



Πανεπιστήμιο Δυτικής Μακεδονίας
Τμήμα Μηχανικών Πληροφορικής & Τηλεπικοινωνιών

Διπλωματική Εργασία

Σχεδιασμός και Ανάπτυξη Εφαρμογής
Android με Υποστηρικτική Ιστοσελίδα για
Δημιουργία Ταξιδιωτικού Οδηγού Δυτικής
Μακεδονίας

Ρέγκου Ελένη

Επιβλέπων Καθηγητής : Δρ. Μηνάς Δασυγένης
Εργαστήριο Ψηφιακών Συστημάτων και
Αρχιτεκτονικής Υπολογιστών

Κοζάνη, Ιούνιος 2018



Στους γονείς μου
και την αδερφή μου



Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον Δρ. Μηνά Δασυγένη για την πολύτιμη συμβολή και καθοδήγηση του καθ' όλη τη διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένεια μου για την αμέριστη ηθική και υλική υποστήριξη που μου παρείχε κατά τη διάρκεια των σπουδών μου.

Τέλος, θα ήθελα να ευχαριστήσω τους φίλους μου που στάθηκαν δίπλα μου και βοήθησαν με όποιο τρόπο ο καθένας μπορούσε.



Περίληψη

Στην παρούσα διπλωματική εργασία σχεδιάστηκε και αναπτύχθηκε μία Android Εφαρμογή Τουριστικού Περιεχομένου, επικεντρωμένη στη Δυτική Μακεδονία, μαζί με υποστηρικτική Ιστοσελίδα. Πρόκειται για μια εφαρμογή και έναν ιστότοπο, τα οποία δημιουργήθηκαν με σκοπό τη διευκόλυνση της εύρεσης τουριστικών πληροφοριών και της πλοήγησης σε διάφορα σημεία ενδιαφέροντος.

Πιο συγκεκριμένα, μέσω ενός φιλικού προς το χρήστη γραφικού περιβάλλοντος και με τη βοήθεια της πλατφόρμας του Google Maps API, ο χρήστης μπορεί εύκολα, γρήγορα και άμεσα από την Ιστοσελίδα να αποθηκεύσει μία περιοχή ή μια περιήγηση. Με την αποθήκευση τους αμέσως είναι διαθέσιμες στην Κινητή Εφαρμογή.

Με τη σύνδεση του χρήστη στην εφαρμογή, εντοπίζεται η τοποθεσία του και είναι δυνατή η πλοήγησή του όπου και αν βρίσκεται σε κάποια από τις αποθηκευμένες περιοχές ή περιηγήσεις.

Οι διαχειριστές του συστήματος έχουν τις δυνατότητες των απλών χρηστών, αλλά μπορούν επιπλέον να επεξεργάζονται ή να διαγράφουν τους χρήστες, να βλέπουν, να επεξεργάζονται και να διαγράφουν όλες τις αποθηκευμένες περιοχές και περιηγήσεις, και να κάνουν κάποιο άλλο χρήστη διαχειριστή.

Λέξεις κλειδιά: Χάρτες, GPS, PHP, MySQL, Ajax, JavaScript, Προγραμματισμός Διαδικτύου, Προγραμματισμός Εφαρμογών, Android, Βάση Δεδομένων, Τουρισμός, Δυτική Μακεδονία



Abstract

Title: Design and Development of an Android Application with Supportive Website in order to create a Travel Guide for Western Macedonia

In this diploma thesis, an Android Application for tourist content, focused on Western Macedonia, along with a Supportive Website was designed and developed. This application and website were created to facilitate the search of touristic information, as also the navigation at various points of interest.

More specifically, through a user-friendly graphical environment and with the help of the Google Maps API platform, the user can easily, quickly and directly store a place or a tour on the website. By saving it, it is immediately available in the Mobile Application.

When the user starts the application, the current location is located and the user can be easily navigated in one of the saved areas or tours.

System administrators have the same capabilities as simple users but additionally, they can process or delete users, view, edit, and delete all of the saved areas and tours, and make another user also an admin.

Keywords: Maps, GPS, PHP, MySQL, Ajax, JavaScript, Web programming, App programming, Android, Database, Tourism, Western Macedonia

Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο “Σχεδιασμός και Ανάπτυξη Εφαρμογής Android με Υποστηρικτική Ιστοσελίδα για Δημιουργία Ταξιδιωτικού Οδηγού Δυτικής Μακεδονίας” καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Δασυγένη Μηνά αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

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

Copyright (C) Ρέγκου Ελένη & Δασυγένης Μηνάς , 2018 , Κοζάνη

Περιεχόμενα

1	Εισαγωγή	21
1.1	Ορισμός του προβλήματος	23
1.2	Σκοπός και Πλεονεκτήματα	24
1.3	Σχετική Έρευνα	25
1.4	Αντίστοιχα Εμπορικά Συστήματα	27
1.4.1	iGuide	28
1.4.2	Field Trip	28
1.4.3	Strayboots	28
1.4.4	AroundMe	29
1.5	Διάρθρωση κειμένου	29
1.6	Σύνοψη Κεφαλαίου	30
2	Θεωρητικό Υπόβαθρο	31
2.1	Προγραμματισμός Διαδικτύου	31
2.1.1	HTML	31
2.1.2	CSS	32
2.1.3	JavaScript	32
2.1.4	jQuery	33
2.1.5	PHP	34
2.1.6	AJAX	34
2.1.7	MySQL	35
2.1.8	Google Maps API	36
2.1.9	Bootstrap	37
2.2	Εργαλεία Λογισμικού	37
2.2.1	Notepad++	37
2.2.2	phpMyAdmin	38

2.2.3	WinSCP	39
2.2.4	PuTTY	39
2.2.5	MySQL Workbench	40
2.2.6	GIMP	41
2.2.7	Android Studio	41
2.3	Σύνοψη Κεφαλαίου	42
3	Ανάλυση και Σχεδίαση	43
3.1	Ανάλυση και Σχεδίαση Ιστοχώρου	43
3.2	Ρόλοι	43
3.3	Απαιτήσεις	44
3.4	Περιπτώσεις Χρήσης	44
3.5	Σχεδιασμός Βάσης Δεδομένων	46
3.5.1	Πίνακας users	46
3.5.2	Πίνακας allplaces	48
3.5.3	Πίνακας tours	49
3.5.4	Σχεσιακό Διάγραμμα Βάσης Δεδομένων	50
3.6	Ασφάλεια	51
3.7	Ανάλυση και Σχεδίαση Κινητής Εφαρμογής	53
3.7.1	Πρωτόκολλο Επικοινωνίας Εφαρμογής με Ιστότοπο	53
3.7.2	Απαιτήσεις Κινητής Εφαρμογής	53
3.7.3	Περιπτώσεις και Δικαιώματα Χρήσης	54
3.8	Σύνοψη Κεφαλαίου	54
4	Λειτουργίες Ιστοχώρου	57
4.1	Λειτουργίες Ανώνυμου Χρήστη	57
4.1.1	Σύνδεση Χρήστη	57
4.1.2	Εγγραφή Χρήστη	58
4.1.3	Ανάκτηση Κωδικού	59
4.2	Λειτουργίες Συνδεδεμένου Χρήστη	60
4.2.1	Χάρτες	61
4.2.2	Περιηγήσεις	64
4.2.3	Σχετικά Με	67

4.2.4	Αποσύνδεση	68
4.2.5	Ρυθμίσεις Λογαριασμού	68
4.3	Λειτουργίες Διαχειριστή	69
4.3.1	Χρήστες	70
4.3.2	Επιπλέον Δυνατότητες	71
4.4	Ανάλυση Σημαντικών Σημείων Κώδικα Ιστοσελίδας	72
4.5	Σύνοψη Κεφαλαίου	82
5	Λειτουργίες Κινητής Εφαρμογής	85
5.1	Λειτουργίες Χρήστη	85
5.1.1	Βασικό Μενού	86
5.1.2	Χάρτες	87
5.1.3	Περιοχές	88
5.1.4	Λεπτομέρειες Περιοχής	89
5.1.5	Πλοήγηση σε Περιοχή	90
5.1.6	Άφιξη σε Περιοχή	91
5.1.7	Κοντινές Περιοχές	92
5.1.8	Περιηγήσεις	93
5.1.9	Λεπτομέρειες Περιήγησης	94
5.1.10	Πλοήγηση σε Περιήγηση	95
5.1.11	Σχετικά Με	96
5.2	Ανάλυση Σημαντικών Σημείων Κώδικα Εφαρμογής	96
5.3	Σύνοψη Κεφαλαίου	105
6	Επίλογος	107
6.1	Σύνοψη Συστήματος	107
6.2	Μετρικά Κώδικα Συστήματος	108
6.2.1	Μετρικά Κώδικα Ιστοχώρου	108
6.2.2	Μετρικά Κώδικα Εφαρμογής	109
6.3	Συμπεράσματα	109
6.4	Ανάλυση SWOT	110
6.4.1	Δυνατά Σημεία	110
6.4.2	Αδυναμίες	110

6.4.3	Ευκαιρίες	111
6.4.4	Απειλές	111
6.5	Μελλοντικές Επεκτάσεις	111
6.5.1	Προσθήκη Περισσότερων Περιοχών	111
6.5.2	Ανάπτυξη σε Διαφορετική Πλατφόρμα	112
6.5.3	Προσθήκη Αξιολογήσεων Χρηστών	112
6.5.4	Επαυξημένη Πραγματικότητα	112
6.5.5	Εικονική Πραγματικότητα	112
6.5.6	Ηχητικές Περιγραφές	112
6.6	Σύνοψη Κεφαλαίου	113
Παραρτήματα		117
A' Οδηγίες Εγκατάστασης Ιστοχώρου		119
B' Οδηγίες Εγκατάστασης Εφαρμογής		121

Κατάλογος Σχημάτων

2.1	Παράδειγμα HTML	32
2.2	Παράδειγμα CSS	32
2.3	Παράδειγμα JavaScript	33
2.4	Παράδειγμα jQuery	33
2.5	Παράδειγμα PHP	34
2.6	Παράδειγμα AJAX	35
2.7	Παράδειγμα MySQL	35
2.8	Παράδειγμα Google Maps API	36
2.9	Παράδειγμα Bootstrap	37
2.10	Notepad++	38
2.11	phpMyAdmin	38
2.12	WinSCP	39
2.13	PuTTY	40
2.14	MySQL Workbench	40
2.15	GIMP	41
2.16	Android Studio	42
3.1	Σχεσιακό Διάγραμμα Βάσης Δεδομένων	51
4.1	Οθόνη Σύνδεσης Χρήστη	58
4.2	Οθόνη Εγγραφής Χρήστη	58
4.3	Οθόνη Ανάκτησης Κωδικού	59
4.4	Οθόνη Βασικού Μενού Απλού Χρήστη	60
4.5	Οθόνη Χαρτών	61
4.6	Παράθυρο Πληροφοριών	62
4.7	Αλλαγή Συντεταγμένων Περιοχής	62
4.8	Παράθυρο Επεξεργασίας	63

4.9	Επεξεργασία Εικόνας	63
4.10	Εισαγωγή Περιοχής	64
4.11	Οθόνη Αποθηκευμένων Περιηγήσεων	65
4.12	Οθόνη Προσθήκης Νέας Περιήγησης	65
4.13	Οθόνη Προβολής Περιήγησης	66
4.14	Οθόνη Επεξεργασίας Περιήγησης	67
4.15	Οθόνη Σχετικά Με	67
4.16	Οθόνη Ρυθμίσεις Λογαριασμού	68
4.17	Οθόνη Επεξεργασίας Λογαριασμού	69
4.18	Οθόνη Αλλαγής Κωδικού	69
4.19	Οθόνη Μενού Διαχειριστή	70
4.20	Οθόνη Αποθηκευμένων Χρηστών	70
4.21	Οθόνη Επεξεργασίας Στοιχείων Χρήστη	71
4.22	adminsession.php	72
4.23	connect.php	73
4.24	random_str()-forgottenpassword.php	73
4.25	forgottenpassword.php	74
4.26	getjson.php	74
4.27	https.php	75
4.28	login.php	76
4.29	Σχεδίαση Αποθηκευμένης Περιοχής - maps.php	77
4.30	Σχεδίαση Νέας Περιοχής - maps.php	77
4.31	Επιλογή Περιοχής και Αλλαγή Συντεταγμένων - maps.php	78
4.32	Χρωματισμός Επιλεγμένης Εγγραφής στο Χάρτη - readRecords.php	78
4.33	Αποθήκευση Εικόνας - savedata.php	79
4.34	savepass.php	80
4.35	Έλεγχος - saveuser.php	81
4.36	Αποθήκευση - saveuser.php	81
4.37	session.php	82
5.1	Οθόνη Βασικού Μενού	86
5.3	Οθόνη Περιοχές	88
5.4	Οθόνη Λεπτομέρειες Περιοχής	89

5.5	Οθόνη Πλοήγηση σε Περιοχή	90
5.6	Οθόνη Άφιξη σε Περιοχή	91
5.7	Οθόνη Κοντινές Περιοχές	92
5.8	Οθόνη Περιηγήσεις	93
5.9	Οθόνη Λεπτομέρειες Περιήγησης	94
5.10	Οθόνη Πλοήγηση σε Περιήγηση	95
5.11	Οθόνη Σχετικά Με	96
5.12	Έλεγχος αν Βρίσκεται μέσα σε Περιοχή	97
5.13	onsensorchanged()	98
5.14	Κίνηση Εικόνας	99
5.15	Λήψη Τοποθεσίας	100
5.16	loadjsontask.java	101
5.17	placestoring.java	102
5.18	response.java	102
5.19	Comparator<T>	102
5.20	compare()	102
5.21	Collections.sort()	102
5.22	Ταξινόμηση Περιοχών Περιήγησης	104

Κατάλογος Πινάκων

3.1	Πίνακας Χρηστών (users)	46
3.2	Πίνακας Περιοχών (allplaces)	48
3.3	Πίνακας Περιηγήσεων (tours)	50
6.1	Μετρικές Ιστοχώρου	108
6.2	Μετρικές Εφαρμογής	109

Κεφάλαιο 1

Εισαγωγή

Ο τουρισμός, είτε ως διαχρονικό κοινωνικό φαινόμενο είτε ως διαδεδομένος τρόπος ψυχαγωγίας, ορίζεται όταν εκείνος που ταξιδεύει διαμένει ή επισκέπτεται περιοχή διαφορετική από τη δική του και περιλαμβάνει συνήθως την επίσκεψη σε προορισμούς για τουριστικούς – ψυχαγωγικούς λόγους. Σήμερα, μεταξύ άλλων, η τεχνολογική ανάπτυξη, η εξέλιξη των μεταφορικών μέσων, οι ανέσεις των καταλυμάτων και οι εξειδικευμένες παρεχόμενες υπηρεσίες, προσδίδουν μια ραγδαία αύξηση στη μετακίνηση πληθυσμών για τουριστικούς λόγους.

Οι άνθρωποι επιθυμούν ενδιαφέρουσες και αξέχαστες εμπειρίες και με τα ταξίδια μπορούν να ικανοποιήσουν την ανάγκη τους αυτή. Το ταξίδι γενικά προσφέρει άφθονες ευκαιρίες για κοινωνικό-πολιτισμική εξερεύνηση και προσωπική ενδοσκόπηση. Καθώς οι άνθρωποι ταξιδεύουν, βρίσκουν ευκαιρίες κοινωνικοποίησης και γνωριμίας διαφορετικών πολιτισμών και εμπειριών. Πολλές φορές οι άνθρωποι ταξιδεύουν όχι μόνο για να συναντήσουν αγαπημένα τους πρόσωπα που ζουν μακριά αλλά και για να επισκεφθούν μέρη και να απολαύσουν εμπειρίες που δεν υπάρχουν στο μέρος που ζουν.

Εκτός από την εμπειρία του κοινωνικό-πολιτιστικού πλούτου οι άνθρωποι ξεκινούν ταξίδια για να μάθουν περισσότερα για τους εαυτούς σας και για να επισκεφθούν μέρη ιστορικής σημασίας. Αυτές οι ταξιδιωτικές εμπειρίες προσφέρουν στους ανθρώπους μια ευπρόσδεκτη αλλαγή από τις καθημερινές τους συνήθειες και παρουσιάζουν ευκαιρίες για την δημιουργία καινούργιων αναμνήσεων.

Καθώς οι άνθρωποι ζουν αυτές τις νέες εμπειρίες ενώ ταξιδεύουν, επισκέπτονται διάφορα αξιοθέατα που χρησιμεύουν ως βασικά σημεία ενδιαφέροντος (POI) στα ταξίδια τους. Τα αξιοθέατα χρησιμοποιούνται όχι μόνο ως εργαλεία πλοήγησης,

αλλά και ως εικονικά μέρη που μπορούν να επισκεφθούν οι τουρίστες. Αυτά τα σημεία συνήθως αντιπροσωπεύουν κομμάτια ιστορίας, πολιτισμού, αρχιτεκτονικής ή φυσικής ομορφιάς.

Ο τουρισμός έχει μεγάλη σημασία όχι μόνο για τους ανθρώπους αλλά και για την για την παγκόσμια οικονομία. Η αυξανόμενη χρήση των κινητών τηλεφώνων, όπως και η φορητότητα και διαθεσιμότητα που προσφέρουν στους χρήστες τα έχει καταστήσει πλέον ως την επικρατέστερη πλατφόρμα για τουριστικές υπηρεσίες [26]. Επίσης τα κινητά τηλέφωνα έχουν βελτιωθεί όσον αφορά τη διάρκεια ζωής της μπαταρίας και την επεξεργαστική δύναμη, χαρακτηριστικά τα οποία τα κάμουν ιδανικά για την υποστήριξη πολύπλοκων εφαρμογών όπως είναι οι τουριστικές.

Τα τελευταία χρόνια έχει σημειωθεί σημαντική αύξηση στη συλλογή και χρήση γεωγραφικών δεδομένων υψηλής ανάλυσης τα οποία έχουν τη δυνατότητα να ανιχνεύουν τις κινήσεις αντικειμένων και ανθρώπων χάρη στην ανάπτυξη διαφόρων τεχνολογιών όπως οι ηλεκτρονικοί χάρτες και οι δορυφορικές εικόνες της γης, τα οποία είναι διαθέσιμα χωρίς κόστος και εύκολα στη χρήση. Έρευνες στην ανθρώπινη συμπεριφορά έχουν οδηγήσει στην αύξηση της χρήσης του GPS και άλλων συσκευών παρακολούθησης (Bluetooth, WI-FI κλπ.) σε διάφορους τομείς όπως η ιατρική, η περιβαλλοντική υγεία, η καρδιολογία, η φυσιολογία αλλά και στον τουρισμό [10], [25] [12]. Τα τελευταία χρόνια οι ερευνητές έχουν επωφεληθεί πολύ από την ανάπτυξη και διάδοση των έξυπνων κινητών (smartphones), τα οποία μπορούν να μεταφέρουν τις ανακτημένες τοποθεσίες εύκολα και σε προσιτές τιμές, όπως επίσης έχουν την επεξεργαστική δύναμη να αναλύσουν τον όγκο των δεδομένων που συλλέγονται [28].

Στην παρούσα εργασία, με όνομα "Σχεδιασμός και Ανάπτυξη Εφαρμογής Android με Υποστηρικτική Ιστοσελίδα για Δημιουργία Ταξιδιωτικού Οδηγού Δυτικής Μακεδονίας", παρουσιάζετε ο σχεδιασμός και η υλοποίηση μιας εφαρμογής για κινητά με την οποία οι χρήστες μπορούν να βρουν πολύτιμες πληροφορίες σχετικά με τα διάφορα αξιοθέατα της Δυτικής Μακεδονίας όπως επίσης και να βρουν τα επιθυμητά μέρη που είναι πιο κοντά στην περιοχή που ήδη βρίσκονται. Με τη χρήση της παρούσας εφαρμογής οι χρήστες οι χρήστες μπορούν να λάβουν λεπτομερείς πληροφορίες για σημαντικά αξιοθέατα σε μορφή σε κείμενου, εικόνων και χαρτών με μεγαλύτερη ευκολία οποιαδήποτε στιγμή και οπουδήποτε βρίσκονται. Μπορούν

επίσης να δουν προτεινόμενες περιηγήσεις, οι οποίες αποτελούνται από τις αποθηκευμένες περιοχές, και να πλοηγηθούν σε αυτές όσο και σε μεμονωμένες περιοχές. Δυνατή από τον χρήστη είναι και η αναζήτηση διευθύνσεων και συντεταγμένων.

Η εφαρμογή είναι βασισμένη σε υπολογιστική συννέφου (cloud computing) ώστε να αποφευχθεί η δημιουργία μιας βαριάς και ογκώδης εφαρμογής που μπορεί να καταλαμβάνει μεγάλο μέρος του αποθηκευτικού χώρου του κινητού τηλεφώνου των χρηστών. Επίσης συνδυάζοντας τα πλεονεκτήματα τόσο των υπολογιστών κινητής τηλεφωνίας όσο και του cloud computing, μπορεί να παρέχει τη βέλτιστη υπηρεσία για τους χρήστες της εφαρμογής. Με τη βοήθεια του συννέφου (cloud) οι χρήστες του συστήματος μπορούν να προσεγγίσουν αυτά τα δεδομένα από οπουδήποτε στον κόσμο. Στην συνοδευτική ιστοσελίδα της πλατφόρμας μπορούν να αποθηκευτούν περιοχές και αξιοθέατα με διάφορες σχετικές πληροφορίες, αλλά και διαδρομές αποτελούμενες από αυτά.

1.1 Ορισμός του προβλήματος

Ένα από τα κίνητρα για την επιλογή αυτού του θέματος υπήρξε η παρατήρηση πως στην Δυτική Μακεδονία δεν υπάρχει μεγάλη ανάδειξη των τουριστικών και φυσικών της αξιοθέατων. Και για τέτοιες αναξιοποίητες περιοχές, που είναι από τις λιγότερο γνωστές και τις όχι πιο εύκολα προσβάσιμες της Ελλάδας έχει μεγάλη σημασία η προώθηση του τουρισμού της, ίσως όχι με την παραδοσιακή έννοια (νησιά, παραλίες) αλλά σε ότι έχει η κάθε περιοχή να προσφέρει.

Στην περιοχή της Δυτικής Μακεδονίας υπάρχουν πολλά αξιοθέατα που είναι άγνωστα και αναξιοποίητα, αλλά και αφύλακτες τοποθεσίες πολιτιστικού ή φυσικού περιβάλλοντος για τις οποίες είτε δεν υπάρχουν διαθέσιμες πληροφορίες, είτε είναι δυσεύρετες. Δεδομένου ότι ο Πολιτικός τουρισμός και ο Οικοτουρισμός είναι τομείς ζωτικής σημασίας για την ανάπτυξη της οικονομίας της κάθε περιοχής, είναι πολύ σημαντική η ανάπτυξή τους. Μεγάλο ρόλο στην ανάπτυξη αυτή μπορούν να παίξουν και οι ηλεκτρονικές υπηρεσίες τουρισμού και συναφών υπηρεσιών ψηφιακού περιεχομένου. Έτσι ο κάθε επισκέπτης ενός τέτοιου σημείου ενδιαφέροντος θα μπορεί να έχει επί τόπου, άμεση και έγκυρη πληροφόρηση.

Λαμβάνοντας υπόψη την επίδραση του τουρισμού στην οικονομία της Ελλάδας, υπάρχει η ανάγκη για την ανάπτυξη μιας κινητής εφαρμογής με βάση το GPS για

την περιοχή της Δυτικής Μακεδονίας η οποία θα συμβάλει στην τόνωση και την αύξηση του ποσοστού των τουριστών που έρχονται στη περιοχή και κατά συνέπεια θα βοηθήσει την αύξηση των εσόδων στην οικονομία της μέσω του τουριστικού τομέα. Παρόλα αυτά ως τώρα δεν έχει υπάρξει καμία προηγμένη κινητή εφαρμογή με χρήση GPS για τον τουρισμό στην περιοχή της Δυτικής Μακεδονίας.

1.2 Σκοπός και Πλεονεκτήματα

Για τη επίλυση των προβλημάτων που αναλύθηκαν στην προηγούμενη ενότητα, υλοποιήθηκε η παρούσα διπλωματική εργασία η οποία αποσκοπεί στη ύπαρξη ενός τουριστικού οδηγού μέσω κινητού, προσβάσιμο από το προσωπικό έξυπνο κινητό του κάθε χρήστη, για την ανάδειξη ενός μεγαλύτερου αριθμού χώρων από τους υπάρχοντες, βελτίωση της εμπειρίας της επίσκεψης όπως και διευκόλυνση της περιπλάνησης σε τόπους ενδιαφέροντος.

Αυτή η εργασία αποτελείται από έναν ιστότοπο οποίος μέσα από ένα ευχάριστο περιβάλλον δίνει την δυνατότητα σε ανθρώπους που θέλουν να αναδείξουν κάποιες περιοχές, δήμους που θέλουν να δείξουν τα όχι και τόσο γνωστά τους μνημεία και αξιοθέατα, ακόμα και επιχειρήσεις που θέλουν να διαφημιστούν, να μπορούν εύκολα να αποθηκεύσουν κάποια περιοχή οποιουδήποτε μεγέθους- από ένα κτήριο μέχρι και ολόκληρη πόλη. Να δώσουν όνομα, περιγραφή και εικόνα. Και στη συνέχεια αφού αποθηκευτεί μπορεί να χρησιμοποιηθεί ως αυτούσια ή ως μέρος μιας διαδρομής. Η χρήση της εφαρμογής προσφέρει δύο βασικά πλεονεκτήματα:

1. **Στην οικονομία:** Ο τουρισμός γενικά φέρνει χρήματα σε κάθε περιοχή, αλλά εξαρτάται από τη δημοσιότητα, την προσβασιμότητα και την οικονομική προσιτότητα του τουριστικού ταξιδιού. Με τη βοήθεια της εφαρμογής κινητού οι τουρίστες θα μπορούν να ταξιδεύουν πιο οικονομικά, καθώς θα υπάρχει ελάχιστη ή και καθόλου ανάγκη να χρησιμοποιήσουν υπηρεσίες ταξιδιωτικού οδηγού και θα μπορούν να ξοδέψουν περισσότερα χρήματα στην τοπική κοινωνία. Επίσης η εφαρμογή θα προσελκύσει τους τουρίστες λόγω της προσιτότητας και της δημοσιότητας που θα δώσει στην περιοχή της Δυτικής Μακεδονίας.
2. **Στους τουρίστες:** Η εφαρμογή παρέχει σημαντικές ταξιδιωτικές πληροφορί-

ες για τους τουρίστες όπως τα σημεία ενδιαφέροντος (Points Of Interest - POI) που μπορούν να επισκεφθούν. Ακόμη η χρήση της εφαρμογής προσφέρει το πλεονέκτημα της εύκολης πρόσβασης στα αξιοθέατα της περιοχής από οπουδήποτε, ακόμα και εν κινήσει. Επίσης περιλαμβάνει ένα καλό εργαλείο το οποίο προσφέρει εναλλακτικές διαδρομές και προτάσεις περιηγήσεων στο πλησιέστερο σημείο ενδιαφέροντος (POI) για τους τουρίστες και ειδοποιήσεις όταν φτάσει ο χρήστης σε κάποια από τις περιοχές.

1.3 Σχετική Έρευνα

Σε αυτή την ενότητα παρουσιάζονται παρόμοιες έρευνες που έχουν γίνει στον τομέα των τουριστικών εφαρμογών με χρονολογική σειρά.

Οι Umlauf (2003) κ.α. δημιούργησαν μια εφαρμογή βασισμένη στην τοποθεσία του χρήστη που στοχεύει στην υποστήριξη των τουριστών. Η εφαρμογή προσφέρει χάρτες, φωνητικές εντολές, εντοπισμό και δρομολόγηση. Ακόμη προσφέρει ημερολόγιο των ταξιδιών για να μένουν σε επαφή οι χρήστες με την οικογένεια και τους φίλους τους πίσω στο σπίτι [4].

Οι Mathkour κ.α. (2011) πρότειναν και ανέπτυξαν ένα GPS σύστημα εντοπισμού κινητών υπηρεσιών ώστε να βοηθήσει άτομα σε διαφορετικά κοινωνικά στρώματα να βρουν διευθύνσεις, υπηρεσίες έκτακτης ανάγκης και άλλες υπηρεσίες ενδιαφέροντος μέσω των κινητών συσκευών τους. Η απόσταση μεταξύ του χρήστη και της επιθυμητής υπηρεσίας ήταν σε θέση να υπολογίζεται από το προτεινόμενο σύστημα. Επίσης το σύστημα είναι ευέλικτο και επεκτάσιμο για την εύκολη ενσωμάτωση νέων υπηρεσιών [18].

Οι Gavalas και Kenteris (2011) διεύρυναν την ιδέα των προτεινόμενων συστημάτων για ταξίδια (recommender systems) αξιοποιώντας τεχνικές φιλτραρίσματος και λαμβάνοντας υπόψη πληροφορίες του περιβάλλοντος (όπως ο χρόνος, οι καιρικές συνθήκες, η τρέχουσα τοποθεσία του χρήστη και οι τοποθεσίες που έχει ήδη επισκεφθεί) για την εξαγωγή βελτιωμένων προτάσεων. Οι ερευνητές πρότειναν την εγκατάσταση ασύρματων δικτύων αισθητήρων (WSN) γύρω από τα τουριστικά αξιοθέατα για να προσφέρουν στους χρήστες της εφαρμογής ένα βολικό και φθηνό τρόπο για τη μεταφόρτωση τουριστικών πληροφοριών και αξιολογήσεων για τα σημεία ενδιαφέροντος (POI) [6].

Επίσης οι Gavallas κ.α. (2011) μελέτησαν το πρόβλημα των εξατομικευμένων προτάσεων αποκομίζοντας ημερήσιες τουριστικές διαδρομές για τουρίστες που επισκέπτονται οποιοδήποτε προορισμό. Χρησιμοποίησαν τα μέρη ενδιαφέροντος (POI) που επέλεξαν οι χρήστες ότι θα επιθυμούσαν ενδεχομένως να επισκεφθούν και δημιούργησαν τη βέλτιστη διαδρομή λαμβάνοντας υπόψη λαμβάνοντας υπόψη τον διαθέσιμο χρόνο για να επισκεφθούν τα αξιοθέατα, τις ημέρες που τα αξιοθέατα είναι ανοιχτά, τις προτιμήσεις των χρηστών και το μέσο αριθμό επισκεπτών για αυτά τα αξιοθέατα [7].

