



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας.
Όνοματεπώνυμο Φοιτητή	Νικόλαος Παπανικολάου
Πατρώνυμο	Κωνσταντίνος
Αριθμός Μητρώου	ΜΠΠΛ/ 09030
Επιβλέπων	Δουληγέρης Χρήστος, Καθηγητής

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Πίνακας περιεχομένων

Περίληψη.....	6
Abstract.....	6
Εισαγωγή.....	7
Κεφάλαιο 1.....	9
1.1 Εισαγωγή.....	9
1.2 Το αντικείμενο της διατριβής.....	9
1.3 Σχετικές τεχνολογίες.....	10
1.4 Διαδικασία ανάπτυξης.....	12
1.5 Προκλήσεις κατά την εργασία.....	12
1.6 Δομή της διατριβής.....	13
Κεφάλαιο 2.....	15
Εισαγωγή.....	15
2.1 Αναλυτική περιγραφή του πλαισίου.....	15
2.1.1 Παρόμοιες προσπάθειες ανάπτυξης.....	15
2.1.2 Cross-site Scripting.....	16
2.1.3 Minification.....	18
2.1.4 Χρήση «έξυπνων συσκευών» και παραδοσιακών περιηγητών (smartphones vs traditional browsers).....	18
2.1.5 Εγγενείς, Διαδικτυακές και Υβριδικές εφαρμογές.....	19
2.2 Αναλυτική περιγραφή των τεχνολογιών που χρησιμοποιήθηκαν.....	22
2.2.1 Phoneygap.....	22
2.2.2 JavaScript libraries.....	23
Phoneygap.js.....	24
jQuery.js.....	24
jQuery.blockUI.js.....	25
Mimic.js.....	25
Application-js.js.....	27
List.js.....	27
2.2.3 HTML5.....	27
2.2.4 Πρωτόκολλο XML-RPC.....	28
2.2.5 Spitogatos.gr webservice.....	29
2.3 Αναλυτική περιγραφή όλων των τεχνολογιών που μελετήθηκαν αλλά δε χρησιμοποιήθηκαν.....	30
jQueryUI.js.....	30
jQueryMobile.js.....	30
JSON-XML-RPC.....	31
JS-XMLRPC (XML-RPC for JavaScript).....	31
Jsolait.....	32
Κεφάλαιο 3.....	33
3.1 Περιγραφή της εφαρμογής.....	33

3.2 Περιγραφή της αρχιτεκτονικής της εφαρμογής.....	33
3.3 Περιγραφή της λειτουργικότητας της εφαρμογής	37
3.3.1 Διάγραμμα σειράς (Sequence diagram)	37
3.3.2 Διάγραμμα δραστηριοτήτων (Activity diagram)	39
3.3.3 Διάγραμμα τάξεων (Class diagram).....	40
3.3.4 Διάγραμμα συνεργασίας (Collaboration diagram)	41
3.3.5 Διάγραμμα κατάστασης (Statechart diagram)	42
3.3.6 Διάγραμμα αντικειμένων (Object diagram)	43
3.3.7 Διάγραμμα εξαρτημάτων (Component diagram)	44
3.3.8 Διάγραμμα διανομής (Deployment diagram)	45
3.3.9 Διάγραμμα περιπτώσεων χρήσης (Use Case diagram).....	46
3.4 Περιορισμοί – Restrictions.....	47
3.4.1 Προσομοιωτής εικονικής συσκευής – Android Virtual Device Manager	47
3.4.2 Τεχνολογικοί περιορισμοί – Cross-Site scripting	48
3.4.3 Συγκεκριμένες απαιτήσεις της διαδικτυακής πύλης.....	48
3.5 Παραδοχές κατά την ανάπτυξη της εφαρμογής	49
3.5.1 Υποστηριζόμενες τεχνολογίες.....	49
3.5.2 Ρυθμίσεις και περιεχόμενο της εφαρμογής.....	50
Κεφάλαιο 4	51
Συμπεράσματα	51
4.1 Ανακεφαλαίωση.....	51
4.2 Περιορισμοί στην υλοποίηση.....	51
4.3 Μελλοντικές επεκτάσεις.....	53
4.4 Διαθέσιμες λειτουργίες.....	53
4.5 Διεπαφή με το χρήστη	54
4.6 Ενσωματωμένες τεχνολογίες.....	54
4.7 Συμπεράσματα για τις τεχνολογίες που χρησιμοποιήθηκαν	55
4.8 Συμπεράσματα για το Phoneygap.....	55
4.9 Συμπεράσματα για το ρόλο των smartphones	56
4.10 Γενικά συμπεράσματα	57
Βιβλιογραφία	58
Παράρτημα 1 – Εγχειρίδια.....	61
1.1 Εγχειρίδιο χρήστη	61
1.1.1 Εισαγωγή	61
1.1.2 Γνωριμία με το περιβάλλον	61
1.1.3 Καταχώρηση αγγελίας	67
1.1.4 Καταχώρηση φωτογραφίας.....	68
1.1.5 Κουμπιά της εφαρμογής	69
1.1.6 Μηνύματα λάθους	71
1.2 Εγχειρίδιο προγραμματιστή.....	73

Μεταπτυχιακή Διατριβή	Παπανικολάου Νικόλαος
1.2.1 Σύνολο αρχείων	73
1.2.2 Χρήση των αρχείων	74
1.2.3 Βιβλιοθήκη application-js.js.....	74
Παράρτημα II – Ο κώδικας της εφαρμογής.....	76
Index.html	76
Application-js.js.....	81
List.js.....	90
Style.css	125

Περίληψη

Η παρούσα εργασία, πραγματεύεται την ιδέα της υλοποίησης μιας εφαρμογής για «έξυπνα» κινητά τηλέφωνα (smartphones). Στόχος της εργασίας, ήταν να δημιουργηθεί μια εφαρμογή η οποία θα δίνει τη δυνατότητα στους χρήστες να κάνουν καταχώρηση επιθυμητών στοιχείων και πληροφοριών σε πραγματικό χρόνο σε ένα ιστότοπο. Ωστόσο, η εφαρμογή, δεν αναπτύχθηκε στην εγγενή γλώσσα ενός συγκεκριμένου λειτουργικού συστήματος για το οποίο προορίζεται η εφαρμογή. Αντίθετα, γίνεται μέσω της πλατφόρμας Phonegap, η οποία επιτρέπει την ανάπτυξη κώδικα με χρήση HTML, JavaScript και CSS προκειμένου να μπορεί να δημιουργηθεί εγγενής εφαρμογή για τα περισσότερα λειτουργικά συστήματα που διατίθενται στις «έξυπνες» συσκευές.

Η καταχώρηση της πληροφορίας στη βάση δεδομένων της επιχείρησης γίνεται μέσω της χρήσης μιας διαδικτυακής υπηρεσίας (web service) την οποία διαθέτει η επιχείρηση (web service consumption) ενώ η μεταφορά των δεδομένων και η επικοινωνία με τη διαδικτυακή υπηρεσία γίνεται με χρήση του πρωτοκόλλου XML-RPC.

Για τις ανάγκες επίτευξης της υλοποίησης σε όσο το δυνατόν πιο ρεαλιστικά δεδομένα, η επιχείρηση Spitogatos.gr προσέφερε όλες τις απαραίτητες τεχνικές λεπτομέρειες της δικής της διαδικτυακής υπηρεσίας, προκειμένου να γίνεται καταχώρηση των δεδομένων στη βάση δεδομένων της επιχείρησης. Λόγω αυτής της συνεργασίας, και με δεδομένο ότι η επιχείρηση δραστηριοποιείται στις κτηματομεσιτικές υπηρεσίες, η εφαρμογή διαμορφώθηκε ως μία εφαρμογή παροχής υπηρεσιών καταχώρησης αγγελιών σε μεσίτες. Ωστόσο αυτή η χρήση είναι ενδεικτική και μπορεί επεκταθεί σε πολλά διαφορετικά πεδία, ανάλογα με τις απαιτήσεις του εκάστοτε εγχειρήματος.

Abstract

This thesis, is about developing a native application for smartphones. The aim of this paper, is to build an application through which, users will be able to submit via a web service information in real time, on a website. However, the application, is not developed in a native programming language that matches the operating system in which it is aimed to be installed. The application is developed by using a platform called Phonegap, which allows developing application using source code that is written using HTML, JavaScript and CSS, in order for the programmer to be able to compile the project in different operating systems that are used on smart phones.

The data submission in the database of the company is made, through the consumption of a web service that is already installed in the company's side, while the data transfer, and the communication with the web service takes place through the use of XML-RPC protocol in the online database of a business by consuming the webservices that the business provides.

In order to make this application as realistically specified and operated as possible, there was established a cooperation with a company called Spitogatos.gr which offered, all necessary technical information, of its own web service, which was allowed to be used in order to perform test of the application and submit the necessary information and data. Because of this type of cooperation, and due to the fact that the company specializes in real estate, the application was developed in order to serve the needs of someone that would need to submit real estate information. However, this implementation is only a demonstration, which means that there is a variety of applicable fields that this project could be used.

Εισαγωγή

Με την εξέλιξη της τεχνολογίας και την ευρύτερη χρήση των προσωπικών ηλεκτρονικών υπολογιστών, οι οποίοι αποτελούν πλέον σημαντικό κομμάτι της ζωής πολλών εκατομμυρίων ανθρώπων, άρχισε να γίνεται επιτακτικότερη η ανάγκη στους καταναλωτές να μπορούν να έχουν μαζί τους, τους ηλεκτρονικούς υπολογιστές και κατά συνέπεια να έχουν πρόσβαση στις απαιτούμενες πληροφορίες.

Σταδιακά και συμπληρωματικά προς τον επιτραπέζιο ηλεκτρονικό υπολογιστή (desktop) άρχισαν να εμφανίζονται και διάφορες μορφές φορητών ηλεκτρονικών υπολογιστών και έξυπνων συσκευών. Αρχικά εμφανίστηκαν οι φορητοί ηλεκτρονικοί υπολογιστές (notebooks/laptops) οι οποίοι έδιναν τη δυνατότητα στο χρήστη να μπορεί να εκτελέσει τις εργασίες του, από διάφορα σημεία, μεταφέροντας μαζί τον ηλεκτρονικό υπολογιστή.

Στο χώρο των κινητών τηλεφώνων ήδη από τις αρχές της δεκαετίας του 2000, είχε αρχίσει να προσφέρεται πρόσβαση στο διαδίκτυο μέσω WAP (Wireless Application Protocol) (World Wide Web Consortium (W3C), 1999), το οποίο ωστόσο προσέφερε κάποιες στοιχειώδεις πληροφορίες και δεν ακολουθούσε τη χρηστικότητα των ηλεκτρονικών υπολογιστών.

Ωστόσο, σε συνδυασμό με την αρκετά ευκολότερη και ευρύτερη δυνατότητα πρόσβασης στο διαδίκτυο που προσφέρεται στις μέρες μας, άρχισαν να προτείνονται λύσεις για υπολογιστές και έξυπνες συσκευές ακόμα μικρότερες και πιο βολικές στη μεταφορά. Στον τομέα των ηλεκτρονικών υπολογιστών εμφανίστηκαν τα netbooks, ενώ τα κινητά τηλέφωνα άρχισαν να ενσωματώνουν επιπλέον λειτουργίες και να μετατρέπονται σε έξυπνες συσκευές ή έξυπνα τηλέφωνα (smart phones).

Τον τελευταίο καιρό, στη θέση των netbooks, εμφανίστηκε μια νέα κατηγορία έξυπνων συσκευών, οι υπολογιστές τύπου «ταμπλέτας» (tablet pcs) οι οποίοι αποτελούν μια ενδιάμεση κατάσταση των έξυπνων κινητών τηλεφώνων και των ηλεκτρονικών υπολογιστών, καθώς τα περισσότερα φέρουν λειτουργικά συστήματα όμοια με αυτά των έξυπνων κινητών τηλεφώνων (IOS, Android, κλπ) ενώ παράλληλα λόγω της μεγαλύτερης οθόνης (σε σχέση με τα κινητά τηλέφωνα) προσφέρουν χρηστικότητα αντίστοιχη των notebooks.

Παράλληλα με την εξέλιξη από πλευράς υλικού (hardware) ακολούθησε και η εξέλιξη από πλευράς λογισμικού (software) προκειμένου να μπορέσει να αξιοποιήσει ο χρήστης στο έπακρο τις διαθέσιμες τεχνολογίες. Ο χρήστης μπορεί πλέον με την έξυπνη συσκευή που διαθέτει, να έχει πρόσβαση εύκολα και γρήγορα σε πάρα πολλές υπηρεσίες και πληροφορίες που είναι διαθέσιμες μέσω του διαδικτύου.

Στα πλαίσια αυτής της εξέλιξης, και εστιάζοντας στο κομμάτι των έξυπνων κινητών τηλεφώνων, με την παρούσα μεταπτυχιακή διατριβή, γίνεται μελέτη σε μία νέα προσέγγιση ανάπτυξης εφαρμογών για έξυπνες συσκευές, η οποία εστιάζει στη συγγραφή ενιαίου κώδικα, και στη δημιουργία εκτελέσιμου αρχείου για την πλειοψηφία των λειτουργικών συστημάτων που υποστηρίζονται από τις έξυπνες συσκευές.

Η μελέτη και η υλοποίηση της προσέγγισης που πραγματεύεται η παρούσα εργασία, γίνεται με χρήση της πλατφόρμας Phonegap (Nitobi, 2011), η οποία αποτελεί ένα καινοτόμο εργαλείο ανάπτυξης εγγενών προγραμμάτων για έξυπνες συσκευές και δη για έξυπνα κινητά τηλέφωνα (smart phones). Το αντικείμενο της συγκεκριμένης υλοποίησης είναι η επικοινωνία με μια υπηρεσία διαδικτύου μέσω της χρήσης προκαθορισμένου πρωτοκόλλου επικοινωνίας και ανταλλαγής δεδομένων, το οποίο έχει οριστεί από την υπηρεσία διαδικτύου.

Προκειμένου η μελέτη και η υλοποίηση, της εφαρμογής να προσομοιωθεί σε όσο το δυνατόν ρεαλιστικότερες συνθήκες, και όχι σε μία ad hoc τεχνική και επιχειρησιακή παραδοχή, αναπτύχθηκε μια συνεργασία με την επιχείρηση Spitogatos.gr. Η επιχείρηση προσφέρθηκε και διέθεσε όλες της απαραίτητες τεχνικές πληροφορίες σχετικά με την υλοποιημένη υπηρεσία διαδικτύου που χρησιμοποιεί για τις επιχειρηματικές της δραστηριότητες με αποτέλεσμα να δημιουργηθεί μια πρότυπη εφαρμογή για κινητά τηλέφωνα, η οποία επικοινωνεί με την υπηρεσία διαδικτύου της επιχείρησης.

Με βάση τη δεδομένη υπηρεσία διαδικτύου και το χώρο που δραστηριοποιείται η εν λόγω επιχείρηση, διαμορφώθηκε μια εφαρμογή η οποία διαχειρίζεται πληροφορίες σχετικές με το χώρο της αγοράς ακινήτων. Πιο συγκεκριμένα, ο χρήστης, έχει την δυνατότητα να συμπληρώσει μια φόρμα από την έξυπνη συσκευή που διαθέτει, με στοιχεία που αφορούν ένα ακίνητο, και να

κάνει καταχώρηση των στοιχείων αυτών μέσω της έξυπνης συσκευής στην υπηρεσία διαδικτύου της επιχείρησης. Ταυτόχρονα, μπορεί να αξιοποιήσει και επιμέρους λειτουργίες της έξυπνης συσκευής προκειμένου να εμπλουτίσει το περιεχόμενο της αγγελίας που θέλει να δημοσιεύσει.

Κατά την εκπόνηση της μεταπτυχιακής διατριβής, χρειάστηκε η μελέτη αρκετών διαφορετικών τεχνολογιών που εμπλέκονταν στην υλοποίηση της εργασίας. Για την επιτυχή ολοκλήρωση της εργασίας, ήταν καταλυτική η βοήθεια τόσο του αξιότιμου καθηγητή κυρίου Χρήστου Δουληγέρη που επέβλεπε την πρόοδο της εργασίας. Επίσης εξίσου πολύτιμη ήταν και η βοήθεια του αξιότιμου διδάκτορα κυρίου Βασίλη Μενεκλή, ο οποίος επεσήμανε εύστοχα διάφορα προβλήματα που έπρεπε να μελετηθούν, και έδινε τις κατευθυντήριες γραμμές προκειμένου να γίνει πλήρης και εμπειριστατωμένη μελέτη. Επίσης, συνέβαλε αποφασιστικά και στη σύναψη της συνεργασίας με την επιχείρηση, γεγονός που αποτέλεσε ιδιαίτερη πρόκληση κατά την εκπόνηση της διατριβής, καθώς προσομοίαζε ως ένα βαθμό και την ενδοεπιχειρησιακή διαδικασία υλοποίησης ενός έργου.

Θέλω να ευχαριστήσω από καρδιάς και τους δύο για την καθοδήγηση και τη συμπαράσταση που μου παρείχαν καθόλη την εξέλιξη της υλοποίησης του προγράμματος και της συγγραφής της μεταπτυχιακής διατριβής, αλλά και όλους τους διδάσκοντες του μεταπτυχιακού προγράμματος σπουδών «Πληροφορική» του Πανεπιστημίου Πειραιώς για την πολύ καλή συνεργασία κατά τη διάρκεια του κύκλου σπουδών.

Τέλος θέλω να ευχαριστήσω την οικογένειά μου, για την υλική και ηθική στήριξη που μου παρείχε προκειμένου να συμμετάσχω και να ολοκληρώσω το μεταπτυχιακό πρόγραμμα σπουδών «Πληροφορική».

Κεφάλαιο 1

1.1 Εισαγωγή

Τα κινητά τηλέφωνα, αποτελούν πλέον κομμάτι της καθημερινότητας πολλών εκατομμυρίων ανθρώπων. Αρχικά για βασικές λειτουργίες, όπως κλήσεις και αποστολή σύντομων μηνυμάτων, ενώ με το πέρασμα των χρόνων, και την εισαγωγή των «έξυπνων» κινητών τηλεφώνων (smartphones) οι δυνατότητες τους άρχισαν να γίνονται πολύ ευρύτερες και αυτό έχει ως συνεπακόλουθο οι απαιτήσεις των χρηστών για νέες λειτουργίες και υπηρεσίες να γίνονται ολοένα και περισσότερες με πολύ γρήγορο ρυθμό.

Η παρούσα μεταπτυχιακή διατριβή πραγματεύεται την υλοποίηση μίας εφαρμογής για έξυπνες συσκευές η οποία θα έχει στοχευμένη λειτουργία και θα ενσωματώνει και θα παρέχει συγκεκριμένες υπηρεσίες στον χρήστη.

Η ιδέα είναι να δημιουργηθεί μια εφαρμογή η οποία θα εκτελείται στην έξυπνη συσκευή και με τη χρήση του πρωτοκόλλου XML-RPC θα επικοινωνεί με μια υλοποιημένη διαδικτυακή υπηρεσία (web service). Πιο συγκεκριμένα, η διαδικτυακή υπηρεσία με την οποία θα επικοινωνεί η εφαρμογή της πτυχιακής παρέχει τη λειτουργικότητα για καταχώρηση αγγελιών ακινήτων στη διαδικτυακή πύλη του Spitogatos.gr. Η ιδιαιτερότητα της συγκεκριμένης ιδέας και ταυτόχρονα η ιδιαίτερη πρόκληση, είναι η όλη ανάπτυξη της εφαρμογής να υλοποιηθεί με τη χρήση μιας καινοτομικής «πλατφόρμας», στον χώρο της ανάπτυξης λογισμικού για έξυπνες συσκευές (smartphones) η οποία ονομάζεται Phonegap (Nitobi, 2011), ενώ ταυτόχρονα, η εφαρμογή δε γίνεται πάνω σε μια εικονική δικτυακή υπηρεσία φτιαγμένη ειδικά για τις ανάγκες της παρούσας διατριβής, αλλά σε ήδη υπάρχουσα και λειτουργική διαδικτυακή υπηρεσία μιας επιχείρησης.

Το ιδιαίτερο χαρακτηριστικό, που ταυτόχρονα αποτελεί και τη μεγάλη πρόκληση στην επαλήθευση της δυνατότητας να πραγματοποιηθεί η αρχική ιδέα, είναι ότι σε αντίθεση με την παραδοσιακή ανάπτυξη εφαρμογών, η χρήση της πλατφόρμας Phonegap, επιτρέπει στον προγραμματιστή, να αναπτύξει έναν ενιαίο και κοινό πηγαίο κώδικα, από τον οποίο θα δημιουργούνται εφαρμογές για ορισμένα λειτουργικά συστήματα κινητών. Το ιδιαίτερο αυτό χαρακτηριστικό, θέτει παράλληλα και σημαντικούς περιορισμούς αναφορικά με τις υποστηριζόμενες γλώσσες προγραμματισμού αλλά και τις τεχνολογίες τις οποίες μπορεί να χρησιμοποιήσει για την ανάπτυξη της εφαρμογής.

Κατά τη δημιουργία μιας εγγενούς εφαρμογής (native application) για μια έξυπνη συσκευή, ο προγραμματιστής, παραδοσιακά χρησιμοποιεί τη γλώσσα προγραμματισμού η οποία είναι συμβατή με το εν λόγω λειτουργικό σύστημα για το οποίο προορίζεται η εφαρμογή, ενώ ταυτόχρονα μπορεί να έχει πλήρη πρόσβαση σε εξειδικευμένες λειτουργίες της έξυπνης συσκευής. Αυτήν ακριβώς τη λειτουργικότητα, προσπαθεί να εξασφαλίσει και το Phonegap, στον προγραμματιστή, (πλήρη πρόσβαση στις εξειδικευμένες λειτουργίες των εκάστοτε έξυπνων συσκευών) με τη διαφορά ότι δεν απαιτεί ο πηγαίος κώδικας να γραφεί σε εγγενή γλώσσα της στοχευμένης πλατφόρμας, ανάλογα με το λειτουργικό σύστημα, αλλά σε γλώσσες προγραμματισμού οι οποίες είναι περισσότερο «οικείες» σε πολλούς προγραμματιστές, όπως η HTML5, η JavaScript και η CSS3.

1.2 Το αντικείμενο της διατριβής

Η βασική ιδέα που πραγματεύεται η διατριβή, είναι η υλοποίηση μιας εφαρμογής, η οποία θα δίνει την δυνατότητα στο χρήστη να κάνει καταχώρηση μιας αγγελίας στη διαδικτυακή πύλη της επιχείρησης Spitogatos.gr μέσω της χρήσης της διαδικτυακής υπηρεσίας που παρέχεται από την επιχείρηση.

Για το λόγο αυτό, αναπτύχθηκε μια εγγενής εφαρμογή (native application) με κοινό πηγαίο κώδικα για το σύνολο των διαφορετικών λειτουργικών συστημάτων των σύγχρονων έξυπνων συσκευών (smartphones), η οποία θα επιτρέπει στο χρήστη να εισάγει τα επιθυμητά δεδομένα, αλλά και να χρησιμοποιεί ορισμένες από τις λειτουργίες των έξυπνων συσκευών όπως ο δέκτης γεωγραφικού στίγματος, αλλά και της ενσωματωμένης φωτογραφικής μηχανής των κινητών τηλεφώνων.

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

Προκειμένου να επικοινωνήσει η εφαρμογή με την υλοποιημένη διαδικτυακή υπηρεσία (web service), θα πρέπει το σύνολο των δεδομένων που εισάγει ο χρήστης κατά τη χρήση της εφαρμογής, να μετασχηματίζεται και να μορφοποιείται σε μια προκαθορισμένη μορφή-δομή XML αρχείου (Extensible Markup Language). Η προκαθορισμένη μορφοποίηση των δεδομένων με βάση τόσο τους κανόνες της XML όσο και των απαιτήσεων του ιστότοπου, θα αποστέλλεται με ένα HttpRequest στη διαδικτυακή υπηρεσία (web service) της επιχείρησης, η οποία θα επαληθεύει την εγκυρότητα και την ορθότητα των δεδομένων αλλά και της δομής της XML μορφοποίησης του περιεχομένου που αποστέλλεται και στη συνέχεια θα καταχωρείται στη βάση δεδομένων του ιστότοπου, της επιχείρησης.

Οι παρακάτω απαιτήσεις θεωρήθηκαν βασικές για την εφαρμογή:

1. να διαθέτει κατανοητό και φιλικό προς το χρήστη περιβάλλον επικοινωνίας (user interface) έτσι ώστε να μπορεί ο χρήστης να τη χρησιμοποιεί με ευκολία,
2. να μπορεί να συλλέγει όλα τα στοιχεία που θα επιθυμεί να εισάγει ο χρήστης και σχετίζονται με το ακίνητο προς δημοσίευση,
3. να εμφανίζει κατανοητά μηνύματα σε περίπτωση που ο χρήστης παραλείψει κάποια ενέργεια η οποία θεωρείται απαραίτητη,
4. να μετατρέπει όλα τα στοιχεία που εισάγει ο χρήστης σε συμβατή μορφή και δομή με αυτήν που αναγνωρίζει η διαδικτυακή υπηρεσία του Spitogatos.gr (πρωτόκολλο XML-RPC)
5. να αποστέλλει τα στοιχεία στην κατάλληλη μορφή και δομή, έτσι ώστε να καταχωρούνται στη συνέχεια με επιτυχία στη βάση δεδομένων της Spitogatos.gr,
6. να ενημερώνει τον χρήστη για την επιτυχή καταχώρηση των δεδομένων μέσω της διαδικτυακής υπηρεσίας της επιχείρησης, παρέχοντας και οποιεσδήποτε άλλες πληροφορίες που μπορεί να κρίνονται απαραίτητες.

1.3 Σχετικές τεχνολογίες

Αν και η παρούσα διατριβή, ασχολείται με την δυνατότητα κατασκευής συγκεκριμένης εγγενούς εφαρμογής (native application) για κινητά τηλέφωνα, ωστόσο, για την υλοποίηση της ιδέας είναι απαραίτητη η μελέτη πολλών διαφορετικών τεχνολογιών αλλά και ειδικών λειτουργιών.

Η ανάπτυξη της εφαρμογής προϋποθέτει αφενός την επιτυχή αλληλεπίδραση μεταξύ του χρήστη και της εφαρμογής, και αφετέρου, την επιτυχή επικοινωνία και αλληλεπίδραση μεταξύ της εφαρμογής και της διαδικτυακής υπηρεσίας που υπάρχει ήδη στον ιστότοπο της επιχείρησης Spitogatos.gr.

Για την εκπόνηση της διατριβής, κρίθηκε σκόπιμο, πως, εκτός από τις απαιτούμενες τεχνολογίες που θα χρησιμοποιηθούν και πρέπει να μελετηθούν ώστε να χρησιμοποιηθούν αποτελεσματικά, χρειάζεται να μελετηθούν και τεχνολογίες που έχουν ήδη χρησιμοποιηθεί στην υπάρχουσα διαδικτυακή υπηρεσία (web service) της επιχείρησης έτσι ώστε να κατανοηθεί πλήρως η φύση του προβλήματος και των προκλήσεων που υπάρχουν, αλλά και να γίνει πληρέστερη ανάλυση τους ώστε να επιτευχθεί η επιθυμητή λειτουργικότητα.

Οι τεχνολογίες που σχετίζονται αμιγώς με την εφαρμογή είναι οι εξής:

- Phonegap: η πλατφόρμα Phonegap αποτελεί το εργαλείο που επιτρέπει στον προγραμματιστή να αναπτύξει εφαρμογές για κινητά τηλέφωνα με διαφορετικά λειτουργικά συστήματα (Apple - IOS, Google – Android, BlackBerry, Hewlett Packard – WebOS, Symbian) χρησιμοποιώντας κοινό κώδικα.
- HTML (HyperText Markup Language) και πιο συγκεκριμένα η νεώτερη και αρκετά καινοτόμος HTML5 όπου αποτελεί τη βασική γλώσσα ανάπτυξης του κώδικα.
- JavaScript: η χρήση JavaScript, βοήθησε ώστε να γίνει επιτυχημένη συνέργεια των διαφορετικών υποσυστημάτων του κινητού τηλεφώνου, αλλά και ο επιτυχής μετασχηματισμός των δεδομένων, ώστε να καταχωρηθούν στην βάση δεδομένων της επιχείρησης.
- CSS3 (Cascading Style Sheets): μαζί με την HTML5 και την JavaScript, τα CSS ολοκληρώνουν το σύνολο των συμβατών γλωσσών προγραμματισμού τα οποία επιτρέπει το Phonegap να χρησιμοποιηθούν από το χρήστη, ώστε να αναπτύξει την εφαρμογή.

- XML (Extensible Markup Language): αποτελεί έναν πολύ διαδομένο και αποτελεσματικό τρόπο, «αποθήκευσης» πληροφοριών στο διαδίκτυο.
- XML-RPC (EXTensible Markup Language -Remote Procedure Calling): είναι ένα πρωτόκολλο που επιτρέπει σε προγράμματα που τρέχουν σε διαφορετικά και ανόμοια λειτουργικά περιβάλλοντα, να κάνουν κλήσεις διαδικασιών μέσω του διαδικτύου.
- Web services: υπηρεσίες δικτύου, όπου επιτρέπουν την επικοινωνία δύο «μερών» μέσω του διαδικτύου.

Εκτός από τις αμιγώς σχετιζόμενες, με την ανάπτυξη της εφαρμογής τεχνολογίες, χρειάστηκε να μελετηθούν και τεχνολογίες οι οποίες επηρέαζαν έμμεσα την ανάπτυξη αλλά και την λειτουργικότητα της εφαρμογής.

- Cross-Site Scripting: κατά τη μεταφορά δεδομένων με HttpRequest με χρήση XML-RPC πρωτοκόλλου, προκύπτουν σημαντικά θέματα ασφάλειας, τα οποία, ειδικά στη χρήση των περιηγητών (web browsers), περιορίζονται με τον αποκλεισμό χρήσης ορισμένων τεχνολογιών για την επίτευξη επικοινωνίας μεταξύ δύο μερών, προκειμένου να μεγιστοποιηθεί το επίπεδο ασφάλειας του τελικού χρήστη.
- Minification: πρόκειται για τον περιορισμό του μεγέθους του κώδικα (ειδικά σε JavaScript) με την αφαίρεση όλων των περιττών χαρακτήρων από τον κώδικα, όπως ο «κενός» χαρακτήρας ή αλλαγή γραμμής κλπ

Το Phonegap, αποτελεί κάτι εντελώς νέο στο χώρο της ανάπτυξης λογισμικού για κινητά τηλέφωνα, και μάλιστα, κατά την έναρξη εκπόνησης της μεταπτυχιακής διατριβής και ανάπτυξης της εφαρμογής, η ίδια η πλατφόρμα (Phonegap) βρισκόταν ακόμα σε δοκιμαστικό "beta" στάδιο ανάπτυξης, με όσα προβλήματα και ιδιαιτερότητες μπορεί να συνεπάγεται αυτό. Ωστόσο με την ανταπόκριση και αποδοχή που βρήκε από αρκετούς προγραμματιστές, κατά την εξέλιξη της μεταπτυχιακής διατριβής, έγινε διαθέσιμη και η πρώτη σταθερή έκδοση του Phonegap, το οποίο εξακολουθεί να εξελίσσεται δυναμικά.

Ένα σημαντικό στοιχείο αλλά ταυτόχρονα και πρόκληση για τους προγραμματιστές, αποτελεί το γεγονός ότι το Phonegap, επιτρέπει τη χρήση της HTML5. Αυτό το γεγονός δίνει αρκετά περισσότερες δυνατότητες αναφορικά με την έως τώρα ανάπτυξη κώδικα με χρήση HTML, προσφέροντας επιπλέον δυναμική στο Phonegap αφού πλέον μπορούν οι προγραμματιστές να ενσωματώσουν διάφορες λειτουργίες, πιο απλά σε σχέση με τις προηγούμενες εκδόσεις της HTML.

Η δυναμική που έχει ήδη διαμορφωθεί αλλά και συνεχίζει να εξελίσσεται και να ενισχύεται αναφορικά με την ανάπτυξη εφαρμογών, για «έξυπνες» συσκευές και ιδιαίτερα σε «έξυπνα» κινητά τηλέφωνα, είναι συνδεδεμένη με την ταχεία ανάπτυξη που σημειώνεται στον τομέα των κινητών τηλεφώνων, τόσο σε επίπεδο υλικού (hardware) όσο και σε επίπεδο λογισμικού (software). Οι χρήστες, πλέον επιθυμούν να μπορούν να υλοποιήσουν αρκετές λειτουργίες που παραδοσιακά χρειάζονταν έναν σταθερό ηλεκτρονικό υπολογιστή (desktop) ή ένα φορητό υπολογιστή (laptop). Στο γεγονός αυτό, έχει συμβάλει καταλυτικά και η ολοένα και ευρύτερη δυνατότητα για πρόσβαση στο διαδίκτυο με προσιτό κόστος, τόσο μέσω WiFi, όσο και μέσω δικτύων κινητής τηλεφωνίας 3G.

Είναι η ίδια δυναμική, που έχει διαμορφώσει έναν αγώνα δρόμου ανάμεσα σε εταιρείες και προγραμματιστές για την ανάπτυξη τόσο native όσο και web εφαρμογών, με στόχο η εκάστοτε επιχείρηση να παρέχει τη μέγιστη χρησιμότητα στους πελάτες της, κερδίζοντας έτσι ανταγωνιστικό πλεονέκτημα σε σχέση με «αντίπαλους» επιχειρήσεις, με βραδύτερα ίσως αντανακλαστικά (αναφορικά με την υιοθέτηση νέων τεχνολογιών) (McIntosh, 2005).

Με βάση αυτά τα χαρακτηριστικά, επιδιώχθηκε η ανάπτυξη της εφαρμογής στα πλαίσια της μεταπτυχιακής διατριβής, να γίνει σε μια πραγματική επιχείρηση, ώστε να συνδυαστεί η ακαδημαϊκή πτυχή της διατριβής, με τις πραγματικές, εμπορικές και επιχειρησιακές τεχνολογίες και απαιτήσεις που ενσωματώνει η επιχείρηση Spiritogatos.gr. Φυσικά σημαντικό ρόλο έπαιξε και η προθυμία της επιχείρησης να διαθέσει τις απαραίτητες πληροφορίες και στοιχεία για την ανάπτυξη της εφαρμογής.

Ο χώρος των έξυπνων κινητών τηλεφώνων, και ιδιαίτερα των εφαρμογών που προορίζονται για τις έξυπνες συσκευές, αποτελεί ένα χώρο που αναπτύσσεται ταχύτατα και αποτελεί ευκαιρία δραστηριοποίησης και επίτευξη ανταγωνιστικού πλεονεκτήματος για όσες επιχειρήσεις αντιληφθούν την δυναμική του. Οι εφαρμογές για έξυπνες συσκευές, δίνουν ένα

στρατηγικό πλεονέκτημα στις επιχειρήσεις (Atkins, et al., 2006), καθώς τους επιτρέπουν να προσεγγίσουν περισσότερους πελάτες, να αυξήσουν την επιχειρηματική αποτελεσματικότητα και να μειώσουν τα λειτουργικά κόστη (Vriddhi Technology Services, 2012). Ωστόσο μια εφαρμογή, πρέπει να έχει τα κατάλληλα στοιχεία, τα οποία θα την κάνουν προσιτή και αποδεκτή από τους χρήστες και αυτά είναι:

- Φιλικότητα προς το χρήστη: να έχει περιβάλλον αλληλεπίδρασης με το χρήστη (user interface), το οποίο θα είναι κατανοητό και απλό προς το μέσο χρήστη, ώστε να επιτευχθεί η μέγιστη αποδοχή και χρήση της εφαρμογής από τους χρήστες.
- Λειτουργικότητα: θα πρέπει η εφαρμογή, να καταφέρνει να «φέρνει εις πέρας» αυτό για το οποίο προορίζεται, και να μην «προβληματίζει» το χρήστη με δυσνόητα σφάλματα και μηνύματα που θα προκαλούν σύγχυση κατά τη χρήση της εφαρμογής.
- Αποτελεσματικότητα: να επιτελεί το ρόλο της αποτελεσματικά, χρησιμοποιώντας τους διαθέσιμους πόρους και τεχνολογίες με όσο το δυνατόν πιο αποτελεσματικό τρόπο.

Εξαρχής, στόχος της διατριβής, ήταν η απόδειξη της ιδέας (proof of concept) για το κατά πόσο ήταν εφικτή η υλοποίηση μιας εφαρμογής η οποία με τη χρήση του Phonegar, θα κατάφερε να επικοινωνήσει επιτυχώς με μία υπάρχουσα διαδικτυακή υπηρεσία (web service consumption). Η έρευνα και η ανάλυση εστιάστηκε στο μεγαλύτερο βαθμό στο να επιτευχθεί η επιθυμητή αποτελεσματικότητα της εφαρμογής αναφορικά με την επικοινωνία με την υπηρεσία δικτύου (web service) της επιχείρησης, αλλά και με τη λειτουργικότητα ώστε να μην προβληματίζει το χρήστη κατά τη χρήση της, γεγονός το οποίο κατέστη δυνατό μέσα από αρκετή συστηματική εργασία σχετικά με τις διαθέσιμες δυνατότητες ανάπτυξης και τις προσφερόμενες τεχνολογίες.

Στο κομμάτι της φιλικότητας προς το χρήστη, δόθηκε στοιχειώδης προσοχή, με βάση την παραδοχή πως η φιλικότητα στο χρήστη, μπορεί να απαιτεί από μόνη της ενσωμάτωση τεχνολογιών, αλλά και έρευνα σχετικά με μελέτες που αναδεικνύουν πτυχές που έχουν μεγαλύτερη απήχηση στους χρήστες κατά τη χρησιμοποίηση μιας εφαρμογής, γεγονός που μπορεί να αποτελέσει πεδίο ξεχωριστής και ιδιαίτερης μελέτης.

1.4 Διαδικασία ανάπτυξης

Για την εκπόνηση της διατριβής, ήταν πολύ σημαντικό να αναλυθούν όλες οι πτυχές του προβλήματος αναλυτικά, ώστε να προσδιοριστούν με σαφήνεια οι στόχοι που έπρεπε να επιτευχθούν. Για να καταστεί αυτό δυνατό χρειάστηκε να γίνουν τα ακόλουθα βήματα:

- Εξερεύνηση της έννοιας που πραγματεύεται το θέμα της διατριβής και προσδιορισμός του προβλήματος.
- Ανάλυση των απαιτήσεων για την επίλυση του προβλήματος.
- Σχεδιασμός του προγράμματος (εφαρμογής).
- Υλοποίηση του προγράμματος.
- Έλεγχος του προγράμματος και επαναπροσδιορισμός απαιτήσεων και ζητημάτων που προέκυψαν κατά την υλοποίηση και τον έλεγχο.

1.5 Προκλήσεις κατά την εργασία

Κατά τη διάρκεια της εκπόνησης της μεταπτυχιακής διατριβής, και ειδικότερα στη διαδικασία ανάπτυξης του λογισμικού αντιμετωπίστηκαν αρκετές δυσκολίες τόσο σχετικά με την ανάπτυξη του προγράμματος καθαυτή, όσο και με επιμέρους θέματα, όπως την διαδικασία αποσφαλμάτωσης (debugging), τους περιορισμούς ασφαλείας που έθεταν οι «περιρηγητές», αλλά και ορισμένες λειτουργικές απαιτήσεις της διαδικτυακής υπηρεσίας της επιχείρησης. Ένα μέρος των προβλημάτων θα μπορούσε να αποδοθεί και στο γεγονός ότι το Phonegar, βρισκόταν και το ίδιο σε φάση ανάπτυξης, με αποτέλεσμα πολλά θέματα να είναι είτε ακόμα υπό διερεύνηση, είτε να μην είναι αρκετά σαφής ο τρόπος αντιμετώπισής τους.

Ανάμεσα στις διάφορες δυσκολίες που αντιμετωπίστηκαν κατά τη διαδικασία ανάπτυξης της εφαρμογής, θα μπορούσε να εστιαστεί ιδιαίτερα η προσοχή σε τρία συγκεκριμένα προβλήματα, τα οποία κατά την εμφάνισή τους (το κάθε ένα σε διαφορετική χρονική περίοδο) κλόνησαν το

κατά πόσο πράγματι θα μπορούσε να «αποδειχθεί» η ιδέα που πραγματευόταν η συγκεκριμένη διατριβή.

Το πρώτο σημαντικό πρόβλημα που υπήρχε ήταν η περιορισμένη δυνατότητα αποσφαλμάτωσης (debugging) στο εγγενές περιβάλλον κινητού, ή ακόμα και σε προσομοιωτή (emulator). Αυτό είχε σαν αποτέλεσμα να μην είναι δυνατός ο προσδιορισμός αρκετών προβλημάτων, ειδικά στα πρώτα στάδια εξέλιξης της διατριβής. Για την επίλυση αυτού του προβλήματος, κρίθηκε επιτακτική (εφόσον η ανάπτυξη γίνεται σε HTML με JavaScript και CSS) η ανάγκη να δοκιμάζεται ο κώδικας, σε περιηγητές (browsers) οι οποίοι ενσωματώνουν κατάλληλα εργαλεία ανάπτυξης για προγραμματιστές με αναλυτικά στοιχεία για τη φύση του προβλήματος. Για το λόγο αυτό χρησιμοποιήθηκαν οι Firefox (με χρήση του FireBug) και ο Google Chrome.

Με την επίλυση του πρώτου προβλήματος και εν μέρει ως απόρροια αυτού, προέκυψε το δεύτερο πρόβλημα. Το συγκεκριμένο πρόβλημα που προέκυψε, χαρακτηρίστηκε, κατά τη διαδικασία εκπόνησης της διατριβής σαν σημαντικό σημείο καμπής, καθώς έθεσε όλη την ιδέα υπό αμφισβήτηση, από τα θεμέλιά της. Το συγκεκριμένο θέμα αφορούσε τη δυνατότητα ή μη υλοποίησης αιτημάτων μεταφοράς δεδομένων με χρήση JavaScript από την εφαρμογή. Ο λόγος που σχετίζεται το συγκεκριμένο πρόβλημα με τη διαδικασία επίλυσης του πρώτου, είναι το γεγονός, ότι το Phoneygar, σαν πλατφόρμα, δε θέτει περιορισμούς αιτημάτων μεταφοράς δεδομένων με χρήση JavaScript, κάτι το οποίο δεν ισχύει και με τους περιηγητές (browsers) καθώς η συγκεκριμένη λειτουργία, εμπίπτει στους περιορισμούς ασφαλείας που έχουν ενσωματωμένους οι περιηγητές αναφορικά με την επικοινωνία δύο απομακρυσμένων συσκευών με χρήση JavaScript το οποίο συναντάται στη βιβλιογραφία ως cross-site scripting (Anon., 2011). Εφόσον όμως, όλη η διαδικασία αποσφαλμάτωσης βασιζόταν στην χρήση των περιηγητών, οι ίδιοι οι περιηγητές προκαλούσαν «πρόβλημα», στην εξέλιξη της εφαρμογής καθώς λόγω των περιορισμών, δεν μπορούσε να επαληθευτεί η λειτουργικότητα της εφαρμογής. Πρόκειται για το σημείο, όπου ουσιαστικά εντοπίστηκε αυτή η σημαντική διαφορά ανάμεσα στη λειτουργία του Phoneygar, και των περιηγητών, καθώς το θέμα του cross-site scripting, ως περιορισμός, δεν είχε απασχολήσει νωρίτερα την ανάπτυξη, με αποτέλεσμα να μην ανιχνευθεί έγκαιρα η συγκεκριμένη ιδιαιτερότητα, αλλά και να χρειαστεί ένα σημαντικό διάστημα μέχρι να προσδιοριστεί επ' ακριβώς η φύση του προβλήματος.

Το τρίτο βασικό πρόβλημα, αφορούσε την επικοινωνία με τη διαδικτυακή υπηρεσία του Spitogatos.gr. Η αλληλεπίδραση με τη διαδικτυακή υπηρεσία της επιχείρησης καθώς και όποιες δοκιμές θα έπρεπε να μην επηρέαζαν την εύρυθμη λειτουργία της εταιρείας. Για το λόγο αυτό και προκειμένου να εξασφαλιστεί ότι δε θα επηρεαστεί με κανένα τρόπο η λειτουργία της ηλεκτρονικής υπηρεσίας, η επιχείρηση, διατηρεί δύο διακριτές υπηρεσίες. Η μία υπηρεσία αφορά το κομμάτι της ανάπτυξης, όπου γίνονται όλες οι δοκιμές εισαγωγής νέων τεχνολογιών. Η δεύτερη υπηρεσία, είναι φυσικά το εμπορικό τμήμα-εμπορική υπηρεσία, όπου είναι αυτή που χρησιμοποιεί ο τελικός χρήστης. Οι δοκιμές, συμφωνήθηκε να γίνουν στην υπηρεσία ανάπτυξης, η οποία όμως, απαιτούσε επιβεβαίωση ονόματος χρήστη και κωδικού πρόσβασης (Http-Authentication) για πρόσβαση στις επιμέρους υπηρεσίες, γεγονός που δημιούργησε επίσης αρκετά προβλήματα καθώς δεν είχε βρεθεί συμβατός κώδικας που να μπορεί να ενσωματώσει άμεση επαλήθευση των στοιχείων του χρήστη κατά την αποστολή του αιτήματος HttpRequest.

1.6 Δομή της διατριβής

Σε αυτό το κεφάλαιο γίνεται αναφορά στο αντικείμενο της διατριβής, τα προβλήματα και τις δυσκολίες που αντιμετωπίστηκαν, τις προκλήσεις που προέκυψαν, αλλά και τις τεχνολογίες που ενσωματώθηκαν και μελετήθηκαν κατά την εκπόνησή της.

Στο επόμενο κεφάλαιο (δεύτερο) που ακολουθεί γίνεται αναλυτική περιγραφή των τεχνολογιών που χρησιμοποιήθηκαν, αλλά και των τεχνολογιών που μελετήθηκαν αλλά για διάφορους λόγους δε χρησιμοποιήθηκαν τελικά. Επίσης αναλύονται θέματα με τεχνολογίες που χρειάστηκε να μελετηθούν για να επιλυθούν διάφορα προβλήματα όπως, για παράδειγμα, οι περιορισμοί που έθεταν οι περιηγητές στην εκτέλεση απομακρυσμένων αιτημάτων.

Στο τρίτο κεφάλαιο, γίνεται αναλυτική περιγραφή της λειτουργίας της εφαρμογής. Περιγράφεται η αρχιτεκτονική της, η λειτουργικότητά της, οι περιορισμοί που υπάρχουν καθώς και όποιες παραδοχές έγιναν κατά την υλοποίησή της.

Στο τέταρτο κεφάλαιο, γίνεται ανακεφαλαίωση όλων όσων έχουν αναφερθεί, οι περιορισμοί που προέκυψαν κατά την εκπόνηση της διατριβής καθώς και οι πιθανές μελλοντικές επεκτάσεις της εφαρμογής. Επίσης αναλύονται τα συμπεράσματα που προέκυψαν τόσο από τη χρήση του Phonegap όσο και των υπολοίπων τεχνολογιών που χρησιμοποιήθηκαν, ενώ γίνεται και ειδική αναφορά στα smartphones και τον ρόλο που φαίνονται να αρχίζουν να διαδραματίζουν στην καθημερινότητα πολλών ανθρώπων.

Τέλος, υπάρχουν δύο παραρτήματα μετά την ολοκλήρωση των κεφαλαίων. Στο πρώτο παράρτημα υπάρχει αναλυτικό εγχειρίδιο χρήσης της εφαρμογής το οποίο απευθύνεται στον χρήστη, με αναλυτικές οδηγίες και εικόνες από την εφαρμογή, προκειμένου να τον κατατοπίσει για την ορθή χρήση της. Επίσης υπάρχει και εγχειρίδιο για τον προγραμματιστή όπου αναφέρονται και επεξηγούνται κάποια βασικά σημεία του κώδικα της εφαρμογής. Στο δεύτερο παράρτημα υπάρχει ο κώδικας της εφαρμογής, με σχόλια ώστε να είναι ευκολότερη η ανάγνωσή του και ο προσδιορισμός της χρήσης των εκάστοτε συναρτήσεων ή μεταβλητών.

Κεφάλαιο 2

Εισαγωγή

Κατά τη διαδικασία εκπόνησης της παρούσας μεταπτυχιακής διατριβής, χρειάστηκε να μελετηθούν αρκετά διαφορετικά αντικείμενα, και να γίνει σύνθεση των κατάλληλων πληροφοριών ώστε να εξαχθεί το επιθυμητό αποτέλεσμα. Όταν, δε, πρόκειται για ανάπτυξη εφαρμογής, η διαδικασία της ανάπτυξης συνεπάγεται και τη γνωριμία με αρκετές νέες τεχνολογίες, καθώς και την εξοικείωση με αυτές, αλλά και τη μελέτη άλλων τεχνολογιών ή στοιχείων που μπορεί να επηρεάζουν την έρευνα είτε άμεσα, είτε έμμεσα.

Κατά την πρώτη φάση του σχεδιασμού έκβασης της εργασίας για την εκπόνηση της διατριβής, έγινε σαφές, ότι τα άμεσα σχετιζόμενα αντικείμενα – τεχνολογίες που χρειαζόνταν να μελετηθούν, ήταν το Phoneyar, καθώς αποτελούσε τη βασική πλατφόρμα πάνω στην οποία θα βασιζόταν η εφαρμογή, η HTML5, η JavaScript και τα CSS3 (Cascading Styles Sheets) τα οποία συνιστούν το σύνολο των υποστηριζόμενων τεχνολογιών για το Phoneyar. Επίσης, καθώς αντικείμενο της διατριβής ήταν η ανάπτυξη εφαρμογής για επικοινωνία με μια υπηρεσία δικτύου (web service) μέσω του πρωτοκόλλου XML-RPC, ήταν εύλογο πως και αυτές οι τεχνολογίες έπρεπε να μελετηθούν, ώστε να αποφασιστεί ποιος θα ήταν ο κατάλληλος τρόπος για να γίνουν διαχειρίσιμες.

Κατά τη διαδικασία ανάπτυξης της εφαρμογής, χρειάστηκε να μελετηθεί τόσο το «cross-site scripting» και ειδικότερα οι περιορισμοί που το συνόδευαν και φάνηκαν κατά την εξέλιξη της εφαρμογής, αλλά και το «minification», η διαδικασία σύμπτυξης δηλαδή ενός αρχείου π.χ. JavaScript.

2.1 Αναλυτική περιγραφή του πλαισίου

2.1.1 Παρόμοιες προσπάθειες ανάπτυξης

Η ανάπτυξη των έξυπνων συσκευών, και η ανάγκη παροχής υπηρεσιών στον τελικό χρήστη, έχει σαν αποτέλεσμα να υπάρχουν αρκετές μελέτες σχετικά με την επικοινωνία των έξυπνων κινητών τηλεφώνων με διάφορες υπηρεσίες διαδικτύου. Ανάλογα με την εκάστοτε προσέγγιση ο ερευνητής προσπαθεί να επιτύχει ένα συγκεκριμένο στόχο από την πλευρά του.

Ήδη, πολύ πριν την δυνατότητα σύνδεσης των έξυπνων συσκευών μέσω ασύρματων δικτύων (WiFi) και δικτύων κινητής τηλεφωνίας, 3G, υπήρχε η αναζήτηση δυνατότητας πρόσβασης σε διαδικτυακές υπηρεσίες, μέσω του ασύρματου πρωτοκόλλου εφαρμογής – WAP (Wireless Application Protocol) (World Wide Web Consortium (W3C), 1999), το οποίο χρησιμοποιούταν για πρόσβαση στο διαδίκτυο στις «πρώτες» έξυπνες συσκευές, σε αντίθεση με τα σημερινά δεδομένα, όπου η χρήση του πρωτοκόλλου HTTP είναι δεδομένη μέσω των έξυπνων συσκευών. Σε ένα επιστημονικό άρθρο το 2004, μελετάται η δυνατότητα επικοινωνίας με μια υπηρεσία διαδικτύου μέσω WAP, με χρήση πλατφόρμας J2ME (Java 2 Platform Micro Edition), (Gehlen & Bergs, 2004). Οι ερευνητές, με χρήση του πρωτοκόλλου SOAP (Simple Object Access Protocol), μελέτησαν την επικοινωνία και την αποστολή δεδομένων μέσω της υπηρεσίας διαδικτύου μετάφρασης της BabelFish, με χρήση WBXML (Wap Binary XML).

Η χρησιμότητα των υπηρεσιών διαδικτύου, και το πώς μπορούν αυτές να προσφέρουν επιπλέον χρησιμότητα στον τελικό χρήστη, και οφέλη στην επιχείρηση, τονίζεται στο άρθρο Mobile Web Services (Farley & Capp, 2005). Ωστόσο, σε αντίθεση με την παρούσα διατριβή, όπου χρησιμοποιήθηκε μια υπηρεσία διαδικτύου η οποία εξυπηρετούσε διαφορετικές ανάγκες, στο συγκεκριμένο άρθρο, τονίζεται, η χρησιμότητα εξειδικευμένων υπηρεσιών διαδικτύου για κινητές συσκευές. Ταυτόχρονα, στην έρευνα, θεωρείται ως δεδομένος περιοριστικός παράγοντας, η χαμηλή επεξεργαστική ισχύς των έξυπνων συσκευών, αλλά και το περιορισμένο εύρος στην επικοινωνία με το διαδίκτυο, ώστε να δικαιολογηθεί η ύπαρξη συγκεκριμένων υπηρεσιών διαδικτύου αποκλειστικά για έξυπνες συσκευές, όμως αυτοί οι περιορισμοί δεν συντρέχουν πλέον. Επίσης, προτείνεται και μια ολοκληρωμένη αρχιτεκτονική μιας υπηρεσίας

διαδικτύου για κινητές συσκευές, και η επικοινωνία με μια τυπική υπηρεσία διαδικτύου, αλλά και τα οφέλη που προκύπτουν για τους καταναλωτές με την αλληλεπίδραση με τις υπηρεσίες διαδικτύου, μέσα από ένα εύρος διαφορετικών συσκευών, όπως κινητά τηλέφωνα, ηλεκτρονικοί υπολογιστές και pda.

Σε αντίστοιχο κλίμα, κινείται και το άρθρο A Web Services-based System for Ad-hoc Mobile Application Integration (Steele, 2003), όπου διαπιστώνεται η ευρεία χρήση των έξυπνων συσκευών, και η ανάγκη επικοινωνίας μέσω υπηρεσιών διαδικτύου. Ωστόσο, και σε αυτήν την περίπτωση, θεωρούνται ανασταλτικοί παράγοντες, το μικρό εύρος ζώνης και η χαμηλή επεξεργαστική ισχύς, τα οποία όμως δεν αποτελούν θέματα που αφορούν τις σύγχρονες συσκευές.

Τέλος, σε μια εργασία με τίτλο «Optimizing Web Service messaging performance in mobile computing» (Oh & Fox, 2007) οι ερευνητές μελετούν τον τρόπο βελτιστοποίησης της μεταφοράς δεδομένων μέσω υπηρεσιών διαδικτύου. Χαρακτηρίζοντας τη χρήση του πρωτοκόλλου SOAP και της αναπαράστασης των δεδομένων με χρήση XML, ως αρκετά «φλύαρη» προσπαθούν να διαχωρίσουν το περιεχόμενο των μηνυμάτων, από τη δομή τους, και χρησιμοποιούν μια μέθοδο που την ονομάζουν HHFR (Handheld Flexible Representation). Τα συμπεράσματα είναι πως μπορούν να υπάρξουν σημαντικά οφέλη και περιθώρια βελτιστοποίησης των υπηρεσιών διαδικτύου αναφορικά με την αλληλεπίδραση με έξυπνες συσκευές αναφορικά με το μέγεθος των ανταλλασσόμενων μηνυμάτων αλλά και το χρόνο μετάδοσής τους.

Σε αντίθεση ωστόσο με τις συγκεκριμένες έρευνες, το πρωτόκολλο επικοινωνίας με την υπηρεσία διαδικτύου στην παρούσα μεταπτυχιακή διατριβή, δεν προσδιορίστηκε με βάση τη βελτιστοποίηση των υπηρεσιών, ούτε αποτέλεσε αντικείμενο έρευνας, προκειμένου να χρησιμοποιηθεί κάποιο εναλλακτικό, καθώς ήταν εξαρχής δεδομένο. Η εφαρμογή, έπρεπε να αναπτυχθεί και αναπτύχθηκε με βάση την υπάρχουσα δομή που είχε η επιχείρηση. Στην προκειμένη περίπτωση, το πρωτόκολλο XML-RPC ήταν το πρωτόκολλο το οποίο χρησιμοποιεί η επιχείρηση για την επίτευξη των διάφορων λειτουργιών και επομένως ή όποια έρευνα και ανάλυση έπρεπε να γίνει πάνω σε αυτό. Ωστόσο, σε αντίθεση με τα μέσα και τις υποδομές που υπήρχαν στις έρευνες που αναφέρθηκαν, πλέον η επεξεργαστική ισχύς των έξυπνων συσκευών (εμφάνιση τετραπύρηνων επεξεργαστών), η διαθέσιμη μνήμη (πάνω από 1gb μνήμης ram), αλλά και η φθηνότερη πρόσβαση στο διαδίκτυο, αποτελούν στοιχεία τα οποία δίνουν μεγαλύτερη άνεση στον προγραμματιστή, προκειμένου να αναπτύξει μια εφαρμογή.

Αυτό έχει σαν αποτέλεσμα η ανάπτυξη μιας ad hoc υπηρεσίας διαδικτύου, προκειμένου να καλυφθούν ειδικές ανάγκες να μην αποτελεί μονόδρομο, αλλά να μπορούν πιο εύκολα να μελετηθούν εναλλακτικά σενάρια, και να κριθεί από τους ενδιαφερόμενους, σε όρους κόστους και διάθεσης πόρων ποια λύση είναι βέλτιστη για κάθε περίπτωση.

2.1.2 Cross-site Scripting

Το cross-site scripting, είναι ένα είδος ευάλωτου σημείου ασφαλείας στους υπολογιστές, που κατά κύριο λόγο συναντάται σε δικτυακές εφαρμογές (Ahoi., 2011). Επιτρέπει στους επιτιθέμενους, να παρεμβάλουν κώδικα από την μεριά του πελάτη, (client-side script) στις σελίδες που βλέπουν/επισκέπτονται διάφοροι χρήστες. Ένα ευάλωτο σημείο του cross-site scripting, μπορεί να χρησιμοποιηθεί από επιτιθέμενους, ώστε να παρακάμψουν περιορισμούς πρόσβασης.

