς τρόπους. Οι δικτυακές υπηρεσίες μπορεί να χρησιμοποιούνται τόσο σε υπολογιστές όσο και σε φορητές ψηφιακές συσκευές. Μπορούν να συμμετέχουν σε συστήματα παραγγελιών, κρατήσεων, B2B integration, κ.α. Και αυτός φαίνεται να είναι ο βασικός τους ρόλος: αποτελούν το μέσο που συνδέει διαφορετικά συστήματα ανεξάρτητα από το λογισμικό που χρησιμοποιούν και ανεξάρτητα από το που βρίσκονται.

Η ανάπτυξη των διαδικτυακών υπηρεσιών βασίζεται σε διάφορα πρότυπα, τα κυριότερα από τα οποία είναι το SOAP (Simple Object Access Protocol), το WSDL (Web Services Description Language), το UDDI (Universal Description, Discovery and Integration), και το WSIL (Web Services Inspection Language).

Η ασφάλεια είναι ένα πολύ σημαντικό τμήμα των δικτυακών υπηρεσιών, καθώς τα δεδομένα δεν πρέπει ούτε να τροποποιηθούν αλλά ούτε και να διαβαστούν από άλλους χρήστες. Στην ασφάλεια των δικτυακών υπηρεσιών έχουν προταθεί μια σειρά από standards όπως το SAML (Security Authorization Markup Language) και το XKMS (XML Key Management Specification). Φυσικά σε αυτά προστίθεται το SSL και συγκεκριμένα το HTTPS για το πρωτόκολλο HTTP

2.1.3 Java 2 Micro Edition

Η Java είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού, που αναπτύχθηκε από την Sun στις αρχές της δεκαετίας του 1990. Το βασικό στοιχείο που τη διαφοροποιεί από τις υπόλοιπες γλώσσες προγραμματισμού είναι ότι η εκτέλεση μιας εφαρμογής δεν απαιτεί την μετάφραση σε κώδικα μηχανής ούτε την μετάφραση κατά την εκτέλεση. Αντίθετα χρησιμοποιεί ένα διαφορετικό σύστημα, την μετάφραση σε bytecode το οποίο εκτελείται μέσω μιας εικονικής μηχανής Java (Java virtual machine).

Τα ειδικά χαρακτηριστικά της γλώσσας της δίνουν τη δυνατότητα να μεταφέρεται, δηλαδή να μπορεί να εκτελείται σε διαφορετικές πλατφόρμες. Σημαντικό είναι επίσης ότι έχει τη δυνατότητα πέρα από τους υπολογιστές να εκτελεστεί και σε φορητές ψηφιακές συσκευές. Σήμερα η συντριπτική πλειοψηφία των κινητών τηλεφώνων που πωλούνται υποστηρίζουν εγγενώς την εκτέλεση εφαρμογών Java.

Η Java 2 Micro Edition είναι το περιβάλλον ανάπτυξης και εκσφαλμάτωσης για εφαρμογές που προορίζονται για καταναλωτικά ηλεκτρονικά και ενσωματωμένες συσκευές. Η ανάγκη για μια διαφορετική έκδοση της Java προέκυψε από τις τεράστιες διαφορές στις επεξεργαστικές δυνατότητες ενός κινητού τηλεφώνου και ενός υπολογιστή.

Η αρχιτεκτονική της J2ME είναι σχεδιασμένη ώστε να ικανοποιεί τους παρακάτω στόχους:

- Να υποστηρίζει μια ποικιλία συσκευών με διαφορετικές προδιαγραφές. Αυτές οι συσκευές συχνά διαφέρουν στο περιβάλλον χρήστη, στην αποθήκευση δεδομένων, στην ικανότητα διασύνδεσης, στο μέγεθος της μνήμης και στην κατανάλωση ενέργειας.
- Να παρέχει μια αρχιτεκτονική που μπορεί να βελτιστοποιείται για περιορισμένους χώρους.
- Να επικεντρώνεται σε συσκευές που συχνά χρησιμοποιούνται από μόνο ένα άτομο.
- Να παρέχει δυνατότητα δικτύωσης μέσω διαφόρων μεθόδων σύνδεσης,
- Να παρέχει τη δυνατότητα λήψης εφαρμογών και δεδομένων μέσω του δικτύου.
- Να μεγιστοποιεί την ευελιξία και να παρέχει μέσα υποστήριξης εφαρμογών που υπάρχουν ή θα υλοποιηθούν στο μέλλον.
- Να παρέχει τη δυνατότητα συγγραφής και ανάπτυξης εφαρμογών σε όλους.

2.1.4 Σχεσιακές Βάσεις -Γλώσσα SQL

2.1.4.1 Σχεσιακές Βάσεις

Το σχεσιακό μοντέλο είναι σήμερα το κυρίαρχο μοντέλο δεδομένων για εφαρμογές επεξεργασίας δεδομένων. Στο σχεσιακό μοντέλο μια βάση δεδομένων αποτελείται από ένα σύνολο από πίνακες, καθένας εκ των οποίων έχει ένα μοναδικό όνομα. Κάθε πίνακας περιέχει γραμμές, όπου κάθε γραμμή απεικονίζει μια σχέση μεταξύ ενός συνόλου από τιμές. Έτσι ο πίνακας αντιστοιχίζεται σε μια μαθηματική ιδέα της σχέσης, από όπου παίρνει το όνομα του και το μοντέλο.

Στα πλεονεκτήματα χρήσης του σχεσιακού μοντέλου περιλαμβάνεται η απλότητά του, η οποία όμως δεν περιορίζει τις δυνατότητες του. Ακόμη ,και λόγω της δομής του, η δουλειά των προγραμματιστών και των διαχειριστών της βάσης γίνεται σχετικά εύκολη, αν το συγκρίνουμε με εναλλακτικά μοντέλα.

Για την πρόσβαση σε σχεσιακές βάσεις δεδομένων χρησιμοποιείται κατά κόρον η γλώσσα SQL και οι διάφορες παραλλαγές της. Φυσικά υπάρχουν και άλλες εναλλακτικές γλώσσες, όπως η QBE και η Datalog. Τρία χαρακτηριστικά παραδείγματα συστημάτων βάσεων δεδομένων είναι η Oracle, η IBM DB2, και ο Microsoft SQL Server. Σύμφωνα με τα τελευταία στατιστικά της εταιρίας Gartner ¹ η Oracle κατέχει το 48,6 % της αγοράς, η IBM το 22 % ενώ η Microsoft το 15 %.

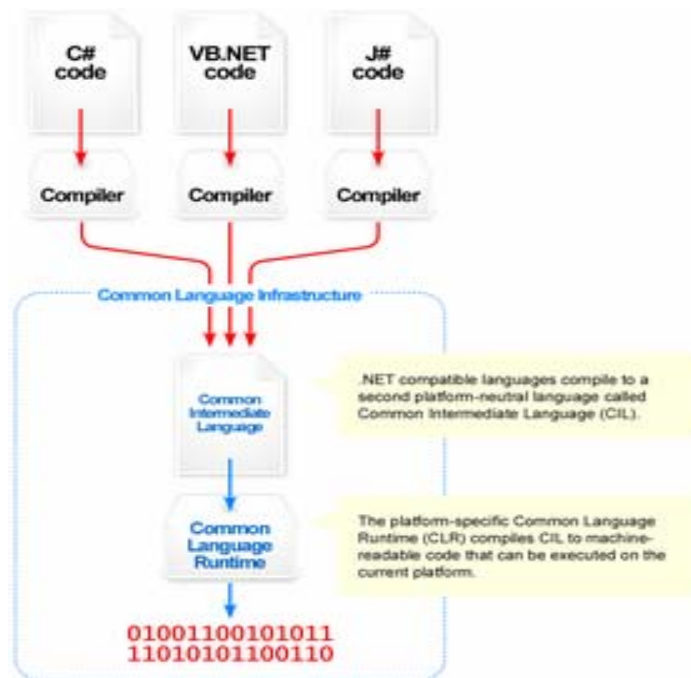
2.1.4.2 MS SQL Server

Το σύστημα που πραγματεύεται ο παρόν τόμος, αναπτύχθηκε με τη χρήση του Microsoft SQL Server 2000 και αυτή η επιλογή έγινε λαμβάνοντας υπόψη τα ιδιαίτερα στοιχεία που παρουσιάζει το λογισμικό της Microsoft. Ο Microsoft SQL Server βασίζεται στον κώδικα της Sybase και στις τελευταίες του εκδόσεις δεν υπολείπεται σε κάτι των ανταγωνιστών. Σημαντικό είναι το γεγονός ότι προσφέρει επιπλέον λειτουργικότητες που σχετίζονται τόσο με την επικοινωνία με άλλα λογισμικά πακέτα, όσο και με την XML.

2.1.5 .NET Framework

Το Microsoft .NET Framework είναι ένα συστατικό τμήμα λογισμικού που μπορεί να προστεθεί στο λειτουργικό σύστημα Windows και παρέχει επιπλέον δυνατότητες. Προσφέρει μια πληθώρα έτοιμων λύσεων σε συγκεκριμένες απαιτήσεις προγραμμάτων και ελέγχει την εκτέλεση προγραμμάτων γραμμένων για αυτό. Θεωρείται από την κατασκευάστρια εταιρία ότι θα αποτελεί θεμελιώδες τμήμα κάθε νέου προγράμματος που αναπτύσσεται για το περιβάλλον των Windows.

¹ http://www.gartner.com/press_releases/asset_152619_11.html



Σχήμα 2: Παράδειγμα εκτέλεσης κώδικα μέσω του CLR

Ένα από τα σημαντικότερα χαρακτηριστικά του είναι η δυνατότητα για εκτέλεση προγραμμάτων σε ένα ξεχωριστό περιβάλλον λογισμικού, που ονομάζεται CLR (Common Language Runtime). Το CLR παρουσιάζει κοινά σημεία με μια εικονική μηχανή, καθώς ο προγραμματιστής δεν χρειάζεται να γνωρίζει την ακριβή σύνθεση του μηχανήματος. Ακόμη προσφέρει ασφάλεια, πληθώρα γλωσσών, αυτόματη διαχείριση της μνήμης και των εξαιρέσεων λογισμικού. Έτσι, η βιβλιοθήκη κλάσεων και το κοινό περιβάλλον εκτέλεσης συναποτελούν το .NET Framework. Για την ανάπτυξη εφαρμογών παρέχεται το ενοποιημένο περιβάλλον Visual Studio.

2.1.5.1 Windows Services

Οι υπηρεσίες των Windows είναι προγράμματα που έχουν σχεδιαστεί να εκτελούνται χωρίς παρέμβαση του χρήστη. Συνήθως η λειτουργία τους ξεκινά με την εκκίνηση του λειτουργικού συστήματος. Είναι ένα εργαλείο διαθέσιμο μόνο στα λειτουργικά συστήματα Windows NT, Windows 2000, Windows XP και Windows Server 2003. Χρησιμοποιούνται συχνά για εξυπηρέτηση πελατών– (server – side tasks).

2.1.5.2 ASP .NET

Το όνομα ASP .NET χρησιμοποιείται για να περιγράψει μια σειρά τεχνολογιών ανάπτυξης δικτυακών εφαρμογών της Microsoft. Οι τεχνολογίες αυτές μπορούν να χρησιμοποιηθούν για την υλοποίηση ιστοτόπων, δικτυακών υπηρεσιών και άλλων δικτυακών εφαρμογών. Συνεργάζεται με την πλατφόρμα .NET και έτσι δεν περιορίζεται στην χρήση γλωσσών. Μπορεί δηλαδή η ανάπτυξη εφαρμογών ASP .NET να περιλαμβάνει κώδικα σε οποιαδήποτε γλώσσα συμβατή με το .NET.

2.2 Στόχος του Συστήματος

2.2.1 Παρούσα κατάσταση συστημάτων στάθμευσης

Σήμερα υπάρχουν διάφορα συστήματα ελεγχόμενης στάθμευσης, που όμως το καθένα παρουσιάζει τα δικά του πλεονεκτήματα και μειονεκτήματα.

- Παρκόμετρα στα οποία ο οδηγός μπορεί να αγοράσει χρόνο στάθμευσης είτε για μία θέση, είτε για περισσότερες. Ο έλεγχος των παραβατών μπορεί να γίνει και με PDAs ασύρματα.
- Οργανωμένοι χώροι στάθμευσης που η πληρωμή γίνεται κατά την έξοδο από το χώρο.
- Διόδια για περιορισμό της εισόδου οχημάτων στο κέντρο της πόλης (άρα περιορίζονται έτσι και τα προβλήματα με τις θέσεις στάθμευσης).
- Υπάρχουν περιπτώσεις όπου εγγεγραμμένοι χρήστες μπορούν να αγοράζουν χρόνο στάθμευσης, χρησιμοποιώντας το κινητό τους τηλέφωνο.
- Έχουν σχεδιαστεί και προωθούνται προς χρήση διάφορα συστήματα που απαιτούν εξειδικευμένο εξοπλισμό από τη μεριά του τελικού χρήστη, ειδικές κάρτες ή συσκευές.

Τα παραπάνω συστήματα, αδυνατούν να καλύψουν πλήρως τις ανάγκες του Δήμου και του πολίτη. Και αυτό καθώς παραμένουν δύσχρηστα και δυσνόητα, τόσο για τους Δημοτικούς υπαλλήλους – ελεγκτές όσο και για τον τελικό χρήστη. Ακόμη, δε δίνουν δυνατότητες διαλειτουργικότητας, δεσμεύουν τον ιδιοκτήτη σε περιορισμένες (proprietary) λύσεις χωρίς δυνατότητες επέκτασης ή τροποποίησης των σεναρίων χρήσης και με αυξημένο κόστος εφαρμογής και συντήρησης.

2.2.2 Γενική περιγραφή του συστήματος

Το σύστημα που προτείνεται περιλαμβάνει αρκετά χαρακτηριστικά κι έτσι διαφοροποιείται σε σχέση με τα υπόλοιπα συστήματα ελέγχου στάθμευσης. Πρώτα από όλα το προτεινόμενο σύστημα χρησιμοποιεί την τεχνολογία των δικτυακών υπηρεσιών (web services) η οποία δίνει τη δυνατότητα ανάπτυξης τμημάτων λογισμικού, σε διαφορετικές πλατφόρμες από τα υπόλοιπα. Για παράδειγμα, ένας διαφορετικής πλατφόρμας, διαφορετικής τεχνολογίας, διαφορετικής συσκευής πελάτης μπορεί να ενσωματωθεί εύκολα στο σύστημα, αρκεί να συμμορφώνεται στα πρότυπα που η δικτυακή υπηρεσία χρησιμοποιεί. Ακόμη, η ευελιξία στη μεταφερόμενη πληροφορία είναι εξαιρετικά αυξημένη δίνοντας τη δυνατότητα για επέκταση ή τροποποίηση των διακινούμενων πληροφοριών κατά βούληση.

Επιπροσθέτως, η δομή του συστήματος είναι τέτοια που να επιτρέπει την εφαρμογή τελείως διαφορετικών σεναρίων, διατηρώντας τις αλλαγές που πρέπει να γίνουν στο ελάχιστο. Δηλαδή, αν θεωρηθεί το παρόν σύστημα κατάλληλο για περιπτώσεις διαφορετικές από τη στάθμευση στην πόλη (π.χ. σε ελεγχόμενους χώρους) τότε με ελάχιστες ή καθόλου μετατροπές μπορεί εφαρμοστεί εκεί. Έτσι, το παρόν σύστημα μπορεί να αποτελέσει ένα ευρύτερο πλέγμα υπηρεσιών για διαφορετικές περιπτώσεις στάθμευσης ανεξάρτητα από τον ιδιοκτήτη του χώρου.

Το προτεινόμενο σύστημα προσφέρει ικανοποιητικές λύσεις σε μια πληθώρα προβλημάτων που εμφανίζονται στη διαδικασία της ελεγχόμενης στάθμευσης σε όλα τα επίπεδα που αυτή περιλαμβάνει, ξεκινώντας από το διαχειριστή του συστήματος και καταλήγοντας στον τελικό χρήστη.

- **Δέσμευση θέσης.** Η δέσμευση της θέσης στάθμευσης γίνεται εύκολα και γρήγορα από τον πελάτη, ο οποίος μπορεί να χρησιμοποιήσει το κινητό του τηλέφωνο (είτε την ειδική εφαρμογή που προσφέρεται, είτε την αποστολή SMS) και σε ελάχιστο χρόνο να δεσμεύσει τη θέση που έχει σταθμεύσει. Δίνεται η δυνατότητα πολλαπλών σεναρίων χρήσης χωρίς να αυξάνει στο ελάχιστο η πολυπλοκότητα του συστήματος. Δεν απαιτεί προηγούμενες υποδομές όπως παρκόμετρα ή ηλεκτρονικό εξοπλισμό, εκτός από πινακίδες σήμανσης της περιοχής.
- **Χρέωση.** Η χρέωση του πελάτη γίνεται με ευελιξία, είτε σε προπληρωμένες κάρτες είτε σε χρεωστικές και πιστωτικές κάρτες. Ακόμα, δίνεται μελλοντικά η δυνατότητα ώστε η χρέωση να γίνεται μέσω του τηλεπικοινωνιακού παρόχου.
- **Διαχείριση λογαριασμού.** Ο χρήστης έχει τη δυνατότητα να παρακολουθεί τις χρεώσεις που του γίνονται λόγω της στάθμευσης του. Επίσης μπορεί να παρακολουθεί και να προσαρμόζει

τις προσωπικές του ρυθμίσεις είτε χρησιμοποιώντας κινητό τηλέφωνο, είτε μέσω του δικτυακού τόπου.

- **Έλεγχος.** Ο έλεγχος των παραβάσεων γίνεται εύκολα και σύντομα από την επιβλέπουσα αρχή. Ο ελεγκτής δεν έχει παρά να πληκτρολογήσει τον αριθμό κυκλοφορίας του οχήματος, και άμεσα μπορεί να γνωρίζει αν το όχημα είναι νόμιμα ή παράνομα σταθμευμένο. Ακολούθως μπορεί επιτόπου να τυπώνει την κλήση για παράνομη στάθμευση, η οποία φυσικά αποθηκεύεται και στο σύστημα.

3

Ανάλυση και Σχεδίαση

Στο παρόν κεφάλαιο παρουσιάζονται τα σχετικά με την ανάλυση και σχεδίαση του συστήματος ζητήματα μέσω τυποποιημένων εγγράφων όπως προβλέπονται από την τεχνολογία λογισμικού. Συγκεκριμένα τα έγγραφα αυτά είναι βασισμένα στα αντίστοιχα έγγραφα προσδιορισμού απαιτήσεων από το λογισμικό και αρχιτεκτονικής της διαδικασίας ανάπτυξης λογισμικού Rational Unified Process (R.U.P.). Στην πρώτη ενότητα περιγράφονται οι λειτουργίες του συστήματος μέσω του εγγράφου με τίτλο «Έγγραφο Προσδιορισμού Απαιτήσεων από το Λογισμικό». Στη δεύτερη και στην τρίτη ενότητα παρουσιάζεται ο βασικός και ο λεπτομερής σχεδιασμός του συστήματος αντίστοιχα, μέσω των εγγράφων «Έγγραφο Περιγραφής του Αρχιτεκτονικού Σχεδίου» και «Έγγραφο Περιγραφής του Λεπτομερούς Σχεδίου».

3.1 Έγγραφο Προσδιορισμού Απαιτήσεων από το Λογισμικό

3.1.1 Εισαγωγή

3.1.1.1 Σκοπός

Σκοπός της παρούσας ενότητας είναι η παρουσίαση των απαιτήσεων για το σύστημα ελέγχου στάθμευσης. Το προτεινόμενο σύστημα θα υλοποιεί όχι μόνο τις δικτυακές υπηρεσίες που απαιτούνται για τη στάθμευση, αλλά και εφαρμογές πελάτη που εγκαθίστανται στις ασύρματες φορητές συσκευές (κινητά τηλέφωνα, pdas, κτλ) του χρήστη και του ελεγκτή. Ο χρήστης του συστήματος θα έχει επιλογές που θα αφορούν στον τρόπο και τη διάρκεια χρέωσης του. Ακόμη, ο ελεγκτής του συστήματος θα μπορεί με ευκολία να πραγματοποιεί τον έλεγχο της νομιμότητας της

στάθμευσης οχημάτων. Επιπροσθέτως, θα αναπτυχθεί ένα τμήμα λογισμικού που βασίζεται σε ιστοσελίδες και δίνει τη δυνατότητα στο χρήστη για τον έλεγχο του λογαριασμού του, αλλά και στην επιβλέπουσα αρχή για πλήρη έλεγχο του συστήματος και για εξαγωγή στατιστικών και συμπερασμάτων από αυτά.

3.1.1.2 Εμβέλεια

Παρακάτω παρουσιάζονται οι εργασίες που θα αυτοματοποιούνται από το σύστημα, και γίνεται μια ανάλυση των περιπτώσεων χρήσης που προκύπτουν. Στα διαγράμματα χρησιμοποιείται η γλώσσα UML.

3.1.1.3 Ορισμοί, Ακρωνύμια, Συντομογραφίες

Web Services: Δικτυακές Υπηρεσίες

UML: η γλώσσα που χρησιμοποιήθηκε κατά την σχεδίαση των διαγραμμάτων.

Χρήστης: ο τελικός χρήστης του συστήματος ελέγχου στάθμευσης, δηλαδή ο οδηγός.

Ελεγκτής: ο υπεύθυνος του συστήματος που επιφορτίζεται με το καθήκον να παρακολουθεί τη νόμιμη διαδικασία στάθμευσης.

Όχημα: η οντότητα που δεσμεύει μια θέση στάθμευσης. Μπορεί να είναι τετράτροχο ή δίτροχο – δεν έχει εισαχθεί αντίστοιχος περιορισμός κατά τη μελέτη του συστήματος.

Διαχειριστής: ο υπεύθυνος που ορίζεται από τον ιδιοκτήτη – επιβλέποντα του συστήματος ώστε να παρακολουθεί τις διαδικασίες ελέγχου στάθμευσης, και με τη βοήθεια των στατιστικών να μπορεί να τροποποιεί τα σενάρια χρήσης και γενικότερα τις παραμέτρους του συστήματος.

3.1.1.4 Βιβλιογραφία

Οι παρακάτω ενότητες περιέχουν αναφορές στη βιβλιογραφία που καταγράφεται στο τέλος του παρόντος τόμου.

3.1.2 Απαιτήσεις

3.1.2.1 Προδιαγραφές περιπτώσεων χρήσης

Οι προδιαγραφές περιπτώσεων χρήσης αναλύονται με σαφήνεια στις ενότητες §3.1.3 και §3.1.4 που ακολουθούν.

3.1.2.2 Λειτουργικότητα

Ακολούθως περιγράφεται η λειτουργικότητα που πρέπει να περιλαμβάνουν τα τμήματα του λογισμικού.

3.1.2.2.1 Λογισμικό πελάτη:

Το λογισμικό πελάτη του χρήστη θα πρέπει να παρέχει ένα ενιαίο περιβάλλον χρήσης μέσω του οποίου θα δίνεται πρόσβαση στις υπηρεσίες που παρέχονται. Επιπροσθέτως, θα πρέπει να δίνεται η δυνατότητα για πρόσβαση στις δικτυακές υπηρεσίες μέσω σύνδεσης με το δίκτυο. Αφαιρετικά, η λειτουργικότητα αυτή απαιτείται τόσο από το λογισμικό πελάτη που θα χρησιμοποιεί ο χρήστης όσο και από αυτό που θα χρησιμοποιεί ο ελεγκτής.

3.1.2.2.2 Application Server:

Ο εξυπηρετητής του συστήματος όχι μόνο θα προσφέρει τις δικτυακές υπηρεσίες τις οποίες θα καταναλώνει ο πελάτης, αλλά θα αποτελεί μια ευρύτερη διαπροσωπεία για το σύστημα. Έτσι θα παρέχει (ενδεχομένως σε διαφορετικούς κόμβους) τη δυνατότητα λήψης SMS αλλά και ένα ολοκληρωμένο δικτυακό τόπο.

3.1.2.3 Απαιτήσεις χρήσης

Η χρήση του λογισμικού πελάτη αλλά και του δικτυακού τόπου θα πρέπει να είναι εύκολη. Πρέπει να μην αποπροσανατολίζεται ο χρήστης με πολλαπλές επιλογές, χωρίς όμως αυτό να στερεί κάτι από τις δυνατότητες του συστήματος.

3.1.2.4 Απαιτήσεις αξιοπιστίας

Η αξιοπιστία του συστήματος βασίζεται τόσο στον εξυπηρετητή όσο και στον πελάτη. Από την πλευρά του εξυπηρετητή απαιτείται η αδιάλειπτη λειτουργία του ενώ από την πλευρά του πελάτη η δυνατότητα δικτύωσης του.

3.1.2.5 Απαιτήσεις ασφάλειας

Οι απαιτήσεις για την ασφάλεια του συστήματος είναι αρκετά υψηλές καθώς όχι μόνο μεταφέρονται δεδομένα των χρηστών όπως τα οχήματα τους και η στάθμευση τους, αλλά το πλέον σημαντικό είναι ότι περιλαμβάνονται πληροφορίες βάσει των οποίων γίνεται η χρέωση των πελατών του συστήματος. Η ασφάλεια προτείνεται να γίνει σε δύο επίπεδα, τόσο με τη χρήση ονόματος χρήστη και κωδικού όσο και με την εφαρμογή ενός ασφαλούς στρώματος – επικοινωνία μέσω SSL.

3.1.2.6 Απαιτήσεις επιδόσεων

Οι απαιτήσεις για τις επιδόσεις του συστήματος δεν οφείλουν να υπακούουν σε συγκεκριμένες δεσμεύσεις, αλλά πρέπει να είναι τέτοιες που να καθιστούν τη χρήση του εφικτή και χωρίς καθυστερήσεις για τον τελικό χρήστη. Από την πλευρά του εξυπηρετητή απαιτείται αρκετή υπολογιστική ισχύς και ενδεχομένως κατανομή των λειτουργιών σε περισσότερους από έναν κόμβους. Από την πλευρά του πελάτη απαιτείται ικανοποιητική παροχή σύνδεσης στο διαδίκτυο.

3.1.2.7 Πρότυπα

Τα πρότυπα που ακολουθεί η παρούσα σχεδίαση και υλοποίηση του συστήματος αναλύονται στην παράγραφο 2.1.

3.1.3 Εργασίες προς αυτοματοποίηση

3.1.3.1 Σύνοψη

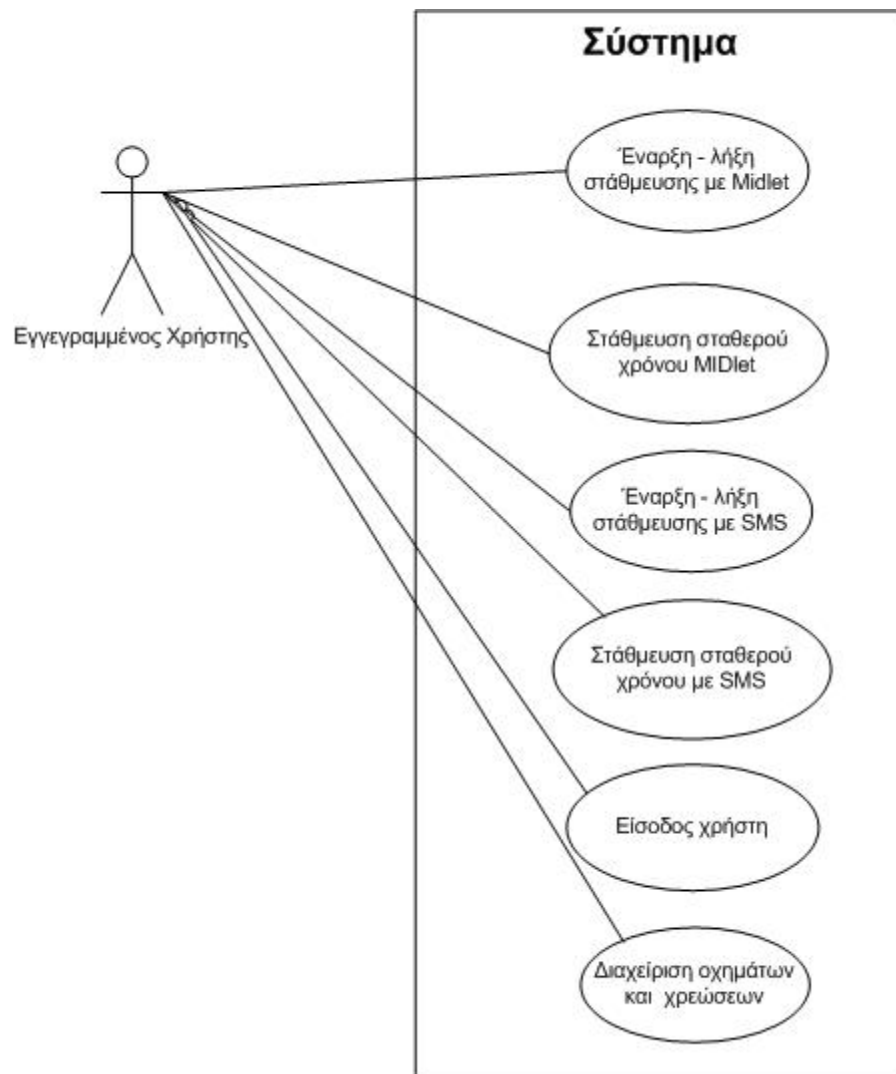
Οι εργασίες που αυτοματοποιούνται από το συνδυασμένο σύστημα υλικού και λογισμικού, είναι οι παρακάτω:

- Έναρξη και λήξη στάθμευσης για εγγεγραμμένο χρήστη
- Στάθμευση σταθερού χρόνου για εγγεγραμμένο χρήστη

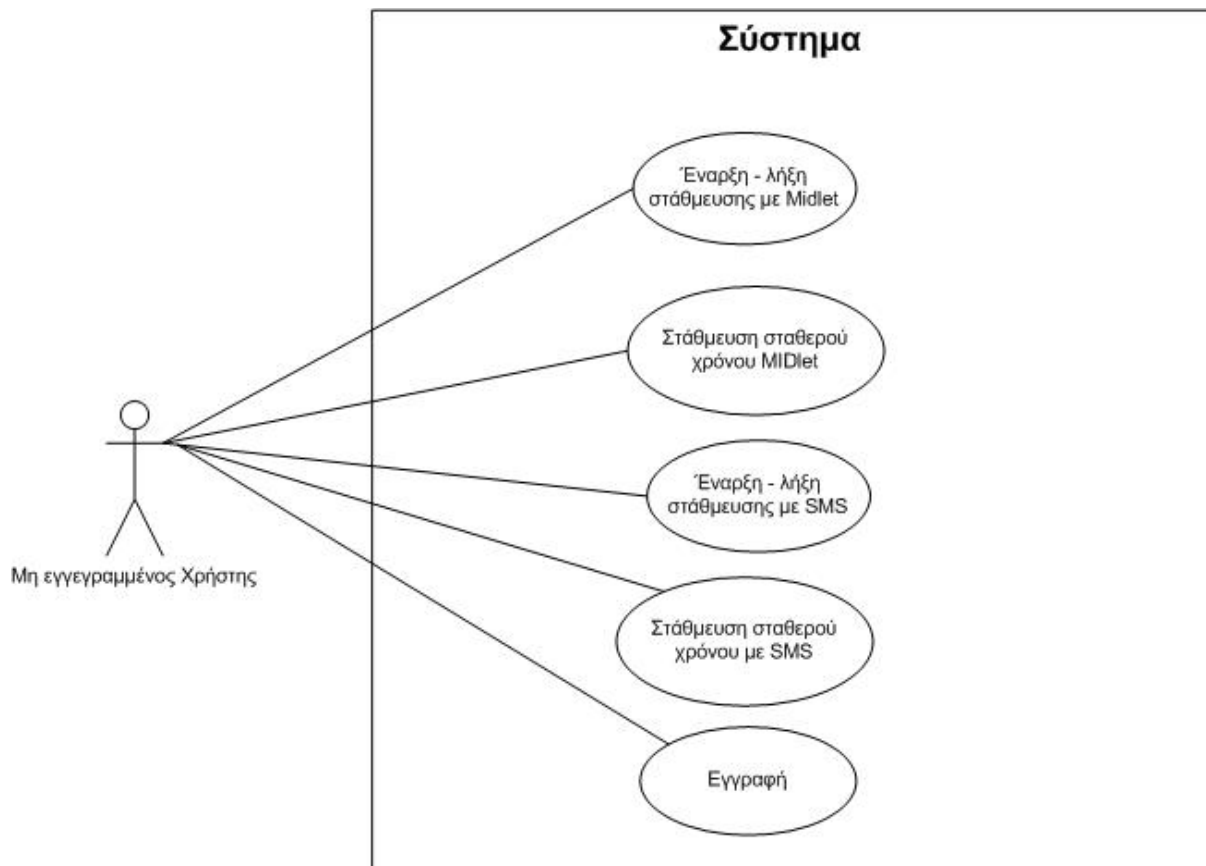
- Έναρξη και λήξη στάθμευσης για μη εγγεγραμμένο χρήστη
- Στάθμευση σταθερού χρόνου για μη εγγεγραμμένο χρήστη
- Έλεγχος οχήματος, καταγραφή και εκτύπωση παράβασης
- Εγγραφή χρήστη
- Διαχείριση οχημάτων – χρεώσεων από το χρήστη
- Έλεγχος συστήματος
- Παρακολούθηση στατιστικών

Από τα παρακάτω έχουν παραλειφθεί ορισμένες περιπτώσεις χρήσης που έχουν ρόλο συμπληρωματικό στις παραπάνω. Οι εν λόγω περιπτώσεις χρήσης δεν αναλύονται αλλά χρησιμοποιούνται τόσο στο δικτυακό τόπο του συστήματος, όσο και στην εφαρμογή πελάτη. Αναφέρονται στην είσοδο του χρήστη ή του διαχειριστή, στον έλεγχο του υπολοίπου του λογαριασμού, στην ενημέρωση των στοιχείων της κατάστασης του και σε απευθείας επιλογή ορισμένων ρυθμίσεων.