Οι Noguera κ.α. (2012) παρουσίασαν μια νέα εφαρμογή κινητού η οποία συνδυάζει την κινητή 3D GIS αρχιτεκτονική και μια υβριδική μηχανή συστάσεων. Η εφαρμογή επιτρέπει στους τουρίστες να επωφελούνται από χαρακτηριστικά όπως η ευαίσθητη σε πραγματικό χρόνο τοποθεσία, συστάσεις και 3D διεπαφή με βάση το χάρτη [21].

Οι Yang και Hwang (2013) παρατήρησαν ότι τα συστήματα προτάσεων στον τομέα του τουρισμού έχουν αυξηθεί την τελευταία δεκαετία. Στην έρευνα τους προτείνουν την πρόταση αξιοθέατων εξετάζοντας τις βαθμολογίες άλλων τουριστών για το αξιοθέατο. Η προτεινόμενη εφαρμογή "iTravel" χρησιμοποιεί κινητές επικοινωνίες τύπου peer-to-peer για την ανταλλαγή αξιολογήσεων μέσω των κινητών τηλεφώνων [32].

Οι Panahi κ.α. (2013) ανέφεραν από την έρευνά τους ότι ο κινητός τουρισμός είναι μια σχετικά νέα τάση στον τουρισμό η οποία περιλαμβάνει τουριστικούς οδηγούς σε κινητές συσκευές. Η εργασία τους περιγράφει το σχεδιασμό και την ανάπτυξη μιας εφαρμογής του κινητού τουρισμού για πολιτιστικό τουρισμό στη Μαλαισία με τη χρήση μιας cloud-based πλατφόρμας. Το σύστημα τους αποτελείται από τρία επίπεδα. Το πρώτο είναι η front-end εφαρμογή σε iOS λειτουργικό σύστημα. Το δεύτερο είναι ένα web service το οποίο παράγει XML από την σχεσιακή βάση δεδομένων για την ανταλλαγή των δεδομένων από την εφαρμογή στους cloud servers. Το τρίτο επίπεδο είναι το back end στην πλατφόρμα cloud AWS, το οποίο προσφέρει υπηρεσίες όπως το Apache Web Server στην υπηρεσία σχεσιακής βάσης δεδομένων του Amazon (RDs) και σε ένα cloud storage που χρησιμοποιεί την απλή υπηρεσία αποθήκευσης της Amazon (S3). Η εφαρμογή που ανέπτυξαν επιτρέπει επίσης την αποθήκευση προσωρινών εικόνων στο τοπικό αποθηκευτικό χώρο της

κινητής συσκευής για να παρέχει ορισμένες υπηρεσίες χωρίς να χρειάζεται σύνδεση στο διαδίκτυο [22].

Οι Anacleto κ.α. (2014) βρήκαν από τη μελέτη τους ότι τα συστήματα συστάσεων των τουριστών έχουν αυξηθεί τελευταία κυρίως λόγω των πιο σχετικών συστημάτων στον τομέα τα τελευταία χρόνια και παρουσιάζει το "PSiS mobile" το οποίο είναι ένα σύστημα προτάσεων και έχει σχεδιαστεί για να υποστηρίζει έναν τουρίστα κατά τη διάρκεια των διακοπών του. Παρέχει προτάσεις σχετικά με τα σημεία ενδιαφέροντος με βάσει τις προτιμήσεις των τουριστών και του χρήστη και τα διαθέσιμα αξιοθέατα. Επίσης, οι ερευνητές πρότειναν τον δυναμικό σχεδιασμό της διαδρομής επίσκεψης και ανέπτυξαν μια υπηρεσία που λειτουργεί σαν ταξιδιωτικό ημερολόγιο και καταγράφει τα μέρη που επισκέφθηκαν ώστε να τους βοηθήσει να θυμούνται πως ήταν το ταξίδι τους [3].

Η ανάπτυξη του τουρισμού συνεχίζει να αυξάνεται παρά την πολιτική αστάθεια, τις φθίνουσες οικονομίες και τον πολλαπλασιασμό της τρομοκρατίας. Το δέλεαρ της εξερεύνησης νέων περιοχών συνεχίζει να προσελκύει εκατομμύρια ανθρώπους κάθε χρόνο. Ένα από τα πράγματα που κάνουν το τουριστικό ταξίδι πιο ελκυστικό και εύκολο είναι ο αυξανόμενος αριθμός των εφαρμογών που χρησιμοποιούν GPS και λειτουργούν ως ταξιδιωτικοί οδηγοί. Όπως προαναφέρθηκε στη σχετική έρευνα οι τουριστικές εφαρμογές είναι σε θέση να παρέχουν πληροφορίες για την κάθε περιοχή με ευκολία και αποτελεσματικότητα καθώς είναι βασισμένες σε ελαφριές συνδυαστικές τεχνολογίες που κάνουν χρήση πολλαπλών πηγών δεδομένων για τη δημιουργία προστιθέμενης αξίας στις υπηρεσίες τους.

1.4 Αντίστοιχα Εμπορικά Συστήματα

Υπάρχουν πολλές τουριστικές εφαρμογές γενικού περιεχομένου, άλλες προσφέρουν περισσότερες λειτουργίες άλλες λιγότερες, άλλες είναι με κάποια χρέωση άλλες δωρεάν, μερικές επικεντρώνονται σε συγκεκριμένες γεωγραφικές περιοχές και άλλες προσπαθούν να τις περιλάβουν όλες. Παρακάτω περιγράφονται μερικά παρόμοια με το παρόν συστήματα εφαρμογών που κυκλοφορούν:

1.4.1 iGuide

Μία σχετική εφαρμογή είναι η iGuide από την ομάδα Mobics. Το iGuide δίνει στο χρήστη τη δυνατότητα να περιηγηθεί στην Παλιά Πόλη της Ξάνθης, στη Μαρώνεια, στο Πολύστυλο Αβδήρων, στην Αλυκή της Θάσου και στα Στενά του Νέστου και με λειτουργία φωνητικής καθοδήγησης δίνει όλες τις απαραίτητες πληροφορίες για αξιοθέατα και σημεία που συναντά. Παρέχει τις δυνατότητες να βρει το σημείο του χρήστη στον χάρτη, να του δείξει διαθέσιμες διαδρομές, να δει σχετικά πολυμέσα και άλλα κοντινά σημεία στο χάρτη να χαρτογραφήσει σημεία της πόλης να τραβήξει φωτογραφίες διαφόρων σημείων και να τα τοποθετήσει στο χάρτη. Μπορεί ακόμα να αποθηκεύσει τα σημεία που έχει επισκεφθεί, τις διαδρομές που έχει ολοκληρώσει. Παρέχει χάρτες που λειτουργούν και χωρίς πρόσβαση στο ίντερνετ, περιγραφές για περιοχές φυσικού, ιστορικού ενδιαφέροντος, ξενάγηση με ομιλία, προτεινόμενες διαδρομές, και τη δυνατότητα να δημιουργήσει τις δικές του διαδρομές. Διατίθεται δωρεάν μέσω του PlayStore.

1.4.2 Field Trip

Μια άλλη εφαρμογή είναι η Field Trip από την Niantic, Inc. Η εφαρμογή Field Trip εκτελείται στο παρασκήνιο του τηλεφώνου του χρήστη και όταν πλησιάζει κάποιο σημείο ενδιαφέροντος, εμφανίζει μια κάρτα με λεπτομέρειες σχετικά με την τοποθεσία. Ακόμα εάν υπάρχουν συνδεδεμένα ακουστικά, μπορεί να ακούσει αυτές τις πληροφορίες. Περιλαμβάνει πολλές κατηγορίες ενδιαφέροντος όπως: Αρχιτεκτονική, Ιστορικά μέρη και Συμβάντα, Lifestyle, Προσφορές και Ευκαιρίες, Φαγητό, Ποτό και Διασκέδαση, Κινηματογραφικές αίθουσες, Καλλιτεχνικές εκδηλώσεις εξωτερικών χώρων και Άγνωστα σημεία ενδιαφέροντος. Επίσης η εφαρμογή Field Trip μπορεί να χρησιμοποιηθεί και ενώ οδηγείτε εντοπίζοντας και "μιλώντας" αυτόματα για τα ενδιαφέροντα μέρη που περνάει ο χρήστης. Μπορεί να βρεθεί μέσω του PlayStore, δωρεάν.

1.4.3 Strayboots

Μια ακόμα παρόμοια εφαρμογή είναι η Strayboots από την ομάδα Stray Boots. Δείχνει τα ενδιαφέροντα σημεία της πόλης με πιο διασκεδαστικό τρόπο συνδυάζοντας την περιήγηση με το παιχνίδι. Δίνει την δυνατότητα εκδρομών σε πολλές

μεγάλες πόλεις των ΗΠΑ, με διάφορες προκλήσεις σε μεγάλα αξιοθέατα ανά εκδρομή και ενδιαφέροντα στοιχεία για κάθε τοποθεσία που επισκέπτεται ο χρήστης. Ακόμα δίνει τη δυνατότητα για ανάγνωση πληροφοριών για διάφορες τοποθεσίες ακόμα και αν ο χρήστης δεν βρίσκεται εκεί. Ένα από τα αρνητικά της είναι πως είναι επικεντρωμένη σε πόλεις και τοποθεσίες των ΗΠΑ. Είναι δωρεάν και διανέμεται μέσω του PlayStore.

1.4.4 AroundMe

Η εφαρμογή Aroundme από την ομάδα Flying Code είναι μια εφαρμογή που εντοπίζει τη θέση του χρήστη και του εμφανίζει τις πλησιέστερες τοποθεσίες. Το AroundMe όμως, σε σχέση με τη παρούσα εφαρμογή, επικεντρώνεται σε υπηρεσίες και καταστήματα, όπως την πλησιέστερη τράπεζα, βενζινάδικο, νοσοκομείο, ξενοδοχείο, σινεμά, εστιατόριο, σούπερ μάρκετ, ταξί κλπ, και όχι σε αξιοθέατα. Η εφαρμογή εμφανίζει μια πλήρη λίστα με όλες τις επιχειρήσεις στην κατηγορία που έχει επιλέξει ο χρήστης μαζί με την απόσταση τους από το σημείο που βρίσκεται. Προσφέρεται δωρεάν μέσω του PlayStore.

Κατά κύριο λόγο, η παρούσα εφαρμογή διαφοροποιείται σε σχέση με τις υπάρχουσες στο ότι επικεντρώνεται στη Δυτική Μακεδονία, επιτρέπει ανάγνωση πληροφοριών χωρίς ο χρήστης να βρίσκεται στο σημείο, αφορά όλα τα είδη τόπων ενδιαφέροντος χωρίς να περιορίζεται σε κάποια συγκεκριμένη κατηγορία, αλλά δεν λειτουργεί χωρίς πρόσβαση στο διαδίκτυο και δεν παρέχει την δυνατότητα εισαγωγής στοιχείων.

1.5 Διάρθρωση κειμένου

Τα κεφάλαια αυτής της διπλωματικής οργανώνονται ως εξής.

Στο παρόν και πρώτο κεφάλαιο γίνεται μια εισαγωγή στο θέμα που πραγματεύεται η εργασία. Μια γενική περιγραφή, ο λόγος που επιλέχθηκε, παρόμοια συστήματα αλλά και τι αναλύουν τα επόμενα κεφάλαια.

Στο δεύτερο κεφάλαιο αναφέρεται το θεωρητικό υπόβαθρο. Γίνεται μια καταγραφή των εργαλείων και των τεχνικών που χρησιμοποιήθηκαν τόσο για την ιστοσελίδα όσο και για την κινητή εφαρμογή. αναλύει τις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της διπλωματικής.

Στο τρίτο κεφάλαιο γίνεται η ανάλυση και σχεδίαση του συστήματος πριν την υλοποίηση, αναφέρονται οι ρόλοι, οι απαιτήσεις και οι λειτουργίες που είναι απαραίτητες. Στη συνέχεια γίνεται ο σχεδιασμός της Βάσης Δεδομένων, αναφορά πιθανών περιπτώσεων χρήσης και σεναρίων, πώς επιτεύχθηκε η ασφάλεια. Επίσης αναφέρεται και ο σχεδιασμός της κινητής εφαρμογής και το πρωτόκολλο επικοινωνίας κινητού με την ιστοσελίδα.

Στο τέταρτο κεφάλαιο περιγράφονται οι λειτουργίες του ιστοχώρου. Επεξηγούνται διάφορες λειτουργίες της ιστοσελίδας με συνοδευτικές εικόνες, οι λειτουργίες του απλού χρήστη και του διαχειριστή και περιγράφονται μερικά σημαντικά κομμάτια κώδικα.

Στο πέμπτο κεφάλαιο αναφέρονται οι λειτουργίες της android κινητής εφαρμογής. Γίνεται επεξήγηση διαφόρων λειτουργιών με ανάλογες συνοδευτικές εικόνες, το πώς είναι δυνατή η μεταγλώττιση εφαρμογής, τα δικαιώματα χρήσης που απαιτεί, την κατανάλωση μνήμης και τα τεχνικά χαρακτηριστικά και αναφορά σε μερικά σημαντικά σημεία του κώδικα της εφαρμογής.

Στο έκτο κεφάλαιο είναι ο επίλογος. Κλείνει η παρούσα εργασία με κάποια συμπεράσματα κάποιες υποθέσεις για μελλοντικές επεκτάσεις και μια ανάλυση των δυνατών και αδύνατων σημείων, των ευκαιριών και των απειλών που ενέχονται, καθώς αναφέρονται και τα μετρικά στοιχεία του συστήματος.

Τέλος, υπάρχει ένα παράρτημα με οδηγίες για την εγκατάσταση του ιστοχώρου και της εφαρμογής και η βιβλιογραφία, όπου αναγράφονται οι πηγές που χρησιμοποιήθηκαν για τη συγγραφή της εργασίας.

1.6 Σύνοψη Κεφαλαίου

Σε αυτό το κεφάλαιο αναφέρονται τα κίνητρα που μας ώθησαν στην επιλογή ανάπτυξης αυτού του θέματος. Έγινε μία πρώτη παρουσίαση του συστήματος που αναπτύσσεται στη παρούσα διπλωματική εργασία, ως λύση σε υπάρχοντα προβλήματα, αναλύθηκαν τα πλεονεκτήματά του και έγινε αναφορά σε κάποιες υπάρχοντες εφαρμογές. Τέλος, παρουσιάστηκε συνοπτικά η δομή των κεφαλαίων που ακολουθούν.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

Στο παρόν κεφάλαιο γίνεται μία καταγραφή εργαλείων και τεχνικών που χρησιμοποιήθηκαν για την υλοποίηση αυτής της διπλωματικής εργασίας. Γίνονται γνωστές κάποιες έννοιες και αναλύονται σημαντικοί όροι ώστε να γίνει η συνέχεια κατανοητή από τον αναγνώστη.

2.1 Προγραμματισμός Διαδικτύου

Ο προγραμματισμός διαδικτύου είναι ένας γενικός όρος για οτιδήποτε σχετίζεται με την ανάπτυξη ιστοσελίδων στο Διαδίκτυο (World Wide Web). Η ανάπτυξη ιστού μπορεί να κυμαίνεται από την ανάπτυξη της απλούστερης στατικής ενιαίας σελίδας απλού κειμένου στις πιο σύνθετες ηλεκτρονικές εφαρμογές και στις υπηρεσίες κοινωνικών δικτύων. Περιλαμβάνει το σχεδιασμό ιστοσελίδων, την ανάπτυξη περιεχομένου ιστού, τη σύνδεση με πελάτες, το scripting από πλευράς πελάτη/διακομιστή, τη διαμόρφωση διακομιστή ιστού και δικτύου και την ανάπτυξη ηλεκτρονικού εμπορίου. Η "ανάπτυξη ιστού" αναφέρεται συνήθως στις κύριες πτυχές του σχεδιασμού ιστοσελίδων: γραφή και κωδικοποίηση. [29]

2.1.1 HTML

Η HTML είναι η βασική τεχνολογία που χρησιμοποιείται για οριστεί η δομή μιας ιστοσελίδας. Το ακρωνύμιο HTML αντιπροσωπεύει το Hyper Text Markup Language (Γλώσσα Σήμανσης Υπερκειμένου). Χρησιμοποιείται για να καθοριστεί αν το περιεχόμενό στο διαδίκτυο θα πρέπει να αναγνωρίζεται ως παράγραφος, λίστα, επικεφαλίδα, σύνδεσμος, εικόνα, πολυμέσα, φόρμα ή ένα από τα πολλά άλλα διαθέσιμα στοιχεία. [14] Στην εικόνα 2.1 μπορούμε να δούμε ένα παράδειγμα.

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

Σχήμα 2.1: Παράδειγμα HTML

2.1.2 CSS

Επίσης σημαντικό στοιχείο του προγραμματισμού διαδικτύου είναι το CSS (Cascading Stylesheets)(Επικαλυπτόμενα Φύλλα Στυλ). Είναι ο τρόπος που καθορίζεται το στυλ και η τοποθέτηση των στοιχείων της HTML και πως αυτά θα εμφανίζονται στην οθόνη. Γλυτώνει πολλή δουλειά γιατί μπορεί να ελέγξει τη διάταξη πολλών ιστοσελίδων ταυτόχρονα. Για παράδειγμα μπορείτε να χρησιμοποιήσετε το CSS για να αλλάξετε τη γραμματοσειρά, το χρώμα, το μέγεθος και την απόσταση του περιεχομένου σας, να το διαχωρίσετε σε πολλές στήλες ή να προσθέσετε κινούμενα σχέδια και άλλα διακοσμητικά χαρακτηριστικά. [17] Παράδειγμά του βλέπουμε στην εικόνα 2.2.

```
<style>
p {
  text-align: center;
  color: red;
}
</style>
```

Σχήμα 2.2: Παράδειγμα CSS

2.1.3 JavaScript

Η JavaScript είναι ο τρόπος για να προστεθούν δυναμικές λειτουργίες σε ιστοσελίδες. Για παράδειγμα μπορεί να βρει την τοποθεσία και να τη σχεδιάσει σε

χάρτη, να δημιουργήσει/αφαιρέσει στοιχεία του περιβάλλοντος χρήστη όταν αλλάζει ένα κουμπί, να αποθηκεύει τα δεδομένα των χρηστών τοπικά στους υπολογιστές τους και πολλά άλλα. Ουσιαστικά η JavaScript είναι μια γλώσσα προγραμματισμού που επιτρέπει να εφαρμοστούν πολύπλοκες δομές σε ιστοσελίδες. [16] Βλέπουμε παράδειγμα κώδικα στο σχήμα 2.3.

```
<script>
function myFunction() {
    document.getElementById("demo").innerHTML =
    "Paragraph changed.";
}
</script>
```

Σχήμα 2.3: Παράδειγμα JavaScript

2.1.4 jQuery

Το jQuery είναι μια γρήγορη, μικρή και πλούσια σε χαρακτηριστικά βιβλιοθήκη της JavaScript. Κάνει διαδικασίες όπως η μετακίνηση και ο χειρισμός εγγράφων HTML, η διαχείριση συμβάντων, τα κινούμενα σχέδια και την Ajax πολύ απλούστερα, με μία εύκολη στη χρήση Διεπαφή Προγραμματισμού Εφαρμογών(API) που λειτουργεί σε πλήθος προγραμμάτων περιήγησης. Με έναν πολύ καλό συνδυασμό ευελιξίας και επεκτασιμότητας, το jQuery έχει αλλάξει τον τρόπο με τον οποίο εκατομμύρια άνθρωποι γράφουν στην JavaScript. [30] Παράδειγμα στο σχήμα 2.4.

```
<script>
$(document).ready(function(){
    $("#myInput").on("keyup", function() {
        var value = $(this).val().toLowerCase();
        $("#myTable tr").filter(function() {
            $(this).toggle($(this).text().toLowerCase().indexOf(value) > -1)
        });
    });
});
</script>
```

Σχήμα 2.4: Παράδειγμα jQuery

2.1.5 PHP

Η PHP είναι μια ευρέως χρησιμοποιούμενη γλώσσα λογισμικού γενικής χρήσης ανοιχτού κώδικα που είναι ιδιαίτερα κατάλληλη για την ανάπτυξη ιστού και μπορεί να ενσωματωθεί στην HTML. Το αναδρομικό της ακρωνύμιο είναι Hypertext Preprocessor(Προεπεξεργαστής Υπερκειμένου). Οι σελίδες PHP περιέχουν HTML με ενσωματωμένο κώδικα που κάνει κάποια εργασία αντί για πολλές εντολές για την εξαγωγή HTML. Ο κώδικας της PHP ξεχωρίζει γιατί περικλείεται σε ειδικές οδηγίες έναρξης και λήξης `<? Php και ?>` τα οποία επιτρέπουν τη μετάβαση και την έξοδο από τη λειτουργία της PHP.

Αυτό που διακρίνει την PHP από την JavaScript είναι ότι ο κώδικας εκτελείται στον εξυπηρετητή, δημιουργώντας HTML που στη συνέχεια αποστέλλεται στον πελάτη(client). Ο πελάτης(client) θα λάβει τα αποτελέσματα της εκτέλεσης του σεναρίου, αλλά δεν θα ξέρει ποιος ήταν ο κώδικας. Παρόλο που η ανάπτυξη της PHP επικεντρώνεται στη δημιουργία σεναρίων από τη μεριά του διακομιστή(server), μπορούν να γίνουν πολλά περισσότερα με αυτήν. [31] Ένα παράδειγμα χρήσης της PHP στην εικόνα 2.5.

```
<?php  
echo "Hello World!";  
?>
```

Σχήμα 2.5: Παράδειγμα PHP

2.1.6 AJAX

Η AJAX σημαίνει Ασύγχρονη JavaScript και XML. Με λίγα λόγια, είναι η χρήση του αντικειμένου XMLHttpRequest για την επικοινωνία με τους διακομιστές. Μπορεί να στέλνει και να λαμβάνει πληροφορίες σε διάφορες μορφές, όπως JSON, XML, HTML και αρχεία κειμένου. Το πιο ελκυστικό χαρακτηριστικό της AJAX είναι ο "ασύγχρονος" χαρακτήρας της, ο οποίος σημαίνει ότι μπορεί να επικοινωνεί με τον διακομιστή, να ανταλλάσσει δεδομένα αλλά και να ενημερώνει τη σελίδα χωρίς να χρειάζεται να την ανανεώνει. Η AJAX σας επιτρέπει κυρίως να πραγματοποιήσετε αιτήσεις στον διακομιστή χωρίς να φορτώσετε ξανά τη σελίδα και να κάνετε λήψη

και επεξεργασία δεδομένων από το διακομιστή. [8] Στο σχήμα 2.6 μπορούμε να δούμε ένα κομμάτι κώδικα.

```
<script>
function loadDoc() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status ==
200) {
            document.getElementById("demo").innerHTML =
                this.responseText;
        }
    };
    xhttp.open("GET", "ajax_info.txt", true);
    xhttp.send();
}
</script>
```

Σχήμα 2.6: Παράδειγμα AJAX

2.1.7 MySQL

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακής βάσης δεδομένων ανοιχτού κώδικα. Το όνομά της είναι ένας συνδυασμός του "My", το όνομα της κόρης του ενός εκ των ιδρυτών, και της "SQL", συντομογραφία της Structured Query Language (Δομημένης Γλώσσας Ερωτημάτων). [13] Χρησιμοποιείται η γλώσσα SQL που είναι η κύρια γλώσσα ερωτημάτων για βάσεις δεδομένων.

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,
    ....
);
```

Σχήμα 2.7: Παράδειγμα MySQL

Οι βάσεις δεδομένων δίνουν τη δυνατότητα τα δεδομένα ενός ιστότοπου να αποθηκεύονται, να αναζητούνται, να ταξινομούνται και να ανακαλούνται όπως και

να τα διαχειρίζονται πολλοί επαληθευμένοι χρήστες ταυτόχρονα. Στο σχήμα 2.7 βλέπουμε πως δημιουργείται ένας πίνακας με την MySQL.

2.1.8 Google Maps API

Ένα API (Διεπαφή Προγραμματισμού Εφαρμογών) είναι ένα σύνολο μεθόδων και εργαλείων που μπορούν να χρησιμοποιηθούν για την κατασκευή εφαρμογών λογισμικού. Το API των Χαρτών της Google επιτρέπει να προβληθούν χάρτες σε ιστότοπους και εφαρμογές. [11] Μπορούμε να δούμε παράδειγμα στην εικόνα 2.8.

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Google Map</h1>

<div id="googleMap" style="width:100%;
height:400px;"></div>

<script>
function myMap() {
var mapProp= {
  center:new google.maps.LatLng(51.508742,-
0.120850),
  zoom:5,
};
var map=new
google.maps.Map(document.getElementById("googleMap
"),mapProp);
}
</script>

<script src="https://maps.googleapis.com/maps/api
/js?key=AizaSyBu-916DdpKajTmJNIngS6HL_kDIKU0aU&
callback=myMap"></script>
</body>
</html>
```

My First Google Map



Σχήμα 2.8: Παράδειγμα Google Maps API

2.1.9 Bootstrap

Η Bootstrap είναι ένα ελεύθερο και ανοικτού κώδικα περιβάλλον εργασίας δικτύου για το σχεδιασμό ιστοσελίδων και εφαρμογών ιστού. Περιέχει διάφορα πρότυπα σχεδίασης HTML και CSS για τυπογραφία, φόρμες, κουμπιά, πλοήγηση και άλλα στοιχεία διεπαφής, καθώς και προαιρετικές επεκτάσεις JavaScript. [5] Παρακάτω στο σχήμα 2.9 βλέπουμε ένα παράδειγμα της.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container">
  <h1>My First Bootstrap Page</h1>
  <p>This part is inside a .container class.</p>
  <p>The .container class provides a responsive fixed width container.</p>
</div>

</body>
</html>
```

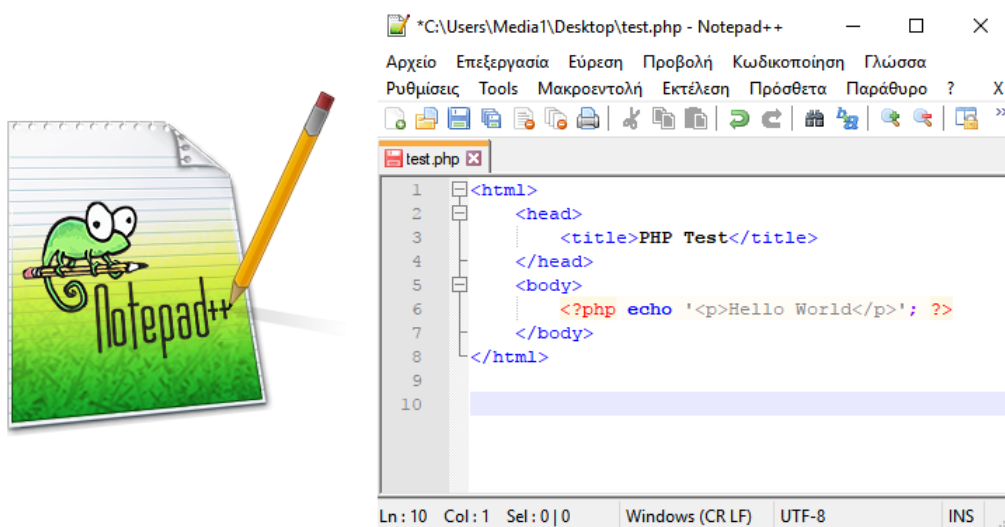
Σχήμα 2.9: Παράδειγμα Bootstrap

2.2 Εργαλεία Λογισμικού

Εκτός από τον προγραμματισμό διαδικτύου, υπήρξαν και αρκετά εργαλεία λογισμικού τα οποία βοήθησαν να υλοποιηθεί αυτή η διπλωματική.

2.2.1 Notepad++

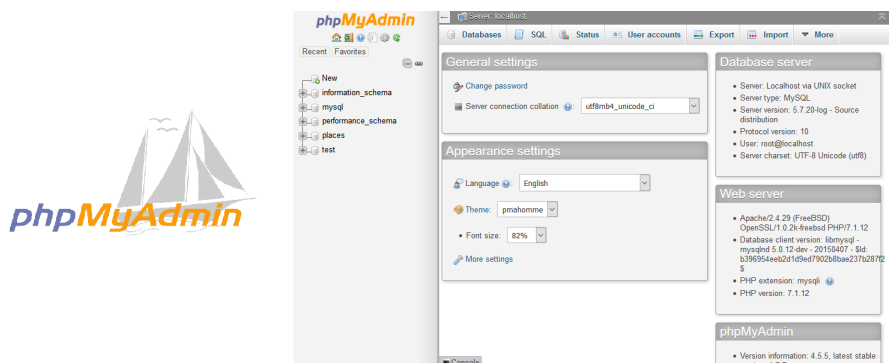
Το Notepad++ είναι ένας δωρεάν επεξεργαστής πηγαίου κώδικα, και αντικαταστάτης του Notepad, που υποστηρίζει πολλές γλώσσες. Λειτουργεί σε περιβάλλον MS Windows και η χρήση του διέπεται από την άδεια GPL(Γενική Δημόσια Άδεια)[1]. Χρησιμοποιήθηκε στη συγγραφή του κώδικα της ιστοσελίδας. Στο σχήμα 2.10 βλέπουμε το εικονίδιο και την βασική του οθόνη.



Σχήμα 2.10: Notepad++

2.2.2 phpMyAdmin

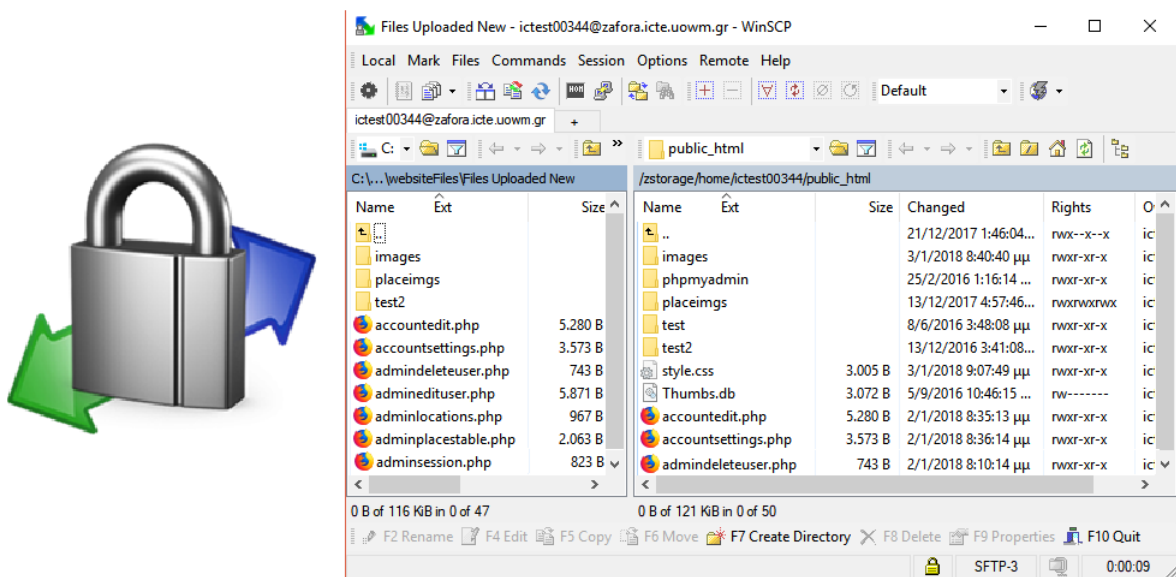
Το phpMyAdmin είναι ένα εργαλείο ελεύθερου λογισμικού γραμμένο σε PHP, το οποίο προορίζεται για τη διαχείριση της MySQL μέσω του Διαδικτύου. Υποστηρίζει ένα ευρύ φάσμα λειτουργιών στις υπηρεσίες MySQL και MariaDB. Συχνά χρησιμοποιούμενες λειτουργίες (διαχείριση βάσεων δεδομένων, πίνακες, στήλες, σχέσεις, ευρετήρια, χρήστες, δικαιώματα κ.λπ.) μπορούν να εκτελεστούν μέσω του περιβάλλοντος χρήστη, ενώ υπάρχει ακόμα η δυνατότητα να εκτελέσετε άμεσα οποιαδήποτε δήλωση SQL[2]. Έκανε ευκολότερο και απλούστερο το χειρισμό της Βάσης μας και των πινάκων της. Μπορούμε να δούμε το εικονίδιο του και μία απεικόνιση της σελίδας στο σχήμα 2.11.



Σχήμα 2.11: phpMyAdmin

2.2.3 WinSCP

Το WinSCP είναι ένα δωρεάν ανοιχτού κώδικα πρόγραμμα πελάτης (client) SFTP (SSH File Transfer Protocol - Πρωτόκολλο Μεταφοράς Αρχείων), πρόγραμμα-πελάτης FTP, πελάτης WebDAV (Web Distributed Authoring and Versioning - Κατανεμημένη Διαδικτυακή Συγγραφή και Έκδοση) και πρόγραμμα-πελάτης SCP (Secure Copy Protocol - Πρωτόκολλο Ασφαλούς Αντιγραφής) για Windows. Η κύρια λειτουργία του είναι η μεταφορά αρχείων μεταξύ τοπικού και απομακρυσμένου υπολογιστή. Εκτός αυτού, το WinSCP προσφέρει δέσμες ενεργειών και βασική λειτουργία διαχείρισης αρχείων[15]. Έγινε χρήση του για πρόσβαση στο διακομιστή και ανανέωση των αρχείων με άλλα με βελτιωμένο κώδικα. Βλέπουμε το εικονίδιο της εφαρμογής και μία απεικόνισή της στο σχήμα 2.12.



Σχήμα 2.12: WinSCP

2.2.4 PuTTY

Το PuTTY είναι ένας πελάτης(client) SSH(Secure Shell) και Telnet, ο οποίος αναπτύχθηκε αρχικά για την πλατφόρμα Windows. Το PuTTY είναι λογισμικό ανοιχτού κώδικα που διατίθεται με πηγαίο κώδικα και αναπτύσσεται και υποστηρίζεται από μια ομάδα εθελοντών[24]. Μέσω του PuTTY έγινε δυνατό να δημιουργηθεί και να ενεργοποιηθεί τόσο ο ιστόχωρος, όσο και η προσωπική βάση δεδομένων. Στην εικόνα 2.13 βλέπουμε το εικονίδιο και μία οθόνη του.