Ο λόγος που γίνεται αναφορά στο cross-site scripting, είναι γιατί αποτέλεσε σημαντικό εμπόδιο κατά τη διαδικασία αποσφαλμάτωσης της εφαρμογής. Πιο συγκεκριμένα, επειδή τουλάχιστον ο προσομοιωτής (emulator) του λειτουργικού συστήματος android, δεν προσφέρει χρηστικό περιβάλλον, λεπτομερούς καταγραφής των διεργασιών που λαμβάνουν χώρα, ώστε να μπορούν να εντοπίζονται λάθη τόσο στην JavaScript, όσο και στην επικοινωνία με την υπηρεσία δικτύου (web service), θεωρήθηκε πως κάτι τέτοιο θα μπορούσε να γίνει με χρήση των περιηγητών ιστού (browsers) όπως Firefox, Chrome, Opera κλπ (ιδιαίτερα οι Chrome και Firefox, ενσωματώνουν εργαλεία ειδικά για προγραμματιστές, με αναλυτικές πληροφορίες σχετικά με όλες τις εκτελούμενες διεργασίες) αφού ο πηγαίος κώδικας είναι γραμμένος σε HTML5, JavaScript και CSS, γλώσσες δηλαδή που υποστηρίζουν κατ' εξοχήν οι περιηγητές.

Ωστόσο, δε φάνηκε εξ' αρχής μια πολύ σημαντική αδυναμία που είχαν οι περιηγητές, και δεν έδιναν πλήρη εικόνα των προβλημάτων που προέκυπταν ή έδιναν σε ορισμένες περιπτώσεις, εσφαλμένη εντύπωση αναφορικά με το πιθανολογούμενο πρόβλημα. Η «αδυναμία» αυτή, επρόκειτο ουσιαστικά για το γεγονός ότι οι περιηγητές, για λόγους ασφαλείας, και προστασίας του χρήστη, είναι ρυθμισμένοι να αποκλείουν το cross-site scripting. Αυτό είχε σαν αποτέλεσμα, πριν εντοπιστεί τελικά η συγκεκριμένη ιδιαιτερότητα, να λαμβάνονται διαδοχικά μηνύματα λάθους κατά την κλήση της υπηρεσίας του δικτύου (web service), με αποτέλεσμα, αρχικά να εξάγεται το συμπέρασμα, ότι είτε γινόταν εσφαλμένος χειρισμός των απαιτούμενων παραμέτρων κατά την χρησιμοποίηση των βιβλιοθηκών XML-RPC, είτε ότι οι ίδιες οι βιβλιοθήκες, δεν ήταν λειτουργικές.

Με λεπτομερέστερη αναζήτηση όμως των κωδικών σφαλμάτων, στις περιπτώσεις που ανέκυψαν, και διασταύρωση των στοιχείων αυτών από αρκετές διαφορετικές πηγές, είχε ως αποτέλεσμα να ανακαλυφθεί, ότι η εξομοίωση «έπεφτε» στη συγκεκριμένη δικλείδα ασφαλείας των περιηγητών.

Με τον εντοπισμό του θέματος περιορισμού των Http-Requests λόγω περιορισμών cross-site scripting από την πλευρά των περιηγητών, και με δεδομένο πως το εν λόγω θέμα δεν είχε αξιολογηθεί ως πιθανό πρόβλημα από την αρχή, γεννήθηκε άμεσα το ερώτημα, αν το ίδιο το Phoneygar, δεδομένου ότι βασίζεται στις ίδιες γλώσσες προγραμματισμού και στις ίδιες τεχνολογίες, θέτει και εκείνο περιορισμούς αναφορικά με το cross-site scripting, γεγονός, που έθετε υπό άμεση αμφισβήτηση, την αρχική ιδέα υλοποίησης της εφαρμογής. Ευτυχώς, η αναζήτηση, στην τεκμηρίωση του Phoneygar, είχε θετικά αποτελέσματα, καθώς έγινε σαφές, ότι δεν υπήρχε ανάλογος περιορισμός για την ανάπτυξη κώδικα και την δημιουργία εφαρμογών μέσω του Phoneygar.

Το σημαντικό θέμα όμως, της αποτελεσματικής αποσφαλμάτωσης, παρέμεινε, καθώς ο προσομοιωτής του λειτουργικού συστήματος Android, εξακολουθούσε να μην μπορεί να εξασφαλίσει αποτελεσματική αποσφαλμάτωση, ενώ όπως ήδη αναφέρθηκε, οι περιηγητές, υστερούσαν από τον εν γένει σχεδιασμό τους, και τους περιορισμούς που έθεταν για λόγους ασφαλείας.

Το γεγονός αυτό αποτέλεσε σημαντική πρόκληση για την συνέχιση της ανάπτυξης της εφαρμογής, και αυτό, γιατί φαινόταν πρακτικά αδύνατο η όποια συνέχισή της, καθώς χωρίς συγκεκριμένα μηνύματα λάθους, ώστε να εντοπίζονται τα προβληματικά σημεία στον κώδικα, η όλη διαδικασία ανάπτυξης θα ήταν πολύ πιο δύσκολο να προχωρήσει. Για το λόγο αυτό, αποφασίστηκε, να γίνεται σταδιακά ανάπτυξη του κώδικα και ανά πιο συχνά διαστήματα, από ότι αρχικά κρινόταν αναγκαίο, να γίνεται επίσκεψη στο Spitogatos.gr ώστε να δοκιμάζεται ο κώδικας, και κατά τη διάρκεια της κάθε επίσκεψης, να γίνεται και επιπλέον προσπάθεια ανάπτυξης του κώδικα, ώστε να προχωράει η ανάπτυξη της εφαρμογής.

Ευτυχώς, σχεδόν ταυτόχρονα, με την πρώτη επιτυχημένη προσπάθεια σύνδεσης, με την υπηρεσία δικτύου της Spitogatos.gr, βρέθηκε, και ένα μικρό παράθυρο αναφορικά με τους περιορισμούς στους περιηγητές για το cross-site scripting.

Προηγούμενη αναζήτηση, στο Google, για τον περιηγητή Firefox, είχε επιστρέψει διάφορα αποτελέσματα, ωστόσο αυτά δεν έδειξαν ότι κατάφερναν να λύσουν το πρόβλημα. Μετά από κάποιο διάστημα ωστόσο, και έπειτα από αντίστοιχη έρευνα για τον περιηγητή Chrome, φάνηκε, ότι το ίδιο θέμα (για διαφορετικούς προφανώς λόγους) είχε απασχολήσει και άλλους χρήστες, και προτεινόταν τρόποι, για προσωρινή απενεργοποίηση της ασφάλειας που ενσωματώνει ο Chrome (εκτέλεση από command prompt-cmd, του εκτελέσιμου αρχείου του Chrome, με τις παραμέτρους `-disable-web-security`), (McGinnis, 2011).

Το γεγονός αυτό αποτελούσε μία ακόμα ευχάριστη θετική εξέλιξη, μετά την πρώτη επιτυχημένη σύνδεση με την υπηρεσία του δικτύου της Spitogatos.gr καθώς, πλέον επέτρεπε, να προχωρήσει η ανάπτυξη της εφαρμογής με ακόμα πιο γρήγορο ρυθμό, ενώ ταυτόχρονα, επέτρεπε και τον πειραματισμό με επιπλέον προσθήκες κώδικα όπου υπήρχαν ενδείξεις ότι θα βελτίωναν την εφαρμογή.

2.1.3 Minification

Κατά την εκπόνηση της μεταπτυχιακής διατριβής, χρησιμοποιήθηκαν βιβλιοθήκες γραμμένες σε JavaScript, οι οποίες βρίσκονταν σε συνεπτυγμένη μορφή. Πρώτο και χαρακτηριστικότερο παράδειγμα αποτελούσε η βιβλιοθήκη «Mimic», η οποία ήταν διαθέσιμη μόνο σε συνεπτυγμένη (minified) μορφή. Στην συνέχεια, και εφόσον έγινε αντιληπτή η χρησιμότητα του «minification» έγινε η διαπίστωση, ότι και άλλες βιβλιοθήκες ήταν διαθέσιμες τόσο σε «αναπτυγμένη» όσο κι σε «συνεπτυγμένη» μορφή, όπως η JQuery.

Minification είναι η διαδικασία κατά την οποία αφαιρούνται όλοι οι περιττοί χαρακτήρες από τον πηγαίο κώδικα, χωρίς ωστόσο να επηρεάζεται η λειτουργικότητα (Anon., 2011). Ανάμεσα σε αυτούς τους περιττούς χαρακτήρες, συνήθως περιλαμβάνονται ο «κενός» χαρακτήρας, οι χαρακτήρες για αλλαγή γραμμής, και τα σχόλια, ενώ επίσης απαλείφονται και οι εσοχές ολόκληρων πεδίων κώδικα που συνήθως χρησιμοποιούνται ώστε να είναι ευανάγνωστος ο κώδικας, αλλά δεν απαιτούνται για την εκτέλεση. Οι συνεπτυγμένοι πηγαίοι κώδικες, είναι ιδιαίτερα χρήσιμοι για γλώσσες οι οποίες μεταφράζονται περαιτέρω από άλλα προγράμματα (interpreted languages), όπως είναι η JavaScript, διότι μειώνουν το σύνολο των δεδομένων που χρειάζεται να μεταφερθούν.

Για τη σύμπτυξη αλλά και την ανάπτυξη πηγαίου κώδικα, υπάρχουν ειδικά εργαλεία, διαθέσιμα και on-line, όπου μετατρέπουν για παράδειγμα τον συμπτυγμένο (minified) πηγαίο κώδικα, σε μορφή που μπορεί να διαβάσει και να κατανοήσει καλύτερα ο προγραμματιστής (beautify) (Lielmanis, 2011).

2.1.4 Χρήση «έξυπνων συσκευών» και παραδοσιακών περιηγητών (smartphones vs traditional browsers)

Οι έξυπνες συσκευές «smartphones» κερδίζουν όλο και περισσότερους χρήστες, ενώ σταδιακά καλύπτουν και ανάγκες που παραδοσιακά ο χρήστης θα χρειαζόταν έναν επιτραπέζιο ή ένα φορητό ηλεκτρονικό υπολογιστή.

Μέχρι πρόσφατα, εκτός από τους επιτραπέζιους ηλεκτρονικούς υπολογιστές (desktops) και τους φορητούς (laptops, notebooks), το σύνολο των ηλεκτρονικών υπολογιστών συμπληρώναν οι μικροί φορητοί υπολογιστές (netbooks) των οποίων ο κύριος προσανατολισμός ήταν η περιήγηση στο διαδίκτυο και κάποιες στοιχειώδεις εφαρμογές επεξεργασίας κειμένου. Σταδιακά, τουλάχιστον ως προς τους μικρούς φορητούς υπολογιστές, οι οποίοι έκαναν ένα δυναμικό ξεκίνημα στην αγορά αρχικά, το ενδιαφέρον των χρηστών άρχισε να στρέφεται προς τους ηλεκτρονικούς υπολογιστές τύπου tablet και τα smartphones, καθώς τα τελευταία άρχισαν να ενσωματώνουν ολοένα και περισσότερες δυνατότητες διατηρώντας και το χαρακτηριστικό τους γνώρισμα ως «τηλέφωνα» γεγονός που τα καθιστούσε αναπόσπαστο κομμάτι στην καθημερινότητα των περισσότερων ανθρώπων. Έτσι, πλέον, ο χρήστης δε χρειάζεται να έχει μαζί του (εφόσον δεν πρόκειται να κάνει κάποια πιο απαιτητική χρήση) και ένα netbook ή ένα laptop, καθώς μπορεί να κάνει την πλοήγηση στο διαδίκτυο κατευθείαν από το κινητό του τηλέφωνο, χρησιμοποιώντας εκδόσεις περιηγητών που ήδη του είναι οικεία από τους σταθερούς ή ηλεκτρονικούς φορητούς υπολογιστές που μπορεί να διαθέτει στο σπίτι ή στην εργασία του. Εξάλλου και το μέγεθος της οθόνης των «έξυπνων κινητών τηλεφώνων» κινείται πλέον σε επίπεδα, που να μπορεί να προσφέρει μια σχετικά άνετη πλοήγηση στον χρήστη.

Επιπλέον, πολλές εταιρείες, διαθέτουν, είτε εγγενείς εφαρμογές για συγκεκριμένες συσκευές (native applications) είτε δικτυακές εφαρμογές (web applications) με στόχο να κάνουν ακόμα καλύτερη την εμπειρία του χρήστη, παρέχοντάς του όλες τις δυνατότητες που η εκάστοτε εταιρεία ενσωματώνει μέσω της ιστοσελίδας της, μέσω μιας εφαρμογής και όχι μέσω του κλασσικού περιηγητή. Αυτό έχει σαν αποτέλεσμα, πολλές φορές οι χρήστες να προτιμούν ακόμα και όταν βρίσκονται σε χώρο που υπάρχει επιτραπέζιος ή φορητός ηλεκτρονικός υπολογιστής, να προτιμούν, αν πρόκειται για σύντομη περιήγηση, να χρησιμοποιήσουν το «έξυπνο» κινητό τους τηλέφωνο, από το να «καθίσουν» μπροστά από τον «κλασσικό» υπολογιστή. Αυτό δεν θα πρέπει να θεωρείται υπερβολή, αν αναλογιστεί κανείς το γεγονός, ότι πλέον ο χρήστης μπορεί ακόμα και να παίξει ένα ηλεκτρονικό παιχνίδι ή και να δει μια ταινία με τη χρήση του τηλεφώνου του.

2.1.5 Εγγενείς, Διαδικτυακές και Υβριδικές εφαρμογές

Ένα σημείο διαμάχης αναφορικά με την ανάπτυξη λογισμικού για τις έξυπνες συσκευές (smartphone) και κατ' επέκταση και για τους υπολογιστές τύπου tablet (καθώς πολλοί βασίζονται σε αντίστοιχες εκδόσεις λειτουργικών συστημάτων που υπάρχουν στις έξυπνες συσκευές), είναι η επιλογή στη δημιουργία και χρήση εγγενών εφαρμογών (native applications) ή διαδικτυακών εφαρμογών (web applications).

Με τον όρο εγγενή εφαρμογή (native application) εννοούμε μια εφαρμογή η οποία έχει αναπτυχθεί είτε για συγκεκριμένη έξυπνη συσκευή, είτε (πιο συχνά) για συγκεκριμένο λειτουργικό σύστημα το οποίο προορίζεται για μια ομάδα έξυπνων συσκευών. Η εγγενής εφαρμογή, περιέχει όλα τα απαραίτητα στοιχεία, προκειμένου να εγκατασταθεί στην έξυπνη συσκευή του χρήστη, και να είναι πλήρως λειτουργική για το σκοπό για τον οποίο προορίζεται. Πρόκειται για εφαρμογή η οποία έχει γραφεί σε μία συγκεκριμένη γλώσσα προγραμματισμού, η οποία είναι απόλυτα συμβατή με το εκάστοτε λειτουργικό όπως για παράδειγμα η αντικειμενοστραφής C για το λειτουργικό σύστημα IOS, και η Java για το λειτουργικό σύστημα Android.

Από την άλλη πλευρά υπάρχουν οι εφαρμογές διαδικτύου (web applications) οι οποίες πρόκειται για εφαρμογές όπου το περιεχόμενο είναι προσβάσιμο όταν υπάρχει σύνδεση στο διαδίκτυο και πρόκειται για εφαρμογές οι οποίες είναι συνήθως γραμμένες σε HTML με χρήση JavaScript και αναφέρονται κατά κανόνα στο σύνολο των λειτουργικών συστημάτων των διαφορετικών εταιρειών, μέσω συνήθως των περιηγητών που η κάθε συσκευή διαθέτει (Safari, Opera mini, Google περιηγητής)

Ωστόσο, η διάκριση αναφορικά με τον τρόπο ανάπτυξης των εφαρμογών και ειδικότερα με τις γλώσσες προγραμματισμού που χρησιμοποιούνται, έχει γίνει αρκετά πιο σύνθετη τον τελευταίο καιρό, καθώς πλέον αρχίζουν να εμφανίζονται περιπτώσεις που το μέλλον θα δείξει, αν θα αποτελέσουν υποκατηγορίες των δύο βασικών μέχρι στιγμής κατηγοριών που αναφέρθηκαν, ή θα αποτελέσουν οι ίδιες αυτόνομες κατηγορίες (Cavazza, 2011).

Αυτές οι περιπτώσεις θα μπορούσαν προς το παρόν να περιγραφούν ως «υβρίδια» ή «υβριδικές» εφαρμογές, καθώς φαίνεται να χρησιμοποιούν στοιχεία και των δύο κατηγοριών ως προς τον τρόπο ανάπτυξης του πηγαίου κώδικα, με αποτέλεσμα να είναι σχετικά δύσκολη η κατηγοριοποίησή τους.

Ένα χαρακτηριστικό παράδειγμα αποτελεί το Phonegap. Με το Phonegap, ο προγραμματιστής έχει τη δυνατότητα, να δημιουργήσει εγγενείς εφαρμογές (native applications) για τα συγκεκριμένα λειτουργικά συστήματα για τα οποία ενδιαφέρεται. Η ιδιαιτερότητα ωστόσο που κάνει το Phonegap, αλλά και άλλες πλατφόρμες που βασίζονται στην ίδια φιλοσοφία, να διαφέρουν, και να χαρακτηρίζονται οι εφαρμογές που δημιουργούνται με τη χρήση αυτών των εργαλείων, ως «υβριδικές» οφείλεται στον τρόπο ανάπτυξης των εφαρμογών που προκύπτουν από τη χρήση των συγκεκριμένων εργαλείων.

Πιο συγκεκριμένα, το Phonegap, «καταφέρνει» να παρέχει εγγενείς εφαρμογές χρησιμοποιώντας ένα χαρακτηριστικό που είναι κατ' εξοχήν γνώρισμα των διαδικτυακών εφαρμογών (web applications). Αντίθετα με τη δημιουργία των κατεξοχήν εγγενών εφαρμογών, όπου γίνεται χρήση συγκεκριμένων γλωσσών προγραμματισμού, η δημιουργία μιας εγγενούς εφαρμογής με το Phonegap, δεν απαιτεί τη χρήση συγκεκριμένης γλώσσας προγραμματισμού, «συμβατής» με το εκάστοτε λειτουργικό σύστημα, αλλά μόνο χρήση HTML, JavaScript και CSS, γλώσσες προγραμματισμού δηλαδή οι οποίες προσδιορίζουν κατά κύριο λόγο την ανάπτυξη διαδικτυακών εφαρμογών (web applications).

Άλλες πλατφόρμες που εμφανίζουν γνωρίσματα παρόμοια με του Phonegap, είναι οι

- Sencha: ένα πλαίσιο ανάπτυξης (framework), με το οποίο μπορεί ο χρήστης να δημιουργήσει εφαρμογές για έξυπνες συσκευές, χρησιμοποιώντας τη γλώσσα προγραμματισμού HTML5. Συνδυάζει τα χαρακτηριστικά των διαδικτυακών εφαρμογών (web applications) σε ένα εγγενές «πακέτο» (native) ώστε να μπορεί να εξασφαλίσει χρήση των συγκεκριμένων λειτουργιών της έξυπνης συσκευής (Sencha, 2012). Το πακέτο ανάπτυξης διατίθεται δωρεάν.
- Titanium: το εμπορικό πακέτο της εταιρείας Appcelerator (Appcelerator, 2012) το οποίο επιτρέπει στους προγραμματιστές να δημιουργήσουν εγγενείς εφαρμογές

χρησιμοποιώντας μια κοινή πλατφόρμα ανάπτυξης κώδικα. Η Appcelerator υποστηρίζεται από μεγάλες επιχειρήσεις όπως οι Storm Ventures, Sierra Ventures, TransLink Capital, Mayfield Fund, Red Hat and eBay. Το πακέτο ανάπτυξης διατίθεται δωρεάν, ωστόσο απαιτείται εγγραφή του χρήστη για να το «κατεβάσει».

- Rhomobile: μια εταιρεία που πλέον ανήκει στην «Motorola Solutions» και διαθέτει πακέτο ανάπτυξης για τους προγραμματιστές όπως το Rhodes 3.0. Επιτρέπει τη δημιουργία εγγενών εφαρμογών, με χρήση δεξιοτήτων που σχετίζονται με την ανάπτυξη διαδικτυακών εφαρμογών (Motorola Solutions, 2012). Το πακέτο ανάπτυξης διατίθεται ελεύθερα.
- ParticleCode: αποτελεί άλλη μια εταιρεία η οποία έχει εξαγοραστεί από την Appcelerator, και η οποία επιτρέπει στους προγραμματιστές να χρησιμοποιήσουν αντικειμενοστραφείς γλώσσες προγραμματισμού όπως η Java και η Actionscript3 για την δημιουργία εγγενών εφαρμογών (native applications) αλλά και διαδικτυακών εφαρμογών (web applications) (Appcelerator, 2012) το πακέτο ανάπτυξης είναι ελεύθερα διαθέσιμο στους χρήστες.
- Corona: πρόκειται για ένα εμπορικό πακέτο που διαθέτει η εταιρεία Anscamobile (Anscamobile, 2012) το οποίο επιτρέπει στους χρήστες να δημιουργήσουν εφαρμογές για διάφορα λειτουργικά συστήματα, χρησιμοποιώντας τη γλώσσα προγραμματισμού Lua. Το πακέτο, διατίθεται προς πώληση από την εταιρεία στους προγραμματιστές, σε τρεις εκδόσεις: την έκδοση για IOS, την έκδοση για Android, και την «πλήρη» έκδοση που υποστηρίζει IOS, Android, Kindle Fire και nook.
- Worklight: είναι μια εταιρεία της IBM, η οποία διαθέτει μια πλατφόρμα ανάπτυξης εγγενών εφαρμογών με τη χρήση της HTML5 (IBM, 2012). Η πλατφόρμα διατίθεται δωρεάν.
- WOPE (Write Once Publish Everywhere): είναι ένα εργαλείο της εταιρείας Backelite (Backelite, 2012) το οποίο την νέα έκδοση του πακέτου BKRender το οποίο με τη HTML5 επιτρέπει τη δημιουργία εγγενών εφαρμογών για τα περισσότερα λειτουργικά συστήματα. Η εταιρεία προσφέρει μια ολοκληρωμένη στρατηγική για τις επιχειρήσεις που θέλουν να δραστηριοποιηθούν με παρουσία στις έξυπνες συσκευές. Η εταιρεία διαθέτει δοκιμαστική έκδοση για τους προγραμματιστές.

	Πρόσβαση στη συσκευή	Ταχύτητα	Κόστος ανάπτυξης	Διάθεση σε κατάσταση εφαρμογών	Διαδικασία έγκρισης
Εγγενής Εφαρμογή	Πλήρης	Πολύ γρήγορη	Ακριβό	Διαθέσιμο	Υποχρεωτική
Υβριδική Εφαρμογή	Πλήρης	Εγγενής ταχύτητα, όση απαιτείται	Λογικό	Διαθέσιμο	Μικρή επιβάρυνση
Διαδικτυακή Εφαρμογή	Μερική	Γρήγορη	Λογικό	Δεν διατίθεται	Καμία

Εικόνα 1 - Χαρακτηριστικά των εφαρμογών

Στον εικόνα 1, παρουσιάζονται συνοπτικά μερικά στοιχεία συγκριτικά ανάμεσα στις καθαρά εγγενείς εφαρμογές, στις υβριδικές εφαρμογές αλλά και στις διαδικτυακές εφαρμογές (ο πίνακας έχει μεταφραστεί στα ελληνικά από τον αντίστοιχο που έχει παρουσιάσει η εταιρεία Worklight (Worklight, 2012).

Σύμφωνα με τα στοιχεία του πίνακα, παρουσιάζονται τα πλεονεκτήματα καθώς και τα σημεία που υστερεί η κάθε μια εφαρμογή, αναφορικά με τον προσανατολισμό της και τις τεχνολογίες που ενσωματώνει. Αν δηλαδή πρόκειται για καθαρά εγγενή εφαρμογή, για υβριδική εφαρμογή ή για διαδικτυακή εφαρμογή.

Οι εφαρμογές συγκρίνονται σε πέντε βασικές παραμέτρους, οι οποίες είναι αρκετά σημαντικές κατά την ανάπτυξη, και τη διάθεση μιας εφαρμογής και πιο συγκεκριμένα είναι:

- **Πρόσβαση στη συσκευή:** με τον όρο πρόσβαση στη συσκευή, αναφέρεται στη δυνατότητα μιας εφαρμογής, να έχει πρόσβαση σε συγκεκριμένες λειτουργίες που ενσωματώνει η έξυπνη συσκευή, όπως για παράδειγμα τα σύντομα μηνύματα, ο τηλεφωνικός κατάλογος, η φωτογραφική μηχανή, το άλμπουμ φωτογραφιών και άλλα. Τόσο οι εγγενείς εφαρμογές όσο και οι υβριδικές, έχουν πλήρη πρόσβαση σε αυτές τις λειτουργίες, ενώ οι διαδικτυακές εφαρμογές έχουν περιορισμένη πρόσβαση.
- **Ταχύτητα:** αναφέρεται στο πόσο γρήγορα μπορεί να υλοποιηθεί η εφαρμογή τα αιτήματα του χρήστη. Η εγγενής εφαρμογή έχει πολύ γρήγορη ταχύτητα, καθώς εκμεταλλεύεται πλήρως τις δυνατότητες της συσκευής, καθώς είναι γραμμένη σε γλώσσα απόλυτα συμβατή με το λειτουργικό σύστημα της έξυπνης συσκευής. Αρκετά ικανοποιητική ταχύτητα παρουσιάζουν και οι υβριδικές εφαρμογές, ενώ οι διαδικτυακές εφαρμογές, έχουν μια μικρή υστέρηση.
- **Κόστος ανάπτυξης.** Το γεγονός ότι μια καθαρά εγγενή εφαρμογή πρέπει να γραφεί σε συγκεκριμένη γλώσσα προγραμματισμού που προορίζεται για συγκεκριμένη έξυπνη συσκευή, έχει σαν συνέπεια το αρκετά υψηλό κόστος, ιδιαίτερα, αν χρειάζεται η ίδια η εφαρμογή να μεταφερθεί και σε έξυπνες συσκευές με διαφορετικά λειτουργικά συστήματα γεγονός που αυξάνει ακόμα περισσότερο το κόστος. Αντίθετα, η δυνατότητα της υβριδικής ανάπτυξης εφαρμογών, με κοινό πηγαίο κώδικα, όπου μπορεί να

παραχθεί το κατάλληλο εκτελέσιμο αρχείο για την πλειοψηφία των λειτουργικών συστημάτων, έχει ως αποτέλεσμα το κόστος να διατηρείται σε «λογικά» επίπεδα. Το ίδιο ισχύει και για τις διαδικτυακές εφαρμογές, καθώς δεν χρειάζεται ιδιαίτερη ανάπτυξη για το εκάστοτε λειτουργικό σύστημα.

- Διάθεση σε κατάσταση εφαρμογών: τόσο οι καθαρά εγγενείς εφαρμογές όσο και οι υβριδικές εφαρμογές μπορούν να διατεθούν από τα διαδικτυακά καταστήματα εφαρμογών, ώστε να τις «κατεβάσει» ο χρήστης και να τις εγκαταστήσει στην έξυπνη συσκευή του, ενώ αντίθετα, κάτι τέτοιο δεν προσφέρεται για τις διαδικτυακές εφαρμογές, καθώς αυτές είναι προσβάσιμες μόνο όταν ο χρήστης είναι συνδεδεμένος στο διαδίκτυο.
- Διαδικασία έγκρισης: αναφέρεται στην διαδικασία, προκειμένου να γίνει μια εφαρμογή διαθέσιμη μέσω των διαδικτυακών ηλεκτρονικών καταστημάτων. Για τις εγγενείς εφαρμογές, είναι μια υποχρεωτική διαδικασία, η οποία «ταλαιπωρεί» τους προγραμματιστές, ενώ στις υβριδικές εφαρμογές, η επιβάρυνση είναι πολύ μικρή. Αντίθετα, δεν υπάρχει καμία επιβάρυνση στις διαδικτυακές εφαρμογές καθώς εξ'ορισμού δεν διατίθενται από τα διαδικτυακά ηλεκτρονικά καταστήματα.

Στα πλαίσια της μεταπτυχιακής διατριβής, όπου έγινε ανάπτυξη εφαρμογής με χρήση της πλατφόρμας Phonegap (υβριδική εφαρμογή), μόνο τρεις από τις συνολικά πέντε παραμέτρους, αποτέλεσαν σημείο αναφοράς. Πιο συγκεκριμένα, κατέστη πράγματι δυνατή η πρόσβαση στις εγγενείς λειτουργίες της έξυπνης συσκευής, ενώ ο χρόνος εκτέλεσης της εφαρμογής ήταν αρκετά ικανοποιητικός. Επιπλέον, εκτός από την αρχική ανάγκη εξοικείωσης με την πλατφόρμα, γεγονός το οποίο συναντάται γενικότερα σε κάθε νέα τεχνολογία που εισάγεται, μπορεί πολύ εύκολα ο προγραμματιστής, μετά την αρχική υλοποίηση, να δημιουργήσει τις υπόλοιπες υλοποιήσεις που να αναφέρονται σε διαφορετικά λειτουργικά συστήματα, γεγονός, που μειώνει σημαντικά το κόστος. Η διαδικασία έγκρισης, και διάθεσης μέσω ηλεκτρονικών καταστημάτων της εφαρμογής δεν ενέπιπτε με το στόχο και το σκοπό της υλοποίησης, με αποτέλεσμα, να μην μελετηθούν αυτές οι παράμετροι.

2.2 Αναλυτική περιγραφή των τεχνολογιών που χρησιμοποιήθηκαν

2.2.1 Phonegap

Η μεγάλη ανάπτυξη στον τομέα των κινητών τηλεφώνων και συγκεκριμένα των έξυπνων τηλεφώνων (smartphones) που κινείται με ιλιγγιώδεις ταχύτητες τα τελευταία χρόνια, δημιούργησε μια τεράστια ζήτηση για ενσωμάτωση νέων χαρακτηριστικών και νέων δυνατοτήτων στα κινητά τηλέφωνα. Αυτό είχε σαν αποτέλεσμα πολλές εταιρείες να αρχίσουν να αναπτύσσουν εφαρμογές για κινητά τηλέφωνα, προκειμένου να διεκδικήσουν μερίδιο στη νέα αυτή αγορά για να κερδίσουν νέους πελάτες, αλλά και να αυξήσουν την ικανοποίηση ενδεχομένως των πελατών που ήδη είχαν, από την πιο «παραδοσιακή» μέχρι τώρα παρουσία τους.

Τα έξυπνα κινητά τηλέφωνα, όπως και οι υπολογιστές, έχουν ένα λειτουργικό σύστημα, προκειμένου να λειτουργήσουν, και πάνω στο οποίο αναπτύσσονται και εγκαθίστανται οι εκάστοτε εφαρμογές. Σε αντίθεση όμως με τους υπολογιστές, όπου το μεγαλύτερο μέρος των χρηστών χρησιμοποιούν ηλεκτρονικού υπολογιστές με το λειτουργικό σύστημα Windows, και ακολουθούν σε πιο περιορισμένη έκταση η Apple με το MacOS και το Linux, στα κινητά ο ανταγωνισμός προς το παρόν είναι πολύ πιο έντονος, όπου πέρα από τις αρκετές διαφορετικές εταιρείες κατασκευής τηλεφώνων, συναντάμε και ποικίλα λειτουργικά συστήματα, όπως το Android της Google, το iOS της Apple, το BlackBerry, το Windows Mobile της Microsoft, το WebOS της Hewlett Packard, και το Symbian.

Η ύπαρξη τόσων πολλών διαφορετικών λειτουργικών συστημάτων, όπου το κάθε ένα έχει συγκεκριμένες απαιτήσεις αναφορικά με τις εφαρμογές που θα γραφούν για να λειτουργήσουν πάνω σε αυτό, αλλά και την απαίτηση για ειδικά εργαλεία ανάπτυξης, έχει σαν αποτέλεσμα, πολλές φορές να χρειάζεται η ανάπτυξη της ίδιας εγγενούς εφαρμογής (native application) να γίνεται από την αρχή, καθώς υπάρχει ένα χάσμα, ανάμεσα στα λειτουργικά αυτά συστήματα και στις απαιτήσεις τους.

Το χάσμα αυτό που υπάρχει στα λειτουργικά συστήματα που χρησιμοποιούνται στα κινητά τηλέφωνα έρχεται να καλύψει το PhoneGap. Το Phonegap, είναι μια «πλατφόρμα» ανοιχτού κώδικα, όπου ενσωματώνει ανοιχτά πρότυπα και τεχνολογίες διαδικτύου προκειμένου να γεφυρώσει το κενό που υπάρχει ανάμεσα στα λειτουργικά συστήματα των διαφόρων εταιρειών. Το Phonegap, δημιουργήθηκε από την Nitobi, ενώ τον Οκτώβριο του 2011 η Adobe, (αντιλαμβανόμενη την δυναμική που αποκτά το Phonegap – πάνω από 600.000 λήψεις του πακέτου που μετρά λίγους μόλις μήνες από την πρώτη σταθερή έκδοση που διαμοιράστηκε) (Nitobi, 2011) απέκτησε τη Nitobi, προκειμένου η τελευταία να ασχοληθεί εξολοκλήρου με την περεταίρω ανάπτυξη του PhoneGap.

Ο προγραμματιστής που χρησιμοποιεί το Phonegap, χρειάζεται μόνο να γράψει τον κώδικά του, με τη χρήση HTML5, CSS και JavaScript. Στη συνέχεια, από το πακέτο ανοιχτού κώδικα του Phonegap, χρησιμοποιεί το κατάλληλο αρχείο που προορίζεται για το συγκεκριμένο λειτουργικό σύστημα το οποίο στοχεύει ο προγραμματιστής, και με τα ειδικά εργαλεία ανάπτυξης, που διαθέτουν οι εταιρείες των αντίστοιχων λειτουργικών συστημάτων γίνεται η δημιουργία (compile) της εκτελέσιμης εφαρμογής. Προκειμένου να δημιουργηθεί ένα εκτελέσιμο αρχείο για λειτουργικό σύστημα άλλης εταιρείας, ο προγραμματιστής το μόνο που έχει να κάνει, είναι να ενσωματώσει το αντίστοιχο αρχείο από το πακέτο του Phonegap, και με το αντίστοιχο επίσης εργαλείο ανάπτυξης, να δημιουργήσει (compile) την εφαρμογή, χωρίς την ανάγκη να ξαναγράψει από την αρχή το πρόγραμμα, εφόσον ο πηγαίος κώδικας (HTML5, CSS, JavaScript) παραμένει ίδιος.

Υπάρχουν ωστόσο σε ορισμένες συσκευές κινητών τηλεφώνων, κάποιες παραπάνω δυνατότητες, ή κάποιιο περιορισμοί τους οποίους ενδεχομένως να χρειαστεί να λάβει υπόψη του ο προγραμματιστής, κατά τη μετάβαση από το ένα λειτουργικό σύστημα στο άλλο.

Η εφαρμογή που αναπτύσσεται και τεκμηριώνεται στα πλαίσια της συγκεκριμένης μεταπτυχιακής διατριβής, βασίστηκε στην πλατφόρμα Phonegap. Ωστόσο, επειδή δεν ήταν δυνατή η παραγωγή εφαρμογών για όλα τα διαθέσιμα λειτουργικά συστήματα, καθώς αυτό θα προϋπέθετε πολύ μεγάλη δαπάνη για την απόκτηση αντιπροσωπευτικών συσκευών για την κάθε κατηγορία λειτουργικού συστήματος, έγινε εστίαση στην παραγωγή εγγενούς εφαρμογής (native) για το λειτουργικό σύστημα Android και πιο συγκεκριμένα για τις εκδόσεις 2.2 (froyo) και 2.3 (gingerbread).

Ωστόσο, επειδή αντικείμενο της παρούσας μεταπτυχιακής διατριβής δεν ήταν η χρήση του Phonegap, να θεωρηθεί αυτοσκοπός, χωρίς να εξυπηρετεί την ουσία ύπαρξής του (την δυνατότητα ο πηγαίος κώδικας να εφαρμόζεται και στα υπόλοιπα-πλην του Android-λειτουργικά συστήματα), η ανάπτυξη του πηγαίου κώδικα, καθώς και η ενσωμάτωση εξειδικευμένων λειτουργιών των τηλεφώνων όπως η χρήση της κάμερας και ο προσδιορισμός του γεωγραφικού στίγματος του τηλεφώνου, έγινε με προσήλωση στη δυνατότητα αυτούσιας χρησιμοποίησής του (του πηγαίου κώδικα) σε οποιοδήποτε άλλο υποστηριζόμενο (από το Phonegap) λειτουργικό σύστημα χωρίς την ανάγκη οποιασδήποτε τροποποίησης.

Στην περίπτωση του Android, το Phonegap, διαθέτει στο πακέτο που διαμοιράζεται δύο αρχεία, τα phonegap.js και phonegap.jar, ενώ για την ανάπτυξη της εφαρμογής, συνιστώνται η χρήση του προγράμματος Eclipse και των εργαλείων ανάπτυξης Android SDK (Software Development/Developer's Kit) και ADT plugin (Android Development Tools) ενώ ενδείκνυται και η χρήση του AVD manager (Android Virtual Devices) όπου μπορεί να γίνει προσομοίωση του λειτουργικού συστήματος Android σε μια εικονική συσκευή στον ηλεκτρονικό υπολογιστή.

2.2.2 JavaScript libraries

Οι JavaScript libraries, είναι βιβλιοθήκες όπου περιέχουν επαναχρησιμοποιήσιμα σενάρια εντολών (JavaScript) και βοηθούν στο να καταστεί ο προγραμματισμός πιο εύκολος, πιο περιεκτικός και καλύτερα δομημένος, αφού το μεγαλύτερο μέρος του κώδικα είναι συγκεντρωμένο σε ξεχωριστό αρχείο, την βιβλιοθήκη, από την οποία αντλεί ο προγραμματιστής τις πληροφορίες που χρειάζεται.

Για την υλοποίηση της εφαρμογής έγινε χρήση τόσο έτοιμων βιβλιοθηκών, όσο και βιβλιοθηκών που δημιουργήθηκαν ειδικά για τους σκοπούς της συγκεκριμένης εφαρμογής.

Οι ανοιχτού κώδικα (open source) βιβλιοθήκες που χρησιμοποιήθηκαν είναι οι εξής:

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

- phonedgap.js
- jquery.js
- jquery.blockUI.js
- mimic.js

Ενώ οι βιβλιοθήκες που δημιουργήθηκαν ειδικά για την εφαρμογή (με τα ονόματα που συναντώνται στο αρχείο .html) είναι οι εξής:

- application-js.js
- list.js

Phonedgap.js

Η πλατφόρμα phonedgap, (Nitobi, 2011) διαθέτει διάφορα αρχεία, προκειμένου να γίνει επιτυχής δημιουργία του εκτελέσιμου αρχείου στο συγκεκριμένο λειτουργικό σύστημα που είναι επιθυμητό. Στην προκειμένη περίπτωση για την υλοποίηση εφαρμογής για το λειτουργικό σύστημα Android, είναι απαραίτητη η χρήση της βιβλιοθήκης Phonedgap.js η οποία περιέχει συναρτήσεις οι οποίες «γεφυρώνουν» το χάσμα, ανάμεσα στον «ενιαίο» κώδικα που γράφει ο προγραμματιστής, και στη προκειμένη περίπτωση στις εντολές που «αντιλαμβάνεται» ως ορθές το λειτουργικό σύστημα Android.

Ο προγραμματιστής, γράφει τον κώδικα, με βάση τις οδηγίες που παρέχονται από το εγχειρίδιο του Phonedgap, χρησιμοποιώντας συναρτήσεις που έχει ορίσει το Phonedgap, και με τη χρήση της βιβλιοθήκης, οι συναρτήσεις αυτές αντιστοιχίζονται με τις συγκεκριμένες εντολές που δέχεται το λειτουργικό σύστημα Android.

Εκτός από την εν γένει ανάγκη ενσωμάτωσης της βιβλιοθήκης Phonedgap.js, η χρήση της μπορεί να φανεί και στη δυνατότητα που δίνεται στο χρήστη μέσα από την εφαρμογή να αποκτήσει πρόσβαση στην κάμερα του κινητού, αλλά και να πάρει πληροφορίες για το γεωγραφικό στίγμα του, τα οποία αποτελούν αμιγή χαρακτηριστικά και λειτουργίες του κινητού τηλεφώνου.

JQuery.js

Η βιβλιοθήκη JQuery.js (jQuery Foundation, 2011) αποτελεί ένα πολύ χρήσιμο εργαλείο για τον προγραμματιστή, αφού του επιτρέπει, να απλοποιήσει το κώδικα που χρειάζεται να γράφει κατά την δημιουργία ενός JavaScript αρχείου, και να ενσωματώσει στον κώδικά του πολλά χαρακτηριστικά εύκολα και γρήγορα.

Περιέχει συναρτήσεις για πάρα πολλές παραμετροποιήσεις που ενδεχομένως να χρειαστεί ο προγραμματιστής να ενσωματώσει, τις οποίες μπορεί να τις ενσωματώσει, «καλώντας» απλά την κατάλληλη συνάρτηση, χωρίς να χρειάζεται να γράφει τη δική του για οποιαδήποτε λειτουργικότητα επιθυμεί να ενσωματώσει.

Ειδικά, η JQuery απλοποιεί τον χειρισμό αρκετών γεγονότων, τη χρήση κινούμενων λειτουργιών (animation) ενώ είναι και πολύ χρήσιμη για την υλοποίηση αλληλεπιδράσεων με τεχνικές AJAX (Asynchronous JavaScript and XML).

Στα πλαίσια της εφαρμογής, η JQuery, χρησιμοποιήθηκε για δύο βασικούς λόγους, αφενός για να υλοποιηθούν ορισμένες επιθυμητές λειτουργίες και αφετέρου προκειμένου να αποτελέσει ένα εύφορο έδαφος για την περαιτέρω ανάπτυξη της συγκεκριμένης εφαρμογής στο μέλλον.

Αναφορικά με τους λειτουργικούς λόγους για τους οποίους χρησιμοποιήθηκε η JQuery, είναι κυρίως ο χειρισμός συγκεκριμένων συμβάντων. Πιο συγκεκριμένα, προκειμένου να οριστούν ορισμένες ενέργειες κατά την επιλογή συγκεκριμένων κουμπιών, όπου κρίθηκε αναγκαίο, (έλεγχος απαιτούμενων προϋποθέσεων κατά την επιλογή ενός κουμπιού), κρίθηκε πως θα μπορούσε να γίνει ο χειρισμός των γεγονότων με χρήση της JQuery (Castledine & Sharkie, 2010). Επίσης, κατά τη χρήση της εφαρμογής ο χρήστης διαπιστώνει πως ορισμένα τμήματα στην επιφάνεια διεπαφής μπορεί να εμφανιστούν ή να αποκρυφτούν. Η συγκεκριμένη λειτουργικότητα υλοποιείται επίσης με τη χρήση JQuery όπου επιτυγχάνεται η εμφάνιση και απόκρυψη τμημάτων του κώδικα HTML ανάλογα με τις επιλεγμένες ενέργειες.

Επίσης, κατά την ενσωμάτωση του πρόσθετου (plugin) JQuery.blockUI.js χρησιμοποιήθηκε και η συνάρτηση «setTimeout» καθώς κρίθηκε απαραίτητη για την ορθή ενσωμάτωση του πρόσθετου.

JQuery.blockUI.js

Το JQuery.blockUI.js (Alsup, 2009) είναι μια πρόσθετη (plugin) βιβλιοθήκη η οποία προσθέτει επιπλέον λειτουργικότητα βασισμένη ωστόσο στον ήδη υπάρχοντα κώδικα που είναι αναπτυγμένος στη JQuery. Το γεγονός ότι χαρακτηρίζεται σαν «πρόσθετη» (plugin) οφείλεται στο γεγονός, ότι απαραίτητη προϋπόθεση για να είναι λειτουργική, είναι η ύπαρξη της βιβλιοθήκης JQuery.

Η συγκεκριμένη βιβλιοθήκη, επιτρέπει στον προγραμματιστή, να απενεργοποιήσει την επιφάνεια διεπαφής του χρήστη, όταν το κρίνει απαραίτητο. Συνήθως κάτι τέτοιο επιδιώκεται κατά τη διάρκεια AJAX κλήσεων.

Για τον ίδιο ακριβώς λόγο, έγινε η ενσωμάτωσή της και στην εφαρμογή. Κατά τη διάρκεια της επικοινωνίας της εφαρμογής με την υπηρεσία δικτύου της Spitogatos.gr και ανάλογα με την ποιότητα της σύνδεσης που υπάρχει, χρειάζεται ένα χρονικό διάστημα προκειμένου να γίνει η επικοινωνία με την υπηρεσία, να μεταφερθούν τα δεδομένα και να επιστρέψει η απάντηση επιβεβαίωσης. Για το συγκεκριμένο διάστημα, υπάρχει περίπτωση, ο χρήστης, να υποθέσει ότι κάτι δεν έχει γίνει σωστά από την πλευρά του, και να πατήσει το κουμπί καταχώρησης περισσότερες φορές πιστεύοντας ενδεχομένως ότι το αίτημά του δεν καταχωρήθηκε.

Με τη συγκεκριμένη βιβλιοθήκη, κατά την επιλογή της καταχώρησης των δεδομένων για την αγγελία (και με δεδομένο ότι όλα τα πεδία έχουν καταχωρηθεί σωστά), απενεργοποιείται η επιφάνεια εργασίας του χρήστη, με την ταυτόχρονη εμφάνιση ενός μηνύματος ότι το αίτημά του είναι σε εξέλιξη, ενώ όταν επιστρέψει η απάντηση από την υπηρεσία δικτύου (webservice) η επιφάνεια εργασίας, ενεργοποιείται και πάλι, προκειμένου να συνεχίσει την εργασία του ο χρήστης.

Για την επιτυχή χρήση του plugin, χρειάστηκε να χρησιμοποιηθεί μια καθυστέρηση ενός δευτερολέπτου (timeout) στην ακολουθούμενη συνάρτηση κατά την εκτέλεση της οποίας (συνάρτησης) ήταν επιθυμητή η απενεργοποίηση του περιβάλλοντος διεπαφής, καθώς χωρίς την «τεχνητή» καθυστέρηση, δεν γινόταν σωστή εκτέλεση του πρόσθετου.

Mimic.js

Η mimic.js (Goncalves, 2009) είναι μια βιβλιοθήκη XML-RPC από την πλευρά του πελάτη (client side), ανοιχτού κώδικα, που υλοποιείται με χρήση JavaScript. Χαρακτηρίζεται ως «πελάτη» για το λόγο ότι μπορεί μόνο να παράγει αιτήματα και να αναλύσει τις αντίστοιχες απαντήσεις, και δεν μπορεί να χρησιμοποιηθεί στο να αναλύσει ερωτήματα και να παράγει απαντήσεις όπως απαιτείται σε μια εφαρμογή που θα χαρακτηριζόταν ως «από την πλευρά του εξυπηρετητή» (server).

Προορίζεται για να ενσωματώνεται μέσα σε ιστοσελίδες, και να επιτρέπει σε αυτές να έχουν πρόσβαση σε υπηρεσίες δικτύου, βασισμένες στο πρωτόκολλο XML-RPC μέσω της διεπαφής W3C XMLHttpRequest. Η τεκμηρίωσή της είναι στοιχειώδης και αποτελείται ουσιαστικά από ένα στοιχειώδες παράδειγμα λειτουργίας της.

Το βασικό αρχείο της mimic, (mimic.js) διατίθεται συνεπτυγμένο (minified) σε σχεδόν 4 γραμμές με αποτέλεσμα να είναι πολύ δύσκολος ο χειρισμός και η μελέτη της, τουλάχιστον μέχρι να επιτευχθεί η ανάπτυξη του κώδικα με χρήση κατάλληλων online εργαλείων. Το γεγονός αυτό σε συνδυασμό και με την ελλιπέστατη τεκμηρίωση του κώδικα έκανε αρκετά πιο επίπονη τη μελέτη και τη χρήση της συγκεκριμένης βιβλιοθήκης κατά τα πρώτα στάδια.

Η mimic, ήταν γνωστό εξ' αρχής, ότι υστερεί στο κομμάτι, της επιβεβαίωσης των στοιχείων, δεν υποστηρίζει δηλαδή μέσα στον κώδικά της λειτουργία επιβεβαίωσης όνομα χρήστη και κωδικού πρόσβασης (Http-Authentication). Αυτό είχε σαν αποτέλεσμα, να «παραμεριστεί» για ένα διάστημα, καθώς ήταν τεχνικά αδύνατο, να δοκιμαστεί στην Spitogatos.gr και συγκεκριμένα στην υπηρεσία για την ανάπτυξη των υπηρεσιών, καθώς απαραίτητη προϋπόθεση ήταν το να μπορεί να γίνει επαλήθευση στοιχείων προκειμένου, να γίνει στη συνέχεια σύνδεση με την

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

υπηρεσία δικτύου για να καταχωρηθούν τα δεδομένα. Από την άλλη πλευρά ωστόσο το εμπορικό κομμάτι της Spitogatos.gr, που είναι και ο βασικός στόχος, δεν παρουσιάζει αυτόν τον περιορισμό, για τον επιπρόσθετο λόγο, ότι κατά την ανάλυση των στοιχείων που ενσωματώνονται στον XML κώδικα που αποστέλλεται, υπάρχουν και στοιχεία, επιβεβαίωσης ότι ο αποστολέας, είναι πιστοποιημένος χρήστης της υπηρεσίας, και κατέχει το κατάλληλο κωδικό, όνομα χρήστη και κλειδί για την εφαρμογή.

Για το λόγο αυτό, δεν απορρίφθηκε, η *mimic*, καθώς όλες οι βιβλιοθήκες ήταν δυνητικά πιθανές για χρήση, μέχρι να απορριφθούν για συγκεκριμένους λόγους, γεγονός που επιβεβαιώθηκε μετά από κάποιο διάστημα, καθώς άλλες βιβλιοθήκες που στις οποίες είχε δοθεί προτεραιότητα στην αρχή φάνηκαν τελικά να μην ανταποκρίνονται στις απαιτήσεις. Αντίθετα, η *mimic*, με εξαίρεση το κομμάτι της επιβεβαίωσης των στοιχείων (HttpAuthentication) κατάφερε να επικοινωνήσει επιτυχώς με την υπηρεσία δικτύου της Spitogatos.gr.

Προκειμένου να παραμεριστεί το εμπόδιο της ανάγκης για επιβεβαίωση των στοιχείων, η ομάδα της Spitogatos.gr απενεργοποίησε την ανάγκη για HttpAuthentication κατά την επικοινωνία με την υπηρεσία δικτύου στο περιβάλλον ανάπτυξης (*test environment*), προκειμένου να γίνουν όλοι οι απαραίτητοι έλεγχοι σχετικά με τις δυνατότητες της βιβλιοθήκης *mimic*.

Για την επιτυχή αποστολή των απαιτούμενων δεδομένων με χρήση του πρωτοκόλλου XML-RPC, είναι απαραίτητη η κλήση της αντίστοιχης συνάρτησης από τη *mimic.js*.

Με την κλήση της συνάρτησης της *mimic*, γίνεται και το «πέρασμα» των απαραίτητων δεδομένων που χρειάζονται, σε συμβατή μορφή προκειμένου να τα χειριστεί η *mimic*. Καλείται η αρχική συνάρτηση, στην οποία περνιούνται δύο βασικές παράμετροι, οι οποίες είναι ο υπερσύνδεσμος της υπηρεσίας διαδικτύου, και η συγκεκριμένη μέθοδος, σύμφωνα με την οποία θα αποσταλούν τα δεδομένα και στη συνέχεια, προστίθενται οι παράμετροι, που είναι απαραίτητες προκειμένου τα αποστελλόμενα δεδομένα να έχουν συμβατή μορφή με αυτήν που περιμένει να «αναγνωρίσει» η υπηρεσία διαδικτύου.

Για την καταχώρηση νέας αγγελίας, η Spitogatos.gr χρησιμοποιεί μία μέθοδο που την ονομάζει «*sync.newListing*». Για την επιτυχή κλήση της μεθόδου, προστίθεται μέσω της *mimic*, οι ακόλουθες απαραίτητες παράμετροι:

- το κλειδί της εφαρμογής
- το id του χρήστη
- το όνομα χρήστη
- ο κωδικός χρήστη
- και ένα αντικείμενο JavaScript το οποίο περιέχει τη δομή όλων των καταχωρούμενων δεδομένων σύμφωνα με την μέθοδο «*sync.newListing*».

Αφού γίνει η «κατασκευή» των δεδομένων που πρέπει να αποσταλούν, αποστέλλονται στη Spitogatos.gr, και καταχωρούνται στη βάση δεδομένων. Εφόσον η καταχώρηση είναι επιτυχής, επιστρέφει μια απάντηση όπου ενημερώνεται ο χρήστης για τον αριθμό της αγγελίας που καταχώρησε.

Για την καταχώρηση νέας φωτογραφίας σε ήδη υπάρχουσα αγγελία, η Spitogatos.gr χρησιμοποιεί μία μέθοδο που την ονομάζει «*sync.addImage*». Για την επιτυχή κλήση της μεθόδου, προστίθεται μέσω της *mimic*, οι ακόλουθες απαραίτητες παράμετροι:

- το κλειδί της εφαρμογής
- το όνομα χρήστη
- ο κωδικός χρήστη
- ο αριθμός της αγγελίας
- η θέση στην οποία θα καταχωρηθεί η φωτογραφία
- τα δεδομένα της φωτογραφίας σε μορφή *base64*

Και σε αυτή την περίπτωση, αφού γίνει η «κατασκευή» των δεδομένων που πρέπει να αποσταλούν, αποστέλλονται στην Spitogatos.gr, και καταχωρούνται στη βάση δεδομένων. Εφόσον η καταχώρηση είναι επιτυχής, επιστρέφει μια απάντηση όπου ενημερώνεται ο χρήστης για τον αριθμό της θέσης στην οποία καταχωρήθηκε η φωτογραφία.

Application-js.js

Η βιβλιοθήκη application-js.js περιέχει το σύνολο του κώδικα JavaScript που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής αλλά και για την κλήση άλλων βιβλιοθηκών JavaScript.

Στην βιβλιοθήκη application-js.js υπάρχουν:

- οι απαιτούμενες συναρτήσεις προκειμένου να ανακτηθεί το γεωγραφικό στίγμα που βρίσκεται ο χρήστης, με κλήση της βιβλιοθήκης rhonegar.js και η μεταφορά του και εμφάνισή του σε προεπιλεγμένο σημείο της επιφάνειας διεπαφής του χρήστη.
- οι απαιτούμενες συναρτήσεις προκειμένου να υπάρχει πρόσβαση τόσο στη φωτογραφική μηχανή του κινητού τηλεφώνου, όσο και στο φάκελο με τις αποθηκευμένες φωτογραφίες, με χρήση στη συνέχεια της βιβλιοθήκης rhonegar.js προκειμένου, να μπορεί να τραβάει ή να επιλέγει φωτογραφία ο χρήστης και να αποστέλεται στην υπηρεσία δικτύου της Spitogatos.gr. (κατά την ανάκτηση της φωτογραφίας, γίνεται αποθήκευση των δεδομένων της σε μία μεταβλητή, με μορφή base64 προκειμένου να αποσταλεί (στη συμβατή μορφή) στην υπηρεσία δικτύου
- το σύνολο των στοιχείων που απαιτούνται προκειμένου να γίνει ορθή κατασκευή της συμβατής μορφής που χρειάζεται για να αποσταλούν τα δεδομένα ανάλογα με την καλούμενη μέθοδο. Στην καταχώρηση νέας αγγελίας, περνιέται στην mimic, ένα και μόνο αντικείμενο με το σύνολο των πληροφοριών. Το συγκεκριμένο αντικείμενο της JavaScript, είναι ένα «αντικείμενο, αντικειμένων» πρόκειται δηλαδή για ένα αντικείμενο το οποίο αποτελείται από πέντε εμφωλευμένα αντικείμενα. Το κάθε ένα από τα εμφωλευμένα αντικείμενα, περιέχει ένα σύνολο στοιχείων (σύμφωνα με τις προδιαγραφές που δόθηκαν από την Spitogatos.gr) τα οποία στοιχεία, περιέχουν τα δεδομένα που εισάγει ο χρήστης και επιθυμεί να καταχωρήσει στην υπηρεσία δικτύου της Spitogatos.gr.
- οι συναρτήσεις όπου ορίζουν τις ενέργειες που πρέπει να γίνουν κατά την επιλογή συγκεκριμένων κουμπιών (έλεγχος αν έχουν συμπληρωθεί τα απαιτούμενα πεδία, εμφάνιση και απόκρυψη τμημάτων του περιβάλλοντος διεπαφής, καθώς και κλήση των επιθυμητών συναρτήσεων)

List.js

Η εφαρμογή έχει, για την καταχώρηση των δεδομένων, εκτός από τα πεδία όπου εισάγει ο χρήστης δεδομένα, και ορισμένα μενού προκαθορισμένων επιλογών. Επειδή όμως ορισμένα μενού επιλογών περιέχουν πάρα πολλά δεδομένα, κρίθηκε απαραίτητο να υπάρχει μια αλληλεξάρτηση ορισμένων επιλογών σε κάποια μενού, με τις διαθέσιμες επιλογές στα εξαρτώμενα μενού, προκειμένου να είναι η εφαρμογή πιο φιλική προς το χρήστη. Ανάλογα δηλαδή με την επιλογή που γίνεται στα «γονικά» μενού, εμφανίζονται και οι αντίστοιχες επιλογές στα θυγατρικά μενού. Από τα μενού προκαθορισμένων επιλογών, συνολικά τέσσερα παρουσιάζουν μία εξάρτηση μεταξύ τους ανά δύο. Συγκεκριμένα το μενού επιλογών για την «κατηγορία» του ακινήτου, επιρραάζει το μενού «είδος» ακινήτου, ενώ αντίστοιχα το μενού επιλογών για «περιοχή» επιρραάζει το αντίστοιχο μενού επιλογών «υποπεριοχή».