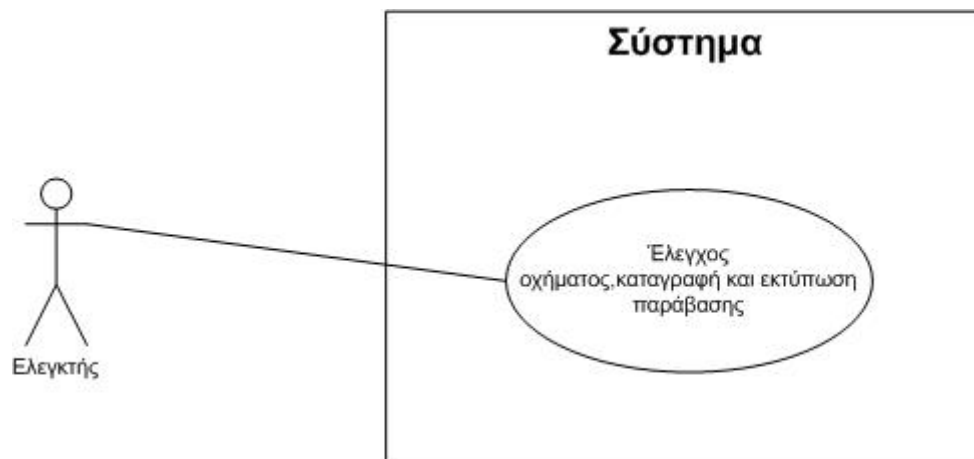
Παρακάτω φαίνονται σε διαγράμματα οι περιπτώσεις χρήσης για 4 διαφορετικούς τύπους δραστών σε σχέση με το σύστημα.



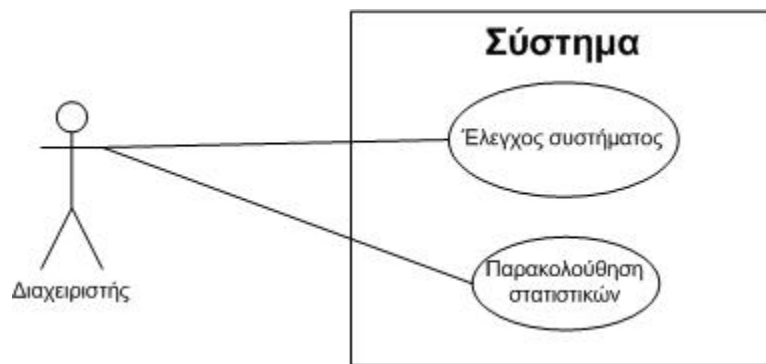
Σχήμα 3: Διάγραμμα Περιπτώσεων Χρήσης - εγγεγραμμένος χρήστης



Σχήμα 4: Διάγραμμα Περιπτώσεων Χρήσης - μη εγγεγραμμένος χρήστης



Σχήμα 5: Διάγραμμα Περιπτώσεων Χρήσης - ελεγκτής



Σχήμα 6: Διάγραμμα Περιπτώσεων Χρήσης - διαχειριστής

3.1.4 Περιπτώσεις χρήσης

3.1.4.1 Εισαγωγή

Παρακάτω περιγράφεται ξεχωριστά κάθε περίπτωση χρήσης και δίνεται η ροή γεγονότων σύμφωνα με την οποία ο δράστης (actor) προχωρά στη συγκεκριμένη δραστηριότητα. Ακόμη, αναφέρονται αναλυτικά εναλλακτικές πορείες δράσης. επίσης, περιλαμβάνονται διαγράμματα δραστηριότητας τα οποία δίνουν την ακριβή ροή των γεγονότων.

3.1.4.2 Έναρξη και λήξη στάθμευσης για εγγεγραμμένο χρήστη

3.1.4.2.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο εγγεγραμμένος χρήστης του συστήματος δεσμεύει χρόνο στάθμευσης σε δύο στάδια. Σε πρώτο στάδιο, αφού σταθμεύσει το όχημα του επιλέγει την έναρξη της χρέωσης και σε δεύτερο στάδιο, όταν αναχωρεί, επιλέγει τη λήξη της στάθμευσης. Έτσι ο χρήστης χρεώνεται μόνο για το διάστημα για το οποίο έχει σταθμευμένο το όχημά του.

3.1.4.2.2 Ροή γεγονότων

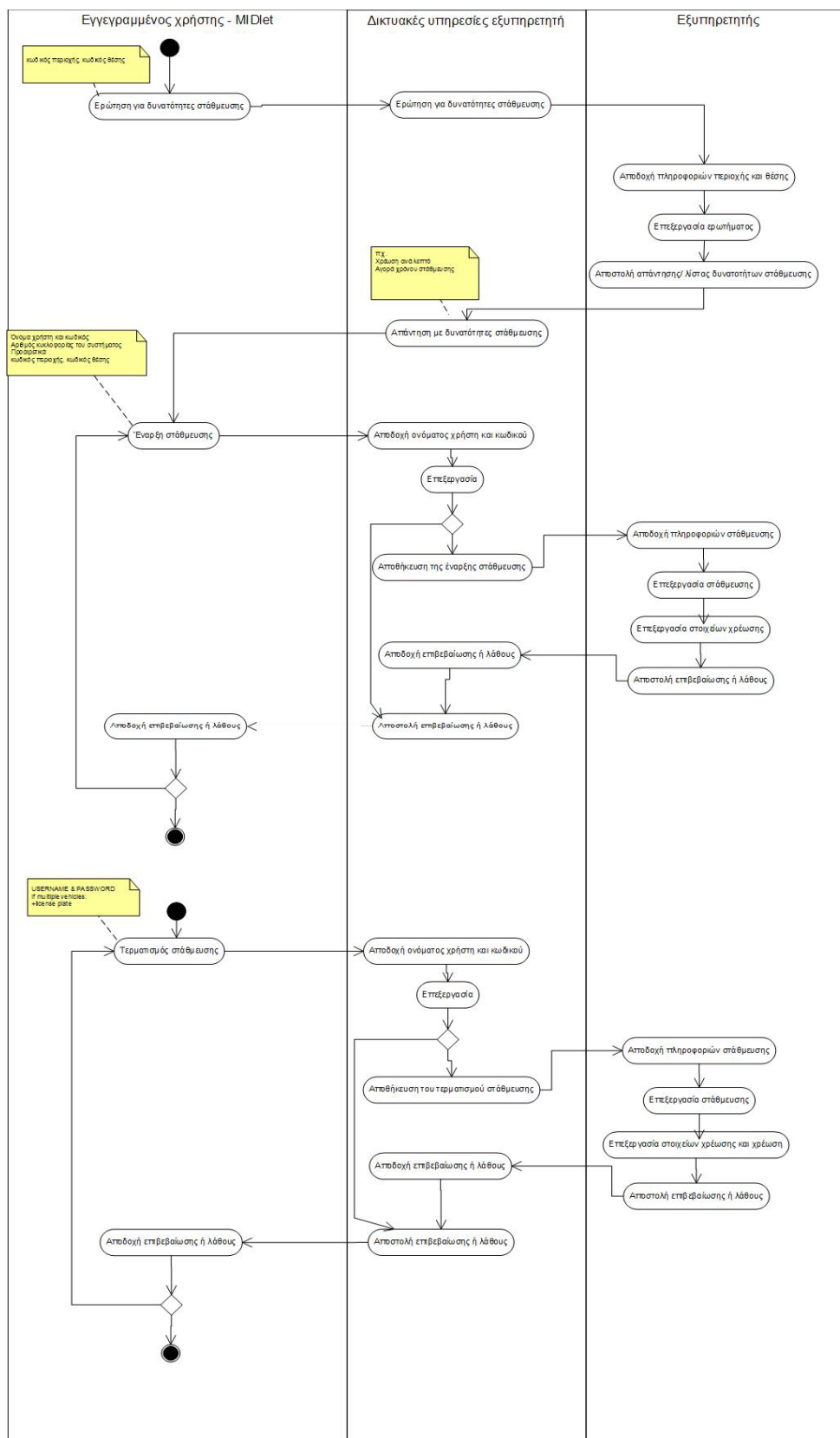
3.1.4.2.2.1 Βασική Ροή - MIDlet

1. Επιλογή της στάθμευσης από το μενού της εφαρμογής του πελάτη
2. Επιλογή των επιπλέον ρυθμίσεων (περιοχή – θέση)
3. Επικοινωνία με τον εξυπηρετητή του συστήματος

4. Επιλογή της χρέωσης ανά λεπτό
5. Έλεγχος για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη
6. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση ξεκίνησε κανονικά ή ότι για κάποιο λόγο δεν εκκινήθηκε η διαδικασία χρέωσης .
7. Επιλογή του τερματισμού χρέωσης από το μενού της εφαρμογής του πελάτη
8. Επικοινωνία με τον εξυπηρετητή του συστήματος
9. Έλεγχος για το αν μπορεί να σταματήσει η χρέωση και αν υπάρχουν κάποιες απρόβλεπτες καταστάσεις.
10. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση σταμάτησε και για τα προβλήματα που τυχόν προέκυψαν.

3.1.4.2.2.2 Εναλλακτικές Ροές - SMS

1. Ο εγγεγραμμένος χρήστης στέλνει ένα SMS από το κινητό του σε προκαθορισμένο αριθμό
2. Ο εξυπηρετητής ενημερώνεται και ελέγχει για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη.
3. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση ξεκίνησε κανονικά ή ότι για κάποιο λόγο δεν εκκινήθηκε η διαδικασία χρέωσης
4. Ο εγγεγραμμένος χρήστης στέλνει ένα SMS από το κινητό του σε προκαθορισμένο αριθμό
5. Ο εξυπηρετητής ενημερώνεται και ζητά το τέλος της χρέωσης
6. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση σταμάτησε και για τα προβλήματα που τυχόν προέκυψαν.



Σχήμα 7: Ροή γεγονότων έναρξης και λήξης στάθμευσης για εγγεγραμμένο χρήστη

3.1.4.3 Στάθμευση σταθερού χρόνου για εγγεγραμμένο χρήστη

3.1.4.3.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο εγγεγραμμένος χρήστης του συστήματος δεσμεύει χρόνο στάθμευσης σε ένα μόνο στάδιο. Ο χρήστης στην ουσία αγοράζει έναν συγκεκριμένο χρόνο στάθμευσης για μια συγκεκριμένη θέση.

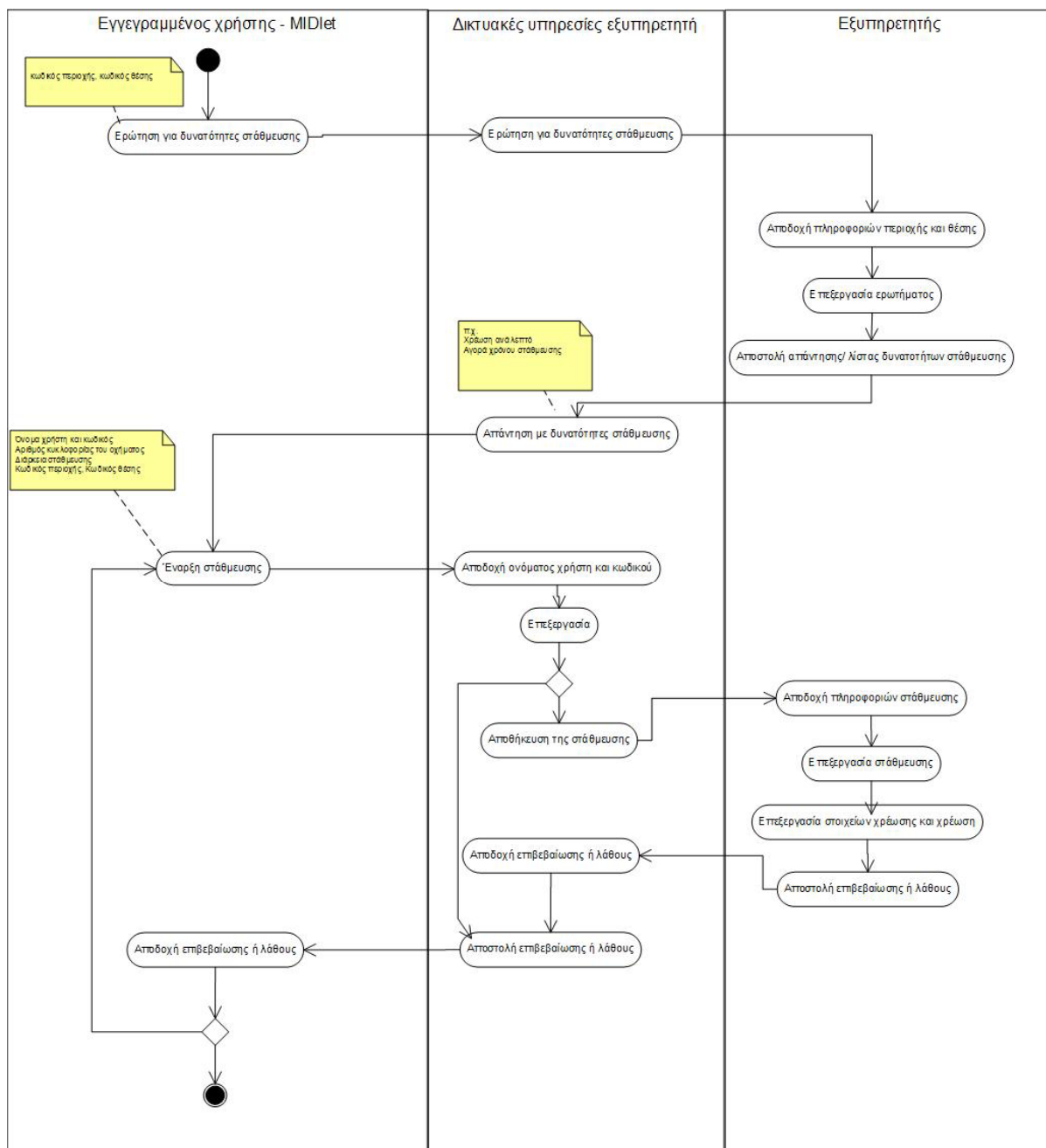
3.1.4.3.2 Ποή γεγονότων

3.1.4.3.2.1 Βασική Ποή - MIDlet

1. Επιλογή της στάθμευσης από το μενού της εφαρμογής του πελάτη
2. Επιλογή των επιπλέον ρυθμίσεων (περιοχή – θέση)
3. Επικοινωνία με τον εξυπηρετητή του συστήματος
4. Επιλογή της σταθερής χρέωσης για κάποιο συγκεκριμένο χρόνο
5. Έλεγχος για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη
6. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση έγινε κανονικά **ή** ότι για κάποιο λόγο δεν είναι εφικτή η αγορά χρόνου στάθμευσης

3.1.4.3.2.2 Εναλλακτικές Ροές - SMS

1. Ο εγγεγραμμένος χρήστης στέλνει ένα SMS από το κινητό του σε προκαθορισμένο αριθμό
2. Ο εξυπηρετητής ενημερώνεται και ελέγχει για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη.
3. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση έγινε κανονικά **ή** ότι για κάποιο λόγο δεν είναι εφικτή η αγορά χρόνου στάθμευσης



Σχήμα 8: Ροή γεγονότων στάθμευσης σταθερού χρόνου για εγγεγραμμένο χρήστη

3.1.4.4 Έναρξη και λήξη στάθμευσης για μη εγγεγραμμένο χρήστη

3.1.4.4.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο μη εγγεγραμμένος χρήστης του συστήματος δεσμεύει χρόνο στάθμευσης σε δύο στάδια. Σε πρώτο στάδιο, αφού σταθμεύσει το όχημα του, επιλέγει την έναρξη της χρέωσης και σε δεύτερο στάδιο, όταν αναχωρεί, επιλέγει τη λήξη της στάθμευσης. Έτσι ο χρήστης χρεώνεται μόνο για το διάστημα για το οποίο έχει σταθμευμένο το όχημά του. Ο χρήστης εφόσον δεν κατέχει ένα λογαριασμό στο σύστημα, οφείλει να έχει κάποιον τρόπο πληρωμής- εδώ θεωρούμε ότι είναι μια προπληρωμένη κάρτα. Αυτή η πληρωμή θα μπορούσε να γίνεται και μέσω του παρόχου κινητής τηλεφωνίας.

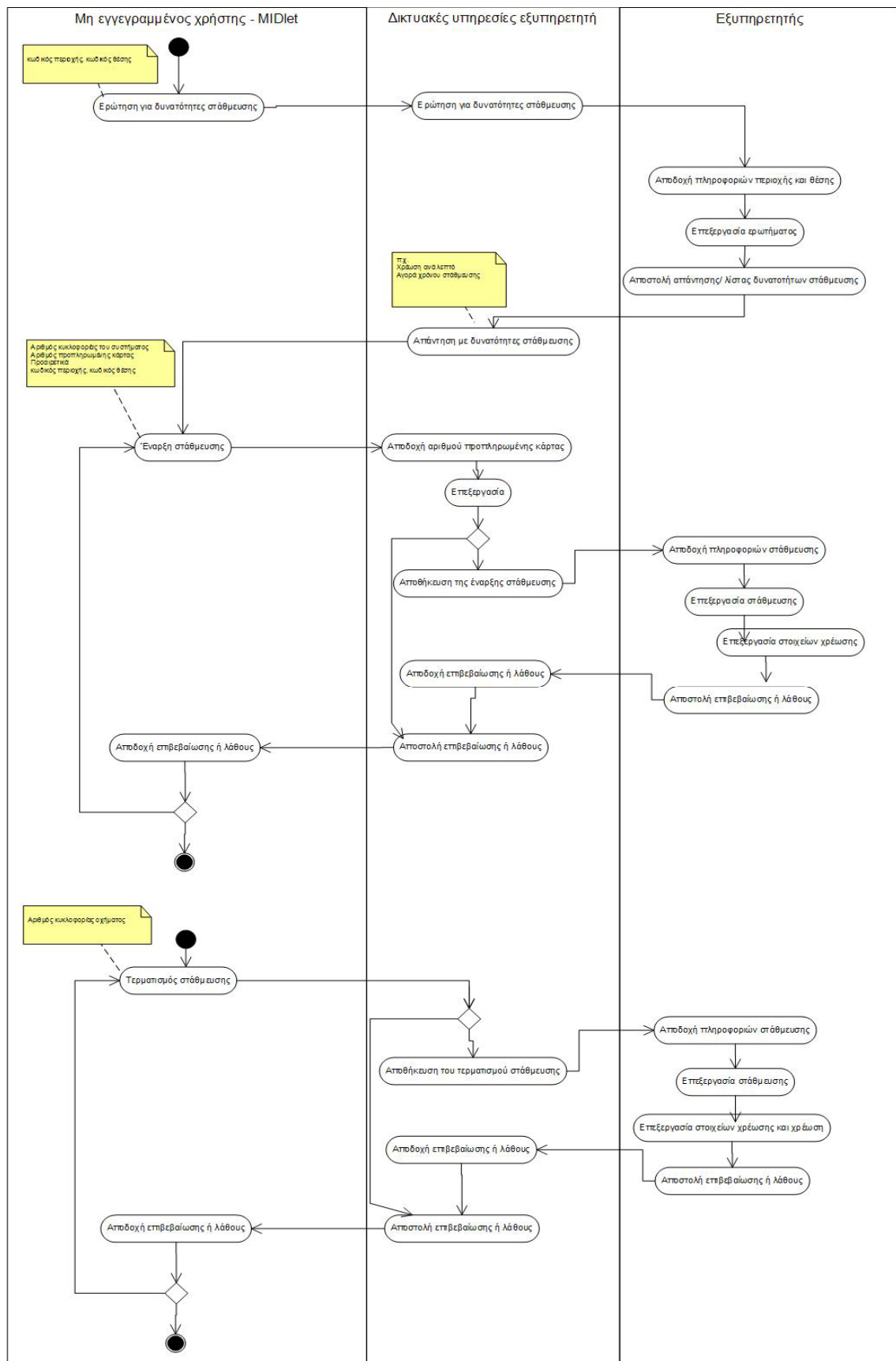
3.1.4.4.2 Ποή γεγονότων

3.1.4.4.2.1 Βασική Ποή - MIDlet

1. Επιλογή της στάθμευσης από το μενού της εφαρμογής του πελάτη
2. Εισαγωγή των στοιχείων του οχήματος αλλά και του αριθμού της προπληρωμένης κάρτας η οποία χρεώνεται.
3. Επιλογή των επιπλέον ρυθμίσεων (περιοχή – θέση)
4. Επικοινωνία με τον εξυπηρετητή του συστήματος
5. Επιλογή της χρέωσης ανά λεπτό
6. Έλεγχος για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη
7. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση ξεκίνησε κανονικά ή ότι για κάποιο λόγο δεν εκκινήθηκε η διαδικασία χρέωσης .
8. Επιλογή του τερματισμού χρέωσης από το μενού της εφαρμογής του πελάτη
9. Επικοινωνία με τον εξυπηρετητή του συστήματος
10. Έλεγχος για το αν μπορεί να σταματήσει η χρέωση και αν υπάρχουν κάποιες απρόβλεπτες καταστάσεις.
11. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση σταμάτησε και για τα προβλήματα που τυχόν προέκυψαν.

3.1.4.4.2 Εναλλακτικές Ροές - SMS

1. Ο χρήστης στέλνει ένα SMS από το κινητό του σε προκαθορισμένο αριθμό. Το μήνυμα αυτό θα πρέπει να περιέχει υποχρεωτικά ορισμένα στοιχεία του πελάτη όπως ο αριθμός κυκλοφορίας του οχήματος του και ο αριθμός της προπληρωμένης κάρτας.
2. Ο εξυπηρετητής ενημερώνεται και ελέγχει για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη.
3. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση ξεκίνησε κανονικά ή ότι για κάποιο λόγο δεν εκκινήθηκε η διαδικασία χρέωσης
4. Ο χρήστης στέλνει ένα SMS από το κινητό του σε προκαθορισμένο αριθμό
5. Ο εξυπηρετητής ενημερώνεται και ζητά το τέλος της χρέωσης
6. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση σταμάτησε και για τα προβλήματα που τυχόν προέκυψαν.



Σχήμα 9: Ροή γεγονότων έναρξης και λήξης στάθμευσης για μη εγγεγραμμένο χρήστη

3.1.4.5 Στάθμευση σταθερού χρόνου για μη εγγεγραμμένο χρήστη

3.1.4.5.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο μη εγγεγραμμένος χρήστης του συστήματος δεσμεύει χρόνο στάθμευσης άμεσα. Στην ουσία αναφερόμαστε στην αγορά ενός συγκεκριμένου χρόνου στάθμευσης από το σύστημα. Ισχύουν για το χρήστη οι περιορισμοί που αναφέρθηκαν παραπάνω και αφορούν το σύστημα χρέωσής του

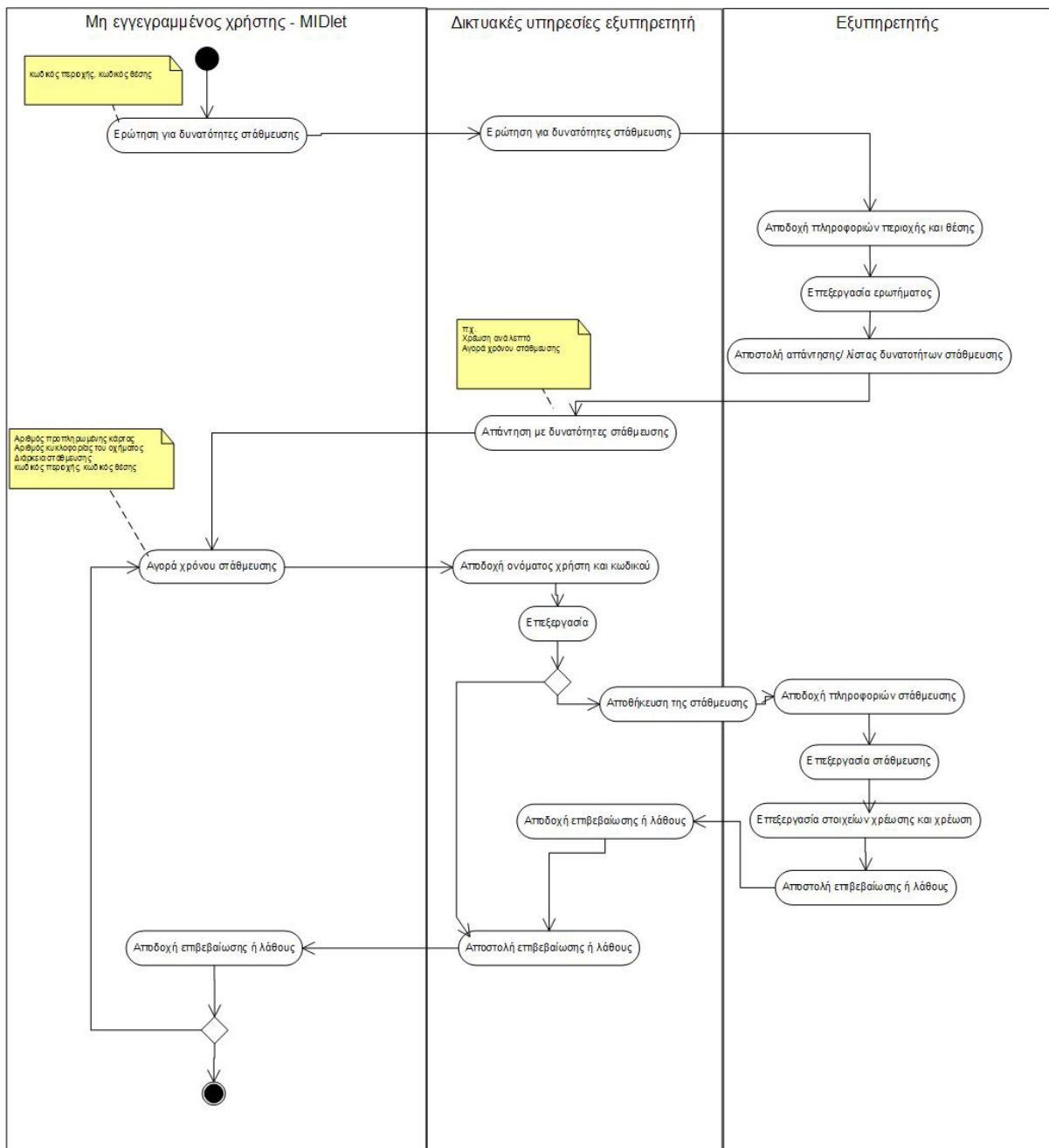
3.1.4.5.2 Ροή γεγονότων

3.1.4.5.2.1 Βασική Ροή - MIDlet

1. Επιλογή της στάθμευσης από το μενού της εφαρμογής του πελάτη
2. Εισαγωγή των στοιχείων του οχήματος αλλά και του αριθμού της προπληρωμένης κάρτας η οποία χρεώνεται.
3. Επιλογή των επιπλέον ρυθμίσεων (περιοχή – θέση)
4. Επικοινωνία με τον εξυπηρετητή του συστήματος
5. Επιλογή της σταθερής χρέωσης για κάποιο συγκεκριμένο χρόνο
6. Έλεγχος για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη
7. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση έγινε κανονικά ή ότι για κάποιο λόγο δεν είναι εφικτή η αγορά χρόνου στάθμευσης