```
zefora.ict.e.uowm.gr - PuTTY
login as: ictest00344
-----
      http://zefora.ict.e.uowm.gr
      ( 83.212.19.218 )
      ( 2001:648:3820:5200:20e:cff:feb6:8e6d )

      Department of Informatics
      & Telecommunications Engineering
      ( http://www.ict.e.uowm.gr )

      University of Western Macedonia
      ( http://www.uowm.gr )

      Unauthorized Access is Prohibited.
      For security purposes all actions are logged.
-----

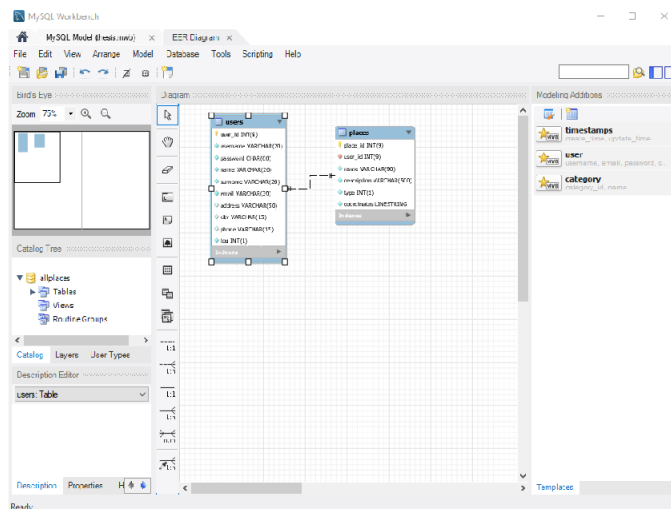
Using keyboard-interactive authentication.
Password for ictest00344@zefora.ict.e.uowm.gr:
Last login: Tue Dec 12 16:45:29 2017 from starbucks-ac-08.forthnet.gr
FreeBSD 11.1-RELEASE-p4 (GENERIC) #0: Tue Nov 14 06:12:40 UTC 2017