Η βιβλιοθήκη list.js, περιέχει το σύνολο των δεδομένων που περιλαμβάνονται στα συγκεκριμένα μενού επιλογών, φροντίζοντας, ανάλογα με την επιλογή που κάνει ο χρήστης να εμφανίζει τα αντίστοιχα δεδομένα.

Επίσης, τόσο ο όγκος των δεδομένων, όσο και η φύση τους, καθιστά καλύτερη επιλογή τη συγκέντρωσή τους σε εξωτερικό αρχείο, προκειμένου να είναι συγκεντρωμένα τα δεδομένα και να μπορεί να γίνει πιο εύκολα η όποια προσθήκη ή αφαίρεση αλλά και η γενικότερη διαχείρισή τους. Το σύνολο των δεδομένων που εισήχθησαν στο αρχείο list.js προέρχεται από την τεκμηρίωση που παρείχε η επιχείρηση σχετικά με τις χρησιμοποιούμενες περιοχές και τον τρόπο διάρθρωσής τους (Spitogatos.gr, 2011).

2.2.3 HTML5

Η HTML (Hypertext Markup Language) (World Wide Web Consortium (W3C), 2011) είναι μια γλώσσα «σήμανσης» (markup), η οποία χρησιμοποιείται ώστε να δώσει μια «δομή» σε έγγραφα Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

κειμένου και πολυμέσων, και για να δώσει υπερασυνδέσμους μεταξύ διαφορετικών εγγράφων που χρησιμοποιούνται εκτενώς στον παγκόσμιο ιστό (World Wide Web).

Μέχρι σήμερα, η πλειοψηφία των ιστοσελίδων που υπάρχουν στο διαδίκτυο, βασίζονται στην έκδοση 4.0 της HTML η οποία εμφανίστηκε και χρησιμοποιείται από το 1999. Το γεγονός όμως ότι ολόκληρος ο παγκόσμιος ιστός, έχει αλλάξει και εξελιχθεί πολύ από τότε, καθιστούσε επιτακτική την ανάγκη για «εκσυγχρονισμό» της HTML.

Η HTML5, αποτελεί πλέον την νέα «έκδοση» ή καλύτερα, την νέα και πιο σύγχρονη προτυποποίηση της HTML, XHTML και HTML DOM. Η ανάπτυξή της, βρίσκεται ακόμα σε εξέλιξη, ωστόσο πολλοί σύγχρονοι περιηγητές, υποστηρίζουν πολλά από τα νέα χαρακτηριστικά που ενσωματώνει η HTML5.

Για τη δημιουργία της HTML5 χρειάστηκε η συνεργασία ανάμεσα στο World Wide Web Consortium (W3C) και στο Web Hypertext Application Working Group (WHATWG). Πιο συγκεκριμένα, το WHATWG ασχολούταν με τις δικτυακές φόρμες και εφαρμογές, ενώ το W3C ασχολούταν με την XHTML 2.0. Το 2006 αποφάσισαν να συνεργαστούν και να δημιουργήσουν την νέα έκδοση της HTML.

Για την υλοποίηση της νέας έκδοσης HTML, την HTML5, ήταν απαραίτητο να θεσπιστούν ορισμένοι νέοι κανόνες. Ορισμένοι από αυτούς, συνοψίζονται στην ακόλουθη λίστα:

- τα νέα χαρακτηριστικά θα πρέπει να βασίζονται στην HTML, τα CSS, το DOM (Document Object Model) και την JavaScript
- μείωση στην ανάγκη για εξωτερικά πρόσθετα όπως flash
- καλύτερος χειρισμός των σφαλμάτων
- περισσότερη σήμανση (markup) που θα αντικαταστήσει τα σενάρια εντολών
- η HTML5 θα πρέπει να μην εξαρτάται από την συσκευή που την «διαβάζει»
- η διαδικασία ανάπτυξης θα πρέπει να είναι ορατή στο κοινό

Η HTML5, εισάγει αρκετά νέα σύγχρονα στοιχεία, τα οποία παρουσιάζουν εξαιρετικό ενδιαφέρον, όπως:

- το στοιχείο του καμβά για ζωγραφική
- τα στοιχεία «video» και «audio» για την αναπαραγωγή πολυμέσων
- καλύτερη υποστήριξη για τοπική αποθήκευση εκτός σύνδεσης
- νέα στοιχεία ειδικά για το περιεχόμενο, όπως άρθρο, κεφαλίδα, υποσέλιδο, πλοήγηση
- νέους χειρισμούς στις φόρμες, όπως ημερολόγια, ημερομηνία, ώρα, email, url, αναζήτηση κλπ

Αν και η HTML5 δεν αποτελεί ακόμα επίσημο πρότυπο (standard, πολλοί μεγάλοι και σύγχρονοι περιηγητές όπως οι Safari, Chrome, Firefox, Opera, Internet Explorer προσθέτουν συνεχώς νέα χαρακτηριστικά στις νεότερες εκδόσεις τους.

Το Phoneygar, αποτελώντας και το ίδιο, μια σύγχρονη πλατφόρμα, δημιουργίας εφαρμογών, υποστηρίζει την HTML5, προσφέροντας με αυτόν τον τρόπο περισσότερες επιλογές στον προγραμματιστή για την ενσωμάτωση επιπλέον στοιχείων, χωρίς να χρειάζονται σε πολλές περιπτώσεις, όπως στο παρελθόν, επιπλέον σενάρια εντολών.

Επιπλέον, η ενσωμάτωση νέων στοιχείων στην HTML5 κάνει πολύ πιο ευχάριστο και εύχρηστο το περιβάλλον διεπαφής με το χρήστη, καθώς όπως και στους περιηγητές, έτσι και στην ανάπτυξη εφαρμογών με χρήση του Phoneygar, όλο το περιβάλλον διεπαφής βασίζεται στον προγραμματισμό με HTML.

2.2.4 Πρωτόκολλο XML-RPC

Η διαδικτυακή υπηρεσία (webservice) της επιχείρησης είναι ανεπτυγμένη με βάση το πρωτόκολλο XML-RPC για την επικοινωνία του με τις εφαρμογές που επιχειρούν να αποκτήσουν πρόσβαση στη βάση δεδομένων για να καταχωρήσουν στοιχεία.

Το XML-RPC (EXtensible Markup Language -Remote Procedure Calling) (Scripting News, Inc, 1999) είναι ένα σύνολο εφαρμογών, που επιτρέπουν σε προγράμματα που τρέχουν σε διαφορετικά και ανόμοια λειτουργικά περιβάλλοντα, να κάνουν κλήσεις διαδικασιών μέσω του διαδικτύου. Πρόκειται για μία απομακρυσμένη διαδικασία κλήσεις, που χρησιμοποιεί την HTTP σαν «βαγόνι» για τη μεταφορά και την XML σαν κωδικοποίηση. Ένα XML-RPC μήνυμα, είναι εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

μία «αίτηση» HTTP-POST. Το σώμα της αίτησης-μηνύματος είναι σε XML. Κατά την αποστολή του (HTTP-POST) μια διαδικασία τρέχει από την πλευρά του εξυπηρετητή (server) και η τιμή-απάντηση που επιστρέφει, είναι επίσης μορφοποιημένη σε XML. Η XML-RPC, έχει σχεδιαστεί να είναι όσο το δυνατόν πιο απλή ενώ ταυτόχρονα να επιτρέπει σύνθετες δομές δεδομένων να εκπέμπονται, να προωθούνται και να επιστρέφουν.

Επικοινωνία με την διαδικτυακή υπηρεσία του Spitogatos.gr.

Η αρχική σκέψη, για την επικοινωνία, με τη διαδικτυακή υπηρεσία (web service) της επιχείρησης ήταν να γίνει, ανάπτυξη κώδικα σε JavaScript, όπου θα μαζεύονταν τα κατάλληλα δεδομένα από τα πεδία που θα συμπλήρωνε ο χρήστης και τα οποία θα μορφοποιούνταν σε συμβατή μορφή XML με την μέθοδο που υποστηρίζει η επιχείρηση (Christophilus, 2005). Ωστόσο, κατά τη διάρκεια της έρευνας αυτού του πεδίου, της εφαρμογής, κρίθηκε ότι η ανάπτυξη κώδικα, που στην ουσία θα αποτελούσε μια «αυτοσχέδια» βιβλιοθήκη, αφενός, ξέφυγε από τους αντικειμενικούς στόχους της παρούσας διατριβής, και αφετέρου, μια τέτοια ανάπτυξη, ίσως να είχε περιορισμούς αναφορικά με την μελλοντική επέκταση της εφαρμογής, αφού θα στηριζόταν μόνο στα συγκεκριμένα πεδία που υποστηρίζει κατά την τρέχουσα έκδοσή της η εφαρμογή, και όχι, ενδεχομένως σε ακόμα περισσότερα..

Γι' αυτό το λόγο, η έρευνα στράφηκε για την εύρεση συμβατών βιβλιοθηκών που θα μπορούσαν με μικρή ή καθόλου παραμετροποίηση, να εξυπηρετήσουν τους στόχους της εφαρμογής. Με τον όρο συμβατές, ουσιαστικά εννοούμε βιβλιοθήκες, που θα βασιζόταν στους δύο θεμελιώδεις και κεντρικούς πυλώνες για την ανάπτυξη της εφαρμογής, που είναι φυσικά η HTTP και η JavaScript. Είναι πολύ σημαντικό να οριστεί αυτό, καθώς ενώ η αναζήτηση επέστρεφε πολλά αποτελέσματα, όπου αρκετά από αυτά βασίζονταν σε χρήση PHP (η οποία **δεν** υποστηρίζεται και **δεν** είναι συμβατή με το Phonegap), ενώ αντίθετα τα αποτελέσματα τα οποία ήταν ανεπτυγμένα αποκλειστικά σε HTTP και JavaScript ήταν πολύ περιορισμένα.

2.2.5 Spitogatos.gr webservice

Σύμφωνα με την W3C, μια υπηρεσία δικτύου (webservice) ορίζεται ένα σύστημα (λογισμικού) το οποίο είναι σχεδιασμένο να υποστηρίζει διαλειτουργικές αλληλεπιδράσεις μεταξύ μηχανών (ηλεκτρονικών υπολογιστών) μέσω ενός δικτύου. Έχει ένα περιβάλλον διεπαφής το οποίο ορίζεται σε μια μορφή που είναι προσβάσιμη από τις «μηχανές» (υπολογιστές), μια ειδική γλώσσα περιγραφής υπηρεσιών διαδικτύου η οποία είναι γνωστή ως WDSL (WebServices Description Language). Τα άλλα συστήματα, αλληλεπιδρούν με την υπηρεσία δικτύου με έναν τρόπο που περιγράφεται από την περιγραφή της, χρησιμοποιώντας SOAP (Simple Object Access Protocol) μηνύματα, τα οποία μεταφέρονται χρησιμοποιώντας HTTP με XML σε μορφή που μπορεί να μεταφερθεί εύκολα μέσω διαδικτύου (serialization), σε συνδυασμό με άλλα πρότυπα σχετισμένα με δίκτυα.

Η διαδικτυακή υπηρεσία του Spitogatos.gr είναι βασισμένη στο πρωτόκολλο XML-RPC το οποίο είναι ένα πρωτόκολλο «κλήσης απομακρυσμένης διαδικασίας» το οποίο λειτουργεί μέσω του διαδικτύου. Το μήνυμα XML-RPC είναι ένα αίτημα HTTP-POST, ενώ το κυρίως σώμα του αιτήματος είναι δομημένο σε XML. Η διαδικασία εκτελείται στον διακομιστή (server), ενώ η τιμή που επιστρέφεται είναι επίσης δομημένη σε XML (Spitogatos.gr, 2011).

Για κάθε αίτημα προς την διαδικτυακή υπηρεσία της επιχείρησης είναι απαραίτητη η ύπαρξη και η κατοχή εκ μέρους του χρήστη, ενός κλειδιού για χρήση της εφαρμογής (application key) το οποίο διατίθεται σε συγκεκριμένους χρήστες-συνεργάτες της επιχείρησης.

Η διαδικτυακή υπηρεσία της επιχείρησης προσφέρει συγκεκριμένες υπηρεσίες, οι οποίες στα πλαίσια της υπηρεσίας καλούνται «μέθοδοι» και είναι οι:

- προσθήκη νέας αγγελίας
- προσθήκη φωτογραφίας
- διαγραφή φωτογραφίας
- ανάκτηση αποθηκευμένης φωτογραφίας
- τροποποίηση υπάρχουσας αγγελίας
- διαγραφή αγγελίας
- ανάκτηση καταχωρημένης αγγελίας
- αναζήτηση αγγελιών

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

Ανάλογα με την μέθοδο που θέλει να χρησιμοποιήσει ο χρήστης, πρέπει να χρησιμοποιήσει και τις κατάλληλες παραμέτρους όπως είναι ορισμένες και υποστηρίζονται από τη διαδικτυακή υπηρεσία της επιχείρησης, ενώ σε περίπτωση που υπάρξει κάποιο πρόβλημα στην επεξεργασία του αιτήματος, η υπηρεσία δικτύου, επιστρέφει μια τιμή σε XML με τον κωδικό σφάλματος και την περιγραφή που αντιστοιχεί στο συγκεκριμένο σφάλμα.

Στα πλαίσια της συγκεκριμένης διατριβής, χρησιμοποιήθηκαν δύο από τις διαθέσιμες μεθόδους που υποστηρίζει η διαδικτυακή υπηρεσία της επιχείρησης και συγκεκριμένα η «καταχώρηση νέας αγγελίας» και η «καταχώρηση φωτογραφιών». Για τον λόγο αυτό η Spiritogtos.gr παρείχε και το απαραίτητο κλειδί για χρήση της εφαρμογής αλλά και όνομα χρήστη και κωδικό πρόσβασης και έλεγχο των διεργασιών που απαιτούνταν.

2.3 Αναλυτική περιγραφή όλων των τεχνολογιών που μελετήθηκαν αλλά δε χρησιμοποιήθηκαν

Εκτός από τις τεχνολογίες που χρησιμοποιήθηκαν στα πλαίσια εκπόνησης της διατριβής, υπήρχαν και αρκετές τεχνολογίες και πιο συγκεκριμένα βιβλιοθήκες οι οποίες μελετήθηκαν αλλά τελικά δεν χρησιμοποιήθηκαν για διάφορους λόγους, ξεχωριστούς σε κάθε περίπτωση.

Πρόκειται είτε για βιβλιοθήκες των οποίων η λειτουργικότητα δεν επαληθεύτηκε κατά την μελέτη, για βιβλιοθήκες οι οποίες δεν ήταν ολοκληρωμένες, βιβλιοθήκες οι οποίες βρίσκονταν ακόμα υπό ανάπτυξη, αλλά και βιβλιοθήκες οι οποίες αν και πληρούν τις προϋποθέσεις χρήσης τους για περαιτέρω βελτίωση της εφαρμογής δεν αποτελούν στόχο περαιτέρω ανάπτυξης με βάση τους αντικειμενικούς στόχους της συγκεκριμένης διατριβής, και με δεδομένο ότι δεν αποτελούν ικανή και αναγκαία προϋπόθεση για την λειτουργικότητα της εφαρμογής, δεν ενσωματώθηκαν, παρατίθενται ωστόσο και μελλοντική μελέτη και αναβάθμιση της εφαρμογής.

JQueryUI.js

Η βιβλιοθήκη jqueryUI.js (jQuery UI Team, 2011), παρέχει στον προγραμματιστή τη δυνατότητα σε συνδυασμό με τη χρήση της jquery.js να φτιάξει ένα περιβάλλον διεπαφής το οποίο μπορεί να είναι πολύ περισσότερο παραμετροποιήσιμο, αλλά και να ενσωματώνει επιπλέον λειτουργίες που αφορούν την αλληλεπίδραση με τον χρήστη (σε αυτό οφείλονται και τα αρχικά UI στο όνομά της – User Interface). Πρόκειται για μια πολύ πλούσια βιβλιοθήκη, η οποία εστιάζει στην βελτίωση της εμπειρίας του χρήστη μέσω της ακόμα καλύτερης διάρθρωσης, οργάνωσης αλλά και δομής του περιβάλλοντος διεπαφής με τον ίδιο (τον χρήστη).

Ωστόσο, επειδή το κομμάτι του περιβάλλοντος διεπαφής δεν αποτελούσε αντικειμενικό σκοπό του θέματος της διατριβής, σε συνδυασμό με το γεγονός ότι τα απαραίτητα και αναγκαία εργαλεία διεπαφής που παρέχονταν από το Phonegap σε συνδυασμό με τις υπάρχουσες λειτουργίες που ενσωματώνουν τα κινητά τηλέφωνα, ήταν αρκετά για την εκπόνηση του αντικειμενικού στόχου και ιδέας (proof of concept) τη παρούσας εργασίας, επιλέχθηκε να μην ενσωματωθεί η συγκεκριμένη βιβλιοθήκη στην **πρώτη, βασική και σταθερή** έκδοση της εφαρμογής.

JQueryMobile.js

Η βιβλιοθήκη JQueryMobile (jQuery foundation, 2011) θα μπορούσε να χαρακτηριστεί ως «συγγενής» της JQuery, με προσανατολισμό όπως προδίδει και το όνομά της την εφαρμογή σε κινητές συσκευές όπως κινητά τηλέφωνα (smartphones) και tablets. Πιστή στο δόγμα «γράψτε λιγότερα, κάντε περισσότερα» η JQueryMobile, φιλοδοξεί να παρέχει στους προγραμματιστές τα κατάλληλα εργαλεία όπου να μπορούν να αναπτύξουν δικτυακές εφαρμογές (web applications) όπου μπορούν να είναι προσβάσιμες και αξιοποιήσιμες από όλα τα λειτουργικά συστήματα που χρησιμοποιούνται στα κινητά τηλέφωνα (smartphones) και στα tablet.

Κατά τη διάρκεια εκπόνησης της διατριβής η JQueryMobile βρισκόταν και αυτή σε beta (δοκιμαστική) έκδοση, το οποίο αποτέλεσε ανασταλτικό παράγοντα στην περαιτέρω χρησιμοποίησή της, καθώς με δεδομένο ότι και το Phonegap ήταν εξίσου σε δοκιμαστική έκδοση, υπήρχε μεγάλος κίνδυνος για αστοχίες και «συγκρούσεις» οι οποίες θα ήταν δύσκολο

να προσδιοριστούν αν οφείλονται σε εν γένει διαφορετική προσέγγιση των δύο τεχνολογιών (η JQueryMobile προσανατολίζεται στην ανάπτυξη δικτυακών εφαρμογών – web applications ενώ το Phonegap κάνοντας χρήση δικτυακών τεχνολογιών στοχεύει στην ανάπτυξη εγγενών εφαρμογών – native applications), ή απλά στο ότι βρίσκονται σε διαφορετική έκδοση.

JSON-XML-RPC

Η Json-XML-RPC (Ruter, 2007) πρόκειται για την πρώτη βιβλιοθήκη που δοκιμάστηκε, και ήταν και η πρώτη αποτυχημένη προσπάθεια σύνδεσης με τη διαδικτυακή υπηρεσία (webservice) της επιχείρησης. Η συγκεκριμένη βιβλιοθήκη, σύμφωνα με την τεκμηρίωσή της, παρείχε όλα τα απαραίτητα εργαλεία, που χρειαζόνταν για μια επιτυχημένη σύνδεση με την δικτυακή υπηρεσία της επιχείρησης, τόσο με την υπηρεσία ανάπτυξης όσο και με την εμπορική υπηρεσία αργότερα. Πιο συγκεκριμένα, επέτρεπε την επικοινωνία τόσο μέσω JSON-RPC όσο και με XML-RPC. Στην περίπτωση μας, μας ενδιέφερε η περίπτωση της XM-RPC υλοποίησης.

Η συγκεκριμένη βιβλιοθήκη επιτρέπει τόσο σύγχρονες όσο και ασύγχρονες κλήσεις σε διακομιστές που υποστηρίζουν είτε το πρωτόκολλο XML-RPC είτε το JSON-RPC. Η τεκμηρίωσή της είναι αρκετά καλή, και με συγκεκριμένα παραδείγματα, ωστόσο η ανάπτυξή της έχει εγκαταλειφθεί όπως επιβεβαίωσε και ο ίδιος ο δημιουργός της έπειτα από επικοινωνία μαζί του.

Από το σύνολο των βιβλιοθηκών που μελετήθηκαν, η συγκεκριμένη έμοιαζε να συνοψίζει την πιο πλήρη υλοποίηση ανάμεσα σε όλες τις βιβλιοθήκες που βασίζονταν σε JavaScript για επικοινωνία με διακομιστές μέσω XML-RPC. Ωστόσο, αυτή ακριβώς η πληρότητά της, αποτελούσε ταυτόχρονα και τον ανασταλτικό παράγοντα που έκανε αδύνατη τη χρησιμοποίησή της. Ειδικότερα, με δεδομένη την ύπαρξη περιορισμών από την πλευρά των περιηγητών για την υλοποίηση επικοινωνιών cross-site για λογους ασφαλείας, η συγκεκριμένη βιβλιοθήκη, προχώραγε ένα βήμα παραπάνω από τις υπόλοιπες, και εμπόδιζε η ίδια της cross-site επικοινωνίες και ποιο συγκεκριμένα τις «σύγχρονες» κλήσεις.

Αυτό είχε σαν αποτέλεσμα, παρόλο που το phonegap δεν παρείχε περιορισμούς αναφορικά με το cross-site scripting, η ίδια η βιβλιοθήκη να αναγνωρίζει ότι η επιχειρούμενη κλήση παραβίαζε τους ελέγχους «cross-site» που υπήρχαν ενσωματωμένοι, με αποτέλεσμα να μην επιτρέπει να γίνονται οι κλήσεις.

Προκειμένου να χρησιμοποιηθεί η συγκεκριμένη βιβλιοθήκη, είναι απαραίτητο να μελετηθεί εξονυχιστικά όλος ο κώδικάς της, προκειμένου να γίνει αντιληπτό πως χειρίζεται της περιπτώσεις cross-site που οδηγούν σε εξαίρεση τον κώδικα, έτσι ώστε αφενός να απενεργοποιηθούν τα συγκεκριμένα κομμάτια του κώδικα, αλλά ταυτόχρονα να μην επηρεαστούν άλλα τμήματα τα οποία είναι επιθυμητά για την επίτευξη της ζητούμενης λειτουργικότητας. Για τον λόγο αυτό, και με δεδομένο ότι η ανάπτυξη μιας βιβλιοθήκης, για επίτευξη κλήσεων μέσω του πρωτοκόλλου XML-RPC δεν αποτελούσε αντικειμενικό σκοπό της διατριβής, η συγκεκριμένη βιβλιοθήκη αρχικά παραμερίστηκε, προκειμένου να εξεταστεί αν κάποια από τις υπόλοιπες μπορούσε να ανταποκριθεί στις απαιτήσεις χωρίς καθόλου, ή με ελάχιστες τροποποιήσεις.

Όταν πλέον έγινε κατορθωτή η επίτευξη κλήσης με τη χρήση της βιβλιοθήκης mimic.js, απορρίφθηκε οριστικά όσον αφορά την συγκεκριμένη υλοποίηση η JSON-XML RPC. Ωστόσο, παραμένει μια αρκετά καλή βιβλιοθήκη, όπου ειδικά αν το πρόβλημα της επιβεβαίωσης ονόματος χρήστη και κωδικού πρόσβασης (HTTP-Authentication) δεν μπορεί να παρακαμφθεί διαφορετικά (όπως στη περίπτωση της υπηρεσίας ανάπτυξης της Spitogatos.gr όπου απενεργοποιήθηκε, καθώς δεν αποτελούσε άλλωστε ουσιαστικό εμπόδιο με δεδομένο ότι η εμπορική υπηρεσία δεν είχε ανάλογη απαίτηση) τότε σίγουρα αξίζει η περαιτέρω μελέτη και ανάπτυξή της για ενσωμάτωση σε κώδικα, όπου κριθεί απαραίτητη.

JS-XMLRPC (XML-RPC for JavaScript)

Η JS-XMLRPC (Giunta & Lott, 2009) βιβλιοθήκη, αποτέλεσε την δεύτερη κατά σειρά βιβλιοθήκη, η οποία εξετάστηκε. Όπως και οι υπόλοιπες, έτσι και στη συγκεκριμένη βιβλιοθήκη, η τελευταία έκδοση που υπάρχει διαθέσιμη στο διαδίκτυο είναι σχετικά παλιά, και συγκεκριμένα είναι η

έκδοση 0.4 που υπάρχει από τον Σεπτέμβριο του 2009. Μαζί με τη συγκεκριμένη βιβλιοθήκη, παρέχεται και μία αρκετά κατατοπιστική τεκμηρίωση (documentation), ενώ υπάρχει και online εργαλείο αποσφαλμάτωσης (debugger) προκειμένου να ελέγξει ο προγραμματιστής αν τα δεδομένα που εισάγει για να επιτύχει επικοινωνία με τον επιθυμητό διακομιστή είναι σωστά.

Με τη συγκεκριμένη βιβλιοθήκη, δεν κατέστη επίσης δυνατό να επιτευχθεί επικοινωνία με τη διαδικτυακή υπηρεσία της επιχείρησης. Τα προβλήματα που φάνηκαν να υπάρχουν αφορούσαν κυρίως την επίτευξη επικοινωνίας με τη διαδικτυακή υπηρεσία της επιχείρησης. Προκειμένου να εξεταστεί εκτενέστερα το θέμα της επικοινωνίας, έγινε και χρήση του εργαλείου αποσφαλμάτωσης που διαθέτει η συγκεκριμένη βιβλιοθήκη (Giunta & Lott, 2009), με την ελπίδα ότι θα βρισκόταν κάτι πιο συγκεκριμένο αναφορικά με το τι προκαλούσε το πρόβλημα, ωστόσο ανεξαρτήτως των παραμέτρων που τίθεντο, το εργαλείο αποσφαλμάτωσης επέστρεφε συνεχώς σαν μήνυμα λάθους «Fault code: [5] Reason: 'Didn't receive 200 OK from remote server. (send failed)»», με αποτέλεσμα να εγκαταλειφθεί.

Με δεδομένο ότι ο περιορισμός του cross-site δεν είχε γίνει αντιληπτός εξ'άρχης αλλά και ότι κατά την εκτέλεση του συγκεκριμένου κώδικα μέσω περιηγητών δεν εμφανίζονταν κάποιο μήνυμα που να παρέπεμπε σε θέμα εξαίρεσης λόγω cross-site περιορισμών, θεωρήθηκε ότι ο συγκεκριμένος κώδικας παρουσίαζε λειτουργικά προβλήματα.

Αργότερα, και όταν πλέον είχε εντοπιστεί το πρόβλημα που δημιουργούσε ο περιορισμός του cross-site scripting, έγινε δοκιμή του εργαλείου αποσφαλμάτωσης εκ νέου (αυτή τη φορά μέσω του Google Chrome ο οποίος είχε απενεργοποιημένες όλες τις επιλογές ασφαλείας του δικτύου) και όντως, το εργαλείο αποσφαλμάτωσης, έδειξε σημάδια ότι η βιβλιοθήκη ενδεχομένως να είναι όντως λειτουργική. Ωστόσο, κατά το συγκεκριμένο χρόνο, η mimic.js είχε «εδραιωθεί» αναφορικά με τη χρήση σε συνδυασμό με την απαραίτητη αλλαγή στην υπηρεσία ανάπτυξης της επιχείρησης όπου είχε απενεργοποιηθεί η ανάγκη επιβεβαίωσης ονόματος χρήστη και κωδικού πρόσβασης (HTTP-Authentication) και δεν συνέτρεχε πλέον λόγος αλλαγής της καθώς ανεξάρτητα με τα θετικά σημάδια που φάνηκαν κατά την δοκιμή της JS-XMLRPC με τον περιηγητή Google Chrome (με συγκεκριμένες προϋποθέσεις) εκκρεμούσε, η προβληματική επικοινωνία με τη χρήση του Phoneygar και η αμφίβολη περαιτέρω λειτουργικότητα αναφορικά με την σωστή δόμηση των δεδομένων.

Μια παρατήρηση που ενδεχομένως αξίζει να σημειωθεί σχετικά με τη συγκεκριμένη βιβλιοθήκη, αποτελεί ίσως το μήνυμα που χρησιμοποιείται στο εργαλείο αποσφαλμάτωσης προκειμένου να περιγράψει την αδυναμία επικοινωνίας. Αν και όντως η αδυναμία επικοινωνίας με τον διακομιστή περιγράφεται σαν «σφάλμα 200» στη προκειμένη περίπτωση, θα έπρεπε να τόνιζεται στο χρήστη, (όπως γινόταν στην JSON-XML-RPC βιβλιοθήκη) το ενδεχόμενο να υπάρχει εξαίρεση λόγω cross-site περιορισμών ασφαλείας.

Jsolait

Η Jsolait (Kollhof, 2003), αποτελούσε την πιο «παλιά» βιβλιοθήκη, από όσες βρέθηκαν που θα μπορούσαν να αξιοποιηθούν. Για το λόγο αυτό, αποτελούσε την «έσχατη» λύση για διερεύνηση εφόσον δεν απέδιδε η μελέτη και η εφαρμογή των υπολοίπων πιο «σύγχρονων» βιβλιοθηκών. Το γεγονός ωστόσο ότι η mimic.js κάλυψε τις ανάγκες, και φάνηκε να είναι αρκετά απλή στη χρήση της, είχε σαν αποτέλεσμα, να μην γίνει περαιτέρω έρευνα για την Jsolait. Αποτελεί ωστόσο σημείο αναφοράς, καθώς υπάρχουν αρκετές αναφορές στο διαδίκτυο, αλλά και ο δημιουργός της JSON-XML-RPC αναφέρει, πως τμήματα αυτής αποτέλεσαν «έμπνευση» για την ανάπτυξη της JSON-XML-RPC (ειδικά ο χειρισμός του «πελάτη» (client) JavaScript-RPC).

Κεφάλαιο 3

3.1 Περιγραφή της εφαρμογής

Προκειμένου ο χρήστης να μπορέσει να κάνει χρήση της εφαρμογής, θα πρέπει να την εγκαταστήσει στην έξυπνη συσκευή του. Εφόσον η εφαρμογή εγκατασταθεί, μπορεί πλέον ο χρήστης να τη χρησιμοποιήσει, επιλέγοντας την από το μενού των εφαρμογών και να τη χρησιμοποιήσει. Το γραφικό περιβάλλον διεπαφής με το χρήστη (GUI – Graphic User Interface) που εμφανίζεται στην εφαρμογή, είναι κοινό ανεξάρτητα από το λειτουργικό σύστημα στο οποίο προορίζεται (συνέπεια του κοινού πηγαίου κώδικα στον οποίο είναι γραμμένη η εφαρμογή). Ωστόσο οι ιδιαιτερότητες των εκάστοτε λειτουργικών συστημάτων, μπορεί να επηρεάζουν τον τρόπο με τον οποίο ο χρήστης αλληλεπιδρά με την εφαρμογή χωρίς ωστόσο να επηρεάζεται η λειτουργικότητα της εφαρμογής καθαυτή.

Η βασική λειτουργικότητα της εφαρμογής είναι η καταχώρηση αγγελιών ακινήτων μέσω της διαδικτυακής υπηρεσίας (web service) του Spitogatos.gr.

3.2 Περιγραφή της αρχιτεκτονικής της εφαρμογής

Οι εφαρμογές οι οποίες είναι γραμμένες σε συμβατό κώδικα με το αντίστοιχο λειτουργικό για το οποίο προορίζονται, έχουν απευθείας πρόσβαση στις διάφορες λειτουργίες της έξυπνης συσκευής, καθώς γίνεται άμεσα η επικοινωνία με το λειτουργικό σύστημα. Στην περίπτωση της εφαρμογής η οποία αναπτύσσεται με τη χρήση του Phonegap, η πρόσβαση της εφαρμογής στις διάφορες υπηρεσίες και λειτουργίες της έξυπνης συσκευής και του λειτουργικού συστήματος, εξασφαλίζεται με τα αντίστοιχα εργαλεία που παρέχει το Phonegap, προκειμένου να επιτευχθεί αυτή η δυνατότητα.

Όταν ο χρήστης ανοίγει την εφαρμογή, η εφαρμογή αποκτά πρόσβαση στις διαθέσιμες υπηρεσίες και λειτουργίες που παρέχει η έξυπνη συσκευή. Στην προκειμένη περίπτωση, η εφαρμογή που αναπτύχθηκε στα πλαίσια της παρούσας διατριβής, προκειμένου να επιτύχει τον αντικειμενικό της σκοπό, θα πρέπει να έχει πρόσβαση στο διαδίκτυο. Επιπλέον, για να επιτευχθεί μεγαλύτερη αξιοποίηση των δυνατοτήτων που έχουν οι σύγχρονες έξυπνες συσκευές, υπάρχει η δυνατότητα από το χρήστη να αποκτήσει πρόσβαση τόσο στον κατάλογο των φωτογραφιών που είναι αποθηκευμένες στην εφαρμογή, όσο και στην φωτογραφική μηχανή που είναι ενσωματωμένη σε όλες πλέον τις έξυπνες συσκευές, αλλά και σε λειτουργίες εύρεσης και προσδιορισμού του γεωγραφικού στίγματος στο οποίο βρίσκεται ο χρήστης τη δεδομένη στιγμή.

Κατά την εκκίνηση της εφαρμογής, εμφανίζεται στο χρήστη η βασική οθόνη της εφαρμογής, όπου πρόκειται για μία φόρμα συμπλήρωσης στοιχείων σχετικά με το είδος του ακινήτου που θέλει να καταχωρήσει ο χρήστης, όπως φαίνεται και στην εικόνα 2.

Η φόρμα, η οποία καλείται να χρησιμοποιήσει και να συμπληρώσει ο χρήστης, αποτελείται μόνο από τα απαραίτητα στοιχεία τα οποία έχουν οριστεί ως προαπαιτούμενα στη διαδικτυακή υπηρεσία του Spitogatos.gr, και τα οποία είναι:

- κατηγορία ακινήτου, αν το ακίνητο είναι κατοικία, επαγγελματικό ακίνητο ή αξιοποιήσιμη γη,
- είδος ακινήτου, εξαρτάται από την κατηγορία του ακινήτου,
- νόμισμα, ευρώ δολάριο, ή λίρα Αγγλίας
- είδος αγγελίας, αν πρόκειται για αγγελία πώλησης ή ενοικίασης
- περιοχή, που βρίσκεται το ακίνητο
- υποπεριοχή, της κύριας περιοχής η οποία έχει προσδιοριστεί

στην περίπτωση που η κατηγορία του ακινήτου είναι κατοικία ή επαγγελματικό ακίνητο, εμφανίζονται στη φόρμα και 3 (τρεις) επιπλέον υποχρεωτικές επιλογές, οι οποίες είναι

- αριθμός των δωματίων που έχει το ακίνητο,
- έτος κατασκευής του ακινήτου,
- όροφος, στον οποίο βρίσκεται το ακίνητο



spitogatos.gr
Η Νο1 Ελληνική ιστοσελίδα αγγελιών ακινήτων

Παρακαλώ συμπληρώστε τη φόρμα:

Κατηγορία ακινήτου:

Είδος ακινήτου:

Τιμή (χωρίς δεκαδικά)/Νόμισμα:

Εμβαδό (σε τ.μ.):

Είδος αγγελίας:

Περιοχή:

Υποπεριοχή:

Προσδιορίστε το χειροκίνητα ή πιέστε το κουμπί

Γεωγραφικό πλάτος:

Γεωγραφικό μήκος:

Ωρα προσδιορισμού στίγματος:

Αν θέλετε να μεταβείτε στην καταχώρηση φωτογραφιών πατήστε το κουμπί

Εικόνα 2 - Βασικό περιβάλλον διεπαφής

Τα πεδία:

- γεωγραφικό πλάτος
- γεωγραφικό μήκος
- ώρα προσδιορισμού του στίγματος

δεν είναι απαραίτητα για την καταχώρηση αγγελίας, και συμπληρώνονται από την εφαρμογή με την επιλογή του κουμπιού «εύρεση», όπου ανάλογα με το αν υπάρχει πρόσβαση στο διαδίκτυο (κατά προτεραιότητα) προσδιορίζεται το γεωγραφικό στίγμα της έξυπνης συσκευής. Ωστόσο, σε περίπτωση που ο χρήστης θέλει να προσδιορίσει ο ίδιος το γεωγραφικό στίγμα, μπορεί να το κάνει, εισάγοντας τα αντίστοιχα στοιχεία στα αντίστοιχα πεδία.

Το πεδίο «ώρα προσδιορισμού του στίγματος» εξυπηρετεί μόνο στο να ενημερωθεί ο χρήστης, αν τα γεωγραφικά δεδομένα που του παρέχει η εφαρμογή είναι τρέχοντα, ή αν έχουν εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

παραμένει στην προσωρινή μνήμη του λειτουργικού συστήματος της έξυπνης συσκευής από προηγούμενη χρήση.

Το Phonegap, είναι «υπεύθυνο» για να «τραβήξει» από το λειτουργικό σύστημα της εφαρμογής τα γεωγραφικά δεδομένα, προκειμένου αυτά να περαστούν και να εμφανιστούν στην εφαρμογή. Με χρήση της σύνδεσης στο διαδίκτυο, προσδιορίζεται το γεωγραφικό στίγμα του οποίου η ακρίβεια εξαρτάται από το είδος της σύνδεσης. Σε περίπτωση που η έξυπνη συσκευή έχει συνδεθεί μέσω WiFi, που σημαίνει, ότι υπάρχει κοντά κάποια σταθερή ενσύρματη σύνδεση, το στίγμα παρουσιάζει αρκετά καλή ακρίβεια. Σε περίπτωση που η συσκευή έχει συνδεθεί στο διαδίκτυο μέσω δικτύου 3G/GPRS το γεωγραφικό στίγμα έχει μικρότερη ακρίβεια. Για το λόγο αυτό, ενώ στη βάση δεδομένων του Spitogatos.gr επιτρέπεται να οριστεί η ακρίβεια του σημείου σε τρία διαφορετικά επίπεδα:

- επίπεδο δήμου
- επίπεδο γειτονιάς
- ακριβής τοποθεσία,

η εφαρμογή στέλνει εξ'ορισμού (by default) την επιλογή ότι η ακρίβεια του σημείου είναι σε «επίπεδο γειτονιάς». Ένας ακόμα λόγος που γίνεται αυτό, είναι ότι ο χρήστης βλέπει μόνο τις συντεταγμένες, και όχι την ακριβή τοποθεσία σε χάρτη, ώστε να ξέρει αν έχει προσδιοριστεί η το στίγμα του με μεγάλη ακρίβεια.

Για τον έλεγχο της ορθής συμπλήρωσης της φόρμας γίνονται απαραίτητοι και αναγκαίοι έλεγχοι μέσω των βιβλιοθηκών JavaScript που έχουν συμπεριληφθεί. Ο λόγος επίσης που πολλά από τα πεδία που καλείται να συμπληρώσει ο χρήστης, εμφανίζονται με τη μορφή μενού επιλογών, είναι προκειμένου να περιοριστεί η περίπτωση σφάλματος από την πλευρά του χρήστη, καθώς το σύνολο των καταχωρημένων τιμών που εμφανίζονται στα μενού επιλογών, αποτελούν τις τιμές που είναι συμβατές με τις αναμενόμενες τιμές που αναγνωρίζει η διαδικτυακή υπηρεσία της επιχείρησης.

Κατά την επιλογή του κουμπιού «Καταχώρηση» από την εφαρμογή, γίνεται απόπειρα σύνδεσης της εφαρμογής με τη διαδικτυακή υπηρεσία (web service) της επιχείρησης. Εφόσον η σύνδεση επιτευχθεί, γίνεται αποστολή των δεδομένων σε μορφή XML προς την διαδικτυακή υπηρεσία, η οποία ελέγχει την ορθότητα των τιμών που αποστέλλονται. Αν τα δεδομένα που στάλθηκαν στην υπηρεσία διαδικτύου της επιχείρησης είναι ορθά και σύμφωνα με τους κανόνες που έχουν οριστεί, τότε γίνεται καταχώρησή τους στη βάση δεδομένων της επιχείρησης, και αντιστοιχίζεται ένας μοναδικός αριθμός στην αγγελία (listing ID). Ο συγκεκριμένος μοναδικός αριθμός, επιστρέφεται μέσω της διαδικτυακής υπηρεσίας ως «απάντηση» στην εφαρμογή, οπότε ο χρήστης ενημερώνεται ότι η αγγελία έχει καταχωρηθεί επιτυχώς με το συγκεκριμένο αναγνωριστικό αριθμό της αγγελίας. Εφόσον η καταχώρηση στη βάση δεδομένων είναι επιτυχημένη, η αγγελία δημοσιεύεται αυτόματα, και εμφανίζεται στα αποτελέσματα αναζήτησης εφόσον πληροί τα κριτήρια αναζήτησης που θα θέσει ο εκάστοτε χρήστης.

Σε περίπτωση που παρουσιαστεί κάποιο σφάλμα, δε γίνεται καταχώρηση της αγγελίας και θα πρέπει ο χρήστης να ξαναπροσπαθήσει, αλλάζοντας τα απαραίτητα δεδομένα τα οποία οδήγησαν στο σφάλμα.

Τη διαδικασία της επικοινωνίας με τη διαδικτυακή υπηρεσία της επιχείρησης, την αποστολή των δεδομένων και την εμφάνιση της απάντησης, αναλαμβάνει η βιβλιοθήκη JavaScript Mimic που χρησιμοποιείται στην εφαρμογή.

Εφόσον η καταχώρηση της αγγελίας γίνει με επιτυχία, εμφανίζεται στον χρήστη η οθόνη όπου του δίνεται η επιλογή να προσθέσει φωτογραφίες, ενώ του εμφανίζεται και ο αριθμός τη αγγελίας, στην οποία θα καταχωρηθούν οι φωτογραφίες. Ο χρήστης έχει τη δυνατότητα είτε να προσθέσει φωτογραφίες που βρίσκονται αποθηκευμένες στην έξυπνη συσκευή του, είτε να τραβήξει σε πραγματικό χρόνο μία φωτογραφία του ακινήτου, τα στοιχεία του οποίου έχει ήδη καταχωρήσει.

Η εφαρμογή, επιτρέπει στο χρήστη να βλέπει μια «προεπισκόπηση» της φωτογραφίας που έχει επιλέξει σε μικρό μέγεθος, και ο χρήστης, μπορεί να αποστείλει τη φωτογραφία μέσω της εφαρμογής, όπου η διαδικτυακή υπηρεσία της επιχείρησης, ελέγχει την ορθότητα των δεδομένων και την καταχωρεί στην βάση δεδομένων της επιχείρησης.



Εικόνα 3 - Οθόνη προσθήκης φωτογραφιών

Η εφαρμογή δεν «επεμβαίνει» η ίδια στην έξυπνη συσκευή προκειμένου να γίνει η σύνδεση στο διαδίκτυο, ωστόσο ελέγχει πριν την προσπάθεια καταχώρησης των δεδομένων, αν όντως η έξυπνη συσκευή είναι ήδη συνδεδεμένη με το διαδίκτυο. Για το λόγο αυτό, θα πρέπει ο χρήστης να εξασφαλίσει ότι υπάρχει σύνδεση με το διαδίκτυο είτε μέσω τεχνολογίας WiFi είτε μέσω GPRS/3G.

Επίσης, η εφαρμογή δεν ανακτά πληροφορίες από τη βάση δεδομένων της επιχείρησης. Αυτό επηρέασε την ανάπτυξη της εφαρμογής με δύο τρόπους.

Αφενός, όλα τα μενού επιλογών που διατίθενται στο χρήστη προκειμένου να επιλέξει από ένα σύνολο πληροφοριών στα εκάστοτε πεδία, δεν ανακτώνται από ένα σύνολο διαθέσιμων τιμών που υπάρχει καταχωρημένο στη βάση δεδομένων, αλλά είναι ήδη καταχωρημένα στον πηγαίο κώδικα της εφαρμογής. Αυτό πρακτικά σημαίνει, ότι οποιαδήποτε μεταβολή γίνει ως

προς τις διαθέσιμες επιλογές που δίνονται μέσω της διαδικτυακής υπηρεσίας θα πρέπει να αλλαχθούν τα αντίστοιχα πεδία με τις επιλογές από τον πηγαίο κώδικα της εφαρμογής.

Αφετέρου, ένα άλλο στοιχείο που επηρέασε την αρχιτεκτονική της εφαρμογής, αφορά στον τρόπο διαχείρισης των φωτογραφιών. Σε κάθε αγγελία, υπάρχουν διαθέσιμες 10 θέσεις (slots) όπου επιτρέπεται να αντιστοιχιστούν μέχρι 10 φωτογραφίες. Σε περίπτωση που επιχειρηθεί να εισαχθεί νέα φωτογραφία, σε θέση που υπάρχει ήδη αποθηκευμένη φωτογραφία, τότε αυτή θα αντικαταστήσει την παλιά. Επειδή ωστόσο, η εφαρμογή δεν ανακτά πληροφορίες από τη βάση δεδομένων αναφορικά με το αν υπάρχουν διαθέσιμες θέσεις, και ποιες είναι αυτές, αναγκαστικά χειρίζεται το συγκεκριμένο θέμα με διαφορετικό τρόπο. Με την καταχώρηση μιας νέας αγγελίας, ένας εσωτερικός μετρητής στην εφαρμογή, παίρνει την τιμή 0. Σε κάθε καταχώρηση φωτογραφίας, ο συγκεκριμένος μετρητής αυξάνει την τιμή του κατά 1. Όταν η τιμή του μετρητή γίνει 9, σημαίνει ότι έχει συμπληρωθεί ο αριθμός των διαθέσιμων θέσεων στη βάση δεδομένων, και δεν επιτρέπεται περαιτέρω καταχώρηση φωτογραφιών, καθώς αυτό θα είχε σαν αποτέλεσμα να διαγραφούν οι παλιότερες.

Επίσης, επειδή οι φωτογραφίες, αποτελούν αρχεία με σχετικά μεγάλο μέγεθος (σε συνάρτηση και με την ποιότητα της ενσωματωμένης φωτογραφικής μηχανής της έξυπνης συσκευής), υπήρχε ένα δίλλημα αναφορικά με το πώς θα έπρεπε να γίνει η διαχείριση των φωτογραφιών. Η μία επιλογή ήταν να υπάρχει ένας δυναμικός πίνακας, όπου θα επέτρεπε στο χρήστη να προσθέτει και άλλες φωτογραφίες, οι οποίες θα ήταν διαθέσιμες για προεπισκόπηση, και να επιλέγει κάθε φορά μία για αποστολή. Η άλλη επιλογή ήταν να υπάρχει διαθέσιμη μόνο μία φωτογραφία κάθε στιγμή, όπου ο χρήστης είτε θα την καταχωρούσε στην διαδικτυακή υπηρεσία της επιχείρησης είτε θα επέλεγε νέα. Στην πρώτη περίπτωση ωστόσο, υπήρχε σοβαρός κίνδυνος, κάποιες έξυπνες συσκευές να αντιμετωπίσουν θέματα επάρκειας προσωρινής μνήμης ram και κατάρρευση της εφαρμογής (καθώς τα τεχνικά χαρακτηριστικά διαφέρουν ανάμεσα στις έξυπνες συσκευές), καθώς όλες οι φωτογραφίες, θα αποθηκεύονταν στην προσωρινή μνήμη. Για το λόγο αυτό επιλέχθηκε να γίνεται διαχείριση μίας φωτογραφίας κάθε φορά.

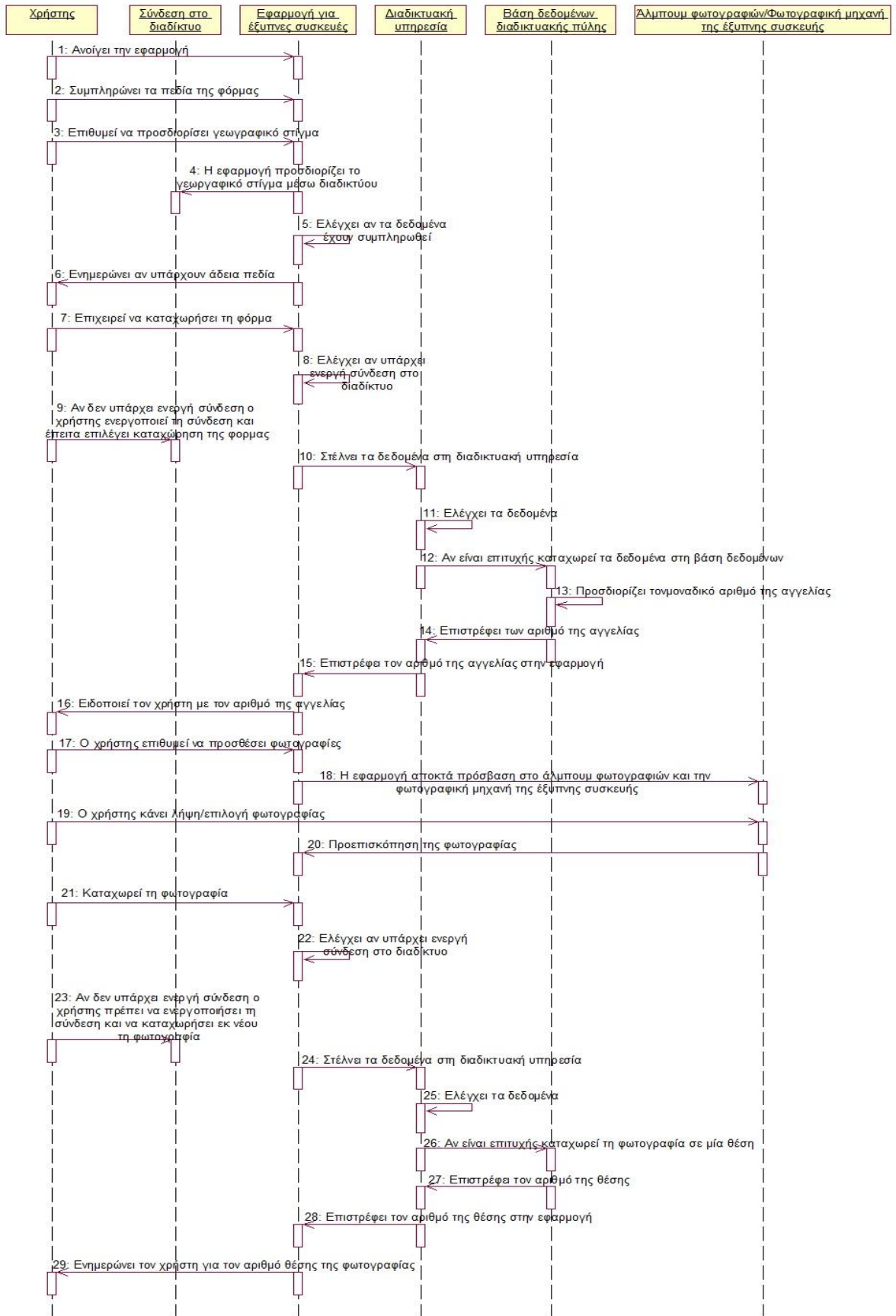
Τέλος, επειδή ανάλογα με την ποιότητα της σύνδεσης (WiFi, 3G) και την ταχύτητα αυτής, η διάρκεια της μεταφοράς των δεδομένων αλλά και της απάντησης από την διαδικτυακή υπηρεσία μπορεί να διαφέρουν κατά περίπτωση. Προκειμένου, κατά τη διάρκεια της κλήσης, ο χρήστης να μην είναι σε θέση να κάνει πολλαπλές καταχωρήσεις, θεωρώντας πως το αίτημά του δεν είναι «υπο επεξεργασία», με την επιλογή της καταχώρησης των δεδομένων, η επιφάνεια διεπαφής του χρήστη, απενεργοποιείται, μέχρι να επιστρέψει η απάντηση με την επιτυχημένη καταχώρηση από την διαδικτυακή υπηρεσία.

3.3 Περιγραφή της λειτουργικότητας της εφαρμογής

3.3.1 Διάγραμμα σειράς (Sequence diagram)

Τα διαγράμματα σειράς, αναπαριστούν αλληλεπιδράσεις ανάμεσα στα αντικείμενα από μία χρονική άποψη. Σε ένα διάγραμμα σειράς, αναπαρίσταται μια αλληλεπίδραση σε αντικείμενα που επικεντρώνεται στη χρονολογία εκπομπής μηνυμάτων (Βίρβου, 2008). Για την παρουσίαση του διαγράμματος σειράς, έγιναν ορισμένες παραδοχές, που αφορούν τη μη ύπαρξη σύνδεσης στο διαδίκτυο, την προαιρετική καταχώρηση του γεωγραφικού στίγματος, και την καταχώρηση μίας φωτογραφίας. Η διαδικασία ξεκινάει με την ενεργοποίηση της εφαρμογής από την πλευρά του χρήστη (η αρίθμηση αντιστοιχεί, στην αρίθμηση που παρουσιάζεται στη εικόνα).

1. Ο χρήστης ανοίγει την εφαρμογή.
2. Ο χρήστης συμπληρώνει τα απαιτούμενα πεδία της εφαρμογής.
3. Ο χρήστης επιθυμεί να εισάγει το γεωγραφικό στίγμα.
4. Η εφαρμογή ανακτά το γεωγραφικό στίγμα από την έξυπνη συσκευή μέσω του διαδικτύου (εφόσον υπάρχει ενεργή σύνδεση στο διαδίκτυο).
5. Η εφαρμογή ελέγχει την ακεραιότητα των πεδίων.
6. Η εφαρμογή ενημερώνει τον χρήστη αν υπάρχουν κενά πεδία.



Εικόνα 4 – Διάγραμμα σειράς

7. Ο χρήστης υποβάλλει την συμπληρωμένη φόρμα.
8. Η εφαρμογή ελέγχει αν υπάρχει ενεργή σύνδεση στο διαδίκτυο.
9. Αν δεν υπάρχει ενεργή σύνδεση, ο χρήστης ενεργοποιεί τη σύνδεση και υποβάλλει εκ νέου τη συμπληρωμένη φόρμα.
10. Η εφαρμογή στέλνει τα δεδομένα στην υπηρεσία διαδικτύου
11. Η υπηρεσία διαδικτύου επαληθεύει την ορθότητα των δεδομένων.
12. Σε περίπτωση επιτυχίας, γίνεται καταχώρηση της αγγελίας στη βάση δεδομένων.
13. Αντιστοιχίζεται στην αγγελία μοναδικός αριθμός (ListingID) για την αγγελία.
14. Επιστρέφεται ο μοναδικός αριθμός στην υπηρεσία διαδικτύου.
15. Επιστρέφεται ο μοναδικός αριθμός στην εφαρμογή.
16. Ενημερώνεται ο χρήστης για την επιτυχή καταχώρηση και εμφανίζεται στο χρήστη ο μοναδικός αριθμός της αγγελίας.
17. Ο χρήστης επιθυμεί να εισάγει φωτογραφίες.
18. Η εφαρμογή αποκτά πρόσβαση στην φωτογραφική μηχανή της συσκευής ή στο άλμπουμ φωτογραφιών
19. Ο χρήστης, επιλέγει φωτογραφία, ή τραβάει μία καινούρια.
20. Η εφαρμογή παρουσιάζει μια προεπισκόπηση της τρέχουσας επιλεγμένης φωτογραφίας.
21. Ο χρήστης καταχωρεί τη φωτογραφία.
22. Η εφαρμογή ελέγχει αν υπάρχει ενεργή σύνδεση στο διαδίκτυο.
23. Αν δεν υπάρχει ενεργή σύνδεση, ο χρήστης ενεργοποιεί τη σύνδεση και υποβάλλει εκ νέου την φωτογραφία προς καταχώρηση.
24. Η εφαρμογή στέλνει τα δεδομένα στην υπηρεσία διαδικτύου.
25. Η υπηρεσία διαδικτύου επαληθεύει την ορθότητα των δεδομένων.
26. Σε περίπτωση επιτυχίας, γίνεται καταχώρηση της φωτογραφίας στη διαθέσιμη θέση.
27. Επιστρέφεται ο αριθμός της θέσης της φωτογραφίας στην υπηρεσία διαδικτύου.
28. Επιστρέφεται ο αριθμός της θέσης της φωτογραφίας στην εφαρμογή.
29. Ενημερώνεται ο χρήστης για την επιτυχή καταχώρηση και εμφανίζεται στο χρήστη ο αριθμός της θέσης της φωτογραφίας.

Το διάγραμμα σειράς, δείχνει με σαφήνεια τις διεργασίες που λαμβάνουν χώρα κατά σειρά, προκειμένου να επιτύχει ο χρήστης το επιθυμητό αποτέλεσμα, που είναι η καταχώρηση μιας αγγελίας και στη συνέχεια η καταχώρηση των αντίστοιχων φωτογραφιών. Για την παρουσίαση του διαγράμματος σειράς, έγινε μία ευρύτερη οριοθέτηση του συστήματος, καθώς εκτός από την εφαρμογή και τις λειτουργίες που αυτή επιτελεί, παρουσιάστηκε και η αλληλεπίδρασή της, τόσο με την διαδικτυακή υπηρεσία της επιχείρησης όσο και με τη βάση δεδομένων αυτής. Ωστόσο, τόσο η υπηρεσία διαδικτύου όσο η βάση δεδομένων, δεν αποτελούν αντικείμενο ανάλυσης της παρούσας διατριβής και για το λόγο αυτό δεν έγινε εκτενέστερη ανάλυση των διεργασιών που εκτελούνται σε αυτές.