3.1.4.5.2.2 Εναλλακτικές Ροές - SMS

1. Ο χρήστης στέλνει ένα SMS από το κινητό του σε προκαθορισμένο αριθμό. . Το μήνυμα αυτό θα πρέπει να περιέχει υποχρεωτικά ορισμένα στοιχεία του πελάτη όπως ο αριθμός κυκλοφορίας του οχήματος του και ο αριθμός της προπληρωμένης κάρτας.
2. Ο εξυπηρετητής ενημερώνεται και ελέγχει για το αν μπορεί να ξεκινήσει κανονικά η απρόσκοπτη χρέωση του πελάτη.
3. Απάντηση του εξυπηρετητή στον πελάτη, ότι η χρέωση έγινε κανονικά ή ότι για κάποιο λόγο δεν είναι εφικτή η αγορά χρόνου στάθμευσης



Σχήμα 10: Ροή γεγονότων στάθμευσης σταθερού χρόνου για μη εγγεγραμμένο χρήστη

3.1.4.6 Έλεγχος οχήματος, καταγραφή και εκτύπωση παράβασης

3.1.4.6.1 Περιγραφή περίπτωσης χρήσης

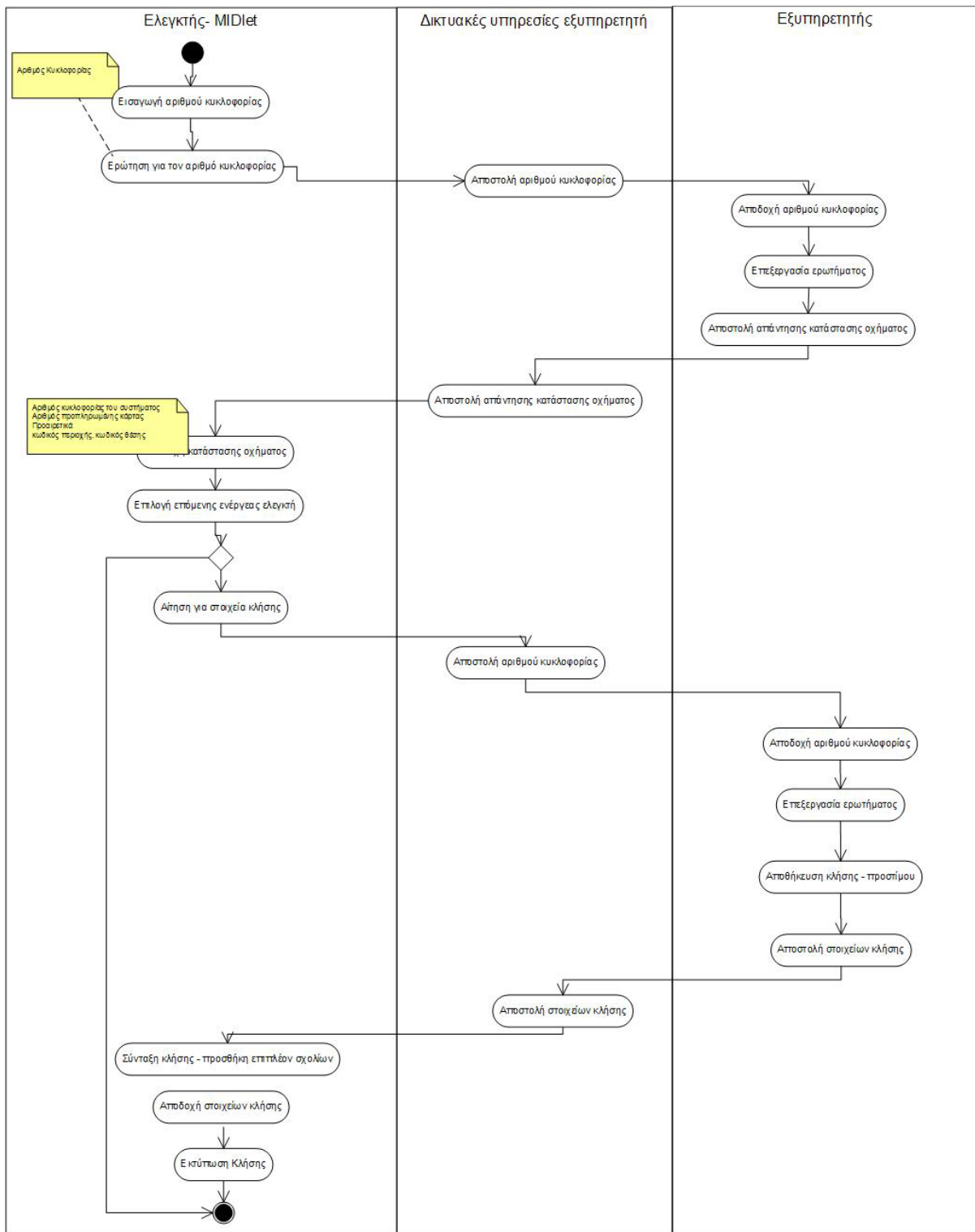
Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία κάποιος ελεγκτής του συστήματος ενημερώνεται για το αν το όχημα που βλέπει είναι νόμιμο ή παράνομο σταθμευμένο. Ο έλεγχος ενός οχήματος γίνεται με χρήση της εφαρμογής πελάτη. Ακολουθώντας, αν το όχημα δεν πληροί τις

προϋποθέσεις της νόμιμης στάθμευσης, ο ελεγκτής μπορεί αυτόματα να εκδώσει κλήση σε βάρος του συγκεκριμένου οχήματος με στοιχεία που θα προμηθευτεί από το ίδιο το σύστημα.

3.1.4.6.2 Ποή γεγονότων

3.1.4.6.2.1 Βασική Ποή - MIDlet

1. Ο ελεγκτής εισάγει στη φορητή ψηφιακή συσκευή του τον αριθμό κυκλοφορίας του οχήματος προς έλεγχο.
2. Επικοινωνία με τον εξυπηρετητή του συστήματος
3. Ο εξυπηρετητής του συστήματος αποκρίνεται με την πληροφορία της νομιμότητας ή όχι της στάθμευσης
4. Ο ελεγκτής, κατά την κρίση του αν συντρέχουν και άλλοι λόγοι (στάθμευση σε δύο θέσεις, πάνω σε πεζοδρόμιο), επιλέγει αν θα ζητήσει τα στοιχεία έκδοσης κλήσης.
5. Επικοινωνία με τον εξυπηρετητή του συστήματος
6. Ο εξυπηρετητής του συστήματος αποκρίνεται με τα αναγκαία δεδομένα για την έκδοση κλήσης.
7. Τα στοιχεία αυτά εκτυπώνονται στον φορητό εκτυπωτή που έχει μαζί του ο ελεγκτής.



Σχήμα 11 Ροή γεγονότων ελέγχου οχήματος, καταγραφής και εκτύπωσης παράβασης

3.1.4.7 Εγγραφή χρήστη

3.1.4.7.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη δυνατότητα που δίνεται σε έναν η εγγεγραμμένο χρήστη του συστήματος να εγγραφεί στο σύστημα, ώστε να μπορεί με μεγαλύτερη ευκολία να σταθμεύει το όχημά του.

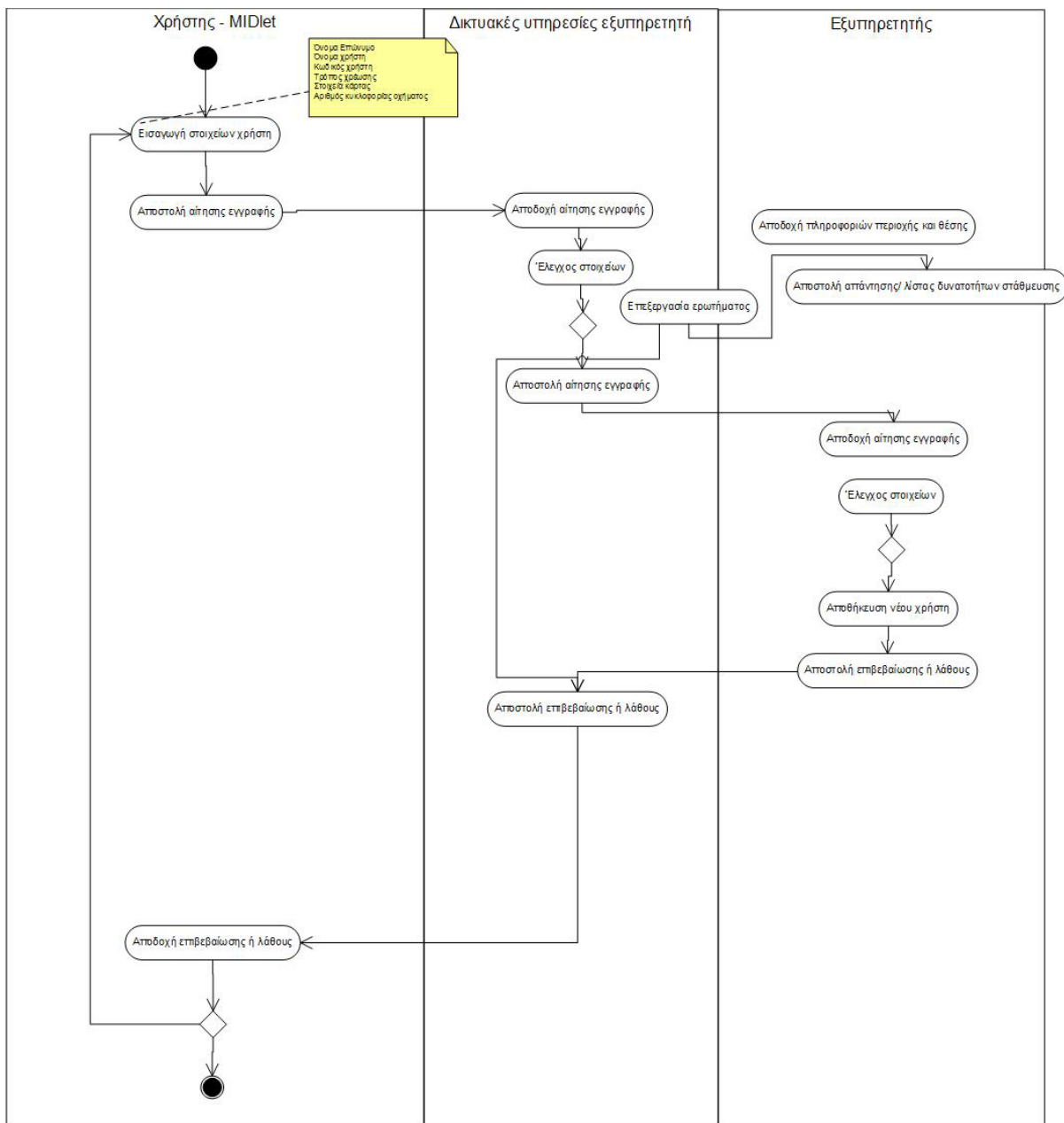
3.1.4.7.2 Ροή γεγονότων

3.1.4.7.2.1 Βασική Ροή - MIDlet

1. Ο χρήστης χρησιμοποιεί το λογισμικό πελάτη για φορητές ψηφιακές συσκευές και εισέρχεται στο μενού που αντιστοιχεί στην εγγραφή.
2. Εισάγει τα απαραίτητα δεδομένα για την εγγραφή του στο σύστημα.
3. Ο εξυπηρετητής θα ελέγχει την εγκυρότητα των δεδομένων και θα εισάγει τον χρήστη
4. Ο χρήστης ενημερώνεται για την επιτυχία ή όχι της εγγραφής

3.1.4.7.2.2 Εναλλακτική Ροή - Web

1. Ο χρήστης θα εισέρχεται στο δικτυακό τόπο του συστήματος μέσω κάποιου φυλλομετρητή
2. Επιλέγει και εισάγει με ασφάλεια τα αναγκαία δεδομένα. Όπως ονοματεπώνυμο, όνομα χρήστη, κωδικό, όχημα, τρόπο χρέωσης.
3. Ο εξυπηρετητής ελέγχει την εγκυρότητα των δεδομένων και θα εισάγει τον χρήστη
4. Ο χρήστης ενημερώνεται για την επιτυχία ή όχι της εγγραφής
5. Αν θέλει θα μπορεί να «κατεβάσει» στο κινητό του το αντίστοιχο λογισμικό πελάτη (MIDlet)



Σχήμα 12: Ροή γεγονότων εγγραφής χρήστη

3.1.4.8 Διαχείριση οχημάτων – χρεώσεων από το χρήστη

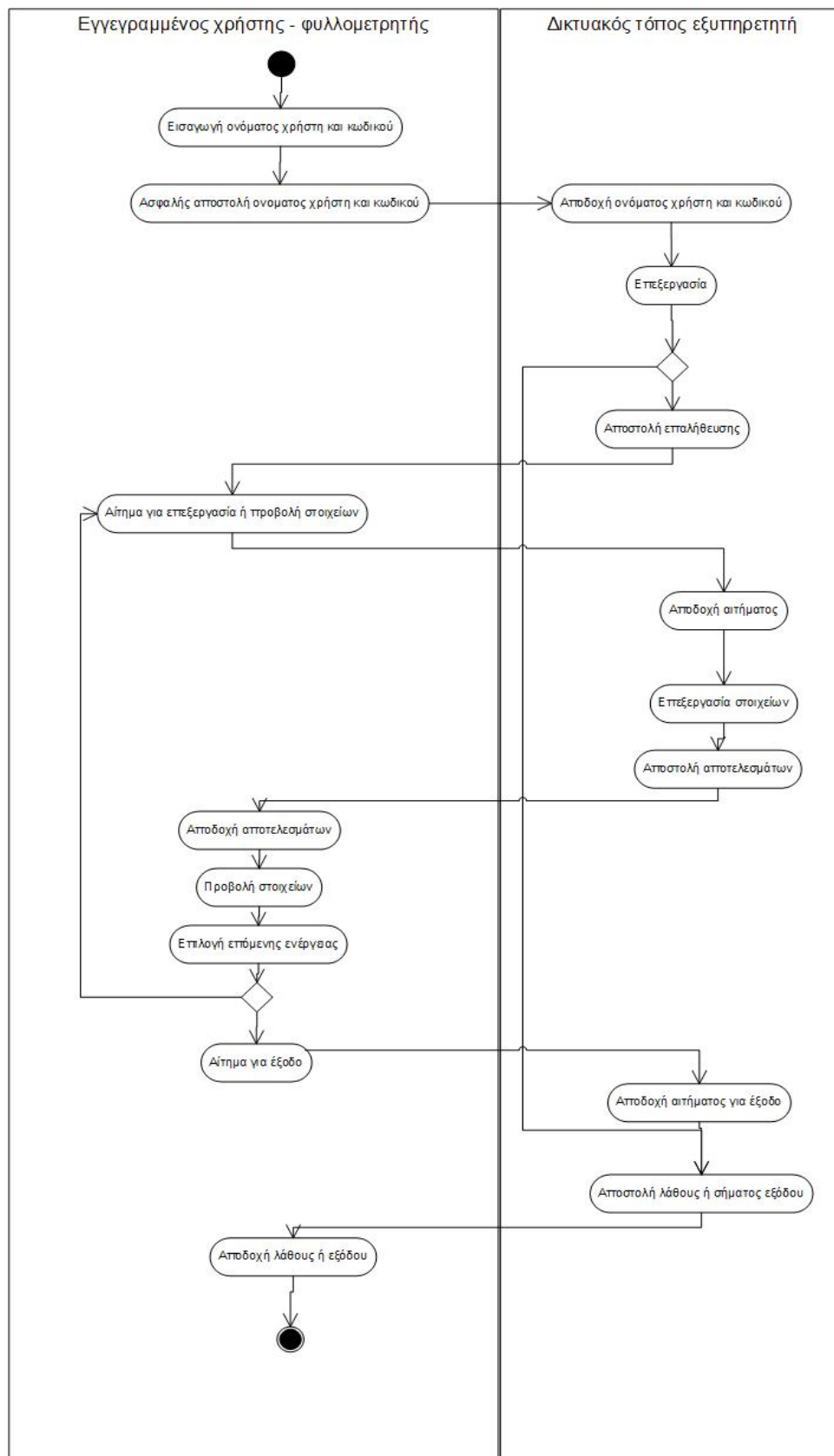
3.1.4.8.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο εγγεγραμμένος χρήστης του συστήματος εισέρχεται στο δικτυακό τόπο του συστήματος όπου έχει τη δυνατότητα να διαχειριστεί τα οχήματά του και να παρακολουθήσει τις χρεώσεις που του έχουν πραγματοποιηθεί

3.1.4.8.2 Ποή γεγονότων

3.1.4.8.2.1 Βασική Ποή - Web

1. Ο χρήστης θα εισέρχεται στο δικτυακό τόπο του συστήματος μέσω κάποιου φυλλομετρητή
2. Εισέρχεται στο σύστημα με ασφάλεια παρέχοντας το όνομα χρήστη και τον κωδικό του.
3. Του δίνεται η δυνατότητα να παρακολουθήσει την πορεία των χρεώσεων του, να επεξεργαστεί τα οχήματά του και τις κλήσεις που έχει δεχτεί.
4. Εξέρχεται από το σύστημα



Σχήμα 13: Ροή γεγονότων διαχείρισης οχημάτων - χρεώσεων από τον χρήστη

3.1.4.9 Έλεγχος συστήματος

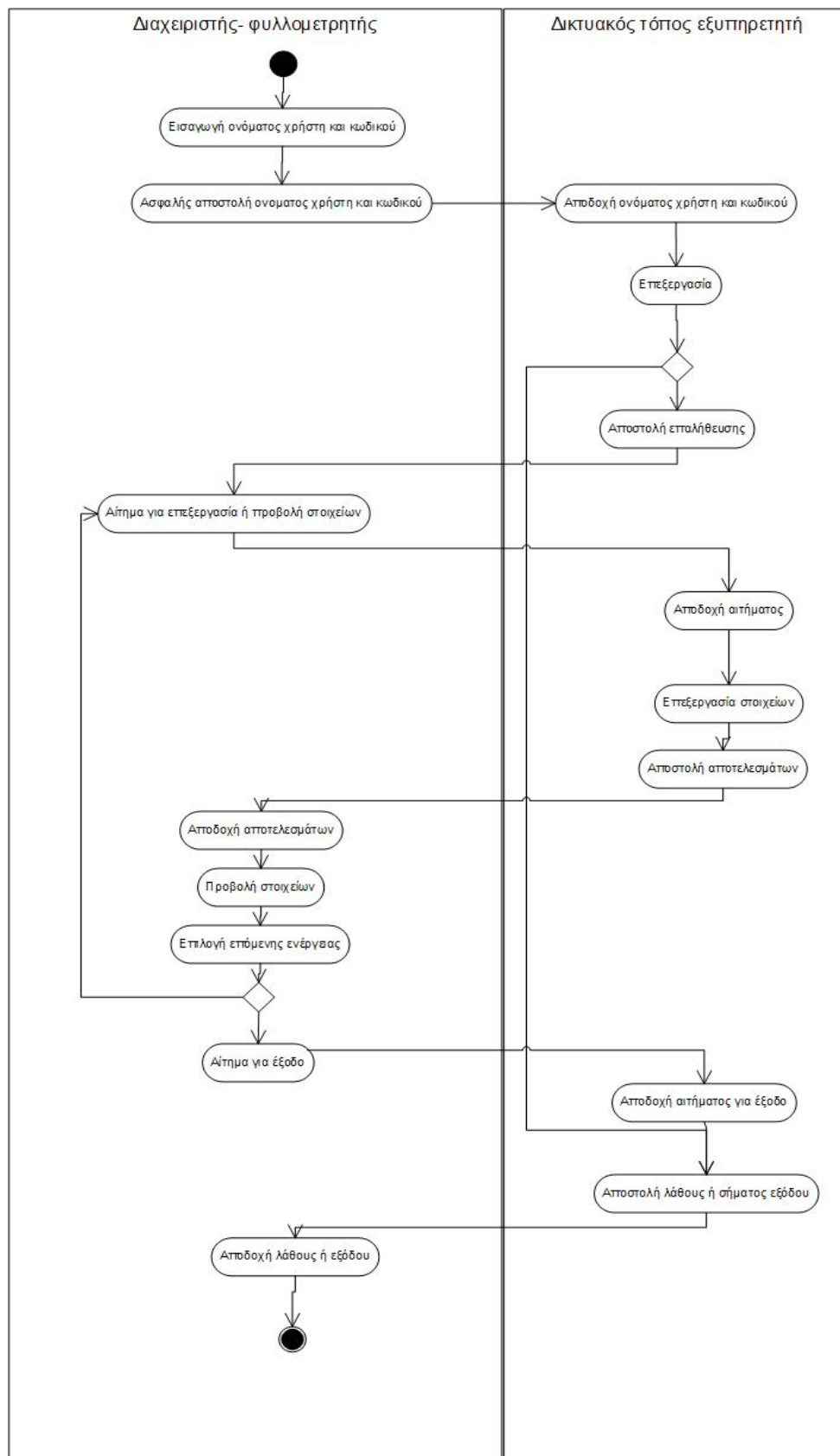
3.1.4.9.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο διαχειριστής του συστήματος εισέρχεται στο δικτυακό τόπο του συστήματος όπου έχει τη δυνατότητα να πραγματοποιήσει τον έλεγχο του συστήματος. Δηλαδή μπορεί να διαχειριστεί τους ελεγκτές του συστήματος, να ορίσει και να θέσει σε λειτουργία διαφορετικά σενάρια χρέωσης και να διαχειριστεί τα συστήματα χρέωσης που παρέχει το σύστημα (π.χ. προπληρωμένες κάρτες)

3.1.4.9.2 Ροή γεγονότων

3.1.4.9.2.1 Βασική Ροή - Web

1. Ο διαχειριστής εισέρχεται στο δικτυακό τόπο του συστήματος μέσω κάποιου φυλλομετρητή
2. Εισέρχεται στο σύστημα με ασφάλεια παρέχοντας το όνομα χρήστη και τον κωδικό του – τα οποία είναι διαφορετικά και αποθηκεύονται σε ξεχωριστή βάση δεδομένων από αυτά των χρηστών και των ελεγκτών.
3. Του δίνεται η δυνατότητα να διαχειριστεί τους ελεγκτές του συστήματος, να ορίσει και να θέσει σε λειτουργία διαφορετικά σενάρια χρέωσης και να διαχειριστεί τα συστήματα χρέωσης που παρέχει το σύστημα (π.χ. προπληρωμένες κάρτες)
4. Εξέρχεται από το σύστημα



Σχήμα 14 Ροή γεγονότων ελέγχου συστήματος

3.1.4.10 Παρακολούθηση στατιστικών.

3.1.4.10.1 Περιγραφή περίπτωσης χρήσης

Η περίπτωση αυτή αναφέρεται στη διαδικασία με την οποία ο διαχειριστής του συστήματος εισέρχεται στο δικτυακό τόπο του συστήματος όπου έχει τη δυνατότητα να παρακολουθήσει τα στατιστικά του συστήματος. Αυτό είναι ένα εξαιρετικά χρήσιμο στοιχείο καθώς δίνει την δυνατότητα για εξαγωγή συμπερασμάτων

3.1.4.10.2 Ροή γεγονότων

3.1.4.10.2.1 Βασική Ροή - Web

1. Ο διαχειριστής εισέρχεται στο δικτυακό τόπο του συστήματος μέσω κάποιου φυλλομετρητή
2. Εισέρχεται στο σύστημα με ασφάλεια παρέχοντας το όνομα χρήστη και τον κωδικό του – τα οποία είναι διαφορετικά και αποθηκεύονται σε ξεχωριστή βάση δεδομένων από αυτά των χρηστών και των ελεγκτών.
3. Του δίνεται η δυνατότητα να δει τα στατιστικά χρήσης του συστήματος κατά περίπτωση.
4. Εξέρχεται από το σύστημα

Το διάγραμμα δραστηριοτήτων περιλαμβάνεται στην προηγούμενη κατάσταση χρήσης.

3.2 Έγγραφο Περιγραφής Αρχιτεκτονικού Σχεδίου

3.2.1 Εισαγωγή

3.2.1.1 Σκοπός

Σκοπός των παρακάτω είναι η παρουσίαση της αρχιτεκτονικής του συστήματος ελεγχόμενης στάθμευσης. Αποτελεί τμήμα της σχεδίασης του συστήματος.

3.2.1.2 Εμβέλεια

Αποδέκτες θεωρούνται οι μηχανικοί λογισμικού που θα αναλάβουν την υλοποίηση του συστήματος και οι χρήστες του συστήματος – ιδίως η πλευρά του διαχειριστή.

3.2.1.3 Ορισμοί, ακρωνύμια, συντομογραφίες

Web Services: Δικτυακές Υπηρεσίες

UML: η γλώσσα που χρησιμοποιήθηκε κατά την σχεδίαση των διαγραμμάτων.

Χρήστης: ο τελικός χρήστης του συστήματος ελέγχου στάθμευσης, δηλαδή ο οδηγός.

Ελεγκτής: ο υπεύθυνος του συστήματος που επιφορτίζεται με το καθήκον να παρακολουθεί τη νόμιμη διαδικασία στάθμευσης.

Όχημα: η οντότητα που δεσμεύει μια θέση στάθμευσης. Μπορεί να είναι τετράτροχο ή δίτροχο – δεν έχει εισαχθεί αντίστοιχος περιορισμός κατά τη μελέτη του συστήματος.

Διαχειριστής: ο υπεύθυνος που ορίζεται από τον ιδιοκτήτη – επιβλέποντα του συστήματος ώστε να παρακολουθεί τις διαδικασίες ελέγχου στάθμευσης, και με τη βοήθεια των στατιστικών να μπορεί να τροποποιεί τα σενάρια χρήσης και γενικότερα τις παραμέτρους του συστήματος.

3.2.1.4 Βιβλιογραφία

Οι παρακάτω ενότητες περιέχουν αναφορές στη βιβλιογραφία που καταγράφεται στο τέλος του παρόντος τόμου.

3.2.2 Αρχιτεκτονικές Αποφάσεις

3.2.2.1 Πλατφόρμα

Η υλισμική πλατφόρμα του συστήματος χωρίζεται σε δύο μέρη. Αφενός χρειάζεται ένας αριθμός από συνεργαζόμενους ηλεκτρονικούς υπολογιστές οι οποίοι θα στεγάζουν τη βάση δεδομένων, το λογισμικό εξυπηρέτησης που θα εμπεριέχει τον δικτυακό τόπο του συστήματος, καθώς και τις δικτυακές υπηρεσίες. Τέλος, εκεί θα περιλαμβάνεται και η υπηρεσία αποστολής και λήψης των συντόμων μηνυμάτων (SMS). Το δεύτερο μέρος θα αποτελείται από το υλικό του πελάτη, το οποίο θα πρέπει να είναι μια ασύρματη φορητή συσκευή η οποία θα επιτρέπει την λειτουργία και ανάπτυξη προγραμμάτων σε πλατφόρμα Java 2 Micro Edition.

3.2.2.2 Γλώσσες – σχολή προγραμματισμού

Ως σχολή προγραμματισμού θα χρησιμοποιηθεί κατά βάση ο αντικειμενοστραφής προγραμματισμός, όπως αυτός συναντάται στις διαφορετικές γλώσσες και πλατφόρμες που συναποτελούν το σύστημα.

3.2.2.3 Κωδική γλώσσα

Στην εφαρμογή πελάτη θα χρησιμοποιηθεί η Java 2 Micro Edition, ενώ στο λογισμικό του εξυπηρετητή θα χρησιμοποιηθεί η πλατφόρμα .NET της Microsoft,, δηλαδή VB, C# και ASP .NET .

3.2.3 Αρχιτεκτονικές Όψεις

3.2.3.1 Περιβαλλοντική / Υπηρεσιακή

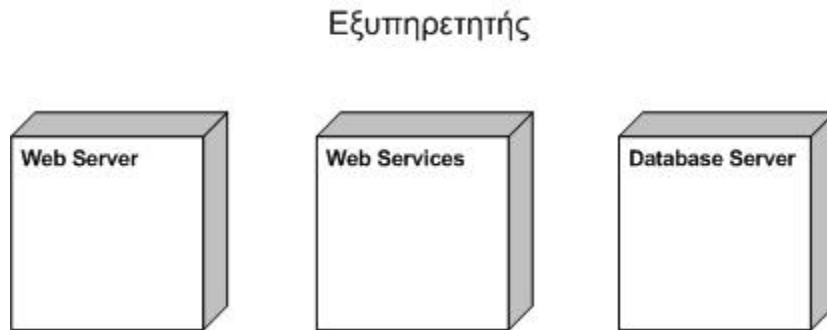
Σε αυτή την όψη φαίνονται οι δράστες – actors που εργάζονται στο σύστημα και οι βασικές δυνατότητες που τους παρέχει. Όπως και στην ανάλυση των απαιτήσεων, είναι σαφές πως δεν περιγράφονται όλες οι χρήσεις του συστήματος αλλά μόνον οι βασικές. Υπάρχουν και συμπληρωματικές ως προς την λειτουργία χρήσεις οι οποίες παραλείπονται από το παρακάτω διάγραμμα.



Σχήμα 15: Περιβαλλοντική - Υπηρεσιακή Όψη

3.2.3.2 Πλατφορμική

Παρακάτω βλέπουμε την πλατφορμική όψη του συστήματος όπως αυτή απεικονίζεται σε ένα deployment diagram (διάγραμμα ανάπτυξης). Περιλαμβάνονται οι υπολογιστές και οι συσκευές που θα λάβουν μέρος στην υλοποίηση του συστήματος. Τονίζεται πως κάποιες από τις λειτουργίες μπορούν να συστεγαστούν στην υλοποίηση στον ίδιο κόμβο.