Welcome to FreeBSD!
Storage Space: You are using [ 209M ] of the maximum allowed [ 1.17G ]
Processes: You are utilizing [ 8 ] number of processes of the maximum allowed [
96 ]
Logins: You are currently using [ 0 ] number of virtual terminals
[ictest00344@zafora ~]$
```

Σχήμα 2.13: PuTTY

2.2.5 MySQL Workbench

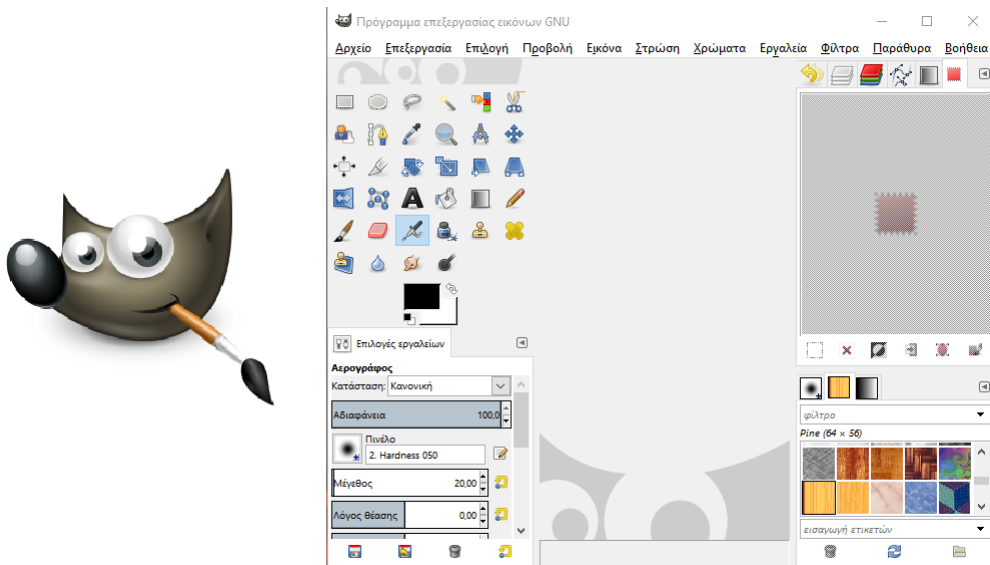
Το MySQL Workbench είναι ένα ενιαίο οπτικό εργαλείο για αρχιτέκτονες βάσεων δεδομένων, προγραμματιστές και DBAs. Το MySQL Workbench παρέχει μοντελοποίηση δεδομένων, ανάπτυξη SQL και ολοκληρωμένα εργαλεία διαχείρισης για τη διαμόρφωση του διακομιστή, τη διαχείριση χρηστών, τα αντίγραφα ασφαλείας και άλλα. [20] Στο σχήμα 2.14 βλέπουμε την επιφάνεια εργασίας του και το εικονίδιο του.



Σχήμα 2.14: MySQL Workbench

2.2.6 GIMP

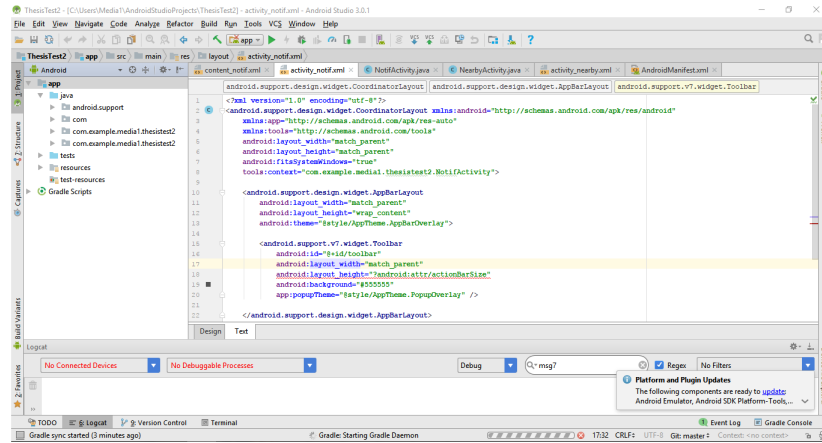
Το GIMP (GNU Image Manipulation Program - Πρόγραμμα χειρισμού εικόνων GNU) είναι ένας ελεύθερος και ανοιχτού κώδικα επεξεργαστής γραφικών που χρησιμοποιείται για ρετουσάρισμα και επεξεργασία εικόνων, για σχέδιο ελεύθερης μορφής, μετατροπή μεταξύ διαφορετικών μορφών εικόνας και ακόμα πιο εξειδικευμένες εργασίες. [9] Μπορούμε να δούμε στο σχήμα 2.15, το εικονίδιο και τον χώρο εργασίας που προσφέρει.



Σχήμα 2.15: GIMP

2.2.7 Android Studio

Το Android Studio είναι το επίσημο ολοκληρωμένο αναπτυξιακό περιβάλλον (IDE) για την ανάπτυξη εφαρμογών Android και είναι βασισμένο στο IntelliJ IDEA. Εκτός των εργαλείων επεξεργασίας κώδικα και των εργαλείων ανάπτυξης του IntelliJ, το Android Studio προσφέρει ακόμα περισσότερες λειτουργίες που βοηθούν στην κατασκευή εφαρμογών Android. [19] Στην εικόνα 2.16 βλέπουμε το εικονίδιο και μία απεικόνιση της οθόνης του.



Σχήμα 2.16: Android Studio

2.3 Σύνοψη Κεφαλαίου

Σε αυτό το κεφάλαιο παρουσιάστηκε το θεωρητικό υπόβαθρο της εργασίας, αναφέρθηκαν οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος και αναλύθηκαν οι γλώσσες προγραμματισμού. Επίσης περιγράφηκαν και τα εργαλεία λογισμικού που χρησιμοποιήθηκαν για την ανάπτυξη του πληροφοριακού συστήματος.

Κεφάλαιο 3

Ανάλυση και Σχεδίαση

Σε αυτό το κεφάλαιο, γίνεται η ανάλυση του συστήματος που έγινε πριν το στάδιο της υλοποίησης. Συγκεκριμένα, γίνεται γνωστή η αρχιτεκτονική του συστήματος που θα υλοποιηθεί και αναδεικνύεται πως συνδέονται οι τεχνολογίες που αναφέρθηκαν στο προηγούμενο κεφάλαιο ώστε να δημιουργηθεί ο συγκεκριμένος ιστοχώρος.

3.1 Ανάλυση και Σχεδίαση Ιστοχώρου

Αρχικά ξεκινάμε με την ανάλυση και σχεδίαση του ιστότοπου. Πρέπει να καθοριστούν οι ρόλοι, οι απαιτήσεις όπως και οι περιπτώσεις χρήσης του συστήματος. Βασικό κομμάτι είναι η σχεδίαση της βάσης δεδομένων και τι πρέπει να γίνει ώστε να επιτευχθεί η ασφάλεια του συστήματος.

3.2 Ρόλοι

Σημαντικό ρόλο στο να γίνει σωστά μια ανάλυση ενός συστήματος είναι αρχικά να καθοριστούν ποιοι θα είναι οι ρόλοι οι οποίοι θα αλληλεπιδρούν με αυτό. Στην δική μας περίπτωση είναι αυτοί του διαχειριστή, του εγγεγραμμένου χρήστη και του ανώνυμου χρήστη.

- **Διαχειριστής:** Ο διαχειριστής είναι ο χρήστης, ο οποίος έχει πρόσβαση σε επιπρόσθετες λειτουργίες και έχει περισσότερα δικαιώματα.
- **Εγγεγραμμένος Χρήστης:** Ο εγγεγραμμένος χρήστης μπορεί μέσω του γραφικού περιβάλλοντος, να επιδρά στο σύστημα, να αντλεί και να εναποθέτει πληροφορίες σε αυτό.

- **Ανώνυμος Χρήστης:** Ο ανώνυμος χρήστης είναι ο χρήστης που δεν έχει εγγραφεί στο σύστημα, και η μόνη αλληλεπίδραση που μπορεί να έχει είναι η πραγματοποίηση της εγγραφής του.

3.3 Απαιτήσεις

Με βάση την περιγραφή του συστήματος και σύμφωνα με τις ανάγκες των χρηστών καθορίζονται και οι απαιτήσεις του, δηλαδή οι προδιαγραφές που πρέπει να τηρούνται ώστε το σύστημα να είναι λειτουργικό και διαχειρίσιμο. Οι ανάγκες και οι προδιαγραφές του λογισμικού παρουσιάζονται στη συνέχεια.

Το σύστημα θα πρέπει να υποστηρίζει κατηγορίες χρηστών. Οι ρόλοι των διαχειριστών και των απλών χρηστών είναι ξεχωριστοί και αλληλεπιδρούν με τις διεργασίες που ανήκουν στο αντίστοιχο επίπεδο ιεραρχίας. Τα δικαιώματα αυτών των κατηγοριών χρηστών διαφέρουν, καθώς μόνο οι διαχειριστές είναι σε θέση να έχουν πρόσβαση σε ό,τι παρέχει το σύστημα.

Η πρόσβαση στον ιστότοπο θα είναι δυνατή μόνο κατόπιν εγγραφής. Οι χρήστες θα πρέπει να εγγραφούν στο σύστημα και να καταχωρίσουν τα στοιχεία τους μέσω της φόρμας για να κάνουν λογαριασμό. Στην συνέχεια θα έχουν τη δυνατότητα να συνδεθούν στο σύστημα. Αφού εγγραφεί κάποιος απλός χρήστης θα μπορεί να επεξεργάζεται τα στοιχεία του, να προσθέτει και να διαχειρίζεται τις δικές του περιοχές και περιηγήσεις. Ένας διαχειριστής θα μπορεί να διαχειρίζεται χρήστες, να κάνει κάποιο χρήστη διαχειριστή, να προσθέτει και να τροποποιεί τις δικές του περιοχές και περιηγήσεις αλλά και να έχει πρόσβαση σε περιοχές και περιηγήσεις οποιουδήποτε χρήστη του συστήματος.

Το περιβάλλον θα πρέπει να είναι εύχρηστο ώστε σε περίπτωση που κάποιος χρήστης δεν έχει μεγάλη εξοικείωση με την τεχνολογία να του προσφέρεται μια φιλική εμπειρία περιήγησης στον ιστότοπο.

3.4 Περιπτώσεις Χρήσης

Στην συνέχεια αναλύονται τα βασικότερα σενάρια χρήσης του συστήματος.

- **Δημιουργία Λογαριασμού.** Αρχικά και για να αποκτήσει κάποιος πρόσβαση στο σύστημα είναι η "Δημιουργία Λογαριασμού". Με αυτή την ενέργεια ένας

ανώνυμος χρήστης μπορεί να καταχωρήσει τα στοιχεία του και να εγγραφεί στο σύστημα.

- **Είσοδος στο σύστημα.** Ο χρήστης, χρησιμοποιώντας τα στοιχεία του λογαριασμού του συνδέεται στο σύστημα.
- **Προβολή περιοχών.** Ο χρήστης επιλέγει από το μενού τους Χάρτες(Maps) και εκεί έχει τη δυνατότητα να δει τις αποθηκευμένες του περιοχές.
- **Προσθήκη Περιοχής.** Ο χρήστης πατώντας μία από τις επιλογές σχεδιασμού που υπάρχουν στο χάρτη, επιλέγει την περιοχή που θέλει, δίνει τα στοιχεία της και την αποθηκεύει.
- **Επεξεργασία Περιοχής.** Από τον πίνακα αριστερά ο χρήστης μπορεί να επιλέξει να επεξεργαστεί κάποια από τις αποθηκευμένες περιοχές.
- **Διαγραφή Περιοχής.** Από τον πίνακα αριστερά ο χρήστης μπορεί να επιλέξει να διαγράψει κάποια από τις αποθηκευμένες περιοχές του.
- **Προβολή Περιηγήσεων.** Ο χρήστης επιλέγει από το μενού τις Περιηγήσεις (Tours) και εκεί έχει τη δυνατότητα να δει τις αποθηκευμένες του περιηγήσεις.
- **Προσθήκη Περιήγησης.** Πατώντας το κουμπί Προσθήκη Περιήγησης (Add New Tour) ο χρήστης μπορεί να προσθέσει μία περιήγηση δίνοντας τα στοιχεία της και επιλέγοντας από τα αποθηκευμένα του μέρη ποια θα περιλαμβάνονται.
- **Προβολή Περιήγησης.** Από τον πίνακα με τις αποθηκευμένες Περιηγήσεις ο χρήστης μπορεί να κάνει προβολή κάποιας περιήγησης του.
- **Επεξεργασία Περιήγησης.** Από τον πίνακα με τις αποθηκευμένες Περιηγήσεις ο χρήστης μπορεί να κάνει επεξεργασία κάποιας περιήγησης του.
- **Διαγραφή Περιήγησης.** Από τον πίνακα με τις αποθηκευμένες Περιηγήσεις ο χρήστης μπορεί να κάνει διαγραφή κάποιας περιήγησης του.
- **Επεξεργασία Στοιχείων Χρήστη.** Ο χρήστης έχει τη δυνατότητα να επεξεργαστεί τα στοιχεία του και να αλλάξει τον κωδικό του.

3.5 Σχεδιασμός Βάσης Δεδομένων

Σύμφωνα λοιπόν με τις απαιτήσεις του συστήματος, σχεδιάστηκε ανάλογα και η βάση δεδομένων. Ο σχεδιασμός της βάσης δεδομένων είναι πολύ δύσκολη αλλά σημαντική εργασία ώστε να επιτευχθεί η σωστή και χωρίς προβλήματα ανάπτυξη του συστήματος. Η βάση δεδομένων παίζει βασικό ρόλο στην ανάπτυξη του κώδικα αφού είναι το μέρος για την αποθήκευση και την οργάνωση των δεδομένων και των πληροφοριών, με τον καλύτερο δυνατό τρόπο, που είναι αναγκαία για τη λειτουργία του συστήματος. Αποτελείται από τρεις πίνακες οι οποίοι περιγράφονται αναλυτικά στη συνέχεια, μαζί με το είδος των δεδομένων που αποθηκεύουν και τα πεδία από τα οποία αποτελούνται.

3.5.1 Πίνακας users

Στον πίνακα users αποθηκεύονται τα προσωπικά στοιχεία των χρηστών και τα στοιχεία με τα οποία συνδέονται στο σύστημα. Στον πίνακα 3.1 βλέπουμε τη δομή του.

Όνομα	Τύπος	Κενό	Προεπιλογή	Πρόσθετα
user_id (PK)	int(9)	No	None	AUTO_INCREMENT
username	varchar(20)	No	None	
password	char(60)	No	None	
name	varchar(20)	No	None	
surname	varchar(20)	No	None	
email	varchar(30)	No	None	
address	varchar(50)	Yes	NULL	
city	varchar(15)	Yes	NULL	
phone	varchar(15)	Yes	NULL	
tou	int(1)	No	None	

Πίνακας 3.1: Πίνακας Χρηστών (users)

- **user_id:** Το id του χρήστη είναι το πρωτεύον κλειδί(Primary Key - PK) του πίνακα γιατί χαρακτηρίζει μοναδικά τον κάθε χρήστη. Είναι ακέραιος με μέγιστο μήκος 9 ψηφία και δεν μπορεί να είναι κενό(NULL). Αφού σίγουρα θα πάρει τιμή δεν έχει κάποια προεπιλεγμένη τιμή και αυξάνεται αυτόματα με την εγγραφή κάθε καινούριου χρήστη.
- **username:** Είναι το όνομα χρήστη που χρησιμοποιείται για την πραγματοποίηση της εισόδου στο σύστημα. Αποτελείται από χαρακτήρες και το μέγιστο μήκος του είναι 20. Δεν μπορεί να είναι κενό, γιατί χωρίς αυτό δεν είναι δυνατή η πρόσβαση στο σύστημα και δεν έχει κάποια προεπιλεγμένη τιμή.
- **password:** Είναι ο κωδικός πρόσβασης που χρησιμοποιεί ο χρήστης για να συνδεθεί στο σύστημα. Ο κωδικός δεν αποθηκεύεται στη βάση δεδομένων όπως είναι, για λόγους ασφαλείας, αλλά σε μια κρυπτογραφημένη μορφή του μέσω του bcrypt αλγόριθμου. Δεν μπορεί να πάρει κενό για τιμή γιατί είναι απαραίτητος για τη σύνδεση του χρήστη, έτσι και η τιμή του δεν είναι προκαθορισμένη.
- **name:** Είναι το μικρό όνομα του χρήστη και αποτελείται από χαρακτήρες με το μέγιστο μήκος του να είναι 20. Δεν μπορεί να είναι κενό, γιατί θέλουμε να γνωρίζουμε κάποια βασικά στοιχεία των χρηστών που εγγράφονται. Επίσης δεν έχει κάποια προεπιλεγμένη τιμή.
- **surname:** Είναι το επώνυμο του χρήστη και αποτελείται μέχρι και από 20 χαρακτήρες. Όπως και το όνομα δεν μπορεί να είναι κενό και δεν έχει κάποια προεπιλεγμένη τιμή.
- **email:** Είναι η διεύθυνση ηλεκτρονικού ταχυδρομείου του χρήστη. Αποτελείται από χαρακτήρες και έχει μέγιστο μήκος 30. Δεν μπορεί να είναι κενό γιατί είναι ο βασικότερος τρόπος επικοινωνίας με τον χρήστη, κατά συνέπεια δεν έχει και κάποια προεπιλεγμένη τιμή.
- **address:** Είναι η διεύθυνση του χρήστη και μπορεί να έχει μέχρι και 50 χαρακτήρες. Η δήλωσή της δεν είναι υποχρεωτική έτσι μπορεί να είναι και κενή τιμή, η οποία της δίνεται και ως προεπιλογή.

- **city:** Είναι η πόλη διαμονής του χρήστη και μπορεί να αποτελείται από 15 χαρακτήρες. Δεν είναι υποχρεωτικό να δοθεί, άρα μπορεί να έχει και κενή τιμή. Η τιμή της είναι προκαθορισμένη ως κενό.
- **phone:** Το τηλέφωνο του χρήστη δεν είναι απαραίτητη πληροφορία αφού έχει ήδη δοθεί το email. Μπορεί να έχει μέχρι και 15 χαρακτήρες, για να καλυφθεί και η περίπτωση τηλεφωνικού αριθμού εκτός Ελλάδος, και από προεπιλογή είναι κενό.
- **you:** Είναι το είδος του χρήστη(type of user). Παίρνει ακέραια τιμή ενός ψηφίου η οποία δείχνει αν ο χρήστης είναι διαχειριστής(1) ή απλός(2). Δεν μπορεί να είναι κενό αφού όλοι οι χρήστες πρέπει να ανήκουν σε μία από τις δύο κατηγορίες και δεν έχει κάποια προκαθορισμένη τιμή.

3.5.2 Πίνακας allplaces

Στον πίνακα allplaces αποθηκεύονται οι περιοχές που αποθηκεύουν οι χρήστες και τα στοιχεία που εισάγουν για αυτές. Μπορούμε να δούμε τη δομή του στον πίνακα 3.2.

Όνομα	Τύπος	Κενό	Προεπιλογή	Πρόσθετα
place_id (PK)	int(9)	No	None	AUTO_INCREMENT
user_id (FK)	int(9)	No	None	
name	varchar(90)	No	None	
description	varchar(500)	No	None	
type	int(1)	No	None	
coordinates	varchar(10000)	No	None	
pfimage	varchar(500)	Yes	NULL	

Πίνακας 3.2: Πίνακας Περιοχών (allplaces)

- **place_id:** Το id της περιοχής είναι το πρωτεύον κλειδί(Primary Key - PK) του πίνακα allplaces γιατί χαρακτηρίζει μοναδικά την κάθε περιοχή. Ακέραιος με

μέγιστο μήκος τα 9 ψηφία, δεν μπορεί να είναι κενό(NULL). Δεν έχει κάποια προεπιλεγμένη τιμή και αυξάνεται αυτόματα με την προσθήκη κάθε νέας περιοχής.

- **user_id:** Το id του χρήστη είναι ξένο κλειδί του πίνακα που δημιουργεί την σύνδεση με τον Πίνακα 3.1 users ώστε να φαίνεται ποιος χρήστης δημιούργησε την περιοχή.
- **name:** Είναι το όνομα της περιοχής αποτελείται από χαρακτήρες με το μέγιστο μήκος του να είναι 90. Δεν μπορεί να είναι κενό, μιας και είναι απαραίτητο για την αποθήκευση της περιοχής και η τιμή του δεν είναι προκαθορισμένη.
- **description:** Είναι η περιγραφή της περιοχής και μπορεί να έχει το περισσότερο 500 χαρακτήρες. Δεν μπορεί να πάρει ως τιμή το κενό και δεν έχει κάποια προεπιλεγμένη τιμή.
- **type:** Είναι ο τύπος της αποθηκευμένης περιοχής, μπορεί να είναι τετράγωνο(2) ή κύκλος(3). Είναι ακέραιος αριθμός με μόνο ένα ψηφίο, δεν μπορεί να είναι κενό και δεν έχει προκαθορισμένη τιμή.
- **coordinates:** Είναι οι συντεταγμένες της περιοχής. Είναι χαρακτήρες με μέγιστο τους 10000, δεν μπορεί να είναι κενό και δεν έχει κάποια τιμή εξ' αρχής.
- **pfimage:** Είναι η εικόνα που μπορεί να έχει προστεθεί για την περιοχή. Είναι μέχρι 500 χαρακτήρες, μπορεί να είναι κενό το οποίο είναι και η προκαθορισμένη του τιμή.

3.5.3 Πίνακας tours

Στον πίνακα tours αποθηκεύονται οι περιηγήσεις που έχουν δημιουργήσει οι χρήστες και τα στοιχεία που εισάγουν για αυτές. Η δομή του φαίνεται στον πίνακα 3.3.

- **tour_id:** Το id της περιήγησης είναι το πρωτεύον κλειδί(Primar Key - PK) του πίνακα tours. Χαρακτηρίζει μοναδικά την κάθε περιήγηση και είναι ακέραιος με μέχρι 9 ψηφία. Δεν μπορεί να είναι κενό(NULL), δεν έχει κάποια προεπιλεγμένη τιμή και με την εισαγωγή νέας περιήγησης αυξάνεται αυτόματα.

Όνομα	Τύπος	Κενό	Προεπιλογή	Πρόσθετα
tour_id (PK)	int(9)	No	None	AUTO_INCREMENT
user_id (FK)	int(9)	No	None	
tour_name	varchar(90)	No	None	
tour_places	varchar(500)	No	None	
tour_desc	varchar(500)	Yes	NULL	

Πίνακας 3.3: Πίνακας Περιηγήσεων (tours)

- **user_id:** Είναι ξένο κλειδί του πίνακα που δημιουργεί σύνδεση με τον Πίνακα 3.1 users ώστε να φαίνεται ποιος χρήστης δημιούργησε την κάθε περιήγηση.
- **tour_name:** Είναι το όνομα της περιήγησης και μπορεί να έχει μέχρι και 90 χαρακτήρες. Δεν μπορεί να είναι κενό και η τιμή του δεν είναι προκαθορισμένη.
- **tour_places:** Είναι τα μέρη που περιέχει η περιήγηση. Αποτελείται από χαρακτήρες και ο μέγιστος αριθμός τους μπορεί να είναι 500. Δεν μπορεί να είναι κενό, χωρίς περιοχές δεν έχει νόημα να υπάρχει περιήγηση και δεν έχει προεπιλεγμένη τιμή.
- **tour_desc:** Είναι η περιγραφή της περιήγησης και μπορεί να αποτελείται από το περισσότερο 500 χαρακτήρες. Μπορεί να πάρει ως τιμή το κενό το οποίο είναι και καθορισμένο ως προεπιλογή του.

3.5.4 Σχεσιακό Διάγραμμα Βάσης Δεδομένων

Στο διάγραμμα 3.1 που ακολουθεί μπορούμε να δούμε μια συνολική εικόνα της βάσης δεδομένων. Το διάγραμμα οντοτήτων-συσχετίσεων της βάσης δεδομένων αποτυπώνει μια ολοκληρωμένη αναπαράσταση της δομής της βάσης με τους πίνακες, τα πεδία, τους τύπους δεδομένων αλλά και τις εξαρτήσεις μεταξύ τους.



Σχήμα 3.1: Σχεσιακό Διάγραμμα Βάσης Δεδομένων

3.6 Ασφάλεια

Στη βάση δεδομένων αποθηκεύονται δεδομένα τα οποία είναι ζωτικής σημασίας και απόρρητα, έτσι δεν θα πρέπει να υπάρχει η δυνατότητα για μη εξουσιοδοτημένη πρόσβαση σε αυτά. Η ασφάλεια, λοιπόν, παίζει σημαντικό ρόλο και είναι ένα από τα σημαντικότερα κομμάτια στον σχεδιασμό, αλλά και της υλοποίησης ενός συστήματος. Για να είναι ασφαλές το σύστημα που χρησιμοποιούμε σε αυτή την εργασία πάρθηκαν τα παρακάτω μέτρα.

- **Έλεγχος Σύνδεσης.** Δεν μπορεί να υπάρξει καμία αλληλεπίδρασή με το σύστημα, εκτός από την Εγγραφή και την Υπενθύμιση Κωδικού, αν ο χρήστης δεν είναι συνδεδεμένος. Όταν υπάρχει επικοινωνία με το διακομιστή γίνεται έλεγχος αν ο χρήστης είναι συνδεδεμένος, ώστε να δοθεί η άδεια να έχει πρόσβαση στις σελίδες του ιστοχώρου. Επίσης γίνεται και έλεγχος σύνδεσης όταν κάποιος χρήστης προσπαθεί να συνδεθεί σε σελίδες διαχειριστή, αν δεν έχει τα κατάλληλα δικαιώματα ανακατευθύνεται.
- **Χρήση Κωδικού.** Η σύνδεση του χρήστη γίνεται με κωδικό ο οποίος προστατεύεται με διάφορους τρόπους. Αρχικά αποθηκεύεται σε κατακερματισμένη(hash) μορφή και όχι στη μορφή που τον έχει εισάγει ο χρήστης στη φόρμα επικοινωνίας. Η κατακερμάτηση γίνεται με τη βοήθεια του αλγορίθμου bcrypt. Ο bcrypt είναι ένας αλγόριθμος που κάνει τις επιθέσεις brute force πιο αργές και ελαχιστοποιεί το αντίκτυπο τους. Βασίζεται στον αλγόριθμο κρυπτογράφησης Blowfish που μέχρι στιγμής δεν έχει καταφέρει κάποιος να τον αποκρυπτογραφήσει [27]. Αυτός είναι και ο λόγος που επιλέχθηκε για την παρούσα εργασία. Επίσης, κατά τη σύνδεση στο σύστημα γίνεται επιβεβαίωση του κωδικού που δίνει ο χρήστης αν ταιριάζει με το αποθηκευμένο hash. Έπειτα, κατά τη δημιουργία ή την αλλαγή του κωδικού πρόσβασης, ο χρήστης πρέπει να τον εισάγει δύο φορές. Αυτό γίνεται για την επαλήθευση του κωδικού, και γίνεται αποδεκτός μόνο όταν είναι ίδιοι. Τέλος, ο κωδικός δεν μπορεί να είναι μικρότερος από 6 χαρακτήρες, έτσι προσθέτεται ένα επιπλέον επίπεδο ασφάλειας.
- **Φιλτράρισμα Δεδομένων.** Τα δεδομένα που εισάγει ο χρήστης, φιλτράρονται και καθαρίζονται από ανεπιθύμητους χαρακτήρες πριν τοποθετηθούν στη βάση δεδομένων. Επίσης, επικυρώνονται οι τύποι των μεταβλητών που αποθηκεύονται τα πεδία της κάθε φόρμας της ιστοσελίδας, το οποίο θεωρείται απαραίτητο για την αποφυγή εισαγωγής κακόβουλων δεδομένων και την διατήρησή της ασφάλειας του συστήματος.
- **Σύνδεση στη Βάση.** Για την σύνδεση και εκτέλεση ερωτημάτων στη βάση δεδομένων χρησιμοποιήθηκαν τα Αντικείμενα Δεδομένων PHP (PHP Data Objects, PDO). Η χρήση PDO είναι μια τεχνική ασφάλειας για τη βάση δεδομένων και

αποτελεί την πιο αποτελεσματική μέθοδο για την αποτροπή επιθέσεων SQL injection. Οι επιθέσεις SQL injection γίνονται από κακόβουλους χρήστες οι οποίοι εισάγουν εντολές SQL μέσω των πεδίων εισόδου μιας ιστοσελίδας με σκοπό να αλλάξουν τα χαρακτηριστικά και τις ρυθμίσεις της βάσης δεδομένων και να θέσουν σε κίνδυνο την ασφάλεια όλου του συστήματος. Αυτό που κάνει το PDO είναι ότι διαχωρίζει το ερώτημα από τα δεδομένα, τα οποία αντικαθιστώνται με σύμβολα, οδηγώντας έτσι στο να μην μπορούν να αποθηκευτούν επιπλέον ανεπιθύμητες πληροφορίες

- **Σύνδεση HTTPS.** Εξασφαλίζεται η πρόσβαση στις ιστοσελίδες του ιστότοπου μέσω κρυπτογράφησης https.

3.7 Ανάλυση και Σχεδίαση Κινητής Εφαρμογής

Σημαντικό ρόλο παίζει και η ανάλυση και σχεδίαση της κινητής εφαρμογής όπου πρέπει να καθοριστούν το πρωτόκολλο το οποίο χρησιμοποιεί για τη επικοινωνία της με τον ιστότοπο όπως και οι περιπτώσεις και τα δικαιώματα χρήσης της.

3.7.1 Πρωτόκολλο Επικοινωνίας Εφαρμογής με Ιστότοπο

Η κινητή εφαρμογή πρέπει με κάποιο τρόπο να συνδέεται με τον ιστότοπο και τη βάση δεδομένων ώστε να μπορεί να παίρνει τα απαραίτητα δεδομένα κάθε φορά που απαιτείται για την ολοκλήρωση κάποιας ενέργειας. Αυτή τη δουλειά κάνει το πρωτόκολλο επικοινωνίας και για την παρούσα εργασία αυτό που επιλέχθηκε είναι το JSON. Αυτό έγινε γιατί το JSON είναι μια ελαφριά μορφή ανταλλαγής κειμένου-δεδομένων, μικρότερο από το XML, γρηγορότερο και ευκολότερο ανάλυση(parsing). Χρησιμοποιεί συντακτικό σεναρίου(script) της JAVA για την περιγραφή αντικειμένων δεδομένων, αλλά δεν εξαρτάται από καμία γλώσσα.[23]

3.7.2 Απαιτήσεις Κινητής Εφαρμογής

Όπως και ο Ιστότοπος έτσι και η εφαρμογή πρέπει να τηρεί κάποια στάνταρ ώστε να είναι λειτουργική και διαχειρίσιμη. Η κυριότερη απαίτηση για την εφαρμογή είναι να υπάρχει ένα εύχρηστο και εύκολα κατανοητό γραφικό περιβάλλον. Επίσης πρέπει να είναι απλή έτσι ώστε και ο πιο αρχάριος χρήστης να μπορεί με λίγες

κινήσεις να πάρει αυτό που ζητάει. Κυρίως να προσφέρεται μια φιλική εμπειρία ανεξάρτητα την εξοικειώσή του με τις έξυπνες κινητές συσκευές.

3.7.3 Περιπτώσεις και Δικαιώματα Χρήσης

Σημαντική είναι και η ανάλυση των περιπτώσεων χρήσης της κινητής εφαρμογής. Η εφαρμογή χρησιμοποιείται μόνο από ανώνυμους χρήστες, δεν χρειάζεται γίνει εγγραφή. Πιθανές περιπτώσεις χρήσης είναι:

- **Αναζήτηση στους χάρτες.** Ο χρήστης επιλέγοντας την αντίστοιχη επιλογή από το μενού μπορεί να μπει στον χάρτη στον οποίο εμφανίζονται οι αποθηκευμένες περιοχές. Εκεί μπορεί να κάνει αναζήτηση κάποιας διεύθυνσης ή συντεταγμένων.
- **Προβολή Περιοχών.** Ο χρήστης μπορεί να δει μια λίστα με όλες τις αποθηκευμένες περιοχές. Διαλέγοντας κάποια μπορεί να δει παραπάνω στοιχεία για αυτή όπως περιγραφή, φωτογραφία και απεικόνισή της στον χάρτη. Πατώντας το κουμπί μπορεί να ξεκινήσει τη καθοδήγησή του σε αυτή.
- **Προβολή Κοντινότερων Περιοχών.** Ο χρήστης μπορεί να δει μια λίστα με όλες τις αποθηκευμένες περιοχές ταξινομημένες από την κοντινότερη στην πιο απομακρυσμένη. Και εδώ επιλέγοντας κάποια μπορεί να δει τα στοιχεία της και να καθοδηγηθεί σε αυτή.
- **Πλοήγηση σε Περιοχή.** Γίνεται πλοήγηση στην περιοχή.
- **Προβολή Περιηγήσεων.** Ο χρήστης μπορεί να δει μια λίστα με όλες τις αποθηκευμένες περιηγήσεις. Διαλέγοντας κάποια μπορεί να δει παραπάνω στοιχεία για αυτή όπως περιγραφή και τα μέρη από τα οποία αποτελείται. Πατώντας το κουμπί μπορεί να ξεκινήσει τη καθοδήγησή στα μέρη που την αποτελούν.
- **Πλοήγηση σε Περιήγηση.** Πλοήγηση στα μέρη από τα οποία αποτελείται η περιήγηση.

3.8 Σύνοψη Κεφαλαίου

Σε αυτό το κεφάλαιο περιγράφηκε αναλυτικά ο τρόπος σχεδιασμού και ανάπτυξης του συστήματος. Περιγράφηκαν οι ρόλοι, οι απαιτήσεις και οι περιπτώσεις

χρήσης του. Έγινε αναφορά στον σχεδιασμό της Βάσης Δεδομένων και στους πίνακές της όπως και στα μέτρα ασφαλείας που λήφθηκαν για τη σωστή λειτουργία. Επίσης έγινε αναφορά και στα βήματα που ακολουθήθηκαν για ανάλυση και σχεδίαση της κινητής εφαρμογής.

Κεφάλαιο 4

Λειτουργίες Ιστοχώρου

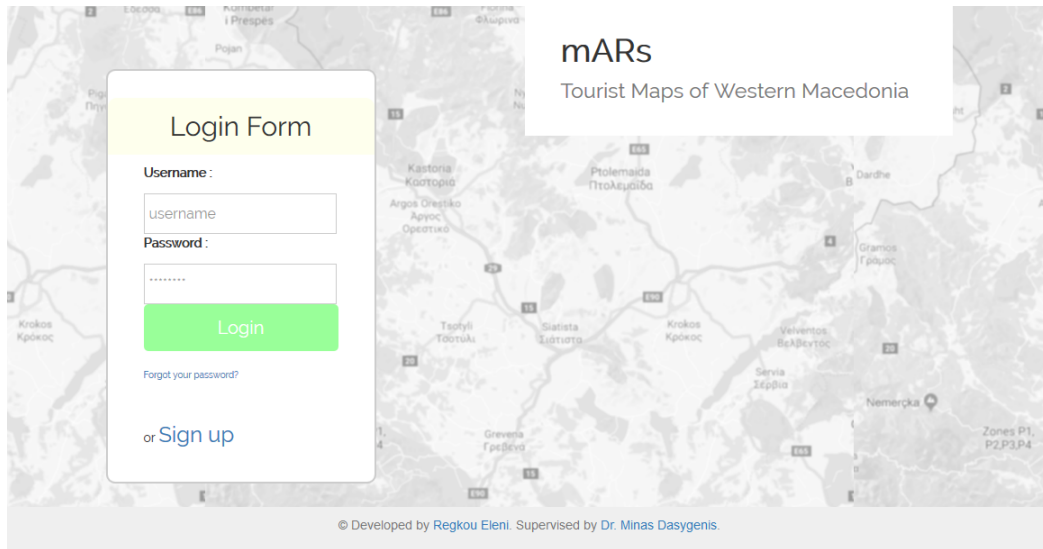
Στο παρόν κεφάλαιο παρουσιάζεται η λειτουργία και υλοποίηση του Ιστοχώρου που αναπτύχθηκε. Ο Ιστοχώρος αποτελεί τη διεπαφή του χρήστη και έχει σκοπό την εύκολη περιήγηση στις λειτουργίες που παρέχονται. Μέσω του Ιστοχώρου οι ενδιαφερόμενοι μπορούν να προσθέτουν περιοχές, αξιοθέατα, σημεία ενδιαφέροντος κλπ όπως περιγράφεται παρακάτω. Συγκεκριμένα παρουσιάζονται οι λειτουργίες εγγραφής και σύνδεσης, οι λειτουργίες χρήστη και διαχειριστή και γίνεται ανάλυση σημαντικών κομματιών κώδικα.

4.1 Λειτουργίες Ανώνυμου Χρήστη

Ο χρήστης που δεν είναι συνδεδεμένος στο σύστημα έχει τη δυνατότητα να: εγγραφεί, να συνδεθεί, να ανακτήσει κωδικό που έχει ξεχάσει, τα οποία αναλύονται παρακάτω.

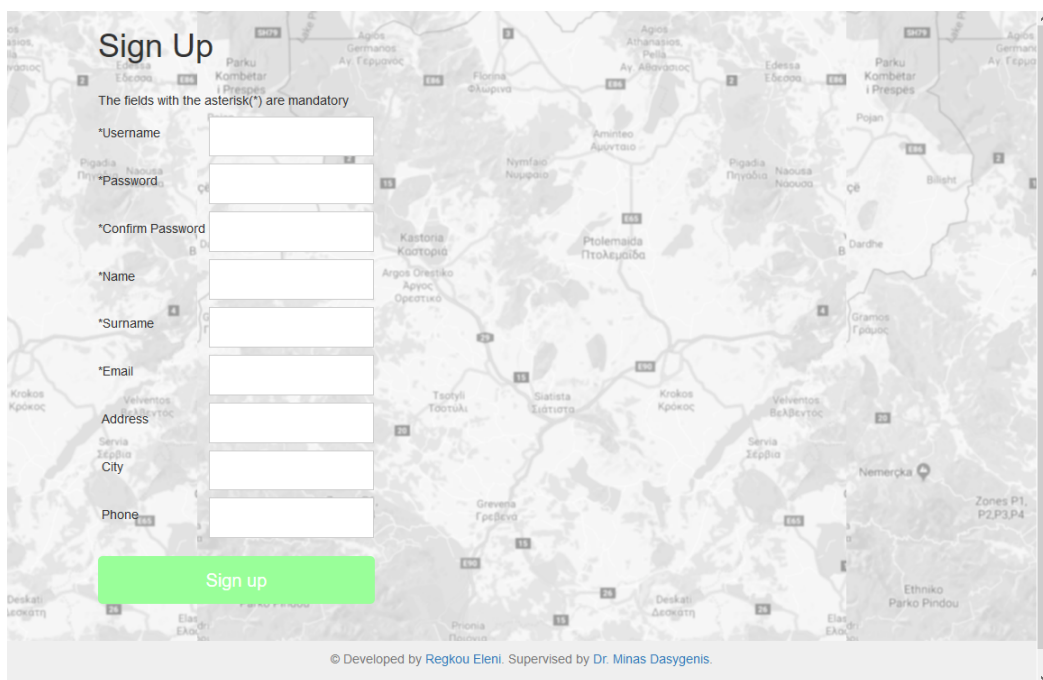
4.1.1 Σύνδεση Χρήστη

Η αρχική σελίδα που βλέπει ο χρήστης όταν επισκέπτεται τον ιστότοπο είναι η σελίδα σύνδεσης, όπως φαίνεται στην Εικόνα 4.1. Εδώ ζητείται από τον χρήστη το όνομα και ο κωδικός του, όπου συμπληρώνοντας τα και πατώντας το κουμπί "Login", αν είναι σωστά συνδέεται στην Ιστοσελίδα. Αν κάποιος από τα δύο είναι λάθος ή μη συμπληρωμένο εμφανίζονται αντίστοιχα μηνύματα για λάθος συνδυασμό ονόματος χρήστη(username) και κωδικού(password) ή μη συμπλήρωσης πεδίου. Από τη σελίδα Σύνδεσης ο χρήστης που δεν έχει συνδεθεί έχει επίσης τη δυνατότητα να επιλέξει να κάνει εγγραφή ως νέος χρήστης αλλά και να κάνει ανάκτηση κωδικού τα οποία βλέπουμε στη συνέχεια.



Σχήμα 4.1: Οθόνη Σύνδεσης Χρήστη

4.1.2 Εγγραφή Χρήστη

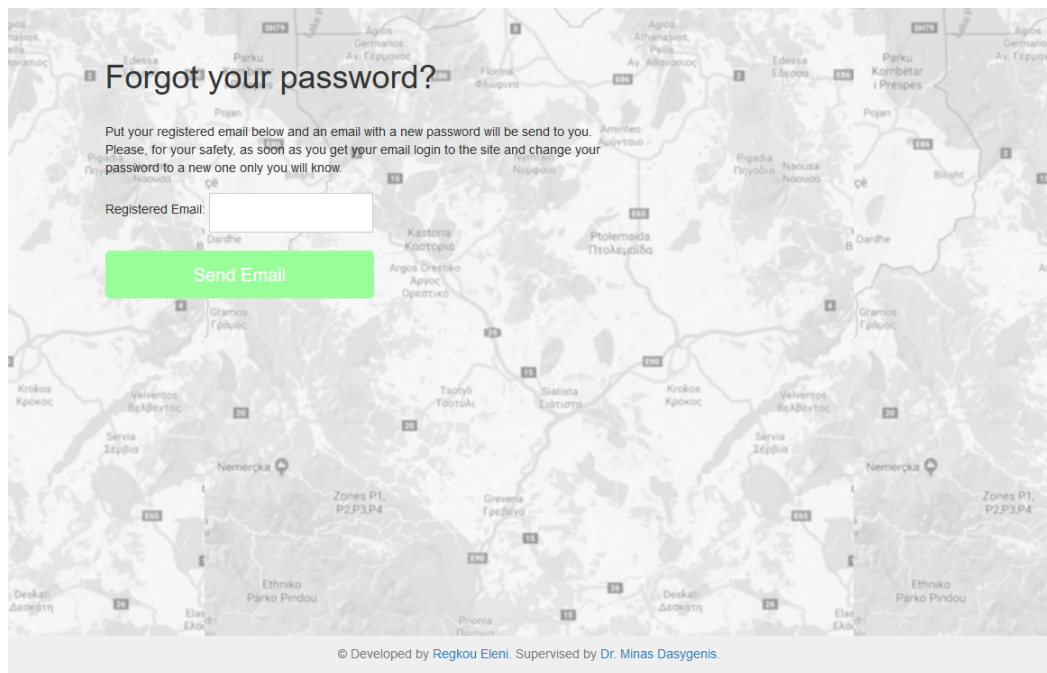


Σχήμα 4.2: Οθόνη Εγγραφής Χρήστη

Πατώντας από τη σελίδα σύνδεσης τον σύνδεσμο "Sign up" ο χρήστης μεταφέρεται στη σελίδα εγγραφής νέου χρήστη. Η σελίδα εγγραφής περιλαμβάνει τη φόρμα εγγραφής με πεδία προς συμπλήρωση τα Username (Όνομα Χρήστη), Password (Κωδικός), Confirm Password (Επαλήθευση Κωδικού), Name (Όνομα), Surname (Επίθετο), Email (Διεύθυνση Ηλεκτρονικού Ταχυδρομείου), Address (Διεύθυνση), City (Πόλη), Phone (Τηλέφωνο) και μπορούμε να τη δούμε στην Εικόνα 4.2. Από αυτά

υποχρεωτικά πρέπει να συμπληρωθούν τα Username, Password, Confirm Password, Name, Surname, Email. Τα πεδία Address, City, Phone είναι προαιρετικά. Επίσης η σελίδα περιλαμβάνει ένα κουμπί "Sign up", με το πάτημα του οποίου υποβάλλεται η φόρμα. Μετά το πάτημα του κουμπιού υποβολής ελέγχεται αν έχουν συμπληρωθεί τα απαραίτητα πεδία. Αν όχι εμφανίζεται στο χρήστη μήνυμα που τον ενημερώνει ότι δεν έχουν συμπληρωθεί όλα τα απαραίτητα πεδία. Στην συνέχεια ελέγχεται αν το email και το username υπάρχουν ήδη στην βάση, αν ναι ειδοποιείται με ανάλογο μήνυμα. Επίσης ελέγχεται αν ταιριάζουν οι εισαγωγές στα πεδία password και confirm password και αν ο κωδικός είναι μεγαλύτερος των 6 ψηφίων, αν όχι εμφανίζονται τα αντίστοιχα μηνύματα. Τέλος, με την επιτυχή υποβολή της φόρμας δημιουργείται ένας καινούριος λογαριασμός, ο χρήστης οδηγείται στη σελίδα σύνδεσης και μπορεί πλέον να συνδεθεί βάζοντας τα στοιχεία με τα οποία έκανε εγγραφή.

4.1.3 Ανάκτηση Κωδικού

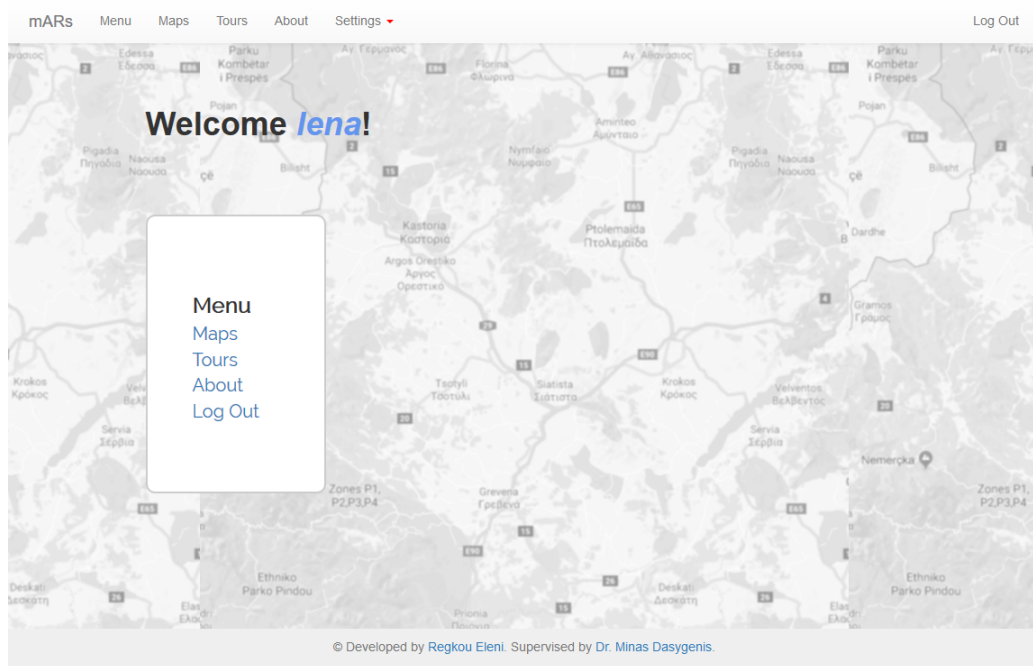


Σχήμα 4.3: Οθόνη Ανάκτησης Κωδικού

Αν ο χρήστης επιλέξει από τη σελίδα σύνδεσης το σύνδεσμο "Forgot your password?" μεταφέρεται στη σελίδα Ανάκτησης Κωδικού που βλέπουμε στην Εικόνα 4.3. Εκεί υπάρχει πεδίο στο οποίο του ζητείται α καταχωρήσει το email με το οποίο εγγράφηκε και το κουμπί "Send Email" με το οποίο το καταχωρεί. Πατώντας το κουμπί,

γίνεται έλεγχος αν το email υπάρχει στη βάση και αν ναι του στέλνεται email, στο οποίο του έχει σταλεί ένας τυχαία παραγόμενος κωδικός με το οποίο μπορεί πλέον να συνδεθεί στον λογαριασμό του. Προτρέπεται μόλις συνδεθεί να τον αλλάξει σε κάποιος που επιθυμεί αυτός. Αν καταχωρηθεί email που δεν υπάρχει στη βάση τότε εμφανίζεται μήνυμα ότι το email δεν υπάρχει.

4.2 Λειτουργίες Συνδεδεμένου Χρήστη



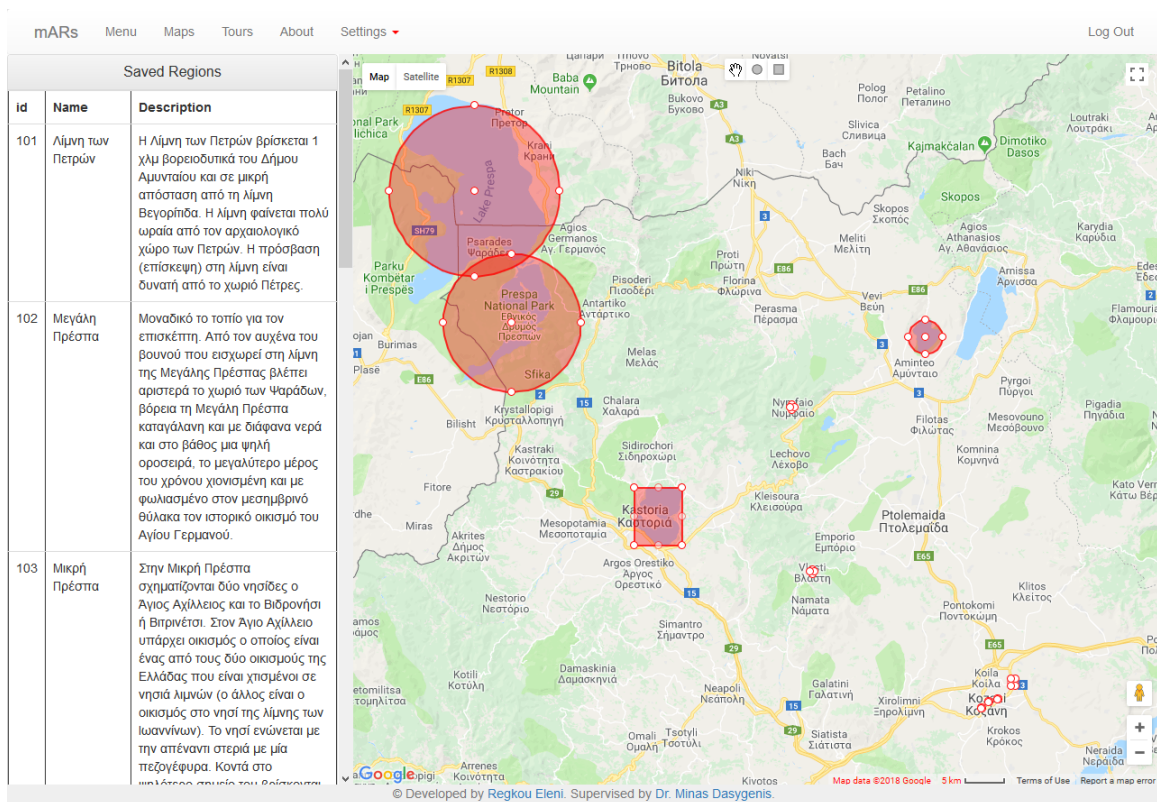
Σχήμα 4.4: Οθόνη Βασικού Μενού Απλού Χρήστη

Από τη στιγμή που ο χρήστης συνδεθεί στο σύστημα μεταφέρεται στην σελίδα του Βασικού Μενού όπως παρουσιάζεται στην Εικόνα 4.4. Όπως φαίνεται εκεί υπάρχει ένα μήνυμα υποδοχής και το μενού μέσα από το οποίο μπορεί να αποκτήσει πρόσβαση σε άλλες σελίδες του ιστοτόπου. Το μενού αποτελείται από τους συνδέσμους "Maps"(Χάρτες), "Tours"(Περιηγήσεις), "About"(Σχετικά Με) και "Log Out"(Αποσύνδεση). Επίσης στο πάνω μέρος της οθόνης βλέπουμε την Μπάρα Πλοήγησης, η οποία υπάρχει σε όλες τις σελίδες του ιστοχώρου. Παρέχει του ίδιους συνδέσμους με το Μενού ώστε να γίνεται πιο εύκολη η πλοήγηση για το χρήστη και επιπλέον δίνει στο χρήστη να μεταβεί στις ρυθμίσεις του (Settings>Account Settings) μέσα από τις οποίες μπορεί να αλλάξει στοιχεία του λογαριασμού του. Όπως φαίνεται υπάρχει ένα πλήθος ενεργειών που μπορεί να εκτελέσει οι οποίες

περιγράφονται παρακάτω.

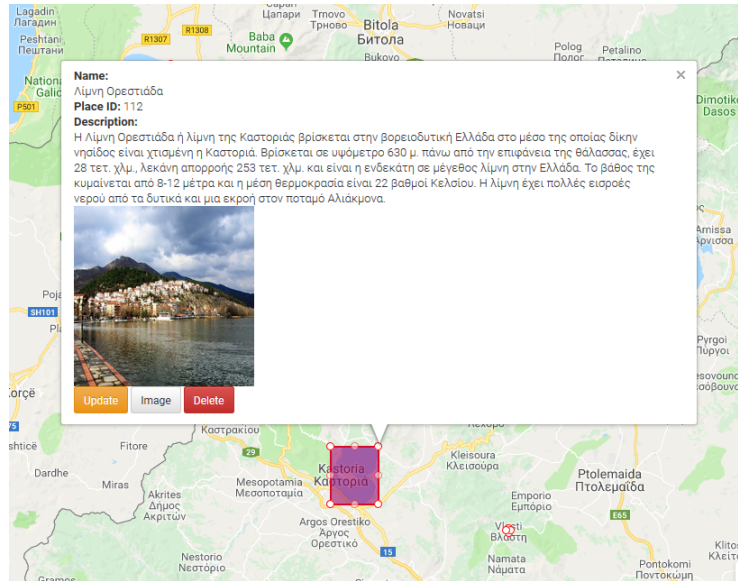
4.2.1 Χάρτες

Με την επιλογή του συνδέσμου "Maps" εμφανίζεται στο χρήστη η σελίδα με τους Χάρτες. Όπως φαίνεται στην Εικόνα 4.5, η σελίδα των Χαρτών αποτελείται από ένα μεγάλο χάρτη στα δεξιά της οθόνης και τον πίνακα "Saved Regions" στα αριστερά.



Σχήμα 4.5: Οθόνη Χαρτών

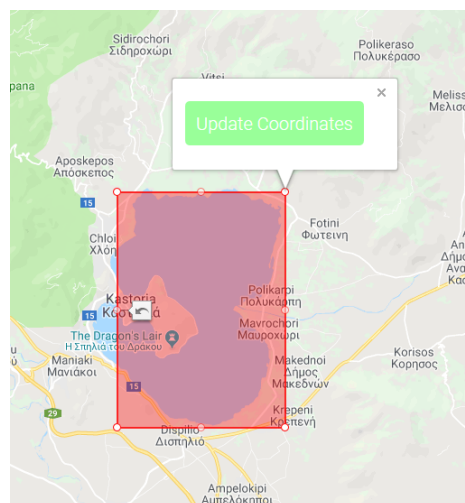
Ξεκινώντας με την ανάλυση του χάρτη, ο χρήστης μπορεί να δει κάποιες περιοχές σημειωμένες με κόκκινα χρώμα σε ορθογώνια ή κυκλικά σχήματα. Αυτές είναι οι αποθηκευμένες περιοχές του. Πατώντας πάνω σε κάποια, εμφανίζεται ένα παράθυρο πληροφοριών, Εικόνα 4.6, με το όνομα της επιλεγμένης περιοχής, την περιγραφή και την εικόνα της, εάν αυτή υπάρχει. Επιπλέον, υπάρχει η δυνατότητα να αλλάξουν οι συντεταγμένες των αποθηκευμένων περιοχών. Τραβώντας τα κυκλικά σημεία που εμφανίζονται στο κάθε αποθηκευμένο σχήμα αλλάζουν οι διαστάσεις της περιοχής. Με το τέλος του "τραβήγματος" εμφανίζεται ένα παράθυρο, Εικόνα 4.7, και πατώντας το "Update Coordinates" ο χρήστης μπορεί να αποθηκεύσει τις



Σχήμα 4.6: Παράθυρο Πληροφοριών

καινούριες συντεταγμένες.

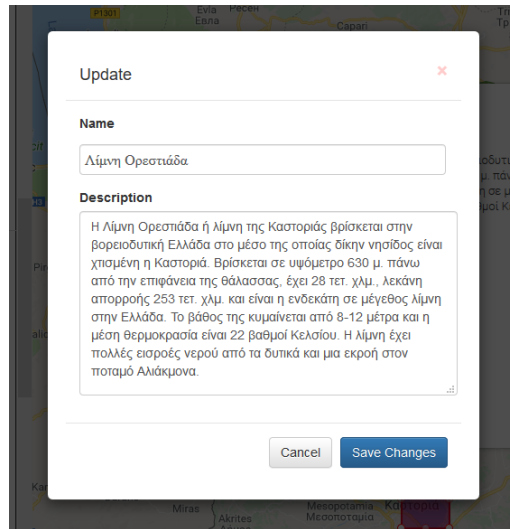
Επίσης υπάρχουν τα κουμπιά "Update", "Image" και "Delete". Πατώντας το κουμπί "Update", Εικόνα 4.8, εμφανίζεται ένα παράθυρο στο οποίο υπάρχει μια φόρμα με φορτωμένα αποθηκευμένα στοιχεία και μπορούμε να κάνουμε επεξεργασία το όνομα και την περιγραφή της περιοχής. Με το κουμπί "Save Changes" οι αλλαγές αποθηκεύονται. Με την επιλογή του κουμπιού "Image" ο χρήστης μεταφέρεται στην



Σχήμα 4.7: Αλλαγή Συντεταγμένων Περιοχής

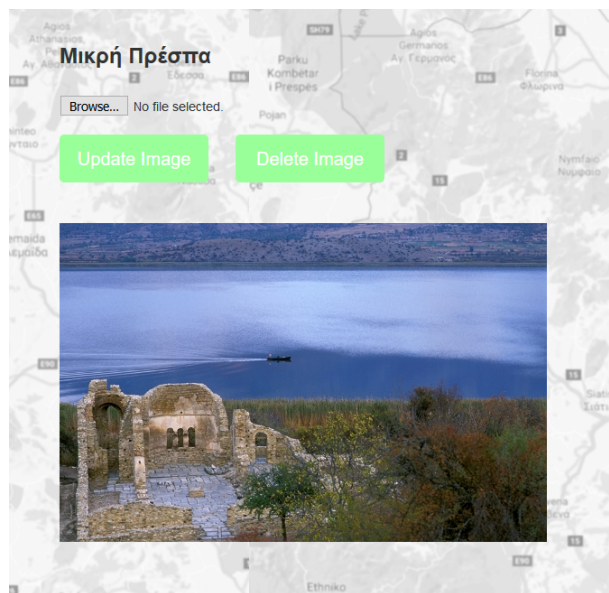
οθόνη της Εικόνας 4.9. Στην οθόνη αυτή ο χρήστης μπορεί να δει την αποθηκευμένη εικόνα σε μεγαλύτερη ανάλυση και, μέσω των κουμπιών "Update Image", "Delete Image", να την αντικαταστήσει ή να τη διαγράψει αντίστοιχα. Πατώντας το κουμπί "Delete" εμφανίζεται μήνυμα επιβεβαίωσης διαγραφής και αν ο χρήστης συμφωνή-

σει η περιοχή διαγράφεται.



Σχήμα 4.8: Παράθυρο Επεξεργασίας

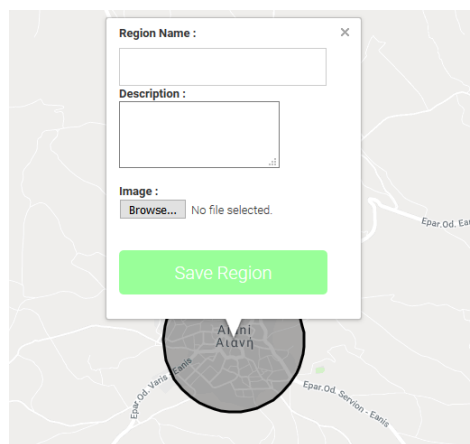
Στη μέση του πάνω μέρος του χάρτη επίσης, υπάρχουν τρία εικονίδια στη σειρά, ένα χέρι, ένας κύκλος και ένα ορθογώνιο. Αυτά είναι τα εργαλεία με τα οποία ξεκινάει η εισαγωγή μιας περιοχής. Επιλέγοντας τον κύκλο ή το ορθογώνιο και πατώντας κάπου στο χάρτη ξεκινάει η σχεδίαση της περιοχής. Ο χρήστης μπορεί να επιλέξει μια περιοχή όσο μικρή ή μεγάλη θέλει. Μόλις είναι ικανοποιημένος με το μέγεθος, πατάει δεύτερη φορά και τερματίζει η σχεδίαση περιοχής. Με τον τερμα-



Σχήμα 4.9: Επεξεργασία Εικόνας

τισμό της σχεδίασης εμφανίζεται ένα παράθυρο, που όπως παρουσιάζεται και στην Εικόνα 4.10 πρέπει να συμπληρωθούν το όνομα και η περιγραφή της επιλεγμένης

περιοχής, τα οποία είναι απαραίτητα για την αποθήκευση της. Υπάρχει επίσης η δυνατότητα να προστεθεί εικόνα της περιοχής η οποία είναι προαιρετική. Πατώντας το "Save Region" η περιοχή αποθηκεύεται και πλέον είναι ορατή στο χάρτη όπως και στον πίνακα που θα αναλύσουμε στη συνέχεια. Με το εργαλείο με το χέρι αποεπιλέγονται τα άλλα δύο εργαλεία. Ο χάρτης επιπλέον παρέχει τις δυνατότητες: να μπει σε πλήρη οθόνη (εικονίδιο πάνω δεξιά), να εμφανίζεται από δορυφόρο ή όχι (εικονίδια πάνω αριστερά) να γίνει μεγιστοποίηση ή ελαχιστοποίηση του (εικονίδια κάτω δεξιά). Έπειτα αναλύεται ο πίνακας "Saved Regions" ο οποίος περιλαμβάνει

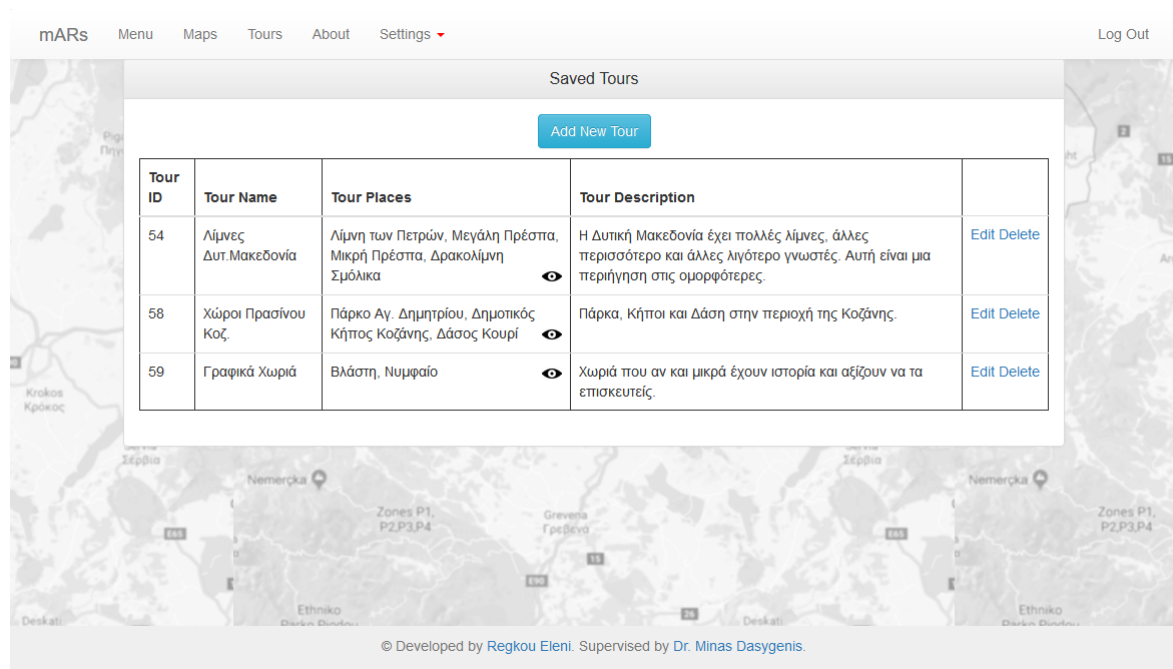


Σχήμα 4.10: Εισαγωγή Περιοχής

νει όλες τις περιοχές του χρήστη. Όπως φαίνεται και στην Εικόνα 4.5, ο πίνακας εμφανίζει για την κάθε περιοχή το id, το όνομα και την περιγραφή της. Επίσης, πατώντας σε κάποια εγγραφή του πίνακα, χρωματίζεται στο χάρτη η περιοχή στην οποία αναφέρεται και εμφανίζεται το παράθυρο με τις πληροφορίες της, το οποίο φαίνεται και στην Εικόνα 4.6. Στον πίνακα "Saved Regions" είναι επίσης δυνατό να γίνει αύξουσα ή φθίνουσα αλφαβητική ταξινόμηση κατά τα ονόματα των περιοχών, πατώντας πάνω στον τίτλο "Name" του πίνακα.

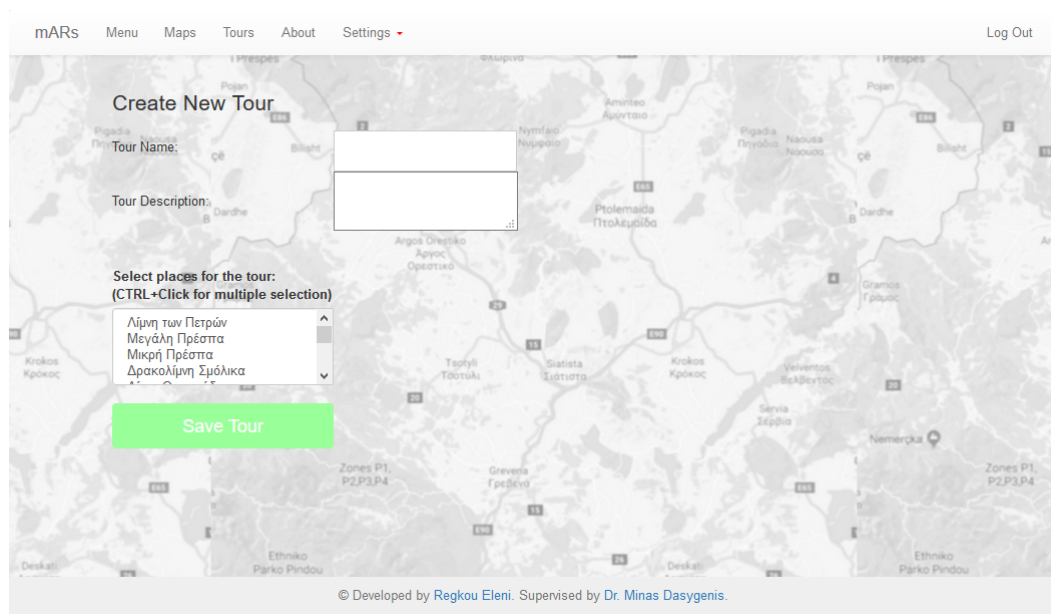
4.2.2 Περιηγήσεις

Επιλέγοντας το "Tours" (Περιηγήσεις) ο χρήστης μεταφέρεται στην σελίδα Αποθηκευμένες Περιηγήσεις (Saved Tours). Εκεί, όπως φαίνεται και στην Εικόνα 4.11, εμφανίζεται ένα κουμπί με όνομα "Add New Tour" με το οποίο γίνεται προσθήκη καινούριας Περιήγησης και ένα πίνακας με τα αποθηκευμένα tours με το όνομα (Tour Name), τα ονόματα από τα μέρη που περιλαμβάνουν (Tour Places) και την



Σχήμα 4.11: Οθόνη Αποθηκευμένων Περιηγήσεων

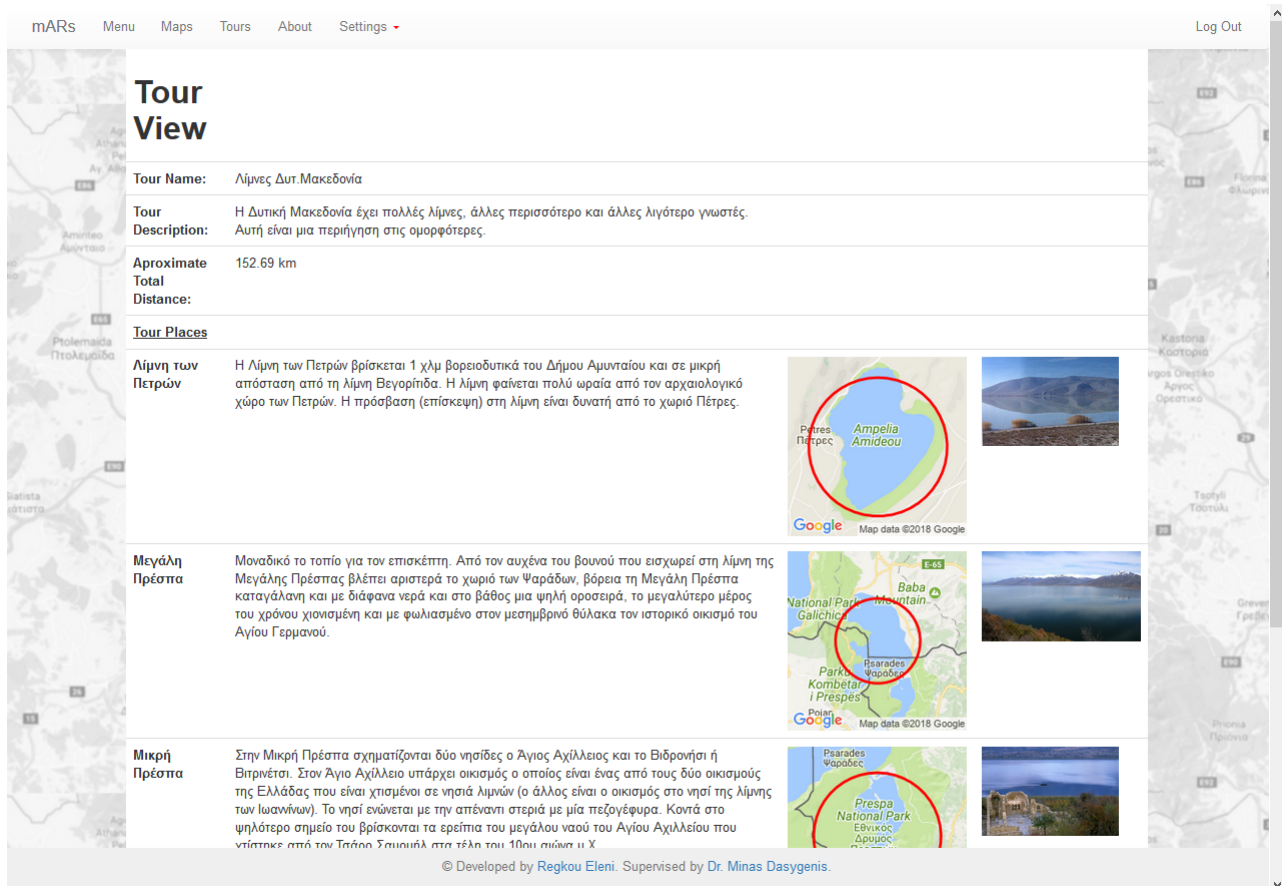
περιγραφή τους (Tour Description). Για την κάθε εγγραφή υπάρχει επίσης ένα Κουμπί Προβολής Περιήγησης (View Tour) που είναι το εικονίδιο με το μάτι και δύο σύνδεσμοι για επεξεργασία (Edit) και διαγραφή (Delete) της Περιήγησης.



Σχήμα 4.12: Οθόνη Προσθήκης Νέας Περιήγησης

Επιλέγοντας το κουμπί "Add New Tour" εμφανίζεται η σελίδα προσθήκης νέας περιήγησης, όπως δείχνει η Εικόνα 4.12. Αυτή περιέχει μια φόρμα με τα πεδία Tour Name (Όνομα Περιήγησης), Tour Description (Περιγραφή Περιήγησης) και μία

Λίστα με της αποθηκευμένες περιοχές του χρήστη ώστε να επιλέξει ποιες θέλει να περιλαμβάνει η Περιήγηση που θα δημιουργήσει. Συμπληρώνοντας τα πεδία και πατώντας το κουμπί "Save Tour" γίνεται αποθήκευση της νέας περιήγησης.



Σχήμα 4.13: Οθόνη Προβολής Περιήγησης

Πατώντας το κουμπί της Προβολής Περιήγησης ο χρήστης μπορεί να δει αναλυτικά τα στοιχεία της Περιήγησης, όπως στην Εικόνα 4.13. Φαίνεται το όνομα, η περιγραφή, μια εκτίμηση της συνολικής απόστασης, και τα μέρη που το αποτελούν αναλυτικά, με τα ονόματα, τις περιγραφές τους, ένα χάρτη που δείχνει που βρίσκονται και τη φωτογραφία τους, αν υπάρχει.

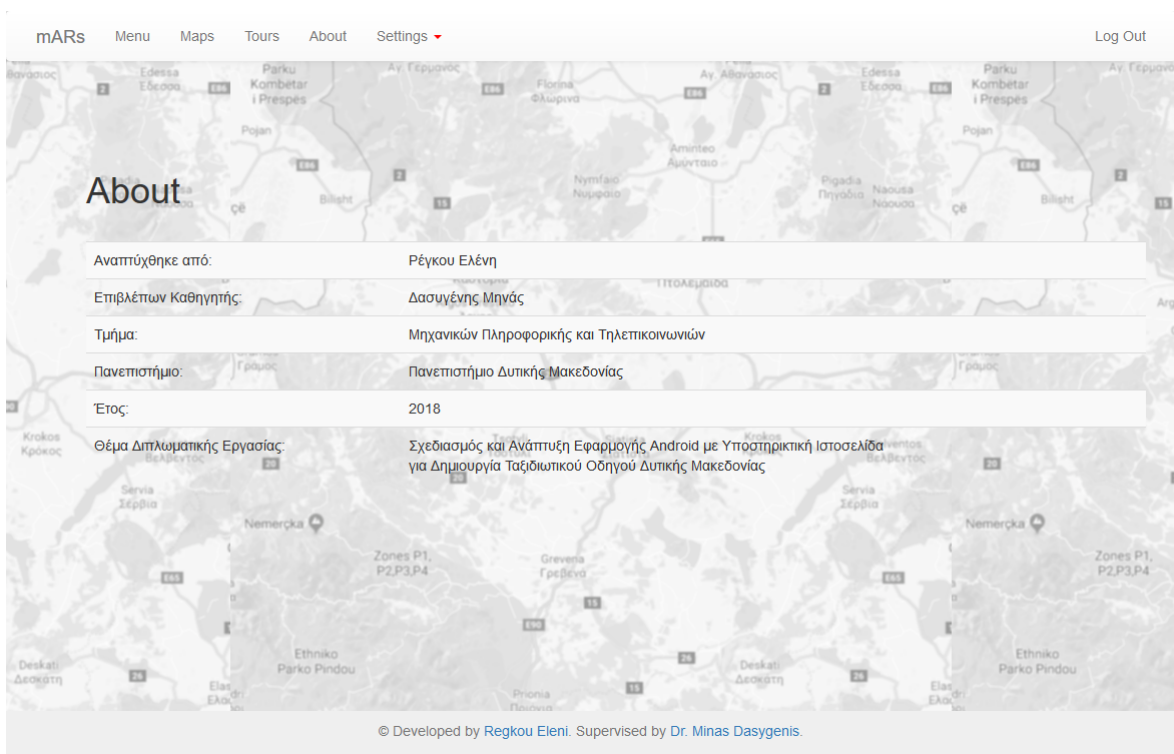
Πατώντας την Επεξεργασία της Περιήγησης εμφανίζεται η αντίστοιχη σελίδα, Εικόνα 4.14. Εκεί υπάρχει η δυνατότητα να γίνει αλλαγή του ονόματος, της περιγραφής αλλά και των περιοχών που περιλαμβάνονται. Στη συνέχεια πατώντας το "Submit Edits" οι αλλαγές καταχωρούνται.

Τέλος με το κουμπί "Delete" εμφανίζεται ένα μήνυμα επιβεβαίωσης για τη διαγραφή της Περιήγησης και αν ο χρήστης συμφωνήσει αυτή διαγράφεται.



Σχήμα 4.14: Οθόνη Επεξεργασίας Περιήγησης

4.2.3 Σχετικά Με



Σχήμα 4.15: Οθόνη Σχετικά Με

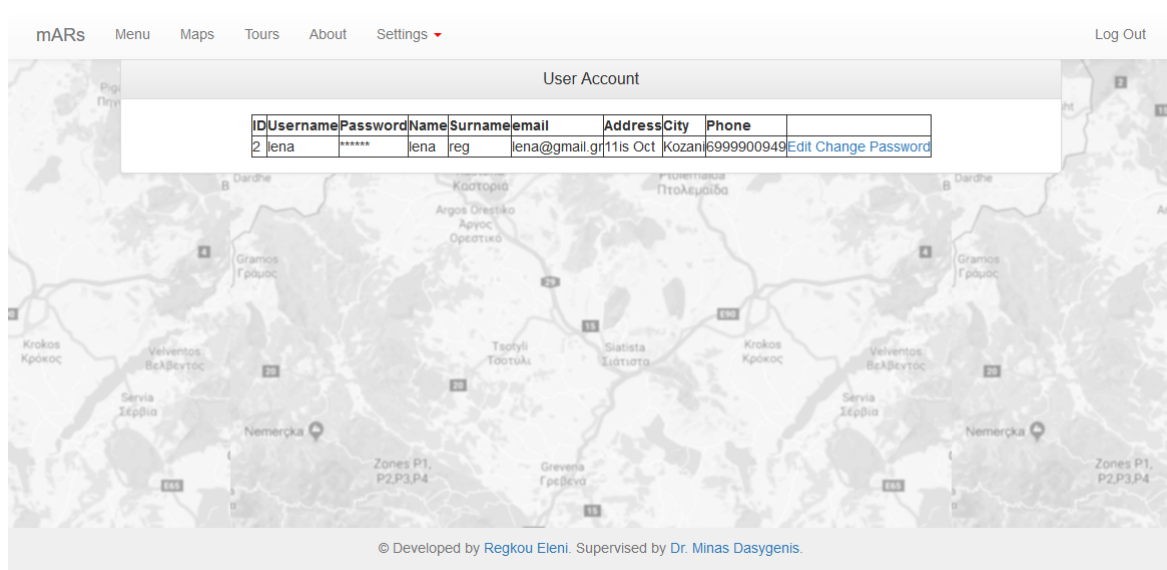
Με την επιλογή του "Σχετικά Με" ο χρήστης οδηγείται στη σχετική σελίδα, Εικόνα 4.15, όπου μπορεί να δει πληροφορίες για το ποιος ανέπτυξε την ιστοσελίδα, το όνομα του επιβλέπων, το τμήμα, το πανεπιστήμιο, το έτος και το θέμα της δι-

πλωματικής εργασίας της οποίας είναι μέρος.

4.2.4 Αποσύνδεση

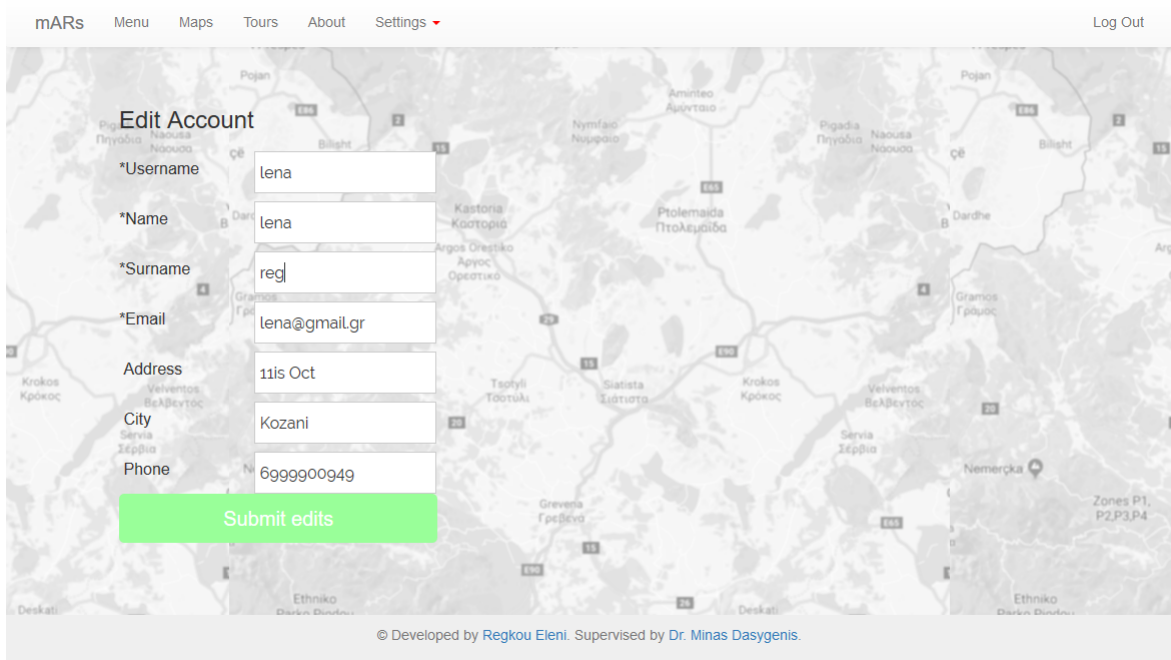
Με το πάτημα της Αποσύνδεσης ("Log Out"), ο χρήστης αποσυνδέεται από το σύστημα και μεταφέρεται στη σελίδα σύνδεσης. Για να κάνει οποιαδήποτε ενέργεια πρέπει να επανασυνδεθεί.

4.2.5 Ρυθμίσεις Λογαριασμού

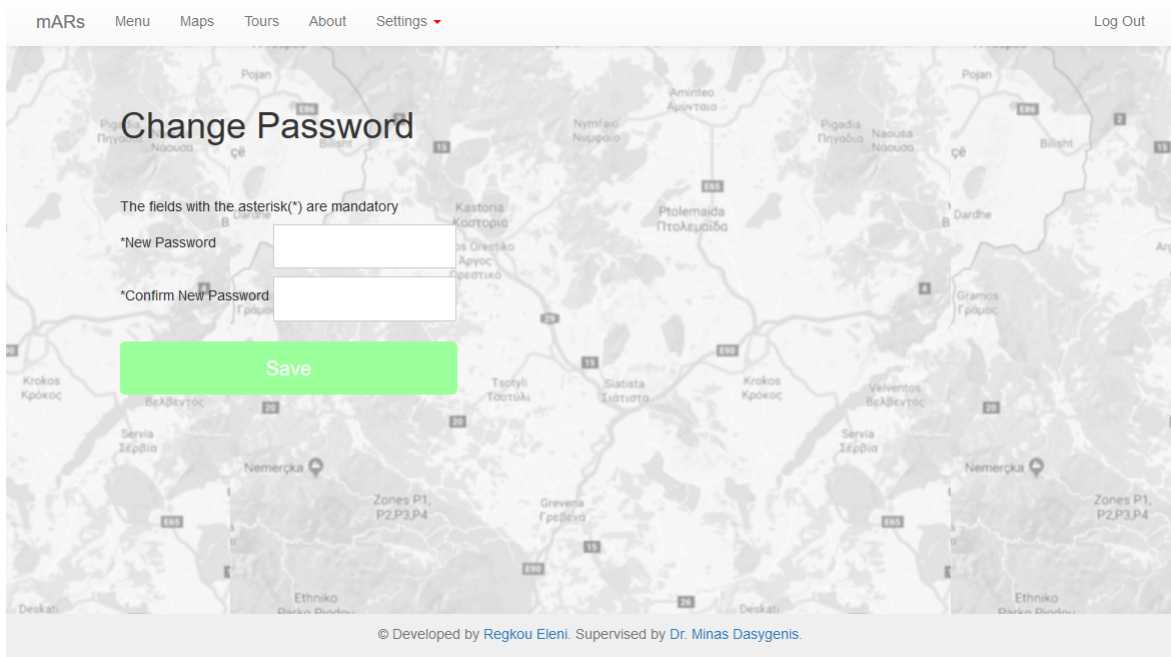


Σχήμα 4.16: Οθόνη Ρυθμίσεις Λογαριασμού

Με την επιλογή του "Ρυθμίσεις Λογαριασμού" ο χρήστης μεταφέρεται στη σελίδα Λογαριασμός Χρήστη, όπως φαίνεται στην Εικόνα 4.16. Εκεί μπορεί να δει τα αποθηκευμένα του προσωπικά στοιχεία, να τα αλλάξει ή να αλλάξει τον κωδικό του. Αν επιλέξει τον σύνδεσμο "Edit" (Επεξεργασία) οδηγείται σε μια οθόνη με μια φόρμα, Εικόνα 4.17, με φορτωμένα τα ήδη αποθηκευμένα στοιχεία του, όπου μπορεί να αλλάξει ότι θέλει, αρκεί να είναι συμπληρωμένα τα απαραίτητα πεδία τα οποία είναι τα ίδια με αυτά της εγγραφής, και πατώντας το κουμπί Submit Edits (Καταχώρηση Αλλαγών) αποθηκεύονται στη βάση. Αν επιλέξει το σύνδεσμο "Change Password" (Αλλαγή Κωδικού) οδηγείται στην αντίστοιχη σελίδα, Εικόνα 4.18. Εκεί του ζητείται να δώσει νέο κωδικό και την επαλήθευσή του. Όπως και στη σελίδα εγγραφής, ισχύει ότι πρέπει να είναι ίδιοι μεταξύ τους και να είναι παραπάνω από 6 χαρακτήρες. Πατώντας το κουμπί "Save" (Αποθήκευση) ο καινούριος



Σχήμα 4.17: Οθόνη Επεξεργασίας Λογαριασμού



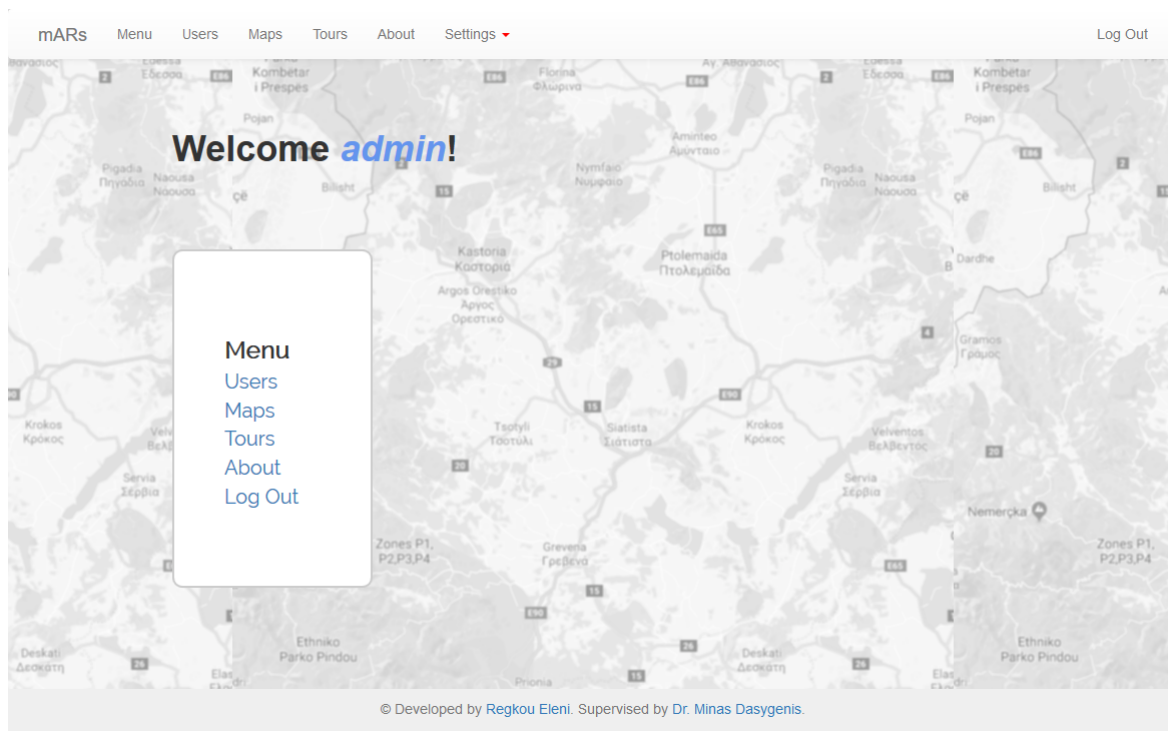
Σχήμα 4.18: Οθόνη Αλλαγής Κωδικού

κωδικός αποθηκεύεται.

4.3 Λειτουργίες Διαχειριστή

Ο διαχειριστής έχει τη δυνατότητα να εκτελέσει όλες τις λειτουργίες που μπορεί να εκτελέσει ο απλός χρήστης αλλά και κάποιες περισσότερες. Το Μενού του

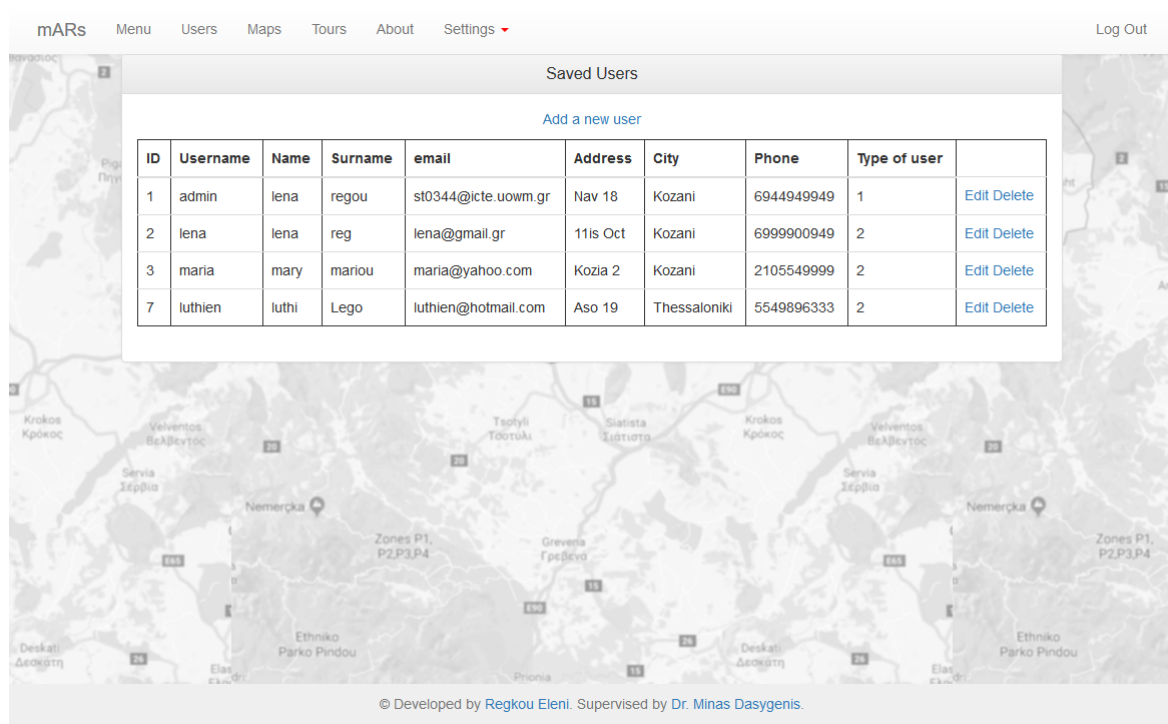
Κεφάλαιο 4. Λειτουργίες Ιστοχώρου



Σχήμα 4.19: Οθόνη Μενού Διαχειριστή

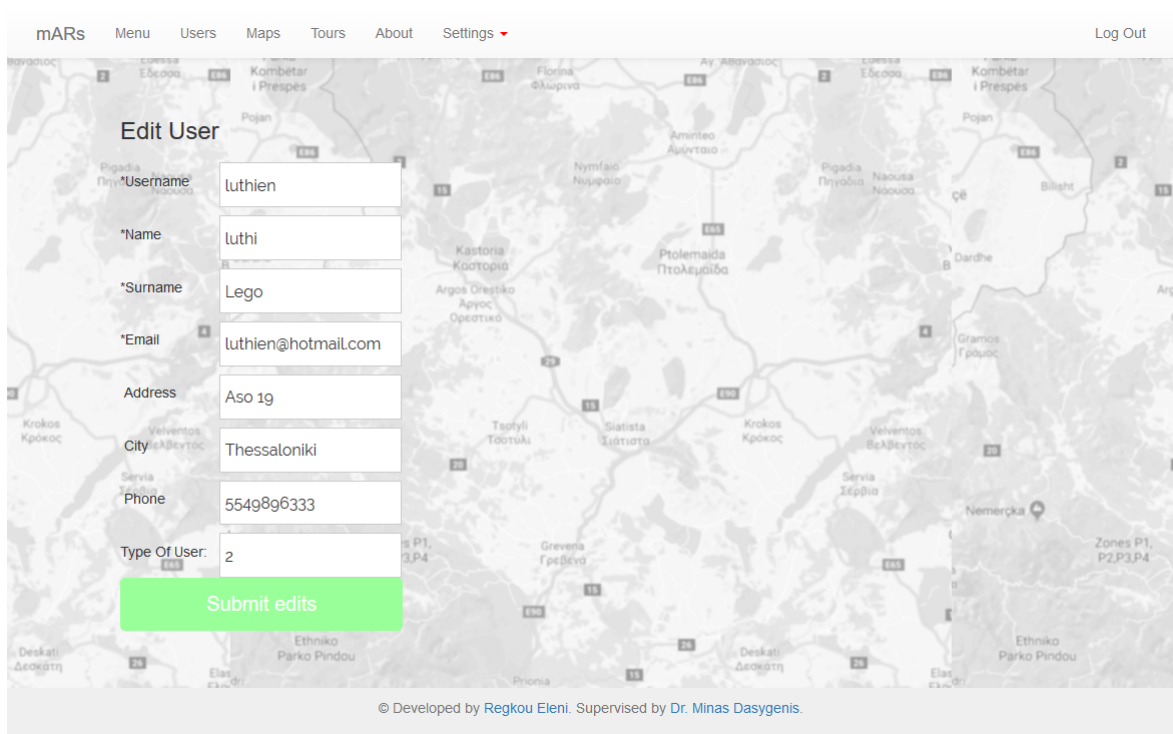
Διαχειριστή φαίνεται στην Εικόνα 4.19

4.3.1 Χρήστες



Σχήμα 4.20: Οθόνη Αποθηκευμένων Χρηστών

Στο Μενού και στη Μπάρα Περιήγησης του εμφανίζεται μία επιπλέον επιλογή "Users"(Χρήστες). Επιλέγοντάς την μεταφέρεται στη σελίδα Saved Users(Αποθηκευμένοι Χρήστες) όπου, όπως φαίνεται και στην Εικόνα 4.20, αποτελείται από ένα πίνακα με τους αποθηκευμένους χρήστες και τα προσωπικά τους στοιχεία και τον τύπο χρήστη, αν δηλαδή είναι διαχειριστές(1) ή απλοί χρήστες(2).



Σχήμα 4.21: Οθόνη Επεξεργασίας Στοιχείων Χρήστη

Επίσης μπορεί να διαλέξει το Edit για να μεταφερθεί στη σελίδα επεξεργασίας των στοιχείων του χρήστη, Εικόνα 4.21 όπου μπορεί να κάνει κάποια αλλαγή όπως και να αλλάξει και τον τύπο του χρήστη από απλό σε διαχειριστή ή και το ανάποδο. Πατώντας το "Submit Edits" τα αλλαγμένα στοιχεία αποθηκεύονται. Επιλέγοντας το Delete εμφανίζεται ένα παράθυρο επιβεβαίωσης διαγραφής και αν ο διαχειριστής συμφωνήσει ο χρήστης διαγράφεται.

4.3.2 Επιπλέον Δυνατότητες

Στις λειτουργίες που έχει κοινές με τον απλό χρήστη, ο διαχειριστής μπορεί να δει/επεξεργαστεί/διαγράψει όλες τις αποθηκευμένες Περιοχές και Περιηγήσεις που είναι αποθηκευμένες στη βάση, δηλαδή από όλους τους χρήστες και όχι μόνο αυτές που έχει δημιουργήσει ο ίδιος. Επίσης μπορεί να δει σε κάθε μία από τις περιοχές και τις περιηγήσεις ποιος χρήστης τα έχει δημιουργήσει.

4.4 Ανάλυση Σημαντικών Σημείων Κώδικα Ιστοσελίδας

Σε αυτό το σημείο αναλύονται κάποια σημαντικά κομμάτια κώδικα που έπαιξαν σημαντικό ρόλο στην ανάπτυξη της Ιστοσελίδας.

adminsession.php