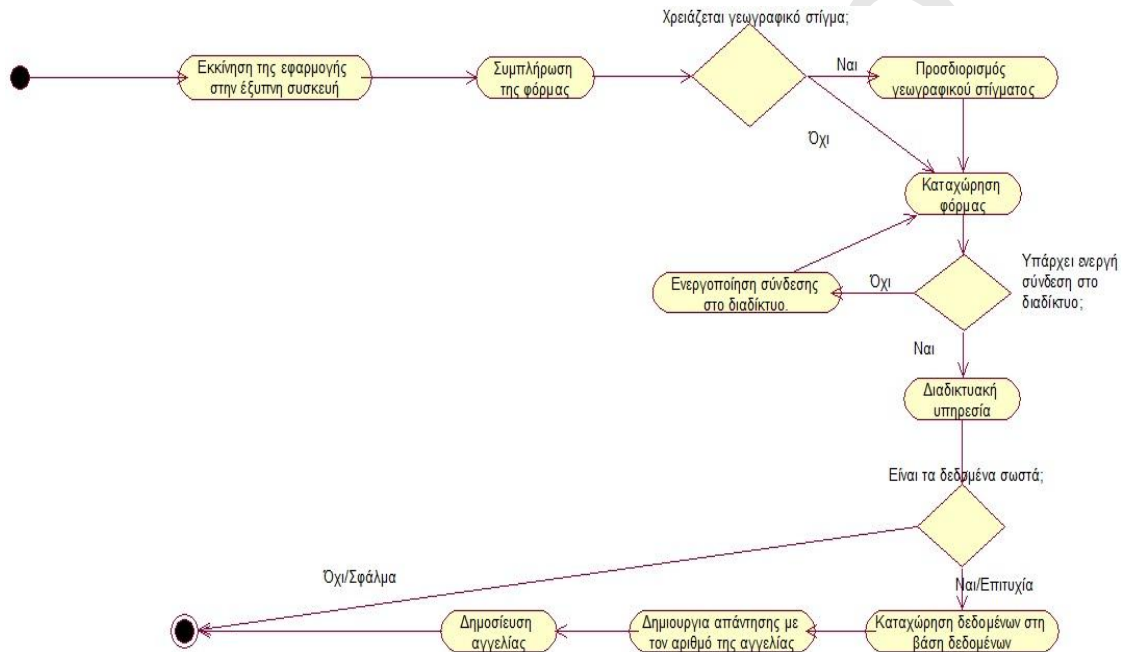
3.3.2 Διάγραμμα δραστηριοτήτων (Activity diagram)

Το διάγραμμα δραστηριοτήτων, αναπαριστά τη συμπεριφορά μια λειτουργίας ως σύνολο ενεργειών (Βίρβου, 2008). Στην προκειμένη περίπτωση, αναλύεται η λειτουργία της καταχώρησης αγγελίας, στις επιμέρους ενέργειες που λαμβάνουν χώρα για την επίτευξη της καταχώρησης. Για την αναπαράσταση του συγκεκριμένου διαγράμματος το σύστημα οριοθετείται στα πλαίσια της εφαρμογής και της επικοινωνίας με την διαδικτυακή υπηρεσία.

Πιο συγκεκριμένα, το σύνολο των δραστηριοτήτων, ξεκινάει με την εκκίνηση της εφαρμογής που είναι εγκατεστημένη στην έξυπνη συσκευή. Στη συνέχεια ο χρήστης, συμπληρώνει τη φόρμα με τα απαιτούμενα πεδία, και καλείται να αποφασίσει αν θα εισάγει και το γεωγραφικό στίγμα στην φόρμα ή όχι. Στην περίπτωση που επιθυμεί να το εισάγει, η έξυπνη συσκευή προσδιορίζει το γεωγραφικό στίγμα μέσω του διαδικτύου είτε μέσω του ενσωματωμένου αισθητήρα GPS, ενώ σε περίπτωση που δεν επιθυμεί να το εισάγει, προχωράει απευθείας στο επόμενο στάδιο που είναι η καταχώρηση της φόρμας στη διαδικτυακή υπηρεσία.

Κατά την επιλογή της καταχώρησης της φόρμας, η έξυπνη συσκευή ελέγχει αν υπάρχει ενεργή σύνδεση στο διαδίκτυο, και αν όντως υπάρχει καταχωρεί την φόρμα στη διαδικτυακή υπηρεσία, διαφορετικά ενημερώνει το χρήστη, πως δεν υπάρχει ενεργή σύνδεση, οπότε πρέπει ο χρήστης να ενεργοποιήσει τη σύνδεση στο διαδίκτυο και να καταχωρήσει εκ νέου τη φόρμα.

Με την καταχώρηση των δεδομένων στη διαδικτυακή υπηρεσία, η τελευταία, ελέγχει για την ορθότητα των δεδομένων. Στην περίπτωση αποτυχίας, ενημερώνει το χρήστη, ο οποίος πρέπει να προβεί στις απαραίτητες ενέργειες ώστε να καταχωρηθεί η αγγελία. Σε περίπτωση επιτυχίας, τα δεδομένα καταχωρούνται στη διαδικτυακή πύλη της επιχείρησης, και επιστρέφεται στο χρήστη μια απάντηση με τον αριθμό με τον οποίο έχει καταχωρηθεί η αγγελία, ενώ ταυτόχρονα η αγγελία δημοσιεύεται στη διαδικτυακή πύλη της επιχείρησης.



Εικόνα 5 - Διάγραμμα δραστηριοτήτων

Το διάγραμμα δραστηριοτήτων που παρουσιάστηκε, αναφέρεται μόνο στην καταχώρηση της αγγελίας, χωρίς την προσθήκη φωτογραφιών, καθώς αποτελεί μια διαφορετική λειτουργία, η οποία μπορεί να περιγραφεί με ένα αντίστοιχο διάγραμμα δραστηριοτήτων.

3.3.3 Διάγραμμα τάξεων (Class diagram)

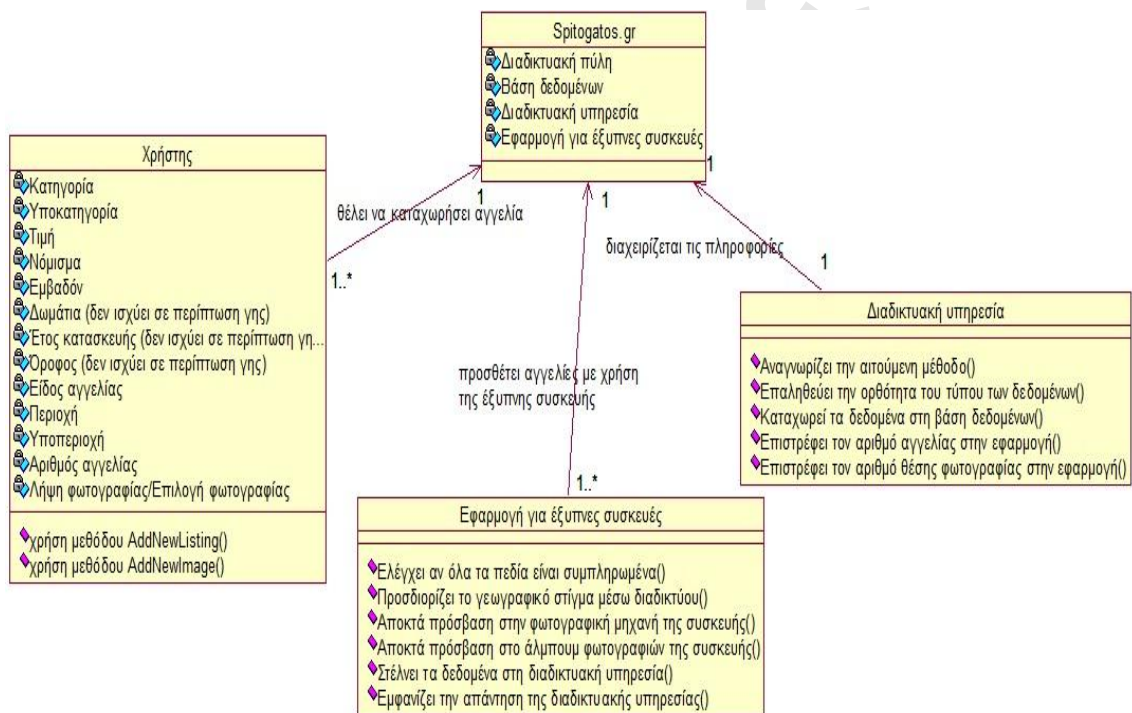
Τα διαγράμματα τάξεων, αναπαριστούν τη στατική δομή όσον αφορά στις τάξεις και στις σχέσεις τους. Για την αναπαράσταση του συγκεκριμένου διαγράμματος, έγινε μια ευρύτερη οριοθέτηση του συστήματος, όπου παρουσιάζονται όλα τα επιμέρους τμήματα της επιχείρησης. Πιο συγκεκριμένα, παρουσιάζεται η κεντρική υπηρεσία του Spitogatos.gr η οποία περιλαμβάνει:

- Τη διαδικτυακή πύλη (βασικό ιστότοπο).
- Τη βάση δεδομένων.
- Τη διαδικτυακή υπηρεσία.
- Τη διάθεση της εφαρμογής για έξυπνες συσκευές προς τους χρήστες.

Επίσης προσδιορίζεται:

- η συμμετοχή του χρήστη, με τα πεδία που καλείται να συμπληρώσει ως «χαρακτηριστικά», και τις «μεθόδους» που θέλει να χρησιμοποιήσει ως «λειτουργίες» προκειμένου να καταχωρήσει την επιθυμητή αγγελία.
- η χρήση της εφαρμογής για τις έξυπνες συσκευές όπου αποτελεί το εργαλείο που χρησιμοποιεί ο χρήστης στη προκειμένη περίπτωση και οι λειτουργίες που αυτή εκτελεί:
 - έλεγχος των πεδίων (αν είναι όλα συμπληρωμένα)
 - προσδιορισμός του γεωγραφικού στίγματος (κυρίως μέσω διαδικτύου)

- πρόσβαση στη φωτογραφική μηχανή της έξυπνης συσκευής
- πρόσβαση στο άλμπουμ φωτογραφιών της έξυπνης συσκευής
- καταχώρηση των δεδομένων στη βάση δεδομένων της επιχείρησης με χρήση της διαδικτυακής υπηρεσίας
- εμφάνιση της απάντησης της διαδικτυακής υπηρεσίας, ως αποτέλεσμα της καταχώρησης των δεδομένων
- η υπηρεσία διαδικτύου η οποία διαχειρίζεται τα δεδομένα και οι λειτουργίες που αυτή εκτελεί:
 - αναγνώριση της καλούμενης μεθόδου
 - επαλήθευση της ορθότητα του τύπου των δεδομένων
 - καταχώρηση των δεδομένων στη βάση δεδομένων
 - επιστροφή απάντησης στην εφαρμογή με τον μοναδικό αριθμό της αγγελίας
 - επιστροφή απάντησης με τον αριθμό θέσης φωτογραφίας



Εικόνα 6 - Διάγραμμα τάξεων

3.3.4 Διάγραμμα συνεργασίας (Collaboration diagram)

Με τα διαγράμματα συνεργασίας, μπορούμε να απεικονίσουμε τα αντικείμενα που αποτελούν το σύστημά μας, καθώς και τις συνδέσεις τους και τις αλληλεπιδράσεις τους (Βίρβου, 2008).

Όπως φαίνεται και στην εικόνα, το σύστημα αποτελείται από τους χρήστες, οι οποίοι επιθυμούν να κάνουν χρήση της εφαρμογής από την έξυπνη συσκευή τους, την εφαρμογή καθαυτή εγκατεστημένη στην «έξυπνη» συσκευή και τις δύο παρεχόμενες υπηρεσίες προς τους χρήστες μέσω της εφαρμογής. Οι παρεχόμενες υπηρεσίες προς τους χρήστες, είναι αφενός η δυνατότητα αιτήματος καταχώρησης νέων αγγελιών, και αφετέρου η δυνατότητα αιτήματος καταχώρησης νέων φωτογραφιών. Τέλος στο σύστημα υπάρχει και η διαδικτυακή πύλη της επιχείρησης, η οποία δέχεται τα αιτήματα των χρηστών και εφόσον αυτά είναι ορθά, δημοσιεύονται στον ιστότοπο της επιχείρησης.

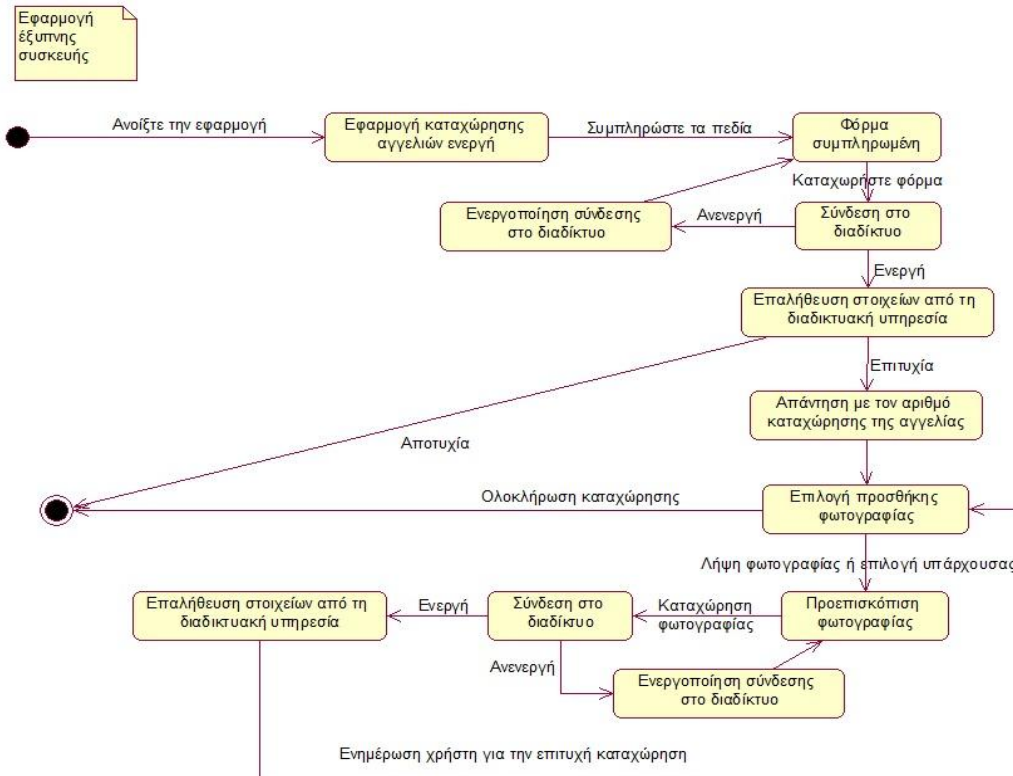


Εικόνα 6 - Διάγραμμα συνεργασίας

3.3.5 Διάγραμμα κατάστασης (Statechart diagram)

Το διάγραμμα κατάστασης, αναπαριστά της συμπεριφορά μιας τάξης όσον αφορά τις καταστάσεις τις (Βίρβου, 2008).

- Γίνεται μετάβαση από την αρχική κατάσταση, στην κατάσταση «Εφαρμογή καταχώρησης αγγελιών ενεργή» ανοίγοντας την εφαρμογή (Ανοίξτε την εφαρμογή).
- Για να γίνει μετάβαση στην κατάσταση «Φόρμα Συμπληρωμένη» πρέπει να έχουν συμπληρωθεί όλα τα απαραίτητα πεδία (Συμπληρώστε τα πεδία).
- Με την καταχώρηση της φόρμας (Καταχωρήστε φόρμα), γίνεται μετάβαση στην κατάσταση όπου ελέγχεται αν υπάρχει ενεργή σύνδεση στο διαδίκτυο.
- Αν η κατάσταση σύνδεσης στο διαδίκτυο αναγνωριστεί ως ενεργή αποστέλλονται τα δεδομένα στη διαδικτυακή υπηρεσία όπου επαληθεύει την εγκυρότητα των δεδομένων, ενώ αν η κατάσταση σύνδεσης στο διαδίκτυο αναγνωριστεί ως ανενεργή, ο χρήστης πρέπει να την ενεργοποιήσει και να αποστείλει εκ νέου τα δεδομένα στην διαδικτυακή υπηρεσία.
- Στην περίπτωση αποτυχίας επαλήθευσης των δεδομένων γίνεται μετάβαση στην κατάσταση τέλους.
- Στην περίπτωση επιτυχίας γίνεται μετάβαση στην κατάσταση όπου εμφανίζεται η απάντηση με τον μοναδικό αριθμό καταχώρησης της αγγελίας η οποία οδηγεί τον χρήστη στη συνέχεια στην κατάσταση επιλογής προσθήκης φωτογραφιών.
- Από την κατάσταση επιλογής προσθήκης φωτογραφιών, μπορεί να γίνει μετάβαση στο τέλος (Ολοκλήρωση καταχώρησης), ή να επιλεγεί να καταχωρηθούν και φωτογραφίες στη συγκεκριμένη αγγελία, είτε τραβώντας νέα φωτογραφία είτε επιλέγοντας υπάρχουσα (Λήψη φωτογραφίας ή επιλογή υπάρχουσας), οπότε γίνεται μετάβαση στην κατάσταση «προεπισκόπησης της φωτογραφίας».
- Με την καταχώρηση της φωτογραφίας, μεταβαίνουμε στην κατάσταση ελέγχου σύνδεσης με το διαδίκτυο.
- Αν η κατάσταση σύνδεσης στο διαδίκτυο αναγνωριστεί ως ενεργή αποστέλλεται η φωτογραφία στη διαδικτυακή υπηρεσία, ενώ αν η κατάσταση σύνδεσης στο διαδίκτυο αναγνωριστεί ως ανενεργή, ο χρήστης πρέπει να την ενεργοποιήσει και να αποστείλει εκ νέου τη φωτογραφία στη διαδικτυακή υπηρεσία.
- Με την ειδοποίηση στον χρήστη ότι έγινε επιτυχής καταχώρηση, μεταβαίνουμε εκ νέου στην κατάσταση επιλογής καταχώρησης φωτογραφίας, από όπου ο χρήστης μπορεί είτε να μεταβεί στο τέλος (Complete Listing) είτε να προσθέσει και νέα φωτογραφία.

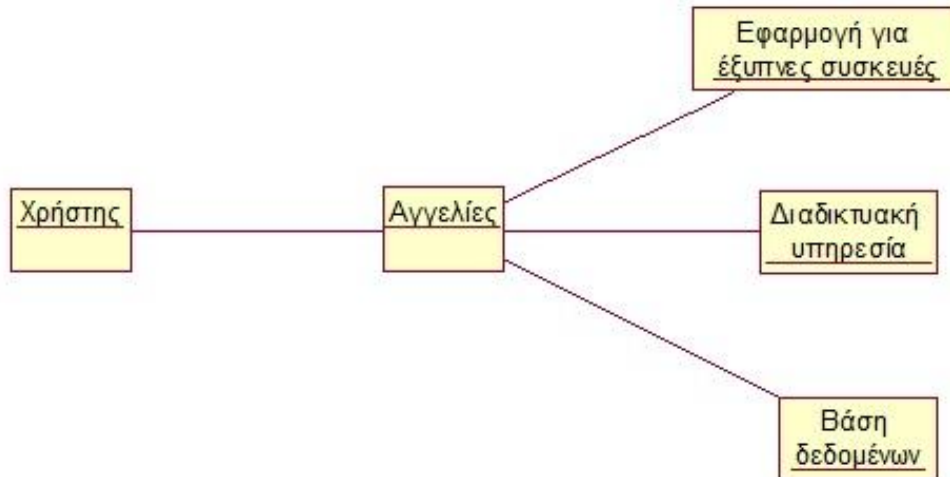


Εικόνα 7 - Διάγραμμα κατάστασης

3.3.6 Διάγραμμα αντικειμένων (Object diagram)

Το διάγραμμα αντικειμένων, αποτελεί μια απλοποιημένη μορφή των διαγραμμάτων συνεργασίας, όπου απεικονίζονται τα αντικείμενα χωρίς ωστόσο να απεικονίζεται η μετάδοση μηνυμάτων μεταξύ τους (Βίρβου, 2008). Στην προκειμένη περίπτωση, υπάρχουν

- ο «χρήστης», ο οποίος αποτελεί τον «τελικό» χρήστη, ο οποίος μέσω της «έξυπνης» συσκευής του έχει τη δυνατότητα να αλληλεπιδράσει με τον ιστότοπο της επιχείρησης,
- οι «αγγελίες», οι οποίες αποτελούν το λόγο για τον οποίο ο χρήστης κάνει χρήση της εφαρμογής, για την καταχώρηση δηλαδή μιας επιθυμητής αγγελίας,
- η «εφαρμογή για τα κινητά τηλέφωνα», η οποία αποτελεί το εργαλείο για την επίτευξη της δυνατότητας καταχώρησης αγγελιών από την πλευρά του χρήστη,
- η διαδικτυακή υπηρεσία της επιχείρησης, την οποία καλεί η εφαρμογή από την «έξυπνη» συσκευή και μέσω της οποίας γίνεται η καταχώρηση της αγγελίας στον ιστότοπο της επιχείρησης,
- η βάση δεδομένων της επιχείρησης, στην οποία καταχωρούνται οι αγγελίες οι οποίες έχει εισάγει ο χρήστης, εφόσον έχει επαληθευτεί η ορθότητα των δεδομένων από την διαδικτυακή υπηρεσία της επιχείρησης.



Εικόνα 8 - Διάγραμμα Αντικειμένων

3.3.7 Διάγραμμα εξαρτημάτων (Component diagram)

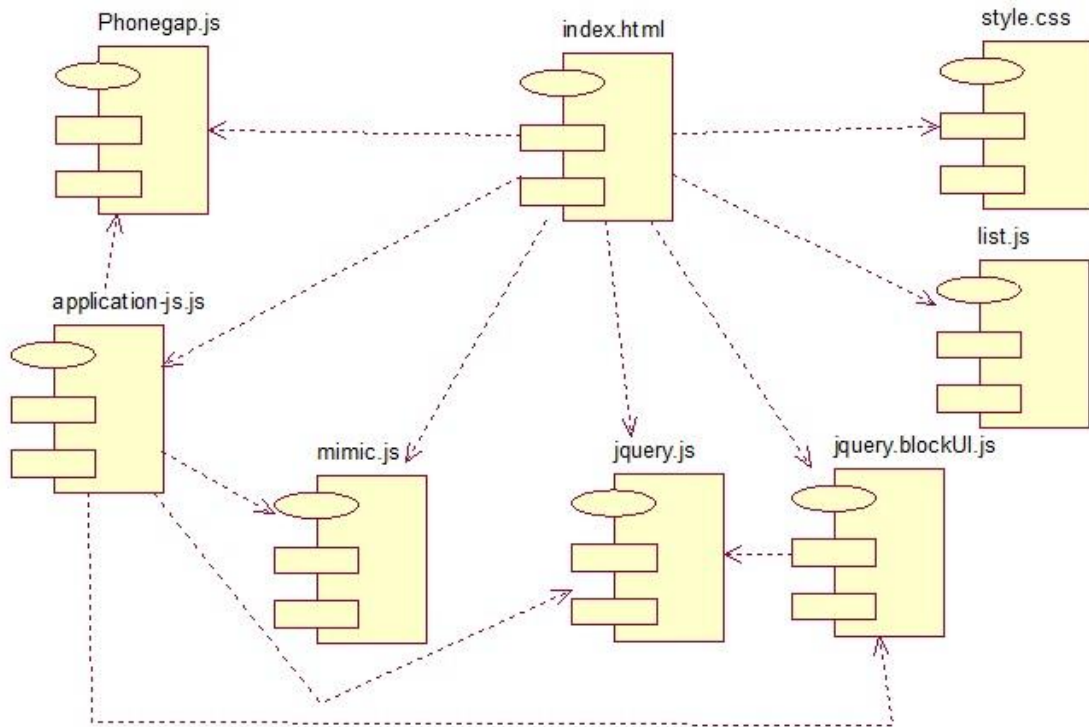
Το διάγραμμα εξαρτημάτων αναπαριστά τα «φυσικά» εξαρτήματα μιας εφαρμογής. Στην προκειμένη περίπτωση ως «φυσικά» εξαρτήματα νοούνται τα αρχεία από τα οποία αποτελείται η εφαρμογή προκειμένου να γίνει λειτουργική, και οι συνδέσεις-αλληλεπιδράσεις που έχουν αυτά μεταξύ τους (Βίρβου, 2008).

Το αρχείο `index.html` αποτελεί το βασικό αρχείο που εμφανίζεται κατά τη φόρτωση της εφαρμογής, και περιλαμβάνει στην επικεφαλίδα του όλα τα αρχεία JavaScript που χρειάζονται προκειμένου να επιτευχθεί η επιθυμητή λειτουργικότητα, και το αρχείο μορφοποίησης CSS.

Το αρχείο `application.js.js` περιέχει το σύνολο των συναρτήσεων, με τις οποίες ανάλογα με τις επιθυμητές λειτουργίες, καλούνται οι ενσωματωμένες βιβλιοθήκες JavaScript προκειμένου να επιτευχθούν οι επιθυμητές λειτουργίες. Πιο συγκεκριμένα καλούνται τα:

- `Phonegap.js` για να γίνει χρήση συγκεκριμένων λειτουργιών του τηλεφώνου, όπως ο προσδιορισμός του γεωγραφικού στίγματος, ή η πρόσβαση στην φωτογραφική μηχανή και στο άλμπουμ φωτογραφιών.
- `Jquery.js` προκειμένου να επιτύχει συγκεκριμένες επιθυμητές λειτουργικότητες αναφορικά με την καθαυτή χρήση της εφαρμογής.
- `Jquery.blockUI.js` προκειμένου να επιτύχει την προσωρινή απενεργοποίηση του περιβάλλοντος διεπαφής του χρήστη.
- `Mimic.js` προκειμένου να γίνει η κλήση προς διαδικτυακή υπηρεσία της επιχείρησης και να μεταφερθούν οι πληροφορίες προς καταχώρηση με τη χρήση του πρωτοκόλλου XML-RPC.

Το αρχείο `jquery.blockUI.js` όντας πρόσθετο, καλεί το αρχείο `jquery.js` προκειμένου να επιτύχει την λειτουργικότητά του η οποία είναι η απενεργοποίηση της επιφάνειας διεπαφής κάτω υπό ορισμένες συνθήκες και πιο συγκεκριμένα κατά τη διαδικασία αναμονής αποστολής του αιτήματος προς την διαδικτυακή υπηρεσία και λήψη της αντίστοιχης απάντησης από αυτή.

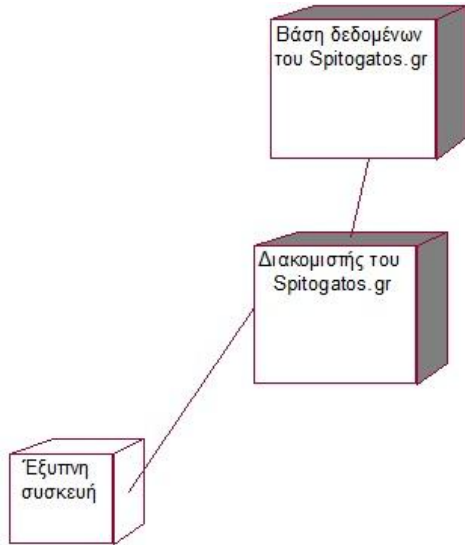


Εικόνα 9 - Διάγραμμα εξαρτημάτων

3.3.8 Διάγραμμα διανομής (Deployment diagram)

Το διάγραμμα διανομής, μας εμφανίζει τη διανομή των εξαρτημάτων σε συγκεκριμένα τμήματα του υλικού (hardware) (Βίρβου, 2008). Στην προκειμένη περίπτωση τα τμήματα του υλικού παρουσιάζονται σε τρία διακριτά επίπεδα, ενώ περαιτέρω ανάλυση είναι δυνατή, ωστόσο ξεφεύγει από τους στόχους της συγκεκριμένης εργασίας.

Πιο συγκεκριμένα, παρουσιάζεται σε ένα επίπεδο η «έξυπνη» συσκευή, την οποία διαθέτει ο χρήστης για να επικοινωνήσει με τον ιστότοπο της επιχείρησης, και αποτελεί απαραίτητο κομμάτι (υλικό) για την επίτευξη του αιτήματος. Τα δύο άλλα επίπεδα που παρουσιάζονται, αφορούν την πλευρά της επιχείρησης, και διακρίνονται στον διακομιστή (server) της επιχείρησης, ο οποίος επικοινωνεί με την έξυπνη συσκευή του χρήστη μέσω της διαδικτυακής υπηρεσίας, ενώ στο επόμενο επίπεδο, υπάρχει η βάση δεδομένων της επιχείρησης, όπου καταχωρούνται τα δεδομένα που στέλνει ο χρήστης, έπειτα από την επαλήθευσή τους από την διαδικτυακή υπηρεσία της επιχείρησης.

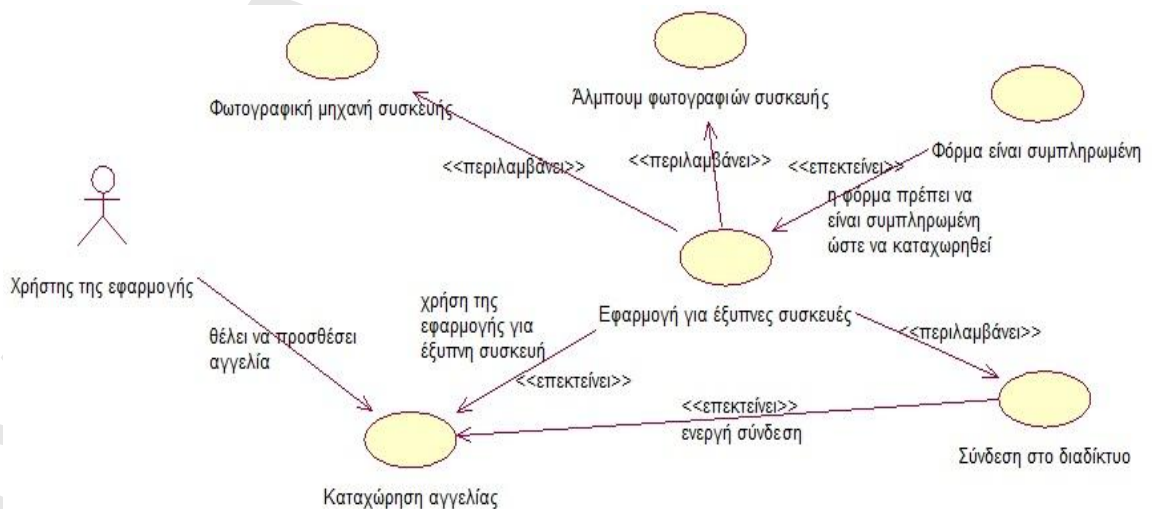


Εικόνα 10 - Διάγραμμα διανομής

3.3.9 Διάγραμμα περιπτώσεων χρήσης (Use Case diagram)

Το διάγραμμα περίπτωσης χρήσης, αναπαριστά τις λειτουργίες ενός συστήματος από την οπτική γωνιά του χρήστη (Βίρβου, 2008).

- Ο χρήστης της εφαρμογής, επιθυμεί να προσθέσει μία νέα αγγελία.
- Για να προστεθεί μία αγγελία είναι απαραίτητη προϋπόθεση να έχει ο χρήστης εγκατεστημένη την εφαρμογή στην έξυπνη συσκευή του.
- Επίσης απαραίτητη προϋπόθεση για την επιτυχή καταχώρηση της αγγελίας είναι η ύπαρξη σύνδεσης στο διαδίκτυο.
- Η εφαρμογή για να επιτελέσει το έργο της, χρησιμοποιεί υποσυστήματα της έξυπνης συσκευής του χρήστη.
 - Φωτογραφική μηχανή της συσκευής
 - Άλμπουμ φωτογραφιών της συσκευής
 - Σύνδεση στο διαδίκτυο
- Προκειμένου η εφαρμογή να αποστείλει τα δεδομένα είναι απαραίτητη προϋπόθεση, όλα τα απαραίτητα πεδία της φόρμας, να είναι συμπληρωμένα.



Εικόνα 11 - Διάγραμμα περιπτώσεων χρήσης

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

3.4 Περιορισμοί – Restrictions

Κατά την ανάπτυξη της εφαρμογής, προέκυψαν διάφοροι παράγοντες, οι οποίοι είχαν ως αποτέλεσμα να δυσχεράνουν την εξέλιξη της ανάπτυξης της εφαρμογής με αποτέλεσμα σε ορισμένες περιπτώσεις να σημειώνονται αρκετές καθυστερήσεις μέχρι την εξεύρεση λύσης, ενώ σε άλλες περιπτώσεις να αναζητούνται εναλλακτικοί τρόποι διαχείρισης του εμφανιζόμενου προβλήματος προκειμένου να συνεχίσει η διαδικασία της ανάπτυξης.

Οι παράγοντες αυτοί αφορούσαν είτε έτοιμα εργαλεία ανάπτυξης (προσομοιωτής εικονικής συσκευής – Android Virtual Device), είτε εν γένει τεχνολογικούς περιορισμούς (θέματα ασφάλειας κατά την εκτέλεση κώδικα ανάμεσα σε απομακρυσμένα εικονικά μέρη – Cross-site Scripting), είτε και παράγοντες που αφορούσαν υπάρχουσες λειτουργίες που εμπλέκονταν στην υλοποίηση της εφαρμογής (ανάγκη επαλήθευσης ονόματος χρήστη και κωδικού πρόσβασης από τη διαδικτυακή υπηρεσία της επιχείρησης).

3.4.1 Προσομοιωτής εικονικής συσκευής – Android Virtual Device Manager

Προκειμένου να ελέγχεται η λειτουργικότητα μιας εφαρμογής, είναι απαραίτητο πολλές φορές να μπορεί να γίνει προσομοίωση πραγματικών συνθηκών, ώστε να δοκιμαστούν οι επιμέρους λειτουργίες και να γίνουν οι απαραίτητες διορθώσεις και επεμβάσεις στον κώδικα της εφαρμογής. Για το λόγο αυτό, διατίθεται μαζί με τα εργαλεία ανάπτυξης του λειτουργικού συστήματος Android, και ένα «πακέτο» όπου μπορεί να εγκατασταθεί στον υπολογιστή του προγραμματιστή ένας διαχειριστής εικονικών «έξυπνων συσκευών», το AVD manager (Android Virtual Device manager).

Ο προγραμματιστής, έχει την ευχέρεια να επιλέξει την έκδοση λειτουργικού συστήματος για την οποία προορίζει την εφαρμογή του, και να δημιουργήσει μια εικονική έξυπνη συσκευή στον υπολογιστή του, όπου μπορεί να εγκαταστήσει την εφαρμογή που αναπτύσσει και να την δοκιμάσει. Η διαδικασία αυτή υπερτερεί από την εγκατάσταση της εφαρμογής σε μία φυσική έξυπνη συσκευή τόσο από άποψη χρόνου καθώς και από την άποψη ότι η διαχείριση της εικονικής συσκευής και η εγκατάσταση της εφαρμογής σε αυτή είναι αρκετά ταχύτερη, αλλά και στο γεγονός ότι παρέχεται στον προγραμματιστή μια οθόνη (κονσόλα) όπου περιγράφονται οι διεργασίες που λαμβάνουν χώρα προκειμένου να γίνει και αποσφαλμάτωση της εφαρμογής.

Ωστόσο στην προκειμένη περίπτωση όπου ο κώδικας δεν ήταν γραμμένος σε γλώσσα προγραμματισμού που ήταν κατεξοχήν προορισμένη για το λειτουργικό σύστημα Android (Java) αλλά σε HTML5 με χρήση JavaScript και CSS3 οι πληροφορίες που παρέχονταν από την κονσόλα καταγραφής, δεν καθιστούσαν συχνά σαφές το που βρισκόταν κάποιο πρόβλημα ενώ αρκετές φορές δε γινόταν καν ο εντοπισμός του, γεγονός που δυσχέραινε τη διαδικασία αποσφαλμάτωσης.

Επίσης, το πακέτο του προσομοιωτή εικονικών συσκευών έχει ορισμένες ελλείψεις, οι οποίες είχαν σαν αποτέλεσμα να γίνεται επιτακτική η ανάγκη μεταφοράς της εφαρμογής σε φυσική συσκευή προκειμένου να γίνει ο απαραίτητος έλεγχος. Πιο συγκεκριμένα, δεν παρέχονταν υποστήριξη χρήσης της κάμερας (σε περίπτωση που διέθετε ο ηλεκτρονικός υπολογιστής), γεγονός που καθιστούσε αδύνατο τον έλεγχο της λειτουργικότητας της συγκεκριμένης παραμέτρου σε περιβάλλον διαφορετικό από εκείνο μιας φυσικής έξυπνης συσκευής.

Η έλλειψη υποστήριξης της συγκεκριμένης λειτουργίας γινόταν ακόμα πιο αισθητή κατά την αρχική ανάπτυξη της εφαρμογής, όπου στην περίπτωση που επιλεγόταν σαν στόχος, μια εικονική συσκευή με λειτουργικό Android 2.2 (το οποίο ήταν αναγκαστικό ώστε να υπάρχει αντιστοιχία με την φυσική συσκευή στην οποία θα προοριζόταν η εφαρμογή) όχι μόνο δεν υποστηριζόταν η χρήση της φωτογραφικής μηχανής, αλλά η εικονική συσκευή παρουσίαζε απροσδόκητα σφάλματα και καταρρεύσεις. Στην πορεία ανάπτυξης της εφαρμογής, όπου κατέστη δυνατή η χρήση εικονικού λειτουργικού Android 2.3.3 (λόγω ύπαρξης αντίστοιχης φυσικής συσκευής) το συγκεκριμένο πρόβλημα μετριάστηκε, καθώς παρόλο που δεν επιτρεπόταν και πάλι η πρόσβαση στην κάμερα του ηλεκτρονικού υπολογιστή, η εικονική συσκευή, παρείχε εναλλακτικά κάποιες φωτογραφίες προκειμένου να γίνουν κάποιιο στοιχειώδεις έλεγχοι.

3.4.2 Τεχνολογικοί περιορισμοί – Cross-Site scripting

Το γεγονός ότι η ανάπτυξη του κώδικα γινόταν σε HTML5 με χρήση JavaScript και CSS, σε συνδυασμό με το γεγονός ότι τα εργαλεία αποσφαλμάτωσης του πακέτου προσομοιωτή εικονικών συσκευών υστερούσαν στον εντοπισμό λαθών κυρίως συντακτικών στην HTML5 και ειδικότερα στην JavaScript, είχε σαν φυσικό επόμενο, να δοκιμάζεται ο κώδικας στα κατεξοχήν εργαλεία που προορίζονται για χρήση HTML, τους περιηγητές.

Για την αποσφαλμάτωση του κώδικα, έγινε χρήση κυρίως των περιηγητών Google Chrome, και Mozilla Firefox, διότι παρέχουν αποτελεσματικά εργαλεία παρακολούθησης των διεργασιών, των απομακρυσμένων κλήσεων καθώς και των απομακρυσμένων απαντήσεων από έναν διακομιστή αλλά και εντοπισμού των σφαλμάτων. Ωστόσο η ευκολία στη χρήση των περιηγητών, σε συνδυασμό με το προβληματικό περιβάλλον αποσφαλμάτωσης του προσομοιωτή εικονικών συσκευών, είχε ως αποτέλεσμα, να γίνει προσπάθεια δοκιμής της πλήρους λειτουργικότητας του κώδικα (σε επίπεδο δυνατότητας απομακρυσμένης αποστολής των δεδομένων και ελέγχου της επιστρεφόμενης απάντησης) μέσω των περιηγητών, καθώς ο προσομοιωτής εικονικών συσκευών δεν εμφάνιζε τα ενδεχόμενα λάθη στον κώδικα.

Ωστόσο, δεν έγινε σωστή εκτίμηση της επίδρασης των περιορισμών απομακρυσμένης επικοινωνίας δύο εικονικών συσκευών για λόγους ασφαλείας. Το γεγονός αυτό είχε αντίκτυπο στην ροή ανάπτυξης του κώδικα με δύο τρόπους. Αφενός χρειάστηκε ένα εύλογο χρονικό διάστημα μέχρι να προσδιοριστεί ακριβώς ότι μεγάλο μέρος των προβλημάτων προέρχονταν από την ενσωματωμένη προστασία που είχαν οι περιηγητές, οι οποίοι απέκλειαν περιπτώσεις εκτέλεσης κώδικα JavaScript ανάμεσα σε απομακρυσμένες εικονικές συσκευές. Αφετέρου, έπρεπε να βρεθεί εναλλακτικός τρόπος χειρισμού του προβλήματος, καθώς ακόμα και η επίτευξη επικοινωνίας μέσω της έξυπνης συσκευής με την διαδικτυακή υπηρεσία, (καθώς το rhonpegar δε θέτει περιορισμούς εκτέλεσης κώδικα JavaScript ανάμεσα σε απομακρυσμένες περιοχές), δε θα μπορούσε να παρέχει πληροφορίες σχετικά με την ανάγκη τροποποίησης των περιεχομένων της απομακρυσμένης κλήσης, ώστε να γίνει προσαρμογή στις πραγματικές απαιτήσεις της εφαρμογής.

Οι ενσωματωμένοι περιορισμοί ασφαλείας, κατέστη δυνατόν να απενεργοποιηθούν στον περιηγητή Google Chrome μέσω της εκτέλεσής του μέσω γραμμών εντολών στο περιβάλλον των Windows (Command Prompt) με χρήση συγκεκριμένων παραμέτρων (--disable-web-security), (McGinnis, 2011).

3.4.3 Συγκεκριμένες απαιτήσεις της διαδικτυακής πύλης

Η ανάπτυξη της εφαρμογής, γινόταν με χρήση της υπηρεσίας ανάπτυξης που έχει ο Spiritogatos.gr (πρόκειται για μια υπηρεσία όπου αναπτύσσονται και δοκιμάζονται όλες οι νέες τεχνολογίες που προορίζονται να ενσωματωθούν στο εμπορικό κομμάτι, προκειμένου να μην επηρεάζεται η εύρυθμη λειτουργία του «επίσημου» ιστότοπου). Παρόλο που οι υπηρεσίες είναι πανομοιότυπες αναφορικά με την λειτουργικότητά τους, η υπηρεσία ανάπτυξης, απαιτεί επαλήθευση των στοιχείων πρόσβασης του χρήστη, προκειμένου να εξασφαλιστεί ότι πρόσβαση σε αυτή θα έχουν μόνο εξουσιοδοτημένοι χρήστες και δει η ομάδα ανάπτυξης της επιχείρησης. Με δεδομένο ότι, το αντικείμενο που πραγματεύεται η διατριβή είχε κατά τα πρώτα στάδια ανάπτυξης ένα σχετικά διερευνητικό χαρακτήρα, αλλά και το γεγονός ότι πολλά από τα διαθέσιμα εργαλεία και τεχνολογίες που ήταν διαθέσιμα για χρησιμοποίηση, παρουσίαζαν διαφορετικές ιδιαιτερότητες, ήταν απαραίτητο να γίνεται αναπροσαρμογή ανά διαστήματα ορισμένων στόχων, σε συνάρτηση πάντα με τα διαθέσιμα εργαλεία ανάπτυξης. Ένα χαρακτηριστικό παράδειγμα, ήταν η παραδοχή ότι θα πρέπει η εφαρμογή να ικανοποιεί την υπάρχουσα δομή των υπηρεσιών, ακόμα και την ιδιαιτερότητα της υπηρεσίας ανάπτυξης όπου απαιτούσε επαλήθευση των στοιχείων του χρήστη.

Ωστόσο, μετά τη μελέτη των διάφορων βιβλιοθηκών και σε συνάρτηση με το γεγονός ότι οι περιορισμοί των αιτημάτων μεταξύ απομακρυσμένων συσκευών (cross-site scripting) προκαλούσαν αρκετά προβλήματα απενεργοποιήθηκε η ανάγκη επαλήθευσης του ονόματος χρήστη και του κωδικού πρόσβασης ειδικά για τη διαδικτυακή υπηρεσία, προκειμένου να γίνει πιο εύκολη και η απομακρυσμένη ανάπτυξη της εφαρμογής.

3.5 Παραδοχές κατά την ανάπτυξη της εφαρμογής

Κατά την ανάπτυξη εφαρμογών συχνά γίνονται ορισμένες παραδοχές, οι οποίες είτε διορθώνονται σε νεώτερες εκδόσεις της εφαρμογής, είτε παραμένουν ως έχουν. Αναφορικά με τη συγκεκριμένη εφαρμογή, έχουν ληφθεί ορισμένες παραδοχές, τόσο σχετικά με τις υποστηριζόμενες τεχνολογίες, όσο και με το περιεχόμενο της εφαρμογής.

3.5.1 Υποστηριζόμενες τεχνολογίες

Σημαντική παράμετρος κατά την ανάπτυξη του κώδικα, και θεμελιώδες χαρακτηριστικό του Phonegap, είναι η δυνατότητα να μπορεί να χρησιμοποιείται ο ίδιος πηγαίος κώδικας, σε διαφορετικές εκδόσεις λειτουργικών συστημάτων. Αυτό είχε σαν αποτέλεσμα, να μην ενσωματωθούν ορισμένες λειτουργίες οι οποίες για παράδειγμα θα είχαν συγκεκριμένα οφέλη στο περιβάλλον του λειτουργικού συστήματος Android, αλλά δε θα ήταν συμβατές με άλλα λειτουργικά συστήματα, και αντίστροφα.

- Γεωγραφικό στίγμα. Για τον προσδιορισμό του γεωγραφικού στίγματος, υπάρχουν διάφορες πηγές όπου μπορούν να αντληθούν πληροφορίες, όπως το ενσωματωμένο, σε αρκετές έξυπνες εφαρμογές, GPS (Global Positioning System), αλλά και διαδικτυακά εργαλεία, όπως η διεύθυνση IP, οι διευθύνσεις RFID, WiFi και Bluetooth MAC, καθώς και οι IDs των κελιών GSM/CDMA. Οι περισσότερες συσκευές εφαρμόζουν τη χρήση αυτών των τεχνολογιών για τον προσδιορισμό του γεωγραφικού στίγματος, ενώ όποιες δεν υποστηρίζουν αυτή την τεχνολογία, το API (Application Programming Interface) του Phonegap, το οποίο συμβαδίζει με τα χαρακτηριστικά που έχουν οριστεί από το W3C, εφαρμόζεται για τον προσδιορισμό του στίγματος (Nitobi, 2011). Αυτό έχει σαν αποτέλεσμα όμως ο τρόπος προσδιορισμού, να επαφίεται στον τρόπο λειτουργίας της έξυπνης συσκευής και όχι στην επιθυμία του χρήστη. Αν για παράδειγμα θέλει, να χρησιμοποιηθεί μόνο ο αισθητήρας GPS προκειμένου να έχει ακριβέστερο σήμα, χρειάζεται να ενσωματωθεί κώδικας, ο οποίος διαφέρει κατά περίπτωση λειτουργικού συστήματος και έξυπνης συσκευής, ενώ σε μερικές δεν υπάρχει καν. Για το λόγο αυτό, έχει εξορισμού, επιλεγεί κατά την αποστολή των δεδομένων στη διαδικτυακή υπηρεσία της επιχείρησης, η ακρίβεια των δεδομένων να θεωρείται σε «επίπεδο γειτονιάς».
- Οθόνη αναμονής/επεξεργασίας αιτήματος. Το Phonegap, στην πρώτη του σταθερή έκδοση, χρησιμοποιούσε τις ενσωματωμένες λειτουργίες των συσκευών, προκειμένου να ενημερώνει το χρήστη, ότι κάποια διεργασία είναι υπό επεξεργασία. Ωστόσο, αυτή η λειτουργία εγκαταλείφθηκε στις νεώτερες εκδόσεις, δίνοντας τη δυνατότητα στο χρήστη, να χρησιμοποιήσει συγκεκριμένα κομμάτια κώδικα ανάλογα με τα εκάστοτε λειτουργικά συστήματα. Επειδή αυτό το ενδεχόμενο δεν ήταν επιθυμητό στην προκειμένη περίπτωση, ενσωματώθηκε η συγκεκριμένη λειτουργία με χρήση JavaScript.
- Σύνδεση στο διαδίκτυο, Internet. Για τη χρήση της εφαρμογής, και πιο συγκεκριμένα για την αποστολή των αιτημάτων προς τη διαδικτυακή υπηρεσία, είναι απαραίτητη η ύπαρξη ενεργής σύνδεσης στο διαδίκτυο. Η εφαρμογή, δεν «αναλαμβάνει» να ενεργοποιήσει τη σύνδεση στο διαδίκτυο σε περίπτωση που αυτή είναι ανενεργή, για το λόγο αυτό, θα πρέπει να έχει μεριμνήσει ο χρήστης για την ενεργοποίησή της.
- Φωτογραφίες. Όπως και με τις υπόλοιπες λειτουργίες, έτσι και με τις φωτογραφίες, οι εκάστοτε συσκευές προσφέρουν κάποιες επιπλέον λειτουργίες, οι οποίες μπορεί να μην υποστηρίζονται από άλλες συσκευές/λειτουργικά συστήματα (για παράδειγμα, το λειτουργικό σύστημα IOS επιτρέπει ειδική λειτουργία λήψης επεξεργάσιμης φωτογραφίας). Για τον λόγο αυτό, χρησιμοποιήθηκαν μόνο οι επιλογές εκείνες, οι οποίες υποστηρίζονται σχεδόν από τα περισσότερα λειτουργικά συστήματα, και αφορούν την πρόσβαση στη φωτογραφική μηχανή της συσκευής αλλά και στο άλμπουμ των φωτογραφιών.
- Έξοδος από την εφαρμογή. Τα διάφορα λειτουργικά συστήματα, έχουν διαφορετικούς τρόπους διαχείρισης των ενεργών εφαρμογών, τερματίζοντας τις μη ενεργές, όταν αυτό «κρίνεται» αναγκαίο, ενώ σε άλλα επιτρέπεται ο τερματισμός της εφαρμογής. Η εφαρμογή δεν προσφέρει πλήκτρο εξόδου, και στην περίπτωση που η εφαρμογή δεν τερματίζεται από το λειτουργικό σύστημα, θα πρέπει ο χρήστης να την τερματίσει από τη διαχείριση εργασιών της συσκευής.

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

3.5.2 Ρυθμίσεις και περιεχόμενο της εφαρμογής

Αντίστοιχα με τις υποστηριζόμενες τεχνολογίες έχουν ληφθεί και ορισμένες παραδοχές αναφορικά με τη διάθεση της εφαρμογής στους χρήστες, αλλά και τις δυνατότητες που αυτή παρέχει.

- Παραμετροποίηση της εφαρμογής. Η εφαρμογή κατά την ανάπτυξή της, περιέχει της προσωπικές πληροφορίες, που αφορούν συγκεκριμένο χρήστη (όνομα χρήστη, κωδικός (password), κλειδί εφαρμογής (application key)), ενσωματωμένες στον πηγαίο κώδικα. Έχει γίνει η παραδοχή, ότι θα διαμοιράζεται εκ των προτέρων «ρυθμισμένη» και «παραμετροποιημένη», ανάλογα με τα στοιχεία πρόσβασης του εκάστοτε χρήστη.
- Διάθεση της εφαρμογής. Η εφαρμογή, έχει σχεδιαστεί με στόχο να εξυπηρετεί τις ανάγκες που έχουν οι μεσίτες για άμεση καταχώρηση αγγελιών από τις έξυπνες συσκευές τους. Ωστόσο, στην τρέχουσα έκδοσή της, υπάρχουν κάποιες παραδοχές οι οποίες καθιστούν αδύνατη τη διάθεσή της μέσω των «αγορών» που διαθέτουν τα λειτουργικά συστήματα (Play Store (πρώην Android Market), Apple Store κλπ). Η εφαρμογή στην παρούσα φάση, έχει καταχωρημένα εξαρχής τα προσωπικά στοιχεία του χρήστη που αντιστοιχούν στο λογαριασμό του στην επιχείρηση και δεν παρέχει επιλογή στον χρήστη που θα την «κατεβάσει» από κάποιο online κατάστημα, να εισάγει τα προσωπικά του στοιχεία. Με δεδομένο ότι χρειάζονται να έχουν καταχωρηθεί συγκεκριμένα προσωπικά στοιχεία στην εφαρμογή αναφορικά με τον συγκεκριμένο χρήστη, η διάθεσή της από τα εν λόγω σημεία, δεν έχει νόημα.
- Μη εξουσιοδοτημένη χρήση. Η εφαρμογή δεν έχει κάποιου είδους περιορισμό σχετικά με την αποτροπή μη εξουσιοδοτημένης χρήσης. Δεν απαιτεί δηλαδή κατά την εκκίνησή της κάθε φορά την εισαγωγή ενός συγκεκριμένου «συνθηματικού», το οποίο θα εξασφαλίζει ότι χρησιμοποιείται από τον εξουσιοδοτημένο χρήστη.
- Πεδία καταχώρησης. Μια επιπλέον παραδοχή που έγινε, αναφορικά με τις διαθέσιμες πληροφορίες προς καταχώρηση, ήταν, να είναι δυνατή μόνο η καταχώρηση των απολύτως απαραίτητων πληροφοριών που είναι στοιχειώδεις από πλευρά απαιτήσεων εκ μέρους του Spitogatos.gr, για την καταχώρηση μιας νέας αγγελίας.

Κεφάλαιο 4

Συμπεράσματα

4.1 Ανακεφαλαίωση

Η συγκεκριμένη διατριβή, πραγματοποιήθηκε την ανάπτυξη μιας εφαρμογής για έξυπνες συσκευές, με χρήση της πλατφόρμας Phonegap, η οποία αποτελεί ένα αρκετά νέο εργαλείο για τους προγραμματιστές προκειμένου να αναπτύξουν τις εφαρμογές τους. Ο κώδικας που αναπτύχθηκε με χρήση των γλωσσών προγραμματισμού HTML5, JavaScript, και CSS3, διαμορφώθηκε κατά τέτοιο τρόπο, ώστε να είναι δυνατή η δημιουργία εκτελέσιμου αρχείου, το οποίο να μπορεί να εγκατασταθεί άμεσα στην αντίστοιχη έξυπνη συσκευή με το συγκεκριμένο λειτουργικό σύστημα, χωρίς να απαιτούνται επιπλέον παρεμβάσεις στον κώδικα. Έγινε δηλαδή προσπάθεια, ο κώδικας, να είναι συμβατός με τα σημαντικότερα λειτουργικά συστήματα τα οποία συναντώνται στις έξυπνες συσκευές και πρόκειται για τα IOS, Android, Windows Mobile Blackberry και άλλα.

Η ανάπτυξη της εφαρμογής έγινε σε πραγματικές συνθήκες και σε ένα πλήρως λειτουργικό σύστημα και δε βασίστηκε σε μία εξιδανικευμένη διαδικτυακή υπηρεσία. Αυτό είχε ως αποτέλεσμα να δημιουργηθούν νέες προκλήσεις σχετικά με την υλοποίηση καθώς έπρεπε το σενάριο που πραγματοποιείται η παρούσα διατριβή να προσαρμοστεί σε πραγματικές συνθήκες και απαιτήσεις.

Αποτέλεσμα αυτής της διαδικασίας, ήταν να προκύψει μία εφαρμογή, η οποία είναι πλήρως λειτουργική και ανταποκρίνεται σε πραγματικές συνθήκες. Η εφαρμογή, μπορεί να καταχωρήσει σε πραγματικό χρόνο μια νέα αγγελία στη βάση δεδομένων που συνδέεται η διαδικτυακή υπηρεσία καθώς και φωτογραφίες που αφορούν τη συγκεκριμένη αγγελία μετά την καταχώρηση τα δεδομένα είναι άμεσα προσβάσιμα στον τελικό χρήστη.

4.2 Περιορισμοί στην υλοποίηση

Κατά την υλοποίηση της εφαρμογής, υπήρξαν διάφοροι παράγοντες οι οποίοι λειτούργησαν ανασταλτικά αναφορικά με την πορεία και το ρυθμό ανάπτυξης της εφαρμογής.

Ένας από τους σημαντικότερους ανασταλτικούς παράγοντες, αποτέλεσε το γεγονός, ότι μέχρι να βρεθεί τρόπος να παρακαμφθούν οι περιορισμοί ασφαλείας που έχουν ενσωματωμένους οι περιηγητές, ο μόνος τρόπος για να γίνονται δοκιμές στον κώδικα της εφαρμογής και ειδικότερα να επιτυγχάνεται αποτελεσματικότερη αποσφαλμάτωση, ήταν με φυσική παρουσία στις εγκαταστάσεις τις επιχείρησης προκειμένου να μην «εγείρονται» θέματα ασφαλείας «cross-site scripting» καθώς οι δοκιμές στα γραφεία της επιχείρησης ήταν δυνατό να γίνονται τοπικά. Το συγκεκριμένο «εμπόδιο» προσπεράστηκε, όταν βρέθηκε αποτελεσματικός τρόπος απενεργοποίησης της ενσωματωμένης διαδικτυακής ασφαλείας στον περιηγητή Google Chrome (McGinnis, 2011).

Ένας άλλος περιορισμός αναφορικά με την υλοποίηση αφορούσε το κομμάτι της διεπαφής με το χρήστη. Κύριο μέλημα κατά την ανάπτυξη της εφαρμογής, ήταν να επιτευχθεί ο αρχικός στόχος της λειτουργικότητάς με τα συγκεκριμένα δεδομένα αναφορικά με τη χρήση των διαθέσιμων τεχνολογιών, αλλά και τις απαραίτητες προσαρμογές και τροποποιήσεις που έπρεπε να γίνουν, προκειμένου η εφαρμογή να συνεργάζεται άψογα με την υπάρχουσα διαδικτυακή υπηρεσία. Αυτό είχε σαν αποτέλεσμα, η ανάπτυξη της διεπαφής με το χρήστη, να μην αποτελέσει αντικείμενο προς μελέτη κατά τη διάρκεια εκπόνησης της διατριβής. Για το λόγο αυτό, το γραφικό περιβάλλον διεπαφής με το χρήστη, μπορεί να χαρακτηριστεί ως λιτό και στοιχειώδες αναφορικά με τις απαιτήσεις. Αποτελεί ένα πεδίο το οποίο επιδέχεται πολλών βελτιώσεων και μελέτης προκειμένου να ενσωματωθούν αποτελεσματικότερα διαθέσιμες τεχνολογίες που μπορεί να βρίσκουν εφαρμογή σε αυτήν την περίπτωση χρήσης.