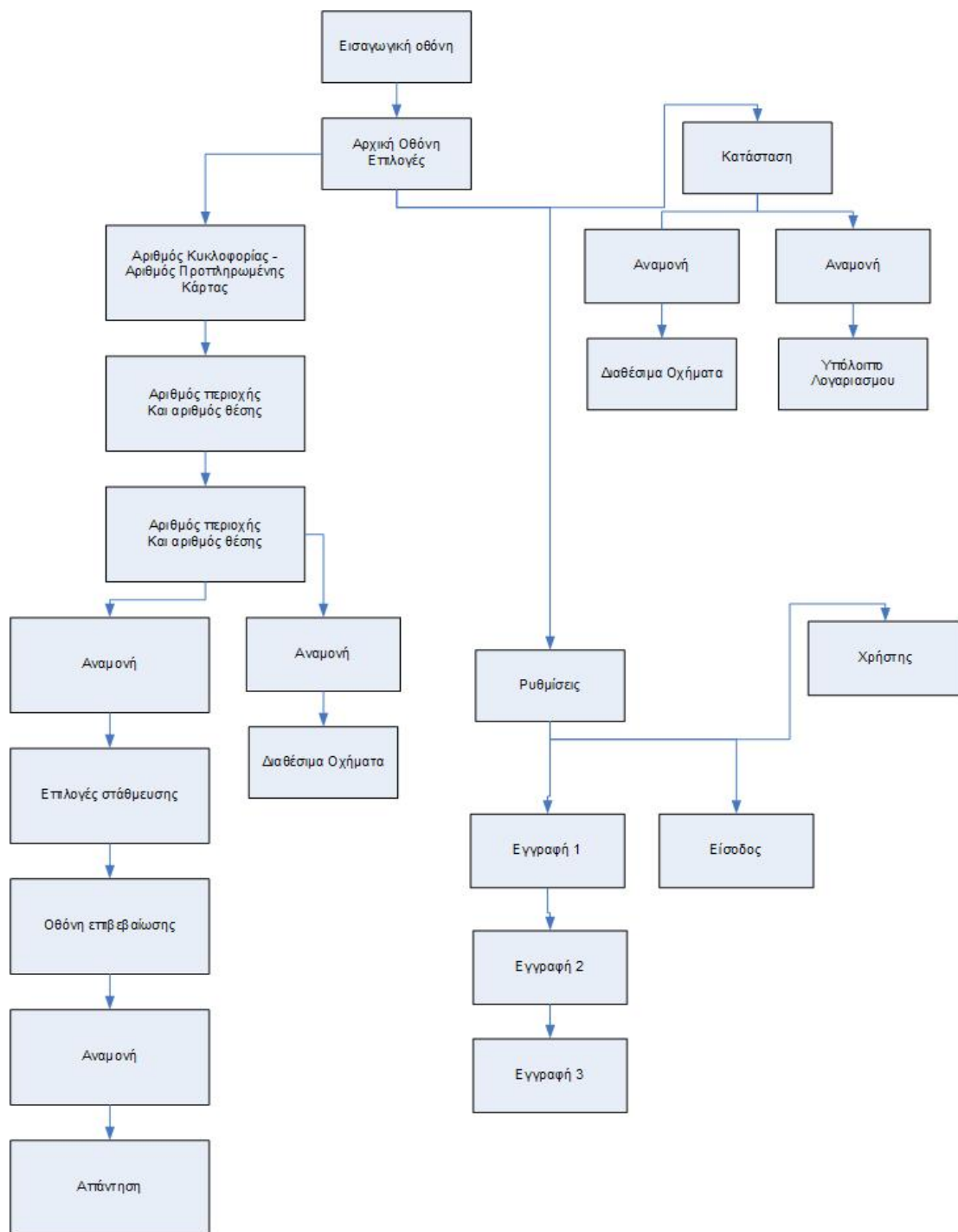
Σχήμα 16: Πλατφορμική όψη

3.2.3.3 Χρηστική

Σε αυτή την ενότητα παρουσιάζονται διάφορες οθόνες που χρησιμοποιεί ο χρήστης, ο ελεγκτής και ο διαχειριστής του συστήματος.

3.2.3.3.1 Σύστημα Πελάτη Χρήστη

Στο παρακάτω σχήμα φαίνεται το σύνολο των οθονών του User Interface του συστήματος που βλέπει ο χρήστης και ακολουθεί αναλυτική περιγραφή τους.



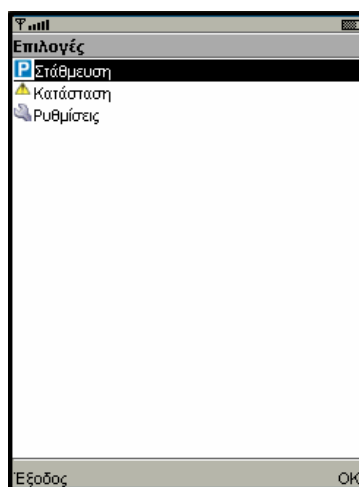
Σχήμα 17: Οθόνες του συστήματος πελάτη - χρήστη

Η παρακάτω οθόνη εμφανίζεται για λίγα δευτερόλεπτα πριν την εκκίνηση της εφαρμογής, και είναι ουσιαστικά η πρώτη οθόνη της εφαρμογής χρήστη.



Σχήμα 18: Χρηστική όψη - Εισαγωγική Οθόνη

Ακολουθεί η οθόνη που περιλαμβάνει τις βασικές επιλογές του χρήστη πελάτη. Οι επιλογές είναι αυτές που φαίνονται παρακάτω: στάθμευση, κατάσταση και ρυθμίσεις. Η στάθμευση οδηγεί στην αγορά χρόνου στάθμευσης για το παρόν όχημα. Η κατάσταση δίνει τη δυνατότητα για εύκολο έλεγχο της κατάστασης. Οι ρυθμίσεις περιλαμβάνουν το τμήμα της εφαρμογής όπου δίνει τις πλέον εξειδικευμένες λειτουργίες.

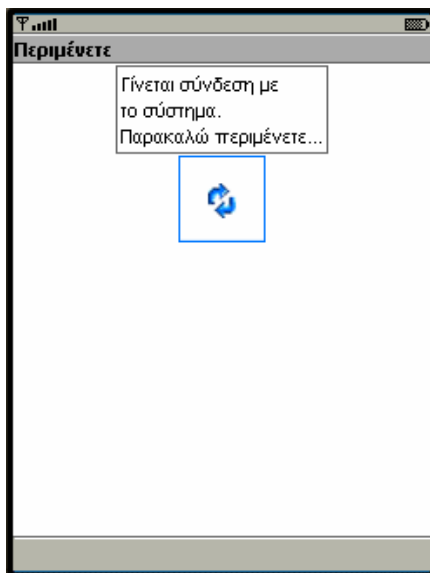


Σχήμα 19: Χρηστική όψη - Αρχική οθόνη επιλογών

Στην οθόνη που ακολουθεί ο πελάτης – χρήστης καλείται να εισάγει τον κωδικό αριθμό της περιοχής στην οποία θα γίνει η στάθμευση. Επίσης, προαιρετικά μπορεί να εισάγει τον κωδικό μιας συγκεκριμένης θέσης στην οποία θα σταθμεύσει το όχημα του.

Σχήμα 20: Χρηστική όψη -Οθόνη επιλογής περιοχής και αριθμού θέσης

Ακολουθεί μια οθόνη αναμονής, όσο πραγματοποιείται η σύνδεση του λογισμικού πελάτη με τον εξυπηρετητή του συστήματος.



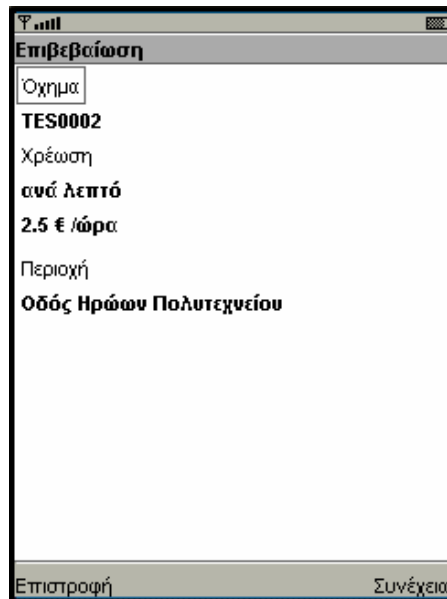
Σχήμα 21: Χρηστική όψη -Οθόνη αναμονής

Παρακάτω φαίνονται οι επιλογές που έχει ο χρήστης για στάθμευση στη συγκεκριμένη περιοχή. Οι επιλογές και η χρέωση παρελήφθησαν δυναμικά από το σύστημα.



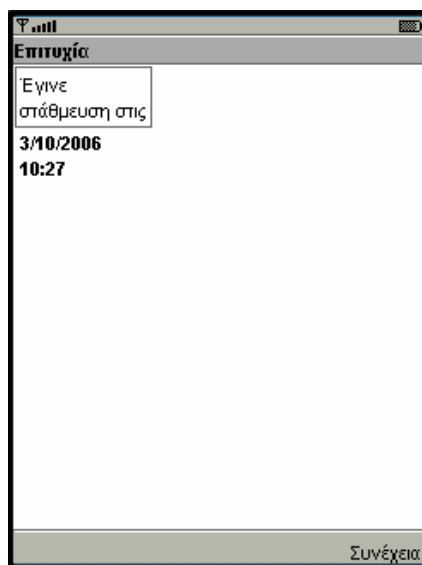
Σχήμα 22: Χρηστική όψη - Επιλογές στάθμευσης

Ο χρήστης επιλέγει μια μεθοδολογία χρέωσης, στη συγκεκριμένη περίπτωση χρέωση ανά λεπτό. Δηλαδή η χρέωση θα πραγματοποιείται ανά λεπτό μέχρι ο χρήστης αποφασίσει να την τερματίσει, αναχωρώντας από τη θέση στάθμευσης.. Εδώ ζητείται μια τελική επιβεβαίωση.



Σχήμα 23 Χρηστική όψη - Οθόνη επιβεβαίωσης

Ακολουθεί το αποτέλεσμα της αίτησης για εκκίνηση στάθμευσης. Στη συγκεκριμένη περίπτωση η έκβαση είναι επιτυχής, και το λογισμικό μας παρουσιάζει την ακριβή ώρα έναρξης της στάθμευσης.



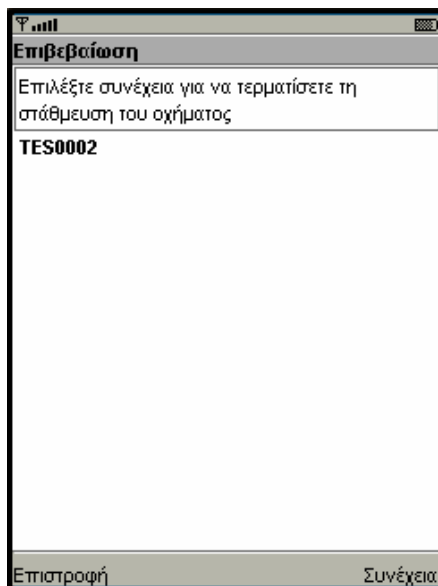
Σχήμα 24: Χρηστική όψη -Οθόνη απάντησης

Με την οθόνη επιβεβαίωσης της στάθμευσης παρατηρούμε ότι η αρχική λίστα επιλογών του συστήματος αλλάζει και η στάθμευση γίνεται τερματισμός αφού πλέον δίνει τη δυνατότητα για τερματισμό της υπάρχουσας στάθμευσης – και όχι για έναρξη.



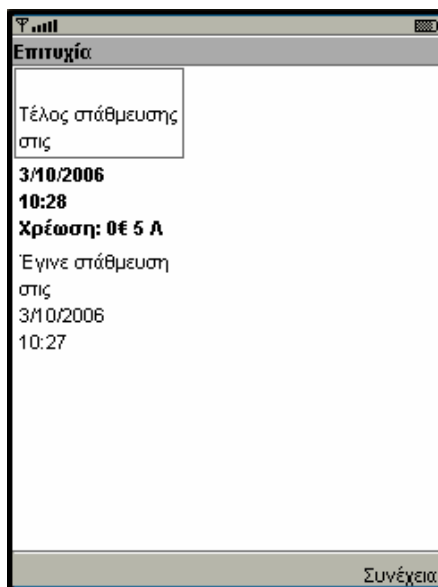
Σχήμα 25: Χρηστική όψη - Οθόνη επιλογών

Με επιλογή του τερματισμού προκύπτει η παρακάτω οθόνη που ζητά επιβεβαίωση από το χρήστη για τη λήξη της στάθμευσης του οχήματος του.



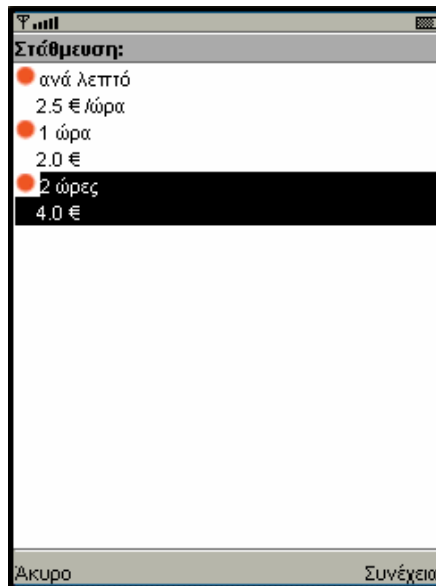
Σχήμα 26: Χρηστική όψη - Οθόνη επιβεβαίωσης τερματισμού

Μετά το τέλος της στάθμευσης απεικονίζεται αναλυτικά μια οθόνη η οποία περιλαμβάνει τον χρόνο έναρξης και λήξης της στάθμευσης καθώς και τη συνολική χρέωση του χρήστη. Ενδιαμέσως μεσολαβεί μια οθόνη αναμονής όπως είδαμε παραπάνω.



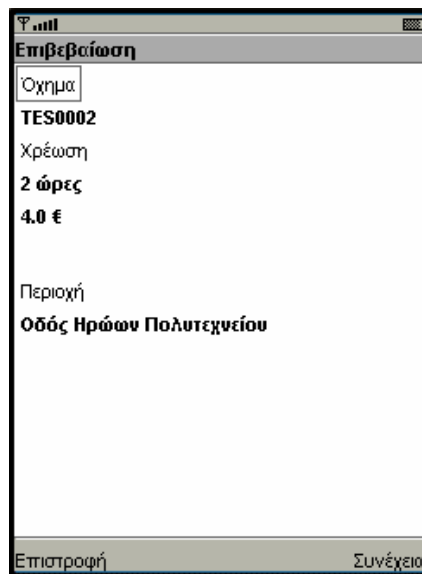
Σχήμα 27: Χρηστική όψη - Οθόνη απάντησης τερματισμού

Ας δούμε τώρα τι αλλάζει στην περίπτωση που ο χρήστης επιλέξει την αγορά συγκεκριμένου χρόνου στάθμευσης, π.χ. 2 ώρες.



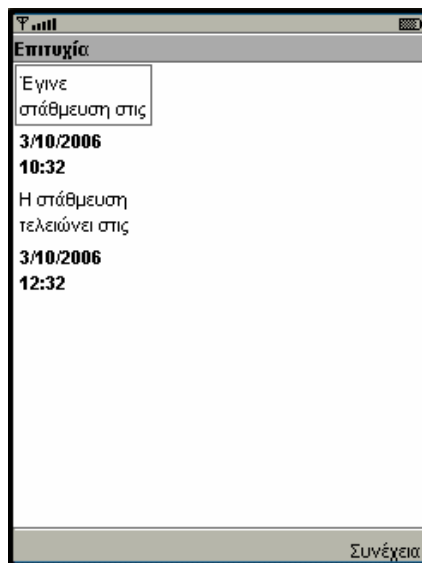
Σχήμα 28: Χρηστική όψη - Οθόνη επιλογών στάθμευσης

Όπως και προηγουμένως, εμφανίζεται μια οθόνη επιβεβαίωσης που εμφανίζει την χρέωση αλλά και την διάρκεια της στάθμευσης στη συγκεκριμένη περιοχή.



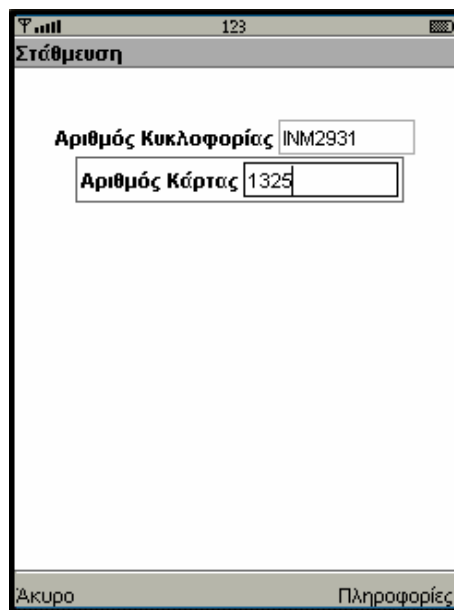
Σχήμα 29: Χρηστική όψη - Οθόνη επιβεβαίωσης

Με την επιβεβαίωση από τον χρήστη η οθόνη που εμφανίζεται περιγράφει αναλυτικά την έναρξη και την λήξη της στάθμευσης.



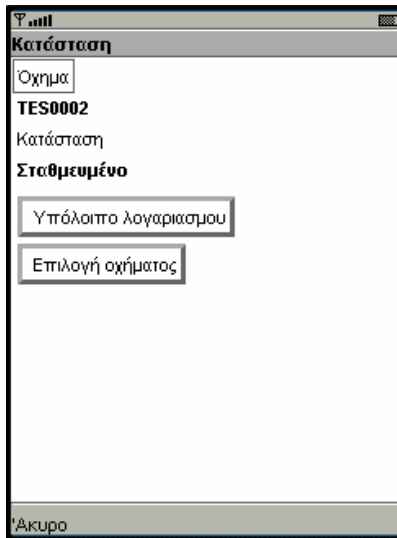
Σχήμα 30: Χρηστική όψη - Οθόνη απάντησης στάθμευσης

Στην περίπτωση που ο χρήστης είναι μη εγγεγραμμένος στο σύστημα τότε πριν την έναρξη της στάθμευσης το λογισμικό του ζητά δυο ακόμη παραμέτρους, τον αριθμό κυκλοφορίας του οχήματος και τον αριθμό της προπληρωμένης κάρτας του.



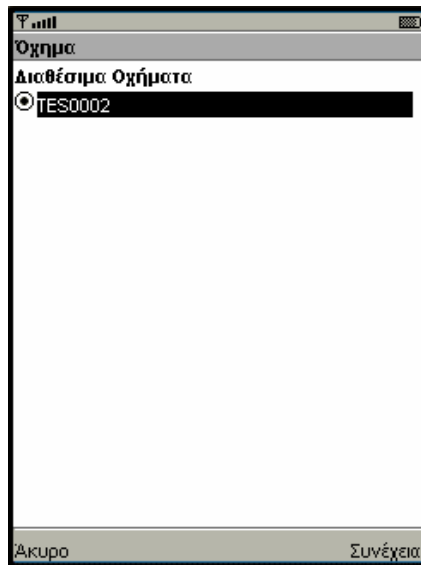
Σχήμα 31: Χρηστική όψη -Οθόνη εισαγωγής αριθμού κυκλοφορίας και αριθμού προπληρωμένης κάρτας

Στην επιλογή κατάσταση ο χρήστης μπορεί να ελέγξει την κατάσταση που βρίσκεται το όχημα του αυτή τη στιγμή.



Σχήμα 32: Χρηστική όψη - Οθόνη κατάστασης

Δίνεται επίσης η δυνατότητα για επιλογή οχήματος. Όπως φαίνεται στην παρακάτω οθόνη εδώ ο χρήστης μπορεί να επιλέξει το τρέχον όχημά του από μια λίστα με τα οχήματα που έχει δηλώσει στο σύστημα.



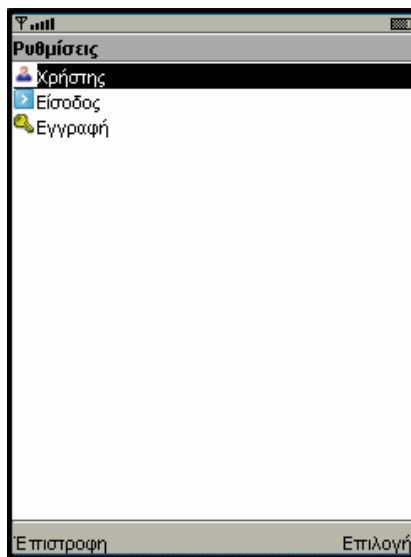
Σχήμα 33: Χρηστική όψη - Οθόνη επιλογής μεταξύ διαθέσιμων οχημάτων

Ακόμη, όπως φαίνεται στην παρακάτω οθόνη μπορεί να ελέγξει το υπόλοιπο του λογαριασμού του, όταν αυτός είναι συνδεδεμένος με κάποια προπληρωμένη κάρτα.



Σχήμα 34: Χρηστική όψη - Οθόνη υπόλοιπου λογαριασμού

Η επιλογή «Ρυθμίσεις» μας οδηγεί στην παρακάτω οθόνη η οποία δίνει τις επιλογές «Χρήστης», «Είσοδος» και «Εγγραφή».



Σχήμα 35: Χρηστική όψη - Οθόνη ρυθμίσεων

Οι ρυθμίσεις χρήστη περιλαμβάνουν τις ρυθμίσεις που χρησιμοποίησε στις σταθμεύσεις του για τελευταία φορά ο χρήστης και είναι αποθηκευμένες στο κινητό τηλέφωνο ώστε να μην χρειάζεται να τις εισάγει κάθε φορά από την αρχή.

Σχήμα 36 :Χρηστική όψη - Οθόνη ρυθμίσεων χρήστη

Η επιλογή είσοδος δίνει την δυνατότητα σε έναν χρήστη να εισέλθει στο σύστημα με τον λογαριασμό του, ακόμη και αν η εφαρμογή δεν έχει ρυθμιστεί για αυτό τον τρόπο χρήσης κατά την αρχική εγκατάσταση. Αν το πεδίο αφεθεί κενό, τότε ο χρήστης θεωρείται από το σύστημα ως μη εγγεγραμμένος.

Σχήμα 37: Χρηστική όψη - Οθόνη εισόδου χρήστη

Επιλέγοντας εγγραφή το λογισμικό πελάτη ζητά μια σειρά από στοιχεία προκειμένου να πραγματοποιήσει την εγγραφή. Ζητάει λοιπόν το όνομα & το επώνυμο, καθώς και το όνομα χρήστη & τον κωδικό που ισχύει για την είσοδο τόσο με το λογισμικό πελάτη, όσο και στον δικτυακό τόπο του συστήματος.

Εγγραφή 1 από 3

Όνομα
Κωνσταντίνος

Επώνυμο
Χρηστίδης

Όνομα χρήστη
[Empty]

Κωδικός
[Empty]

Επιστροφή

Σχήμα 38: Χρηστική όψη - Οθόνη εγγραφής (1)

Ακολούθως, ζητείται ο αριθμός κυκλοφορίας του κύριου οχήματος του χρήστη και επιλέγεται η μέθοδος χρέωσης.

Εγγραφή 2 από 3

Αριθμός Κυκλοφορίας INIK2938

Πληρωμή

☒ χρεωστική κάρτα

☐ πιστωτική κάρτα

☐ προπληρωμένη κάρτα

Επιστροφή Συνέχεια

Σχήμα 39: Χρηστική όψη - Οθόνη εγγραφής (2)

Τέλος, ζητούνται τα στοιχεία της κάρτας όπως ο αριθμός κάρτας, το ονοματεπώνυμο κατόχου όπως αυτό αναγράφεται σε αυτήν και η ημερομηνία λήξης της.

Εγγραφή 3 από 3

Αριθμός Κάρτας
1938408849384937

Όνοματεπώνυμο
A.B.ODHGOS

Ημερομηνία Λήξης
15/05/2008

Επιστροφή Συνέχεια

Σχήμα 40: Χρηστική όψη - Οθόνη εγγραφής (3)

Η διαδικασία της εγγραφής τελειώνει με την απάντηση του συστήματος. Στην παρακάτω οθόνη η εγγραφή έγινε με επιτυχία και έτσι δίνεται και το όνομα του χρήστη που μόλις εγγράφηκε.

Επιτυχία

Η εγγραφή σας πραγματοποιήθηκε με
όνομα χρήστη: testuser1

Συνέχεια

Σχήμα 41: Χρηστική όψη - Οθόνη απάντησης στην εγγραφή

3.2.3.4 Σύστημα Πελάτη Ελεγκτή

Στο παρακάτω σχήμα φαίνεται το σύνολο των οθονών για το λογισμικό σύστημα πελάτη που χρησιμοποιεί ο ελεγκτής του συστήματος. Ακολουθεί αναλυτική περιγραφή της κάθε οθόνης.



Σχήμα 42: Οθόνες του συστήματος πελάτη - ελεγκτή

Στην πρώτη οθόνη της εφαρμογής του ελεγκτή, όπως αυτή φαίνεται παρακάτω, εισάγεται ο αριθμός κυκλοφορίας του οχήματος που ελέγχεται.

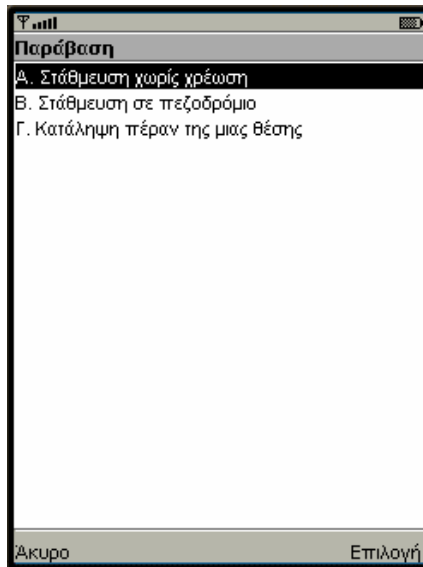
Σχήμα 43: Χρηστική Όψη - Οθόνη εισαγωγής αριθμού κυκλοφορίας

Ακολουθώντας, ο ελεγκτής παρατηρεί στην οθόνη του κινητού του αν το όχημα έχει με κάποιον τρόπο δεσμεύσει χρόνο στάθμευσης στον δρόμο που βρίσκεται. Βλέπει με άλλα λόγια αν το όχημα είναι νόμιμα ή παράνομα σταθμευμένο.

Σχήμα 44: Χρηστική Όψη - Οθόνη αποτελεσμάτων ελέγχου

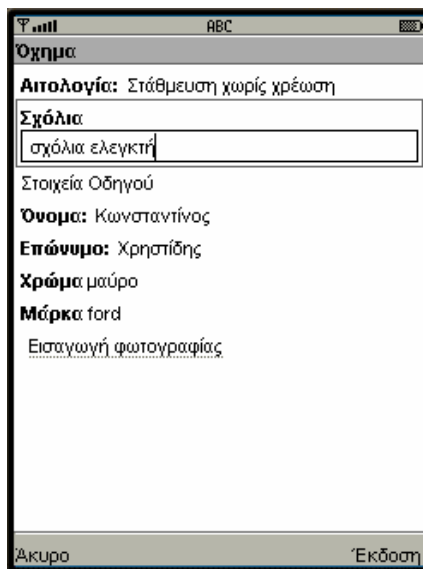
Ο ελεγκτής εδώ αποφασίζει αν θέλει να εκδώσει κλήση ή όχι. Επειδή ο λόγος έκδοσης της κλήσης μπορεί να μην είναι άλλος από αυτόν της μη πληρωμής χρόνου στάθμευσης, ο ελεγκτής μπορεί να

επιλέξει ελεύθερα την επόμενη του ενέργεια. Έτσι έρχεται σε μια οθόνη που επιλέγει ποια παράβαση έχει πραγματοποιηθεί στην παρούσα περίπτωση.



Σχήμα 45: Χρηστική Όψη - Οθόνη επιλογής τύπου παράβασης

Τέλος, ο ελεγκτής συμπληρώνει προαιρετικά σχόλια για την κλήση που θα εκδοθεί. Η κλήση έχει καταχωρηθεί στο σύστημα και συνδεθεί με το όχημα.



Σχήμα 46: Χρηστική Όψη - Οθόνη προσθήκης σχολίων

Τελικά στην οθόνη εμφανίζονται όλα τα στοιχεία της κλήσης και δίνεται η δυνατότητα για εκτύπωση.