```
1 <?php
2 /* user file */
3 require 'connect.php';
4 include('https.php'); //Includes the control file that always redirects to https
5
6 session_start();// Starting Session
7
8 // Storing Session
9 $user_check=$_SESSION['username'];
10
11 //Fetching Complete Information Of User
12 try {
13     $result = $dbh ->prepare("SELECT username, tou FROM users WHERE username=:user_check");
14     $result->bindParam(':user_check', $user_check, PDO::PARAM_STR);
15     $result->execute();
16 }
17 catch(PDOException $e) {
18     echo "Error: " . $e->getMessage();
19 }
20 $row = $result->fetch(PDO::FETCH_ASSOC);
21
22
23 $login_session=$row['username'];
24 $admin_check=$row['tou'];
25 if(!isset($login_session)){
26     $dbh = null;
27     header('Location: index.php'); // Redirecting To Home Page
28 }
29 elseif($admin_check != '1'){
30     $dbh = null;
31     header('Location: profile.php'); // Redirecting To Simple User profile page
32 }
33 ?>
34
```

Σχήμα 4.22: adminsession.php

Το αρχείο στην Εικόνα 4.22 ελέγχει αν ο χρήστης έχει δικαιώματα διαχειριστή. Εάν ναι, τότε επιτρέπει στο χρήστη να ανακατευθυνθεί στην επόμενη διαχειριστική ιστοσελίδα, αλλιώς, με την προϋπόθεση ότι είναι συνδεδεμένος, τον επιστρέφει στη σελίδα του βασικού μενου. Αυτό παρέχει επιπλέον ασφάλεια, γιατί αποτρέπει κάποιον απλό χρήστη να περιηγηθεί στις ιστοσελίδες του συστήματος που δεν πρέπει να έχει πρόσβαση. Το αρχείο αυτό γίνεται include ή require από κάθε αρχείο που μόνο οι διαχειριστές πρέπει να χειρίζονται.