Μια άλλη περίπτωση περιορισμού, αφορά στην επιστροφή συγκεκριμένων μηνυμάτων λαθών, στο χρήστη, στην περίπτωση που αποτύχει η καταχώρηση της αγγελίας ή της φωτογραφίας μέσω της διαδικτυακής υπηρεσίας της επιχείρησης. Η διαδικτυακή υπηρεσία της

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

επιχείρησης επιστρέφει μηνύματα σφαλμάτων αναλόγως με την περίπτωση στην οποία αυτά εμπίπτουν. Ωστόσο, κατά την αποστολή και την παραλαβή μηνυμάτων XML υπάρχει και μια κεφαλίδα (header) η οποία «συνοδεύει» τα μηνύματα, και περιέχει πληροφορίες για τα μηνύματα αυτά προκειμένου να γίνουν διαχειρίσιμα είτε από την πλευρά της έξυπνης συσκευής είτε από την πλευρά της διαδικτυακής υπηρεσίας. Στην περίπτωση των μηνυμάτων σφάλματος, διαπιστώθηκε πως η διαδικτυακή υπηρεσία της επιχείρησης, ενώ επέστρεφε μηνύματα σε XML μορφή, στην κεφαλίδα που τα «συνόδευε» όριζε τα μηνύματα αυτά ως «text/html». Αντίθετα, στις επιτυχείς απαντήσεις, η κεφαλίδα που συνοδεύει τα μηνύματα τα ορίζει ως «text/xml». Αυτό έχει σαν αποτέλεσμα, να μην μπορούν να γίνουν διαχειρίσιμα από τη βιβλιοθήκη `mimic.js` τα μηνύματα λάθους, καθώς δεν αναγνωρίζονται ως συμβατή μορφή μηνύματος με αυτήν που θα ανέμενε η βιβλιοθήκη. Για την αντιμετώπιση αυτής της ιδιαιτερότητας, και προκειμένου να περιοριστούν ενδεχόμενες απροσδόκητες καταστάσεις από τη χρήση της εφαρμογής, χρησιμοποιήθηκε μια λειτουργία της JavaScript και ποιο συγκεκριμένα μια πρόταση, η «`try..catch`». Ο χειρισμός αυτός επιτρέπει την ανίχνευση κάποιου προβλήματος κατά την εκτέλεση του κώδικα, και την ενημέρωση του χρήστη ότι υπήρξε κάποιο πρόβλημα το οποίο δεν μπορεί να προσδιοριστεί. Ωστόσο το ιδανικό σενάριο, θα ήταν να μπορεί να γίνει επεξεργασία από τη βιβλιοθήκη και του μηνύματος λάθους, ώστε να γνωρίζει ακριβώς ο χρήστης το είδος του προβλήματος που προέκυψε.

Τέλος, προκειμένου να επιτευχθεί ο στόχος να υπάρχει δηλαδή δυνατότητα να χρησιμοποιείται αυτούσιος ο κώδικας στα περισσότερα λειτουργικά συστήματα, χωρίς παραλλαγές, προέκυπταν ορισμένοι περιορισμοί και παραδοχές κατά την ανάπτυξη του κώδικα που δεν επέτρεπαν να γίνει ειδική μεταχείριση ορισμένων ζητημάτων και σφαλμάτων που αφορούσαν συγκεκριμένα λειτουργικά συστήματα, όπως στην προκειμένη περίπτωση το λειτουργικό σύστημα Android στην έκδοση 2.3.

Ένα χαρακτηριστικό παράδειγμα, αποτελεί το κομμάτι ελέγχου ύπαρξης ενεργής σύνδεσης στο διαδίκτυο που πραγματοποιεί η εφαρμογή με τη βοήθεια του `PhoneGap`. Παρατηρήθηκε, και στη συνέχεια επιβεβαιώθηκε από παρόμοιες αναφορές στο διαδίκτυο (Anon., 2010) ότι στο Android παρουσιάζεται ένα σφάλμα (bug) που σχετίζεται με την αναγνώριση ύπαρξης διαθέσιμης ενεργής σύνδεσης στο διαδίκτυο κάτω από συγκεκριμένες συνθήκες. Πιο συγκεκριμένα, αν τη στιγμή που εκτελείται η εφαρμογή, δεν υπάρχει ενεργή σύνδεση στο διαδίκτυο, κατά τη διάρκεια της επιλογής καταχώρησης της φόρμας, θα ενημερωθεί ο χρήστης ότι δεν υπάρχει ενεργή σύνδεση στο διαδίκτυο. Ωστόσο, παρατηρείται το εξής «παράδοξο»:

- Αν χωρίς να κλείσει την εφαρμογή, μεταβεί στην ενεργοποίηση της σύνδεσης και επιλέξει να συνδεθεί μέσω WiFi σε ένα διαθέσιμο ασύρματο δίκτυο, και πραγματοποιηθεί η σύνδεση, τότε κατά την επιστροφή στην εφαρμογή, και την επιλογή της καταχώρησης της φόρμας αναγνωρίζεται κανονικά από το σύστημα ότι υπάρχει ενεργή σύνδεση και συγκεκριμένα WiFi σύνδεση.
- Αν όμως, ακολουθήσει την ίδια διαδικασία, με την διαφορά πως αντί για ασύρματη WiFi σύνδεση, επιλέξει να συνδεθεί με το δίκτυο κινητής τηλεφωνίας «3G» τότε, κατά την επιστροφή στην εφαρμογή, θα διαπιστώσει, ότι κατά την καταχώρηση της φόρμας, δεν αναγνωρίζεται ως έγκυρος τύπος σύνδεσης η σύνδεση «3G».

Αντίθετα, αν ενεργοποιήσει την σύνδεση μέσω δικτύου κινητής τηλεφωνίας 3G και στην συνέχεια «ανοίξει» την εφαρμογή τότε, διαπιστώνει ο χρήστης, ότι ενημερώνεται κανονικά για τον τύπο της σύνδεσης το διαδίκτυο, ανεξάρτητα από το ποιος είναι αυτός (ο τύπος σύνδεσης) (Anon., 2010).

Ορισμένοι από τους περιορισμούς στην υλοποίηση που παρατηρήθηκαν, πρόκειται για καταστάσεις οι οποίες πρέπει να προσαρμόσει ο προγραμματιστής τον τρόπο εργασίας του προκειμένου να τους αντιμετωπίσει, όπως για παράδειγμα, το «`cross-site scripting`» στη χρήση περιηγητών, καθώς αποτελεί σημαντικό θέμα ασφάλειας. Ωστόσο, άλλοι από τους περιορισμούς που έχουν επισημανθεί, όπως η δυσχρησία των προσομοιωτών αλλά και ορισμένα εγγενή προβλήματα (bugs) των έξυπνων συσκευών, πρόκειται για καταστάσεις, τις οποίες ενδεχομένως να μην αντιμετωπίσει κάποιος προγραμματιστής στο μέλλον, με δεδομένο ότι πολλά από αυτά τα καταγεγραμμένα θέματα, φροντίζουν σταδιακά, να τα επιλύουν οι εταιρείες ανάπτυξης.

4.3 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η ανάπτυξη εφαρμογών, κατά κανόνα συνεπάγεται την υποστήριξη των χρηστών σχετικά με τη χρήση της εκάστοτε εφαρμογής, αλλά και την προσπάθεια προσθήκης βελτιώσεων και επεκτάσεων οι οποίες καθιστούν την εφαρμογή ακόμα πιο λειτουργική αλλά και σύγχρονη με τις επιταγές της τεχνολογίας και τις ανάγκες των χρηστών.

Η εφαρμογή η οποία παρουσιάζεται με την παρούσα διατριβή, έγινε προσπάθεια να είναι όσο το δυνατόν πιο πλήρης και λειτουργική σε σχέση με την ανάλυση απαιτήσεων που ορίστηκε αρχικά. Στα πλαίσια αυτά μπορεί η πρώτη σταθερή έκδοση της εφαρμογής, να αποτελέσει τη βάση για περαιτέρω ανάπτυξη και βελτίωση σε διάφορους τομείς, τόσο αναφορικά με τις διαθέσιμες λειτουργίες προς τον χρήστη, όσο και ως προς την εμπειρία που αφήνει κατά τη χρήση της, αλλά και με τις τεχνολογίες που ενσωματώνει.

4.4 Διαθέσιμες Λειτουργίες

Η εφαρμογή, στην τρέχουσα, πρώτη, σταθερή έκδοση, δίνει στο χρήστη τη δυνατότητα να εισάγει μια νέα αγγελία μέσω της διαδικτυακής υπηρεσίας της επιχείρησης καθώς και να εισάγει φωτογραφίες για τη συγκεκριμένη αγγελία κατά το χρόνο καταχώρησής της.

Αναφορικά με τις ήδη υποστηριζόμενες λειτουργίες, θα μπορούσε να γίνει μία επέκταση αυτών, προσθέτοντας μερικές επιπρόσθετες επιλογές και δυνατότητες στο χρήστη, ενώ προκειμένου, ο χρήστης να έχει όλες τις διαθέσιμες επιλογές που μπορεί να υποστηρίξει η επιχείρηση, θα ήταν δυνατό σε μεταγενέστερη έκδοση να προστεθούν και άλλες λειτουργίες, όπως:

- Στην καταχώρηση νέας αγγελίας, στην τρέχουσα έκδοση, γίνεται η παραδοχή, ότι ο χρήστης θα εισάγει μόνο τις απολύτως απαραίτητες πληροφορίες που αφορούν κάποιο ακίνητο. Η λειτουργία αυτή θα μπορούσε να επεκταθεί, επιτρέποντας στο χρήστη να εισάγει το σύνολο των διαθέσιμων πληροφοριών που υποστηρίζει η φόρμα της επιχείρησης, εάν αυτός το επιθυμεί, ενημερώνοντας τον ταυτόχρονα, για το ποια πεδία είναι υποχρεωτικά και ποια προαιρετικά.
- Επιπλέον, προκειμένου η εφαρμογή να είναι ακόμα πιο προσιτή στο χρήστη, θα μπορούσε η καταχώρηση μιας αγγελίας, να διαχωρίζεται σε «πλήρης» και «τυπική». Με αυτόν τον τρόπο, ο χρήστης θα επέλεγε από την αρχή, το είδος της καταχώρησης που επιθυμεί με αποτέλεσμα να του εμφανίζεται η κατάλληλη φόρμα συμπλήρωσης με τα αντίστοιχα πεδία, ανάλογα με την επιλογή.
- Στην καταχώρηση φωτογραφιών, θα ήταν χρήσιμο, να δίνεται η δυνατότητα στο χρήστη να εισάγει και ο ίδιος το μοναδικό αριθμό μιας αγγελίας, σε περίπτωση που εκτός από την άμεση εισαγωγή φωτογραφιών κατά την καταχώρηση μιας αγγελίας, θέλει να εισάγει φωτογραφίες σε κάποια αγγελία που είχε καταχωρηθεί σε προγενέστερο χρόνο.
- Προκειμένου να επιτευχθεί αυτό, και θεωρώντας ότι ένας χρήστης είναι πρακτικά απίθανο να απομνημονεύει τους μοναδικούς αριθμούς κάθε αγγελίας που έχει αναρτήσει, θα μπορούσε η εφαρμογή να προσφέρει τη δυνατότητα ανάκτησης των αγγελιών που έχει καταχωρήσει ο χρήστης.
- Σε περίπτωση που οι αγγελίες είναι πολλές, θα μπορούσε να προστεθεί και η δυνατότητα αναζήτησης αγγελίας, έτσι ώστε να μη χρειάζεται να ελέγχει όλες τις αγγελίες μία προς μία ο χρήστης, αλλά να μπορεί να ανατρέχει γρηγορότερα στην επιθυμητή.
- Η ενσωμάτωση αυτών των λειτουργιών, θα είχε ως αποτέλεσμα, να είναι πλέον πιο εύκολη και η ενσωμάτωση ειδικότερων λειτουργιών που αφορούν τη διαχείριση των φωτογραφιών, όπως για παράδειγμα το να προσφέρεται μια προεπισκόπηση του συνόλου των φωτογραφιών που έχουν καταχωρηθεί σε για ένα ακίνητο. Επίσης, εκτός από την προεπισκόπηση, ο χρήστης ενδέχεται να επιτρέπεται να διαγράφει τις φωτογραφίες που δεν επιθυμεί, απευθείας μέσω της εφαρμογής.

- Τέλος θα μπορούσε να δίνεται η δυνατότητα στο χρήστη ακόμα και να διαγράφει εξολοκλήρου κάποια αγγελία, αν το επιθυμεί, απευθείας μέσω της έξυπνης συσκευής του.

Οι λειτουργίες που περιγράφηκαν, και οι οποίες θα μπορούσαν να προσδώσουν επιπλέον χρηστικότητα στην εφαρμογή καθώς ο χρήστης θα μπορεί να διαχειριστεί το σύνολο του λογαριασμού του μέσω της εφαρμογής, αποτελούν ουσιαστικά τις διαθέσιμες μεθόδους που υποστηρίζει η διαδικτυακή υπηρεσία της επιχείρησης.

4.5 Διεπαφή με το χρήστη

Η εφαρμογή δεν αρκεί να ενσωματώνει μόνο πολλές λειτουργίες, αλλά θα πρέπει να το κάνει αυτό, προσφέροντας ένα περιβάλλον διεπαφής με τον χρήστη, εύκολο, απλό, προσίτο και κατανοητό. Ο λόγος που τονίζεται αυτό είναι ότι υπάρχει ο κίνδυνος η διάθεση στον χρήστη επιπλέον λειτουργιών να προκαλέσει σύγχυση, εάν αυτές δεν είναι διαρθρωμένες σωστά.

Εφόσον ο χρήστης θα μπορεί να έχει στη διάθεσή του πολλές περισσότερες επιλογές και λειτουργίες, θα ήταν ενδεχομένως χρήσιμο, να υπάρχει μια κεντρική οθόνη της εφαρμογής από όπου ο χρήστης θα επέλεγε την επιθυμητή λειτουργία, και θα άνοιγε η κατάλληλη οθόνη διεπαφής που θα υποστήριζε όλα τα απαραίτητα στοιχεία για την επιλεγμένη λειτουργία.

Επίσης θα πρέπει να μελετηθούν διάφορα χαρακτηριστικά τα οποία έπειτα από μελέτες θεωρούνται επιθυμητά αναφορικά με την ύπαρξή τους, σε κάποια εφαρμογή και τα οποία θα ορίζουν αν υπάρχουν κάποια ιδανικά πρότυπα αναφορικά με:

- τα χρησιμοποιούμενα χρώματα,
- το μέγεθος των κουμπιών,
- το μέγεθος της γραμματοσειράς
- το είδος της γραμματοσειράς,
- κ.α.

Επιπλέον, η δυνατότητα επιλογής και εναλλαγής στη γλώσσα που υποστηρίζει η εφαρμογή, θα την καθιστούσε ακόμα περισσότερο χρηστική.

4.6 Ενσωματωμένες τεχνολογίες

Εκτός από τις διαθέσιμες λειτουργίες και τη βελτίωση τις διεπαφής με το χρήστη, συνεχώς θα προστίθενται και νέες δυνατότητες οι οποίες θα είναι σε συνάρτηση με τη διαθέσιμη τεχνολογία αλλά και την τεχνογνωσία που αποκτάται με το πέρασμα του χρόνου. Φυσικά, είναι πολύ πιθανό η ενσωμάτωση νέων τεχνολογιών να φέρει και τις αντίστοιχες αλλαγές σε διαθέσιμες λειτουργίες αλλά και χαρακτηριστικά διεπαφής με το χρήστη, το οποίο είναι θεμιτό.

Μερικές από τις τεχνολογίες που θα μπορούσαν να ενσωματωθούν στην εφαρμογή είναι:

- κατά την εγκατάσταση και την πρώτη εκτέλεση της εφαρμογής, θα μπορούσε να ζητείται από το χρήστη, να εισάγει τα προσωπικά στοιχεία που του έχει διαθέσει ο Spitogatos.gr και τα οποία είναι το όνομα χρήστη, ο κωδικός πρόσβασης, και το κλειδί της εφαρμογής. Κατά την εισαγωγή αυτών των στοιχείων, **και μόνο κατά την πρώτη εκτέλεση της εφαρμογής**, θα μπορούσε η εφαρμογή να τα αποθηκεύσει και να χρησιμοποιεί αυτόματα για οποιαδήποτε λειτουργία επιθυμούσε ο χρήστης να εκτελέσει. Με αυτόν τον τρόπο, η εφαρμογή θα ήταν δυνατό να διατίθεται σε μαζικότερο βαθμό, ενδεχομένως και μέσω των «καταστημάτων» που διαθέτουν τα εκάστοτε λειτουργικά συστήματα, χωρίς να υπάρχει η ανάγκη, να τη διαθέτει η ομάδα ανάπτυξης του Spitogatos.gr ήδη ρυθμισμένη ειδικά για τον κάθε χρήστη.
- Σε κάθε εκκίνηση της εφαρμογής, να απαιτείται η εισαγωγή κάποιου συνθηματικού που θα επιλέγει ο χρήστης, προκειμένου να πιστοποιείται ότι πράγματι γίνεται χρήση της εφαρμογής από πιστοποιημένο και εξουσιοδοτημένο χρήστη, και όχι από κάποιον άλλο.
- Κατά τον προσδιορισμό του στίγματος, θα μπορούσε επίσης να δίνεται η δυνατότητα στο χρήστη προεπισκόπησης της τοποθεσίας που έχει προσδιοριστεί από την έξυπνη συσκευή, και να κάνει ο χρήστης κάποιες διορθώσεις ή αλλαγές, ειδικότερα στην

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

περίπτωση που δεν έχει προσδιοριστεί με την απαιτούμενη ακρίβεια, εκ μέρους του χρήστη, το γεωγραφικό στίγμα.

4.7 Συμπεράσματα για τις τεχνολογίες που χρησιμοποιήθηκαν

Για την ανάπτυξη της εφαρμογής έγινε χρήση της «φρέσκιας» HTML5, της JavaScript αλλά και CSS3.

Η HTML5, παρουσιάζει αρκετά ενδιαφέροντα στοιχεία, και καινοτομίες, καθώς τα 10 και πλέον χρόνια από την εισαγωγή της HTML4 καθιστούσαν επιτακτική τόσο την ανανέωσή της όσο και τον εμπλουτισμό της με νέα στοιχεία που να ανταποκρίνεται στα σύγχρονα δεδομένα και στις απαιτήσεις του διαρκώς εξελισσόμενου παγκόσμιου ιστού. Διάφορες επιλογές, πλέον αποτελούν ενσωματωμένα κομμάτια της HTML5 προσδίδοντας νέα δυναμική στη χρήση της HTML.

Ένα μικρό ενδεικτικό παράδειγμα αποτελεί το ότι πλέον η HTML5 επιτρέπει τον ορισμό ενός πεδίου εισαγωγής ως «number» προκειμένου να οριστεί ότι το συγκεκριμένο πεδίο δέχεται αριθμητική τιμή, ενώ μέχρι πρόσφατα, αυτό απαιτούσε ειδικό χειρισμό με χρήση JavaScript.

Η JavaScript, επιτρέπει στο χρήστη να παραμετροποιήσει τον κώδικά του σε μεγάλο βαθμό και να εισάγει πάρα πολλά επιπλέον στοιχεία και λειτουργίες. Επιπρόσθετα, η διάθεση της βιβλιοθήκης jQuery αποτελεί ένα πολύ χρήσιμο εργαλείο για τον προγραμματιστή, καθώς ενσωματώνει πάρα πολλές χρήσιμες συναρτήσεις μέσα στον κώδικά της, επιτρέποντας με τον τρόπο αυτό να προσθέτει ο προγραμματιστής πολλές επιπλέον λειτουργίες και επιλογές στο δικό του κώδικα με λίγες μόνο γραμμές, αντί να χρειάζεται να γράφει από την αρχή συναρτήσεις οι οποίες χρησιμοποιούνται ευρέως. Ταυτόχρονα, υπάρχουν και πρόσθετα στοιχεία που επεκτείνουν τις διαθέσιμες λειτουργίες, όπως το jQuery.blockUI το οποίο χρησιμοποιήθηκε για να απενεργοποιεί κάτω υπό συγκεκριμένες συνθήκες το περιβάλλον διεπαφής του χρήστη, και το οποίο απαιτεί την ύπαρξη της jQuery για να είναι λειτουργικό.

Επίσης το γεγονός ότι τα εργαλεία αυτά είναι εργαλεία ανοιχτού κώδικα (open source) αλλά και το γεγονός ότι υπάρχουν ομάδες συζητήσεων (forums) που υποστηρίζουν την ανάπτυξή τους, σημειώνουν τα παρατηρούμενα σφάλματα και βοηθούν τη βελτιστοποίησή τους, αποτελεί ένα σημαντικό εχέγγυο για τον προγραμματιστή.

Σε αντίθεση ωστόσο με τη jQuery, οι βιβλιοθήκες JavaScript, που υποστήριζαν την επικοινωνία με χρήση του πρωτόκολλου XML-RPC, αν και ήταν ανοιχτού κώδικα, είχαν φτωχή τεκμηρίωση και σχεδόν ανύπαρκτη υποστήριξη προς τον προγραμματιστή. Η mimic.js η οποία αποτελεί και τη βιβλιοθήκη που χρησιμοποιήθηκε για την υλοποίηση των απομακρυσμένων κλήσεων με χρήση του πρωτοκόλλου XML-RPC, ανταποκρίνεται με επιτυχία στις απαιτήσεις τις εφαρμογής, (αν και υστερεί στην υποστήριξη δυνατότητας επαλήθευσης ονόματος χρήστη και κωδικού πρόσβασης), και χειρίζεται με επιτυχία τις απαντήσεις «επιτυχίας» από τη διαδικτυακή υπηρεσία της επιχείρησης. Ωστόσο, λόγω του τεχνικού (από την πλευρά της διαδικτυακής υπηρεσίας της επιχείρησης) περιορισμού αναφορικά με τις περιπτώσεις σφάλματος, δεν έχει επαληθευτεί το αν χειρίζεται με την ίδια επιτυχία και τις περιπτώσεις σφάλματος.

Η CSS3 βρίσκεται και αυτή όπως και η HTML5 σε φάση ανάπτυξης και εξέλιξης ακόμα, από το World Wide Web Consortium (W3C), και ήδη τα «πρώτα» σημάδια, είναι εξίσου θετικά με αυτά της HTML5, καθώς με λίγες γραμμές κώδικα, ενσωματώνει πλέον λειτουργίες οι οποίες απαιτούσαν εξειδικευμένο χειρισμό στις προγενέστερες εκδόσεις. Για παράδειγμα, η περιστροφή ενός πεδίου της HTML (<div>κείμενο</div>), ή η εμφάνισή του «υπό κλίση» μπορεί να απαιτούσε μέχρι και χρήση τεχνολογιών Flash για να υλοποιηθεί, ενώ πλέον μπορεί σύντομα και αρκετά εύκολα.

4.8 Συμπεράσματα για το Phonegap

Το Phonegap, αποτελεί πράγματι μία αρκετά ενδιαφέρουσα πρόταση η οποία δίνει μια ξεχωριστή ώθηση για τη δημιουργία εφαρμογών για τις έξυπνες συσκευές. Η «παραδοσιακή» δημιουργία κώδικα σε συγκεκριμένο λειτουργικό σύστημα, δεν επιτρέπει την «μετανάστευση» (migration) σε ένα άλλο λειτουργικό σύστημα μέσω κάποιας τυποποιημένης διαδικασίας, αλλά απαιτεί να γίνει εκ νέου συγγραφή του προγράμματος. Ωστόσο, με το phonegap, παρέχεται η

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

δυνατότητα στον προγραμματιστή να μπορεί να γράψει ενιαίο κώδικα ο οποίος θα μετατραπεί σε εκτελέσιμη εφαρμογή για διαφορετικά λειτουργικά συστήματα, απλοποιεί αρκετά τη διαδικασία δημιουργίας εφαρμογών για πολλαπλά διαφορετικά λειτουργικά συστήματα. Το γεγονός αυτό συμβάλει και στη μείωση του κόστους ανάπτυξης των εφαρμογών, λόγω και των λιγότερων εργατωρών που απαιτούνται για την ανάπτυξη του κώδικα από τον προγραμματιστή, στην περίπτωση που στόχος είναι η εφαρμογή που αναπτύσσεται να καλύψει την πλειοψηφία των διαθέσιμων λειτουργικών συστημάτων.

Επιπλέον, είναι αρκετά ενθαρρυντικό το γεγονός, ότι η Adobe, αντιλαμβανόμενη τη δυναμική που δημιουργεί το Phonegap, και τα ιδιαίτερα στοιχεία που αυτό εισάγει στο χώρο της ανάπτυξης εφαρμογών για κινητά τηλέφωνα, απέκτησε την Nitobi, την εταιρεία η οποία βρίσκεται πίσω από το Phonegap, ενώ ήδη στα πακέτα που εμπορεύεται η Adobe και συγκεκριμένα στο πακέτο Dreamweaver υπάρχει ενσωματωμένη η δυνατότητα δημιουργίας εφαρμογής με τη χρήση του Phonegap. Επίσης, καθώς το Phonegap, κάνει τα πρώτα του βήματα, έχοντας σε διάθεση σταθερή έκδοση εδώ και λίγο καιρό, παρέχεται αρκετή υποστήριξη από την ομάδα ανάπτυξης αναφορικά με απορίες των χρηστών αλλά και με υποστηριζόμενες λειτουργίες, επίλυση προβλημάτων, και καταχώρηση και εντοπισμό σφαλμάτων (bugs). Πιο συγκεκριμένα, υπάρχει ομάδα συζήτησης στο διαδίκτυο, (Google Groups) όπου μπορούν να τεθούν υπό συζήτηση τα διάφορα θέματα που απασχολούν τους χρήστες.

Ωστόσο η επιλογή ανάπτυξης εφαρμογών βασισμένες στη χρήση της πλατφόρμας του Phonegap, δεν μπορεί να θεωρηθεί σε καμία περίπτωση ότι καταργεί την ανάπτυξη εφαρμογών, οι οποίες γράφονται στη συγκεκριμένη γλώσσα του λειτουργικού συστήματος. Το Phonegap, καθορίζει «αυστηρά» πλαίσια μέσα στα οποία μπορεί να κινηθεί η ανάπτυξη της εφαρμογής, με χρήση HTML5, JavaScript και CSS3. Το γεγονός αυτό, θέτει εν γένει περιορισμούς αναφορικά με τις δυνατότητες που μπορούν να ενσωματωθούν σε μια συγκεκριμένη εφαρμογή. Για το λόγο αυτό, θα πρέπει, να γίνει προσεκτική ανάλυση των απαιτήσεων που υπάρχουν για τη δημιουργία μιας εφαρμογής, και με τις δεδομένες απαιτήσεις να εξεταστεί, αν ενδείκνυται η χρήση του Phonegap για την συγκεκριμένη υλοποίηση.

4.9 Συμπεράσματα για το ρόλο των smartphones

Η ανάγκη των σύγχρονων ανθρώπων να έχουν έναν «μικρό υπολογιστή» πάντα και παντού μαζί τους, σε συνδυασμό με το γεγονός ότι το κινητό τηλέφωνο αποτελούσε μια συσκευή που πράγματι έχουν πάντα μαζί τους, οδήγησε στην εξέλιξη των κινητών τηλεφώνων σε αυτό που πλέον είναι γνωστό ως «έξυπνες συσκευές». Οι έξυπνες συσκευές (smart phones), εδώ και μερικά χρόνια (τα τελευταία χρόνια με ακόμα μεγαλύτερη ένταση) έχουν ξεπεράσει κατά πολύ τον κλασικό ορισμό του τηλεφώνου που χρησιμοποιείται για την πραγματοποίηση κλήσεων και την αποστολή γραπτών μηνυμάτων.

Χαρακτηριστικό της μεγάλης ταχύτητας και εξέλιξης που γνωρίζει η συγκεκριμένη κατηγορία ηλεκτρονικών συσκευών, είναι οι διαρκώς νέες τεχνολογίες που ενσωματώνονται καθώς και η προώθηση διαρκώς νέων μοντέλων με βελτιωμένα τεχνικά χαρακτηριστικά. Από το πληκτρολόγιο, την έγχρωμη οθόνη, και την ύπαρξη μιας κάμερας στοιχειώδους ανάλυσης που χαρακτήριζαν τις πρώτες γενιές των «έξυπνων συσκευών» σε ένα όσο το δυνατόν μικρότερο μέγεθος, πλέον οι έξυπνες συσκευές έχουν διαφοροποιηθεί σε μεγάλο βαθμό. Πλέον οι έξυπνες συσκευές τείνουν να γίνονται όλο και πιο «ογκώδεις» προκειμένου να ενσωματώσουν όσο το δυνατόν περισσότερα χαρακτηριστικά και δυνατότητες που να επιτρέπουν στο χρήστη να έχει πράγματι την εμπειρία ενός μικρού ηλεκτρονικού υπολογιστή στα χέρια του:

- οθόνες (μεγαλύτερες ακόμα και μεγαλύτερες από τέσσερις ίντσες (4") ενώ ήδη υπάρχουν και συσκευές που ενσωματώνουν δυνατότητα θέασης με τεχνολογία τριών διαστάσεων-3D),
- κάμερες 8 ή ακόμα και 12 megapixel
- ενσωματωμένοι δορυφορικοί δέκτες (gps)
- επεξεργαστές με δύο ή τέσσερις πυρήνες
- ξεχωριστή μονάδα επεξεργασίας της εικόνας

- μεταφορά δεδομένων σε υψηλές ταχύτητες, είτε μέσω ασύρματων δικτύων (WiFi) είτε μέσω τεχνολογίας 3G
- λειτουργικά συστήματα αντίστοιχα των «κλασσικών» ηλεκτρονικών υπολογιστών

Το σύνολο αυτών των χαρακτηριστικών, έχει ως αποτέλεσμα να ανθίσει και η ανάπτυξη εφαρμογών που πλέον προορίζονται για της έξυπνες συσκευές.

- Αγορές προϊόντων
- Αγορές υπηρεσιών
- Χρήση τραπεζικών υπηρεσιών μέσω δικτύου
- Υπηρεσίες ενημέρωσης
- Υπηρεσίες ψυχαγωγίας

Αποτελούν μερικά από τα πεδία για τα οποία υπάρχουν ήδη εφαρμογές και διατίθενται στους χρήστες, ενώ ολοένα και περισσότερες επιχειρήσεις διαθέτουν υπηρεσίες προς τους χρήστες μέσω εφαρμογών για έξυπνες συσκευές.

Αυτά τα δεδομένα συνηγορούν, στο ότι η αγορά των έξυπνων συσκευών (smart phones) πρόκειται για μία αγορά η οποία όπως φαίνεται βρίσκεται σε φάση ιδιαίτερης ανάπτυξης τόσο στον τομέα του υλικού όσο πολύ περισσότερο στον τομέα του λογισμικού. Παρέχονται υπηρεσίες και τεχνολογίες οι οποίες χαίρουν ιδιαίτερα ευρείας αποδοχής και δείχνουν πως πλέον τα έξυπνα κινητά τηλέφωνα, έχουν κάνει απαραίτητη την ύπαρξή τους στον τελικό χρήστη. Με τη χρήση των έξυπνων συσκευών, η πρόσβαση σε κάθε μορφής πληροφορία, δεν μένει στα κλειστά όρια ενός γραφείου ή ενός δωματίου, αλλά πλέον βρίσκεται ανά πάσα στιγμή στη διάθεση του ενδιαφερομένου.

4.10 Γενικά συμπεράσματα

Οι τεχνολογίες που μελετήθηκαν, και από τις οποίες άλλες χρησιμοποιήθηκαν ενώ άλλες απορρίφθηκαν, αλλά και τα συμπεράσματα που εξήχθησαν από τη μελέτη των συγκεκριμένων τεχνολογιών, αποτελούν μια πολύ σημαντική παρακαταθήκη σε προσωπικό επίπεδο αναφορικά με τη μελέτη, τη χρήση, και την διαχείριση των διαθέσιμων τεχνολογιών.

Ωστόσο, εκτός από τις πολύ σημαντικές γνώσεις που αποκτήθηκαν και τις δεξιότητες που αναπτύχθηκαν κατά την εκπόνηση της συγκεκριμένης διατριβής, οι συνθήκες στις οποίες αναπτύχθηκε η συγκεκριμένη εφαρμογή προσέφεραν και ένα πρόσθετο όφελος. Το γεγονός ότι η συγκεκριμένη διατριβή και συγκεκριμένα η εφαρμογή που αυτή πραγματεύεται, αναπτύχθηκε πάνω σε πραγματικά δεδομένα, μιας δραστήριας και αρκετά επιτυχημένης επιχείρησης, της οποίας η ομάδα ανάπτυξης διέθεσε όλα τα απαραίτητα στοιχεία που χρειάζονταν, δημιούργησε κάποιες συνθήκες οι οποίες θα μπορούσε να θεωρηθεί ότι προσομοιώνουν σε ένα βαθμό τη διαδικασία εκπόνησης ενός συγκεκριμένου έργου σε επαγγελματικό επίπεδο.

Μπορεί να απουσίαζαν στοιχεία, όπως η διαρκής παρουσία σε ένα φυσικό περιβάλλον εργασίας, και η συνεχής αλληλεπίδραση με τους υπόλοιπους εργαζόμενους, ωστόσο, υπήρχαν άλλα στοιχεία, όπως:

- η καταγραφή και η θέσπιση συγκεκριμένων στόχων, όπως γίνεται σε σύγχρονες επιχειρήσεις κατά την ανάπτυξη συγκεκριμένων έργων,
- ο προσδιορισμός συγκεκριμένων χρονοδιαγραμμάτων, προκειμένου να τίθενται ρεαλιστικοί και πραγματοποιήσιμοι στόχοι μέσα σε συγκεκριμένα χρονικά περιθώρια, προκειμένου να υπάρχει καλύτερη επισκόπηση της εργασίας που εκτελείται και τις προόδου που έχει επιτευχθεί,
- η αλληλεπίδραση με άλλους ανθρώπους προκειμένου να λυθούν προβλήματα τα οποία ενδεχομένως να μην ενέπιπταν αμιγώς στα πλαίσια της εφαρμογής,
- η χρησιμοποίηση πραγματικών δεδομένων και πραγματικών συνθηκών ανάπτυξης (πραγματική και πλήρως λειτουργική διαδικτυακή υπηρεσία, και όχι εικονική προκειμένου να μελετηθεί η συγκεκριμένη τεχνολογία)

τα οποία αποτελούν μια εξίσου πολύ σημαντική παρακαταθήκη σε προσωπικό επίπεδο.

Βιβλιογραφία

- A. -. A. P., 2010. *Issue 11891: ConnectivityManager.getActiveNetworkInfo() reporting NOT CONNECTED when connected (via wifi)*. [Ηλεκτρονικό]
Available at: <http://code.google.com/p/android/issues/detail?id=11891>
- Alsup, M., 2009. *JQuery BlockUI Plugin*. [Ηλεκτρονικό]
Available at: <http://jquery.malsup.com/block/>
- Anon., 2011. [Ηλεκτρονικό]
Available at: [http://en.wikipedia.org/wiki/Minification_\(programming\)](http://en.wikipedia.org/wiki/Minification_(programming))
- Anon., 2011. *Cross-site scripting*. [Ηλεκτρονικό]
Available at: http://en.wikipedia.org/wiki/Cross-site_scripting
- Anscamobile, 2012. *Corona*. [Ηλεκτρονικό]
Available at: <http://www.anscamobile.com>
- Appcelerator, 2012. *ParticleCode*. [Ηλεκτρονικό]
Available at: <http://www.particlecode.com>
- Appcelerator, 2012. *Titanium*. [Ηλεκτρονικό]
Available at: <http://www.appcelerator.com>
- Atkins, A. S., A. H. N. P. H. A. & Shah, H., 2006. Extending E-Business Applications Using Mobile Technology. *The 3rd International Conference on Mobile Technology, Applications and Systems – Mobility 2006*.
- Backelite, 2012. *WOPE (Write Once Publish Everywhere)*. [Ηλεκτρονικό]
Available at: <http://www.backelite.com>
- Castledine, E. & Sharkie, C., 2010. *JQuery : Novice to Ninja*. s.l.:SitePoint Pty Ltd.
- Cavazza, F., 2011. *Mobile Web App vs. Native App? It's Complicated*. [Ηλεκτρονικό]
Available at: <http://www.forbes.com/sites/fredcavazza/2011/09/27/mobile-web-app-vs-native-app-its-complicated/>
- Christophilus, 2005. *Writing XML using JavaScript*. [Ηλεκτρονικό]
Available at: <http://www.codeproject.com/Articles/12504/Writing-XML-using-JavaScript>
- Farley, P. & Capp, M., 2005. Mobile Web Services. *BT Technology Journal*.
- Gehlen, G. & Bergs, R. G. R., 2004. Performance of mobile Web Service Access using the Wireless Application Protocol (WAP). Στο: *Proceedings of World Wireless Congress 2004*. s.l.:s.n., pp. 427-432.
- Giunta, G. & Lott, M., 2009. *JS-XMLRPC debugger*. [Ηλεκτρονικό]
Available at: <http://phpxmlrpc.sourceforge.net/jsxmlrpc/debugger/debugger.html>

- Goncalves, C. E., 2009. *Mimic.js*. [Ηλεκτρονικό]
Available at: <http://mimic-xmlrpc.sourceforge.net/>
- jQuery Foundation, 2011. *jQuery*. [Ηλεκτρονικό]
Available at: <http://www.jquery.com/>
- jQuery foundation, 2011. *jQueryMobile*. [Ηλεκτρονικό]
Available at: <http://www.jquerymobile.com/>
- jQuery UI Team, 2011. *jQueryUI*. [Ηλεκτρονικό]
Available at: <http://www.jqueryui.com>
- Kollhof, J.-K., 2003. *Jsolait*. [Ηλεκτρονικό]
Available at: <http://jan.kollhof.net/projects/svg/examples/jsolait/doc/jsolait.xhtml>
- Lielmanis, E., 2011. [Ηλεκτρονικό]
Available at: <http://jsbeautifier.org/>
- McGinnis, J., 2011. *How to: Disable Same-Origin Policy in Chrome*. [Ηλεκτρονικό]
Available at: <http://joshuamcginnis.com/2011/02/28/how-to-disable-same-origin-policy-in-chrome/>
- McIntosh, J. C., 2005. Mobile commerce's impact on today's workforce: issues, impacts and implications. *International Journal of Mobile Communications*, pp. 99-113.
- Motorola Solutions, 2012. *Rhomobile*. [Ηλεκτρονικό]
Available at: <http://www.rhomobile.com>
- Nitobi, 2011. *Phonegap*. [Ηλεκτρονικό]
Available at: <http://www.phonegap.com>
- Oh, S. & Fox, G. C., 2007. Optimizing Web Service messaging performance in mobile computing. *Future Generation Computer Systems*, pp. 623-632.
- Ruter, W., 2007. *JSON-XML-RPC*. [Ηλεκτρονικό]
Available at: <http://code.google.com/p/json-xml-rpc/>
- Scripting News, Inc, 1999. *XML-RPC*. [Ηλεκτρονικό]
Available at: <http://xmlrpc.scripting.com/>
- Sencha, 2012. [Ηλεκτρονικό]
Available at: <http://www.sencha.com>
- Spitogatos.gr, 2011. *ListingSync 1.0 - Web Service interface Documentation*. s.l.:s.n.
- Spitogatos.gr, 2011. *Regions and Areas v1.0*. s.l.:www.spitogatos.gr.
- Spitogatos.gr, 2011. *Spitogatos.gr listingsync v1.0 data structures - doc 1.4*. s.l.:s.n.

Steele, R., 2003. A Web Services-based System for Ad-hoc Mobile Application Integration. *Proceedings of the International Conference on Information Technology: Computers and Communications*.

Vriddhi Technology Services, 2012. *Mobile Application Development*. [Ηλεκτρονικό]
Available at: <http://www.vriddhi.biz>

Worklight, 2012. *HTML5, Hybrid or Native Mobile App Development Webinar*. [Ηλεκτρονικό]
Available at: <http://www.worklight.com/resources/webinars-and-tools/native-web-hybrid-mobile-app-development>

World Wide Web Consortium (W3C), 1999. [Ηλεκτρονικό]
Available at: <http://www.w3.org/TR/wbxml/>

World Wide Web Consortium (W3C), 2011. *HTML5*. [Ηλεκτρονικό]
Available at: <http://www.w3.org/>

Βίρβου, Μ., 2008. *Τεχνολογία Λογισμικού (Πανεπιστημιακές σημειώσεις)*. Πειραιάς: s.n.

IBM, 2012. *Worklight*. [Ηλεκτρονικό]
Available at: <http://www.worklight.com>

Παράρτημα 1 – Εγχειρίδια

1.1 Εγχειρίδιο χρήστη

1.1.1 Εισαγωγή

Με το τρέχον εγχειρίδιο, παρέχονται όλες οι απαραίτητες πληροφορίες προκειμένου να μπορεί να γίνει αποτελεσματική χρήση της εφαρμογής σε οποιαδήποτε έξυπνη συσκευή εγκατασταθεί.

Με αυτήν την εφαρμογή, είτε δίνεται η δυνατότητα να καταχωρηθούν αγγελίες ακινήτων στο λογαριασμό του χρήστη, οι οποίες θα δημοσιευτούν άμεσα στον ιστότοπο της επιχείρησης, εφόσον ολοκληρωθεί σωστά η όλη διαδικασία.

Προκειμένου να γίνει χρήση της εφαρμογής, θα πρέπει πρώτα να εγκατασταθεί στην έξυπνη συσκευή του χρήστη. Για να γίνει η εγκατάσταση, θα πρέπει ο χρήστης να ανατρέξει στο εγχειρίδιο χρήσης της συσκευής σας προκειμένου να ακολουθηθεί ο πιο ενδεδειγμένος τρόπος εγκατάστασης σύμφωνα με τις οδηγίες του κατασκευαστή. Ωστόσο, ενδεικτικά αναφέρονται, οι πιο συνηθισμένοι τρόποι εγκατάστασης:

- Εγκατάσταση μέσω ηλεκτρονικού υπολογιστή. Ο χρήστης, ακολουθώντας τις οδηγίες από στο εγχειρίδιο χρήσης της έξυπνης συσκευής μπορεί να κάνει την εγκατάσταση μέσω ηλεκτρονικού υπολογιστή, σύμφωνα με τις οδηγίες του κατασκευαστή.
- Εγκατάσταση απευθείας από το τηλέφωνο. Θα πρέπει ο χρήστης να εντοπίσει το εκτελέσιμο αρχείο με την κατάληξη .apk, το οποίο έχει μεταφέρει σε κάποιο φάκελο της κάρτας μνήμης της συσκευής, να το επιλέξει, και να ακολουθήσει τα βήματα εγκατάστασης.

Για να γίνει δυνατή η καταχώρηση αγγελίας, είναι απαραίτητο να υπάρχει πρόσβαση στο διαδίκτυο μέσω της έξυπνης συσκευής. Ανάλογα με το είδος της σύνδεσης η διαδικασία καταχώρησης μπορεί να διαρκέσει από λίγα δευτερόλεπτα έως μερικά λεπτά ενώ ενδέχεται να υπάρξει χρέωση για την χρήση των υπηρεσιών διαδικτύου από τον πάροχο της σύνδεσής του χρήστη.

Συνιστάται, να είναι ήδη ενεργοποιημένη η σύνδεση στο διαδίκτυο κατά το άνοιγμα της εφαρμογής, καθώς θα διαπιστώνεται πως είναι πιο αποτελεσματική και απλή η χρήση της εφαρμογής. Ωστόσο, υπάρχει η δυνατότητα στο χρήστη, να συνδεθεί στο διαδίκτυο και μετά την συμπλήρωση της φόρμας όμως, σε αυτή την περίπτωση θα πρέπει να ληφθούν υπόψη και οι ακόλουθες σημειώσεις:

- Δεν θα πρέπει να πραγματοποιηθεί έξοδος από την εφαρμογή, αλλά παράλληλο άνοιγμα της καρτέλας για τη σύνδεση στο διαδίκτυο, διαφορετικά τα συμπληρωμένα δεδομένα στη φόρμα θα διαγραφούν.
- Ορισμένες φορές, σε εξαιρετικές περιπτώσεις οι έξυπνες συσκευές ενδέχεται να μην «αναγνωρίσουν» την αλλαγή κατάστασης στη σύνδεση με το διαδίκτυο, με αποτέλεσμα να πρέπει να γίνει έξοδος από την εφαρμογή και εκ νέου άνοιγμα και καταχώρηση των στοιχείων.

1.1.2 Γνωριμία με το περιβάλλον

Ανοίγοντας την εφαρμογή θα εμφανιστεί η βασική οθόνη διεπαφής, όπου μπορεί ο χρήστης να εισάγει τα στοιχεία του ακινήτου που επιθυμεί. Τα πεδία που καλείται, είτε να συμπληρώσει είναι τα πλέον απαραίτητα προκειμένου να γίνει αποδεκτή μια αγγελία προς καταχώρηση από τον ιστότοπο της επιχείρησης.

(Ανάλογα με την κατηγορία ακινήτου που θα επιλεγεί, και πιο συγκεκριμένα, με τις επιλογές, κατοικία ή επαγγελματικό ακίνητο, θα εμφανιστούν ορισμένα επιπλέον πεδία που θα κληθεί ο χρήστης να συμπληρώσει)

spitogatos.gr
Η Νο1 Ελληνική ιστοσελίδα αγγελιών ακινήτων

Παρακαλώ συμπληρώστε τη φόρμα:

Κατηγορία ακινήτου

Είδος ακινήτου

Τιμή (χωρίς δεκαδικά)/Νόμισμα Ευρώ

Εμβαδό (σε τ.μ.)

Είδος αγγελίας

Περιοχή

Υποπεριοχή

Προσδιορίστε το χειροκίνητα ή πιέστε το κουμπί

Γεωγραφικό πλάτος:

Γεωγραφικό μήκος:

Ώρα προσδιορισμού στίγματος:

Αν θέλετε να μεταβείτε στην καταχώρηση φωτογραφιών πατήστε το κουμπί.

Εικόνα 7 - Βασική οθόνη διεπαφής

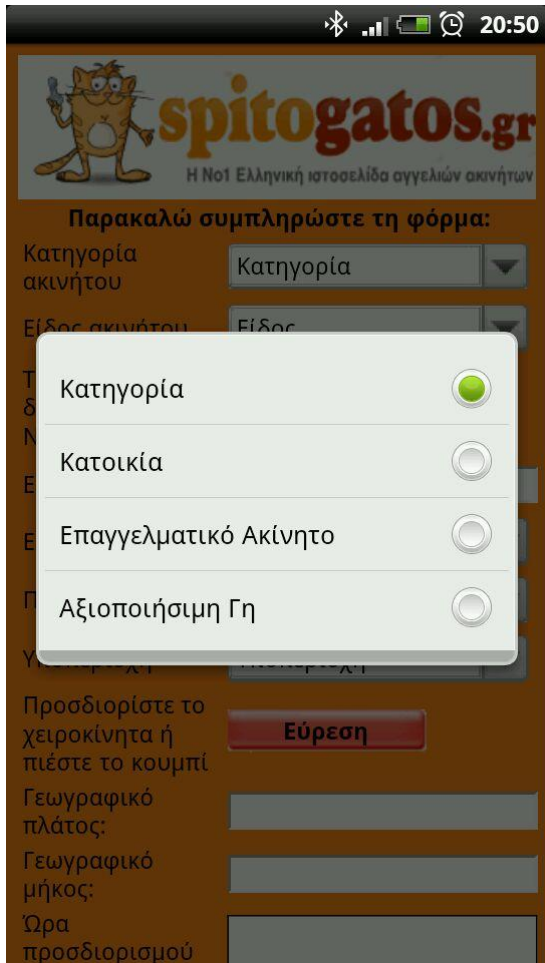
Πεδίο - Κατηγορία ακινήτου

Στην κατηγορία του ακινήτου, πρέπει να επιλεγεί σε ποια γενική κατηγορία ανήκει το ακίνητο το οποίο πρόκειται να καταχωρηθεί, ενώ πρέπει να δοθεί προσοχή πως δεν επιτρέπεται να επιλεγεί «είδος ακινήτου» εάν πρώτα δεν έχει επιλεγεί «κατηγορία ακινήτου». Το πεδίο «είδος ακινήτου» θα είναι διαθέσιμο με τις επιλογές που αντιστοιχούν έπειτα από την αρχική επιλογή της «κατηγορίας ακινήτου».

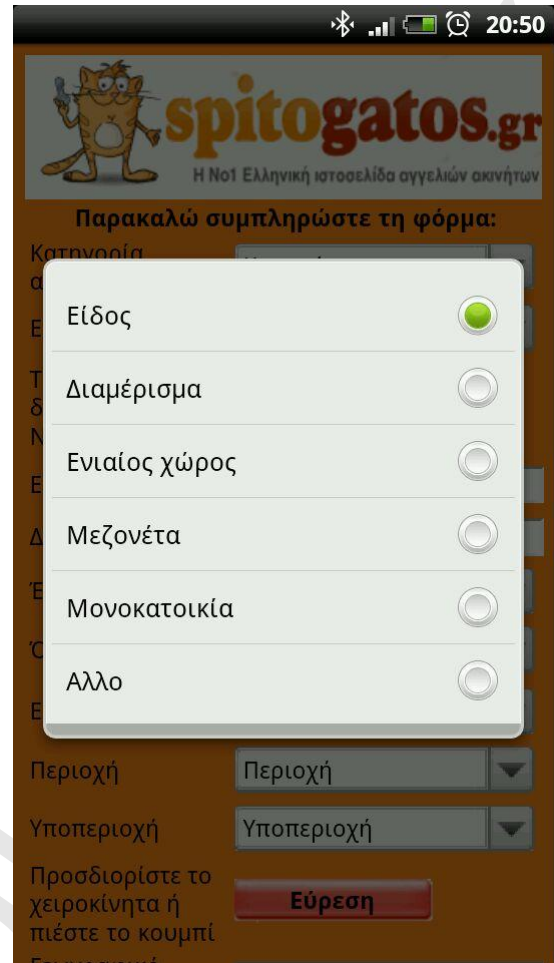
Οι κατηγορίες ακινήτου είναι (εικόνα 2)

- Κατοικία
- Επαγγελματικό Ακίνητο
- Αξιοποιήσιμη Γη

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας



Εικόνα 8 - Κατηγορίες ακινήτου



Εικόνα 9 - Είδος ακινήτου (κατοικία)

Πεδίο – Είδος ακινήτου

Με την επιλογή της κατηγορίας ακινήτου «κατοικία», οι επιλογές που εμφανίζονται από το μενού «είδος ακινήτου» είναι οι ακόλουθες (εικόνα 3):

- Διαμέρισμα
- Ενιαίος χώρος
- Μεζονέτα
- Μονοκατοικία
- Άλλο

Με την επιλογή της κατηγορίας ακινήτου «επαγγελματικό ακίνητο», οι επιλογές που εμφανίζονται από το μενού «είδος ακινήτου» είναι οι ακόλουθες (εικόνα 4):

- Γραφείο
- Κατάστημα
- Αποθήκη
- Άλλο

Με την επιλογή της κατηγορίας ακινήτου «αξιοποιήσιμη γη», οι επιλογές που εμφανίζονται από το μενού «είδος ακινήτου» είναι οι ακόλουθες (εικόνα 5):

- Οικόπεδο
- Αγροτεμάχιο
- Άλλο

spitogatos.gr
Η Νο1 Ελληνική ιστοσελίδα αγγελιών ακινήτων

Παρακαλώ συμπληρώστε τη φόρμα:

Κατηγορία ακινήτου: Επαγγελματικό Ακίνητ

Είδος ακινήτου:

- Είδος
- Γραφείο
- Κατάστημα
- Αποθήκη
- Άλλο

Είδος αγγελίας: Επιλογή

Περιοχή: Περιοχή

Υποπεριοχή: Υποπεριοχή

Προσδιορίστε το χειροκίνητα ή πιάστε το κουμπί: **Εύρεση**

Εικόνα 4 - Είδος ακινήτου (επαγγελματικό ακίνητο)

spitogatos.gr
Η Νο1 Ελληνική ιστοσελίδα αγγελιών ακινήτων

Παρακαλώ συμπληρώστε τη φόρμα:

Κατηγορία ακινήτου: Αξιοποιήσιμη Γη

Είδος ακινήτου:

- Είδος
- Οικόπεδο
- Αγροτεμάχιο
- Άλλο

Προσδιορίστε το χειροκίνητα ή πιάστε το κουμπί: **Εύρεση**

Γεωγραφικό πλάτος:

Γεωγραφικό μήκος:

Ώρα προσδιορισμού

Εικόνα 5 - Είδος ακινήτου (αξιοποιήσιμη γη)

Πεδίο - Τιμή (χωρίς δεκαδικά) / Νόμισμα

Στη συνέχεια πρέπει να συμπληρωθεί η τιμή του ακινήτου καθώς και το νόμισμα στο οποίο αντιστοιχεί η τιμή που θα εισαχθεί. Προκειμένου να αποφευχθεί λανθασμένη εισαγωγή στοιχείων, πως δε μπορεί να γίνει εισαγωγή οποιουδήποτε άλλου χαρακτήρα, εκτός από αριθμούς. Η τιμή που θα εισαχθεί πρέπει υποχρεωτικά να είναι ακέραιος αριθμός (καθώς δεν επιτρέπεται η χρήση της υποδιαστολής), ενώ δεν επιτρέπεται και ο χωρισμός του αριθμού ανά «χιλιάδες».

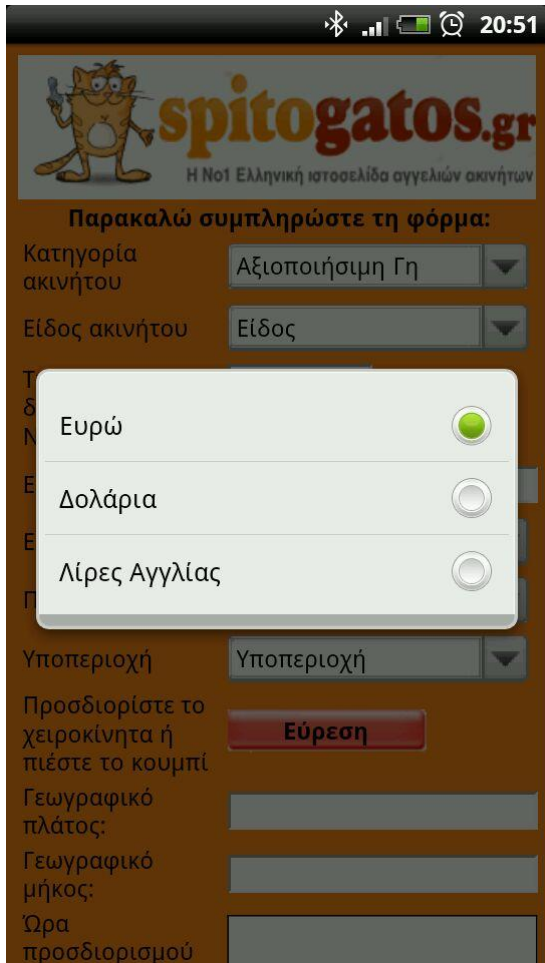
Για παράδειγμα, αν για ένα ακίνητο προς πώληση, η επιθυμητή τιμή που θέλετε για εισαγωγή είναι 250.000€, τότε θα πρέπει να γίνει εισαγωγή τη τιμής 250000 και να επιλεγεί από τα διαθέσιμα νομίσματα (ευρώ, δολάρια, λίρες Αγγλίας) το ευρώ (εικόνα 6).

Πεδίο - Εμβαδό

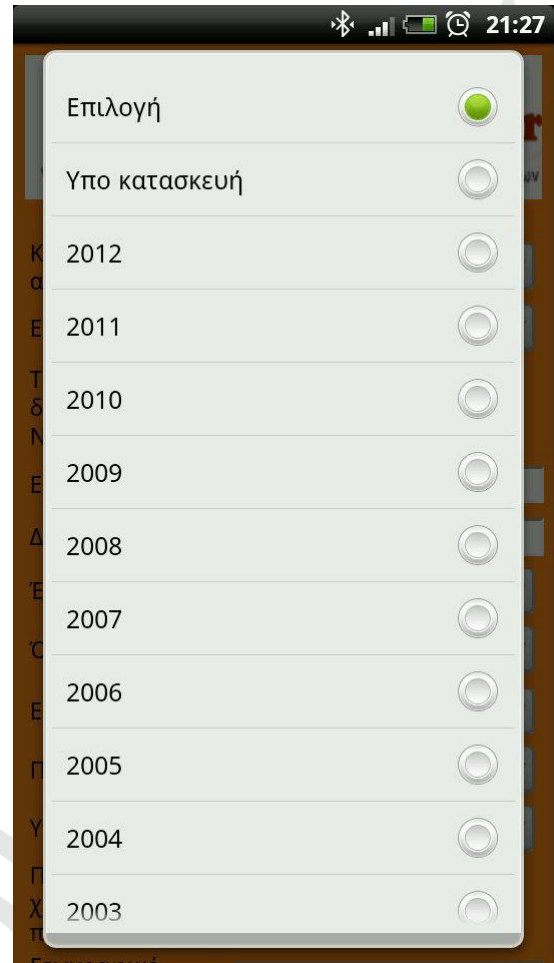
Επόμενο πεδίο που πρέπει να συμπληρωθεί, είναι το εμβαδό του ακινήτου που πρόκειται να καταχωρηθεί. Ομοίως, όπως και στην περίπτωση του πεδίου «τιμή» δεν επιτρέπεται η χρήση χαρακτήρων εκτός από αριθμούς.

Πεδίο - Δωμάτια/Ετος κατασκευής/Όροφος

! Προσοχή τα πεδία Δωμάτια/Ετος κατασκευής/Όροφος εμφανίζονται μόνο στην επιλογή ως κατηγορίας ακινήτου την κατοικία ή το επαγγελματικό ακίνητο. Στην περίπτωση επιλογής αξιοποιήσιμης γης, τα πεδία δεν εμφανίζονται για να μην γίνει λάθος καταχώρηση.



Εικόνα 6 - Νόμισμα



Εικόνα 7 - Έτος κατασκευής

Στο πεδίο δωμάτια, γίνεται εισαγωγή του αριθμού των δωματίων της κατοικίας ή του επαγγελματικού ακινήτου που θα καταχωρηθεί. Η εισαγωγή γίνεται χρησιμοποιώντας αριθμητικά ψηφία και όχι γράφοντας τον αριθμό των δωματίων χρησιμοποιώντας λέξεις.

Στο πεδίο έτος κατασκευής (εικόνα 7) εισάγεται το έτος το οποίο κατασκευάστηκε το ακίνητο, ή εάν βρίσκεται ακόμα υπό κατασκευή. Οι διαθέσιμες χρονολογίες, είναι από το 1920, μέχρι το τρέχον έτος.