Κλήση για Παράνομη Στάθμευση

Κλήση για Παράνομη Στάθμευση

Αιτιολογία: Στάθμευση χωρίς χρέωση

Σχόλια: σχόλια ελεγκτή

Στοιχεία Υπεύθυνου Οδηγού

Όνομα:

Επώνυμο:

Χρώμα: μαύρο

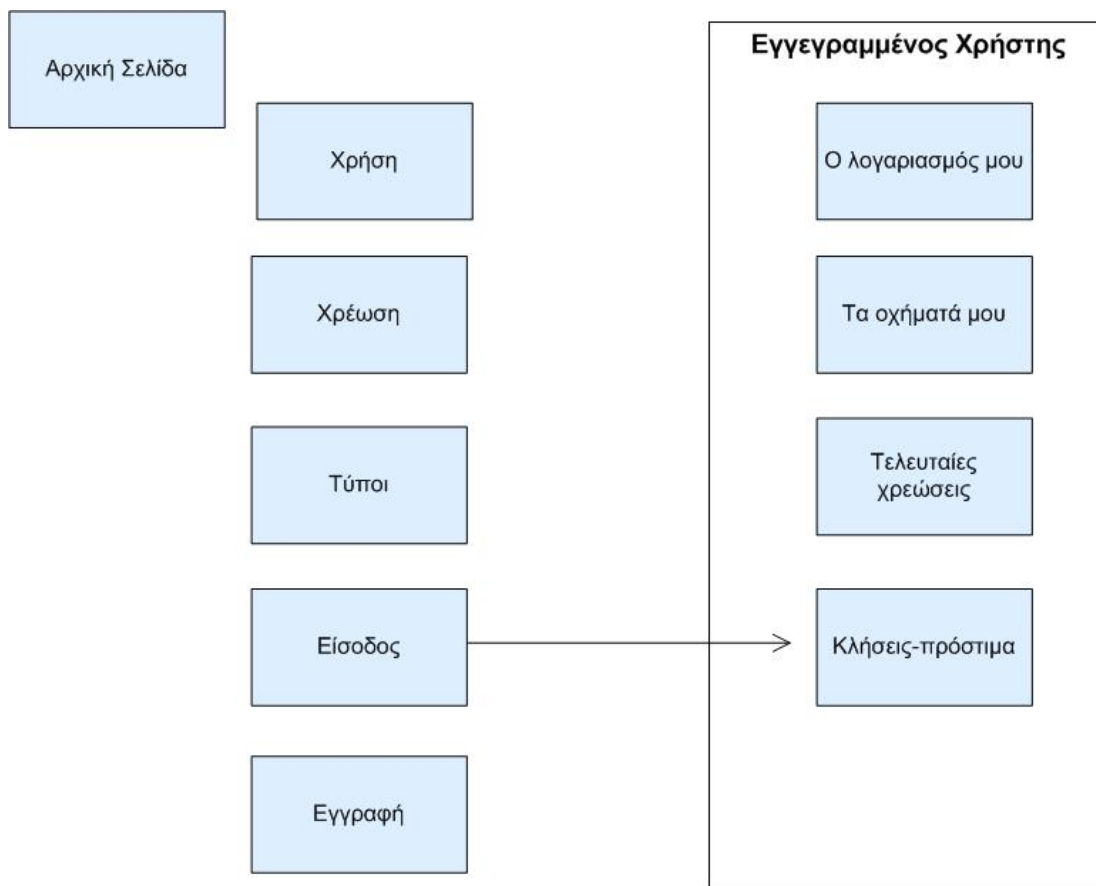
Μάρκα: ford

Άκυρο Εκτύπωση

Σχήμα 47: Χρηστική Όψη - Οθόνη έκδοσης κλήσης

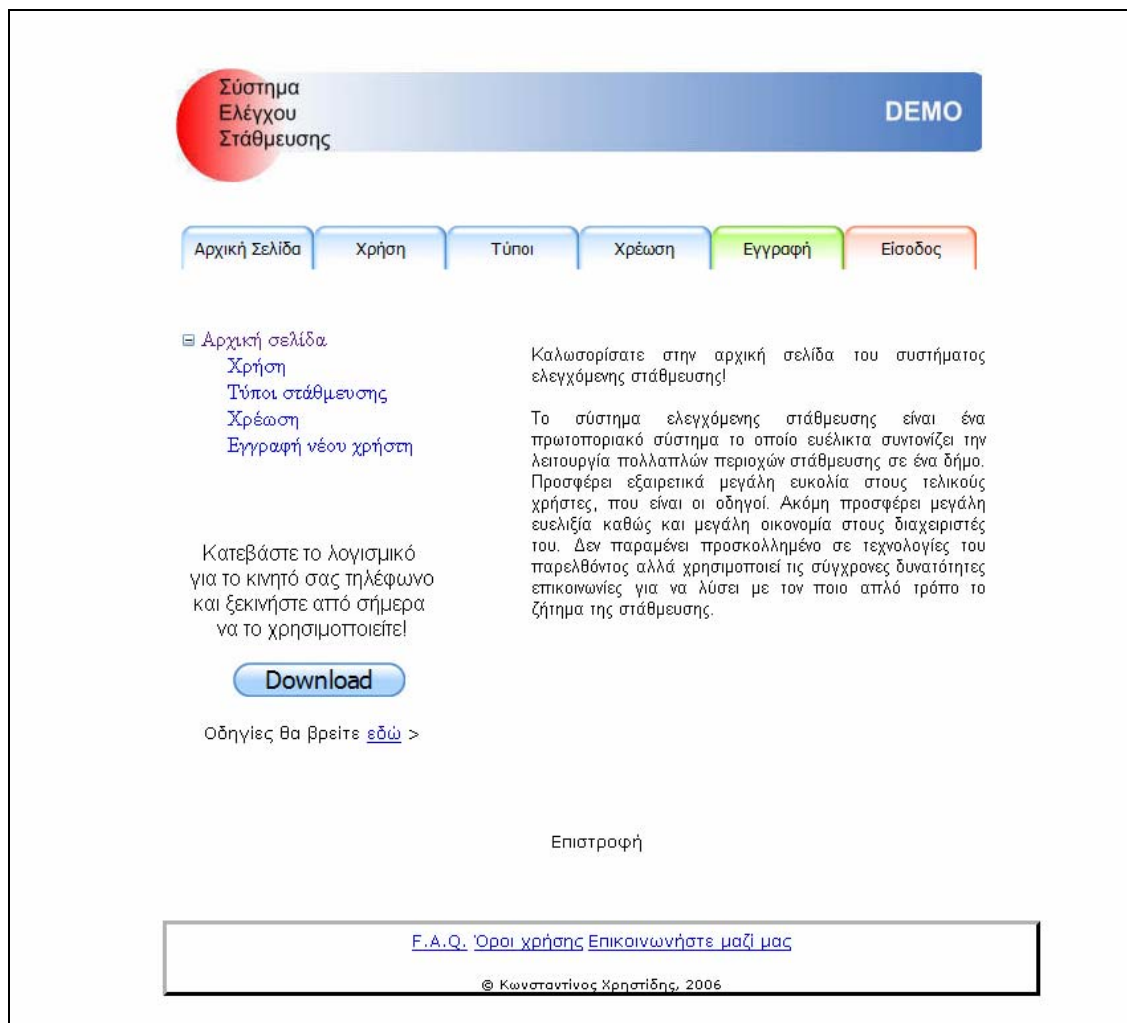
3.2.3.5 Σύστημα Δικτυακού Τόπου

Στο παρακάτω σχήμα φαίνεται ένα αναλυτικό διάγραμμα του δικτυακού τόπου τόσο για τον χρήστη όσο και για τον διαχειριστή. Ακολουθούν μερικές ενδεικτικές οθόνες – σελίδες web που δείχνουν την φιλοσοφία πλοήγησης στον τόπο.



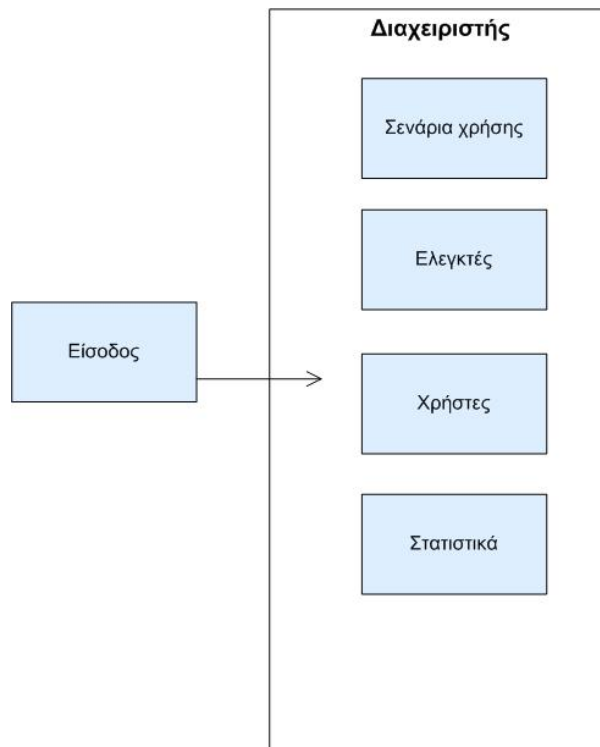
Σχήμα 48: Ο δικτυακός τόπος για τον χρήστη

Στη συνέχεια παρατίθεται η αρχική εικόνα του διαδικτυακού τόπου, όπου ο χρήστης μπορεί να ενημερωθεί για διάφορα θέματα λειτουργίας και εφαρμογής του συστήματος ελεγχόμενης στάθμευσης, να εισαχθεί στο σύστημα σε περίπτωση που διαθέτει κωδικό χρήστη, ή να καταχωρήσει τα στοιχεία του προκειμένου να λάβει κωδικό χρήστη.



Σχήμα 49: Αρχική σελίδα του διαδικτυακού τόπου

Αφού εισέλθει, ο χρήστης μπορεί να περιηγηθεί στη σελίδα, να παρατηρήσει και να επεξεργαστεί τις προσωπικές του ρυθμίσεις.



Σχήμα 51: Ο δικτυακός τύπος για τον διαχειριστή

Σύστημα
Ελέγχου
Στάθμευσης

DEMO

Σενάρια Χρήσης

Χρήση

Προπληρωμένες κάρτες

Πρόγραμμα

Ελεγκτές

Χρήστες

Διαχειριστικό Περιβάλλον

Οχήματα που συμμετέχουν στο σύστημα ελέγχου στάθμευσης.

title	value
Toyota	Toyota
Ford	Ford
Mercedes	Mercedes
Renault	Renault
Opel	Opel
BMW	BMW
Chevrolet	Chevrolet
Hyundai	Hyundai
Rover	Rover
Suzuki	Suzuki
1 2	
title	Toyota
value	Toyota
New	

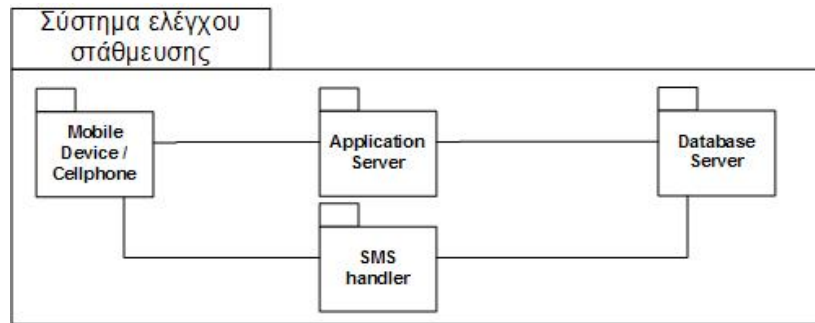
[F.A.Q.](#)
[Όροι χρήσης](#)
[Επικοινωνήστε μαζί μας](#)

© Κωνσταντίνος Χρηστίδης, 2006

Σχήμα 52: Παράδειγμα σελίδας διαχείρισης συστήματος

3.2.3.6 Δομική

Σε αυτή την όψη φαίνονται τα τμήματα του συστήματος σε μακροσκοπικό επίπεδο καθώς και οι εξαρτήσεις τους.

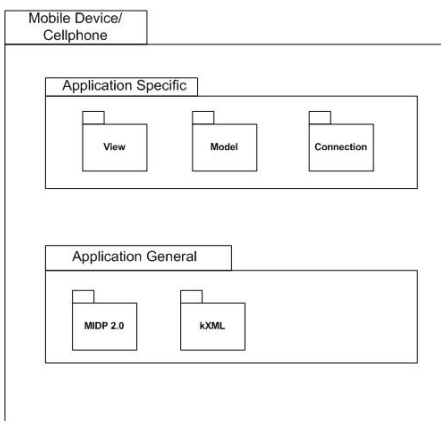


Σχήμα 53: Δομική όψη του συστήματος ελέγχου στάθμευσης

Το σύστημα ελέγχου στάθμευσης μπορεί να χωριστεί σε 4 μέρη:

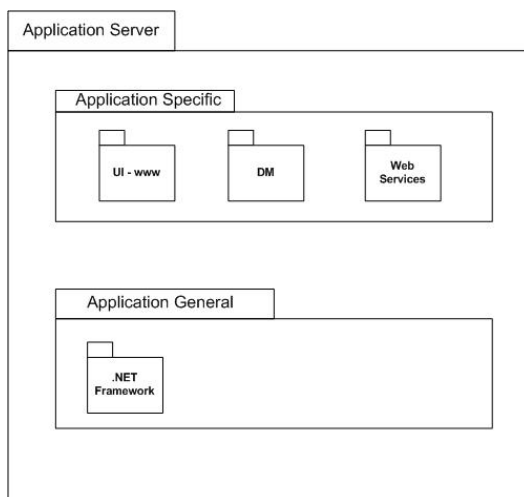
- Mobile Device / Cellphone που αναφέρεται στην ψηφιακή συσκευή – κινητό τηλέφωνο η οποία περιέχει την εφαρμογή πελάτη.
- Application Server, ο οποίος περιέχει όλα τα επιμέρους συστήματα που αφορούν τις δικτυακές υπηρεσίες και τον δικτυακό τόπο.
- SMS handler, που περιέχει τις διαδικασίες ανάλυσης και απάντησης στα μηνύματα των χρηστών.
- Database Server που περιέχει όλες τις πληροφορίες του συστήματος υπό μορφή βάσης δεδομένων. Παρακάτω βλέπουμε τις δομικές όψεις κάθε τμήματος ξεχωριστά

Στο παρακάτω σχήμα βλέπουμε ότι η φορητή συσκευή από την πλευρά του πελάτη περιλαμβάνει κλάσεις που χωρίζονται στο View- Model – Connection, ενώ το γενικό της μέρος χρησιμοποιεί το MIDP 2.0 και το kXML.



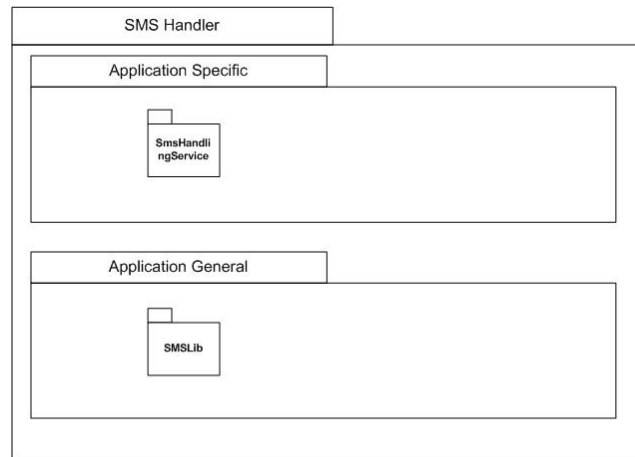
Σχήμα 54: Δομική όψη από την πλευρά του πελάτη

Οι δύο παρακάτω όψεις αναφέρονται εν γένει στην πλευρά του εξυπηρετητή. Ο Application Server στο Application Specific τμήμα του χωρίζεται κι αυτός με τη σειρά του σε 3 μέρη. Στο περιβάλλον χρήστη (UI), στο DM και στις δικτυακές υπηρεσίες. Ακόμη χρησιμοποιεί το .NET Framework.



Σχήμα 55: Δομική όψη του εξυπηρετητή εφαρμογών από την πλευρά του εξυπηρετητή

Ο SMSHandler αποτελεί μια αυτόνομη και ανεξάρτητη οντότητα. Χρησιμοποιεί τις βιβλιοθήκες ανοιχτού κώδικα SMSLib για τη σύνδεση με το κινητό τηλέφωνο ή το GSM modem που θα λαμβάνει τα μηνύματα.



Σχήμα 56: Δομική όψη του διαχειριστή συντόμων μηνυμάτων από την πλευρά του εξυπηρετητή

3.3 Έγγραφο Περιγραφής Λεπτομερούς Σχεδίου

3.3.1 Εισαγωγή

3.3.1.1 Σκοπός

Σκοπός των παρακάτω είναι η παρουσίαση του λεπτομερούς σχεδίου του συστήματος ελεγχόμενης στάθμευσης. Αποτελεί προϊόν της λεπτομερούς σχεδίασης του συστήματος.

3.3.1.2 Εμβέλεια

Αποδέκτες θεωρούνται οι μηχανικοί λογισμικού που θα αναλάβουν την υλοποίηση του συστήματος και οι διαχειριστές του συστήματος.

3.3.1.3 Ορισμοί, ακρωνύμια, συντομογραφίες

Web Services: Δικτυακές Υπηρεσίες

UML: η γλώσσα που χρησιμοποιήθηκε κατά την σχεδίαση των διαγραμμάτων.

Χρήστης: ο τελικός χρήστης του συστήματος ελέγχου στάθμευσης, δηλαδή ο οδηγός.

Ελεγκτής: ο υπεύθυνος του συστήματος που επιφορτίζεται με το καθήκον να παρακολουθεί τη νόμιμη διαδικασία στάθμευσης.

Όχημα: η οντότητα που δεσμεύει μια θέση στάθμευσης. Μπορεί να είναι τετράτροχο ή δίτροχο – δεν έχει εισαχθεί αντίστοιχος περιορισμός κατά τη μελέτη του συστήματος.

Διαχειριστής: ο υπεύθυνος που ορίζεται από τον ιδιοκτήτη – επιβλέποντα του συστήματος ώστε να παρακολουθεί τις διαδικασίες ελέγχου στάθμευσης, και με τη βοήθεια των στατιστικών να μπορεί να τροποποιεί τα σενάρια χρήσης και γενικότερα τις παραμέτρους του συστήματος.

3.3.1.4 Βιβλιογραφία

Οι παρακάτω ενότητες περιέχουν αναφορές στη βιβλιογραφία που καταγράφεται στο τέλος του παρόντος τόμου.

3.3.2 Σχεδιαστικές Αποφάσεις

Στις σχεδιαστικές αποφάσεις του συστήματος συγκαταλέγονται οι αποφάσεις που έγιναν κατά την επιλογή των προτύπων που χρησιμοποιήθηκαν. Τα πρότυπα αυτά, με βασικότερα εκείνα των δικτυακών υπηρεσιών και της XML, αναλύονται στην § 2.1. Επίσης, σχεδιαστική απόφαση αποτελεί η επιλογή των γλωσσών προγραμματισμού όπως αυτή περιγράφεται στην §4.1. Τέλος, η αποσύνθεση του συστήματος όπως περιγράφεται στις ακόλουθες ενότητες, αποτελεί προϊόν σχεδιαστικών αποφάσεων που έχουν σκοπό την ευελιξία, διαλειτουργικότητα και επεκτασιμότητα του συστήματος.

3.3.3 Σχεδιαστικά Μορφήματα

Στο λογισμικό πελάτη χρησιμοποιήθηκε το σχεδιαστικό μόρφημα Model-View. Στα υπόλοιπα συστήματα εφαρμόστηκαν απλώς οι αρχές του αντικειμενοστραφούς προγραμματισμού.

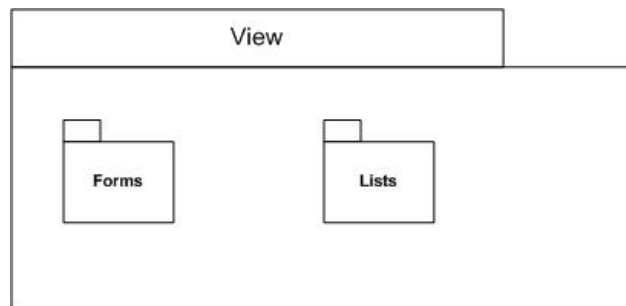
3.3.4 Σχεδιαστικές Όψεις

3.3.4.1 Αποσυνθετική Όψη

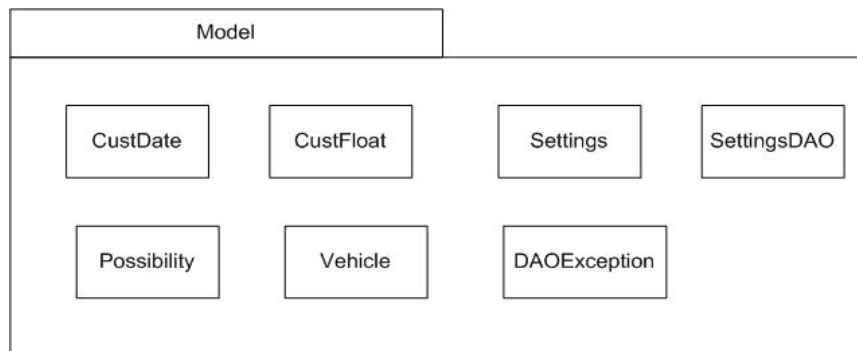
Παρακάτω αναλύεται η αποσυνθετική όψη των κόμβων του συστήματος όπως αυτοί αναλύθηκαν παραπάνω.

3.3.4.1.1 Mobile Device / Cellphone

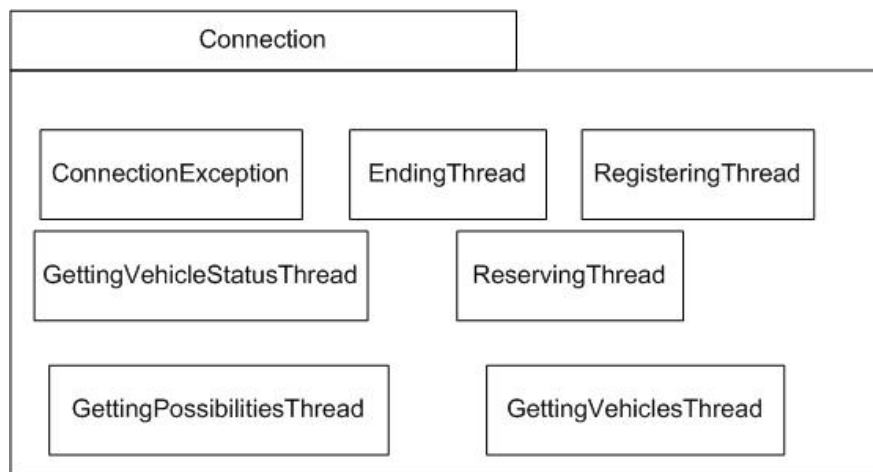
Το λογισμικό της φορητής συσκευής αποσυντίθεται σε τρία τμήματα όπως είδαμε στην παράγραφο 3.2.3.6. Το τμήμα View περιέχει όλες τις φόρμες και τις λίστες που βλέπει ο χρήστης κατά την χρήση του προγράμματος. Το τμήμα Model περιέχει όλες τις κλάσεις δεδομένων αλλά και βοηθητικές για την επεξεργασία τους. Τέλος το τμήμα Connection περιέχει τις κλάσεις που πραγματοποιούν τη σύνδεση με τον Server του συστήματος.



Σχήμα 57: Αποσυνθετική όψη του View του λογισμικού πελάτη για φορητή συσκευή



Σχήμα 58: Αποσυνθετική όψη του Model του λογισμικού πελάτη για φορητή συσκευή



Σχήμα 59:Αποσυνθετική όψη του **Connection** του λογισμικού πελάτη για φορητή συσκευή

Το λογισμικό της συσκευής απαιτεί την ύπαρξη του MIDP 2.0 και χρησιμοποιεί τη βιβλιοθήκη ανοιχτού κώδικα kXML για την επεξεργασία των XML μηνυμάτων.



Σχήμα 60: Αποσυνθετική όψη των ανεξάρτητων τμημάτων του λογισμικού πελάτη για φορητή συσκευή

3.3.4.1.2 Application Server

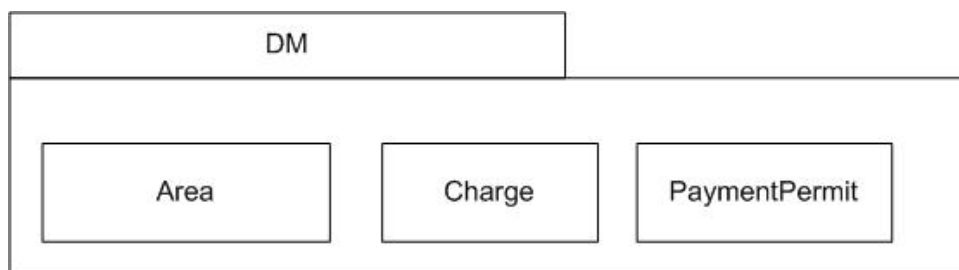
Ο Application Server του συστήματος χωρίζεται και αυτός με τη σειρά του στα UI-www, Web Services και DM. Το UI πέρα από τις ASP σελίδες περιλαμβάνει μια νέα υλοποίηση CustomMembershipProvider και ένα επιπλέον στοιχείο για χρήση στις σελίδες. Οι Web Services αποτελούν ξεχωριστή ενότητα και υλοποιούνται με βάση το Visual Studio .NET. Το DM είναι το τμήμα που περιέχει τις κλάσεις για την εισαγωγή, επεξεργασία και προβολή των οντοτήτων στη βάση δεδομένων. Τέλος, το .NET Framework παρέχει τις απαραίτητες κλάσεις για την ανάπτυξη των σελίδων.



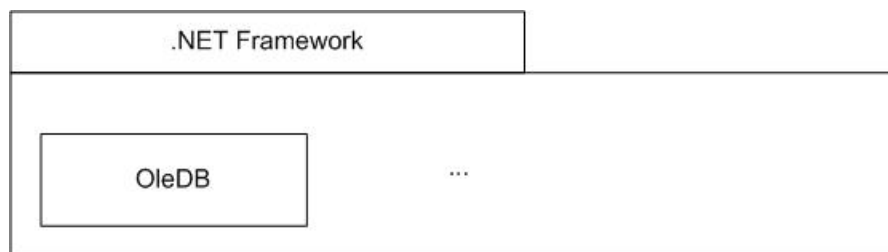
Σχήμα 61: Αποσυνθετική όψη του UI -www του εξυπηρετητή εφαρμογών



Σχήμα 62: Αποσυνθετική όψη των Web Services του εξυπηρετητή εφαρμογών



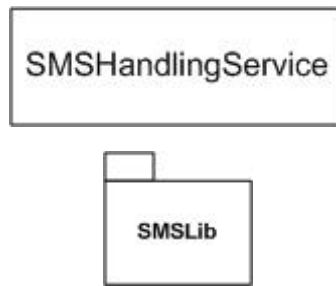
Σχήμα 63: Αποσυνθετική όψη του DM του εξυπηρετητή εφαρμογών



Σχήμα 64: Αποσυνθετική όψη των ανεξάρτητων τμημάτων του εξυπηρετητή εφαρμογών

3.3.4.1.3 SMS Handler

Ο SMS Handler αποτελείται από 2 τμήματα: την υπηρεσία που διαχειρίζεται τα μηνύματα, και το SMSLib που είναι μια βιβλιοθήκη ανοιχτού κώδικα η οποία περιλαμβάνει ρουτίνες για την επικοινωνία κινητού τηλεφώνου και υπολογιστή μέσω σειριακής ή σχεδόν σειριακής θύρας.

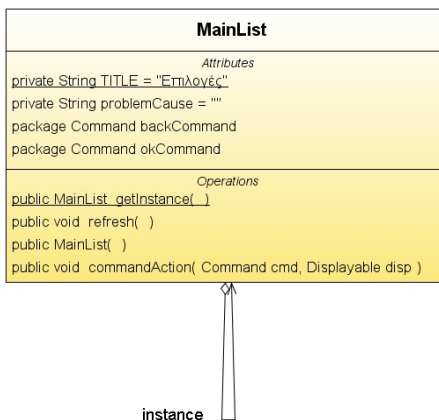


Σχήμα 65: Αποσυνθετική όψη του διαχειριστή σύντομων μηνυμάτων

3.3.4.2 Δεδομενική Όψη

3.3.4.2.1 Mobile Device / Cellphone

Στη συνέχεια βλέπουμε ενδεικτικά την αρχική λίστα επιλογών και την φόρμα επιλογής περιοχής του λογισμικού πελάτη.



Σχήμα 66: Κλάση MainList

AreaForm
<i>Attributes</i>
private String TITLE = "Επιλογή περιοχής"
package TextField AreaField
package TextField PositionField
package TextField AkField
package StringItem WarningString
package Command cancelCommand
package Command doneCommand
package Command changeVehicleCommand
<i>Operations</i>
public AreaForm(Settings settings)
public void commandAction(Command cmd, Item itm)
public void commandAction(Command cmd, Displayable disp)
public void refresh()

Σχήμα 67: Κλάση AreaForm

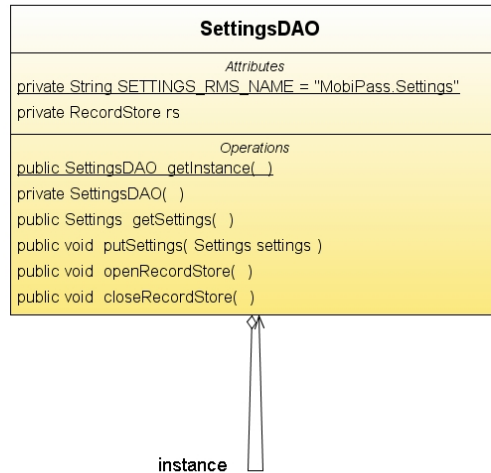
Ακολουθούν οι κλάσεις που ανήκουν στο Model και έχουν το μεγαλύτερο ενδιαφέρον. Εδώ περιγράφονται αναλυτικά κάποιες από τις οντότητες που χρησιμοποιεί το σύστημα. Το Settings και το SettingsDAO έχουν να κάνουν με την εγγραφή των ρυθμίσεων στην μνήμη του κινητού. Το RegExClass έχει να κάνει με τον έλεγχο των δεδομένων όταν αυτά εισάγονται στο κινητό.

CustDate
<i>Attributes</i>
<i>Operations</i>
public CustDate()
public String parseDate(Date date)

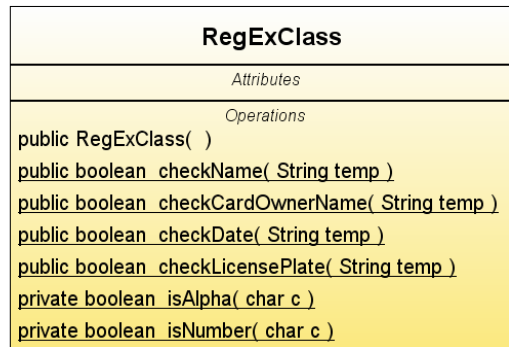
Σχήμα 68: Κλάση CustDate

Vehicle
<i>Attributes</i>
package String licensePlate
package int status
package String statusMessage
<i>Operations</i>
public Vehicle()
public Vehicle(String lp, int s, String sm)
public String getLicensePlate()
public void setLicensePlate(String lp)
public int getStatus()
public void setStatus(int s)
public String getStatusMessage()
public void setStatusMessage(String sm)

Σχήμα 69: Κλάση Vehicle



Σχήμα 70: Κλάση SettingsDAO



Σχήμα 71: Κλάση RegexClass

Settings
<i>Attributes</i> <u>private int MaxVehicles = 15</u> <u>private String ConnectionServer = "http://147.102.12.15/ws/Service1.asmx"</u> <u>private int isConnected = 1</u> private short isRegistered = 0 private short userPin private String licensePlate private String prepaidCardNo private short status private String statusMessage private String previousStatusMessage private long expirationDate
<i>Operations</i> public Settings() <u>public int getMaxVehicles()</u> public void refreshStatus() <u>public int getIsConnected()</u> public String getPreviousStatusMessage() public void setPreviousStatusMessage(String previousStatusMessage) public long getExpirationDate() public void setExpirationDate(long l) public short getStatus() public void setStatus(short status) public String getStatusMessage() public void setStatusMessage(String statusMessage) public short getIsRegistered() public void setIsRegistered(short isRegistered) public short getUserPin() public void setUserPin(short userPin) public String getLicensePlate() public void setLicensePlate(String licensePlate) public String getPrepaidCardNo() public void setPrepaidCardNo(String prepaidCardNo) <u>public Settings bytesToSettings(byte settingsBytes[0..*])</u> public byte[0..*] settingsToBytes() <u>public Settings initializeRecordStore()</u> <u>public String getConnectionServer()</u>

Σχήμα 72: Κλάση Settings

Possibility
<i>Attributes</i>
package int Area
package int Position
package int MaxDuration
package int Duration
package int Charge
package String AreaDescription
package String ChargeDescription
package String TimeDescription
<i>Operations</i>
public void setArea(int a)
public void setPosition(int p)
public void setMaxDuration(int md)
public void setDuration(int d)
public void setCharge(int c)
public void setAreaDescription(String ad)
public void setChargeDescription(String cd)
public void setTimeDescription(String td)
public Possibility()
public Possibility(int a, int p, int md, int d, int c, String ad, String cd, String td)
public StringBuffer printMe()
public String getChargeDescription()
public String getTimeDescription()
public int getArea()
public int getPosition()
public int getDuration()

Σχήμα 73: Κλάση Possibility

DAOException
<i>Attributes</i>
<i>Operations</i>
public DAOException()
public DAOException(String msg)

Σχήμα 74: Κλάση DAOException

CustFloat
<i>Attributes</i>
<i>Operations</i>
public CustFloat()
public int FloatToInt(String temp)

Σχήμα 75: Κλάση CustFloat

Ακολουθούν δυο κλάσεις που ανήκουν στο πακέτο Connection και αφορούν την σύνδεση με τον Application Server. Η ConnectionException είναι η εξαίρεση που εμφανίζεται όταν η σύνδεση αποτυγχάνει, ενώ το GettingPossibilitiesThread είναι το νήμα που ξεκινά να εκτελείται όταν ο χρήστης επιχειρεί να εκκινήσει να σταθμεύσει σε μια περιοχή και το σύστημα του απαντά με τις δυνατότητες που έχει.