```
1 <?php
2 /* user file */
3
4 $host = '████████████████████.sock';
5 $dbname = 'places';
6 $user = '██████';
7 $pass = '██████';
8
9
10 try(
11     $dbh = new PDO("mysql:unix_socket=$host;dbname=$dbname;charset=utf8", $user, $pass);
12     $dbh->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
13     $dbh->setAttribute(PDO::ATTR_EMULATE_PREPARES, false);
14     $dbh->query('set character_set_client=utf8');
15     $dbh->query('set character_set_connection=utf8');
16     $dbh->query('set character_set_results=utf8');
17     $dbh->query('set character_set_server=utf8');
18 )
19 catch(PDOException $e){
20     die('Connection error: ' . $e->getMessage());
21 }
22 >>
```

Σχήμα 4.23: connect.php

connect.php

Το αρχείο connect.php που βλέπουμε στην Εικόνα 4.23 περιέχει όλο τον απαραίτητο κώδικα για τη σύνδεση με τη βάση δεδομένων MySQL. Ορίζονται οι πληροφορίες σύνδεσης, όπως η διαδρομή του socket σύνδεσης, το όνομα της βάσης δεδομένων, το όνομα χρήστη και ο κωδικός πρόσβασης. Η σύνδεση της βάσης δεδομένων και τα ερωτήματα που γίνονται στη συνέχεια γίνονται με εντολές PDO. Επίσης, χρησιμοποιούνται ειδικές εντολές για την υποστήριξη και χρήση ελληνικών χαρακτήρων.

forgottenpassword.php

```
18 function random_str($length, $keyspace = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ')
19 {
20     $str = '';
21     $max = mb_strlen($keyspace, '8bit') - 1;
22     for ($i = 0; $i < $length; ++$i) {
23         $str .= $keyspace[rand(0, $max)]; //rand instead of random_int
24     }
25     return $str;
26 }
27
```

Σχήμα 4.24: random_str()-forgottenpassword.php

Αυτή η συνάρτηση, Εικόνα 4.24, δημιουργεί μια τυχαία συμβολοσειρά χρησιμοποιώντας μία κρυπτογραφικά ασφαλή γεννήτρια αριθμών ψευδοτυχαίων αριθμών την rand().

Ο κώδικας στην Εικόνα 4.25 αλλάζει τον κωδικό στη βάση με τον τυχαίο καινούριο και τον στέλνει με email στον χρήστη μαζί με ένα μήνυμα πληροφόρησης.

```
//NEW Password
$a = random_str(8);
$hash = password_hash($a, PASSWORD_DEFAULT);
//SAVE NEW PASSWORD IN DB
try {
    $query = $dbh ->prepare("UPDATE users SET password='$hash' WHERE email=:email");
    $query->bindParam(':email', $email, PDO::PARAM_STR);
    $query->execute();
}
catch(PDOException $e) {
    echo "Error: " . $e->getMessage();
}
if ($query) {
    //SEND EMAIL - The message
    $message = "You asked your password to be changed. This is your new temporary password: $a
    As soon as you login, go to your settings and change it to a new one only you will know. ";
    // In case any of our lines are larger than 70 characters, we should use wordwrap()
    $message = wordwrap($message, 70, "\r\n");
    // Send
    mail($email, 'Forgotten Password', $message);

    echo ("<SCRIPT LANGUAGE='JavaScript'>
        window.alert('Email sent.')
        window.location.href='index.php'
    </SCRIPT>");
}
}
```

Σχήμα 4.25: forgottenpassword.php

getjson.php

```
1 <?php
2 /* user file */
3 require('connect.php');
4
5 try {
6     $result = $dbh ->prepare("SELECT * FROM allplaces");
7     $result->execute();
8 }
9 catch(PDOException $e) {
10    echo "Error: " . $e->getMessage();
11 }
12 //Create an array
13 $json_response = array();
14 $json_response['allplaces'] = array(); //show the tablename
15
16 while ( $row = $result->fetch(PDO::FETCH_ASSOC) ) {
17
18     $row_array['place_id'] = $row['place_id'];
19     $row_array['user_id'] = $row['user_id'];
20     $row_array['name'] = $row['name'];
21     $row_array['description'] = $row['description'];
22     $row_array['type'] = $row['type'];
23     $row_array['coordinates'] = $row['coordinates'];
24     $row_array['pfimage'] = $row['pfimage'];
25
26     //push the values in the array
27     array_push($json_response['allplaces'],$row_array); //here too
28 }
29 echo json_encode($json_response);
30 ?>
```

Σχήμα 4.26: getjson.php

Η Εικόνα 4.26 είναι το αρχείο μέσω του οποίου επικοινωνεί με κινητή συσκευή. Η εφαρμογή ζητάει αυτό το αρχείο για να πάρει τις πληροφορίες των περιοχών από

τη βάση.

https.php

```
1 <?php
2 /* user file */
3 if( !isset($_SERVER['HTTPS']) || $_SERVER['HTTPS'] == "" ){
4
5     $redirect = "https://".$_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'];
6
7     header("HTTP/1.1 301 Moved Permanently");
8     header("Location: $redirect");
9
10    die("Please visit <a href=$redirect>$redirect </a>");
11 }
12 ?>
```

Σχήμα 4.27: https.php

Όπως φαίνεται στην Εικόνα 4.27, ασφαλέστερος τρόπος πρόσβασης σε μία ιστοσελίδα είναι μέσω κρυπτογράφησης https, για το λόγο αυτό, δημιουργήθηκε το αρχείο https.php, το οποίο γίνεται include ή require από κάθε άλλο αρχείο του συστήματος. Το αρχείο αυτό αποτελεί έναν έλεγχο, ώστε σε περίπτωση που ο χρήστης δώσει http:// να ανακατευθύνεται στο ίδιο αρχείο με https.

login.php

Στην Εικόνα 4.28 ξεκινώντας ανακτώνται τα στοιχεία σύνδεσης που εισήγαγε ο χρήστης, το όνομα χρήστη (username) και ο κωδικός πρόσβασης (password) και απολυμαίνονται. Στη συνέχεια συντάσσεται ένα ερώτημα (query) SQL προς τον πίνακα users. Εφόσον το ερώτημα SELECT επιστρέφει μια εγγραφή, η διαπίστευση των στοιχείων του χρήστη θεωρείται επιτυχής και ο μετρητής (count) λαμβάνει την τιμή 1. Ο έλεγχος του κωδικού πρόσβασης γίνεται στη συνάρτηση επιβεβαίωσης password_verify(), η οποία ελέγχει αν το κρυπτογραφημένο αποτέλεσμα ταιριάζει με το αποθηκευμένο hash. Αν τα στοιχεία που εισάγει ο χρήστης στη φόρμα σύνδεσης είναι ορθά και αντιστοιχούν σε κάποιον χρήστη, ο έλεγχος θα επιτρέψει στο χρήστη να μεταφερθεί στη σελίδα του βασικού μενού αλλιώς θα εμφανιστεί το ανάλογο μήνυμα λάθους.

maps.php

Ένα από τα βασικότερα αρχεία της ιστοσελίδας είναι το maps.php. Στο αρχείο αυτό εκτελούνται οι περισσότερες λειτουργίες σε σχέση με τους χάρτες αλλά αυτές

```

session_start(); // Starting Session
$error=''; // Variable To Store Error Message
if (isset($_POST['submit'])) {
    if (empty($_POST['username']) || empty($_POST['password'])) {
        echo "<SCRIPT LANGUAGE='JavaScript'>
            window.alert('You did not complete all of the required fields')
            window.location.href='index.php'
        </SCRIPT>";
    }
    else
    {
        // Define $username and $password
        $username=$_POST['username'];
        $password=$_POST['password'];

        // To protect MySQL injection for Security purpose
        $username = stripslashes($username);
        $password = stripslashes($password);

        try {
            $result = $dbh ->prepare("SELECT password,tou FROM users WHERE username = :username");
            $result->bindParam(':username', $username, PDO::PARAM_STR);
            $result->execute();
        }
        catch(PDOException $e) {
            echo "Error: " . $e->getMessage();
        }
        $row = $result->fetch(PDO::FETCH_ASSOC);
        if ( ! $row ){
            echo "<SCRIPT LANGUAGE='JavaScript'>
                window.alert('Wrong username password combination. Please re-enter.')
                window.location.href='index.php'
            </SCRIPT>";
        }
        else {
            $hash = $row['password'];
            if (password_verify($password, $hash)) {
                $tou=$row['tou'];
                $_SESSION['username']=$username;           // Initializing Session
                $_SESSION['tou']=$tou;
                header("location:profile.php");
            }
            else {
                //its safer to not let the user know if the username or password was incorrect
                echo "<SCRIPT LANGUAGE='JavaScript'>
                    window.alert('Wrong username password combination. Please re-enter.')
                    window.location.href='index.php'
                </SCRIPT>";
            }
        }
    }
}

```

Σχήμα 4.28: login.php

που ξεχωρίζουν είναι τρεις. Πρώτα είναι η σχεδίαση μιας αποθηκευμένης περιοχής, η οποία παρουσιάζεται στην Εικόνα 4.29 με τη σχεδίαση ενός ορθογωνίου. Με τη βοήθεια του GoogleMaps και δίνοντας τις αποθηκευμένες συντεταγμένες η περιοχή σχηματίζεται στο χάρτη.

Επόμενη έρχεται η σχεδίαση νέας περιοχής, Εικόνα 4.30. Διαλέγοντας ένα από τα εικονίδια που παρουσιάστηκαν στα προηγούμενα υποκεφάλαια, στην επάνω μεριά του χάρτη, ξεκινάει η σχεδίαση. Μόλις τελειώσει η σχεδίαση, εμφανίζεται ένα παράθυρο με μία φόρμα ως προς συμπλήρωση με τις πληροφορίες της νέα περιοχής. Με το "SaveRegion" αποθηκεύεται στη Βάση. Τέλος, στην Εικόνα 4.30, παρουσιάζ-

```
if ( type[i] == 2 ){ //rectangle
// Construct the rectangle.
var srlatne = parseFloat(coo[1]);
var srlngne = parseFloat(coo[2]);
var srlatsw = parseFloat(coo[3]);
var srlngsw = parseFloat(coo[4]);

var conRectangle = new google.maps.Rectangle({
html: contentString,
id: place_id[i],
strokeColor: "#FF0000",
strokeOpacity: 0.8,
strokeWeight: 2,
fillColor: "#FF0000",
fillOpacity: 0.35 ,
editable: true,
bounds: { north: srlatne, south: srlatsw, east: srlngne, west: srlngsw }
});
conRectangle.setMap(map);
```

Σχήμα 4.29: Σχεδίαση Αποθηκευμένης Περιοχής - maps.php

```
var drawingManager = new google.maps.drawing.DrawingManager({
drawingControl: true,
drawingControlOptions: {
position: google.maps.ControlPosition.TOP_CENTER,
drawingModes: [
google.maps.drawing.OverlayType.CIRCLE,
google.maps.drawing.OverlayType.RECTANGLE
]
}
});

drawingManager.setMap(map);

google.maps.event.addListener(drawingManager, "rectanglecomplete", function(rectangle) {

var ne = rectangle.getBounds().getNorthEast();
var sw = rectangle.getBounds().getSouthWest();
var nelat = ne.lat();
var nelng = ne.lng();
var swlat = sw.lat();
var swlng = sw.lng();

var coordsrec = ":" + nelat.toFixed(6) + ":" + nelng.toFixed(6) + ":" + swlat.toFixed(6) + ":" + swlng.toFixed(6);
console.log(coordsrec);

contentsr = '<form action="SaveData.php" method="POST" enctype="multipart/form-data"><input type="hidden" name="coords" type="text" size="50" value="'+
coordsrec+'"/><b>Region Name : </b><br/><input type="text" size="20" name="region_name"/><input type="hidden" name="region_type" value="2"><br/>'+
'<b>Description : </b><br/><textarea name="region_desc" cols="20" rows="3"/></textarea><br/><b>Image : </b><br/><input type="file" name="myimage"/>'+
'<br/><center><br/><input type="submit" value="Save Region" name="save_region"></center></form>';

var boundsr = new google.maps.LatLng(nelat(), nelng());

infoWindow.setContent(contentsr);
infoWindow.setPosition(boundsr);
drawingManager.setDrawingMode(null);
infoWindow.open(map);

});
```

Σχήμα 4.30: Σχεδίαση Νέας Περιοχής - maps.php

ζονται δύο κομμάτια κώδικα. Και τα δύο "ακούνε" σε συμβάντα (event). Το πρώτο ("click") αναφέρεται στο συμβάν του να πατήσει κάποιος μια περιοχή στο χάρτη. Με το πάτημα δίνεται εντολή να εμφανιστεί το παράθυρο με τις πληροφορίες της. Το δεύτερο ("bounds_changed") "ακούει" στην επεξεργασία των συντεταγμένων αποθηκευμένης περιοχής, και μόλις ολοκληρωθεί η επεξεργασία, ο χρήστης ερωτάται αν θέλει να αποθηκεύσει τις νέες συντεταγμένες. Αν ναι, αυτές αποθηκεύονται στη βάση, αλλιώς παραμένουν αποθηκευμένες οι παλιές.

Κεφάλαιο 4. Λειτουργίες Ιστοχώρου

```
google.maps.event.addListener(conRectangle, "click",
function (event) {
    var contentString = this.html;
    infoWindow.setContent(contentString);
    infoWindow.setPosition(event.latLng);
    infoWindow.open(map);
});

google.maps.event.addListener(conRectangle, "bounds_changed",
function (event) {
    console.log('Bounds changed.');
```

```
    var pid = this.id;
    var ne = this.getBounds().getNorthEast();
    var sw = this.getBounds().getSouthWest();
    var nelat = ne.lat();
    var nelng = ne.lng();
    var swlat = sw.lat();
    var swlng = sw.lng();

    var upcoordsrec = ":" + nelat.toFixed(6) + ":" + nelng.toFixed(6) + ":" + swlat.toFixed(6) + ":" + swlng.toFixed(6);
    console.log(upcoordsrec);

    contentstr = '<form action="SaveData.php" method="POST" enctype="multipart/form-data"><input type="hidden" name="pid" type="text" size="50" value="'+pid+'"/>'+
    '<input type="hidden" name="upcoords" type="text" size="50" value="'+upcoordsrec+'"/><input type="hidden" name="region_type" value="2"><center><br/>'+
    '<input type="submit" value="Update Coordinates" name="update_coords"></center></form>';

    var boundsr = new google.maps.LatLng(nelat(), nelng());
    infoWindow.setContent(contentstr);
    infoWindow.setPosition(boundsr);
    drawingManager.setDrawingMode(null);
    infoWindow.open(map);
}
);
```

Σχήμα 4.31: Επιλογή Περιοχής και Αλλαγή Συντεταγμένων - maps.php

readRecords.php

```
//Remove previous chosen shapes
if (cc!=0){
    if (prevshapetype == 2){
        conRectangle2.setMap(null);
    }else{
        conCircle2.setMap(null);
    }
}

if ( type[id] == 2 ){ //rectangle

    // Construct the rectangle.
    var srlatne = parseFloat(coo[1]);
    var srlngne = parseFloat(coo[2]);
    var srlatsw = parseFloat(coo[3]);
    var srlngsw = parseFloat(coo[4]);

    conRectangle2 = new google.maps.Rectangle({
        html: contentString1,
        strokeColor: "#0000ff",//"#FF0000",
        strokeOpacity: 0.8,
        strokeWeight: 2,
        fillColor: "#0000ff",//"#FF0000",
        fillOpacity: 0.35 ,
        bounds: {
            north: srlatne,
            south: srlatsw,
            east: srlngne,
            west: srlngsw
        }
    });
    conRectangle2.setMap(map);

    var infowpos = new google.maps.LatLng(srlatne, srlngne);
    //console.log("infowpos " +infowpos);
    infoWindow.setContent(contentString1);
    infoWindow.setPosition(infowpos);

    infoWindow.open(map);
    cc++;
    prevshapetype = 2;
    google.maps.event.addListener(conRectangle2, "click",
function (event) {
    var contentString1 = this.html;
    infoWindow.setContent(contentString1);
    infoWindow.setPosition(event.latLng);

    infoWindow.open(map);
}
);
```

Σχήμα 4.32: Χρωματισμός Επιλεγμένης Εγγραφής στο Χάρτη - readRecords.php

Ένα ακόμα ενδιαφέρον κομμάτι κώδικα που αν και τα αποτελέσματά του φαίνονται στο maps.php, εκτελείται στη σελίδα readRecords.php, παρουσιάζεται στην Εικόνα 4.32. Χάρη σε αυτό, όταν ο χρήστης πατάει μία εγγραφή του πίνακα "Saved Regions", επισημαίνεται με μπλε χρώμα στο χάρτη δίπλα και εμφανίζεται και το παράθυρο πληροφοριών της μαζί με τα κουμπιά επεξεργασίας της. Το κομμάτι στην αρχή που ελέγχει αν το (cc!=0) είναι αυτό που αν πριν είχε επιλεγεί άλλη περιοχή την σβήνει ώστε να μην υπάρχουν δυο ή περισσότερες περιοχές επισημασμένες ταυτόχρονα και μπερδεύουν το χρήστη.

savedata.php

```
if(empty($_FILES["myimage"]["name"])){
    $upload_image = null;
    $ihash = null;
}else{
    $upload_image = $_FILES["myimage"]["name"];
    $folder = "placeimgs/";

    //Checks if the ihash already exists then creates another
    include 'locations.php';

    $i = 0; $j = 0;

    $ihash = md5(rand());

    while( $j == 0 ){
        $k = 0;
        for($i=0; $i<count($pfimage); $i++){
            if ( $ihash == $pfimage[$i] ){
                $k++;
            }
        }
        if ($k == 0){
            $j = 1;
        } else {
            $ihash = hash_file( 'md5', rand() );
        }
    }
    move_uploaded_file( $_FILES["myimage"]["tmp_name"] , $folder.$ihash );
}
```

Σχήμα 4.33: Αποθήκευση Εικόνας - savedata.php

Εδώ, Εικόνα 4.33, βλέπουμε τον τρόπο που αποθηκεύεται η εικόνα που δίνεται από τον χρήστη. Το αρχείο αποθηκεύεται στον διακομιστή, με όνομα ένα hash, το οποίο ελέγχεται να μην υπάρχει ήδη, ώστε να μην υπάρξουν δύο διαφορετικές εικόνες με το ίδιο όνομα.