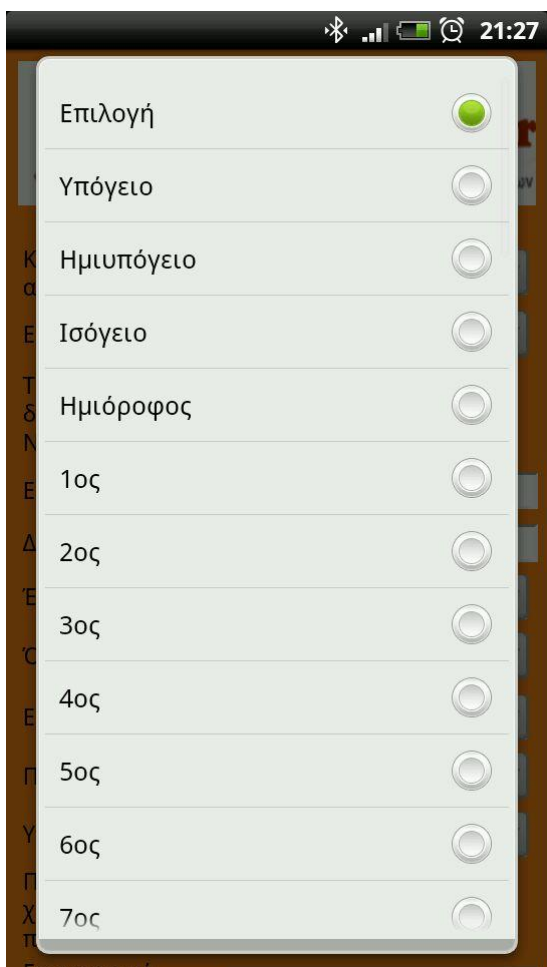
Στο πεδίο όροφος, εισάγεται ο όροφος στον οποίο βρίσκεται το ακίνητο. Οι διαθέσιμες επιλογές είναι (εικόνα 8):

- Υπόγειο
- Ημιυπόγειο
- Ισόγειο
- Ημιώροφος
- Όροφοι από 1 έως 50

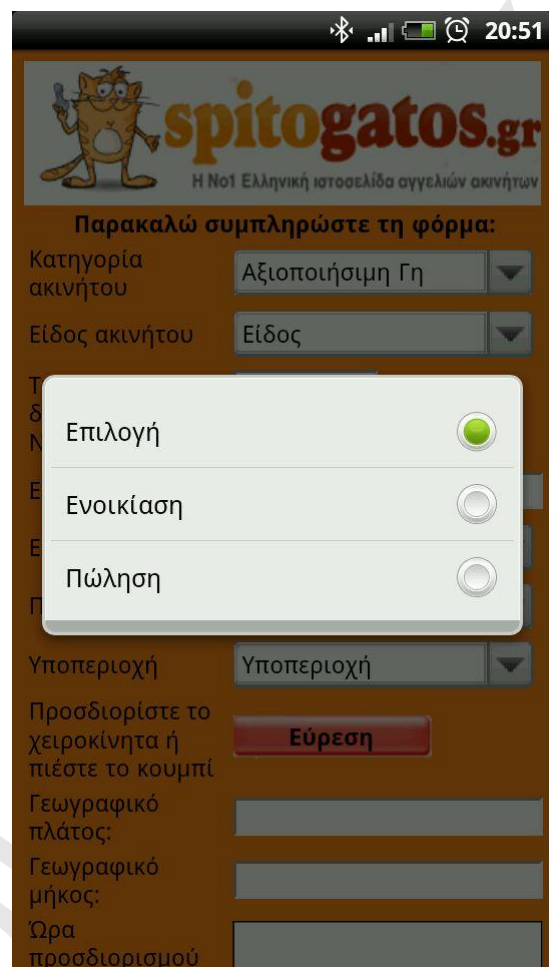
Πεδίο - Είδος αγγελίας

Στο πεδίο «είδος αγγελίας» (εικόνα 9), εισάγεται το είδος της αγγελίας που πρόκειται να καταχωρηθεί. Οι επιλογές είναι:

- Ενοικίαση
- Πώληση



Εικόνα 8 - Όροφος



Εικόνα 9 – Είδος αγγελίας

Πεδίο – Περιοχή / Υποπεριοχή

Στο πεδίο περιοχή (εικόνα 10), γίνεται επιλογή από μια λίστα επιλογών της ευρύτερης περιοχής στην οποία βρίσκεται το ακίνητο. Στη συνέχεια, γίνεται επιλογή από το πεδίο «υποπεριοχή» (εικόνα 11) της συγκεκριμένης υποπεριοχής στην οποία βρίσκεται το ακίνητο.

Για παράδειγμα, αν το ακίνητο προς καταχώρηση βρίσκεται στους Αμπελόκηπους, θα πρέπει πρώτα επιλεγεί η ευρύτερη περιοχή που είναι «Αθήνα – Κέντρο» και στη συνέχεια από το μενού επιλογών «υποπεριοχή» να επιλεγεί συγκεκριμένα η περιοχή των Αμπελοκήπων.

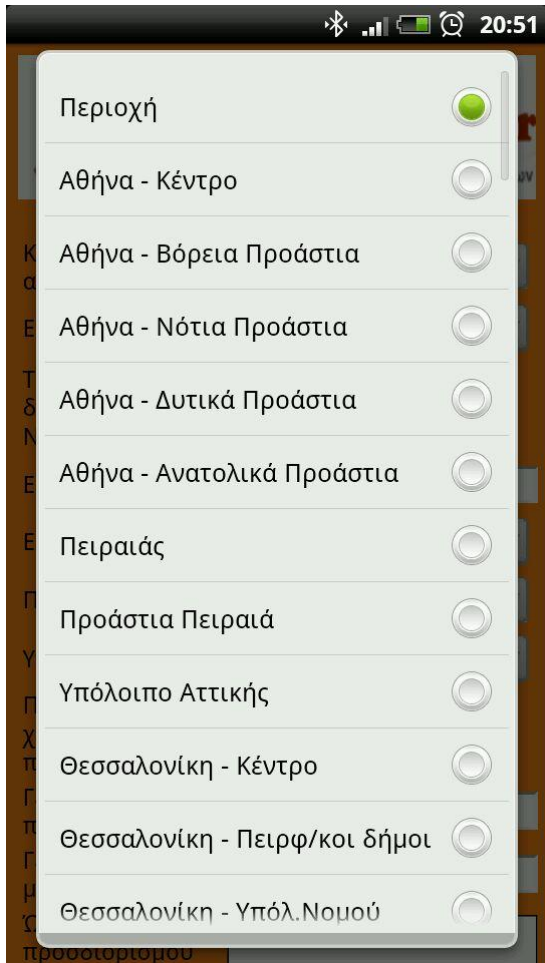
Αν γίνει προσπάθεια επιλογής από το μενού επιλογών «υποπεριοχή» κάποιας επιλογής, χωρίς πρώτα να έχει επιλεγεί η αντίστοιχη επιλογή «περιοχή», θα διαπιστώσει ο χρήστης πως το μενού επιλογών είναι κενό. Το μενού επιλογών «υποπεριοχή» εμφανίζει στοιχεία, μόνο εφόσον, έχει επιλεγεί κάποια περιοχή από το «γονικό» μενού.

Πεδίο - Γεωγραφικό στίγμα

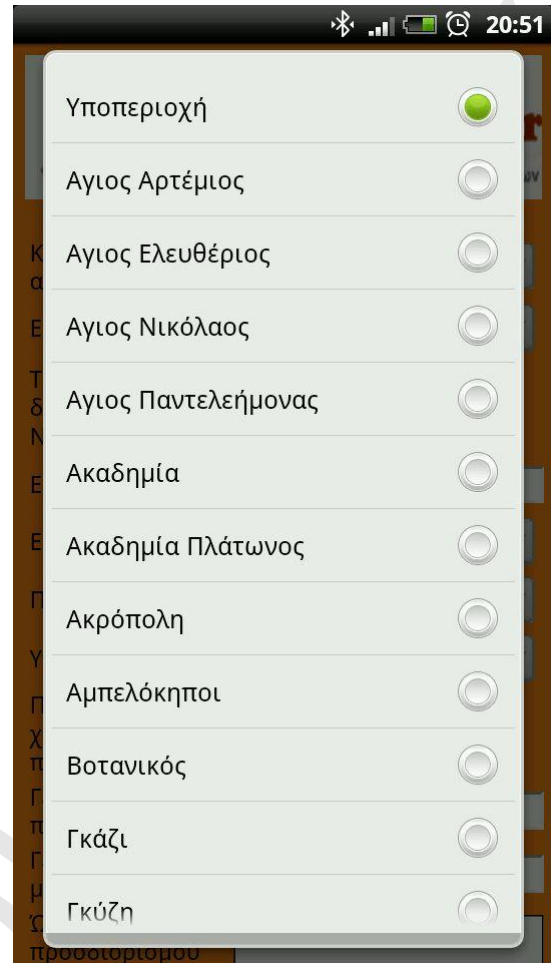
Το γεωγραφικό στίγμα, αποτελεί ένα προαιρετικό πεδίο εισαγωγής, που σημαίνει ότι, δεν θα επηρεάσει την καταχώρηση της αγγελίας, εάν μείνει κενό.

Το γεωγραφικό στίγμα μπορεί να εισαχθεί είτε χρησιμοποιώντας το πληκτρολόγιο (εφόσον είναι γνωστό) ή να επιλεγεί το κουμπί «Εύρεση» όπου θα γίνει προσπάθεια προσδιορισμού του γεωγραφικού στίγματος.

Τα πεδία γεωγραφικό πλάτος και γεωγραφικό μήκος είναι αυτά που θα καταχωρηθούν, εφόσον συμπληρωθούν, ενώ το πεδίο «ώρα προσδιορισμού του στίγματος» είναι ένα βοηθητικό πεδίο, ώστε να διαπιστωθεί, αν το στίγμα που επέστρεψε η συσκευή είναι πρόσφατο, ή παλαιότερο.



Εικόνα 10 - Περιοχή



Εικόνα 11 – Υποπεριοχή

Ο προσδιορισμός του γεωγραφικού στίγματος από την έξυπνη συσκευή, γίνεται κατά προτεραιότητα από το διαδίκτυο. Εάν δεν υπάρχει ενεργή σύνδεση στο διαδίκτυο, ενδέχεται να μην επιστραφεί τιμή στα πεδία, ή να επιστραφεί τιμή που είναι αποθηκευμένη στη συσκευή και δεν ανταποκρίνεται στην τρέχουσα τοποθεσία. Αυτό μπορεί να το επαληθευτεί ελέγχοντας, αν η ώρα προσδιορισμού του γεωγραφικού στίγματος, είναι σωστή.

Εναλλακτικά, αν ο χρήστης βρίσκεται σε ανοιχτό χώρο, μπορείτε να ενεργοποιήσει τον δέκτη GPS της συσκευής, και να γίνει ο προσδιορισμός του γεωγραφικού στίγματος μέσω δορυφόρου.

1.1.3 Καταχώρηση αγγελίας

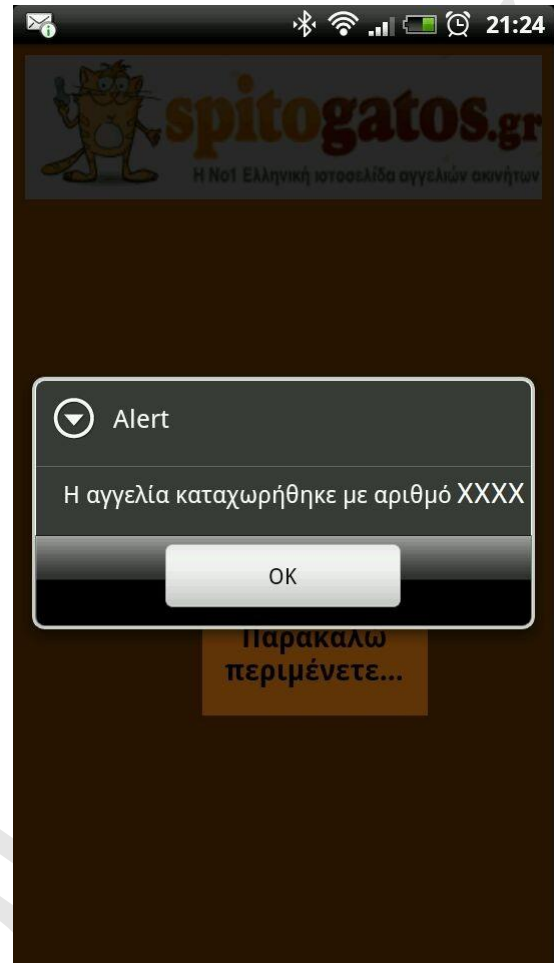
Με την ολοκλήρωση συμπλήρωσης των πεδίων της φόρμας, μπορεί είτε να γίνει καταχώρηση της αγγελίας, είτε να γίνει καθαρισμός των πεδίων αν δεν επιθυμεί ο χρήστης να την καταχωρήσει, επιλέγοντας τα αντίστοιχα κουμπιά «Καταχώρηση» ή «Καθαρισμός».

Σε περίπτωση που έχουν συμπληρωθεί όλα τα πεδία, και υπάρχει ενεργή σύνδεση στο διαδίκτυο, τότε με την επιλογή του κουμπιού «Καταχώρηση» θα απενεργοποιηθεί η επιφάνεια διεπαφής με τον χρήστη, και θα εμφανιστεί μια προσωρινή οθόνη, η οποία ενημερώνει για τον τύπο της σύνδεσης με το διαδίκτυο, και κρατάει την εφαρμογή σε αναμονή (εικόνα 12), μέχρι να επιστρέψει η απάντηση επιτυχούς καταχώρησης από τον ιστότοπο της επιχείρησης (εικόνα 13).

Η απάντηση επιτυχίας, θα επιβεβαιωθεί με την εμφάνιση ενός μηνύματος, με τον μοναδικό αριθμό που έχει αντιστοιχηθεί στην αγγελία.



Εικόνα 12 - Καταχώρηση αγγελίας, οθόνη αναμονής



Εικόνα 13 - Καταχώρηση αγγελίας, επιβεβαίωση καταχώρησης

1.1.4 Καταχώρηση φωτογραφίας

Μετά την επιτυχή καταχώρηση της αγγελίας εμφανίζεται η οθόνη καταχώρησης φωτογραφιών για τη συγκεκριμένη αγγελία, ενώ στο σημείο «XXXX» που εμφανίζεται στην εικόνα 14, αποτελεί το σημείο όπου εμφανίζεται ο μοναδικός αριθμός που έχει αντιστοιχηθεί στην αγγελία που καταχωρήθηκε.

Στη συνέχεια, υπάρχει η δυνατότητα να ληφθεί μία νέα φωτογραφία, ή να ανακτηθεί μία φωτογραφία από το άλμπουμ φωτογραφιών της έξυπνης συσκευής και να γίνει καταχώρηση στην ήδη αποθηκευμένη αγγελία. Με την επιλογή των κουμπιών «Camera» και «Photo Album», δίνεται η δυνατότητα να αποκτήθει πρόσβαση στις αντίστοιχες επιλογές της έξυπνης συσκευής.

Εφόσον επιλεγεί η φωτογραφία, εμφανίζεται μια προεπισκόπηση της επιλεγμένης φωτογραφίας στην επιφάνεια διεπαφής και μπορεί στη συνέχεια να γίνει καταχώρηση στον ιστότοπο της επιχείρησή, στην συγκεκριμένη αγγελία, που μόλις καταχωρήθηκε (εικόνα 15).

Υπάρχει η δυνατότητα να καταχωρηθούν μέχρι 10 φωτογραφίες για μια συγκεκριμένη αγγελία.

! Προσοχή. Μπορείτε να γίνει καταχώρηση μόνο μία φωτογραφία σε κάθε πάτημα του κουμπιού «Αποστολή». Επίσης, η φωτογραφία που καταχωρείται, είναι αυτή που εμφανίζεται στην οθόνη διεπαφής ως προεπισκόπηση σε μικρό μέγεθος της αντίστοιχης φωτογραφίας. Αν επιθυμεί ο χρήστης να αλλάξει φωτογραφία, θα πρέπει να επιλέξει εκ νέου τα κουμπιά «Camera» ή «Photo Album», ώστε να αντικατασταθεί η τρέχουσα επιλεγμένη φωτογραφία.

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας



Εικόνα 14 - Οθόνη προσθήκης φωτογραφιών

Με την επιτυχή καταχώρηση της φωτογραφίας, εμφανίζεται ένα μήνυμα επιβεβαίωσης, όπου ενημερώνεται ο χρήστης για τη θέση στην οποία έχει καταχωρηθεί η αγγελία. Οι φωτογραφίες καταχωρούνται διαδοχικά από τη θέση 1 έως τη θέση 10 για την κάθε αγγελία (εικόνα 16).

1.1.5 Κουμπιά της εφαρμογής

Για την ορθή χρήση της εφαρμογής, πρέπει να είναι γνωστή η λειτουργία του κάθε ενός κουμπιού.

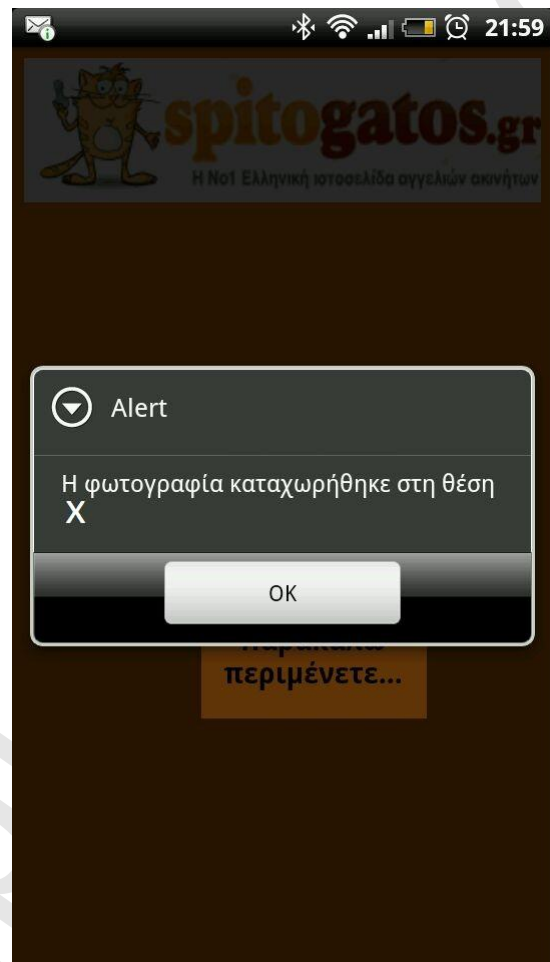
Το κουμπιό εύρεση, προσπαθεί να κάνει ανάκτηση του γεωγραφικού στίγματος, και να εμφανιστεί στην οθόνη καταχώρησης της αγγελίας.

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

Το κουμπί καταχώρηση, κάνει καταχώρηση των στοιχείων που έχουν εισαχθεί, και τα στέλνει στον ιστότοπο της επιχείρησης, προκειμένου να γίνει καταχώρηση και δημοσίευση της αγγελίας.



Εικόνα 15 - Προεπισκόπηση φωτογραφίας προς καταχώρηση



Εικόνα 16 - Επιβεβαίωση επιτυχούς καταχώρησης φωτογραφίας στην διαθέσιμη θέση

Το κουμπί καθαρισμός, καθαρίζει όλα τα στοιχεία της φόρμας, και επαναφέρει την εφαρμογή στην αρχική κατάσταση όπως όταν ήταν κατά την εκκίνησή της.

Το κουμπί μετάβαση, δεν είναι διαθέσιμο από την αρχή χρήσης της εφαρμογής και αν επιλεγεί θα επιστραφεί ένα μήνυμα λάθους-προειδοποίησης. Το κουμπί μετάβαση, οδηγεί το χρήστη στην επόμενη οθόνη, των φωτογραφιών, και χρησιμοποιείται σε συνδυασμό με το κουμπί «επιστροφή» αν είναι επιθυμητή η εναλλαγή της προβολής των οθονών (οθόνη καταχώρησης αγγελίας, οθόνη καταχώρησης φωτογραφίας) εφόσον έχει καταχωρηθεί ήδη η αγγελία. Αντίστοιχα, το κουμπί επιστροφή επιστρέφει τον χρήστη στην αρχική οθόνη της εφαρμογής, όπου μπορεί να ξαναδεί τα δεδομένα που έχουν καταχωρηθεί, ή να κάνει καθαρισμό των πεδίων προκειμένου να καταχωρηθεί νέα αγγελία.

Τα κουμπιά camera και photo album χρησιμοποιούνται για να επιλεγεί η επιθυμητή λειτουργία ανάλογα αν είναι επιθυμητό να αποκτηθεί πρόσβαση στην κάμερα του κινητού τηλεφώνου, ή στο άλμπουμ φωτογραφιών.

Τέλος το κουμπί αποστολή, αποστέλλει την επιλεγμένη φωτογραφία, στον ιστότοπο της επιχείρησης.

1.1.6 Μηνύματα λάθους

Κατά τη χρήση της εφαρμογής, ενδέχεται να προκύψουν κάποια μηνύματα λάθους, τα οποία εμφανίζονται σε ορισμένες συνθήκες.

Αν επιλεγεί να γίνει καταχώρηση μιας αγγελίας, και κάποιο από τα προαπαιτούμενα πεδία δεν έχει συμπληρωθεί, θα εμφανιστεί μήνυμα λάθους που θα υποδεικνύει το σημείο που δεν έχει συμπληρωθεί. Αν είναι περισσότερα από ένα τα πεδία, θα εμφανιστεί σαν λάθος το πρώτο κατά σειρά, και αν στην επόμενη προσπάθεια, τα επόμενα πεδία δεν έχουν καταχωρηθεί, θα εμφανιστεί το επόμενο κατά σειρά πεδίο το οποίο δεν έχει καταχωρηθεί (εικόνα 17).

Αν κατά την καταχώρηση της αγγελίας, δεν υπάρχει σύνδεση με το διαδίκτυο, η εφαρμογή θα εμφανίσει μήνυμα λάθους, το οποίο θα ενημερώνει πως δεν είναι ενεργή η σύνδεση στο διαδίκτυο, και θα πρέπει να ενεργοποιηθεί προκειμένου να γίνει η καταχώρηση της αγγελίας ή της φωτογραφίας (ανάλογα σε ποια οθόνη και ποιο στάδιο καταχώρησης βρίσκεται ο χρήστης), (εικόνα 18).

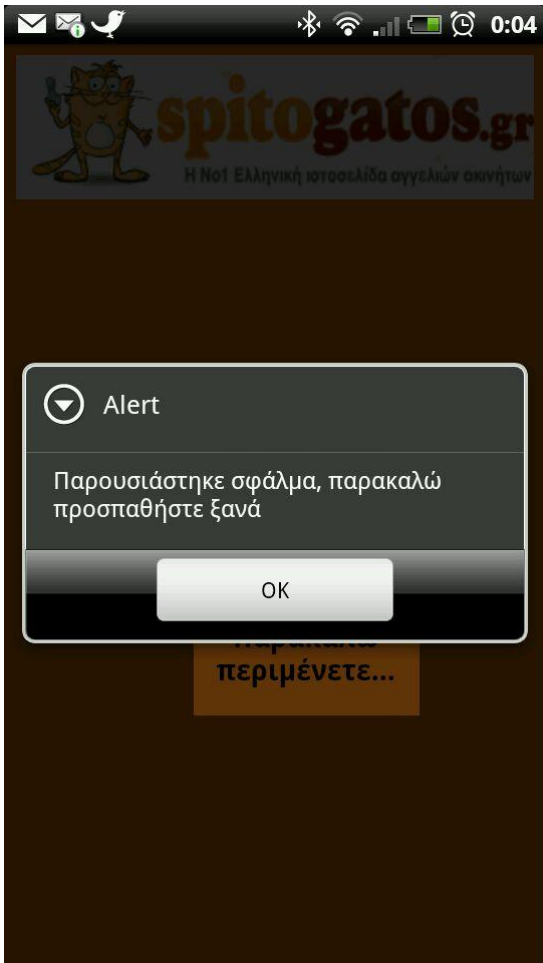
The screenshot shows a mobile application interface for listing a property. At the top, there is a status bar with signal strength, battery, and time (0:46). Below it, a header reads "Η Νο1 Ελληνική ιστοσελίδα αγγελιών ακινήτων". The main heading is "Παρακαλώ συμπληρώστε τη φόρμα:". The form contains several fields: "Κατηγορία ακινήτου" (set to "Κατηγορία"), "Είδος ακινήτου" (set to "Είδος"), "Τιμή (χωρίς δεκαδικά)/Νόμισμα" (set to "Ευρώ"), "Γεωγραφικό πλάτος:", "Γεωγραφικό μήκος:", "Ώρα προσδιορισμού στίγματος:", and two buttons: "Καταχώρηση" and "Καθαρισμός". An "Alert" dialog box is overlaid on the form, displaying the text: "Παρακαλώ επιλέξτε κατηγορία και είδος ακινήτου" and an "OK" button.

Εικόνα 17 - Μήνυμα λάθους, πεδίο που δεν έχει συμπληρωθεί

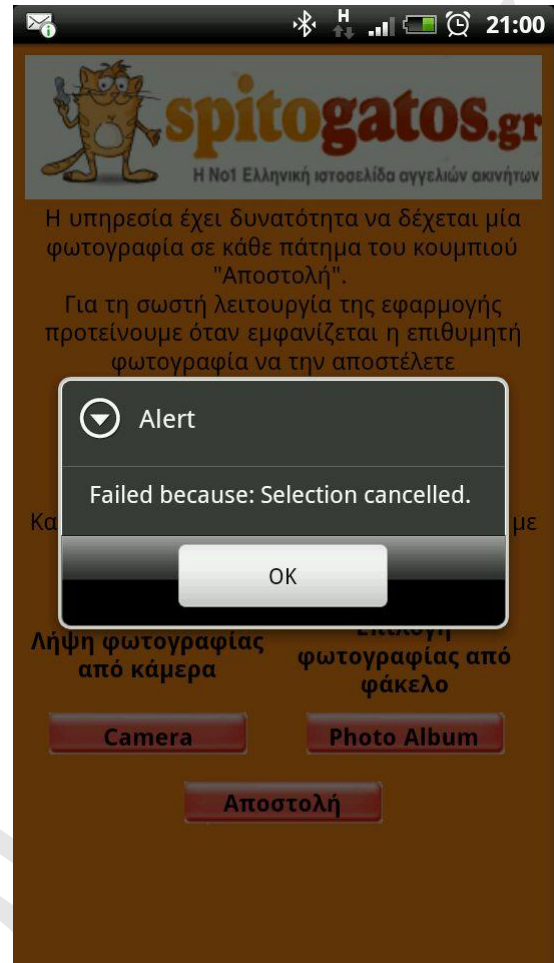
The screenshot shows the same mobile application interface as in Figure 17, but with a different error message. The status bar shows the time as 0:47. The form fields are filled with: "Κατηγορία ακινήτου" (set to "Αξιοποιήσιμη Γη"), "Είδος ακινήτου" (set to "Άλλο"), "Τιμή (χωρίς δεκαδικά)/Νόμισμα" (set to "Ευρώ"), and "Εμβαδό (σε τ.μ.)" (set to "5343"). Below these are fields for "Γεωγραφικό μήκος:", "Ώρα προσδιορισμού στίγματος:", and buttons for "Καταχώρηση" and "Καθαρισμός". An "Alert" dialog box is overlaid, displaying the text: "The web connection of your phone is not working. Please activate internet connection." and an "OK" button.

Εικόνα 18 - Καταχώρηση αγγελίας, επιβεβαίωση καταχώρησης

Αν κατά την καταχώρηση της αγγελίας, προκύψει κάποιο σφάλμα κατά την επικοινωνία με τον ιστότοπο της επιχείρησης, το οποίο μπορεί να οφείλεται σε μη έγκυρα δεδομένα, σφάλμα ταυτοποίησης στοιχείων, ή οποιοδήποτε άλλο λόγο ή πρόβλημα που μπορεί να αντιμετωπίζει η υπηρεσία διαδικτύου του ιστότοπου όπως για παράδειγμα, να μην είναι διαθέσιμη, θα εμφανιστεί μήνυμα λάθους, όπου ενημερώνει ότι προέκυψε κάποιο σφάλμα και θα πρέπει να γίνει προσπάθεια ξανά (εικόνα 19).



Εικόνα 19 - Σφάλμα καταχώρησης κατά την επικοινωνία με τον ιστότοπο

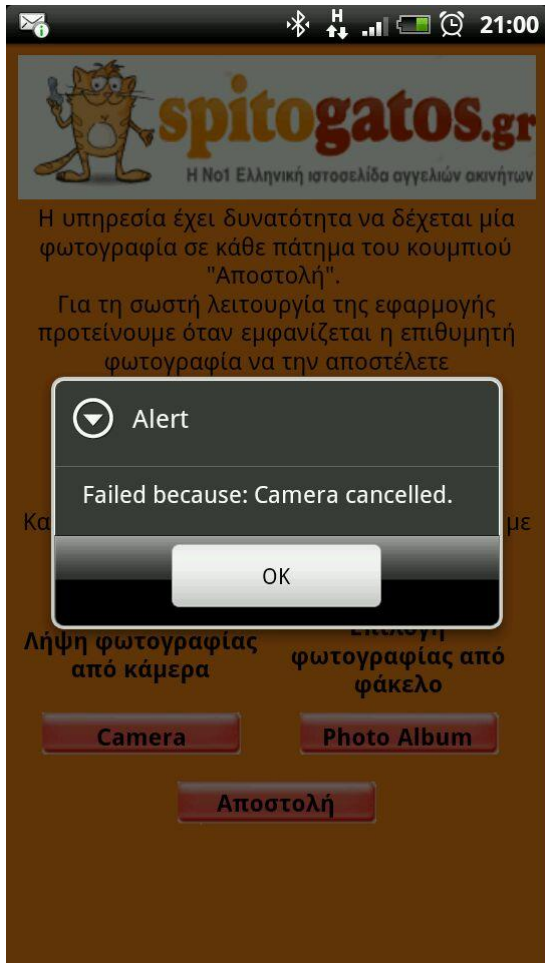


Εικόνα 20 – Ακύρωση χρήσης του άλμπουμ φωτογραφιών

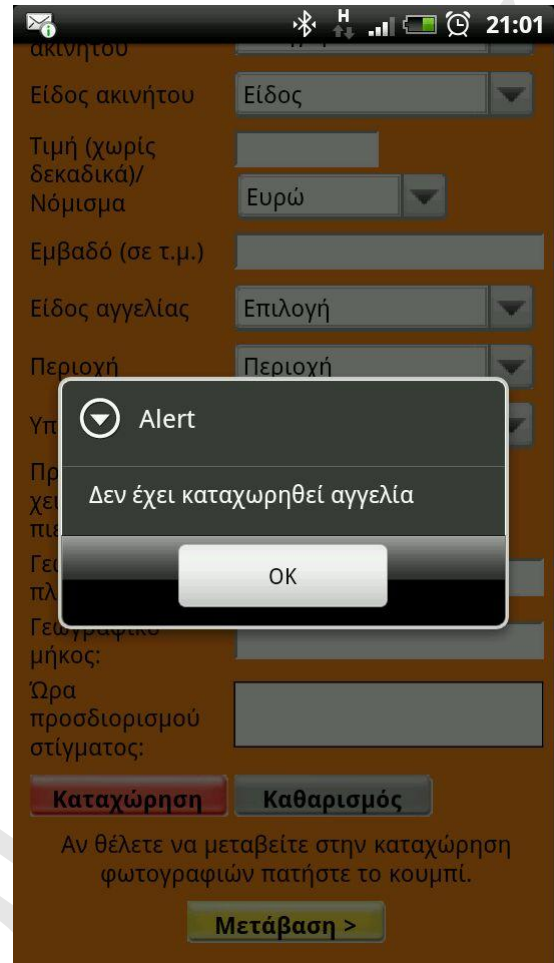
Αν επιλεγεί να γίνει μετάβαση στο photo album, αλλά ακυρωθεί στη συνέχεια η επιλογή, θα εμφανιστεί μήνυμα λάθους, ότι ακυρώθηκε η ενέργειά (εικόνα 20).

Αντίστοιχα, αν επιλεγεί να γίνει μετάβαση στην camera, και ακυρωθεί η επιλογή, θα εμφανιστεί μήνυμα λάθους ότι ακυρώθηκε η λειτουργία λήψης φωτογραφιών από την κάμερα του κινητού (εικόνα 21).

Τέλος, αν επιλεγεί το κουμπί μετάβαση, και δεν έχει καταχωρηθεί αγγελία, δεν θα γίνει μετάβαση στην επόμενη οθόνη, και ότι θα εμφανιστεί ένα μήνυμα λάθους όπου θα ενημερώνει πως δεν έχει καταχωρηθεί κάποια αγγελία.



Εικόνα 21 - Ακύρωση χρήσης της κάμερας



Εικόνα 22 – Μήνυμα λάθους, κατά την επιλογή του κουμπιού μετάβαση

1.2 Εγχειρίδιο προγραμματιστή

1.2.1 Σύνολο αρχείων

Ο κώδικας της εφαρμογής, αποτελείται από τα ακόλουθα αρχεία:

- Index.html
- phonedgap.js
- jquery.js
- jquery.blockUI.js
- mimic.js
- application-js.js
- list.js
- style.css

Από το σύνολο των αρχείων, οι βιβλιοθήκες jquery, jquery.blockUI, και mimic, χρησιμοποιήθηκαν αυτούσιες. Πρόκειται για βιβλιοθήκες ανοιχτού κώδικα, προσβάσιμες από όλους τους χρήστες που επιθυμούν να της μελετήσουν, ενώ σύνδεσμοι για τις τοποθεσίες όπου μπορεί να ανατρέξει ο προγραμματιστής για την τεκμηρίωση των συγκεκριμένων βιβλιοθηκών παρατίθενται στη βιβλιογραφία.

1.2.2 Χρήση των αρχείων

Η βιβλιοθήκη `phonengar.js` περιέχει όλες τις απαραίτητες συναρτήσεις προκειμένου να επιτυγχάνεται η επικοινωνία της εφαρμογής με τις επιμέρους λειτουργίες της έξυπνης συσκευής. Για τον τρόπο χειρισμού της βιβλιοθήκης `phonengar.js` αλλά και τον τρόπο δημιουργίας του εκτελέσιμου αρχείου το οποίο μπορεί να εγκατασταθεί στην έξυπνη συσκευή, θα πρέπει ο προγραμματιστής να ανατρέξει στο εγχειρίδιο που παρέχεται από το `Phonengar`.

Το αρχείο `index.html` πρόκειται για το βασικό αρχείο διεπαφής του χρήστη. Το αρχείο αποτελείται από τα απαραίτητα `html tags` προκειμένου να έχει μια έγκυρη δομή δεδομένων, ενώ στο αρχείο υπάρχει και ένα μικρό κομμάτι κώδικα `JavaScript` το οποίο χειρίζεται την εμφάνιση και την απόκρυψη των επιπλέον πεδίων για τη συμπλήρωση της φόρμας, ανάλογα με το αν επιλέγεται κατοικία και επαγγελματική στέγη ή αξιοποιήσιμη γη. Στα διάφορα πεδία, έχουν αποδοθεί στοιχεία «ταυτότητας» ή «κλάσης» (`id,class`) προκειμένου να γίνεται ειδική διαχείριση στα συγκεκριμένα πεδία, τόσο με τους κανόνες που έχουν οριστεί στο αρχείο μορφοποίησης (`style sheet`) όσο και στο να εκτελούνται συγκεκριμένες εντολές του κώδικα `JavaScript` που χρησιμοποιείται. Στο `html` αρχείο έχει γίνει χρήση παραμέτρων που ισχύουν στην `HTML5`.

Το αρχείο `style.css` περιέχει το σύνολο των κανόνων μορφοποίησης που ορίζουν την εμφάνιση της σελίδας διεπαφής αλλά και των κουμπιών που χρησιμοποιούνται στην εφαρμογή.

Η βιβλιοθήκη `list.js` αποτελεί μια «βιβλιοθήκη δεδομένων» καθώς εκεί είναι αποθηκευμένα τα περισσότερα δεδομένα που εμφανίζονται στα μενού επιλογών (`select boxes/dropdown menus`). Ανάλογα με την επιλογή σε ένα γονικό μενού, παρατίθενται και τα αντίστοιχα στοιχεία που αντιστοιχούν στην επιλογή, στο δευτερεύον μενού που ακολουθεί.

Τέλος, η βιβλιοθήκη `application-js.js` αποτελεί μια βιβλιοθήκη με το σύνολο των συναρτήσεων που χρησιμοποιούνται στη συγκεκριμένη υλοποίηση. Ο κώδικας, παρατίθεται στο παράρτημα 2 με πλήρη σχολιασμό των επιμέρους εντολών και συναρτήσεων που χρησιμοποιούνται, γεγονός που καθιστά πιο εύκολη τη μελέτη από τον προγραμματιστή. Η βιβλιοθήκη περιλαμβάνει συναρτήσεις που αφορούν τόσο την εμφάνιση και λειτουργία της εφαρμογής ως προς τη λειτουργικότητα στα πλαίσια της έξυπνης συσκευής, αλλά και την επικοινωνία με τη υπηρεσία διαδικτύου με τον ιστότοπο της επιχείρησης.

1.2.3 Βιβλιοθήκη `application-js.js`

Κρίθηκε απαραίτητη η χρήση ορισμένων μεταβλητών, οι οποίες να είναι ορατές από το σύνολο των συναρτήσεων, (`global variables`) καθώς μερικές από τις συγκεκριμένες μεταβλητές χρειάζεται να χρησιμοποιηθούν σε περισσότερες από μία συναρτήσεις.

Οι συναρτήσεις που σχετίζονται άμεσα με την αλληλεπίδραση με το `phonengar`, και κατά συνέπεια με τη λειτουργικότητα της εφαρμογής στα πλαίσια τη έξυπνης συσκευής, είναι οι ακόλουθες:

- `function onDeviceReady()`: η συγκεκριμένη συνάρτηση, επαληθεύει την διαθεσιμότητα του `phonengar` να χρησιμοποιηθεί από την εφαρμογή,
- `function onPhotoDataSuccess(imageData)`: η συγκεκριμένη συνάρτηση, χειρίζεται το αποτέλεσμα επιτυχίας των συναρτήσεων `capturePhoto()` και `getPhoto()`, και επιστρέφει στην εφαρμογή και κατ'επέκταση στο χρήστη την επιθυμητή φωτογραφία
- `function capturePhoto()`: η συγκεκριμένη συνάρτηση αλληλεπιδρά με την κάμερα της έξυπνης συσκευής,
- `function getPhoto(source)`: η συγκεκριμένη συνάρτηση αλληλεπιδρά με το άλμπουμ φωτογραφιών της συσκευής,
- `function onFail(message)`: η συγκεκριμένη συνάρτηση χειρίζεται τα λάθη αναφορικά με τη χρήση της κάμερας ή του άλμπουμ φωτογραφιών,
- `function getGeolocation()`: η συγκεκριμένη συνάρτηση κάνει απόπειρα προσδιορισμού του γεωγραφικού στίγματος,
- `function onSuccess(position)`: η συγκεκριμένη συνάρτηση, επιστρέφει το αποτέλεσμα του προσδιορισμού του γεωγραφικού στίγματος στην εφαρμογή και στον χρήστη
- `function onError(error)`: η συγκεκριμένη συνάρτηση επιστρέφει μήνυμα λάθους σε περίπτωση που δεν καταστεί δυνατός ο προσδιορισμός του γεωγραφικού στίγματος

- `function checkConnection()`: η συγκεκριμένη συνάρτηση, ελέγχει την σύνδεση με το διαδίκτυο της έξυπνης συσκευής, και ενημερώνει τον χρήστη για την κατάσταση

Η συνάρτηση `function handleEnter (field, event)` απαγορεύει το να χρησιμοποιηθεί η επιλογή του πλήκτρου «Enter» ως πλήκτρου καταχώρησης της φόρμας. Ο λόγος που χρησιμοποιήθηκε, είναι διότι, κατά την καταχώρηση των πεδίων όπου χρειάζεται να πληκτρολογήσει κάποιες τιμές ο χρήστης, διαπιστώθηκε, πως κατά το πάτημα του πλήκτρου «Enter» ερμηνευόταν από τη φόρμα ως απόπειρα καταχώρησης με αποτέλεσμα να οδηγείται ο χρήστης σε ανεπιθύμητα αποτελέσματα και συνεχή λάθη.

Η συνάρτηση `function isNumberKey(evt)` χρησιμοποιήθηκε στα πεδία όπου δεν πρέπει να καταχωρηθεί διαφορετικός χαρακτήρας εκτός από αριθμό. Η συγκεκριμένη συνάρτηση, απενεργοποιεί τους μη αριθμητικούς χαρακτήρες, με αποτέλεσμα να μην είναι δυνατή η χρήση τους στα πεδία που ζητούνται αριθμητικές τιμές.

Οι συναρτήσεις `function Spitogatosnewlisting()` και `function Spitogatosaddimage()` αποτελούν τις βασικές συναρτήσεις όπου δομούνται τα δεδομένα, προκειμένου να αποσταλούν στην υπηρεσία διαδικτύου (`web service`) μέσω της χρήσης της βιβλιοθήκης `mimic.js`. Οι συναρτήσεις ακολουθούν, τις οδηγίες που ορίζονται στην τεκμηρίωση της `mimic.js` προκειμένου να σταλούν τα δεδομένα σε έγκυρη XML μορφή μέσω του πρωτοκόλλου XML-RPC. Τα δεδομένα που πρέπει να καταχωρηθούν, ορίζονται ως παράμετροι, και αφορούν το όνομα χρήστη, τον κωδικό πρόσβασης, το κλειδί της εφαρμογής, αλλά και τα δεδομένα προς καταχώρηση.

Το σύνολο των δεδομένων που καταχωρούνται από τον χρήστη, μορφοποιείται σαν στοιχείο τύπου «αντικείμενο» `object`, ενώ όλα τα επιμέρους αντικείμενα (`objects`) που δημιουργούνται, δομούνται σε ένα ευρύτερο αντικείμενο (`object`). Το οποίο ορίζεται ως παράμετρος των στοιχείων που πρέπει να καταχωρηθούν. Πρόκειται δηλαδή για μια εμφωλευμένη δομή δεδομένων τύπου «αντικείμενου» (`object`) μέσα σε ένα αντικείμενο (`object`).

Όπως φαίνεται από τον κώδικα, παρόλο που τα απαιτούμενα προς καταχώρηση στοιχεία είναι περιορισμένα, έχουν συμπεριληφθεί όλα τα στοιχεία τα οποία είναι συμβατά με την υπηρεσία διαδικτύου της επιχείρησης, ώστε να είναι εύκολη η χρήση τους, σε μελλοντική επέκταση της εφαρμογής. Το σύνολο των απαιτούμενων αλλά και των έγκυρων στοιχείων, προκύπτει από την τεκμηρίωση που παρείχε η επιχείρηση (`Spitogatos.gr, 2011`).

Τέλος στην `$(document).ready(function()` όπου πρόκειται για μια κατεξοχήν συνάρτηση της `jquery`, γίνεται χειρισμός των κουμπιών που χρησιμοποιούνται στην εφαρμογή, και ποιες λειτουργίες επιτελεί το κάθε ένα κατά την επιλογή του, καταχωρώντας τις αντίστοιχες συναρτήσεις. Επίσης γίνεται έλεγχος διαφόρων πεδίων, καθώς και των τιμών διαφόρων μεταβλητών που χρησιμοποιούνται, όπου ανάλογα με την τιμή ενεργοποιούνται και οι αντίστοιχες ενέργειες.

Παράρτημα II – Ο κώδικας της εφαρμογής

Index.html

```

<!DOCTYPE HTML>
<html>
<head>
<title>Spitogatos.gr</title>
<meta charset="ISO-8859-7">
<script type="text/JavaScript" charset="utf-8" src="phonegap-
1.3.0.js"></script>
<script type="text/JavaScript" charset="utf-8"
src="jquery.js"></script>
<script type="text/JavaScript" charset="utf-8"
src="jquery.blockUI.js"></script>
<script type="text/JavaScript" charset="ISO-8859-7" src="application-
js.js"></script>
<script language="JavaScript" src="list.js"></script>
<script type="text/JavaScript" charset="utf-8"
src="mimic.js"></script>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body onload="fillCategory();" >

<form name="drop_list">
<div id="maintable">
<table width="100%">
<tr>
<th colspan="2" align="center"> Παρακαλώ συμπληρώστε τη
φόρμα:</th>
</tr>
<tr>
<td>Κατηγορία ακινήτου</td>
<td><select name="property" id="property"
onChange="SelectSubprop();" class="box"><Option
value="">Κατηγορία</option></select></td>
</tr>
<tr>
<td>Είδος ακινήτου</td>
<td><select id="Subprop" NAME="Subprop" class="box"><Option
value="">Είδος</option></select></td>
</tr>
<tr>
<td>Τιμή (χωρίς δεκαδικά)/Νόμισμα</td>
<td><input type="number" id="price" name="price"
onkeypress="return isNumberKey(event)" class="smallbox"/><select
name="currency" id="currency">

```

```

    <option selected="selected" value="eur">Ευρώ</option><option
value="usd">Δολάρια</option><option value="gbp">Λίρες
Αγγλίας</option></td>
</tr>
<tr>
    <td>Εμβαδό (σε τ.μ.)</td>
    <td><input type="number" name="sqmeters" id="sqmeters"
onkeypress="return isNumberKey(event)" class="box" /></td>
</tr>
<script type="text/JavaScript">
    $("#property").change(function() {
        if(($ (this).val() == "residential") || ($ (this).val() ==
"commercial")) {
            $("#hiderows").show();
        }
        else {
            $("#hiderows").hide();
        }
    });
</script>
<tbody id="hiderows">
<tr>
    <td>Δωμάτια</td>
    <td><input type="number" name="rooms" id="rooms"
onkeypress="return isNumberKey(event)" class="box" /></td>
</tr>
<tr>
    <td>Έτος κατασκευής</td>
    <td><select name="yearBuilt" id="yearBuilt"
class="box"><option selected="selected"
value="">Επιλογή</option><option value="under construction">Υπο
κατασκευή</option>
        <option value="2012">2012</option><option
value="2011">2011</option><option value="2010">2010</option><option
value="2009">2009</option>
        <option value="2008">2008</option><option
value="2007">2007</option><option value="2006">2006</option><option
value="2005">2005</option>
        <option value="2004">2004</option><option
value="2003">2003</option><option value="2002">2002</option><option
value="2001">2001</option>
        <option value="2000">2000</option><option
value="1999">1999</option><option value="1998">1998</option><option
value="1997">1997</option>
        <option value="1996">1996</option><option
value="1995">1995</option><option value="1994">1994</option><option
value="1993">1993</option>
        <option value="1992">1992</option><option
value="1991">1991</option><option value="1990">1990</option><option
value="1989">1989</option>

```

```
<option value="1988">1988</option><option
value="1987">1987</option><option value="1986">1986</option><option
value="1985">1985</option>
<option value="1984">1984</option><option
value="1983">1983</option><option value="1982">1982</option><option
value="1981">1981</option>
<option value="1980">1980</option><option
value="1979">1979</option><option value="1978">1978</option><option
value="1977">1977</option>
<option value="1976">1976</option><option
value="1975">1975</option><option value="1974">1974</option><option
value="1973">1973</option>
<option value="1972">1972</option><option
value="1971">1971</option><option value="1970">1970</option><option
value="1969">1969</option>
<option value="1968">1968</option><option
value="1967">1967</option><option value="1966">1966</option><option
value="1965">1965</option>
<option value="1964">1964</option><option
value="1963">1963</option><option value="1962">1962</option><option
value="1961">1961</option>
<option value="1960">1960</option><option
value="1959">1959</option><option value="1958">1958</option><option
value="1957">1957</option>
<option value="1956">1956</option><option
value="1955">1955</option><option value="1954">1954</option><option
value="1953">1953</option>
<option value="1952">1952</option><option
value="1951">1951</option><option value="1950">1950</option><option
value="1949">1949</option>
<option value="1948">1948</option><option
value="1947">1947</option><option value="1946">1946</option><option
value="1945">1945</option>
<option value="1944">1944</option><option
value="1943">1943</option><option value="1942">1942</option><option
value="1941">1941</option>
<option value="1940">1940</option><option
value="1939">1939</option><option value="1938">1938</option><option
value="1937">1937</option>
<option value="1936">1936</option><option
value="1935">1935</option><option value="1934">1934</option><option
value="1933">1933</option>
<option value="1932">1932</option><option
value="1931">1931</option><option value="1930">1930</option><option
value="1929">1929</option>
<option value="1928">1928</option><option
value="1927">1927</option><option value="1926">1926</option><option
value="1925">1925</option>
<option value="1924">1924</option><option
value="1923">1923</option><option value="1922">1922</option><option
value="1921">1921</option>
<option value="1920">1920</option></td>
</tr>
```

```

<tr>
  <td>Όροφος</td>
  <td><select name="floor" id="floor" class="box"><option
selected="selected" value="">Επιλογή</option><option
value="basement">Υπόγειο</option>
  <option value="semi basement">Ημιυπόγειο</option><option
value="ground floor">Ισόγειο</option><option value="semi ground
floor">Ημιόροφος</option>
  <option value="1">1ος</option><option
value="2">2ος</option><option value="3">3ος</option><option
value="4">4ος</option><option value="5">5ος</option>
  <option value="6">6ος</option><option
value="7">7ος</option><option value="8">8ος</option><option
value="9">9ος</option><option value="10">10ος</option>
  <option value="11">11ος</option><option
value="12">12ος</option><option value="13">13ος</option><option
value="14">14ος</option><option value="15">15ος</option>
  <option value="16">16ος</option><option
value="17">17ος</option><option value="18">18ος</option><option
value="19">19ος</option><option value="20">20ος</option>
  <option value="21">21ος</option><option
value="22">22ος</option><option value="23">23ος</option><option
value="24">24ος</option><option value="25">25ος</option>
  <option value="26">26ος</option><option
value="27">27ος</option><option value="28">28ος</option><option
value="29">29ος</option><option value="30">30ος</option>
  <option value="31">31ος</option><option
value="32">32ος</option><option value="33">33ος</option><option
value="34">34ος</option><option value="35">35ος</option>
  <option value="36">36ος</option><option
value="37">37ος</option><option value="38">38ος</option><option
value="39">39ος</option><option value="40">40ος</option>
  <option value="41">41ος</option><option
value="42">42ος</option><option value="43">43ος</option><option
value="44">44ος</option><option value="45">45ος</option>
  <option value="46">46ος</option><option
value="47">47ος</option><option value="48">48ος</option><option
value="49">49ος</option><option value="50">50ος</option>
  </select></td>
</tr>
</tbody>
<tr>
  <td>Είδος αγγελίας</td>
  <td><select name="listtype" id="listtype" class="box"><option
selected="selected" value="">Επιλογή</option><option value="for
rent">Ενοικίαση</option><option value="for sale">Πώληση</option></td>
</tr>
<tr>
  <td>Περιοχή</td>
  <td><select name="Region" id="Region" onChange="Selectarea();"
class="box"><option value="">Περιοχή</option></select></td>
</tr>

```

```

<tr>
  <td>Υποπεριοχή</td>
  <td><select id="area" name="area" class="box"><option
value="">Υποπεριοχή</option></select></td>
</tr>
<tr>
  <td>Προσδιορίστε το χειροκίνητα ή πιέστε το κουμπί</td>
  <td align="left"><div id="getgeolocation" class="app_button"
style="display:inline-block; align:left">Εύρεση</div></td>
</tr>
<tr>
  <td>Γεωγραφικό πλάτος: </td>
  <td><input type="number" name="latitude" id="latitude"
onkeypress="return handleEnter(this, event)" class="box"/></td>
</tr>
<tr>
  <td>Γεωγραφικό μήκος: </td>
  <td><input type="number" name="longitude" id="longitude"
onkeypress="return handleEnter(this, event)" class="box"/></td>
</tr>
<tr>
  <td>Ωρα προσδιορισμού στίγματος: </td>
  <td><textarea rows="2" name="timestamp" id="timestamp"
value="" onkeypress="return handleEnter(this, event)"
class="box"></textarea></td>
</tr>
<tr>
  <td align="right"><input type="button" id="submit"
value="Καταχώρηση" class="app_button"></td>
  <td align="left"><input type="reset" value="Καθαρισμός"
class="reset_button" /></td>
</tr>
<tr>
  <td colspan="2" align="center">Αν θέλετε να μεταβείτε στην
καταχώρηση φωτογραφιών πατήστε το κουμπί.</td>
</tr>
<tr>
  <td colspan="2" align="center"><input type="button"
id="forward" value="Μετάβαση >" class="back_button" /></td>
</tr>
</table>
</div>
</form>
<div id="phototable">
<table width="100%">
  <tr>
    <td colspan="2" align="center">Η υπηρεσία έχει δυνατότητα να
δέχεται μία φωτογραφία σε κάθε πάτημα του κουμπιού "Αποστολή". <br />

```



```

        Για τη σωστή λειτουργία της εφαρμογής προτείνουμε όταν
        εμφανίζεται η επιθυμητή φωτογραφία να την αποστείλετε</td>
    </tr>
    <tr>
        <td colspan="2" align="center">Εάν δεν επιθυμείτε την
        καταχώρηση φωτογραφιών επιστρέψτε στη φόρμα</td>
    </tr>
    <tr>
        <td colspan="2" align="center"><input type="button" id="back"
        value="< Επιστροφή" class="back_button" /></td>
    </tr>
    <tr>
        <td colspan="2" align="center">Καταχώρηση φωτογραφιών για την
        αγγελία με αριθμό <div id="listingnumber"></div></td>
    </tr>
    <tr>
        <td align="center"><th>Λήψη φωτογραφίας από κάμερα</th><th>Επιλογή φωτογραφίας
        από φάκελο</th>
    </tr>
    <tr>
        <td align="center"><button onclick="capturePhoto();"
        class="app_button" >Camera</button></td>
        <td align="center"><button
        onclick="getPhoto(pictureSource.SAVEDPHOTOALBUM);"
        class="app_button">Photo Album</button></td>
    </tr>
    <tr>
        <td colspan="2" align="center"><img
        style="display:none;width:80px;height:80px;" id="smallImage" src=""
        /></td>
    </tr>
    <tr>
        <td colspan="2" align="center"><div
        id="send_image">Αποστολή</div></td>
    </tr>
</table>
</div>
<div id="blockmessage" style="display:none;">
<div id="conndiv"></div><br />
<br />
<h3>Παρακαλώ περιμένετε...</h3></div>
</body>
</html>

```

Application-js.js

```

/* Spitogatos.gr functions
 * JavaScript functions related to:
 * a) webservice consumption,

```

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

```

* b) phone's functionality
* are gathered in this .js file
*/
// Global variables
var listingid=0; //variable that holds listing's id -
necessary for add new image in Spitogatosaddimage() function
var image_data=""; //variable that holds base64 photo data
for Spitogatosaddimage() function
var imageData=""; //variable that holds base64 photo data
for integration with Phone's camera
var k=0; //variable that is used as counter to
define slot in Spitogatosaddimage() function
var pictureSource; // picture source
var destinationType; // sets the format of returned value

// Wait for PhoneGap to connect with the device
document.addEventListener("deviceready",onDeviceReady,false);

// PhoneGap is ready to be used!
function onDeviceReady() {
    pictureSource=navigator.camera.PictureSourceType;
    destinationType=navigator.camera.DestinationType;
}

// Disable submit form on Enter press
// This code doesn't allow form to be submitted on pressing the
Enter button
// but in focuses on the next available input field that has to be
filled in
function handleEnter (field, event) {
    var keyCode = event.keyCode ? event.keyCode : event.which ?
event.which : event.charCode;
    if (keyCode == 13) { //Enter's charCode is number "13"
        var i;
        for (i = 0; i < field.form.elements.length; i++)
            if (field == field.form.elements[i])
                break;
        i = (i + 1) % field.form.elements.length;
        field.form.elements[i].focus();
        return false;
    }
    else
        return true;
}

// Disable non-numeric buttons
// This code, doesn't allow buttons that are non numeric to be
entered in the field
function isNumberKey(evt)

```

```

    {
        var charCode = (evt.which) ? evt.which : event.keyCode
        if (charCode > 31 && (charCode < 48 || charCode > 57))
            return false;

        return true;
    }
    // Called when a photo is successfully retrieved
    function onPhotoDataSuccess(imageData) {
        // Uncomment to view the base64 encoded image data
        //console.log(imageData);
        // Get image handle
        var smallImage = document.getElementById('smallImage');
        // Unhide image elements
        smallImage.style.display = 'block';
        // Show the captured photo
        // The inline CSS rules are used to resize the image
        smallImage.src = "data:image/jpeg;base64," + imageData;
        image_data = imageData; //base64 data are parsed to the
        variable that is used in the Spitogatosaddimage() function
    }
    // A button will call this function
    function capturePhoto() {
        // Take picture using device camera and retrieve image as
        base64-encoded string
        navigator.camera.getPicture(onPhotoDataSuccess, onFail, {
            quality: 50 });
    }

    // A button will call this function
    // This is disabled, because not all devices support the feature
    of capturing ediable photo
    /* function capturePhotoEdit() {
        // Take picture using device camera, allow edit, and retrieve
        image as base64-encoded string
        navigator.camera.getPicture(onPhotoDataSuccess, onFail, {
            quality: 20, allowEdit: true });
    }
    */
    // A button will call this function
    function getPhoto(source) {
        // Retrieve image file location from specified source
        navigator.camera.getPicture(onPhotoDataSuccess, onFail, {
            quality: 50,
            destinationType: destinationType.DATA_URL,
            sourceType: source });
    }
    // Called if something bad happens.