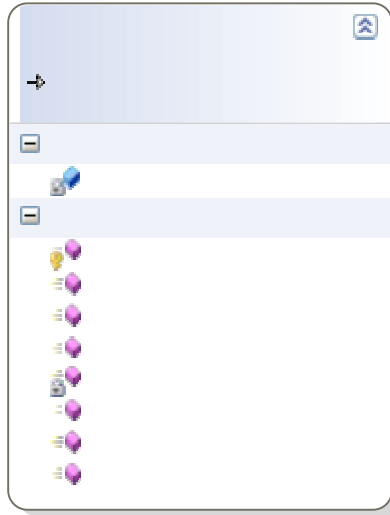
ConnectionException
<i>Attributes</i>
<i>Operations</i>
public ConnectionException()
public ConnectionException(String arg0)

GettingPossibilitiesThread
<i>Attributes</i>
private int type = 0
private String Area = new String("0")
private String Position = new String("0")
public int MaxPossibilities = 0
package int a = 1
<u>private String TAG_POSSIBILITY = "Possibility"</u>
<u>private String TAG_POSITION = "Position"</u>
<u>private String TAG_DURATION = "Duration"</u>
<u>private String TAG_MAX_DURATION = "Max Stay"</u>
<u>private String TAG_AREA_DESCRIPTION = "Area Description"</u>
<u>private String TAG_TIME_DESCRIPTION = "Time Description"</u>
<u>private String TAG_CHARGE = "Euros"</u>
<i>Operations</i>
public GettingPossibilitiesThread(String Area, String Position)
public void run()

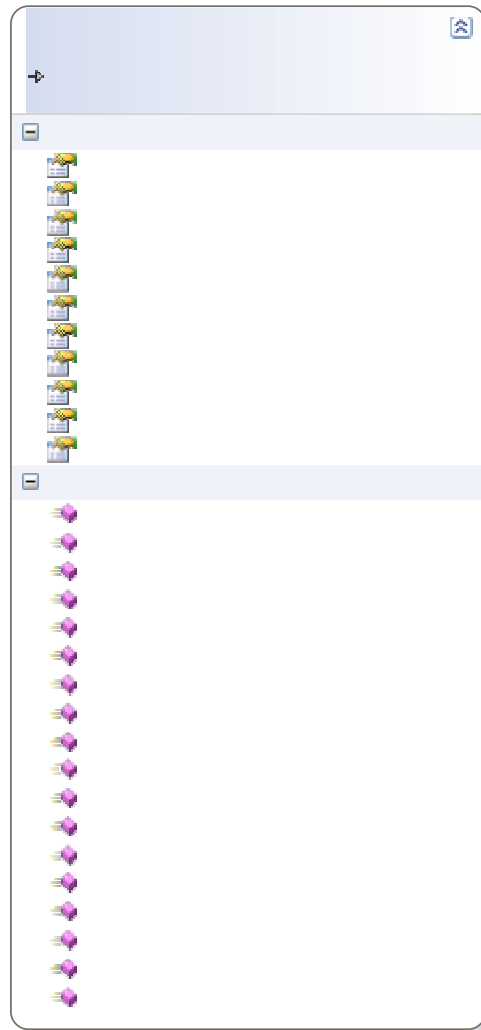
Σχήμα 76: Κλάσεις ConnectionException και GettingPossibilitiesThread

3.3.4.2.2 Application Server

Παρακάτω βλέπουμε αναλυτικά μερικές από τις κλάσεις του Application Server. Βλέπουμε τον ορισμό των δικτυακών υπηρεσιών, την επέκταση του MembershipProvider σε CustomMembershipProvider και μια ενδεικτική κλάση από το κομμάτι DM που αφορά την επεξεργασία και αποθήκευση δεδομένων.



Σχήμα 77: Κλάση Service1 - δικτυακές υπηρεσίες



Σχήμα 78: Κλάση CustomMembershipProvider

Service1

Class

WebService

Fields

component

Methods

Dispose

EndRese

GetPossi

GetVehic

Initialize

New

Register

Reserve

CustomMembership

Class

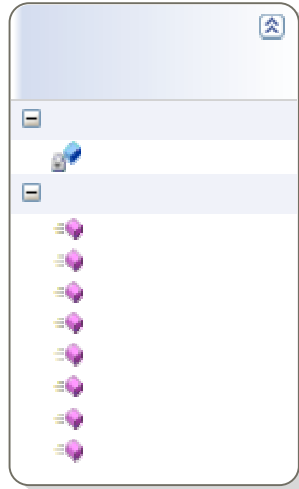
MembershipProvider

Properties

ApplicationNam

EnablePassword

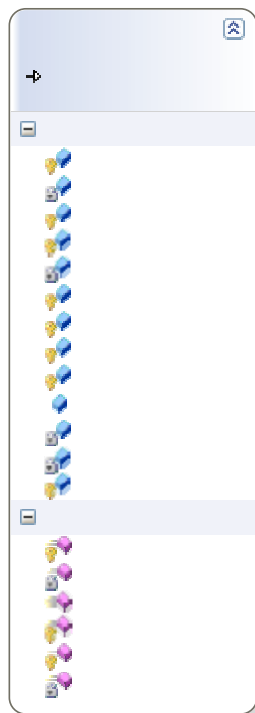
EnablePassword



Σχήμα 79: Κλάση PaymentPermit

3.3.4.2.3 SMS Handler

Όπως αναφέρθηκε και προηγουμένως, ο SMS Handler αποτελείται ουσιαστικά από μια υπηρεσία η οποία συνδέεται με το κινητό τηλέφωνο και ανά τακτά χρονικά διαστήματα ελέγχει για καινούρια μηνύματα. Στο σχήμα που ακολουθεί φαίνεται ο ορισμός της κλάσης.



Σχήμα 80: Κλάση SmsHandlingService

PaymentPermit
Class

Fields

Data

Methods

End

End

get

get

make

req

Res

Sta

3.3.4.2.4 Κλήση Δικτυακών Υπηρεσιών

Ενδεικτικά παραθέτουμε πως επιτυγχάνεται η κλήση μιας δικτυακής υπηρεσίας μέσω POST και με τη χρήση του SOAP.

HTTP POST

```
POST /ws/Service1.asmx/GetPossibilities HTTP/1.1
Host: 147.102.12.15
Content-Type: application/x-www-form-urlencoded
Content-Length: length

AreaCode=string&Position=string
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<DataSet xmlns="http://tempuri.org/mpWebService/Service1">
  <schema xmlns="http://www.w3.org/2001/XMLSchema">schema</schema>xml</DataSet>
```

SOAP 1.2

```
POST /ws/Service1.asmx HTTP/1.1
Host: 147.102.12.15
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetPossibilities xmlns="http://tempuri.org/mpWebService/Service1">
      <AreaCode>string</AreaCode>
```

```

        <Position>string</Position>
    </GetPossibilities>
</soap12:Body>
</soap12:Envelope>

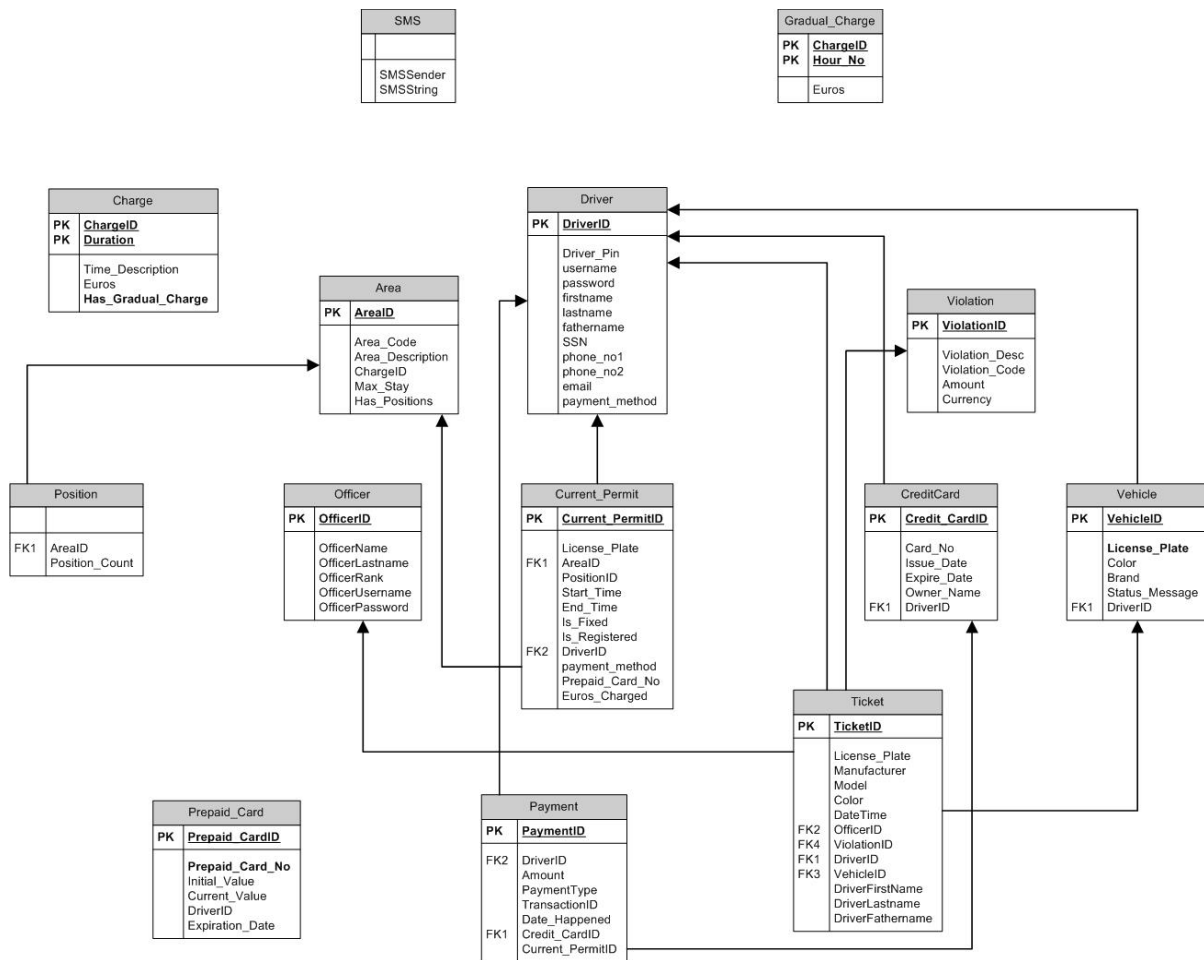
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
    <soap12:Body>
        <GetPossibilitiesResponse xmlns="http://tempuri.org/mpWebService/Service1">
            <GetPossibilitiesResult>
                <xsd:schema>schema</xsd:schema>xml</GetPossibilitiesResult>
            </GetPossibilitiesResponse>
        </soap12:Body>
    </soap12:Envelope>

```

3.3.4.2.5 Σχήμα Βάσης Δεδομένων

Τέλος, ακολουθεί το σχήμα της βάσης δεδομένων του συστήματος.



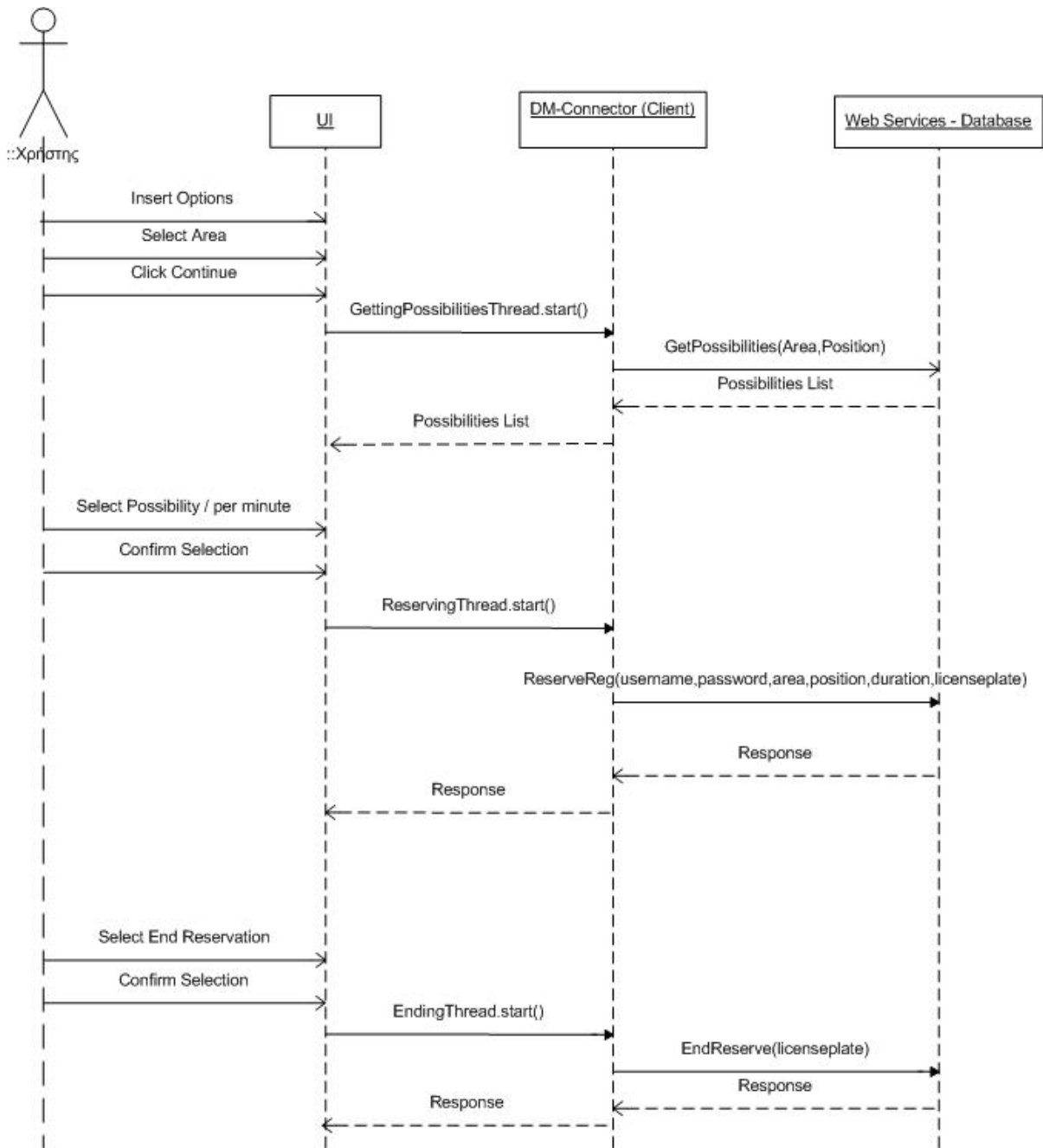
Σχήμα 81: Βάση δεδομένων

3.3.4.3 Συνεργατική Όψη

3.3.4.3.1 Ακολουθιακά Διαγράμματα

Οι εργασίες που πραγματοποιούνται από το σύστημα ελέγχου στάθμευσης είναι αυτές που περιγράφονται στις περιπτώσεις χρήσης και στις απαιτήσεις από το λογισμικό σύστημα. Στη συνέχεια, περιγράφονται αναλυτικά σε ακολουθιακά διαγράμματα οι: έναρξη και λήξη στάθμευσης για εγγεγραμμένο χρήστη, στάθμευση σταθερού χρόνου για εγγεγραμμένο χρήστη, έναρξη και λήξη στάθμευσης για μη εγγεγραμμένο χρήστη, στάθμευση σταθερού χρόνου για μη εγγεγραμμένο χρήστη, έλεγχος οχήματος, καταγραφή και εκτύπωση παράβασης, εγγραφή χρήστη, διαχείριση οχημάτων – χρεώσεων από το χρήστη, έλεγχος συστήματος.

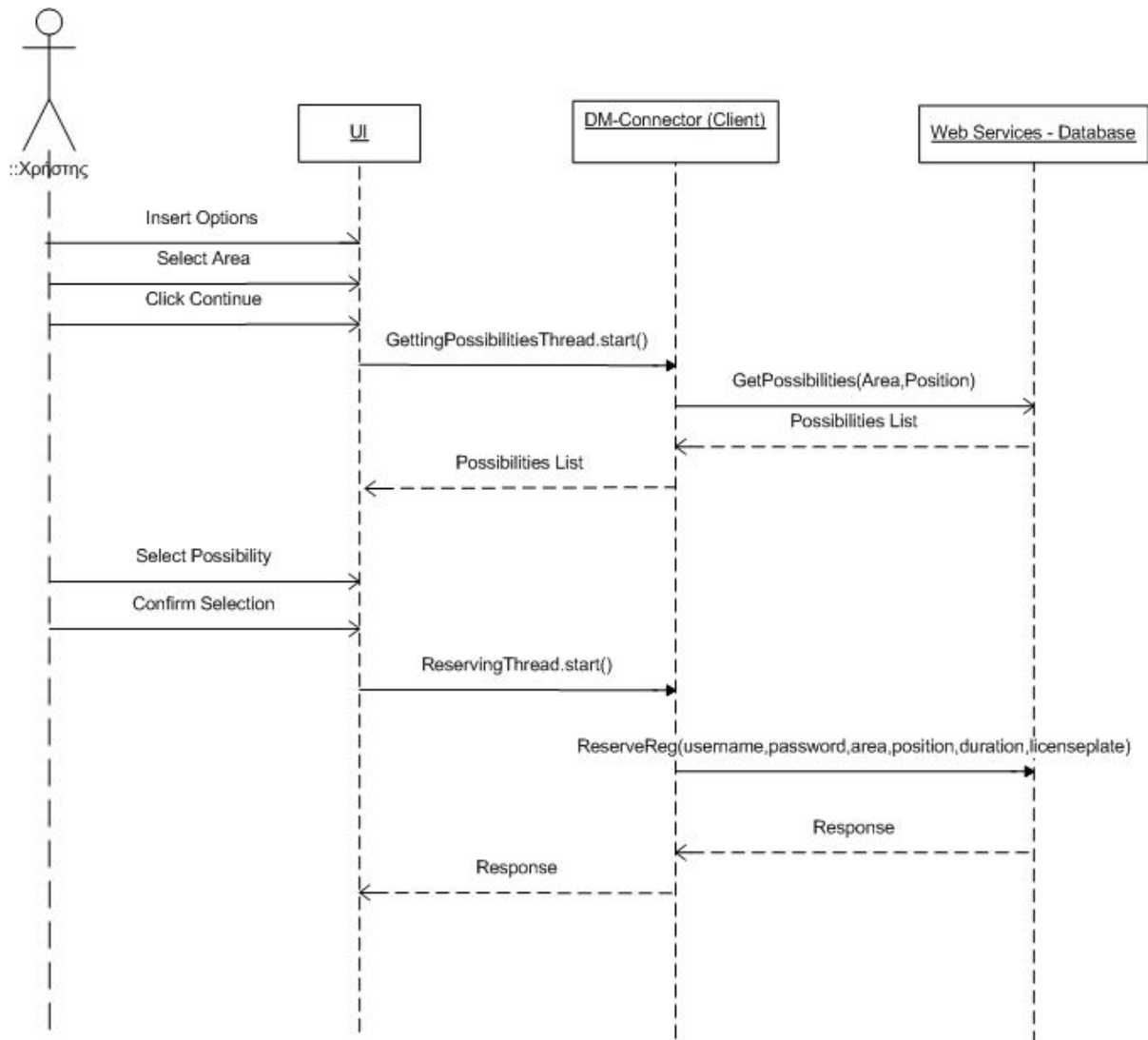
3.3.4.3.1.1 Έναρξη και λήξη στάθμευσης για εγγεγραμμένο χρήστη



Σχήμα 82: Ακολουθιακό διάγραμμα - έναρξη και λήξη στάθμευσης για εγγεγραμμένο χρήστη

Στην παραπάνω όψη είναι ορατά τρία τμήματα του συστήματος. Το γραφικό περιβάλλον χρήσης του λογισμικού πελάτη, το τμήμα του λογισμικού που φροντίζει τη σύνδεση και ο Application Server του συστήματος.

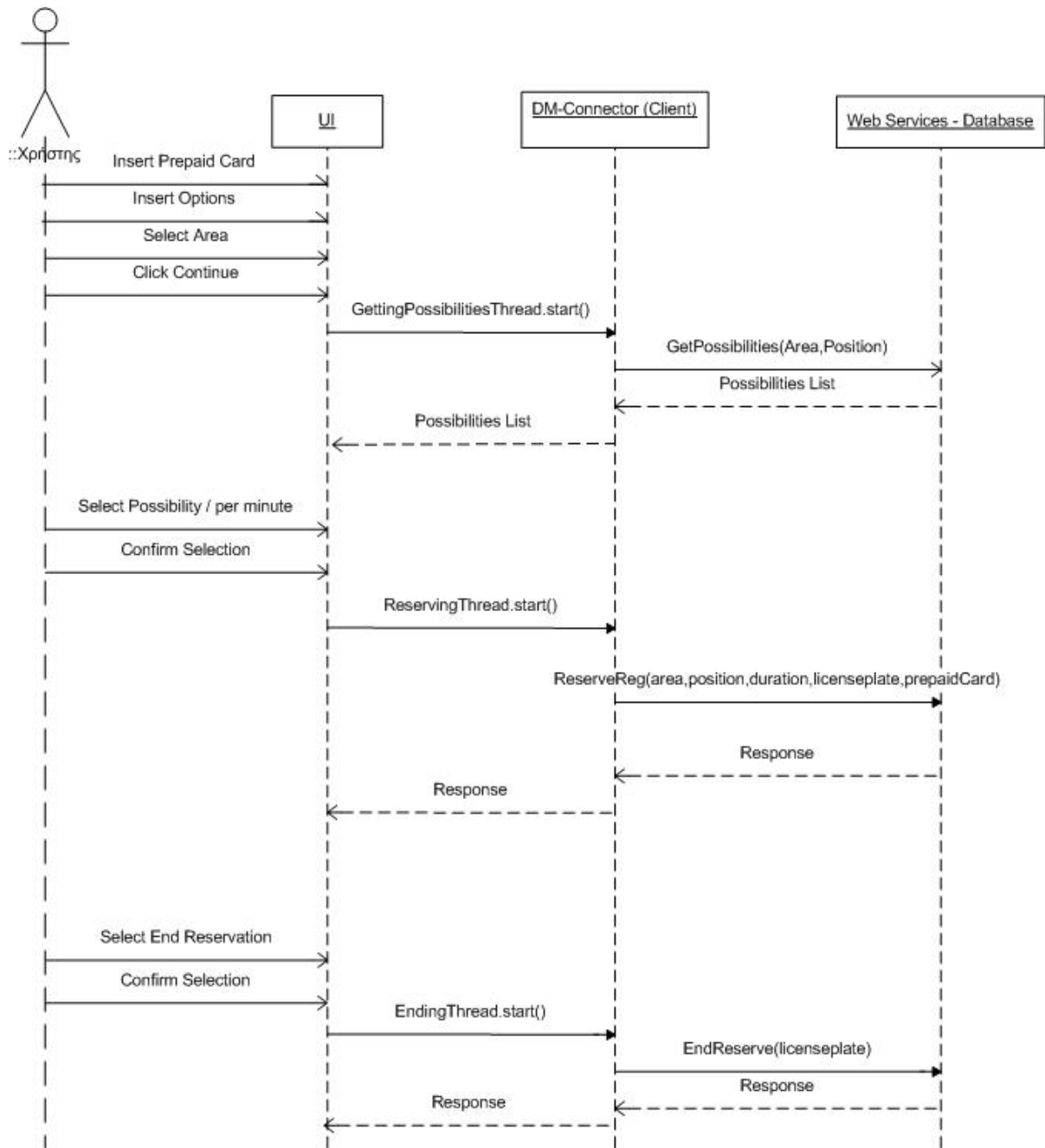
3.3.4.3.1.2 Στάθμευση σταθερού χρόνου για εγγεγραμμένο χρήστη



Σχήμα 83: Ακολουθιακό διάγραμμα -στάθμευση σταθερού χρόνου για εγγεγραμμένο χρήστη

Στο προηγούμενο σχήμα είναι σαφές ότι η διαδικασία για αγορά προπληρωμένου χρόνου είναι παρεμφερής με τη προηγούμενη περίπτωση χρήσης, αλλά δεν απαιτείται ενέργεια του χρήστη για τη λήξη της στάθμευσης.

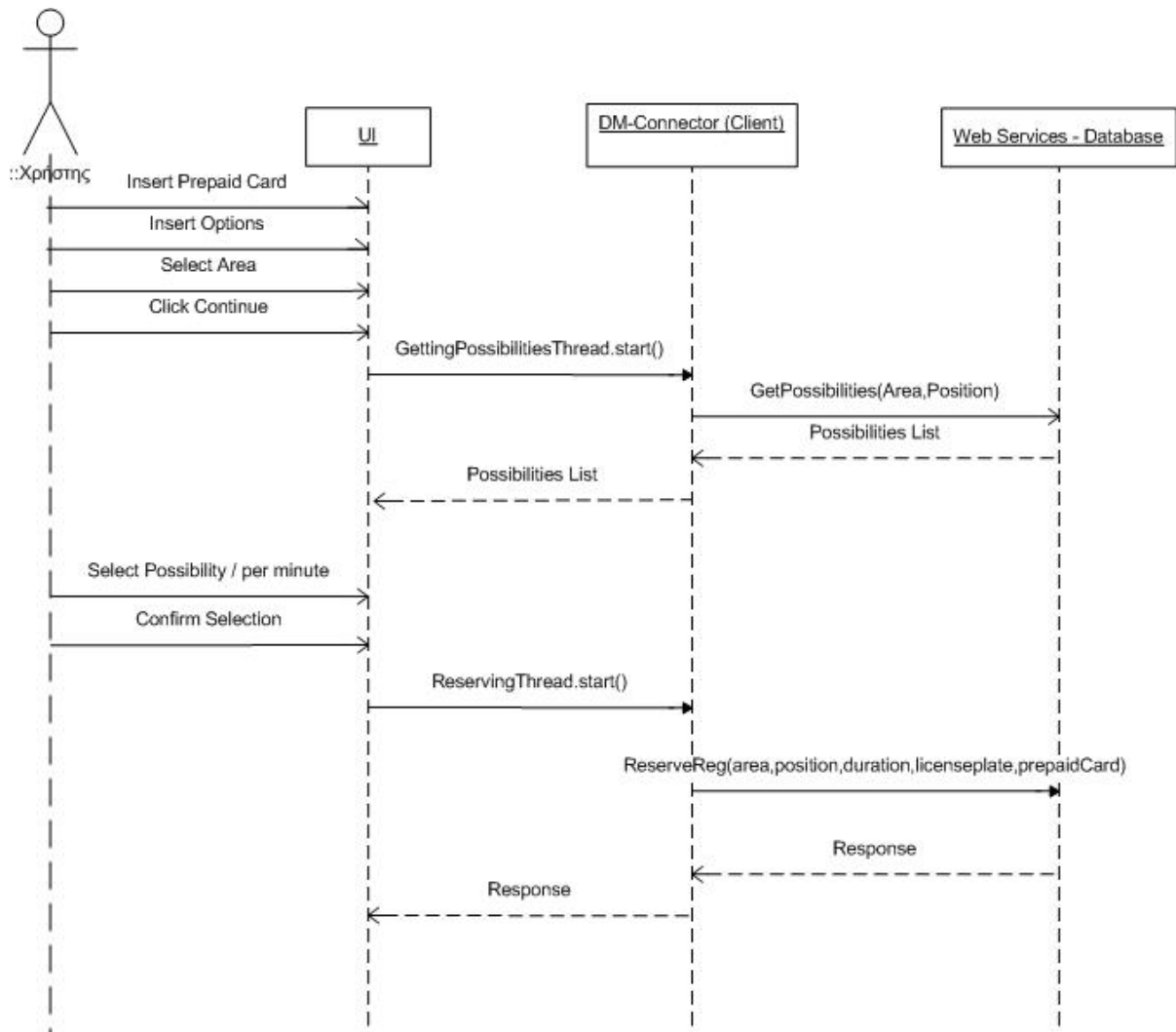
3.3.4.3.1.3 Έναρξη και λήξη στάθμευσης για μη εγγεγραμμένο χρήστη



Σχήμα 84: Ακολουθιακό διάγραμμα - έναρξη και λήξη στάθμευσης για μη εγγεγραμμένο χρήστη

Το παραπάνω σχήμα δείχνει ότι η διαδικασία που ακολουθείται είναι παρεμφερής με αυτήν που ακολουθεί και ο εγγεγραμμένος χρήστης, αν εξαιρέσουμε επιπρόσθετα στοιχεία που απαιτούνται για τη στάθμευση και τη χρέωση (αριθμός κυκλοφορίας και αριθμός προπληρωμένης κάρτας).