```
if(isset($_POST['submit_pass']))
{
    if(!empty($_POST['password']) AND !empty($_POST['cpassword']))
    {
        $password = $_POST['password'];
        $cpassword = $_POST['cpassword'];

        if($password !== $cpassword){
            echo("<SCRIPT LANGUAGE='JavaScript'>
                window.alert('Passwords do NOT match')
                window.location.href='changepassword.php'
            </SCRIPT>");
        }
        elseif( mb_strlen( $password ) < 6 ){
            echo("<SCRIPT LANGUAGE='JavaScript'>
                window.alert('Password must be more than 6 characters')
                window.location.href='changepassword.php'
            </SCRIPT>");
        }
        else{
            $hash = password_hash($password, PASSWORD_DEFAULT); //bcrypt hashing
            try {
                $query = $dbh ->prepare("UPDATE users SET password=:password WHERE user_id = :userid");
                $query->bindParam(':password', $hash, PDO::PARAM_STR);
                $query->bindParam(':userid', $userid, PDO::PARAM_INT);
                $query->execute();
            }
            catch(PDOException $e) {
                echo "Error: " . $e->getMessage();
            }

            if ( $query ){
                header('Location: accountsettings.php');
            } else {
                echo("<SCRIPT LANGUAGE='JavaScript'>
                    window.alert('Error: The new password was NOT saved!')
                    window.location.href='accountsettings.php'
                </SCRIPT>");
            }
        }
    }
}
else
{
    echo("<SCRIPT LANGUAGE='JavaScript'>
        window.alert('You did not complete all of the required fields')
        window.location.href='changepassword.php'
    </SCRIPT>");
}
}
```

Σχήμα 4.34: savepass.php

savepass.php

Στην Εικόνα 4.34 βλέπουμε πώς γίνεται η αλλαγή κωδικού του χρήστη. Ελέγχεται αν ο κωδικός είναι ίδιος με τον κωδικό επαλήθευσης και αν είναι μεγαλύτερος των 6 ψηφίων. Αν ναι δημιουργείται ένα hash για τον κωδικό, το οποίο αποθηκεύεται στη βάση.

saveuser.php

Με το saveuser.php γίνεται η αποθήκευση του χρήστη μετά την καρτέλα εγγραφής.

Στην Εικόνα 4.35 γίνονται οι έλεγχοι, αν τα στοιχεία που δόθηκαν είναι όπως πρέπει και μήπως ο χρήστης υπάρχει ήδη στη βάση.

```
1 <?php
2 /* user file */
3 require 'connect.php'; //not sure if we need it
4 include('https.php'); //Includes the control file that always redirects to https
5
6 if(isset($_POST['submit_user']))
7 {
8     session_start(); //starting the session
9
10    if(!empty($_POST['username']) AND !empty($_POST['password']) AND !empty($_POST['cpassword'])
11        AND !empty($_POST['name']) AND !empty($_POST['surname']) AND !empty($_POST['email']))
12    {
13        $username= trim($_POST[username]);
14        $email= trim($_POST[email]);
15
16        $remove[] = "";
17        $remove[] = "'";
18        $username=str_replace( $remove, "", $username );
19        $email=str_replace( $remove, "", $email );
20
21        $password = $_POST['password'];
22        $cpassword = $_POST['cpassword'];
23        try {
24            $sqlquery = $dbh ->prepare("SELECT 1 FROM users WHERE email = :email");
25            $sqlquery->bindParam(':email', $email, PDO::PARAM_STR);
26            $sqlquery->execute();
27        }
28        catch(PDOException $e) {
29            echo "Error: " . $e->getMessage();
30        }
31        $row = $sqlquery->fetch(PDO::FETCH_ASSOC);
32
33        try {
34            $sqlquery2 = $dbh ->prepare("SELECT 1 FROM users WHERE username = :username");
35            $sqlquery2->bindParam(':username', $username, PDO::PARAM_STR);
36            $sqlquery2->execute();
37        }
38        catch(PDOException $e) {
39            echo "Error: " . $e->getMessage();
40        }
41        $row2 = $sqlquery2->fetch(PDO::FETCH_ASSOC);
42    }
```

Σχήμα 4.35: Έλεγχος - saveuser.php

```
$hash = password_hash($password, PASSWORD_DEFAULT); //bcrypt hashing

$name=trim($_POST[name]);
$surname=trim($_POST[surname]);
$address=trim($_POST[address]);
$city=trim($_POST[city]);
$phone=trim($_POST[phone]);
$name=str_replace( $remove, "", $name );
$surname=str_replace( $remove, "", $surname );
$address=str_replace( $remove, "", $address );
$city=str_replace( $remove, "", $city );
$phone=str_replace( $remove, "", $phone );
$tou='2'; //2 at tou because they are not administrators, only the admin can change the tou

try {
    $query = $dbh ->prepare("INSERT INTO users (username,password,name,surname,email,address,city,phone,tou) VALUES (?, ?, ?, ?, ?, ?, ?, ?)");
    $query->execute(array($username,$hash,$name,$surname,$email,$address,$city,$phone,$tou));
}
catch(PDOException $e) {
    echo "Error: " . $e->getMessage();
}

if (($query){ //=== TRUE) {
    header('Location: index.php');
} else {
    echo ("<SCRIPT LANGUAGE='JavaScript'>
        window.alert('Error: A new record was NOT created!')
        window.location.href='signup.php'
    </SCRIPT>");
}
```

Σχήμα 4.36: Αποθήκευση - saveuser.php

Αν περάσει όλους τους ελέγχους τότε στην Εικόνα 4.36 γίνεται η αποθήκευση των στοιχείων του χρήστη.

session.php

```
1 <?php
2 /* user file */
3 require 'connect.php';
4 include('https.php'); //Includes the control file that always redirects to https
5
6 if(!isset($_SESSION))
7 {
8     session_start(); //Starting Session if it has not already started
9 }
10
11 // Storing Session
12 $user_check=$_SESSION['username'];
13
14 try {
15     $ses_sql = $dbh->prepare("SELECT username, tou, user_id FROM users WHERE username = :username");
16     $ses_sql->bindParam(':username', $user_check, PDO::PARAM_STR);
17     $ses_sql->execute();
18 }
19 catch(PDOException $e) {
20     echo "Error: " . $e->getMessage();
21 }
22
23 $row = $ses_sql->fetch(PDO::FETCH_ASSOC);
24
25 $login_session=$row['username'];
26 $admin_check=$row['tou'];
27 $userid=$row['user_id'];
28 if(!isset($login_session)){
29     $dbh = null;
30     header('Location: index.php'); // Redirecting To Home Page
31 }
32
33 ?>
34
```

Σχήμα 4.37: session.php

Το αρχείο που φαίνεται στην Εικόνα 4.37, ελέγχει τα στοιχεία σύνδεσης του χρήστη. Εάν η διαπίστευση του χρήστη είναι επιτυχής, το σενάριο ελέγχου θα επιτρέψει στο χρήστη να ανακατευθυνθεί στην επόμενη ιστοσελίδα, διαφορετικά τον επιστρέφει στη σελίδα σύνδεσης. Αυτό παρέχει ασφάλεια, καθώς αποτρέπει κάποιον ανώνυμο χρήστη να περιηγηθεί στις ιστοσελίδες του συστήματος χωρίς να είναι συνδεδεμένος. Το αρχείο αυτό γίνεται include ή require από κάθε αρχείο που εκτελεί ενέργειες από τους απλούς χρήστες.

4.5 Σύνοψη Κεφαλαίου

Στο κεφάλαιο αυτό περιγράφηκε αναλυτικά ο τρόπος λειτουργίας του Ιστοχώρου. Αναλύθηκε το γραφικό του περιβάλλον και όλες οι ενέργειες που μπορούν να εκτελέσουν οι χρήστες, ανάλογα με τα δικαιώματα πρόσβασης τους. Τέλος έγινε

μία ανάλυση σε μερικά σημαντικά σημεία του κώδικα και δόθηκαν τα μετρικά της Ιστοσελίδας.

Κεφάλαιο 5

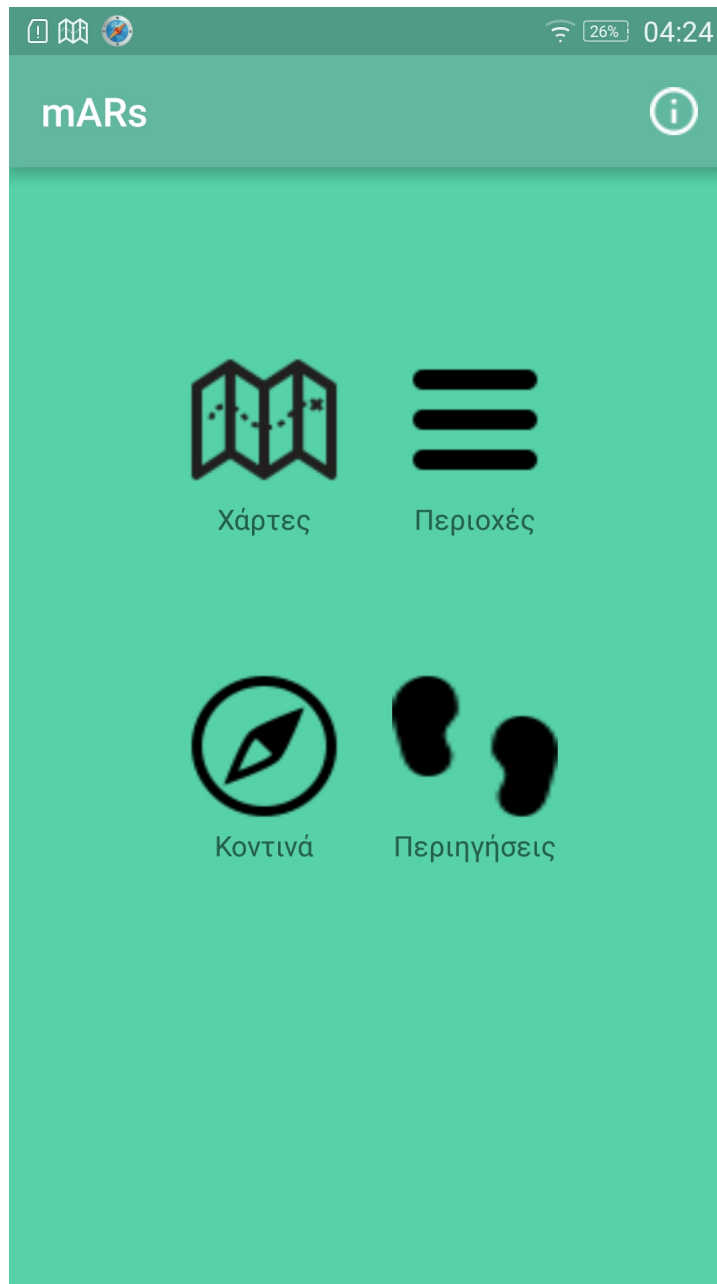
Λειτουργίες Κινητής Εφαρμογής

Στο αυτό το κεφάλαιο παρουσιάζεται η λειτουργία και υλοποίηση της Εφαρμογής Android. Η εφαρμογή έγινε με σκοπό την γρήγορη και εύκολη αναζήτηση τουριστικών αξιοθέατων. Για τη λειτουργία της απαιτείται να υπάρχει σύνδεση στο διαδίκτυο. Με τη βοήθεια του Global Positioning System (GPS) βρίσκει την τοποθεσία του χρήστη ώστε να του εμφανίζει κοντινά σημεία ενδιαφέροντος και τις αποστάσεις από αυτά. Στη συνέχεια, περιγράφονται αναλυτικά οι λειτουργίες χρήστη, τα δικαιώματα χρήσης και τέλος γίνεται ανάλυση σημαντικών της κομματιών κώδικα.

5.1 Λειτουργίες Χρήστη

Στην παρούσα εφαρμογή δεν υπάρχουν κατηγορίες χρηστών, όλοι οι χρήστες θεωρούνται ανώνυμοι. Αυτό συμβαίνει γιατί η εφαρμογή δεν δέχεται εισαγωγή δεδομένων παρά μόνο έξοδό τους, έτσι είναι ασφαλές να μην έχουμε ιεραρχία χρηστών. Έτσι μπορεί ο κάθε ενδιαφερόμενος εύκολα και άμεσα να κάνει χρήση της, χωρίς εγγραφές και άλλες καθυστερήσεις. Παρακάτω, αναλύονται οι λειτουργίες.

5.1.1 Βασικό Μενού



Σχήμα 5.1: Οθόνη Βασικού Μενού

Μπαίνοντας στην εφαρμογή, το πρώτο πράγμα που βλέπει ο χρήστης είναι το Βασικό Μενού. Όπως παρουσιάζεται στην Εικόνα 5.1, το μενού αποτελείται από 4 κουμπιά: τους Χάρτες, τις Περιοχές, τις Κοντινές Περιοχές και τις Περιηγήσεις. Επίσης επάνω δεξιά υπάρχει το κουμπί με το "i", από το οποίο ο χρήστης μπορεί να μεταβεί στην Οθόνη "Σχετικά με". Στη συνέχεια αναλύονται οι λειτουργίες τις κάθε μίας από τις παραπάνω επιλογές.

5.1.2 Χάρτες



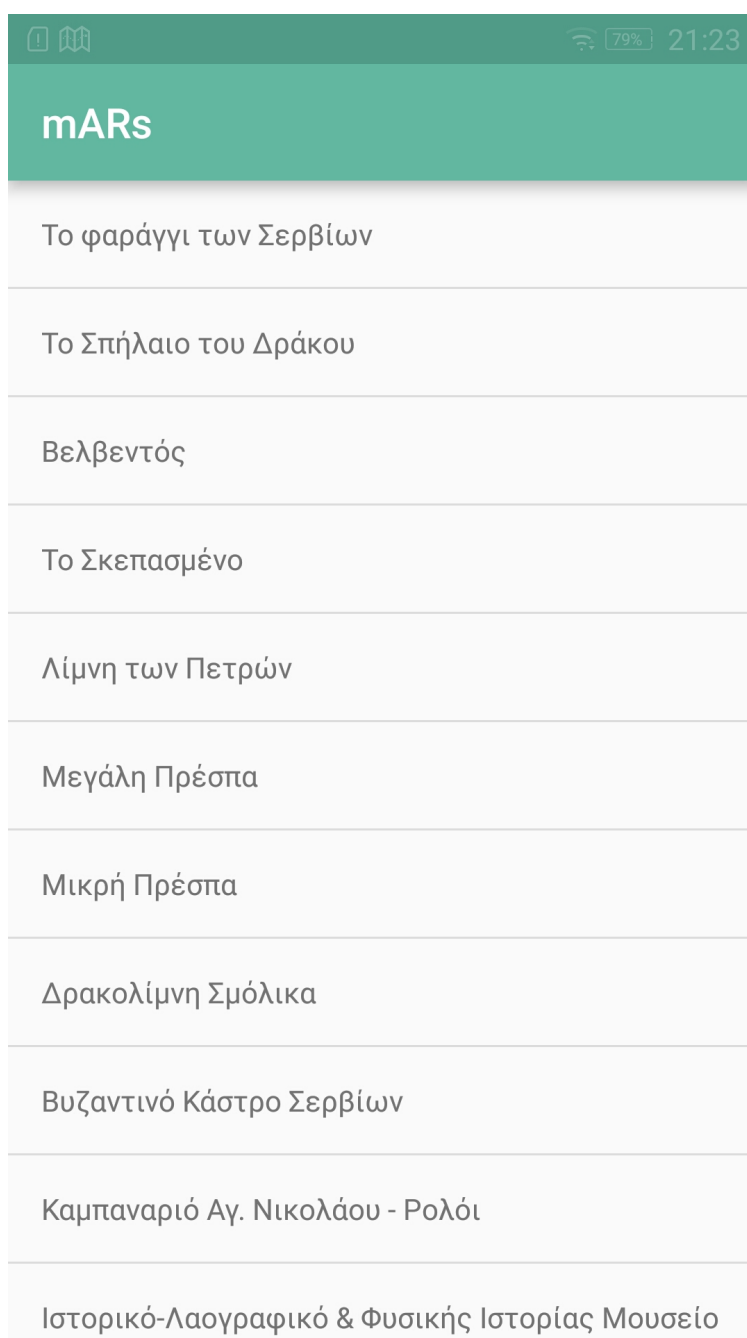
(α) Οθόνη Χάρτες



(β) Αναζήτηση Διεύθυνσης

Πατώντας τους Χάρτες ο χρήστης μεταφέρεται στην ανάλογη οθόνη, η οποία φαίνεται στην Εικόνα 5.2α'. Το μεγαλύτερο μέρος της οθόνης το καταλαμβάνει ένας χάρτης ο οποίος έχει σημειωμένες όλες τις αποθηκευμένες περιοχές. Υπάρχει επίσης ένα πεδίο αναζήτησης όπου μπορούν να αναζητηθεί κάποια διεύθυνση ή συντεταγμένες και να του την εμφανίσει στο χάρτη, Εικόνα 5.2β'. Μετά την πραγματοποίηση της αναζήτησης, κάτω από το χάρτη εμφανίζει την αναλυτική διεύθυνση και τις συντεταγμένες του μέρους που έχει αναζητήσει ο χρήστης.

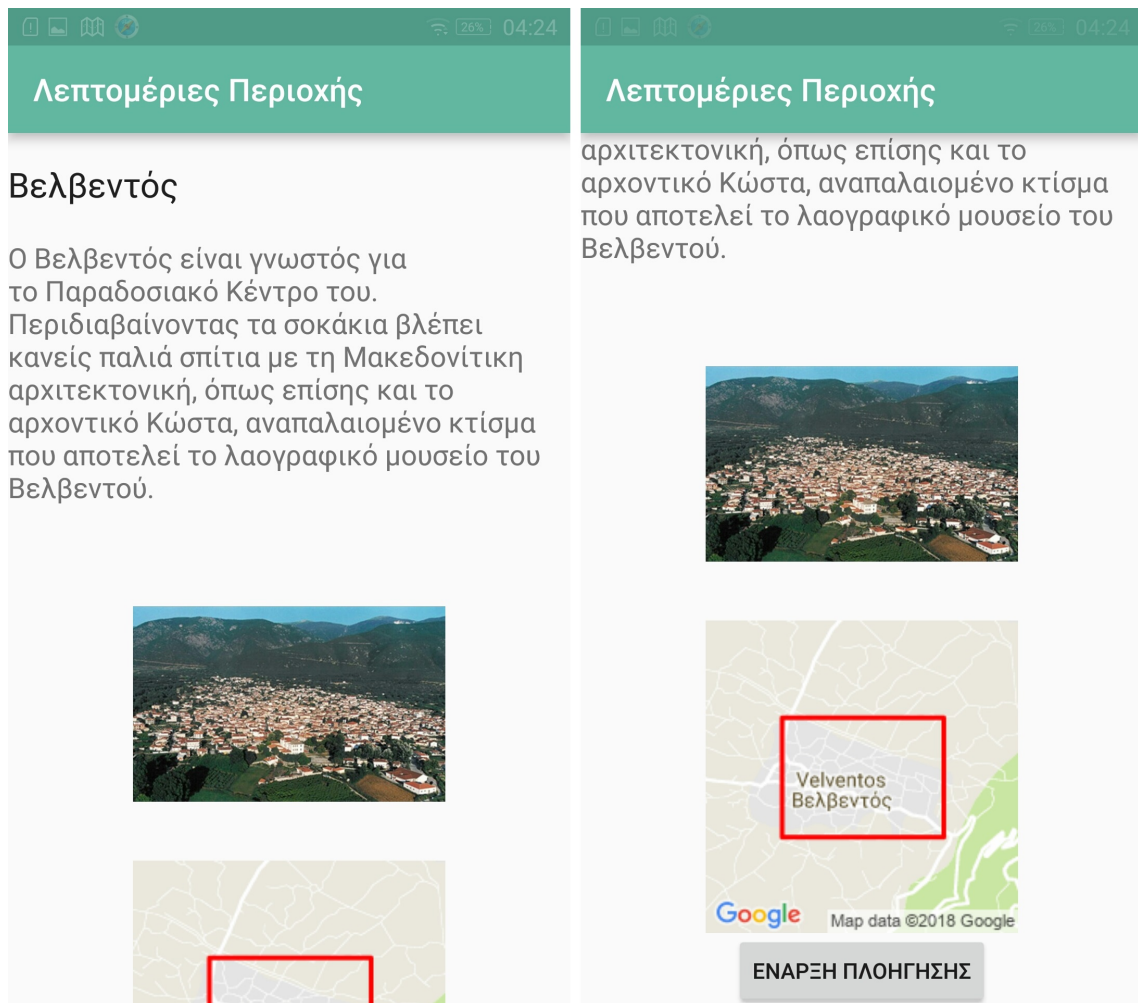
5.1.3 Περιοχές



Σχήμα 5.3: Οθόνη Περιοχές

Επιλέγοντας από το μενού τις Περιοχές, εμφανίζεται στο χρήστη μία λίστα με τα ονόματα όλων των αποθηκευμένων Περιοχών, Εικόνα 5.3. Πατώντας σε κάποια περιοχή γίνεται μεταφορά σε μία οθόνη με αναλυτικές πληροφορίες για αυτή.

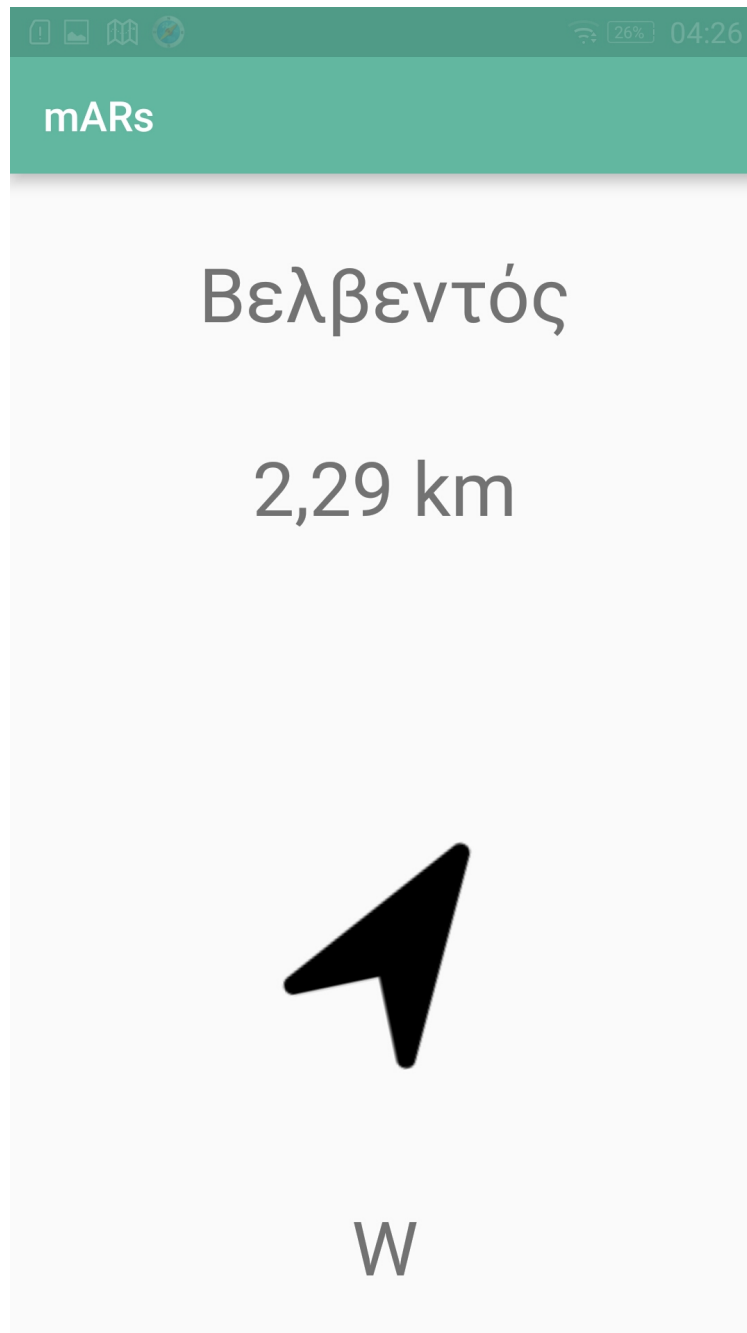
5.1.4 Λεπτομέρειες Περιοχής



Σχήμα 5.4: Οθόνη Λεπτομέρειες Περιοχής

Στην οθόνη Λεπτομέρειες Περιοχής, όπως βλέπουμε στην Εικόνα 5.4, οι πληροφορίες που δίνονται είναι το Όνομα, η Περιγραφή, η Εικόνα, αν υπάρχει, και τέλος ένας μικρός χάρτης που δείχνει την περιοχή. Μετά τα στοιχεία υπάρχει το κουμπί "ΕΝΑΡΞΗ ΠΛΟΗΓΗΣΗΣ". Πατώντας το ξεκινάει η πλοήγηση του χρήστη στην επιλεγμένη περιοχή.

5.1.5 Πλοήγηση σε Περιοχή



Σχήμα 5.5: Οθόνη Πλοήγηση σε Περιοχή

Αν ο χρήστης επιλέξει να ξεκινήσει την πλοήγηση στην περιοχή, του εμφανίζεται η οθόνη της Εικόνας 5.5. Στην οθόνη αυτή φαίνεται το όνομα της περιοχής, η απόσταση που απομένει, και μία πυξίδα που δείχνει την κατεύθυνση που πρέπει να πάρει ο χρήστης. Κάτω από την πυξίδα εμφανίζεται και προς τα ποια διεύθυνση (N: Βορράς, W: Δύση, S: Νότος, E: Ανατολή) κοιτάει η κινητή συσκευή.

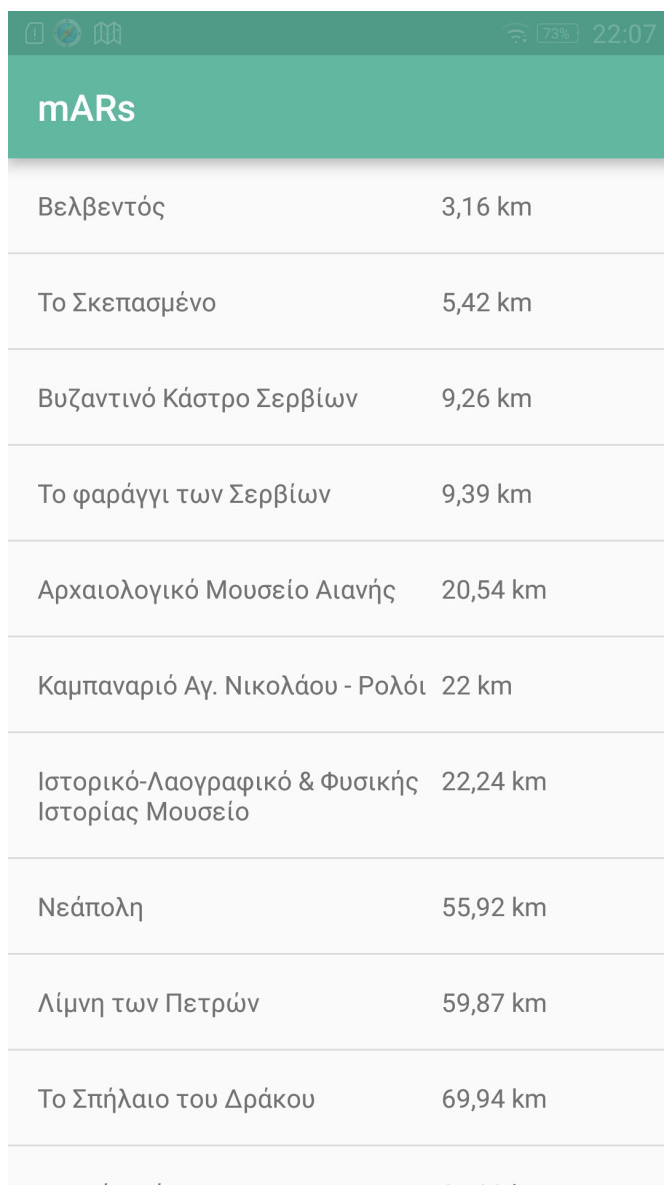
5.1.6 Άφιξη σε Περιοχή



Σχήμα 5.6: Οθόνη Άφιξη σε Περιοχή

Με το που ο χρήστης φτάσει σε μια περιοχή η οθόνη πλοήγησης αλλάζει και τον μεταφέρει στην Οθόνη Άφιξης, η οποία φαίνεται στην Εικόνα 5.6. Σε αυτή την οθόνη εμφανίζεται το μήνυμα, "Φτάσατε σε" και το όνομα της περιοχής στην οποία έφτασε. Αμέσως μετά υπάρχει το κουμπί "ΠΛΗΡΟΦΟΡΙΕΣ ΠΕΡΙΟΧΗΣ", που αν πατηθεί μεταφέρει τον χρήστη στην οθόνη Λεπτομέρειες Περιοχής και το κουμπί "ΕΠΙΣΤΡΟΦΗ ΣΤΟ ΜΕΝΟΥ", το οποίο μεταφέρει τον χρήστη στο αρχικό Μενού.

5.1.7 Κοντινές Περιοχές

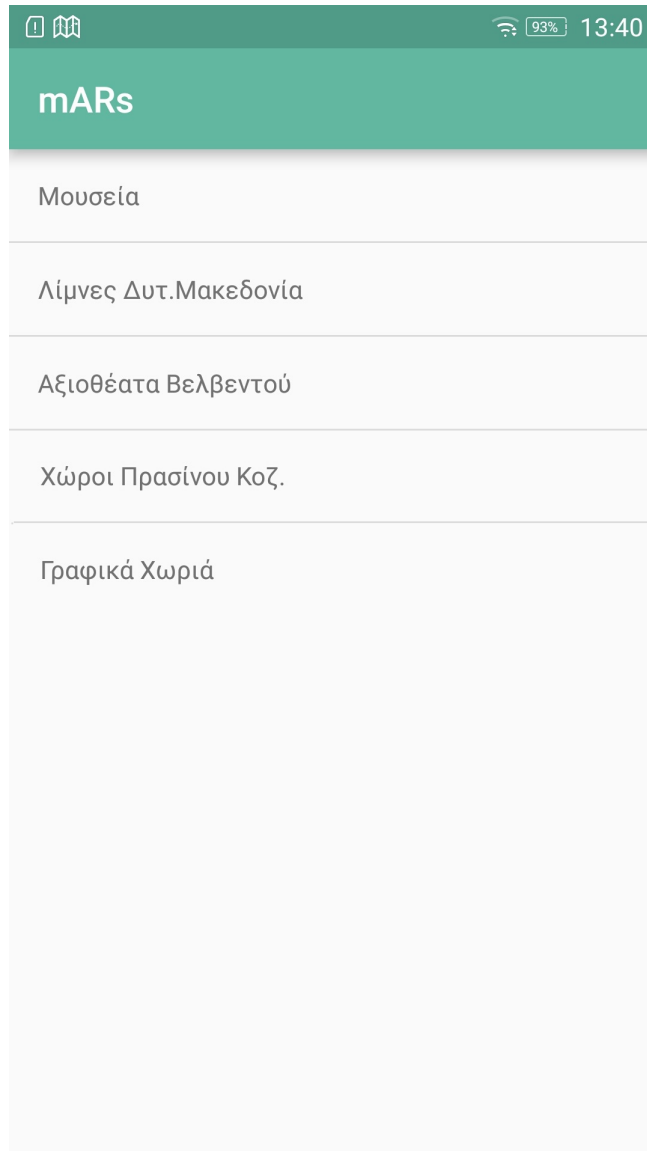


mARs	
Βελβεντός	3,16 km
Το Σκεπασμένο	5,42 km
Βυζαντινό Κάστρο Σερβίων	9,26 km
Το φαράγγι των Σερβίων	9,39 km
Αρχαιολογικό Μουσείο Αιανής	20,54 km
Καμπαναριό Αγ. Νικολάου - Ρολόι	22 km
Ιστορικό-Λαογραφικό & Φυσικής Ιστορίας Μουσείο	22,24 km
Νεάπολη	55,92 km
Λίμνη των Πετρών	59,87 km
Το Σπήλαιο του Δράκου	69,94 km
Μουσείο Πλάσματος	67,88 km

Σχήμα 5.7: Οθόνη Κοντινές Περιοχές

Επιλέγοντας από το μενού την επιλογή Κοντινές Περιοχές, ο χρήστης μεταφέρεται σε μία οθόνη με μία λίστα με τα ονόματα όλων των αποθηκευμένων Περιοχών, που όπως φαίνεται στην Εικόνα 5.7, είναι ταξινομημένα από το κοντινότερο στο πιο μακρινό. Επίσης δίπλα από το όνομα κάθε περιοχής υπάρχει και η απόσταση της από το σημείο που βρισκόμαστε. Πατώντας σε κάποια περιοχή εμφανίζεται η οθόνη "Λεπτομέρειες Περιοχής" που αναφέρθηκε παραπάνω με τις ανάλογες πληροφορίες.

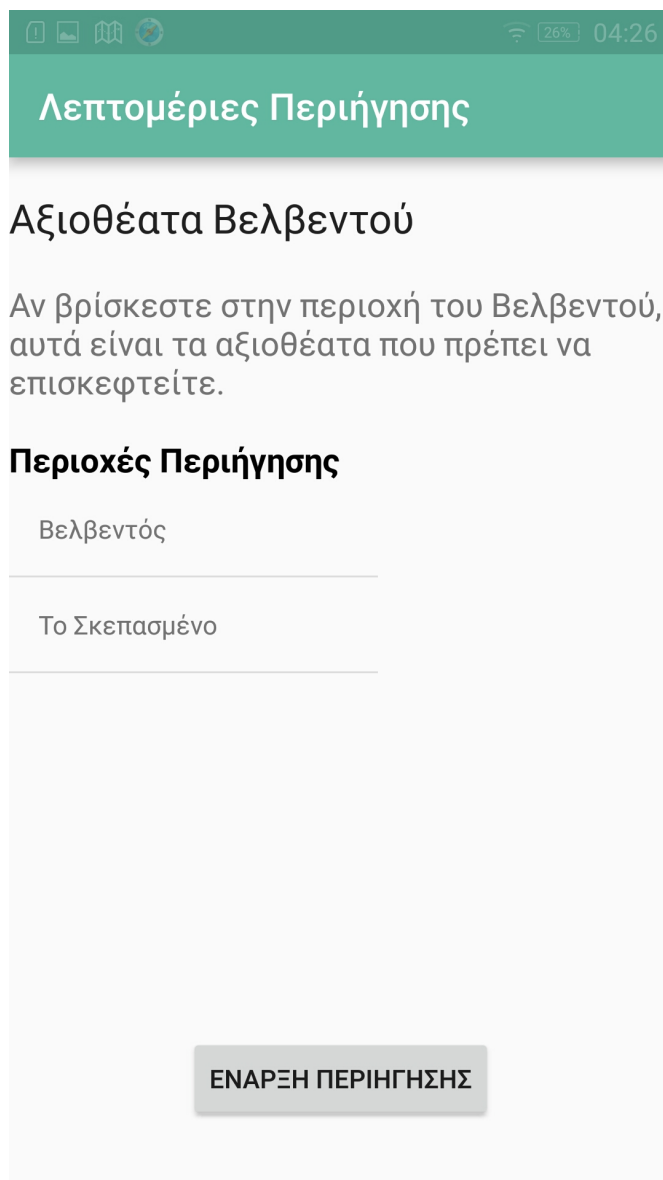
5.1.8 Περιηγήσεις



Σχήμα 5.8: Οθόνη Περιηγήσεις

Με την επιλογή των Περιηγήσεων, ο χρήστης μεταφέρεται σε μια οθόνη με λίστα με τα ονόματα όλων των αποθηκευμένων Περιηγήσεων, όπως στην Εικόνα 5.8. Πατώντας στο όνομα κάποιας περιήγησης παρουσιάζεται μία οθόνη με αναλυτικές πληροφορίες για αυτή.

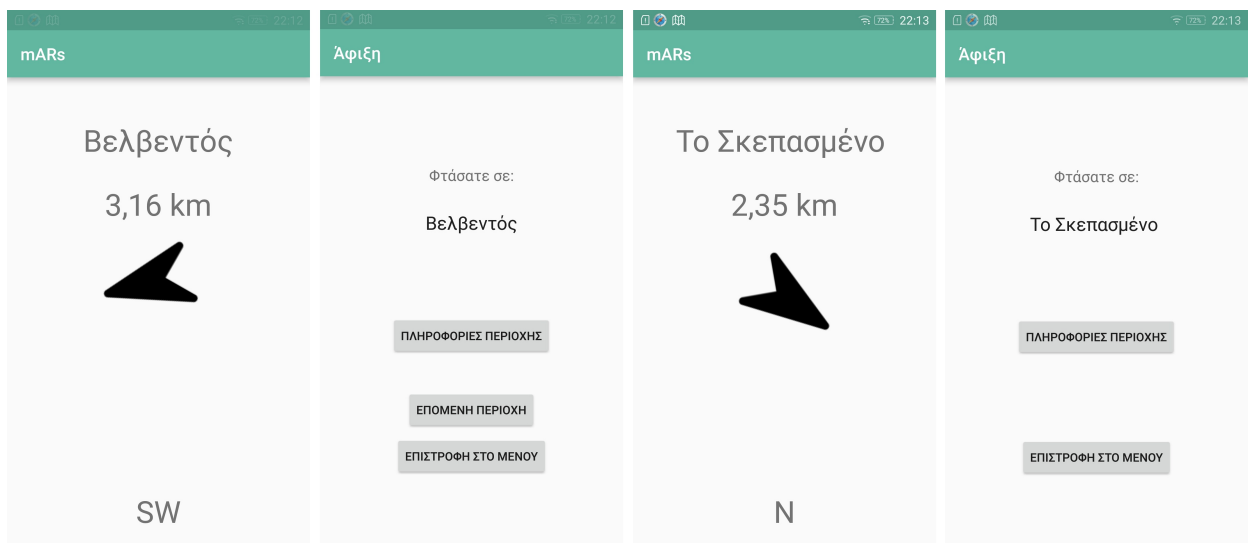
5.1.9 Λεπτομέρειες Περιήγησης



Σχήμα 5.9: Οθόνη Λεπτομέρειες Περιήγησης

Με τη μεταφορά στις Λεπτομέρειες Περιήγησης, Εικόνα 5.9, εμφανίζονται οι πληροφορίες Όνομα, Περιγραφή και τα ονόματα των περιοχών που περιλαμβάνει η Περιήγηση. Υπάρχει η δυνατότητα ο χρήστης να πατήσει πάνω σε κάποιο από τα ονόματα των περιοχών και θα μεταφερθεί στην οθόνη "Λεπτομέρειες Περιοχής" με τις πληροφορίες της εκάστοτε περιοχής. Μετά τα στοιχεία υπάρχει το κουμπί "ΕΝΑΡΞΗ ΠΛΟΗΓΗΣΗΣ". Πατώντας το ξεκινάει η πλοήγηση του χρήστη στην Περιήγηση.

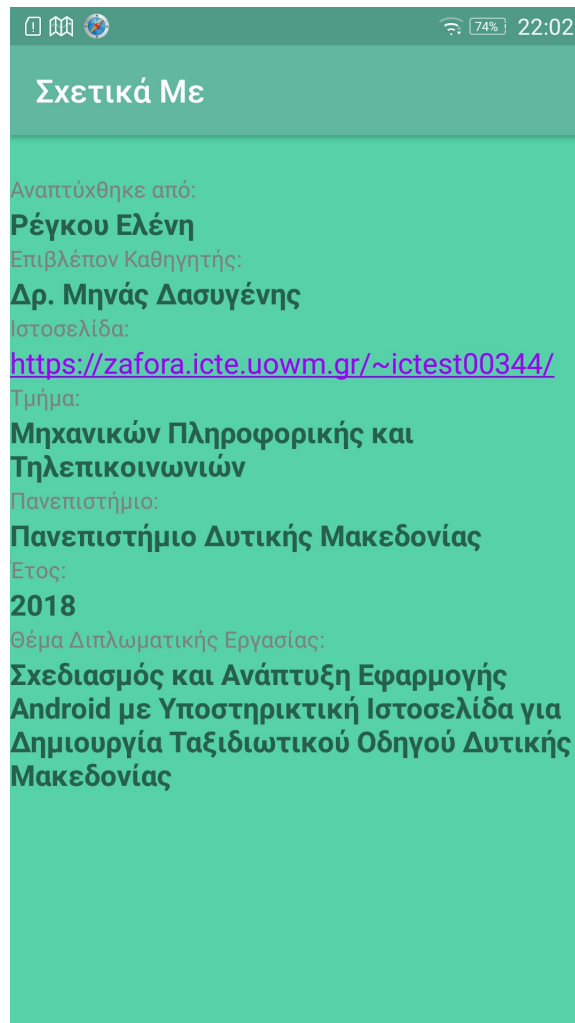
5.1.10 Πλοήγηση σε Περιήγηση



Σχήμα 5.10: Οθόνη Πλοήγηση σε Περιήγηση

Η Πλοήγηση στην Περιήγηση μοιάζει αρκετά με την πλοήγηση σε περιοχή. Οι περιοχές της περιήγησης έχουν μπει σε σειρά, από την κοντινότερη στη πιο μακρινή, και η πλοήγηση στην περιήγηση ξεκινάει από την οθόνη πλοήγησης της κοντινότερης περιοχής από αυτές που την αποτελούν. Μόλις ο χρήστης φτάσει στην πρώτη περιοχή του εμφανίζεται η οθόνη Άφιξης όπως και παραπάνω. Μόνο που τώρα εκτός από τα κουμπιά "ΠΛΗΡΟΦΟΡΙΕΣ ΠΕΡΙΟΧΗΣ" και "ΕΠΙΣΤΡΟΦΗ ΣΤΟ ΜΕΝΟΥ", υπάρχει και το κουμπί "ΕΠΟΜΕΝΗ ΠΕΡΙΟΧΗ". Έτσι ο χρήστης, όταν είναι έτοιμος, μπορεί πατώντας το να οδηγηθεί στην οθόνη πλοήγησης της αμέσως επόμενης περιοχής, και ούτω καθεξής. Όταν φτάσει στην τελευταία περιοχή το κουμπί "ΕΠΟΜΕΝΗ ΠΕΡΙΟΧΗ" παύει να εμφανίζεται στην οθόνη Άφιξης αφού η Περιήγηση δεν έχει άλλη περιοχή για πλοήγηση. Όλα αυτά παρουσιάζονται στην Εικόνα 5.10.

5.1.11 Σχετικά Με



Σχήμα 5.11: Οθόνη Σχετικά Με

Από το αρχικό μενού υπάρχει και η επιλογή για την εμφάνιση πληροφοριών σχετικά με την ανάπτυξη της εφαρμογής. Πατώντας το (i) πάνω δεξιά, και στη συνέχεια το Σχετικά Με, εμφανίζεται η οθόνη με τις πληροφορίες, Εικόνα 5.11. Τα στοιχεία που δίνονται είναι ποιος ανέπτυξε τον ιστότοπο, το όνομα του επιβλέπων, η συνοδευτική ιστοσελίδα, το τμήμα, το πανεπιστήμιο, το έτος και το θέμα της διπλωματικής εργασίας.

5.2 Ανάλυση Σημαντικών Σημείων Κώδικα Εφαρμογής

Στο σημείο αυτό, αναλύονται τα σημαντικότερα κομμάτια κώδικα που έπαιξαν ρόλο στην ανάπτυξη της κινητής Εφαρμογής.

navigation.java