```

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

```

function onFail(message) {
    alert('Failed because: ' + message);
}
function getgeolocation() {
    // Geolocation.usePhoneGap();
    navigator.geolocation.getCurrentPosition(onSuccess, onError);
}
// onSuccess Geolocation
// Geolocation data are parsed to the appropriate input fields
function onSuccess(position) {
    var number = document.getElementById('latitude');
    number.value = position.coords.latitude;
    var number = document.getElementById('longitude');
    number.value = position.coords.longitude;
    var number = document.getElementById('timestamp');
    number.value = new Date(position.timestamp)
}
// onError Callback receives a PositionError object
function onError(error) {
    alert('code: ' + error.code + '\n' +
        'message: ' + error.message + '\n');
}
// Check Internet Connection
// This function checks internet connection.
// If no internet connection is spotted it alerts a message and
returns false.
// If internet connection is spotted it parses the type of the
connection to the blockUI interface
function checkConnection(){
    var networkState = navigator.network.connection.type;
    var states = {};
    states[Connection.UNKNOWN] = 'Unknown connection';
    states[Connection.ETHERNET] = 'Ethernet connection';
    states[Connection.WIFI] = 'WiFi connection';
    states[Connection.CELL_2G] = 'Cell 2G connection';
    states[Connection.CELL_3G] = 'Cell 3G connection';
    states[Connection.CELL_4G] = 'Cell 4G connection';
    states[Connection.NONE] = 'No network connection';
    if(states[networkState] == 'No network connection') {
        alert("The web connection of your phone is not
working. Please activate internet connection.");
        return false;
    }
    else if (states[networkState] == 'Unknown connection'){
        var conntype = document.getElementById('conndiv');
        conntype.innerHTML = 'Connection type cannot be
identified. If request takes too long, please terminate the
application and try again.' + states[networkState];
    }
}

```

```

        return true;}
    else {
        var conntype = document.getElementById('conndiv');
        conntype.innerHTML = 'Connection type: ' +
states[networkState];
        return true;}
    }
    // Webservice consumption using XML-RPC to add new listing
    function Spitogatosnewlisting() {
        var obj1a = new Object();
        obj1a.areaID = parseInt($("#area").val(),10);
        // following minidified data for future use //
        // obj1a.displayAddress = "no";
        // geolocation is parsed as long as timestamp field is
        identified as "not empty"
        if ( $('#timestamp').val() != "" ){
            obj1a.geocodeType = "offset";
            obj1a.latitude = parseFloat($("#latitude").val());
            obj1a.longitude = parseFloat($("#longitude").val());
        }
        //following minidified data for future use //
        //obj1a.streetAddressGR = "test4"; //obj1a.streetAddressEN
= "test4"; //obj1a.zip = "test6";

        var obj1b = new Object();
        obj1b.category = $("#property").val();
        //obj1b.fullBathrooms = 6; //obj1b.halfBathrooms = 3;
        obj1b.livingArea = parseInt($("#sqmeters").val(),10);
        //obj1b.lotSize = 97;
        obj1b.propertyType = $("#Subprop").val();
        if ( $('#property').val() != "land" ){
            obj1b.rooms = parseInt($("#rooms").val(),10);
            obj1b.yearBuilt = $("#yearBuilt").val();}

        var obj1c = new Object();
        //obj1c.brokerListingID = "wqvcaceghnty";
        //obj1c.closingPrice = 7845;
        obj1c.currency = $("#currency").val();
        //following minidified data for future use //
        //obj1c.dateAvailable = "19/3/2009"; //obj1c.dateSold =
"14/12/2005"; //obj1c.descriptionGR = "test1"; //obj1c.descriptionEN =
"test2"; //obj1c.exchangeScheme = "yes";
//obj1c.exchangeSchemePercentage = 6.8; //obj1c.holidayHome = "no";
//obj1c.initialPrice = 987;
        obj1c.listingType = $("#listtype").val();
        obj1c.price = parseInt($("#price").val(),10);
        //following minidified data for future use //

```

```

//obj1c.published = "yes"; //obj1c.publishedBrokerSite =
"yes"; //obj1c.shortTermLetting = "yes"; //obj1c.status = "contract";
//obj1c.newDevelopment = "yes";

//following minidified data for future use //
//var obj1d = new Object();
//obj1d.landingPageURL = "http://www.spitogatos.gr";
//obj1d.virtualTourURL = "http://www.spitogatos.gr";

var obj1e = new Object();
//following minidified data for future use //
//obj1e.airConditioning = "no"; //obj1e.alarm = "yes";
//obj1e.attic = "no"; //obj1e.balcony = "yes";
//obj1e.buildingCoefficient = 4.6; //obj1e.cableReady = "yes";
//obj1e.corner = "yes"; //obj1e.elevator = "yes"; //obj1e.facade =
"no"; //obj1e.fireplace = "yes";
if ( $('#property').val() != "land" )
obj1e.floorNumber = $("#floor").val();
//following minidified data for future use //
//obj1e.freightElevator = "yes"; //obj1e.furnished =
"yes"; //obj1e.furredCeiling = "no"; //obj1e.garage = "no";
//obj1e.garden = "yes"; //obj1e.heatingSystem = "autonomous";
//obj1e.internalStairs = "yes"; //obj1e.loadingDock = "no";
//obj1e.penthouse = "no"; //obj1e.petsAllowed = "yes";
//obj1e.playroom = "no"; //obj1e.residentialArea = "no";
//obj1e.roadType = "dirt road"; //obj1e.satelliteReceiver = "yes";
//obj1e.secureDoor = "yes"; //obj1e.solarWaterHeating = "yes";
//obj1e.storageSpace = "no"; //obj1e.style = "Tudor";
//obj1e.swimmingPool = "yes"; //obj1e.view = "yes";
//obj1e.withinCityPlan = "no"; //obj1e.zoning = "residential";

var obj1 = new Object()
obj1.location = obj1a;
obj1.propertyDetails = obj1b;
obj1.listingDetails = obj1c;
//obj1.links = obj1d; //deactivated due to lack of
elements
obj1.detailedCharacteristics = obj1e;

var method = "sync.newListing";
var request = new XmlRpcRequest("http://www.sp-
orfeas.com/svn/trunk/public_html/webservices/listingSync/v1_0",
method);
request.addParam("XXXXXXXXXXXXXXXXXXXXXXXXXXXX"); //
Application Key
request.addParam(0); // User ID
request.addParam("XXXXXXXXXXXX"); // Username
request.addParam("XXXXXXXXXXXXXXXXXXXXXXXXXXXX"); //
Password
request.addParam(obj1); // Listing
Data

```

```

        try { // Use of
try/catch, in order to handle unexpected errors
        var response = request.send(); // making
the remote call
        var resp = response.parseXML(); //
handling the response
        $.unlockUI(); // unlock
user interface
        alert("Η αγγελία καταχωρήθηκε με αριθμό " + resp);
        $("#phototable").show();
        var listing = document.getElementById('listingnumber');
        listing.innerHTML = resp; // parse
listing id to user interface
        listingid = parseInt(resp,10);
        }
        catch(err) // display
error message in case of unexpected faulure
        {
        $.unlockUI(); // unlock
user interface
        alert("Παρουσιάστηκε σφάλμα, παρακαλώ προσπαθήστε ξανά");
        $("#maintable").show();
        }
    }

    // Webservice consumption using XML-RPC to add new image
    function Spitogatosaddimage() {
        k++; //
increment variable by one to count how many times this function has
been called
        var method = "sync.addImage";
        var request = new XmlRpcRequest("http://www.sp-
orfeas.com/svn/trunk/public_html/webservices/listingSync/v1_0",
method);
        request.addParam("XXXXXXXXXXXXXXXXXXXXXXXXXXXX"); //
Application Key
        request.addParam("XXXXXXXXXXXXXXXXXXXX"); //
Username
        request.addParam("XXXXXXXXXXXXXXXXXXXXXXXXXXXX"); //
Password
        request.addParam(listingid); // Listing
number
        request.addParam(k); // Slot
number
        request.addParam(image_data); // Base64
data
        try { // Use of
try/catch, in order to handle unexpected errors
        var response = request.send(); // making
the remote call
        var resp = response.parseXML(); //
handling the response
        $.unlockUI(); // unlock
user interface
        alert("Η φωτογραφία καταχωρήθηκε στη θέση " + resp);
        $("#phototable").show();
        }
    }

```

```

        catch(err) // display
error message in case of unexpected faulure
        {
            $.unblockUI(); // unblock
user interface
            alert("Παρουσιάστηκε σφάλμα, παρακαλώ προσπαθήστε ξανά");
            $("#phototable").show();
        }
    }
}

```

```

$(document).ready(function() {
    $("#hiderows").hide(); // on
start-up #hiderows is hidden
    $("#phototable").hide(); // on
start-up #phototable is hidden
    // Geolocation button function
    $("#getgeolocation").on("click", function(){ // trigger
geolocation function
        getgeolocation();});
    // Submit form button function
    $("#submit").on("click", function(){ // checks
before submitting form by calling Spitogatosnewlisting() function
        if ( $('#Subprop').val() == "" )
            alert("Παρακαλώ επιλέξτε κατηγορία και είδος ακινήτου");
        else if ( $('#price').val() == "" )
            alert("Παρακαλώ εισάγετε τιμή");
        else if ( $('#sqmeters').val() == "" )
            alert("Παρακαλώ εισάγετε εμβαδό σε τετραγωνικά μέτρα");
        else if (( $('#property').val() != "land" ) && (
$('#rooms').val() == "" ))
            alert("Παρακαλώ εισάγετε αριθμό δωματίων");
        else if (( $('#property').val() != "land" ) && (
$('#yearBuilt').val() == "" ))
            alert("Παρακαλώ εισάγετε έτος κατασκευής");
        else if (( $('#property').val() != "land" ) && (
$('#floor').val() == "" ))
            alert("Παρακαλώ εισάγετε τον όροφο");
        else if ( $('#listtype').val() == "" )
            alert("Παρακαλώ εισάγετε είδος αγγελίας");
        else if ( $('#area').val() == "" )
            alert("Παρακαλώ εισάγετε Περιοχή και Δήμο");
        else {
            if(checkConnection()){ // check
whether there is a live internet connection
                $("#maintable").hide(); //
#maintable is hidden while parsing data to webservice
                k=0; // reset
Spitogatosaddimage() counter
                $.blockUI({ message: $('#blockmessage'),css: {
backgroundColor: '#F88017', border: '0px', width: '150px' } });

```



```

        setTimeout (function () {Spitogatosnewlisting () },1000);}
    }
});
// Reset button function
$("#reset").on("click", function () { // brings
application to initial state
    $("#hiderows").hide (); // hide
appropriate fields
    $("#phototable").hide ();
    var resetprop = document.getElementById('property');// reset
selected data
    resetprop.value = ""; // reset
selected data
    SelectSubprop(""); // reset
selected data
    var resetreg = document.getElementById('Region'); // reset
selected data
    resetreg.value = ""; // reset
selected data
    Selectarea(""); // reset
selected data
    k=0; // reset
Spitogatosaddimage () counter
    listingid=0; // reset
listing ID data
    image_data=""; // reset
variable with base64 data
    imageData=""; // reset
variable with base64 data

document.getElementById("smallImage").style.display="none";});
// Submit image button function
$("#send_image").on("click", function () {
    if (k<10)
        if (image_data == "")
            alert ("Δεν υπάρχει επιλεγμένη φωτογραφία");
        else {
            if(checkConnection()){ // check
whether there is a live internet connection
                $("#phototable").hide (); //
#phototable is hidden while parsing data to webservice
                $.blockUI({ message: $('#blockmessage'), css: {
backgroundColor: '#F88017', border: '0px', width: '150px' } });
setTimeout (function () {Spitogatosaddimage () },1000);}
            }
        else
            alert ("Όλες οι θέσεις είναι γεμάτες");});
// Forward button function
$("#forward").on("click", function () {

```

Εγκαταστάσιμη σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης δεδομένων μέσω διαδικτυακής υπηρεσίας

```

        if ( listingid== 0 )
            alert ("Δεν έχει καταχωρηθεί αγγελία");
        else {
            $("#maintable").hide();
            $("#phototable").show();
        });
        // Back button function
        $("#back").on("click", function() {
            $("#phototable").hide();
            $("#maintable").show();
        });
    });

```

List.js

```

function fillCategory() {
    // this function is used to fill the Region list on load
    addOption(document.drop_list.Region, "100", "Αθήνα - Κέντρο", "");
    addOption(document.drop_list.Region, "101", "Αθήνα - Βόρεια Προάστια", "");
    addOption(document.drop_list.Region, "102", "Αθήνα - Νότια Προάστια", "");
    addOption(document.drop_list.Region, "103", "Αθήνα - Δυτικά Προάστια", "");
    addOption(document.drop_list.Region, "104", "Αθήνα - Ανατολικά Προάστια", "");
    addOption(document.drop_list.Region, "105", "Πειραιάς", "");
    addOption(document.drop_list.Region, "106", "Προάστια Πειραιά", "");
    addOption(document.drop_list.Region, "107", "Υπόλοιπο Αττικής", "");
    addOption(document.drop_list.Region, "108", "Θεσσαλονίκη - Κέντρο", "");
    addOption(document.drop_list.Region, "109", "Θεσσαλονίκη - Περιφ/κοι δήμοι", "");
    addOption(document.drop_list.Region, "110", "Θεσσαλονίκη - Υπόλ.Νομού", "");
    addOption(document.drop_list.Region, "150", "Αιτωλοακαρνανία", "");
    addOption(document.drop_list.Region, "151", "Αργολίδα", "");
    addOption(document.drop_list.Region, "152", "Αρκαδία", "");
    addOption(document.drop_list.Region, "153", "Αρτα", "");
    addOption(document.drop_list.Region, "154", "Αχαΐα", "");
    addOption(document.drop_list.Region, "155", "Βοιωτία", "");
    addOption(document.drop_list.Region, "156", "Τρεβενά", "");
    addOption(document.drop_list.Region, "157", "Δράμα", "");
    addOption(document.drop_list.Region, "158", "Δωδεκάνησα", "");
    addOption(document.drop_list.Region, "159", "Εβρος", "");
    addOption(document.drop_list.Region, "160", "Εύβοια", "");
    addOption(document.drop_list.Region, "161", "Ευρυτανία", "");
}

```

```
addOption (document.drop_list.Region, "162", "Ζάκυνθος", "");  
addOption (document.drop_list.Region, "163", "Ηλεία", "");  
addOption (document.drop_list.Region, "164", "Ημαθία", "");  
addOption (document.drop_list.Region, "165", "Ηράκλειο - Κρήτη", "");  
addOption (document.drop_list.Region, "166", "Θεσπρωτία", "");  
addOption (document.drop_list.Region, "167", "Ιωάννινα", "");  
addOption (document.drop_list.Region, "168", "Καβάλα", "");  
addOption (document.drop_list.Region, "169", "Καρδίτσα", "");  
addOption (document.drop_list.Region, "170", "Καστοριά", "");  
addOption (document.drop_list.Region, "171", "Κέρκυρα", "");  
addOption (document.drop_list.Region, "172", "Κεφαλλονιά", "");  
addOption (document.drop_list.Region, "173", "Κιλκίς", "");  
addOption (document.drop_list.Region, "174", "Κοζάνη", "");  
addOption (document.drop_list.Region, "175", "Κορινθία", "");  
addOption (document.drop_list.Region, "176", "Κυκλάδες", "");  
addOption (document.drop_list.Region, "177", "Λακωνία", "");  
addOption (document.drop_list.Region, "178", "Λαάρισα", "");  
addOption (document.drop_list.Region, "179", "Λασιθι - Κρήτη", "");  
addOption (document.drop_list.Region, "180", "Λέσβος", "");  
addOption (document.drop_list.Region, "181", "Λευκάδα", "");  
addOption (document.drop_list.Region, "182", "Μαγνησία", "");  
addOption (document.drop_list.Region, "183", "Μεσσηνία", "");  
addOption (document.drop_list.Region, "184", "Ξάνθη", "");  
addOption (document.drop_list.Region, "185", "Πέλλα", "");  
addOption (document.drop_list.Region, "186", "Πιερία", "");  
addOption (document.drop_list.Region, "187", "Πρέβεζα", "");  
addOption (document.drop_list.Region, "188", "Ρέθυμνο - Κρήτη", "");  
addOption (document.drop_list.Region, "189", "Ροδόπη", "");  
addOption (document.drop_list.Region, "190", "Σάμος", "");  
addOption (document.drop_list.Region, "191", "Σέρρες", "");  
addOption (document.drop_list.Region, "192", "Τρίκαλα", "");  
addOption (document.drop_list.Region, "193", "Φθιώτιδα", "");  
addOption (document.drop_list.Region, "194", "Φλώρινα", "");  
addOption (document.drop_list.Region, "195", "Φωκίδα", "");  
addOption (document.drop_list.Region, "196", "Χαλκιδική", "");  
addOption (document.drop_list.Region, "197", "Χανιά - Κρήτη", "");  
addOption (document.drop_list.Region, "198", "Ϊίος", "");  
addOption (document.drop_list.Region, "301", "Κύπρος", "");  
addOption (document.drop_list.Region, "401", "Αργεντινή", "");  
addOption (document.drop_list.Region, "402", "Αυστραλία", "");  
addOption (document.drop_list.Region, "403", "Αυστρία", "");  
addOption (document.drop_list.Region, "404", "Βέλγιο", "");  
addOption (document.drop_list.Region, "405", "Βουλγαρία", "");  
addOption (document.drop_list.Region, "406", "Γαλλία", "");  
addOption (document.drop_list.Region, "407", "Γερμανία", "");  
addOption (document.drop_list.Region, "408", "Ελβετία", "");
```

```

addOption (document.drop_list.Region, "409", "Ηνωμένα Αραβικά Εμιράτα",
"");
addOption (document.drop_list.Region, "410", "Ηνωμένες Πολιτείες", "");
addOption (document.drop_list.Region, "411", "Ισπανία", "");
addOption (document.drop_list.Region, "412", "Ιταλία", "");
addOption (document.drop_list.Region, "413", "Καναδάς", "");
addOption (document.drop_list.Region, "414", "Κροατία", "");
addOption (document.drop_list.Region, "415", "Μ.Βρετανία - Λονδίνο",
"");
addOption (document.drop_list.Region, "416", "Μ.Βρετανία - Εκτός
Λονδίνου", "");
addOption (document.drop_list.Region, "417", "Μαυροβούνιο", "");
addOption (document.drop_list.Region, "418", "Ολλανδία", "");
addOption (document.drop_list.Region, "419", "Πολωνία", "");
addOption (document.drop_list.Region, "420", "Πορτογαλία", "");
addOption (document.drop_list.Region, "421", "Ρουμανία", "");
addOption (document.drop_list.Region, "422", "Ρωσία", "");
addOption (document.drop_list.Region, "423", "Σερβία", "");
addOption (document.drop_list.Region, "424", "Σλοβενία", "");
addOption (document.drop_list.Region, "425", "Σουηδία", "");
addOption (document.drop_list.Region, "426", "Τουρκία", "");
addOption (document.drop_list.Region, "427", "Τσεχία", "");
addOption (document.drop_list.Region, "500", "Λοιπές χώρες Εξωτερικού",
"");
addOption (document.drop_list.property, "residential", "Κατοικία", "");
addOption (document.drop_list.property, "commercial", "Επαγγελματικό
Ακίνητο", "");
addOption (document.drop_list.property, "land", "Αξιοποιήσιμη Γη", "");
}

```

```

function SelectSubprop () {
// ON selection of Region this function will work

removeAllOptions (document.drop_list.Subprop);
addOption (document.drop_list.Subprop, "", "Είδος", "");

if (document.drop_list.property.value == 'residential') {
addOption (document.drop_list.Subprop, "apartment", "Διαμέρισμα");
addOption (document.drop_list.Subprop, "studio", "Ενιαίος χώρος");
addOption (document.drop_list.Subprop, "maisonette", "Μεζονέτα");
addOption (document.drop_list.Subprop, "detached", "Μονοκατοικία");
addOption (document.drop_list.Subprop, "other residential", "Άλλο");
}

if (document.drop_list.property.value == 'commercial') {
addOption (document.drop_list.Subprop, "office", "Γραφείο");
addOption (document.drop_list.Subprop, "store", "Κατάστημα");
}

```

```
addOption (document.drop_list.Subprop,"warehouse" , "Αποθήκη");  
addOption (document.drop_list.Subprop,"other commercial" , "Άλλο");  
}
```

```
if(document.drop_list.property.value == 'land'){  
addOption (document.drop_list.Subprop,"plot" , "Οικόπεδο");  
addOption (document.drop_list.Subprop,"parcel" , "Αγροτεμάχιο");  
addOption (document.drop_list.Subprop,"other land" , "Άλλο");  
}  
}
```

```
function Selectarea () {
```

```
// ON selection of Region this function will work
```

```
removeAllOptions (document.drop_list.area);  
addOption (document.drop_list.area, "", "Υποπεριοχή", "");
```

```
if(document.drop_list.Region.value == '100'){  
addOption (document.drop_list.area,"2000" , "Άγιος Αρτέμιος");  
addOption (document.drop_list.area,"2001" , "Άγιος Ελευθέριος");  
addOption (document.drop_list.area,"2002" , "Άγιος Νικόλαος");  
addOption (document.drop_list.area,"2003" , "Άγιος Παντελεήμονας");  
addOption (document.drop_list.area,"2004" , "Ακαδημία");  
addOption (document.drop_list.area,"2005" , "Ακαδημία Πλάτωνος");  
addOption (document.drop_list.area,"2006" , "Ακρόπολη");  
addOption (document.drop_list.area,"2007" , "Αμπελόκηποι");  
addOption (document.drop_list.area,"2008" , "Βοτανικός");  
addOption (document.drop_list.area,"2009" , "Γκάζι");  
addOption (document.drop_list.area,"2010" , "Γκύζη");  
addOption (document.drop_list.area,"2011" , "Γουδί");  
addOption (document.drop_list.area,"2012" , "Ελληνορώσων");  
addOption (document.drop_list.area,"2013" , "Εξάρχεια");  
addOption (document.drop_list.area,"2014" , "Θησείο");  
addOption (document.drop_list.area,"2015" , "Ιλίσια");  
addOption (document.drop_list.area,"2016" , "Καλλιμάρμαρο");  
addOption (document.drop_list.area,"2017" , "Κεραμικός");  
addOption (document.drop_list.area,"2018" , "Κολοκυνθού");  
addOption (document.drop_list.area,"2019" , "Κολωνάκι");  
addOption (document.drop_list.area,"2020" , "Κολωνός");  
addOption (document.drop_list.area,"2021" , "Κουκάκι");  
addOption (document.drop_list.area,"2022" , "Κυψέλη");  
addOption (document.drop_list.area,"2023" , "Λεωφόρος Αχαρνών");  
addOption (document.drop_list.area,"2024" , "Λεωφόρος Λιοσίων");  
addOption (document.drop_list.area,"2025" , "Λεωφόρος Πατησίων");  
addOption (document.drop_list.area,"2026" , "Λόφος Σκουζέ");  
addOption (document.drop_list.area,"2027" , "Λυκαβηττός");
```

```
addOption(document.drop_list.area,"2028", "Μακρυγιάννη");
addOption(document.drop_list.area,"2029", "Μεταξουργείο");
addOption(document.drop_list.area,"2030", "Μετς");
addOption(document.drop_list.area,"2031", "Μοναστηράκι");
addOption(document.drop_list.area,"2032", "Νεάπολη");
addOption(document.drop_list.area,"2033", "Νέος Κόσμος");
addOption(document.drop_list.area,"2034", "Ομόνοια");
addOption(document.drop_list.area,"2035", "Παγκράτι");
addOption(document.drop_list.area,"2036", "Πατήσια");
addOption(document.drop_list.area,"2037", "Πεδίον Άρεως");
addOption(document.drop_list.area,"2038", "Πετράλωνα");
addOption(document.drop_list.area,"2039", "Πλάκα");
addOption(document.drop_list.area,"2040", "Πλατεία Αμερικής");
addOption(document.drop_list.area,"2041", "Πλατεία Αττικής");
addOption(document.drop_list.area,"2042", "Πλατεία Βάθης");
addOption(document.drop_list.area,"2043", "Πλατεία Βικτωρίας");
addOption(document.drop_list.area,"2044", "Πλατεία Κάνιγγος");
addOption(document.drop_list.area,"2045", "Πλατεία Κλαυθμώνος");
addOption(document.drop_list.area,"2046", "Πλατεία Κολιάτσου");
addOption(document.drop_list.area,"2047", "Πλατεία Μαβίλη");
addOption(document.drop_list.area,"2048", "Πολύγωνο");
addOption(document.drop_list.area,"2049", "Ριζούπολη");
addOption(document.drop_list.area,"2050", "Ρουφ");
addOption(document.drop_list.area,"2051", "Σεπόλια");
addOption(document.drop_list.area,"2052", "Σταθμός Λαρίσης");
addOption(document.drop_list.area,"2053", "Σύνταγμα");
addOption(document.drop_list.area,"2054", "Φιλοπάππου");
addOption(document.drop_list.area,"2055", "Χίλτον");
addOption(document.drop_list.area,"2056", "Ψυρρή");
addOption(document.drop_list.area,"2099", "Υπόλοιπο κέντρο Αθήνας");
}
if(document.drop_list.Region.value == '101'){
addOption(document.drop_list.area,"2100", "Αγ. Στέφανος");
addOption(document.drop_list.area,"2101", "Αγία Παρασκευή");
addOption(document.drop_list.area,"2102", "Ανοιξη");
addOption(document.drop_list.area,"2103", "Βριλήσσια");
addOption(document.drop_list.area,"2104", "Διόνυσος");
addOption(document.drop_list.area,"2105", "Δροσιά");
addOption(document.drop_list.area,"2106", "Εκάλη");
addOption(document.drop_list.area,"2107", "Ηράκλειο");
addOption(document.drop_list.area,"2108", "Κηφισιά");
addOption(document.drop_list.area,"2109", "Κρυονέρι");
addOption(document.drop_list.area,"2110", "Λυκόβρυση");
addOption(document.drop_list.area,"2111", "Μαρούσι");
addOption(document.drop_list.area,"2112", "Μελίσσια");
addOption(document.drop_list.area,"2113", "Νέα Ερυθραία");
```

```
addOption (document.drop_list.area, "2114" , "Νέα Πεντέλη");
addOption (document.drop_list.area, "2115" , "Νέο Ψυχικό");
addOption (document.drop_list.area, "2116" , "Παπάγου");
addOption (document.drop_list.area, "2117" , "Πεντέλη");
addOption (document.drop_list.area, "2118" , "Πεύκη");
addOption (document.drop_list.area, "2119" , "Ροδόπολη");
addOption (document.drop_list.area, "2120" , "Σταμάτα");
addOption (document.drop_list.area, "2121" , "Φιλοθέη");
addOption (document.drop_list.area, "2122" , "Χαλάνδρι");
addOption (document.drop_list.area, "2123" , "Χολαργός");
addOption (document.drop_list.area, "2124" , "Ψυχικό");
}
if (document.drop_list.Region.value == '102') {
addOption (document.drop_list.area, "2200" , "Άγιος Δημήτριος");
addOption (document.drop_list.area, "2201" , "Άλιμος");
addOption (document.drop_list.area, "2202" , "Αργυρούπολη");
addOption (document.drop_list.area, "2203" , "Βάρη");
addOption (document.drop_list.area, "2204" , "Βάρκιζα");
addOption (document.drop_list.area, "2205" , "Βούλα");
addOption (document.drop_list.area, "2206" , "Βουλιαγμένη");
addOption (document.drop_list.area, "2207" , "Βύρωνας");
addOption (document.drop_list.area, "2208" , "Γλυφάδα");
addOption (document.drop_list.area, "2209" , "Δάφνη");
addOption (document.drop_list.area, "2210" , "Ελληνικό");
addOption (document.drop_list.area, "2211" , "Ζωγράφου");
addOption (document.drop_list.area, "2212" , "Ηλιούπολη");
addOption (document.drop_list.area, "2213" , "Καβούρι");
addOption (document.drop_list.area, "2214" , "Καισαριανή");
addOption (document.drop_list.area, "2215" , "Καλλιθέα");
addOption (document.drop_list.area, "2216" , "Ν. Σμύρνη");
addOption (document.drop_list.area, "2217" , "Παλαιό Φάληρο");
addOption (document.drop_list.area, "2218" , "Υμητός");
}
if (document.drop_list.Region.value == '103') {
addOption (document.drop_list.area, "2300" , "Αγία Βαρβάρα");
addOption (document.drop_list.area, "2301" , "Άγιοι Ανάργυροι");
addOption (document.drop_list.area, "2302" , "Αιγάλεω");
addOption (document.drop_list.area, "2303" , "Ανω Λιόσια");
addOption (document.drop_list.area, "2304" , "Αχαρνές (Μενίδι)");
addOption (document.drop_list.area, "2305" , "Γαλάτσι");
addOption (document.drop_list.area, "2306" , "Ζεφύρι");
addOption (document.drop_list.area, "2307" , "Θρακομακεδόνες");
addOption (document.drop_list.area, "2308" , "Ιλιον");
addOption (document.drop_list.area, "2309" , "Καματερό");
addOption (document.drop_list.area, "2310" , "Μεταμόρφωση");
addOption (document.drop_list.area, "2311" , "Νέα Ιωνία");
```

```
addOption (document.drop_list.area,"2312" , "Νέα Φιλαδέλφεια");
addOption (document.drop_list.area,"2313" , "Νέα Χαλκηδόνα");
addOption (document.drop_list.area,"2314" , "Περιστερί");
addOption (document.drop_list.area,"2315" , "Πετρούπολη");
addOption (document.drop_list.area,"2316" , "Χαϊδάρη");
}
if(document.drop_list.Region.value == '104'){
addOption (document.drop_list.area,"2400" , "Ανθούσα");
addOption (document.drop_list.area,"2401" , "Αρτέμιδα (Λούτσα)");
addOption (document.drop_list.area,"2402" , "Γέρακας");
addOption (document.drop_list.area,"2403" , "Γλυκά Νερά");
addOption (document.drop_list.area,"2404" , "Κορωπί");
addOption (document.drop_list.area,"2405" , "Μαραθώνας");
addOption (document.drop_list.area,"2406" , "Μαρκόπουλο");
addOption (document.drop_list.area,"2407" , "Νέα Μάκρη");
addOption (document.drop_list.area,"2408" , "Παιανία");
addOption (document.drop_list.area,"2409" , "Παλλήνη");
addOption (document.drop_list.area,"2410" , "Πικέριμι");
addOption (document.drop_list.area,"2411" , "Ραφήνα");
addOption (document.drop_list.area,"2412" , "Σπάτα");
}
if(document.drop_list.Region.value == '105'){
addOption (document.drop_list.area,"2500" , "Αγία Σοφία");
addOption (document.drop_list.area,"2501" , "Αγίος Βασίλειος");
addOption (document.drop_list.area,"2502" , "Καλλιόπολη");
addOption (document.drop_list.area,"2503" , "Καμίνια");
addOption (document.drop_list.area,"2504" , "Καστέλα");
addOption (document.drop_list.area,"2505" , "Κέντρο Πειραιά");
addOption (document.drop_list.area,"2506" , "Νέο Φάληρο");
addOption (document.drop_list.area,"2507" , "Πασαλιμάνι");
addOption (document.drop_list.area,"2508" , "Πειραιΐκη");
addOption (document.drop_list.area,"2509" , "Τερψιθέα");
addOption (document.drop_list.area,"2510" , "Φρεατιύδα");
addOption (document.drop_list.area,"2511" , "Χατζηκυριάκειο");
}
if(document.drop_list.Region.value == '106'){
addOption (document.drop_list.area,"2600" , "Αγιος Ιωάννης Ρέντη");
addOption (document.drop_list.area,"2601" , "Αγκίστρι");
addOption (document.drop_list.area,"2602" , "Αίγινα");
addOption (document.drop_list.area,"2603" , "Αμπελάκια");
addOption (document.drop_list.area,"2604" , "Αντικύθηρα");
addOption (document.drop_list.area,"2605" , "Δραπετσώνα");
addOption (document.drop_list.area,"2606" , "Κερασίμι");
addOption (document.drop_list.area,"2607" , "Κορυδαλλός");
addOption (document.drop_list.area,"2608" , "Κύθηρα");
addOption (document.drop_list.area,"2609" , "Μέθανα");
```



```
addOption (document.drop_list.area,"2610" , "Μοσχάτο");
addOption (document.drop_list.area,"2611" , "Νίκαια");
addOption (document.drop_list.area,"2612" , "Πέραμα");
addOption (document.drop_list.area,"2613" , "Πόρος");
addOption (document.drop_list.area,"2614" , "Σαλαμίνα");
addOption (document.drop_list.area,"2615" , "Σπέτσες");
addOption (document.drop_list.area,"2616" , "Ταυρος");
addOption (document.drop_list.area,"2617" , "Τροιζήνα");
addOption (document.drop_list.area,"2618" , "Υδρα");
}
if (document.drop_list.Region.value == '107') {
addOption (document.drop_list.area,"2700" , "Αγ. Ανδρέας");
addOption (document.drop_list.area,"2701" , "Αγίος Κωνσταντίνος");
addOption (document.drop_list.area,"2702" , "Αλεποχώρι");
addOption (document.drop_list.area,"2703" , "Ανάβυσσος");
addOption (document.drop_list.area,"2704" , "Ασπρόπυργος");
addOption (document.drop_list.area,"2705" , "Αυλώνα");
addOption (document.drop_list.area,"2706" , "Αφίδνες");
addOption (document.drop_list.area,"2707" , "Βαρνάβας");
addOption (document.drop_list.area,"2708" , "Βαρυμπόμπη");
addOption (document.drop_list.area,"2709" , "Βίλια");
addOption (document.drop_list.area,"2710" , "Βραυρώνα");
addOption (document.drop_list.area,"2711" , "Γραμματικό");
addOption (document.drop_list.area,"2712" , "Ελευσίνα");
addOption (document.drop_list.area,"2713" , "Ερυθρές");
addOption (document.drop_list.area,"2714" , "Καβαλλιανή");
addOption (document.drop_list.area,"2715" , "Κάλαμος");
addOption (document.drop_list.area,"2716" , "Καλύβια");
addOption (document.drop_list.area,"2717" , "Καπανδρίτι");
addOption (document.drop_list.area,"2718" , "Κάτω Σούλι");
addOption (document.drop_list.area,"2719" , "Κερατέα");
addOption (document.drop_list.area,"2720" , "Κινέτα");
addOption (document.drop_list.area,"2721" , "Κουβαράς");
addOption (document.drop_list.area,"2722" , "Λαγονήσι");
addOption (document.drop_list.area,"2723" , "Λαύριο");
addOption (document.drop_list.area,"2724" , "Μαγούλα");
addOption (document.drop_list.area,"2725" , "Μαλακάσα");
addOption (document.drop_list.area,"2726" , "Μαλεσίνα");
addOption (document.drop_list.area,"2727" , "Μάνδρα");
addOption (document.drop_list.area,"2728" , "Μαρκόπουλο Ωρωπού");
addOption (document.drop_list.area,"2729" , "Μάτι");
addOption (document.drop_list.area,"2730" , "Μέγαρα");
addOption (document.drop_list.area,"2731" , "Ν. Βουτζάς");
addOption (document.drop_list.area,"2732" , "Ν. Πέραμος (Μ. Πεύκο)");
addOption (document.drop_list.area,"2733" , "Νέα Παλάτια");
addOption (document.drop_list.area,"2734" , "Οινόη");
```

```
addOption(document.drop_list.area,"2735", "Π. Φωκαία");
addOption(document.drop_list.area,"2736", "Πολυδένδρι");
addOption(document.drop_list.area,"2737", "Πόρτο Γερμενό");
addOption(document.drop_list.area,"2738", "Πόρτο Ράφτη");
addOption(document.drop_list.area,"2739", "Σαρωνίδα");
addOption(document.drop_list.area,"2740", "Σκάλα Ωρωπού");
addOption(document.drop_list.area,"2741", "Σκαραμαγκάς");
addOption(document.drop_list.area,"2742", "Σούνιο");
addOption(document.drop_list.area,"2743", "Συκαμίνο");
addOption(document.drop_list.area,"2744", "Σχινιάς");
addOption(document.drop_list.area,"2745", "Φυλή");
addOption(document.drop_list.area,"2746", "Ψάθα");
addOption(document.drop_list.area,"2747", "Ωρωπός");
}
if(document.drop_list.Region.value == '108'){
addOption(document.drop_list.area,"2800", "40 εκκλησίες");
addOption(document.drop_list.area,"2801", "Αγ. Τριάδα- Λ. Στρατού");
addOption(document.drop_list.area,"2802", "Ανάληψη");
addOption(document.drop_list.area,"2803", "Ανω Πόλη");
addOption(document.drop_list.area,"2804", "Ανω Τούμπα");
addOption(document.drop_list.area,"2805", "Βαρδάρης- Δικαστήρια");
addOption(document.drop_list.area,"2806", "Βούλγαρη");
addOption(document.drop_list.area,"2807", "Διοικητήριο");
addOption(document.drop_list.area,"2808", "Ευαγγελιστρία");
addOption(document.drop_list.area,"2809", "Καμάρα - Ροτόντα");
addOption(document.drop_list.area,"2810", "Κάτω Τούμπα");
addOption(document.drop_list.area,"2811", "Κωνσταντινοπολίτικα");
addOption(document.drop_list.area,"2812", "Λευκός Πύργος");
addOption(document.drop_list.area,"2813", "Λιμάνι - Λαδάδικα");
addOption(document.drop_list.area,"2814", "Μαλακοπή");
addOption(document.drop_list.area,"2815", "Μαρτίου");
addOption(document.drop_list.area,"2816", "Μπότσαρη");
addOption(document.drop_list.area,"2817", "Νέα Ελβετία");
addOption(document.drop_list.area,"2818", "Νέα παραλία");
addOption(document.drop_list.area,"2819", "Ντεπώ");
addOption(document.drop_list.area,"2820", "Παναγιά Φανερωμένη");
addOption(document.drop_list.area,"2821", "Σταθμός ΟΣΕ");
addOption(document.drop_list.area,"2822", "Φάληρο");
addOption(document.drop_list.area,"2823", "Χαριλάου");
addOption(document.drop_list.area,"2899", "Υπόλοιπο κέντρου
Θεσσαλονίκης");
}
if(document.drop_list.Region.value == '109'){
addOption(document.drop_list.area,"2900", "Άγιος Παύλος");
addOption(document.drop_list.area,"2901", "Αμπελόκηποι");
addOption(document.drop_list.area,"2902", "Ελευθερίου - Κορδελιό");
```

```
addOption (document.drop_list.area, "2903" , "Ευκαρπία");
addOption (document.drop_list.area, "2904" , "Εύοσμος");
addOption (document.drop_list.area, "2905" , "Θερμαϊκός");
addOption (document.drop_list.area, "2906" , "Θέρμη");
addOption (document.drop_list.area, "2907" , "Καλαμαριά");
addOption (document.drop_list.area, "2908" , "Μενεμένη");
addOption (document.drop_list.area, "2909" , "Νεάπολη");
addOption (document.drop_list.area, "2910" , "Πανόραμα");
addOption (document.drop_list.area, "2911" , "Πεύκα");
addOption (document.drop_list.area, "2912" , "Πολίχνη");
addOption (document.drop_list.area, "2913" , "Πυλαία");
addOption (document.drop_list.area, "2914" , "Σταυρούπολη");
addOption (document.drop_list.area, "2915" , "Συκιές");
addOption (document.drop_list.area, "2916" , "Τριανδρία");
addOption (document.drop_list.area, "2917" , "Χορτιάτης");
addOption (document.drop_list.area, "2918" , "Ωραιόκαστρο");
}
if (document.drop_list.Region.value == '110') {
addOption (document.drop_list.area, "3000" , "Άγιος Αθανάσιος");
addOption (document.drop_list.area, "3001" , "Άγιος Γεώργιος");
addOption (document.drop_list.area, "3002" , "Αξιός");
addOption (document.drop_list.area, "3003" , "Απολλωνία");
addOption (document.drop_list.area, "3004" , "Αρεθούσα");
addOption (document.drop_list.area, "3005" , "Ασσήρος");
addOption (document.drop_list.area, "3006" , "Βασιλικά");
addOption (document.drop_list.area, "3007" , "Βερτίσκος");
addOption (document.drop_list.area, "3008" , "Εγνατία");
addOption (document.drop_list.area, "3009" , "Επανομή");
addOption (document.drop_list.area, "3010" , "Εχέδωρο");
addOption (document.drop_list.area, "3011" , "Καλλιθέα");
addOption (document.drop_list.area, "3012" , "Καλλινδοίων");
addOption (document.drop_list.area, "3013" , "Κορωνειάς");
addOption (document.drop_list.area, "3014" , "Κουφαλίων");
addOption (document.drop_list.area, "3015" , "Λαγκαδάς");
addOption (document.drop_list.area, "3016" , "Λαχανάς");
addOption (document.drop_list.area, "3017" , "Μαδύτου");
addOption (document.drop_list.area, "3018" , "Μηχανιώνα");
addOption (document.drop_list.area, "3019" , "Μίκρα");
addOption (document.drop_list.area, "3020" , "Μυγδονία");
addOption (document.drop_list.area, "3021" , "Πεντίνα");
addOption (document.drop_list.area, "3022" , "Σόχος");
addOption (document.drop_list.area, "3023" , "Χαλάστρα");
addOption (document.drop_list.area, "3024" , "Χαλκηδόνα");
}
if (document.drop_list.Region.value == '150') {
addOption (document.drop_list.area, "3200" , "Άγγελοκάστρου");
```

```
addOption (document.drop_list.area,"3201" , "Αγρίνιο");
addOption (document.drop_list.area,"3202" , "Αιτωλικού");
addOption (document.drop_list.area,"3203" , "Αλυζίας");
addOption (document.drop_list.area,"3204" , "Αμφιλοχία");
addOption (document.drop_list.area,"3205" , "Ανακτορίου");
addOption (document.drop_list.area,"3206" , "Αντιρρίου");
addOption (document.drop_list.area,"3207" , "Αποδοτίας");
addOption (document.drop_list.area,"3208" , "Αρακύνθου");
addOption (document.drop_list.area,"3209" , "Αστακός");
addOption (document.drop_list.area,"3210" , "Θέρμου");
addOption (document.drop_list.area,"3211" , "Θεστιέων");
addOption (document.drop_list.area,"3212" , "Ινάχου");
addOption (document.drop_list.area,"3213" , "Κεκροπίας");
addOption (document.drop_list.area,"3214" , "Μακρυνείας");
addOption (document.drop_list.area,"3215" , "Μεδεώνος");
addOption (document.drop_list.area,"3216" , "Μενιδίου");
addOption (document.drop_list.area,"3217" , "Μεσολόγγι");
addOption (document.drop_list.area,"3218" , "Ναύπακτος");
addOption (document.drop_list.area,"3219" , "Νεάπολη");
addOption (document.drop_list.area,"3220" , "Οινιάδων");
addOption (document.drop_list.area,"3221" , "Παναιτωλικού");
addOption (document.drop_list.area,"3222" , "Παραβόλας");
addOption (document.drop_list.area,"3223" , "Παρακαμπυλίων");
addOption (document.drop_list.area,"3224" , "Πλάτανος");
addOption (document.drop_list.area,"3225" , "Πυλλήνης");
addOption (document.drop_list.area,"3226" , "Στράτου");
addOption (document.drop_list.area,"3227" , "Φυτειών");
addOption (document.drop_list.area,"3228" , "Χάλκειας");
}
if(document.drop_list.Region.value == '151'){
addOption (document.drop_list.area,"3229" , "Αλέας");
addOption (document.drop_list.area,"3230" , "Αργος");
addOption (document.drop_list.area,"3231" , "Ασίνης");
addOption (document.drop_list.area,"3232" , "Ασκληπιείου");
addOption (document.drop_list.area,"3233" , "Αχλαδόκαμπος");
addOption (document.drop_list.area,"3234" , "Επίδαυρος");
addOption (document.drop_list.area,"3235" , "Ερμιόνη");
addOption (document.drop_list.area,"3236" , "Κουτσοποδίου");
addOption (document.drop_list.area,"3237" , "Κρανίδι");
addOption (document.drop_list.area,"3238" , "Λέρνας");
addOption (document.drop_list.area,"3239" , "Λυρκείας");
addOption (document.drop_list.area,"3240" , "Μιδέας");
addOption (document.drop_list.area,"3241" , "Μυκήνες");
addOption (document.drop_list.area,"3242" , "Ναύπλιο");
addOption (document.drop_list.area,"3243" , "Νέας Κίου");
addOption (document.drop_list.area,"3244" , "Νέας Τίρυνθας");
```

```

}
if(document.drop_list.Region.value == '152'){
addOption(document.drop_list.area,"3245", "Απόλλωνος");
addOption(document.drop_list.area,"3246", "Βαλιετσίου");
addOption(document.drop_list.area,"3247", "Βόρεια Κυνουριά");
addOption(document.drop_list.area,"3248", "Βυτίνα");
addOption(document.drop_list.area,"3249", "Γόρτυνος");
addOption(document.drop_list.area,"3250", "Δημητσάνα");
addOption(document.drop_list.area,"3251", "Ηραίας");
addOption(document.drop_list.area,"3252", "Κλείτορος");
addOption(document.drop_list.area,"3253", "Κοντοβαζαίνης");
addOption(document.drop_list.area,"3254", "Κορυθίου");
addOption(document.drop_list.area,"3255", "Κοσμά");
addOption(document.drop_list.area,"3256", "Λαγκαδίων");
addOption(document.drop_list.area,"3257", "Λεβιδίου");
addOption(document.drop_list.area,"3258", "Λεωνίδιο");
addOption(document.drop_list.area,"3259", "Μαντινείας");
addOption(document.drop_list.area,"3260", "Μεγαλόπολη");
addOption(document.drop_list.area,"3261", "Σκυρίτιδας");
addOption(document.drop_list.area,"3262", "Τεγέας");
addOption(document.drop_list.area,"3263", "Τρικολώνων");
addOption(document.drop_list.area,"3264", "Τρίπολη");
addOption(document.drop_list.area,"3265", "Τροπαίων");
addOption(document.drop_list.area,"3266", "Φαλασσίας");
addOption(document.drop_list.area,"3267", "Φαλάνθου");
}
if(document.drop_list.Region.value == '153'){
addOption(document.drop_list.area,"3268", "Αγνάντων");
addOption(document.drop_list.area,"3269", "Αθαμανίας");
addOption(document.drop_list.area,"3270", "Αμβρακικού");
addOption(document.drop_list.area,"3271", "Αράχθου");
addOption(document.drop_list.area,"3272", "Αρταίων");
addOption(document.drop_list.area,"3273", "Βλαχέρνας");
addOption(document.drop_list.area,"3274", "Γεωργίου Καραϊσκάκη");
addOption(document.drop_list.area,"3275", "Ηρακλείας");
addOption(document.drop_list.area,"3276", "Θεοδωριάνων");
addOption(document.drop_list.area,"3277", "Κομμένου");
addOption(document.drop_list.area,"3278", "Κομποτίου");
addOption(document.drop_list.area,"3279", "Μελισσουργών");
addOption(document.drop_list.area,"3280", "Ξηροβουνίου");
addOption(document.drop_list.area,"3281", "Πέτα");
addOption(document.drop_list.area,"3282", "Τετραφυλίας");
addOption(document.drop_list.area,"3283", "Φιλοθέης");
}
if(document.drop_list.Region.value == '154'){
addOption(document.drop_list.area,"3284", "Αιγείρας");