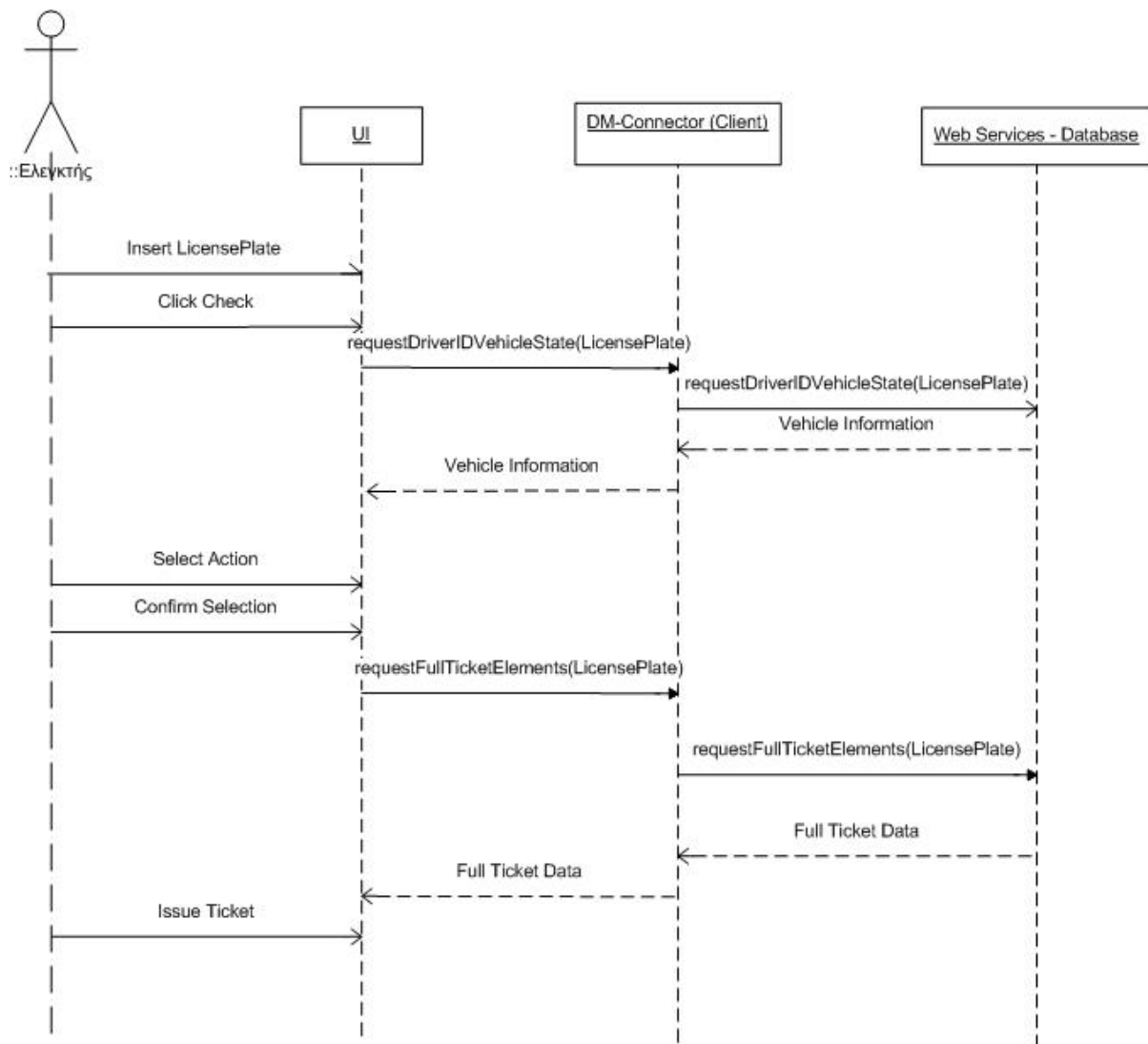
3.3.4.3.1.4 Στάθμευση σταθερού χρόνου για μη εγγεγραμμένο χρήστη



Σχήμα 85: Ακολουθιακό διάγραμμα - στάθμευση σταθερού χρόνου για μη εγγεγραμμένο χρήστη

Το παραπάνω σχήμα δείχνει ότι η διαδικασία που ακολουθείται είναι παρεμφερής με αυτήν που ακολουθεί και ο εγγεγραμμένος χρήστης, αν εξαιρέσουμε επιπρόσθετα στοιχεία που απαιτούνται για την στάθμευση και την χρέωση (αριθμός κυκλοφορίας και αριθμός προπληρωμένης κάρτας).

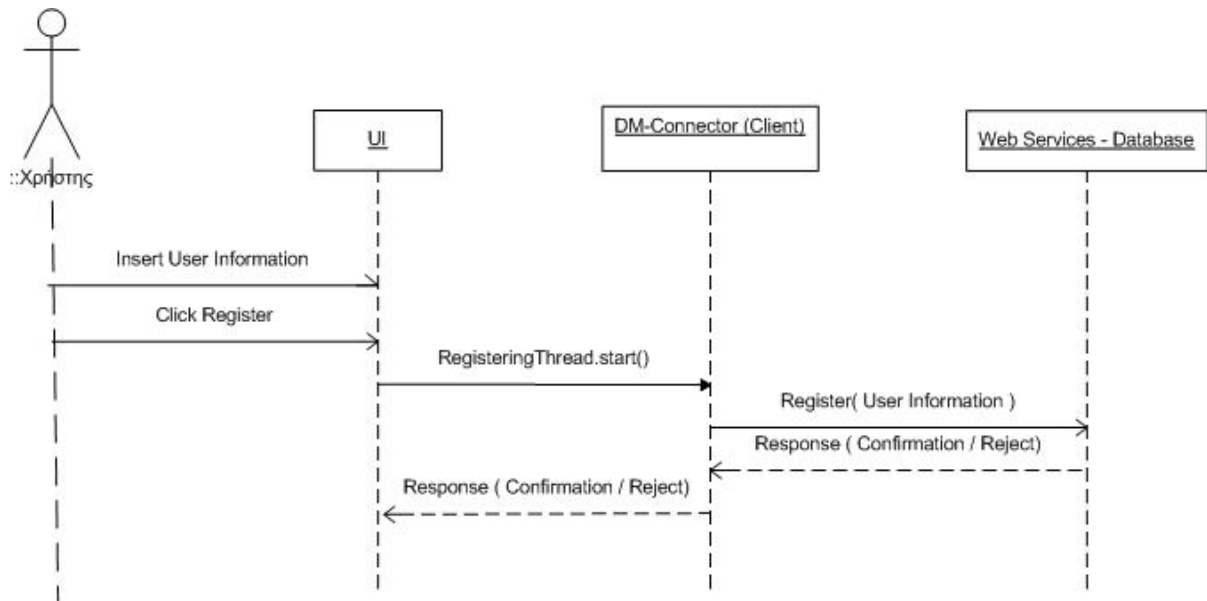
3.3.4.3.1.5 Έλεγχος οχήματος, καταγραφή και εκτύπωση παράβασης



Σχήμα 86: Ακολουθιακό διάγραμμα - έλεγχος οχήματος, καταγραφή και εκτύπωση παράβασης

Στο παραπάνω σχήμα φαίνεται η διαδικασία που ακολουθείται ώστε ο ελεγκτής να καταγράψει έναν έλεγχο και να εκδώσει μια κλήση για παράνομη στάθμευση σε ένα χρήστη.

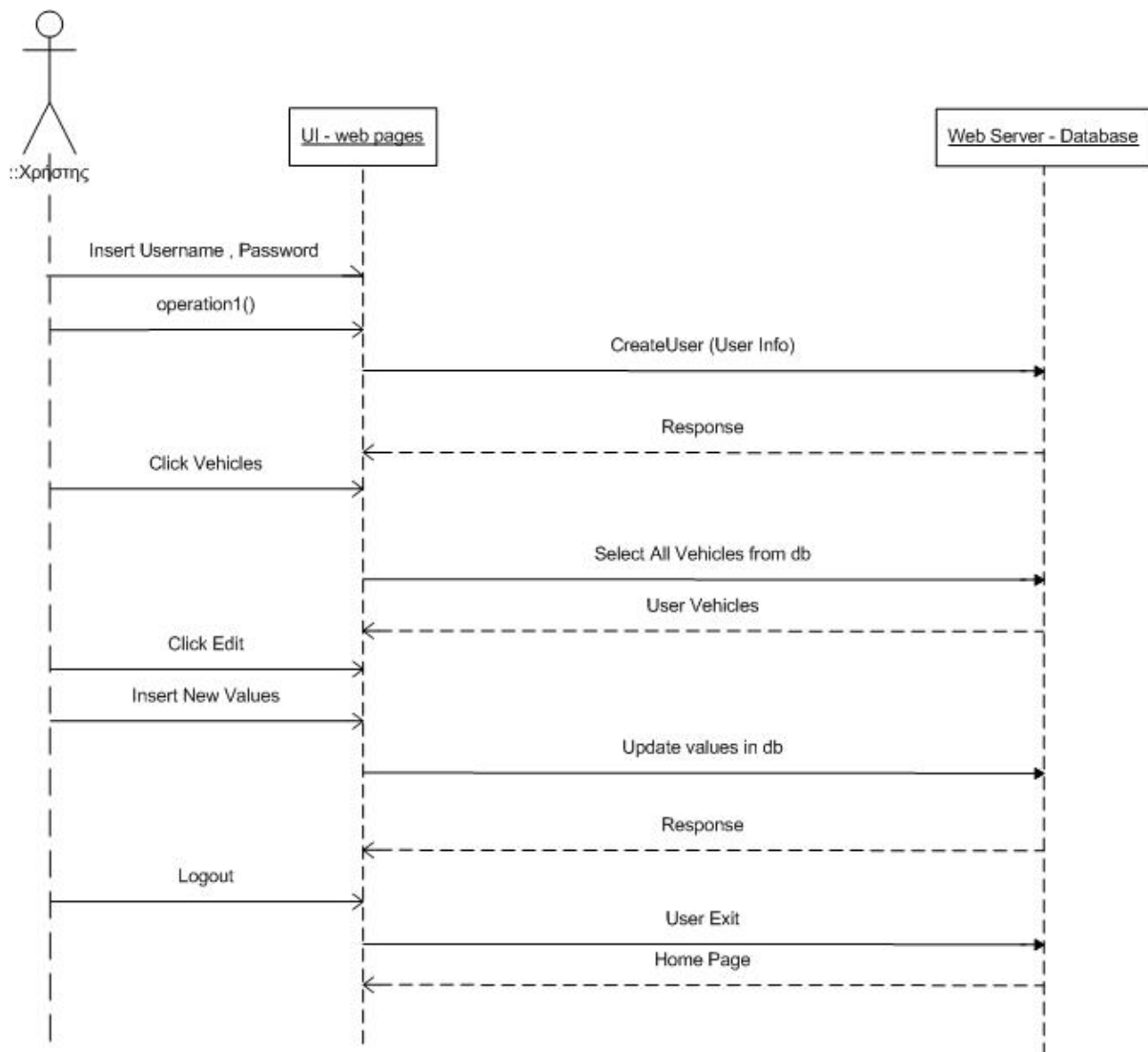
3.3.4.3.1.6 Εγγραφή χρήστη



Σχήμα 87: Ακολουθιακό διάγραμμα - εγγραφή χρήστη

Στο προηγούμενο σχήμα φαίνεται η διαδικασία που ακολουθείται ώστε ένας χρήστης να εγγραφεί στο σύστημα χρησιμοποιώντας το κινητό του τηλέφωνο.

3.3.4.3.1.7 Διαχείριση οχημάτων

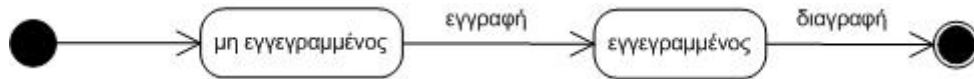


Σχήμα 88: Ακολουθιακό διάγραμμα - διαχείριση οχημάτων

Τέλος, παρατηρούμε ένα ενδεικτικό διάγραμμα της διαχείρισης των οχημάτων ή των άλλων δεδομένων του χρήστη μέσω του δικτυακού τόπου.

3.3.4.3.2 Διαγράμματα Καταστάσεων

Τα διαγράμματα καταστάσεων μας δείχνουν την αλληλουχία σταδίων από τα οποία περνάει μια οντότητα. Επίσης μας δείχνουν τα γεγονότα που μπορεί να συμβούν. Εδώ βλέπουμε διαγράμματα καταστάσεων για δυο βασικές οντότητες.



Σχήμα 89: Διάγραμμα καταστάσεων για την οντότητα Χρήστης

Η οντότητα χρήστης είναι ο οδηγός ο οποίος χρησιμοποιεί το σύστημα ελέγχου στάθμευσης για να σταθμεύσει το όχημά του.



Σχήμα 90: Διάγραμμα καταστάσεων για την οντότητα Όχημα

Η οντότητα όχημα αναφέρεται στο όχημα το οποίο είναι καταγεγραμμένο στη βάση δεδομένων του συστήματος και μπορεί να είναι ελεύθερο ή σταθμευμένο.

3.3.4.4 Διαπροσωπική Όψη

Στην όψη αυτή παρουσιάζονται οι σχεδιαστικές οντότητες μαζί με τις διαπροσωπείες τους. Οι κλάσεις αυτές περιγράφηκαν στη δεδομενική όψη.

4

Υλοποίηση

4.1 Πλατφόρμες και Προγραμματιστικά Εργαλεία

Στο παρόν κεφάλαιο θα γίνει μια αναφορά και σύντομη ανάλυση των προγραμματιστικών εργαλείων και των περιβαλλόντων που χρησιμοποιήθηκαν κατά την διαδικασία της κωδικοποίησης του λογισμικού συστήματος.

4.1.1 *Microsoft Visual Studio .NET*

Το Microsoft Visual Studio είναι ένα ενοποιημένο περιβάλλον ανάπτυξης της Microsoft. Είναι κατάλληλο για ανάπτυξη εφαρμογών, ιστοτόπων και δικτυακών υπηρεσιών και μπορεί να χρησιμοποιείται σε ηλεκτρονικούς υπολογιστές, εξυπηρετητές, φορητές συσκευές και άλλα. Δίνει τη δυνατότητα για προγραμματισμό σε μια σειρά από γλώσσες (Visual Basic, Visual C++, Visual C#, Visual J#, ASP). Η πρώτη έκδοση του περιβάλλοντος έγινε το 1997 ενώ η πλέον πρόσφατη είναι το 2005.

Το περιβάλλον αυτό περιέχει ορισμένα στοιχεία τα οποία το κάνουν εξαιρετικά ελκυστικό για την υλοποίηση του συστήματος του Application Server.

- Δίνει τη δυνατότητα για ανάπτυξη τόσο δικτυακών υπηρεσιών, όσο και δικτυακών τόπων .
- Παρέχει πρόσβαση σε μια σειρά έτοιμων βιβλιοθηκών που μπορούν να χρησιμοποιηθούν από οποιαδήποτε γλώσσα.
- Οι εφαρμογές που δημιουργούνται σε αυτό συνδέονται με ευκολία με τον MS SQL Server.
- Δίνει τη δυνατότητα για επαναχρησιμοποίηση ενός τμήματος λογισμικού.

4.1.2 *NetBeans IDE*

Το NetBeans είναι το ενοποιημένο περιβάλλον ανάπτυξης εφαρμογών Java που αναπτύσσεται από την Sun Microsystems. Ξεκίνησε από μια φοιτητική εργασία και αγοράστηκε από την Sun η οποία άνοιξε τον κώδικά του.

Ένα σημαντικό σημείο του NetBeans είναι ότι καθώς βασίζεται σε modules (τμήματα λογισμικού) ανεξάρτητες ομάδες μπορούν να προσθέσουν ή να αφαιρέσουν τμήματα από αυτό το περιβάλλον. Έτσι έχουν δημιουργηθεί αρκετές προσθήκες, μια από τις οποίες είναι το NetBeans Mobility Pack το οποίο δίνει τη δυνατότητα ανάπτυξης εφαρμογών οι οποίες βασίζεται Java Micro Edition και προορίζονται για χρήση σε κινητά τηλέφωνα ή σε άλλες φορητές συσκευές.

Η επιλογή του για την κωδικοποίηση του λογισμικού συστήματος πελάτη είναι σημαντική καθώς βοηθάει στην κωδικοποίηση και εκσφαλμάτωση του κώδικα, ενσωματώνοντας προσομοιωτή εκτέλεσης σε κινητό τηλέφωνο.

4.1.3 *Microsoft SQL Server*

Ο Microsoft SQL Server όπως έχει περιγραφεί και στο κεφάλαιο 2 αποτελεί ένα από τα σημαντικότερα συστήματα διαχείρισης βάσεων δεδομένων σήμερα. Κάποιοι από τους βασικότερους λόγους που μας οδήγησαν στην επιλογή του συγκεκριμένου συστήματος είναι η εύκολη διασύνδεση του με εφαρμογές που χρησιμοποιούν το .NET Framework και η εγγενής συμβατότητα του με το λειτουργικό σύστημα Windows Server 2003.

4.2 Λεπτομέρειες Υλοποίησης

4.2.1 Βιβλιοθήκες

Κάποιες συγκεκριμένες εργασίες που επιτελεί το σύστημα δεν υπήρχαν διαθέσιμες στα προγραμματιστικά περιβάλλοντα που αναφέρθηκαν προηγουμένως. Έτσι καταλήξαμε σε κάποιες βιβλιοθήκες ανοιχτού κώδικα.

4.2.1.1 *kXML*

Το kXML είναι ένας μικρός XML pull parser (αποκωδικοποιητής) ο οποίος έχει σχεδιαστεί για περιορισμένες εφαρμογές όπως applets και φορητές ηλεκτρονικές συσκευές. Βασίζεται στο κοινή προγραμματιστική διαπροσωπεία XML pull parser API και προσφέρει πολλά πλεονεκτήματα. Η συγκεκριμένη βιβλιοθήκη προορίζεται για χρήση στις εφαρμογές πελάτη στα κινητά τηλέφωνα. Εκεί η εφαρμογή αναλαμβάνει την ανάλυση των εγγράφων XML των οποίων αποστέλλονται από τον εξυπηρετητή εφαρμογών του συστήματος. Η επιλογή της συγκεκριμένης βιβλιοθήκης είναι εξαιρετικά σημαντική καθώς συντελεί τόσο στην ταχεία εξαγωγή των δεδομένων από το ρεύμα εισόδου όσο και στην διατήρηση του όγκου της εφαρμογής σε λογικά επίπεδα.

4.2.1.2 *SMSLib*

Το SMSLib αποτελεί μια βιβλιοθήκη συναρτήσεων και ρουτινών σε εκδόσεις τόσο για .NET Framework όσο και για Java. Η βιβλιοθήκη αυτή δίνει τη δυνατότητα για σύνδεση με κινητά τηλέφωνα χρησιμοποιώντας την σειριακή θύρα του υπολογιστή. Είναι δυνατόν να συνδεθεί με πολλά τηλέφωνα και να τα χρησιμοποιήσει σαν GSM Modems, επικοινωνώντας μαζί τους με εντολές AT. Το ελάχιστο της συμβατότητας που απαιτείται καλύπτεται από τα περισσότερα κινητά τηλέφωνα της αγοράς.

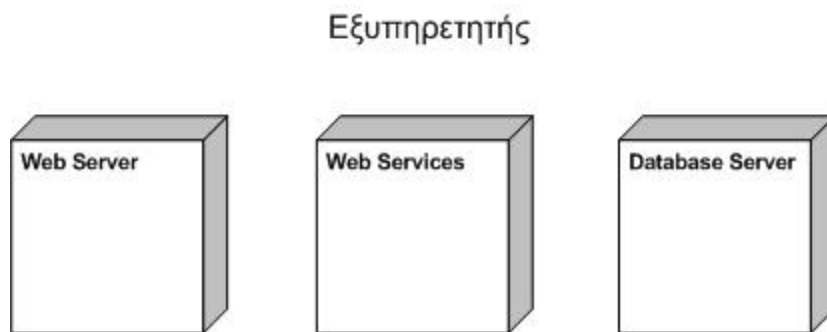
Οι ρουτίνες που προβλέπει η συγκεκριμένη βιβλιοθήκη δίνουν την δυνατότητα στον διαχειριστή μηνυμάτων SMS να διαβάζει ανά τακτά χρονικά διαστήματα τα μηνύματα που εισέρχονται σε ένα κινητό τηλέφωνο. Ακολουθώς μπορούν να εξάγονται σε συμβολοσειρές το περιεχόμενο των μηνυμάτων αλλά και ο αποστολέας, δίνοντας έτσι το αναγκαίο υλικό. Τέλος, σημαντικό είναι ότι η συγκεκριμένη βιβλιοθήκη μπορεί να συνδεθεί με το κινητό τηλέφωνο, ανεξαρτήτως του πραγματικού τρόπου σύνδεσης (USB, Bluetooth) αρκεί αυτό να αντιστοιχίζεται εικονικά σε μια σειριακή θύρα του υπολογιστή.

4.2.2 Υλοποίηση των υποσυστημάτων

4.2.2.1 Εξυπηρετητής

Το κομμάτι του εξυπηρετητή αναλύεται σε τρία ανεξάρτητα τμήματα, στον εξυπηρετητή Web, στις δικτυακές υπηρεσίες και στον εξυπηρετητή βάσεων δεδομένων. Τα στοιχεία αυτά μπορούν να αναπτυχθούν στον ίδιο ή σε διαφορετικούς κόμβους. Η δομή αυτή αναλύεται και στην αμέσως επόμενη ενότητα.

4.2.2.1.1 Αρχιτεκτονική Υλοποίησης



Σχήμα 91: Αρχιτεκτονική υλοποίησης εξυπηρετητή

Ο εξυπηρετητής Web αποτελείται από μια σειρά σελίδων ASP οι οποίες είναι σχεδιασμένες ώστε να περιλαμβάνουν την λειτουργικότητα που είναι αναγκαία για την λειτουργία του συστήματος. Ακόμη περιλαμβάνουν συγκεκριμένες κλάσεις για την επέκταση της λειτουργίας των σελίδων. Παρακάτω παρατίθεται ενδεικτικά ο κώδικας μιας σελίδας ASP καθώς και ένα τμήμα του κώδικα CustomMembershipProvider.

```
namespace CustomMembershipProviderLib
{
    public class CustomMembershipProvider :
        MembershipProvider
    {

        public MembershipUser CreateUser(string username, string
            password, string email, string firstname, string lastname, string fathersname,
                string ssnbox, string paymentmethod, string phone_no1, string
            phone_no2, string licenseplate,
```

```

        string color, string brand, int UserPIN, string CardNo, string CardOwner,
        string ExpirationDate, bool isApproved, out MembershipCreateStatus status)
    {
        OleDbConnection cnn = null;
        OleDbCommand cmd = null;
        MembershipUser newUser = null;
        try
        {
            cnn = new OleDbConnection();
            cnn.ConnectionString = "Provider=SQLOLEDB.1; Data
Source=localhost;uid=costas;pwd=123456; Initial Catalog=mobiPass;";
            cnn.Open();

            string insertQry = "Insert into Driver
(Driver_PIN,username,password,firstname,lastname,fathername,SSN,phone_no1,phone_no2
,email,payment_method) " +
                "Values ('" + UserPIN+"','" + username + "','" + password +
                "','" + firstname + "','" + lastname + "','" + fathersname + "','" + ssnbox + "','" +
                + phone_no1 + "','" + phone_no2 + "','" + email + "','" + paymentmethod+"') ";
            cmd = new OleDbCommand(insertQry, cnn);
            cmd.ExecuteNonQuery();
            cmd.CommandText = "Select DriverID from Driver WHERE username='" +
            username + "'";
            OleDbDataReader rdr = cmd.ExecuteReader();
            String DriverID = "666";

            if (rdr.Read())
            {
                DriverID = rdr.GetInt32(0).ToString();
            }
            rdr.Close();

            cmd.CommandText = " Insert into Vehicle Values ( '" +
            licenseplate + "','" + color + "','" + brand + "','" + DriverID + "')";
            cmd.ExecuteNonQuery();

            if (paymentmethod.Equals("1")) cmd.CommandText = " EXEC
ADD_NEW_CREDIT_CARD '" + username + "','" + CardNo + "','" + ExpirationDate + "','"
            + CardOwner + "'";

```

```

        else if (paymentmethod.Equals("2")) cmd.CommandText = " EXEC
CONNECT_TO_PREPAID_CARD '" + username + "', '" + CardNo + "'";

        cmd.ExecuteNonQuery();

        status = MembershipCreateStatus.Success;
    }
    catch (Exception ex)
    {
        status = MembershipCreateStatus.ProviderError;

        newUser = null;

        throw ex;
    }
    finally
    {
        cmd.Dispose();

        cnn.Close();
    }

    return newUser;

    //return null;
}

```

Πλαίσιο 1: Η κλάση CustomMembershipProvider.cs

Ο εξυπηρετητής δικτυακών υπηρεσιών δίνει την δυνατότητα στο σύστημα να προσφέρει με ευέλικτο τρόπο τις υπηρεσίες του. Παρακάτω φαίνονται οι δικτυακές υπηρεσίες που προβλέπονται και ένα κομμάτι του κώδικα της κλάσης Driver που προσφέρει αυτές τις υπηρεσίες.

Service1

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [EndReserve](#)
- [GetAccount](#)
- [GetPossibilities](#)
- [GetVehicles](#)
- [GetVehicles2](#)
- [RegisterUser](#)
- [Reserve](#)
- [ReserveReq](#)
- [requestDriverIDVehicleState](#)
- [requestFullTicketElements](#)

Σχήμα 92: Παρεχόμενες δικτυακές υπηρεσίες

```
Public Shared Function GetPossibilities(ByVal Area As String, ByVal Position As
String) As DataSet

    Dim strConnection As String = "Provider=SQLOLEDB.1; Data
Source=localhost;uid=costas;pwd=123456; Initial Catalog=mobiPass;"

    Dim conADOConnect As New OleDbConnection(strConnection)
    conADOConnect.Open()

    Dim ds As New DataSet

    Dim comADOCommand As New OleDbCommand
    comADOCommand.Connection = conADOConnect
    comADOCommand.CommandText = "SELECT
Time_Description,Area_Description,Max_Stay,Duration,Euros From Area,Charge WHERE
Area_Code='" + Area + "'" AND Area.ChargeID=Charge.ChargeID;"

    Dim adosc As New OleDbDataAdapter
    adosc.SelectCommand = comADOCommand
    adosc.Fill(ds, "Possibility")

    comADOCommand.CommandText = "SELECT * FROM Gradual_Charge,Area WHERE
Gradual_Charge.ChargeID=Area.ChargeID AND Area.Area_Code='" + Area + "'"

    adosc.SelectCommand = comADOCommand
    adosc.Fill(ds, "Gradual")
```

```

conADOConnect.Close()

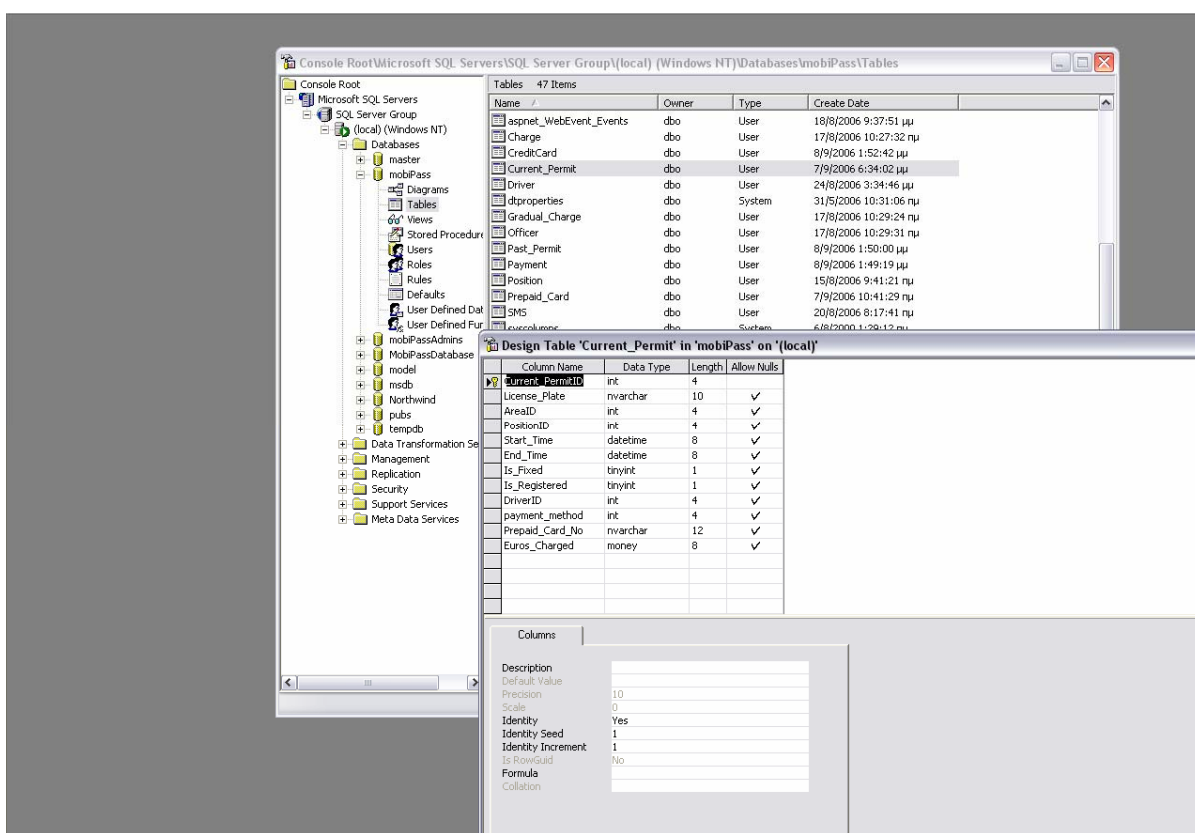
Return ds

End Function

```

Πλαίσιο 2: Η διαδικτυακή υπηρεσία GetPossibilities

Τέλος, η βάση δεδομένων αποτελεί τον πυρήνα του συστήματος όπου αποθηκεύονται όλες οι πληροφορίες των οδηγών και των οχημάτων κατά την διάρκεια του χρόνου. Είναι σημαντικό αυτό το σχήμα να είναι σχεδιασμένο σωστά ώστε να μην προκαλούνται ασυνέπειες κατά τη χρήση.



Σχήμα 93: Σύστημα διαχείρισης βάσης δεδομένων

4.2.2.1.2 Λεπτομέρειες Υλοποίησης

Οι τρεις λειτουργίες εξυπηρέτησης του πελάτη τελικά αναπτύχθηκαν και υλοποιήθηκαν στον ίδιο κόμβο. Οι επιλογές των τμημάτων λογισμικού αλλά και της πλατφόρμας ανάπτυξης έχουν σκοπό να οδηγήσουν στη μέγιστη δυνατή συμβατότητα. Επίσης, είναι σημαντικό να αναφερθεί πως η απαίτηση για εκτέλεση της ίδιας λειτουργίας μέσω του δικτυακού περιβάλλοντος, αλλά και μέσω δικτυακών υπηρεσιών απαιτούσε την αλλαγή ορισμένων από τις προκαθορισμένες κλάσεις της ASP .NET.

4.2.2.2 Εφαρμογή Σύντομων Μηνυμάτων

4.2.2.2.1 Αρχιτεκτονική Υλοποίησης

Η αρχιτεκτονική υλοποίησης της εφαρμογής σύντομων μηνυμάτων είναι σχετικά απλή. Λειτουργεί σαν ένας parser που δέχεται μηνύματα, τα αποκωδικοποιεί και πραγματοποιεί τις αντίστοιχες αιτήσεις μέσω των δικτυακών υπηρεσιών. Παρακάτω φαίνεται ένα τμήμα του κώδικα της κλάσης της υπηρεσίας.

```
Public Class SmsHandlingService
    Inherits System.ServiceProcess.ServiceBase
    Public WithEvents srv As CService

    Protected Overrides Sub OnStart(ByVal args() As String)
        Timer1 = New System.Timers.Timer()
        Timer1.Interval = 120000
        logFile = "SMSLog.txt"
        sw = File.CreateText("c:\smslog.txt")
        Position = ""

        Try
            sw.WriteLine("Before doing anything")
            sw.Flush()

            Dim ds As New DataSet

            Dim msgList As LinkedList(Of CIncomingMessage) = New LinkedList(Of CIncomingMessage)

            If (args.Length > 0) Then
                srv = New CService(args(0), 19200, "", "")
            Else : srv = New CService("COM4", 19200, "", "")
            End If

            sw.WriteLine("ReadMessages(): Synchronous Reading.")
            sw.WriteLine(" Using " & CService._name & " " & CService._version)
            sw.WriteLine()
            sw.Flush()
            srv.Connect()

            ' Lets get info about the GSM device...
```

```

sw.WriteLine("Mobile Device Information:")

sw.WriteLine(" Manufacturer   : {0}", srv.DeviceInfo.Manufacturer)
sw.WriteLine(" Model          : {0}", srv.DeviceInfo.Model)
sw.WriteLine(" SerialNo       : {0}", srv.DeviceInfo.SerialNo)
sw.WriteLine(" IMSI           : {0}", srv.DeviceInfo.Imsi)
sw.WriteLine(" S/W Version    : {0}", srv.DeviceInfo.SwVersion)
sw.WriteLine(" Battery Level : {0}%", srv.DeviceInfo.BatteryLevel)
sw.WriteLine(" Signal Level  : {0}%", srv.DeviceInfo.SignalLevel)
sw.WriteLine()

sw.Flush()

'...

Catch ex As Exception
    sw.WriteLine(ex.ToString())
    sw.Flush()
    sw.Close()
End Try
End Sub

...

```

Πλαίσιο 3: Τμήμα κώδικα υποσυστήματος διαχείρισης μηνυμάτων

4.2.2.2.2 Λεπτομέρειες Υλοποίησης

Ένα δύσκολο κομμάτι της υλοποίησης της διαχείρισης μηνυμάτων ήταν η επικοινωνία με το κινητό τηλέφωνο που λειτουργεί ως δέκτης μηνυμάτων, μέσω της σειριακής θύρας. Η επικοινωνία αυτή λόγω της έλλειψης καθιερωμένων προτύπων αποδείχθηκε δύσκολη και με διαφορετικά επίπεδα δυσκολίας ανάλογα με τη χρησιμοποιούμενη συσκευή.

4.2.2.3 Πελάτης

Με τον όρο πελάτης αναφερόμαστε σε δυο διακριτά συστήματα λογισμικού τα οποία εντούτοις βασίζονται σε κοινές αρχές και διέπονται εν γένει από τους ίδιους κανόνες. Είναι το κομμάτι λογισμικού πελάτη το οποίο θα χρησιμοποιεί ο χρήστης – οδηγός και το κομμάτι που θα χρησιμοποιεί ο ελεγκτής του συστήματος.

4.2.2.3.1 Αρχιτεκτονική Υλοποίησης

Το κομμάτι της εφαρμογής πελάτη για τον οδηγό έχει αναπτυχθεί σύμφωνα με την αρχιτεκτονική που φαίνεται στο παρακάτω σχήμα. Παρακάτω ακολουθεί ένα κομμάτι του κώδικα της κλάσης AreaForm.java και ένα κομμάτι του Vehicle.java.

```
/*
 * AreaForm.java
 * Created on 4 Αύγουστος 2006, 11:24 πμ
 */

package midlet.view;

public class AreaForm extends Form implements CommandListener ,
ItemCommandListener{

    private static final String TITLE = "Περιοχή";

    //στοιχεία Φόρμας
    TextField AreaField;
    TextField PositionField;
    TextField AkField;
    StringItem WarningString;
    Command cancelCommand;
    Command doneCommand;
    Command changeVehicleCommand;
    private Settings settings;

    //κατασκευαστής
    public AreaForm(Settings settings) {
        super(TITLE);
        this.settings=settings;
        AreaField = new TextField("Περιοχή", "", 32, TextField.NUMERIC);
        PositionField= new TextField("Θέση","", 32, TextField.NUMERIC);

        cancelCommand = new Command("Cancel", Command.BACK, 1);
```

```

        doneCommand = new Command("Done", Command.OK, 2);
        append(AreaField);
        append(PositionField);

        if(!settings.getLicensePlate().equals("")){
            AkField = new TextField("Αρ. Κυκλοφορίας",settings.getLicensePlate(), 8,
TextField.UNEDITABLE);
            append(AkField);    }
        else
            {arningString= new StringItem("Δεν έχετε\nΕπιλέξει όχημα!\nΕπιλέξτε
εδώ.",null);
                append(WarningString);
            }

        if ( settings.getIsRegistered()==1) {
            StringItem changeVehicle= new StringItem(" ", "Αλλαγή
οχήματος", Item.BUTTON);
            changeVehicleCommand = new Command("Αλλαγή οχήματος", Command.ITEM, 3);
            changeVehicle.addCommand(changeVehicleCommand);
            changeVehicle.setItemCommandListener(this);
            append(changeVehicle);
        }

        addCommand(doneCommand);
        addCommand(cancelCommand);
        setCommandListener(this);
    }
}

```

Πλαίσιο 4: Η κλάση AreaForm.java

```

/*
 * Vehicle.java
 *
 * Created on 3 Αύγουστος 2006, 11:32 πμ
 */

package midlet.model;

```

```

public class Vehicle {

    String licensePlate;

    int status;

    String statusMessage;

    public Vehicle(){
        licensePlate="";
        status=-1;
        statusMessage="";
    }

    public Vehicle(String lp,int s,String sm) {
        licensePlate=lp;
        status=s;
        statusMessage=sm;
    }

    //...
}

```

Πλαίσιο 5: Η κλάση Vehicle.java

Ακολουθεί η εφαρμογή του ελεγκτή η οποία ακολουθεί μια ελαφρώς απλοποιημένη μορφή της παραπάνω αρχιτεκτονικής. Τέλος, παρατίθεται ένα τμήμα του κώδικα της κλάσης EntryForm.java

4.2.2.3.2 Λεπτομέρειες Υλοποίησης

Η υλοποίηση του λογισμικού πελάτη, τόσο αυτού που προορίζεται για τον τελικό χρήστη, όσο και αυτού που προορίζεται για τον ελεγκτή, αποτέλεσε ένα δύσκολο εγχείρημα καθώς απαιτούσε όχι μόνο ταχύτητα αλλά και συμβατότητα με όσο το δυνατόν περισσότερες συσκευές. Κατά τη σύνδεση του κινητού με τις δικτυακές υπηρεσίες του συστήματος επιλέχθηκε η χρήση POST και όχι η χρήση SOAP, ώστε να μεταδίδονται όσο το δυνατόν λιγότερες πληροφορίες και να αποφευχθεί η άσκοπη καθυστέρηση που προκαλείται κατά την σύνταξη ενός εγγράφου XML.

Ακόμη, πρέπει να τονιστεί πως επιλέχθηκε η δημιουργία καινούριων νημάτων κατά τη διαδικασία σύνδεσης με τον εξυπηρετητή του συστήματος. Αυτή η διαδικασία αυξάνει σημαντικά την ασφάλεια

και την σταθερότητα της εφαρμογής, ανεξάρτητα από την ποιότητα της σύνδεσης και επιπλέον δεν δίνει την αίσθηση του «παγωμένου» προγράμματος στον τελικό χρήστη.

5

Έλεγχος

5.1 Οδηγός Εγκατάστασης

5.1.1 Εγκατάσταση του εξυπηρετητή βάσης δεδομένων

Ο εξυπηρετητής θεωρούμε ότι διαθέτει λειτουργικό σύστημα Microsoft Windows Server 2003 και αρκετή υπολογιστική ισχύ ώστε να ανταποκριθεί στις απαιτήσεις του συστήματος. Στον εξυπηρετητή εγκαθιστούμε το σύστημα διαχείρισης βάσεων δεδομένων Microsoft SQL Server 2000. Το σχήμα της βάσης πρέπει να ονομάζεται MobiPass και ο διαχειριστής να έχει τα ίδια στοιχεία με αυτά που χρησιμοποιούνται στις άλλες εφαρμογές, δηλαδή στις εφαρμογές δικτυακών υπηρεσιών και των δικτυακών τόπων.

Συγκεκριμένα, το σχήμα της βάσης που απαιτείται για την ορθή λειτουργία όλων των εφαρμογών που συναποτελούν το σύστημα παρουσιάζεται στην παράγραφο 3.3.4.2.5. Επίσης, για την ορθή εκκίνηση του συστήματος πρέπει να περιέχονται κάποια αρχικά δεδομένα που θα περιγράφουν ορισμένες περιοχές και τις χρεώσεις τους προς τον χρήστη.

Στο δίσκο που συνοδεύει την διπλωματική εργασία περιλαμβάνεται το αρχείο MobiPass.schema που αποτελεί ένα στιγμιότυπο της βάσης δεδομένων κατά τη διαδικασία του ελέγχου.

5.1.2 Εγκατάσταση του τμήματος των δικτυακών υπηρεσιών και των δικτυακών τόπων

Ο εξυπηρετητής που θα παρέχει τις άνω υπηρεσίες θεωρούμε ότι αν δεν ταυτίζεται με τον εξυπηρετητή βάσης δεδομένων, τότε έχει άμεση δικτυακή σύνδεση με αυτόν - άρα και πρόσβαση στο σύστημα διαχείρισης της βάσης. επίσης, ο εν λόγω εξυπηρετητής διαθέτει εγκατεστημένο το .NET Framework καθώς και έχει τη δυνατότητα εξυπηρέτησης εφαρμογών με χρήση του IIS (Internet Information Services), στοιχεία τα οποία παρέχονται από το λειτουργικό σύστημα Windows Server και από τους δικτυακούς τόπους της Microsoft.

Στο δίσκο που συνοδεύει την διπλωματική εργασία περιλαμβάνεται το αρχείο WSsetup.msi το οποίο εγκαθιστά τις δικτυακές υπηρεσίες που απαιτούνται για την λειτουργία του συστήματος και τα αρχεία SESsetup.msi και SESADMINsetup.msi, τα οποία μπορεί να εγκαταστήσουν τον δικτυακό τόπο των χρηστών και των διαχειριστών του συστήματος αντίστοιχα.

5.1.3 Εγκατάσταση του διαχειριστή σύντομων μηνυμάτων

Ο διαχειριστής σύντομων μηνυμάτων πρέπει και αυτός να είναι εγκατεστημένος σε έναν υπολογιστικό κόμβο που είναι διασυνδεδεμένος με το διαχειριστή δικτυακών υπηρεσιών. Η λειτουργία του γίνεται ως υπηρεσία (service) του λειτουργικού συστήματος των Windows.

Η εγκατάσταση της υπηρεσίας πρέπει να γίνει ως εξής. Ενώ σε ένα σημείο του δίσκου θα βρίσκεται η υπηρεσία σε μορφή “SmsHandlingService.exe”, μέσω της γραμμής εντολών πρέπει να γραφούν οι εντολές “INSTALLUTIL SmsHandlingService.exe”. Η εκκίνηση της υπηρεσίας γίνεται όπως περιγράφεται παρακάτω. Με την ακόλουθη σειρά πρέπει να επιλεγούν ο πίνακας ελέγχου, οι υπηρεσίες (services) και ακολούθως με δικαιώματα διαχειριστή να επιλεγεί και να εκκινηθεί η υπηρεσία SMSHandlingService.

Στο δίσκο που συνοδεύει την διπλωματική εργασία περιλαμβάνεται το αρχείο SESADMINsetup.msi το οποίο μπορεί να εγκαταστήσει σε οποιοδήποτε υπολογιστή με .NET Framework την υπηρεσία που διαχειρίζεται τα σύντομα μηνύματα.

5.1.4 Εγκατάσταση της εφαρμογής πελάτη (χρήστη/ελεγκτή)

Οι εφαρμογές πελάτη αφού μεταφραστούν και συμπιεστούν, αποτελούνται από δύο αρχεία η κάθε μια. Και τα δυο αρχεία έχουν το όνομα της εφαρμογής (π.χ. MobiPassClient, MobiPassInspector), ενώ το ένα έχει κατάληξη .jad και το άλλο κατάληξη .jar. Η διαδικασία εγκατάστασης είναι σχετικά εύκολη στις σύγχρονες φορητές ψηφιακές συσκευές και στα κινητά τηλέφωνα αν και διαφοροποιείται ελαφρά ανάλογα με τη συσκευή.

Στην συσκευή που χρησιμοποιήθηκε στην συγκεκριμένη υλοποίηση (Sony Ericsson K700i), η σύνδεση έγινε με τη χρήση ενός μεσολαβητή δικτύου Bluetooth. Με απλή μετακίνηση του jar αρχείου στο εικονίδιο του τηλεφώνου και ανάλογες απαντήσεις στα μηνύματα που εμφανίζονται, γίνεται η εγκατάσταση της εφαρμογής στο κινητό τηλέφωνο.

Στο δίσκο που συνοδεύει την διπλωματική εργασία περιέχονται τα αρχεία jad και jar των εφαρμογών που προορίζονται για τον χρήστη και για τον ελεγκτή του συστήματος.

5.2 Οδηγός Χρήσης

5.2.1 Χρήση της ιστοσελίδας του συστήματος

Η ιστοσελίδα του συστήματος περιλαμβάνει τρία διακριτά μέρη, που απευθύνονται σε τρεις διαφορετικές κατηγορίες επισκεπτών. Στον ενδιαφερόμενο χρήστη του συστήματος, στον εγγεγραμμένο χρήστη και στον διαχειριστή.

Ο ενδιαφερόμενος μπορεί να ενημερωθεί για τον τρόπο λειτουργίας του συστήματος και να κατεβάσει στο κινητό του το λογισμικό χρήσης. Στον χρήστη δίνεται η δυνατότητα να ελέγξει το λογαριασμό του , τα οχήματα, τις χρεώσεις και τις κλήσεις του. Τέλος, ο διαχειριστής μέσω της ιστοσελίδας μπορεί να ενημερωθεί για τα στατιστικά χρήσης και να επιλέξει το σενάριο χρήσης του συστήματος.

5.2.2 Στάθμευση μέσω MIDlet

5.2.2.1 Εγγεγραμμένος χρήστης

Από τον εγγεγραμμένο χρήστη το σύστημα απαιτεί ελάχιστα στοιχεία για να προχωρήσει στη στάθμευση.

1. Ο χρήστης εισάγει την περιοχή στην οποία σταθμεύει το όχημα του (αν προβλέπεται από το σενάριο χρήσης μπορεί να εισάγει και συγκεκριμένη θέση στάθμευσης)
2. Ο χρήστης επιλέγει «Συνέχεια»



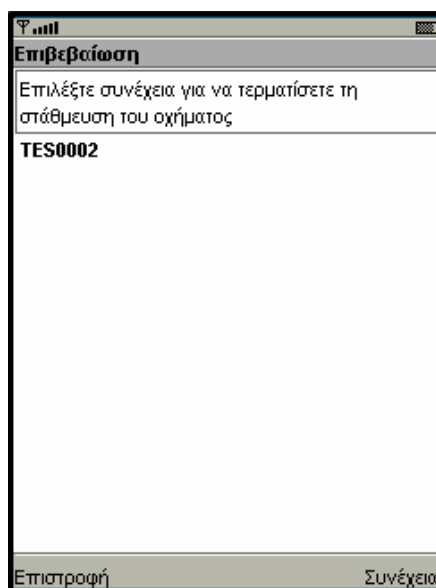
3. Μετά τη σύνδεση με τον server του συστήματος εμφανίζονται οι δυνατές επιλογές χρέωσης για την περιοχή που είχε επιλεγεί προηγουμένως.
4. Ο χρήστης διαλέγει με ποιον τρόπο θα γίνει η χρέωσή του (π.χ. αγορά σταθερού χρόνου, ανά λεπτό)
5. Επιλέγει «συνέχεια»



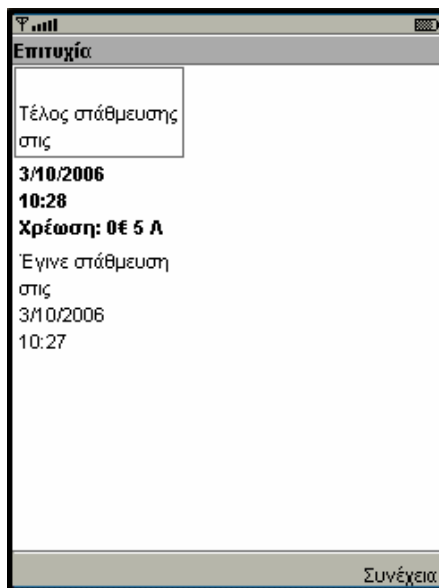
6. Μετά τη σύνδεση με το server εμφανίζεται ένα μήνυμα που ενημερώνει το χρήστη για την επιτυχία ή όχι της στάθμευσης.
7. Ειδικά στην χρέωση ανά λεπτό, όταν ο χρήστης αποχωρεί από τη θέση στάθμευσης πρέπει να ενημερώσει το σύστημα. Έτσι επιλέγει «Τερματισμός» στο αρχικό μενού επιλογών. Εμφανίζεται μια οθόνη επιβεβαίωσης και επιλέγει «Συνέχεια».



8. Εμφανίζεται μια οθόνη επιβεβαίωσης και επιλέγει «Συνέχεια».



9. Μετά τη σύνδεση με το server εμφανίζεται ένα μήνυμα που ενημερώνει το χρήστη για την επιτυχία ή όχι του τερματισμού της στάθμευσης και για το ποσό χρέωσης



5.2.2.2 Μη εγγεγραμμένος χρήστης

Από τον εγγεγραμμένο χρήστη το σύστημα απαιτεί περαιτέρω διευκρινήσεις, κι έτσι προστίθεται μια επιπλέον φόρμα.

Πριν το βήμα 1 της προηγούμενης ενότητας ο μη εγγεγραμμένος χρήστης πρέπει να εισάγει τον αριθμό κυκλοφορίας του οχήματός του καθώς και τον αριθμό της προπληρωμένης κάρτας που θα χρεωθεί για την στάθμευση του.

Στη συνέχεια ακολουθεί όμοια διαδικασία με τον εγγεγραμμένο χρήστη, όπως στην § 5.2.2.1 .

5.2.3 Στάθμευση μέσω σύντομων μηνυμάτων

5.2.3.1 Εγγεγραμμένος χρήστης

Ο εγγεγραμμένος χρήστης μπορεί να αγοράσει χρόνο στάθμευσης στέλνοντας ένα σύντομο μήνυμα (SMS) από το κινητό του τηλέφωνο.

Το μήνυμα θα περιλαμβάνει το γράμμα “T” ακολουθούμενο από το όνομα χρήστη και τον κωδικό χρήστη. Στη συνέχεια πρέπει να προσθέσει τον αριθμό κυκλοφορίας του οχήματος, τον τετραψήφιο κωδικό της περιοχής και την διάρκεια της στάθμευσης σε ώρες.

π.χ.

T username password IBM3958 0003 2

Ο εγγεγραμμένος χρήστης επίσης μπορεί να χρησιμοποιήσει την υπηρεσία έναρξης λήξης στάθμευσης στέλνοντας το μήνυμα

E username password IBM3958 0003

Με επανάληψη της αποστολής του μηνύματος το σύστημα ενημερώνεται για τη λήξη της χρέωσης

5.2.3.2 Μη εγγεγραμμένος χρήστης

Ο μη εγγεγραμμένος χρήστης με την παρούσα υλοποίηση μπορεί να αγοράσει χρόνο στάθμευσης στέλνοντας “T” ακολουθούμενο από τον αριθμό της προπληρωμένης κάρτας, τον αριθμό κυκλοφορίας, τον τετραψήφιο κωδικό της περιοχής και την διάρκεια της στάθμευσης σε ώρες.

π.χ.

T 10294839 IBM3958 0003 2

Ο εγγεγραμμένος χρήστης επίσης μπορεί να χρησιμοποιήσει την υπηρεσία έναρξης λήξης στάθμευσης όπως και ο εγγεγραμμένος στέλνοντας το μήνυμα

E 10294839 IBM3958 0003

6

Επίλογος

Στην πρώτη ενότητα του κεφαλαίου συνοψίζεται το θέμα της διπλωματικής εργασίας και παρουσιάζονται τα συμπεράσματα που προέκυψαν κατά την διαδικασία ανάλυσης, σχεδίασης και υλοποίησης του συστήματος. Στην δεύτερη ενότητα γίνεται αναφορά σε προτεινόμενους τρόπους επέκτασης της εφαρμογής.

6.1 Σύνοψη και συμπεράσματα

6.1.1 Σύνοψη

Η διαδικασία ανάλυσης, σχεδίασης και ανάπτυξης οδήγησε στην υλοποίηση ενός συστήματος ελέγχου στάθμευσης που χρησιμοποιεί διαδικτυακές υπηρεσίες. Πολλές φορές για τη λύση ενός προβλήματος είναι αναγκαία η κατασκευή ενός συστήματος που να προβλέπει ποικίλες διαδικασίες μεταξύ ατόμων και οργανισμών αλλά και να υποστηρίζει ηλεκτρονικές συσκευές που να επικοινωνούν με διαφορετικούς τρόπους μεταξύ τους..

Κατά την προσέγγιση του προβλήματος επιλέχθηκε η υπηρεσιοστραφής (service-oriented) αρχιτεκτονική. Για την παροχή των υπηρεσιών χρησιμοποιήθηκαν οι δικτυακές υπηρεσίες (web services), όπως προβλέπονται από τα πρότυπα διεθνών οργανισμών. *Προέκυψαν λοιπόν τα παρακάτω υποσυστήματα:*

- Εξυπηρετητής Βάσης Δεδομένων. Το υποσύστημα αυτό αναλαμβάνει την διαχείριση της βάσης δεδομένων του συστήματος. Επιλέχθηκε ο MS SQL Server 2000 σε λειτουργικό σύστημα Windows Server 2003.

- Εξυπηρετητής δικτυακών υπηρεσιών και ιστοτόπων. Ο συγκεκριμένος εξυπηρετητής επιβαρύνεται με το χρέος να προσφέρει τις δικτυακές υπηρεσίες αλλά και να παρέχει τις σελίδες Web στο διαδίκτυο. Επιλέχθηκε η χρήση του IIS και του .NET Framework σε λειτουργικό σύστημα Windows Server 2003.
- Διαχειριστής Σύντομων Μηνυμάτων. Λαμβάνει, επεξεργάζεται και αποστέλλει σύντομα μηνύματα από και προς τα κινητά τηλέφωνα των χρηστών. Υλοποιήθηκε ως υπηρεσία του λειτουργικού συστήματος Windows .
- Εφαρμογή πελάτη για το χρήστη και τον διαχειριστή. Οι εφαρμογές αυτές εκμεταλλεύονται τους μειωμένους πόρους των φορητών συσκευών του χρήστη και του διαχειριστή ώστε να πραγματοποιήσουν τις αναγκαίες διαδικασίες. Επιλέχθηκε η χρήση του κυριάρχου περιβάλλοντος Java 2 Micro Edition.

Τελικά το σύστημα περιλαμβάνει τα παρακάτω βασικά χαρακτηριστικά (βλ. και § 1.1)

- Αγορά χρόνου στάθμευσης μέσω απλών κινητών τηλεφώνων
- Ολοκλήρωση των ελέγχων από την Δημοτική Αστυνομία.
- Χαμηλό κόστος κτήσης και λειτουργίας.
- Ευκολία στη χρήση και την εκμάθηση από όλους τους χρήστες του.
- Το σύστημα είναι σε θέση να καλύπτει την ανάγκη των αστικών κέντρων της χώρας για ένα ολοκληρωμένο, φτηνό και άρτιο τεχνολογικά, σύστημα ελεγχόμενης στάθμευσης.

6.1.2 Συμπεράσματα

Οι δικτυακές υπηρεσίες (web services) αποτελούν ένα χρήσιμο βοήθημα στις περιπτώσεις πολύπλοκων προβλημάτων που περιλαμβάνουν οργανισμούς αλλά και απλούς χρήστες – όπως το παρόν σύστημα ελέγχου στάθμευσης. Η λογική που διέπει το σύστημα συντέλεσε στη δυνατότητα επίλυσης προβλημάτων που αλλιώς θα ήταν δυσεπίλυτα.

Συγκεκριμένα, ο τρόπος κατασκευής του συστήματος οδήγησε σε μια σειρά πλεονεκτημάτων, όπως η ανεξαρτησία από τις πλατφόρμες υλοποίησης, η ευελιξία, η επεκτασιμότητα και η εύκολη αποδόμηση του συστήματος σε επιμέρους υποσυστήματα. Ισχυρό εργαλείο στην προαναφερθείσα κατεύθυνση αποτελεί η γλώσσα XML, η οποία δίνει τη δυνατότητα για περιγραφή της πληροφορίας με τέτοιο τρόπο που να συμβαδίζει με τα ανωτέρω στοιχεία του συστήματος.

Η μεθοδολογία που ακολουθεί η παρούσα διπλωματική εργασία μπορεί να λειτουργήσει ως οδηγός για την επίλυση παρόμοιων προβλημάτων. Η αντιμετώπιση των πολύπλοκων διαδικασιών επικοινωνίας μεταξύ χρηστών και συσκευών ως υπηρεσίες και όχι ως μόνιμες συνδέσεις, οδηγεί στην καλύτερη αντιμετώπιση των δυσκολιών που εμφανίζονται.

6.2 Μελλοντικές επεκτάσεις

Οι πιθανές επεκτάσεις του συστήματος μπορούν να κατηγοριοποιηθούν σε εκείνες που προβλέπουν επαναυλοποίηση ενός εκ των υποσυστημάτων με στόχο την υποστήριξη διαφορετικών προτύπων και λειτουργιών και σε εκείνες που περιλαμβάνουν την επέκταση της λειτουργικότητας του συστήματος.

Στην πρώτη κατηγορία ανήκουν οι παρακάτω πιθανές επεκτάσεις:

- Η διαχείριση των σύντομων μηνυμάτων που λαμβάνονται και αποστέλλονται θα μπορούσε να υλοποιηθεί σε έναν εξυπηρετητή που όχι μόνο θα διαθέτει αρκετούς πόρους για να ικανοποιήσει ικανό αριθμό χρηστών, αλλά ακόμη θα είχε έναν άλλον τρόπο αποστολής και λήψης μηνυμάτων χωρίς την μεσολάβηση της σειριακής θύρας του υπολογιστή. Η βελτίωση της συνδεσμολογίας με το κινητό τηλέφωνο ή το GSM modem είναι αναγκαία ώστε να αποφευχθούν καθυστερήσεις και προβλήματα όταν πολλοί χρήστες στέλνουν μηνύματα.
- Η επέκταση της υλοποίησης του ιστοτόπου του συστήματος είναι μια εφικτή πρόταση, η οποία θα οδηγούσε σε βελτιωμένη λειτουργικότητα απέναντι στους χρήστες.
- Στην υλοποίηση των δικτυακών υπηρεσιών μπορεί να προστεθεί το πρότυπο SSL το οποίο θα προσφέρει ασφάλεια κατά τη σύνδεση μεταξύ των φορητών ψηφιακών συσκευών και του κεντρικού εξυπηρετητή.

Στην δεύτερη κατηγορία περιλαμβάνονται οι ακόλουθες προτάσεις:

- Η δυνατότητα προσθήκης άλλων δυνατοτήτων επικοινωνίας και δέσμευσης χρόνου στάθμευσης από τον τελικό χρήστη. Ένα τέτοιο παράδειγμα μπορεί να είναι η τεχνολογία IVR – (σύστημα αυτόματης φωνητικής απόκρισης) που μπορεί σχετικά εύκολα να προστεθεί στο παρόν σύστημα.

- Η υλοποίηση διαφορετικών σεναρίων χρήσης που θα περιλαμβάνουν τη χρέωση θέσεων, τη χρέωση δικύκλων, τη στάθμευση σε οργανωμένους χώρους στάθμευσης, αλλά και οποιαδήποτε παρόμοια διαδικασία ελεγχόμενης στάθμευσης
- Η προσθήκη της τεχνολογίας GPS και τεχνολογιών GIS στις εφαρμογές των χρηστών αλλά και των ελεγκτών θα προσέφερε ενδεχομένως επιπλέον πληροφορίες αλλά και λειτουργίες στο σύστημα.
- Η προσθήκη φωτογραφιών στην εφαρμογή ελέγχου στάθμευσης του ελεγκτή που θα δίνει τη δυνατότητα απόδειξης της διάπραξης της παράβασης.

7

Βιβλιογραφία

- [New02] Eric NewComer. Understanding Web Services Xml, Wsdl, Soap And Uddi, 2002

- [Bal03] Francesco Balena. Programming Microsoft Visual Basic .NET, 2003
- [CJ02] David Chappell, Tyler Jewell. Java Web Services, 2002.
- [Dut03] G.Andrew Duthie, Microsoft ASP.NET Programming with Microsoft Visual Basic .NET, 2003
- [HL03] Dan Hurwitz, Jesse Liberty. Programming .NET Windows Applications, 2003
- [IBM] Service Oriented Architecture, www.ibm.com
- [Jod04] Martin de Jode . Programming Java 2 Micro Edition on Symbian OS, A developer's guide to MIDP 2.0, 2004

- [Kni00] Brian Knight.SQL Server 2000 for Experienced DBAs, 2000

- [LHD01] Simon St. Laurent, Joe Johnston, Edd Dumbill. Programming Web Services with XML-RPC,2001

- [RBJ02] Ray Rankins, Paul Bertucci, Paul Jensen .Microsoft SQL Server 2000 Unleashed, 2002
- [REI] O'Reilly, www.xml.com
- [RTP+03] Roger Riggs, Antero Taivalsaari, Jim Van Peurse, Jyri Huopaniemi, Mark Patel, Aleksi Uotila, Jim Holliday. Programming Wireless Devices with the Java™ 2 Platform, Micro Edition, Second Edition,2003

- [SKS97] Abraham Silberschatz, Henry F. Korth, S. Sudarshan. Database Systems Concepts 1997
- [SUN] Sun Java, <http://java.sun.com>
- [W3C] W3C –XML Technologies, <http://www.w3.org>
- [WH02] James White, David Hemphill. Java 2 Micro Edition, Java in Small Things, 2002