```

```
addOption(document.drop_list.area,"3285", "Αίγιο");
addOption(document.drop_list.area,"3286", "Ακράτα");
addOption(document.drop_list.area,"3287", "Αροανίας");
addOption(document.drop_list.area,"3288", "Βραχναϊκών");
addOption(document.drop_list.area,"3289", "Διακοπιτό");
addOption(document.drop_list.area,"3290", "Δύμης");
addOption(document.drop_list.area,"3291", "Ερινεού");
addOption(document.drop_list.area,"3292", "Καλάβρυτα");
addOption(document.drop_list.area,"3293", "Καλεντζίου");
addOption(document.drop_list.area,"3294", "Λαρισσού");
addOption(document.drop_list.area,"3295", "Λεοντίου");
addOption(document.drop_list.area,"3296", "Λευκασίου");
addOption(document.drop_list.area,"3297", "Μεσσήτιδος");
addOption(document.drop_list.area,"3298", "Μόβρης");
addOption(document.drop_list.area,"3299", "Παίων");
addOption(document.drop_list.area,"3300", "Παραλίας");
addOption(document.drop_list.area,"3301", "Πάτρα");
addOption(document.drop_list.area,"3302", "Ρίο");
addOption(document.drop_list.area,"3303", "Συμπολιτείας");
addOption(document.drop_list.area,"3304", "Τριταίας");
addOption(document.drop_list.area,"3305", "Φαρρών");
}
if(document.drop_list.Region.value == '155') {
addOption(document.drop_list.area,"3306", "Ωλένιας");
addOption(document.drop_list.area,"3307", "Ακραϊφνίας");
addOption(document.drop_list.area,"3308", "Αλιάρτου");
addOption(document.drop_list.area,"3309", "Αντικύρας");
addOption(document.drop_list.area,"3310", "Αράχωβα");
addOption(document.drop_list.area,"3311", "Βάγια");
addOption(document.drop_list.area,"3312", "Δαύλειας");
addOption(document.drop_list.area,"3313", "Δερβενοχώρια");
addOption(document.drop_list.area,"3314", "Δίστομο");
addOption(document.drop_list.area,"3315", "Θεσπιέων");
addOption(document.drop_list.area,"3316", "Θήβα");
addOption(document.drop_list.area,"3317", "Θίσβης");
addOption(document.drop_list.area,"3318", "Κορώνειας");
addOption(document.drop_list.area,"3319", "Κυριακίου");
addOption(document.drop_list.area,"3320", "Λεμβαδιά");
addOption(document.drop_list.area,"3321", "Οινόφυτα");
addOption(document.drop_list.area,"3322", "Ορχομενός");
addOption(document.drop_list.area,"3323", "Πλαταιές");
addOption(document.drop_list.area,"3324", "Σχηματάρι");
addOption(document.drop_list.area,"3325", "Τανάγρα");
addOption(document.drop_list.area,"3326", "Χαιρωνείας");
}
if(document.drop_list.Region.value == '156') {
```

```
addOption (document.drop_list.area,"3327" , "Αβδέλλας");
addOption (document.drop_list.area,"3328" , "Βεντζίου");
addOption (document.drop_list.area,"3329" , "Γόργιανης");
addOption (document.drop_list.area,"3330" , "Γρεβενά");
addOption (document.drop_list.area,"3331" , "Δεσκάτης");
addOption (document.drop_list.area,"3332" , "Δοτισικού");
addOption (document.drop_list.area,"3333" , "Ηρακλεωτών");
addOption (document.drop_list.area,"3334" , "Θεόδωρου Ζιάκα");
addOption (document.drop_list.area,"3335" , "Κοσμά του Αιτωλού");
addOption (document.drop_list.area,"3336" , "Μεσολουρίου");
addOption (document.drop_list.area,"3337" , "Περιβολίου");
addOption (document.drop_list.area,"3338" , "Σαμαρίνης");
addOption (document.drop_list.area,"3339" , "Σμίξης");
addOption (document.drop_list.area,"3340" , "Φιλιππίων");
addOption (document.drop_list.area,"3341" , "Χασίων");
}
if (document.drop_list.Region.value == '157') {
addOption (document.drop_list.area,"3342" , "Δοξάτου");
addOption (document.drop_list.area,"3343" , "Δράμα");
addOption (document.drop_list.area,"3344" , "Καλαμπακίου");
addOption (document.drop_list.area,"3345" , "Κάτω Νευροκόπι");
addOption (document.drop_list.area,"3346" , "Νικηφόρου");
addOption (document.drop_list.area,"3347" , "Παρανεστίου");
addOption (document.drop_list.area,"3348" , "Προσοτσάνης");
addOption (document.drop_list.area,"3349" , "Σιδηρονέρου");
}
if (document.drop_list.Region.value == '158') {
addOption (document.drop_list.area,"3350" , "Σιταγρών");
addOption (document.drop_list.area,"3351" , "Αγαθονήσι");
addOption (document.drop_list.area,"3352" , "Αρχαγγέλου");
addOption (document.drop_list.area,"3353" , "Αστιπάλαια");
addOption (document.drop_list.area,"3354" , "Αιταβύρου");
addOption (document.drop_list.area,"3355" , "Αφάντου");
addOption (document.drop_list.area,"3356" , "Δικαίου");
addOption (document.drop_list.area,"3357" , "Ηρακλειδών");
addOption (document.drop_list.area,"3358" , "Ιαλυσός");
addOption (document.drop_list.area,"3359" , "Καλλιθέα");
addOption (document.drop_list.area,"3360" , "Κάλυμνος");
addOption (document.drop_list.area,"3361" , "Καμερίου");
addOption (document.drop_list.area,"3362" , "Κάρπαθος");
addOption (document.drop_list.area,"3363" , "Κάσος");
addOption (document.drop_list.area,"3364" , "Κως");
addOption (document.drop_list.area,"3365" , "Λειψών");
addOption (document.drop_list.area,"3366" , "Λέρος");
addOption (document.drop_list.area,"3367" , "Λινδίων");
addOption (document.drop_list.area,"3368" , "Μεγίστης");
```

```
addOption (document.drop_list.area, "3369" , "Νίσιυρος");
addOption (document.drop_list.area, "3370" , "Νότιας Ρόδου");
addOption (document.drop_list.area, "3371" , "Ολύμπου");
addOption (document.drop_list.area, "3372" , "Πάτμος");
addOption (document.drop_list.area, "3373" , "Πεταλούδες");
addOption (document.drop_list.area, "3374" , "Ρόδος");
addOption (document.drop_list.area, "3375" , "Σύμη");
addOption (document.drop_list.area, "3376" , "Τήλος");
addOption (document.drop_list.area, "3377" , "Χάλκης");
}
if (document.drop_list.Region.value == '159') {
addOption (document.drop_list.area, "3378" , "Αλεξανδρούπολη");
addOption (document.drop_list.area, "3379" , "Βύσσας");
addOption (document.drop_list.area, "3380" , "Διδυμότειχο");
addOption (document.drop_list.area, "3381" , "Κυπρίνου");
addOption (document.drop_list.area, "3382" , "Μεταξάδων");
addOption (document.drop_list.area, "3383" , "Ορεσιιάδα");
addOption (document.drop_list.area, "3384" , "Ορφέα");
addOption (document.drop_list.area, "3385" , "Σαμοθράκης");
addOption (document.drop_list.area, "3386" , "Σουφλί");
addOption (document.drop_list.area, "3387" , "Τραϊανούπολης");
addOption (document.drop_list.area, "3388" , "Τριγώνου");
addOption (document.drop_list.area, "3389" , "Τυχερού");
addOption (document.drop_list.area, "3390" , "Φέρρες");
}
if (document.drop_list.Region.value == '160') {
addOption (document.drop_list.area, "3391" , "Αιδηψός");
addOption (document.drop_list.area, "3392" , "Αμαρυνθίων");
addOption (document.drop_list.area, "3393" , "Ανθηδώνος");
addOption (document.drop_list.area, "3394" , "Αρτεμίσιο");
addOption (document.drop_list.area, "3395" , "Αυλίδα");
addOption (document.drop_list.area, "3396" , "Αυλώνα");
addOption (document.drop_list.area, "3397" , "Διρφύων");
addOption (document.drop_list.area, "3398" , "Δυστίων");
addOption (document.drop_list.area, "3399" , "Ελυμνίων");
addOption (document.drop_list.area, "3400" , "Ερέτρια");
addOption (document.drop_list.area, "3401" , "Ιστιαίας");
addOption (document.drop_list.area, "3402" , "Κάρυστος");
addOption (document.drop_list.area, "3403" , "Καφηρέως");
addOption (document.drop_list.area, "3404" , "Κηρέως");
addOption (document.drop_list.area, "3405" , "Κονιστρών");
addOption (document.drop_list.area, "3406" , "Κύμη");
addOption (document.drop_list.area, "3407" , "Ληλαντίων");
addOption (document.drop_list.area, "3408" , "Λιχάδος");
addOption (document.drop_list.area, "3409" , "Μαρμαρίου");
addOption (document.drop_list.area, "3410" , "Μεσσαπίων");
```



```
addOption(document.drop_list.area,"3411", "Νέας Αρτάκης");
addOption(document.drop_list.area,"3412", "Νηλέως");
addOption(document.drop_list.area,"3413", "Σκύρος");
addOption(document.drop_list.area,"3414", "Στύρα");
addOption(document.drop_list.area,"3415", "Ταμιναίων");
addOption(document.drop_list.area,"3416", "Χαλκίδα");
addOption(document.drop_list.area,"3417", "Ωρεών");
}
if(document.drop_list.Region.value == '161'){
addOption(document.drop_list.area,"3418", "Αγραφα");
addOption(document.drop_list.area,"3419", "Απεραντίων");
addOption(document.drop_list.area,"3420", "Ασπροπόταμος");
addOption(document.drop_list.area,"3421", "Βίνιανης");
addOption(document.drop_list.area,"3422", "Δομνίστας");
addOption(document.drop_list.area,"3423", "Καρπενήσι");
addOption(document.drop_list.area,"3424", "Κτημενίων");
addOption(document.drop_list.area,"3425", "Ποταμιάς");
addOption(document.drop_list.area,"3426", "Προυσσού");
addOption(document.drop_list.area,"3427", "Φουρνά");
addOption(document.drop_list.area,"3428", "Φραγκίστας");
}
if(document.drop_list.Region.value == '162'){
addOption(document.drop_list.area,"3429", "Αλυκές");
addOption(document.drop_list.area,"3430", "Αρκαδίων");
addOption(document.drop_list.area,"3431", "Αρτεμισίων");
addOption(document.drop_list.area,"3432", "Ελατίων");
addOption(document.drop_list.area,"3433", "Λαγανά");
addOption(document.drop_list.area,"3434", "Πόλη Ζάκυνθος");
}
if(document.drop_list.Region.value == '163'){
addOption(document.drop_list.area,"3435", "Αλιφείρας");
addOption(document.drop_list.area,"3436", "Αμαλιάδα");
addOption(document.drop_list.area,"3437", "Ανδραβίδας");
addOption(document.drop_list.area,"3438", "Ανδριτσαίνης");
addOption(document.drop_list.area,"3439", "Αρχαίας Ολυμπίας");
addOption(document.drop_list.area,"3440", "Βαρθολομιού");
addOption(document.drop_list.area,"3441", "Βουπρασίας");
addOption(document.drop_list.area,"3442", "Βώλακος");
addOption(document.drop_list.area,"3443", "Γαστιούνης");
addOption(document.drop_list.area,"3444", "Ζαχάρως");
addOption(document.drop_list.area,"3445", "Ιαρδάνου");
addOption(document.drop_list.area,"3446", "Κάστρο Κυλλήνης");
addOption(document.drop_list.area,"3447", "Λαμπείας");
addOption(document.drop_list.area,"3448", "Λασιώνος");
addOption(document.drop_list.area,"3449", "Λεχαιών");
addOption(document.drop_list.area,"3450", "Πηνείας");
```

```
addOption (document.drop_list.area, "3451" , "Πύργος");
addOption (document.drop_list.area, "3452" , "Σκιλλούντος");
addOption (document.drop_list.area, "3453" , "Τραγανού");
addOption (document.drop_list.area, "3454" , "Φιγαλείας");
addOption (document.drop_list.area, "3455" , "Φολόης");
addOption (document.drop_list.area, "3456" , "Ωλένης");
}
if (document.drop_list.Region.value == '164') {
addOption (document.drop_list.area, "3457" , "Αλεξάνδρειας");
addOption (document.drop_list.area, "3458" , "Ανθεμίων");
addOption (document.drop_list.area, "3459" , "Αντιγονιδών");
addOption (document.drop_list.area, "3460" , "Αποστόλου Παύλου");
addOption (document.drop_list.area, "3461" , "Βεργίνα");
addOption (document.drop_list.area, "3462" , "Βέροια");
addOption (document.drop_list.area, "3463" , "Δοβρά");
addOption (document.drop_list.area, "3464" , "Ειρηνούπολης");
addOption (document.drop_list.area, "3465" , "Μακεδονίδος");
addOption (document.drop_list.area, "3466" , "Μελίκης");
addOption (document.drop_list.area, "3467" , "Νάουσα");
addOption (document.drop_list.area, "3468" , "Πλατεός");
}
if (document.drop_list.Region.value == '165') {
addOption (document.drop_list.area, "3469" , "Αγία Βαρβάρα");
addOption (document.drop_list.area, "3470" , "Αρκαλοχωρίου");
addOption (document.drop_list.area, "3471" , "Αρχανών");
addOption (document.drop_list.area, "3472" , "Αστερουσιών");
addOption (document.drop_list.area, "3473" , "Βιάννου");
addOption (document.drop_list.area, "3474" , "Γαζίου");
addOption (document.drop_list.area, "3475" , "Γοργολαΐνη");
addOption (document.drop_list.area, "3476" , "Γόρτυνας");
addOption (document.drop_list.area, "3477" , "Γουβών");
addOption (document.drop_list.area, "3478" , "Επισκοπής");
addOption (document.drop_list.area, "3479" , "Ζαρού");
addOption (document.drop_list.area, "3480" , "Θραψανού");
addOption (document.drop_list.area, "3481" , "Καστελλίου");
addOption (document.drop_list.area, "3482" , "Κόφινα");
addOption (document.drop_list.area, "3483" , "Κρουσώνα");
addOption (document.drop_list.area, "3484" , "Μαλλίων");
addOption (document.drop_list.area, "3485" , "Μοιρών");
addOption (document.drop_list.area, "3486" , "Νέα Αλικαρνασός");
addOption (document.drop_list.area, "3487" , "Νίκου Καζαντζάκη");
addOption (document.drop_list.area, "3488" , "Παλιανής");
addOption (document.drop_list.area, "3489" , "Πόλη Ηρακλείου");
addOption (document.drop_list.area, "3490" , "Ρούβα");
addOption (document.drop_list.area, "3491" , "Τεμένους");
addOption (document.drop_list.area, "3492" , "Τυλίσου");
```

```
addOption(document.drop_list.area,"3493", "Τυμπακίου");
addOption(document.drop_list.area,"3494", "Χερσόνησος");
}
if(document.drop_list.Region.value == '166'){
addOption(document.drop_list.area,"3495", "Αχέροντα");
addOption(document.drop_list.area,"3496", "Ηγουμενίτσα");
addOption(document.drop_list.area,"3497", "Μαργαριτίου");
addOption(document.drop_list.area,"3498", "Παραμυθιάς");
addOption(document.drop_list.area,"3499", "Παραποτάμου");
addOption(document.drop_list.area,"3500", "Πέρδικας");
addOption(document.drop_list.area,"3501", "Σαγιάδας");
addOption(document.drop_list.area,"3502", "Σουλίου");
addOption(document.drop_list.area,"3503", "Συβότων");
addOption(document.drop_list.area,"3504", "Φιλιατών");
}
if(document.drop_list.Region.value == '167'){
addOption(document.drop_list.area,"3505", "Άγιος Δημήτριος");
addOption(document.drop_list.area,"3506", "Αετομηλίτσης");
addOption(document.drop_list.area,"3507", "Ανατολής");
addOption(document.drop_list.area,"3508", "Ανατολικού Ζαγορίου");
addOption(document.drop_list.area,"3509", "Ανω Καλαμά");
addOption(document.drop_list.area,"3510", "Ανω Πωγωνίου");
addOption(document.drop_list.area,"3511", "Βαθυπέδου");
addOption(document.drop_list.area,"3512", "Βοβούσης");
addOption(document.drop_list.area,"3513", "Δελβινακίου");
addOption(document.drop_list.area,"3514", "Δερβιζιάνων");
addOption(document.drop_list.area,"3515", "Διστράτου");
addOption(document.drop_list.area,"3516", "Δωδώνης");
addOption(document.drop_list.area,"3517", "Έγνατίας");
addOption(document.drop_list.area,"3518", "Εκάλης");
addOption(document.drop_list.area,"3519", "Ευρυμενών");
addOption(document.drop_list.area,"3520", "Ζίτσας");
addOption(document.drop_list.area,"3521", "Ιωαννιτών");
addOption(document.drop_list.area,"3522", "Καλαριτών");
addOption(document.drop_list.area,"3523", "Καλπακίου");
addOption(document.drop_list.area,"3524", "Κατσανοχωρίων");
addOption(document.drop_list.area,"3525", "Κεντρικού Ζαγορίου");
addOption(document.drop_list.area,"3526", "Κόνιτσα");
addOption(document.drop_list.area,"3527", "Λάβδανης");
addOption(document.drop_list.area,"3528", "Μαστοροχωρίων");
addOption(document.drop_list.area,"3529", "Ματσουκίου");
addOption(document.drop_list.area,"3530", "Μέτσοβο");
addOption(document.drop_list.area,"3531", "Μηλέας");
addOption(document.drop_list.area,"3532", "Μολοσσών");
addOption(document.drop_list.area,"3533", "Μπιζανίου");
addOption(document.drop_list.area,"3534", "Νέσος Ιωαννίνων");
```

```
addOption (document.drop_list.area,"3535" , "Παμβώτιδος");
addOption (document.drop_list.area,"3536" , "Παπίγκου");
addOption (document.drop_list.area,"3537" , "Πασαράνος");
addOption (document.drop_list.area,"3538" , "Περάματος");
addOption (document.drop_list.area,"3539" , "Πραμάντων");
addOption (document.drop_list.area,"3540" , "Πωγωνιανής");
addOption (document.drop_list.area,"3541" , "Σελλών");
addOption (document.drop_list.area,"3542" , "Σιράκου");
addOption (document.drop_list.area,"3543" , "Τζουμέρκα");
addOption (document.drop_list.area,"3544" , "Τύμφη");
addOption (document.drop_list.area,"3545" , "Φούρκας");
}
if (document.drop_list.Region.value == '168') {
addOption (document.drop_list.area,"3546" , "Ελευθερούπολης");
addOption (document.drop_list.area,"3547" , "Ελευθερών");
addOption (document.drop_list.area,"3548" , "Θάσος");
addOption (document.drop_list.area,"3549" , "Καβάλα");
addOption (document.drop_list.area,"3550" , "Κεραμωτής");
addOption (document.drop_list.area,"3551" , "Ορεινού");
addOption (document.drop_list.area,"3552" , "Ορφανού");
addOption (document.drop_list.area,"3553" , "Παγγαίου");
addOption (document.drop_list.area,"3554" , "Πιερέων");
addOption (document.drop_list.area,"3555" , "Φιλίππων");
addOption (document.drop_list.area,"3556" , "Χρυσούπολης");
}
if (document.drop_list.Region.value == '169') {
addOption (document.drop_list.area,"3557" , "Αθαμανών");
addOption (document.drop_list.area,"3558" , "Αργιθέας");
addOption (document.drop_list.area,"3559" , "Αρνης");
addOption (document.drop_list.area,"3560" , "Αχελώου");
addOption (document.drop_list.area,"3561" , "Ιθώμης");
addOption (document.drop_list.area,"3562" , "Ιτάμου");
addOption (document.drop_list.area,"3563" , "Καλλιφώνου");
addOption (document.drop_list.area,"3564" , "Κάμπου");
addOption (document.drop_list.area,"3565" , "Καρδίτσα");
addOption (document.drop_list.area,"3566" , "Μενελαΐδας");
addOption (document.drop_list.area,"3567" , "Μητρόπολης");
addOption (document.drop_list.area,"3568" , "Μουζακίου");
addOption (document.drop_list.area,"3569" , "Νεβρόπολης Αγράφων");
addOption (document.drop_list.area,"3570" , "Παλαμά");
addOption (document.drop_list.area,"3571" , "Παμίσου");
addOption (document.drop_list.area,"3572" , "Πλαστήρα");
addOption (document.drop_list.area,"3573" , "Ρεντίνης");
addOption (document.drop_list.area,"3574" , "Σελλάνων");
addOption (document.drop_list.area,"3575" , "Σοφάδων");
addOption (document.drop_list.area,"3576" , "Ταμασίου");
```

```
addOption(document.drop_list.area,"3577", "Φύλλου");
addOption(document.drop_list.area,"3578", "Αγία Τριάδα");
}
if(document.drop_list.Region.value == '170'){
addOption(document.drop_list.area,"3579", "Άγιοι Ανάργυροι");
addOption(document.drop_list.area,"3580", "Άκριτών");
addOption(document.drop_list.area,"3581", "Αλιάκμονας");
addOption(document.drop_list.area,"3582", "Αρρένων");
addOption(document.drop_list.area,"3583", "Βίτσι");
addOption(document.drop_list.area,"3584", "Γράμου");
addOption(document.drop_list.area,"3585", "Ιωνος Δραγούμη");
addOption(document.drop_list.area,"3586", "Καστοριάς");
addOption(document.drop_list.area,"3587", "Καστρακίου");
addOption(document.drop_list.area,"3588", "Κλεισούρας");
addOption(document.drop_list.area,"3589", "Κορεστίων");
addOption(document.drop_list.area,"3590", "Μακεδόνων");
addOption(document.drop_list.area,"3591", "Νεστορίου");
addOption(document.drop_list.area,"3592", "Ορεστίδος");
}
if(document.drop_list.Region.value == '171'){
addOption(document.drop_list.area,"3593", "Άγιος Γεώργιος");
addOption(document.drop_list.area,"3594", "Αχιλλείων");
addOption(document.drop_list.area,"3595", "Ερεϊκούσσης");
addOption(document.drop_list.area,"3596", "Εσπερίων");
addOption(document.drop_list.area,"3597", "Θιναλίου");
addOption(document.drop_list.area,"3598", "Κασσωπαίων");
addOption(document.drop_list.area,"3599", "Κέρκυρα");
addOption(document.drop_list.area,"3600", "Κορισσίων");
addOption(document.drop_list.area,"3601", "Λευκιμμάϊων");
addOption(document.drop_list.area,"3602", "Μαθρακίου");
addOption(document.drop_list.area,"3603", "Μελιτειέων");
addOption(document.drop_list.area,"3604", "Οθωνών");
addOption(document.drop_list.area,"3605", "Παλαιοκαστριτών");
addOption(document.drop_list.area,"3606", "Παξιό");
addOption(document.drop_list.area,"3607", "Παρελίων");
addOption(document.drop_list.area,"3608", "Φαιάκων");
}
if(document.drop_list.Region.value == '172'){
addOption(document.drop_list.area,"3609", "Αργοστόλι");
addOption(document.drop_list.area,"3610", "Ελειού - Πρόνων");
addOption(document.drop_list.area,"3611", "Ερίσου");
addOption(document.drop_list.area,"3612", "Ιθάκη");
addOption(document.drop_list.area,"3613", "Λειβαθούς");
addOption(document.drop_list.area,"3614", "Ομαλών");
addOption(document.drop_list.area,"3615", "Παλικής");
addOption(document.drop_list.area,"3616", "Πυλαρέων");
```

```
addOption (document.drop_list.area,"3617" , "Σάμης");
}
if(document.drop_list.Region.value == '173'){
addOption (document.drop_list.area,"3618" , "Αξιούπολης");
addOption (document.drop_list.area,"3619" , "Γαλλικού");
addOption (document.drop_list.area,"3620" , "Γουμένισσας");
addOption (document.drop_list.area,"3621" , "Δοϊράνης");
addOption (document.drop_list.area,"3622" , "Ευρωπού");
addOption (document.drop_list.area,"3623" , "Κιλκίς");
addOption (document.drop_list.area,"3624" , "Κρουσών");
addOption (document.drop_list.area,"3625" , "Λιβαδίων");
addOption (document.drop_list.area,"3626" , "Μουριών");
addOption (document.drop_list.area,"3627" , "Πικρολίμνης");
addOption (document.drop_list.area,"3628" , "Πολύκαστρο");
addOption (document.drop_list.area,"3629" , "Χέρσου");
}
if(document.drop_list.Region.value == '174'){
addOption (document.drop_list.area,"3630" , "Αγίας Παρασκευής");
addOption (document.drop_list.area,"3631" , "Αιανής");
addOption (document.drop_list.area,"3632" , "Ασκίου");
addOption (document.drop_list.area,"3633" , "Βελβεντού");
addOption (document.drop_list.area,"3634" , "Βερμίου");
addOption (document.drop_list.area,"3635" , "Βλάστης");
addOption (document.drop_list.area,"3636" , "Δημητρίου Υψηλάντη");
addOption (document.drop_list.area,"3637" , "Ελιμείας");
addOption (document.drop_list.area,"3638" , "Ελλησπόντου");
addOption (document.drop_list.area,"3639" , "Καμβουνίων");
addOption (document.drop_list.area,"3640" , "Κοζάνη");
addOption (document.drop_list.area,"3641" , "Λιβαδερού");
addOption (document.drop_list.area,"3642" , "Μουρικού");
addOption (document.drop_list.area,"3643" , "Νεάπολης");
addOption (document.drop_list.area,"3644" , "Πενταλόφου");
addOption (document.drop_list.area,"3645" , "Πτολεμαΐδας");
addOption (document.drop_list.area,"3646" , "Σερβίων");
addOption (document.drop_list.area,"3647" , "Σιάτιστας");
addOption (document.drop_list.area,"3648" , "Τσοτυλίου");
}
if(document.drop_list.Region.value == '175'){
addOption (document.drop_list.area,"3649" , "Άγιοι Θεόδωροι");
addOption (document.drop_list.area,"3650" , "Άσσου-Λεχαίου");
addOption (document.drop_list.area,"3651" , "Βέλου");
addOption (document.drop_list.area,"3652" , "Βόχας");
addOption (document.drop_list.area,"3653" , "Ευρωστίνης");
addOption (document.drop_list.area,"3654" , "Κόρινθος");
addOption (document.drop_list.area,"3655" , "Λουτράκι - Περαχώρα");
addOption (document.drop_list.area,"3656" , "Νεμέα");
```

```
addOption (document.drop_list.area,"3657" , "Ευλόκαστρο");
addOption (document.drop_list.area,"3658" , "Σαρωνικού");
addOption (document.drop_list.area,"3659" , "Σικυωνίων");
addOption (document.drop_list.area,"3660" , "Σολυγείας");
addOption (document.drop_list.area,"3661" , "Στυμφαλία");
addOption (document.drop_list.area,"3662" , "Τενέας");
addOption (document.drop_list.area,"3663" , "Φενεού");
}
if (document.drop_list.Region.value == '176') {
addOption (document.drop_list.area,"3664" , "Αμοργός");
addOption (document.drop_list.area,"3665" , "Ανάφη");
addOption (document.drop_list.area,"3666" , "Ανδρος");
addOption (document.drop_list.area,"3667" , "Αντίπαρος");
addOption (document.drop_list.area,"3668" , "Ανω Σύρος");
addOption (document.drop_list.area,"3669" , "Δονούσα");
addOption (document.drop_list.area,"3670" , "Δρυμαλία");
addOption (document.drop_list.area,"3671" , "Εξωμβούργου");
addOption (document.drop_list.area,"3672" , "Ερμούπολη");
addOption (document.drop_list.area,"3673" , "Ηρακλεία");
addOption (document.drop_list.area,"3674" , "Ιος");
addOption (document.drop_list.area,"3675" , "Κέα");
addOption (document.drop_list.area,"3676" , "Κίμωλος");
addOption (document.drop_list.area,"3677" , "Κορθίου");
addOption (document.drop_list.area,"3678" , "Κουφονήσια");
addOption (document.drop_list.area,"3679" , "Κύθνος");
addOption (document.drop_list.area,"3680" , "Μήλος");
addOption (document.drop_list.area,"3681" , "Μύκονος");
addOption (document.drop_list.area,"3682" , "Νάξος");
addOption (document.drop_list.area,"3683" , "Οία");
addOption (document.drop_list.area,"3684" , "Πανόρμου");
addOption (document.drop_list.area,"3685" , "Πάρος");
addOption (document.drop_list.area,"3686" , "Ποσειδωνίας");
addOption (document.drop_list.area,"3687" , "Σαντορίνη");
addOption (document.drop_list.area,"3688" , "Σέριφος");
addOption (document.drop_list.area,"3689" , "Σίκινος");
addOption (document.drop_list.area,"3690" , "Σίφνος");
addOption (document.drop_list.area,"3691" , "Σχοινούσα");
addOption (document.drop_list.area,"3692" , "Τήνος");
addOption (document.drop_list.area,"3693" , "Υδρούσα");
addOption (document.drop_list.area,"3694" , "Φολέγανδρος");
}
if (document.drop_list.Region.value == '177') {
addOption (document.drop_list.area,"3695" , "Ανατολικής Μάνης");
addOption (document.drop_list.area,"3696" , "Ασωπού");
addOption (document.drop_list.area,"3697" , "Βοιών");
addOption (document.drop_list.area,"3698" , "Τερόνθρων");
}
```

```
addOption (document.drop_list.area, "3699" , "Τύθειο");
addOption (document.drop_list.area, "3700" , "Ελαφόνησος");
addOption (document.drop_list.area, "3701" , "Ελους");
addOption (document.drop_list.area, "3702" , "Ζάρακα");
addOption (document.drop_list.area, "3703" , "Θεραπνών");
addOption (document.drop_list.area, "3704" , "Καρυών");
addOption (document.drop_list.area, "3705" , "Κροκεών");
addOption (document.drop_list.area, "3706" , "Μολάοι");
addOption (document.drop_list.area, "3707" , "Μονεμβασιά");
addOption (document.drop_list.area, "3708" , "Μυστράς");
addOption (document.drop_list.area, "3709" , "Νιάτων");
addOption (document.drop_list.area, "3710" , "Οινούντος");
addOption (document.drop_list.area, "3711" , "Οιτύλου");
addOption (document.drop_list.area, "3712" , "Πελλάνας");
addOption (document.drop_list.area, "3713" , "Σκάλας");
addOption (document.drop_list.area, "3714" , "Σμύνους");
addOption (document.drop_list.area, "3715" , "Σπάρτη");
addOption (document.drop_list.area, "3716" , "Φαρίδος");
}
if (document.drop_list.Region.value == '178') {
addOption (document.drop_list.area, "3717" , "Αγιάς");
addOption (document.drop_list.area, "3718" , "Αμπελακίων");
addOption (document.drop_list.area, "3719" , "Αμπελώνος");
addOption (document.drop_list.area, "3720" , "Αντιχασίων");
addOption (document.drop_list.area, "3721" , "Αρμενίου");
addOption (document.drop_list.area, "3722" , "Βερδικούσης");
addOption (document.drop_list.area, "3723" , "Γιάννουλης");
addOption (document.drop_list.area, "3724" , "Γόννων");
addOption (document.drop_list.area, "3725" , "Ελασσόνα");
addOption (document.drop_list.area, "3726" , "Ενιππέα");
addOption (document.drop_list.area, "3727" , "Ευρυμενών");
addOption (document.drop_list.area, "3728" , "Καρυάς");
addOption (document.drop_list.area, "3729" , "Κάτω Ολύμπου");
addOption (document.drop_list.area, "3730" , "Κιλελέρ");
addOption (document.drop_list.area, "3731" , "Κοιλάδας");
addOption (document.drop_list.area, "3732" , "Κραννώνος");
addOption (document.drop_list.area, "3733" , "Λακέρειας");
addOption (document.drop_list.area, "3734" , "Λάρισας");
addOption (document.drop_list.area, "3735" , "Λιβαδίου");
addOption (document.drop_list.area, "3736" , "Μακρυχωρίου");
addOption (document.drop_list.area, "3737" , "Μελιβοίας");
addOption (document.drop_list.area, "3738" , "Ναρθακίου");
addOption (document.drop_list.area, "3739" , "Νέσσωνας");
addOption (document.drop_list.area, "3740" , "Νίκαιας");
addOption (document.drop_list.area, "3741" , "Ολύμπου");
addOption (document.drop_list.area, "3742" , "Πλατυκάμπου");
```



```
addOption (document.drop_list.area,"3743" , "Πολυδάμαντα");
addOption (document.drop_list.area,"3744" , "Ποταμιάς");
addOption (document.drop_list.area,"3745" , "Σαρανταπόρου");
addOption (document.drop_list.area,"3746" , "Τυρνάβου");
addOption (document.drop_list.area,"3747" , "Φαρσάλων");
}
if (document.drop_list.Region.value == '179') {
addOption (document.drop_list.area,"3748" , "Άγιος Νικόλαος");
addOption (document.drop_list.area,"3749" , "Ιεράπετρα");
addOption (document.drop_list.area,"3750" , "Ιτάνου");
addOption (document.drop_list.area,"3751" , "Λεύκης");
addOption (document.drop_list.area,"3752" , "Μακρύς Γιαλός");
addOption (document.drop_list.area,"3753" , "Νεάπολη");
addOption (document.drop_list.area,"3754" , "Οροπέδιο Λασιθίου");
addOption (document.drop_list.area,"3755" , "Σητεία");
}
if (document.drop_list.Region.value == '180') {
addOption (document.drop_list.area,"3756" , "Αγία Παρασκευή");
addOption (document.drop_list.area,"3757" , "Αγιάσου");
addOption (document.drop_list.area,"3758" , "Αγίου Ευστρατίου");
addOption (document.drop_list.area,"3759" , "Αττικής");
addOption (document.drop_list.area,"3760" , "Γέρας");
addOption (document.drop_list.area,"3761" , "Ερεσός");
addOption (document.drop_list.area,"3762" , "Ευεργέτουλα");
addOption (document.drop_list.area,"3763" , "Καλλονή");
addOption (document.drop_list.area,"3764" , "Λουτροπόλεως Θερμής");
addOption (document.drop_list.area,"3765" , "Μανταμάδου");
addOption (document.drop_list.area,"3766" , "Μήθυμνας");
addOption (document.drop_list.area,"3767" , "Μούδρου");
addOption (document.drop_list.area,"3768" , "Μύρινας");
addOption (document.drop_list.area,"3769" , "Μυτιλήνη");
addOption (document.drop_list.area,"3770" , "Νέας Κούταλης");
addOption (document.drop_list.area,"3771" , "Πέτρα");
addOption (document.drop_list.area,"3772" , "Πλωμάρι");
addOption (document.drop_list.area,"3773" , "Πολιχνίτου");
}
if (document.drop_list.Region.value == '181') {
addOption (document.drop_list.area,"3774" , "Απολλωνίων");
addOption (document.drop_list.area,"3775" , "Ελλομένου");
addOption (document.drop_list.area,"3776" , "Καλάμου");
addOption (document.drop_list.area,"3777" , "Καρυάς");
addOption (document.drop_list.area,"3778" , "Καστού");
addOption (document.drop_list.area,"3779" , "Λευκάδα");
addOption (document.drop_list.area,"3780" , "Μεγανησίου");
addOption (document.drop_list.area,"3781" , "Σφακιωτών");
}
}
```

```
if(document.drop_list.Region.value == '182') {
addOption(document.drop_list.area,"3782" ,"Αγριάς");
addOption(document.drop_list.area,"3783" ,"Αισωνίας");
addOption(document.drop_list.area,"3784" ,"Αλμυρού");
addOption(document.drop_list.area,"3785" ,"Αλοννήσου");
addOption(document.drop_list.area,"3786" ,"Ανάβρας");
addOption(document.drop_list.area,"3787" ,"Αργαλαστής");
addOption(document.drop_list.area,"3788" ,"Αρτέμιδας");
addOption(document.drop_list.area,"3789" ,"Αφετών");
addOption(document.drop_list.area,"3790" ,"Βόλος");
addOption(document.drop_list.area,"3791" ,"Ζαγοράς");
addOption(document.drop_list.area,"3792" ,"Ιωλκού");
addOption(document.drop_list.area,"3793" ,"Κάρλας");
addOption(document.drop_list.area,"3794" ,"Κεραμιδίου");
addOption(document.drop_list.area,"3795" ,"Μακρινίτισης");
addOption(document.drop_list.area,"3796" ,"Μηλεών");
addOption(document.drop_list.area,"3797" ,"Μουρεσίου");
addOption(document.drop_list.area,"3798" ,"Νέας Αγχιάλου");
addOption(document.drop_list.area,"3799" ,"Νέας Ιωνίας");
addOption(document.drop_list.area,"3800" ,"Πορταριά");
addOption(document.drop_list.area,"3801" ,"Πτελεού");
addOption(document.drop_list.area,"3802" ,"Σηπιάδος");
addOption(document.drop_list.area,"3803" ,"Σκιάθος");
addOption(document.drop_list.area,"3804" ,"Σκόπελος");
addOption(document.drop_list.area,"3805" ,"Σούρπης");
addOption(document.drop_list.area,"3806" ,"Τρικερίου");
addOption(document.drop_list.area,"3807" ,"Φερών");
}
if(document.drop_list.Region.value == '183') {
addOption(document.drop_list.area,"3808" ,"Αβίας");
addOption(document.drop_list.area,"3809" ,"Αειτού");
addOption(document.drop_list.area,"3810" ,"Αιπείας");
addOption(document.drop_list.area,"3811" ,"Ανδανίας");
addOption(document.drop_list.area,"3812" ,"Ανδρούσης");
addOption(document.drop_list.area,"3813" ,"Αριος");
addOption(document.drop_list.area,"3814" ,"Αριστομένους");
addOption(document.drop_list.area,"3815" ,"Αρφαρών");
addOption(document.drop_list.area,"3816" ,"Αυλώνας");
addOption(document.drop_list.area,"3817" ,"Βουφράδων");
addOption(document.drop_list.area,"3818" ,"Γαργαλιάνων");
addOption(document.drop_list.area,"3819" ,"Δωρίου");
addOption(document.drop_list.area,"3820" ,"Είρας");
addOption(document.drop_list.area,"3821" ,"Θουρίας");
addOption(document.drop_list.area,"3822" ,"Ιθώμης");
addOption(document.drop_list.area,"3823" ,"Καλαμάτα");
addOption(document.drop_list.area,"3824" ,"Κορώνη");
```

```
addOption(document.drop_list.area,"3825", "Κυπαρισσία");
addOption(document.drop_list.area,"3826", "Λεύκτρου");
addOption(document.drop_list.area,"3827", "Μεθώνη");
addOption(document.drop_list.area,"3828", "Μελιγαλά");
addOption(document.drop_list.area,"3829", "Μεσσήνη");
addOption(document.drop_list.area,"3830", "Νέστορος");
addOption(document.drop_list.area,"3831", "Οιχαλίας");
addOption(document.drop_list.area,"3832", "Παπαφλέσσα");
addOption(document.drop_list.area,"3833", "Πεταλιδίου");
addOption(document.drop_list.area,"3834", "Πύλος");
addOption(document.drop_list.area,"3835", "Τρικόρφου");
addOption(document.drop_list.area,"3836", "Τριπύλας");
addOption(document.drop_list.area,"3837", "Φιλιατρών");
addOption(document.drop_list.area,"3838", "Χιλιχωρίων");
}
if(document.drop_list.Region.value == '184'){
addOption(document.drop_list.area,"3839", "Αβδήρων");
addOption(document.drop_list.area,"3840", "Βιστωνίδος");
addOption(document.drop_list.area,"3841", "Θερμών");
addOption(document.drop_list.area,"3842", "Κοτύλης");
addOption(document.drop_list.area,"3843", "Μύκης");
addOption(document.drop_list.area,"3844", "Ξάνθη");
addOption(document.drop_list.area,"3845", "Σατρών");
addOption(document.drop_list.area,"3846", "Σελέρου");
addOption(document.drop_list.area,"3847", "Σταυρούπολης");
addOption(document.drop_list.area,"3848", "Τοπέιρου");
}
if(document.drop_list.Region.value == '185'){
addOption(document.drop_list.area,"3849", "Αριδαίας");
addOption(document.drop_list.area,"3850", "Βεγορίτιδας");
addOption(document.drop_list.area,"3851", "Τριανιτισών");
addOption(document.drop_list.area,"3852", "Έδεσσας");
addOption(document.drop_list.area,"3853", "Εξαπλατάνου");
addOption(document.drop_list.area,"3854", "Κρύας Βρύσης");
addOption(document.drop_list.area,"3855", "Κύρρου");
addOption(document.drop_list.area,"3856", "Μεγάλου Αλεξάνδρου");
addOption(document.drop_list.area,"3857", "Μενηίδος");
addOption(document.drop_list.area,"3858", "Πέλλας");
addOption(document.drop_list.area,"3859", "Σκύδρας");
}
if(document.drop_list.Region.value == '186'){
addOption(document.drop_list.area,"3860", "Αιγινίου");
addOption(document.drop_list.area,"3861", "Ανατολικού Ολύμπου");
addOption(document.drop_list.area,"3862", "Δίου");
addOption(document.drop_list.area,"3863", "Ελαφίνας");
addOption(document.drop_list.area,"3864", "Κατερίνης");
```

```
addOption (document.drop_list.area,"3865" , "Κολινδρού");
addOption (document.drop_list.area,"3866" , "Κορινού");
addOption (document.drop_list.area,"3867" , "Λιτοχώρου");
addOption (document.drop_list.area,"3868" , "Μεθώνης");
addOption (document.drop_list.area,"3869" , "Παραλίας");
addOption (document.drop_list.area,"3870" , "Πέτρας");
addOption (document.drop_list.area,"3871" , "Πιερίων");
addOption (document.drop_list.area,"3872" , "Πύδνας");
}
if(document.drop_list.Region.value == '187') {
addOption (document.drop_list.area,"3873" , "Ανωγείου");
addOption (document.drop_list.area,"3874" , "Ζαλόγγου");
addOption (document.drop_list.area,"3875" , "Θεσπρωτικού");
addOption (document.drop_list.area,"3876" , "Κρανέας");
addOption (document.drop_list.area,"3877" , "Λούρου");
addOption (document.drop_list.area,"3878" , "Πάργα");
addOption (document.drop_list.area,"3879" , "Πρέβεζα");
addOption (document.drop_list.area,"3880" , "Φαναρίου");
addOption (document.drop_list.area,"3881" , "Φιλιππιάδος");
}
if(document.drop_list.Region.value == '188') {
addOption (document.drop_list.area,"3882" , "Ανωγείων");
addOption (document.drop_list.area,"3883" , "Αρκαδίου");
addOption (document.drop_list.area,"3884" , "Γεροποτάμου");
addOption (document.drop_list.area,"3885" , "Κουλούκωνα");
addOption (document.drop_list.area,"3886" , "Κουρητών");
addOption (document.drop_list.area,"3887" , "Λάμπης");
addOption (document.drop_list.area,"3888" , "Λαππαίων");
addOption (document.drop_list.area,"3889" , "Νικηφόρου Φωκά");
addOption (document.drop_list.area,"3890" , "Ρέθυμνο");
addOption (document.drop_list.area,"3891" , "Συβρίτου");
addOption (document.drop_list.area,"3892" , "Φοίνικα");
}
if(document.drop_list.Region.value == '189') {
addOption (document.drop_list.area,"3893" , "Αιγείρου");
addOption (document.drop_list.area,"3894" , "Αμαξάδων");
addOption (document.drop_list.area,"3895" , "Αρριανών");
addOption (document.drop_list.area,"3896" , "Ιάσμου");
addOption (document.drop_list.area,"3897" , "Κέχρου");
addOption (document.drop_list.area,"3898" , "Κομοτηνής");
addOption (document.drop_list.area,"3899" , "Μαρωνείας");
addOption (document.drop_list.area,"3900" , "Νέου Σιδηροχωρίου");
addOption (document.drop_list.area,"3901" , "Οργάνης");
addOption (document.drop_list.area,"3902" , "Σαπών");
addOption (document.drop_list.area,"3903" , "Σώστου");
addOption (document.drop_list.area,"3904" , "Φιλλύρας");
}
```

```
}  
if(document.drop_list.Region.value == '190') {  
addOption(document.drop_list.area,"3905", "Αγίου Κηρύκου");  
addOption(document.drop_list.area,"3906", "Βαθέος");  
addOption(document.drop_list.area,"3907", "Εύδηλος - Ικαρία");  
addOption(document.drop_list.area,"3908", "Καρλόβασι");  
addOption(document.drop_list.area,"3909", "Μαραθόκαμπος");  
addOption(document.drop_list.area,"3910", "Πυθαγόρειο");  
addOption(document.drop_list.area,"3911", "Ραχών");  
addOption(document.drop_list.area,"3912", "Φούρνων Κορσεών");  
}  
if(document.drop_list.Region.value == '191') {  
addOption(document.drop_list.area,"3913", "Αγκίστρου");  
addOption(document.drop_list.area,"3914", "Αλιστράτης");  
addOption(document.drop_list.area,"3915", "Αμφίπολης");  
addOption(document.drop_list.area,"3916", "Ανω Βροντούς");  
addOption(document.drop_list.area,"3917", "Αχινού");  
addOption(document.drop_list.area,"3918", "Αχλαδοχωρίου");  
addOption(document.drop_list.area,"3919", "Βισαλτίας");  
addOption(document.drop_list.area,"3920", "Εμμανουήλ Παππά");  
addOption(document.drop_list.area,"3921", "Ηρακλείας");  
addOption(document.drop_list.area,"3922", "Κερκίνης");  
addOption(document.drop_list.area,"3923", "Κορμίστας");  
addOption(document.drop_list.area,"3924", "Λευκώνα");  
addOption(document.drop_list.area,"3925", "Μητροσίου");  
addOption(document.drop_list.area,"3926", "Νέας Ζίχνης");  
addOption(document.drop_list.area,"3927", "Νιγρίτης");  
addOption(document.drop_list.area,"3928", "Ορεινής");  
addOption(document.drop_list.area,"3929", "Πετριτσίου");  
addOption(document.drop_list.area,"3930", "Προμαχώνος");  
addOption(document.drop_list.area,"3931", "Πρώτης");  
addOption(document.drop_list.area,"3932", "Ροδολίβους");  
addOption(document.drop_list.area,"3933", "Σέρρες");  
addOption(document.drop_list.area,"3934", "Σιδηροκάστρου");  
addOption(document.drop_list.area,"3935", "Σκοτούσσης");  
addOption(document.drop_list.area,"3936", "Σκουτάρεως");  
addOption(document.drop_list.area,"3937", "Στυμώνα");  
addOption(document.drop_list.area,"3938", "Στυμωνικού");  
addOption(document.drop_list.area,"3939", "Τραγίλου");  
}  
if(document.drop_list.Region.value == '192') {  
addOption(document.drop_list.area,"3940", "Αιθήκων");  
addOption(document.drop_list.area,"3941", "Ασπροποτάμου");  
addOption(document.drop_list.area,"3942", "Βασιλικής");  
addOption(document.drop_list.area,"3943", "Γόμφων");  
addOption(document.drop_list.area,"3944", "Εστιαιώτιδας");  
}
```

```

addOption (document.drop_list.area,"3945" , "Καλαμπάκα");
addOption (document.drop_list.area,"3946" , "Καλλιιδένδρου");
addOption (document.drop_list.area,"3947" , "Καστανιάς");
addOption (document.drop_list.area,"3948" , "Κλεινοβού");
addOption (document.drop_list.area,"3949" , "Κόζιακα");
addOption (document.drop_list.area,"3950" , "Μαλακασίου");
addOption (document.drop_list.area,"3951" , "Μεγάλων Καλυβίων");
addOption (document.drop_list.area,"3952" , "Μυροφύλλου");
addOption (document.drop_list.area,"3953" , "Νεράιδας");
addOption (document.drop_list.area,"3954" , "Οιχαλίας");
addOption (document.drop_list.area,"3955" , "Παληοκάστρου");
addOption (document.drop_list.area,"3956" , "Παραληθαίων");
addOption (document.drop_list.area,"3957" , "Πελιναίων");
addOption (document.drop_list.area,"3958" , "Πιαλείων");
addOption (document.drop_list.area,"3959" , "Πύλης");
addOption (document.drop_list.area,"3960" , "Πυνδαίων");
addOption (document.drop_list.area,"3961" , "Τρικκαίων");
addOption (document.drop_list.area,"3962" , "Τυμφαίων");
addOption (document.drop_list.area,"3963" , "Φαλωρείας");
addOption (document.drop_list.area,"3964" , "Φαρκαδόνας");
addOption (document.drop_list.area,"3965" , "Χασίων");
}
if(document.drop_list.Region.value == '193'){
addOption (document.drop_list.area,"3966" , "Αγίου Γεωργίου Τυμφ");
addOption (document.drop_list.area,"3967" , "Αγίου Κωνσταντίνου");
addOption (document.drop_list.area,"3968" , "Αμφίκλειας");
addOption (document.drop_list.area,"3969" , "Αταλάντης");
addOption (document.drop_list.area,"3970" , "Γοργοποτάμου");
addOption (document.drop_list.area,"3971" , "Δαφνουσίων");
addOption (document.drop_list.area,"3972" , "Δομοκός");
addOption (document.drop_list.area,"3973" , "Ελάτειας");
addOption (document.drop_list.area,"3974" , "Εχιναίων");
addOption (document.drop_list.area,"3975" , "Θεσσαλιώτιδος");
addOption (document.drop_list.area,"3976" , "Καμμένα Βούρλα");
addOption (document.drop_list.area,"3977" , "Λαμία");
addOption (document.drop_list.area,"3978" , "Λειανοκλάδι");
addOption (document.drop_list.area,"3979" , "Μακρακώμης");
addOption (document.drop_list.area,"3980" , "Μαλεσίνης");
addOption (document.drop_list.area,"3981" , "Μώλου");
addOption (document.drop_list.area,"3982" , "Ξυνιάδος");
addOption (document.drop_list.area,"3983" , "Οπουντίων");
addOption (document.drop_list.area,"3984" , "Παύλιανης");
addOption (document.drop_list.area,"3985" , "Πελασγίας");
addOption (document.drop_list.area,"3986" , "Σπερχειάδος");
addOption (document.drop_list.area,"3987" , "Στυλίδος");
addOption (document.drop_list.area,"3988" , "Τιθορέας");
}

```

```
addOption(document.drop_list.area,"3989", "Τυμφρηστός");
addOption(document.drop_list.area,"3990", "Υπάτης");
}
if(document.drop_list.Region.value == '194'){
addOption(document.drop_list.area,"3991", "Αειτού");
addOption(document.drop_list.area,"3992", "Αμυνταίου");
addOption(document.drop_list.area,"3993", "Βαρικού");
addOption(document.drop_list.area,"3994", "Κάτω Κλεινών");
addOption(document.drop_list.area,"3995", "Κρυσταλλοπηγής");
addOption(document.drop_list.area,"3996", "Λεχόβου");
addOption(document.drop_list.area,"3997", "Μελίτης");
addOption(document.drop_list.area,"3998", "Νυμφαίου");
addOption(document.drop_list.area,"3999", "Περάσματος");
addOption(document.drop_list.area,"4000", "Πρέσπες");
addOption(document.drop_list.area,"4001", "Φιλώτα");
addOption(document.drop_list.area,"4002", "Φλώρινα");
}
if(document.drop_list.Region.value == '195'){
addOption(document.drop_list.area,"4003", "Αμφισσα");
addOption(document.drop_list.area,"4004", "Βαρδούσια");
addOption(document.drop_list.area,"4005", "Γαλαξίδι");
addOption(document.drop_list.area,"4006", "Γραβιάς");
addOption(document.drop_list.area,"4007", "Δελφοί");
addOption(document.drop_list.area,"4008", "Δεσφίνης");
addOption(document.drop_list.area,"4009", "Ευπαλίου");
addOption(document.drop_list.area,"4010", "Ιτεά");
addOption(document.drop_list.area,"4011", "Καλλιέων");
addOption(document.drop_list.area,"4012", "Λιδορικού");
addOption(document.drop_list.area,"4013", "Παρνασσός");
addOption(document.drop_list.area,"4014", "Τολοφώνος");
}
if(document.drop_list.Region.value == '196'){
addOption(document.drop_list.area,"4015", "Ανθεμούντα");
addOption(document.drop_list.area,"4016", "Αρναίας");
addOption(document.drop_list.area,"4017", "Ζερβοχωρίων");
addOption(document.drop_list.area,"4018", "Καλλικράτειας");
addOption(document.drop_list.area,"4019", "Κασσάνδρα");
addOption(document.drop_list.area,"4020", "Μουδανιά");
addOption(document.drop_list.area,"4021", "Ορμυλίας");
addOption(document.drop_list.area,"4022", "Παλλήνη");
addOption(document.drop_list.area,"4023", "Παναγίας");
addOption(document.drop_list.area,"4024", "Πολύγυρος");
addOption(document.drop_list.area,"4025", "Σιθωνία");
addOption(document.drop_list.area,"4026", "Σταγύρων-Ακάνθου");
addOption(document.drop_list.area,"4027", "Τορώνης");
addOption(document.drop_list.area,"4028", "Τρίγλις");
```

```
}  
if(document.drop_list.Region.value == '197') {  
addOption(document.drop_list.area,"4029", "Ακρωτήρι");  
addOption(document.drop_list.area,"4030", "Ανατολικού Σελίνου");  
addOption(document.drop_list.area,"4031", "Αρμένων");  
addOption(document.drop_list.area,"4032", "Ασή Γωνιάς");  
addOption(document.drop_list.area,"4033", "Βάμου");  
addOption(document.drop_list.area,"4034", "Βουκολιών");  
addOption(document.drop_list.area,"4035", "Γαύδος");  
addOption(document.drop_list.area,"4036", "Γεωργιούπολη");  
addOption(document.drop_list.area,"4037", "Ελευθερίου Βενιζέλου");  
addOption(document.drop_list.area,"4038", "Θερίσου");  
addOption(document.drop_list.area,"4039", "Ινναχωρίου");  
addOption(document.drop_list.area,"4040", "Καντάνου");  
addOption(document.drop_list.area,"4041", "Κεραμιών");  
addOption(document.drop_list.area,"4042", "Κισσάμου");  
addOption(document.drop_list.area,"4043", "Κολυμβαρίου");  
addOption(document.drop_list.area,"4044", "Κρυονερίδας");  
addOption(document.drop_list.area,"4045", "Μουσούρων");  
addOption(document.drop_list.area,"4046", "Μυθήμνης");  
addOption(document.drop_list.area,"4047", "Νέα Κυδωνία");  
addOption(document.drop_list.area,"4048", "Πελεκάνου");  
addOption(document.drop_list.area,"4049", "Πλατανιάς");  
addOption(document.drop_list.area,"4050", "Πόλη Χανίων");  
addOption(document.drop_list.area,"4051", "Σούδα");  
addOption(document.drop_list.area,"4052", "Σφακιά");  
addOption(document.drop_list.area,"4053", "Φρέ");  
}  
if(document.drop_list.Region.value == '198') {  
addOption(document.drop_list.area,"4054", "Αγίου Μηνά");  
addOption(document.drop_list.area,"4055", "Αμανής");  
addOption(document.drop_list.area,"4056", "Ιωνίας");  
addOption(document.drop_list.area,"4057", "Καμποχώρων");  
addOption(document.drop_list.area,"4058", "Καρδαμύλων");  
addOption(document.drop_list.area,"4059", "Μαστιχοχωρίων");  
addOption(document.drop_list.area,"4060", "Οινούσες");  
addOption(document.drop_list.area,"4061", "Ομηρούπολης");  
addOption(document.drop_list.area,"4062", "Πόλη Χίου");  
addOption(document.drop_list.area,"4063", "Ψαρά");  
}  
if(document.drop_list.Region.value == '301') {  
addOption(document.drop_list.area,"4500", "Αγία Νάπα");  
addOption(document.drop_list.area,"4501", "Λάρνακα");  
addOption(document.drop_list.area,"4800", "Λεμεσός");  
addOption(document.drop_list.area,"4801", "Λευκωσία");  
addOption(document.drop_list.area,"4802", "Παραλίμνη");
```



```
addOption(document.drop_list.area,"4803", "Πάφος");
addOption(document.drop_list.area,"4804", "Πρωταράς");
addOption(document.drop_list.area,"4805", "Υπόλοιπη Κύπρος");
}
if(document.drop_list.Region.value == '401'){
addOption(document.drop_list.area,"5000", "Μπουένος Άιρες");
addOption(document.drop_list.area,"5001", "Υπόλοιπη Αργεντινή");
}
if(document.drop_list.Region.value == '402'){
addOption(document.drop_list.area,"5002", "Σίδνεϋ");
addOption(document.drop_list.area,"5003", "Υπόλοιπη Αυστραλία");
}
if(document.drop_list.Region.value == '403'){
addOption(document.drop_list.area,"5004", "Βιέννη");
addOption(document.drop_list.area,"5005", "Υπόλοιπη Αυστρία");
}
if(document.drop_list.Region.value == '404'){
addOption(document.drop_list.area,"5006", "Ανβέρσα");
addOption(document.drop_list.area,"5007", "Βρυξέλλες");
addOption(document.drop_list.area,"5008", "Σαρλερουά");
addOption(document.drop_list.area,"5009", "Λιέγη");
addOption(document.drop_list.area,"5010", "Υπόλοιπο Βέλγιο");
}
if(document.drop_list.Region.value == '405'){
addOption(document.drop_list.area,"5011", "Βόρεια Βουλγαρία");
addOption(document.drop_list.area,"5012", "Παράλια Μαύρης Θάλασσας");
addOption(document.drop_list.area,"5013", "Σόφια");
addOption(document.drop_list.area,"5014", "Υπόλοιπη Βουλγαρία");
}
if(document.drop_list.Region.value == '406'){
addOption(document.drop_list.area,"5015", "Άλπεις");
addOption(document.drop_list.area,"5016", "Κορσική");
addOption(document.drop_list.area,"5017", "Νορμανδία");
addOption(document.drop_list.area,"5018", "Παρίσι");
addOption(document.drop_list.area,"5019", "Πυρηναία");
addOption(document.drop_list.area,"5020", "Κυανή Ακτή");
addOption(document.drop_list.area,"5021", "Υπόλοιπη Γαλλία");
}
if(document.drop_list.Region.value == '407'){
addOption(document.drop_list.area,"5022", "Βερολίνο");
addOption(document.drop_list.area,"5023", "Μόναχο");
addOption(document.drop_list.area,"5024", "Φρανκφούρτη");
addOption(document.drop_list.area,"5025", "Υπόλοιπη Γερμανία");
}
if(document.drop_list.Region.value == '408'){
addOption(document.drop_list.area,"5026", "Ζυρίχη");
```

```
addOption(document.drop_list.area,"5027", "Λοζάνη");
addOption(document.drop_list.area,"5028", "Γενέβη");
addOption(document.drop_list.area,"5029", "Υπόλοιπη Ελβετία");
}
if(document.drop_list.Region.value == '409'){
addOption(document.drop_list.area,"5030", "Ντουμπάι");
addOption(document.drop_list.area,"5031", "Υπόλοιπα Η.Α.Ε.");
}
if(document.drop_list.Region.value == '410'){
addOption(document.drop_list.area,"5032", "Βοστώνη");
addOption(document.drop_list.area,"5033", "Καλιφόρνια");
addOption(document.drop_list.area,"5034", "Σικάγο");
addOption(document.drop_list.area,"5035", "Φλόριδα");
addOption(document.drop_list.area,"5036", "Νέα Υόρκη");
addOption(document.drop_list.area,"5037", "Υπόλοιπες Η.Π.Α.");
}
if(document.drop_list.Region.value == '411'){
addOption(document.drop_list.area,"5038", "Ανδαλουσία");
addOption(document.drop_list.area,"5039", "Βαρκελώνη");
addOption(document.drop_list.area,"5040", "Μαδρίτη");
addOption(document.drop_list.area,"5041", "Υπόλοιπη Ισπανία");
}
if(document.drop_list.Region.value == '412'){
addOption(document.drop_list.area,"5042", "Βενετία");
addOption(document.drop_list.area,"5043", "Μιλάνο");
addOption(document.drop_list.area,"5044", "Νάπολη");
addOption(document.drop_list.area,"5045", "Ρώμη");
addOption(document.drop_list.area,"5046", "Σικελία");
addOption(document.drop_list.area,"5047", "Τοσκάνη");
addOption(document.drop_list.area,"5048", "Φλωρεντία");
addOption(document.drop_list.area,"5049", "Υπόλοιπη Ιταλία");
}
if(document.drop_list.Region.value == '413'){
addOption(document.drop_list.area,"5050", "Βανκούβερ");
addOption(document.drop_list.area,"5051", "Μόντρεαλ");
addOption(document.drop_list.area,"5052", "Τορόντο");
addOption(document.drop_list.area,"5053", "Υπόλοιπος Καναδάς");
}
if(document.drop_list.Region.value == '414'){
addOption(document.drop_list.area,"5054", "Ζάγκρεμπ");
addOption(document.drop_list.area,"5055", "Υπόλοιπη Κροατία");
}
if(document.drop_list.Region.value == '415'){
addOption(document.drop_list.area,"5056", "Acton");
addOption(document.drop_list.area,"5057", "Balham");
addOption(document.drop_list.area,"5058", "Barnes");
```

```
addOption (document.drop_list.area, "5059" , "Battersea");
addOption (document.drop_list.area, "5060" , "Bayswater");
addOption (document.drop_list.area, "5061" , "Belgravia");
addOption (document.drop_list.area, "5062" , "Bloomsbury");
addOption (document.drop_list.area, "5063" , "Camden");
addOption (document.drop_list.area, "5064" , "Chelsea");
addOption (document.drop_list.area, "5065" , "Chiswick");
addOption (document.drop_list.area, "5066" , "Clapham");
addOption (document.drop_list.area, "5067" , "Docklands");
addOption (document.drop_list.area, "5068" , "Ealing");
addOption (document.drop_list.area, "5069" , "Fulham");
addOption (document.drop_list.area, "5070" , "Greenwich");
addOption (document.drop_list.area, "5071" , "Hackney");
addOption (document.drop_list.area, "5072" , "Hammersmith");
addOption (document.drop_list.area, "5073" , "Hampstead");
addOption (document.drop_list.area, "5074" , "Islington");
addOption (document.drop_list.area, "5075" , "Kennington");
addOption (document.drop_list.area, "5076" , "Kensington");
addOption (document.drop_list.area, "5077" , "Knightsbridge");
addOption (document.drop_list.area, "5078" , "Maida Vale");
addOption (document.drop_list.area, "5079" , "Marylebone");
addOption (document.drop_list.area, "5080" , "Mayfair");
addOption (document.drop_list.area, "5081" , "Muswell Hill");
addOption (document.drop_list.area, "5082" , "Notting Hill");
addOption (document.drop_list.area, "5083" , "Putney");
addOption (document.drop_list.area, "5084" , "Queens Park");
addOption (document.drop_list.area, "5085" , "Richmond");
addOption (document.drop_list.area, "5086" , "Shoreditch");
addOption (document.drop_list.area, "5087" , "Southwark");
addOption (document.drop_list.area, "5088" , "St. Johns Wood");
addOption (document.drop_list.area, "5089" , "Streatham");
addOption (document.drop_list.area, "5090" , "Tooting");
addOption (document.drop_list.area, "5091" , "Tower Hamlets");
addOption (document.drop_list.area, "5092" , "Twickenham");
addOption (document.drop_list.area, "5093" , "Wandsworth");
addOption (document.drop_list.area, "5094" , "Wimblendon");
addOption (document.drop_list.area, "5095" , "Υπόλοιπο Λονδίνο");
}
if (document.drop_list.Region.value == '416') {
addOption (document.drop_list.area, "5096" , "Birmingham");
addOption (document.drop_list.area, "5097" , "Brighton");
addOption (document.drop_list.area, "5098" , "Cardiff");
addOption (document.drop_list.area, "5099" , "Edinburgh");
addOption (document.drop_list.area, "5100" , "Glasgow");
addOption (document.drop_list.area, "5101" , "Lancaster");
addOption (document.drop_list.area, "5102" , "Leeds");
```

```
addOption (document.drop_list.area,"5103" , "Leicester");
addOption (document.drop_list.area,"5104" , "Liverpool");
addOption (document.drop_list.area,"5105" , "Manchester");
addOption (document.drop_list.area,"5106" , "Nottingham");
addOption (document.drop_list.area,"5107" , "Reading");
addOption (document.drop_list.area,"5108" , "Stirling");
addOption (document.drop_list.area,"5109" , "Υπόλοιπη Μ.Βρετανία");
}
if(document.drop_list.Region.value == '417'){
addOption (document.drop_list.area,"5110" , "Ποντγκόριτσα");
addOption (document.drop_list.area,"5111" , "Υπόλοιπο Μαυροβούνιο");
}
if(document.drop_list.Region.value == '418'){
addOption (document.drop_list.area,"5112" , "Αμστερνταμ");
addOption (document.drop_list.area,"5113" , "Αιντχόβεν");
addOption (document.drop_list.area,"5114" , "Ρότερνταμ");
addOption (document.drop_list.area,"5115" , "Χάγη");
addOption (document.drop_list.area,"5116" , "Ουτρέχτη");
addOption (document.drop_list.area,"5117" , "Υπόλοιπη Ολλανδία");
}
if(document.drop_list.Region.value == '419'){
addOption (document.drop_list.area,"5118" , "Βαρσοβία");
addOption (document.drop_list.area,"5119" , "Υπόλοιπη Πολωνία");
}
if(document.drop_list.Region.value == '420'){
addOption (document.drop_list.area,"5120" , "Ακτές Ατλαντικού");
addOption (document.drop_list.area,"5121" , "Υπόλοιπη Πορτογαλία");
}
if(document.drop_list.Region.value == '421'){
addOption (document.drop_list.area,"5122" , "Βουκουρέστι");
addOption (document.drop_list.area,"5123" , "Κωσταντζα");
addOption (document.drop_list.area,"5124" , "Υπόλοιπη Ρουμανία");
}
if(document.drop_list.Region.value == '422'){
addOption (document.drop_list.area,"5125" , "Μόσχα");
addOption (document.drop_list.area,"5126" , "Πετρούπολη");
addOption (document.drop_list.area,"5127" , "Υπόλοιπη Ρωσία");
}
if(document.drop_list.Region.value == '423'){
addOption (document.drop_list.area,"5128" , "Βελιγράδι");
addOption (document.drop_list.area,"5129" , "Υπόλοιπη Σερβία");
}
if(document.drop_list.Region.value == '424'){
addOption (document.drop_list.area,"5130" , "Λιουμπλιάννα");
addOption (document.drop_list.area,"5131" , "Υπόλοιπη Σλοβενία");
}
}
```

```

if(document.drop_list.Region.value == '425'){
addOption(document.drop_list.area,"5132" , "Στοκχόλμη");
addOption(document.drop_list.area,"5133" , "Υπόλοιπη Σουηδία");
}
if(document.drop_list.Region.value == '426'){
addOption(document.drop_list.area,"5134" , "Κωνσταντινούπολη");
addOption(document.drop_list.area,"5135" , "Παράλια Αιγαίου");
addOption(document.drop_list.area,"5136" , "Υπόλοιπη Τουρκία");
}
if(document.drop_list.Region.value == '427'){
addOption(document.drop_list.area,"5137" , "Πράγα");
addOption(document.drop_list.area,"5138" , "Υπόλοιπη Τσεχία");
}
if(document.drop_list.Region.value == '500'){
addOption(document.drop_list.area,"5139" , "Ασία");
addOption(document.drop_list.area,"5140" , "Βόρεια και Νότια
Αμερική");
addOption(document.drop_list.area,"5141" , "Ευρώπη");
addOption(document.drop_list.area,"5142" , "Υπόλοιπος κόσμος");
}

}

////////////////////

function removeAllOptions(selectbox)
{
var i;
for(i=selectbox.options.length-1;i>=0;i--)
{
//selectbox.options.remove(i);
selectbox.remove(i);
}
}

function addOption(selectbox, value, text )
{
var optn = document.createElement("OPTION");
optn.text = text;
optn.value = value;

selectbox.options.add(optn);
}

```

Style.css

```
body {
```

```
Εγκατάσταση σε έξυπνες συσκευές ανεξαρτήτως πλατφόρμας εφαρμογή καταχώρησης
δεδομένων μέσω διαδικτυακής υπηρεσίας
```

```
        background-color: #F88017;
        font-family: Calibri;
    }
#listingnumber {
    font-family: Calibri;
    font-weight: bold;
    font-size: 150%;
}
#send_image {
    background-color: transparent;
    background-image: url(buttonBackground.gif);
    background-repeat: no-repeat;
    border: none;
    width: 132px;
    height: 28px;
    margin: 5px auto;
    padding: 5px 0 0 0;
    text-align: center;
    font-family: Helvetica, Calibri, Arial, sans-serif;
    font-size: 100%;
    font-weight: bold;
    text-decoration: none;
}

.box{
width:200px;
}
.smallbox{
width:90px;
}
.app_button {
    background-color: transparent;
    background-image: url(buttonBackground.gif);
    background-repeat: no-repeat;
    border: none;
    width: 132px;
    height: 28px;
    margin: 5px auto;
    padding: 5px 0 0 0;
    text-align: center;
    font-family: Helvetica, Calibri, Arial, sans-serif;
    font-size: 100%;
    font-weight: bold;
    text-decoration: none;
}
.reset_button {
```

```
background-color: transparent;
background-image: url(buttonBackground.gif);
background-repeat: no-repeat;
background-position: 0 -72px;
border: none;
width: 132px;
height: 28px;
margin: 5px auto;
padding: 5px 0 0 0;
text-align: center;
font-family: Helvetica, Calibri, Arial, sans-serif;
font-size: 100%;
font-weight: bold;
text-decoration: none;
}
.back_button {
background-color: transparent;
background-image: url(buttonBackground.gif);
background-repeat: no-repeat;
background-position: 0 -36px;
border: none;
width: 132px;
height: 28px;
margin: 5px auto;
padding: 5px 0 0 0;
text-align: center;
font-family: Helvetica, Calibri, Arial, sans-serif;
font-size: 100%;
font-weight: bold;
text-decoration: none;
}
```