```
@Override
protected void onPostExecute(String json) {
    try {
        JSONObject baseJsonResponse = new JSONObject(json); //Convert json String into a JSONObject
        // De-serialize the JSON string into an array of city objects
        JSONArray jsonArray = baseJsonResponse.getJSONArray( name: "allplaces");
        LatLng tap = new LatLng(nlatitude, nlongitude);
        String type = passedVarTn;
        int itype = Integer.parseInt(type);
        int place_id = Integer.parseInt(passedVarPid);
        String coords = passedVarCn;
        String[] coordstable = coords.split( regularExpression: ";" );

        if ( itype == 2 ){ // Rectangle
            // put coordinates in vertices array as LatLng(double lat, double lng) format
            // do Ray-Casting
            //if R-C result is odd then we're in the area, show notification!

            List<LatLng> vertices = new ArrayList<LatLng>();

            if (itype == 2) {
                double platx1 = Double.parseDouble(coordstable[1]);
                double plngy1 = Double.parseDouble(coordstable[2]);
                double platx2 = Double.parseDouble(coordstable[3]);
                double plngy2 = Double.parseDouble(coordstable[4]);

                vertices.add(new LatLng(platx1, plngy1));
                vertices.add(new LatLng(platx1, plngy2));
                vertices.add(new LatLng(platx2, plngy2));
                vertices.add(new LatLng(platx2, plngy1));
            } else {
                for (int j = 1; j < coordstable.length; j = j + 2) {
                    double platx = Double.parseDouble(coordstable[j]);
                    double plngy = Double.parseDouble(coordstable[j + 1]);
                    vertices.add(new LatLng(platx, plngy));
                }
            }

            boolean isitin = PolyUtil.containsLocation( tap, vertices, geodesic: false);
            if (isitin == true){
                Log.d( tag: "msg7", msg: "its in rec place with ID: " + place_id);
                Intent myIntent10 = new Intent( packageContext: NavigationActivity.this, ArrivalActivity.class);
                myIntent10.putExtra(KEY_PID7, passedVarPid);
                myIntent10.putExtra(KEY_NAME7, passedVarNn);
                myIntent10.putExtra(KEY_PLACELIST7, passedVarPL);
                NavigationActivity.this.startActivity(myIntent10);
            }
        } else if ( itype == 3 ){ //Circle
            float pradius = Float.parseFloat(coordstable[1]);
            double platcen = Double.parseDouble(coordstable[2]);
            double plngcen = Double.parseDouble(coordstable[3]);
            float[] distance = new float[2];
            Location.distanceBetween( tap.latitude, tap.longitude, platcen, plngcen, distance );

            if( distance[0] < pradius ){ //itsIN
                Log.d( tag: "msg7", msg: "Its IN circle place" + place_id);
                Intent myIntent9 = new Intent( packageContext: NavigationActivity.this, ArrivalActivity.class);
                myIntent9.putExtra(KEY_PID7, passedVarPid);
                myIntent9.putExtra(KEY_NAME7, passedVarNn);
                myIntent9.putExtra(KEY_PLACELIST7, passedVarPL);
                NavigationActivity.this.startActivity(myIntent9);
            }
        }
    } catch (JSONException e) {
        Log.e( LOG_TAG, msg: "Error processing JSON", e);
    }
}
```

Σχήμα 5.12: Έλεγχος αν Βρίσκεται μέσα σε Περιοχή

Τα αρχείο στην Εικόνα 5.12 ελέγχει αν ο χρήστης είναι μέσα στην περιοχή στην

οποία πλοηγείται. Αν είναι τότε τον μεταφέρει στην οθόνη της άφιξης.

```

/* gia thn puxida */
public void onSensorChanged( SensorEvent event ) {

    if (event.sensor == mAccelerometer) {
        System.arraycopy(event.values, srcPos: 0, mLastAccelerometer, dstPos: 0, event.values.length);
        mLastAccelerometerSet = true;
    } else if (event.sensor == mMagnetometer) {
        System.arraycopy(event.values, srcPos: 0, mLastMagnetometer, dstPos: 0, event.values.length);
        mLastMagnetometerSet = true;
    }
    if (mLastAccelerometerSet && mLastMagnetometerSet) {
        SensorManager.getRotationMatrix(mR, null, mLastAccelerometer, mLastMagnetometer);
        SensorManager.getOrientation(mR, mOrientation);
        Log.d( tag: "OrientationTestActivity", String.format("Orientation: %f, %f, %f",
            mOrientation[0], mOrientation[1], mOrientation[2]));

        float azimuth = (float) Math.toDegrees(mOrientation[0]); // orientation
        azimuth = (azimuth + 360) % 360;

        // If we don't have a Location, we break out
        if (nlocation == null) {
            Log.d( tag: "msg4", msg: "nlocation is null");
            return;
        }

        float baseAzimuth = azimuth;

        GeomagneticField geoField = new GeomagneticField(Double
            .valueOf(nlocation.getLatitude()).floatValue(), Double
            .valueOf(nlocation.getLongitude()).floatValue(),
            Double.valueOf(nlocation.getAltitude()).floatValue(),
            System.currentTimeMillis());

        // converts magnetic north into true north , azimuth = The angle that you've rotated
        // your phone from true north.
        azimuth -= geoField.getDeclination();
        // Store the bearingTo in the bearTo variable
        //bearTo = The angle from true north to the destination location from the point we're your
        // currently standing.
        float bearTo = nlocation.bearingTo(locCn);
        // If the bearTo is smaller than 0, add 360 to get the rotation clockwise.
        if (bearTo < 0) {
            bearTo = bearTo + 360;
        }
        float direction = bearTo - azimuth;
        // If the direction is smaller than 0, add 360 to get the rotation clockwise.
        if (direction < 0) {
            direction = direction + 360;
        }
        rotateImageView(arrow, R.drawable.locationarrowicon, direction); //arrow einai to imageView

        //Set the field
        String bearingText = "N";

        if ((360 >= baseAzimuth && baseAzimuth >= 337.5) || (0 <= baseAzimuth && baseAzimuth <= 22.5))
            bearingText = "N";
        else if (baseAzimuth > 22.5 && baseAzimuth < 67.5) bearingText = "NE";
        else if (baseAzimuth >= 67.5 && baseAzimuth <= 112.5) bearingText = "E";
        else if (baseAzimuth > 112.5 && baseAzimuth < 157.5) bearingText = "SE";
        else if (baseAzimuth >= 157.5 && baseAzimuth <= 202.5) bearingText = "S";
        else if (baseAzimuth > 202.5 && baseAzimuth < 247.5) bearingText = "SW";
        else if (baseAzimuth >= 247.5 && baseAzimuth <= 292.5) bearingText = "W";
        else if (baseAzimuth > 292.5 && baseAzimuth < 337.5) bearingText = "NW";
        else bearingText = "?";

        fieldBearing.setText(bearingText);
    }
}

```

Σχήμα 5.13: onsensorchanged()

Στην Εικόνα 5.13, εμφανίζεται η συνάρτηση που βρίσκει την κατεύθυνση που “κοιτάει” το κινητό. Εφαρμόζεται στη πυξίδα που υπάρχει στη Λειτουργία της Πλοήγησης.

Στην Εικόνα 5.14, φαίνεται ο κώδικας ο οποίος γυρνάει την εικόνα της πυξίδας-

```
private void rotateImageView(ImageView imageView, int drawable, float rotate ) {  
  
    // Decode the drawable into a bitmap  
    Bitmap bitmapOrg = BitmapFactory.decodeResource( getResources(),  
        drawable );  
  
    // Get the width/height of the drawable  
    DisplayMetrics dm = new DisplayMetrics(); getWindowManager().getDefaultDisplay().getMetrics(dm);  
    int width = bitmapOrg.getWidth(), height = bitmapOrg.getHeight();  
  
    // Initialize a new Matrix  
    Matrix matrix = new Matrix();  
  
    // Decide on how much to rotate  
    rotate = rotate % 360;  
  
    // Actually rotate the image  
    matrix.postRotate( rotate, width, height );  
  
    // recreate the new Bitmap via a couple conditions  
    Bitmap rotatedBitmap = Bitmap.createBitmap( bitmapOrg, 0, 0, width, height, matrix, true );  
    //BitmapDrawable bmd = new BitmapDrawable( rotatedBitmap );  
  
    //imageView.setImageBitmap( rotatedBitmap );  
    imageView.setImageDrawable(new BitmapDrawable(getResources(), rotatedBitmap));  
    imageView.setScaleType( ImageView.ScaleType.CENTER );  
}
```

Σχήμα 5.14: Κίνηση Εικόνας

βέλος. Καλείται από το `onsensorchanged()` το οποίο του δίνει την κατεύθυνση (direction) που πρέπει να "κοιτάξει" το βέλος ώστε να δείχνει σωστά προς το σημείο που κατευθυνόμαστε.

googleservice.java

Το `googleservice.java`, Εικόνα 5.15, είναι η υπερεσία (Service) που βρίσκει την τοποθεσία που βρισκόμαστε. Καλείται από αρκετές Λειτουργίες της Εφαρμογής που χρειάζονται να ξέρουν την τοποθεσία.

```

@SuppressLint("MissingPermission")
private void fn_getlocation(){
    locationManager = (LocationManager)getApplicationContext().getSystemService(LOCATION_SERVICE);
    isGPSEnable = locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);
    isNetworkEnable = locationManager.isProviderEnabled(LocationManager.NETWORK_PROVIDER);

    if (!isGPSEnable && !isNetworkEnable){

    }else {

        if (isNetworkEnable){
            location = null;
            locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, minTime: 1000, minDistance: 0, listener: this);
            if (locationManager!=null){
                location = locationManager.getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
                if (location!=null){
                    Log.e( tag: "latitude", msg: location.getLatitude()+"");
                    Log.e( tag: "longitude", msg: location.getLongitude()+"");
                    latitude = location.getLatitude();
                    longitude = location.getLongitude();
                    fn_update(location);
                }
            }
        }

        if (isGPSEnable){
            location = null;
            locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, minTime: 1000, minDistance: 0, listener: this);
            if (locationManager!=null){
                location = locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
                if (location!=null){
                    Log.e( tag: "latitude", msg: location.getLatitude()+"");
                    Log.e( tag: "longitude", msg: location.getLongitude()+"");
                    latitude = location.getLatitude();
                    longitude = location.getLongitude();
                    fn_update(location);
                }
            }
        }
    }
}

private class TimerTaskToGetLocation extends TimerTask{
    @Override
    public void run() {

        mHandler.post(new Runnable() {
            @Override
            public void run() { fn_getlocation(); }
        });
    }
}

private void fn_update(Location location){

    intentloc.putExtra( name: "latitude", value: location.getLatitude()+"");
    intentloc.putExtra( name: "longitude", value: location.getLongitude()+"");
    sendBroadcast(intentloc);
}

```

Σχήμα 5.15: Λήψη Τοποθεσίας

Λήψη Δεδομένων Βάσης

Τα loadjsontask.java (Εικόνα 5.16), placestoring.java (Εικόνα 5.17), response.java (Εικόνα 5.18) είναι τα αρχεία με τη βοήθεια των οποίων η εφαρμογή στο κινητό λαμβάνει τα δεδομένα από τη Βάση.

```
public class LoadJSONTask extends AsyncTask<String, Void, Response> {
    public LoadJSONTask(Listener listener) {
        mListener = listener;
    }

    public interface Listener {
        void onLoaded(List<PlaceStoring> allplacesList);
        void onError();
    }

    private Listener mListener;

    @Override
    protected Response doInBackground(String... strings) {
        try {
            String stringResponse = loadJSON(strings[0]);
            Gson gson = new Gson();
            return gson.fromJson(stringResponse, Response.class);
        } catch (IOException e) {
            e.printStackTrace();
            return null;
        } catch (JsonSyntaxException e) {
            e.printStackTrace();
            return null;
        }
    }

    @Override
    protected void onPostExecute(Response response) {
        if (response != null) {
            mListener.onLoaded(response.getAllplaces());
        } else {
            mListener.onError();
        }
    }

    private String loadJSON(String jsonURL) throws IOException {
        URL url = new URL(jsonURL);
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        conn.setReadTimeout(10000);
        conn.setConnectTimeout(15000);
        conn.setRequestMethod("GET");
        conn.setDoInput(true);
        conn.connect();
        BufferedReader in = new BufferedReader(new InputStreamReader(conn.getInputStream()));
        String line;
        StringBuilder response = new StringBuilder();
        while ((line = in.readLine()) != null) {
            response.append(line);
        }
        in.close();
        return response.toString();
    }
}
```

Σχήμα 5.16: loadjsontask.java

nearby.java

Στο αρχείο nearby.java γίνεται ταξινόμηση των περιοχών που λαμβάνονται από τη Βάση Δεδομένων. Για την υλοποίηση αυτή χρησιμοποιήθηκε, όπως φαίνεται και στις Εικόνες 5.19, 5.20, 5.21, ένας Συγκριτής (Comparator). Ο Comparator είναι μια

```
public class PlaceStoring {  
  
    private String place_id;  
    private String user_id;  
    private String name;  
    private String description;  
    private String type;  
    private String coordinates;  
    private String pimage;  
  
    public String getPlace_id() { return place_id; }  
  
    public String getUser_id() { return user_id; }  
  
    public String getName() { return name; }  
  
    public String getDescription() { return description; }  
  
    public String getType() { return type; }  
  
    public String getCoordinates() { return coordinates; }  
  
    public String getImage() { return pimage; }  
  
}
```

Σχήμα 5.17: placestoring.java

```
public class Response {  
  
    private List<PlaceStoring> allplaces = new ArrayList<PlaceStoring>();  
  
    public List<PlaceStoring> getAllplaces() { return allplaces; }  
  
}
```

Σχήμα 5.18: response.java

```
class SortbyDist extends Context implements Comparator<PlaceStoring> {  
    // Used for sorting in ascending order of distance  
    public int compare(PlaceStoring a, PlaceStoring b) {
```

Σχήμα 5.19: Comparator<T>

```
        double dista = loca.distanceTo(location2);  
        double distb = locb.distanceTo(location2);  
        return Double.compare(dista, distb);
```

Σχήμα 5.20: compare()

```
        Collections.sort(allplacesList, new SortbyDist());
```

Σχήμα 5.21: Collections.sort()

λειτουργία σύγκρισης, η οποία επιβάλλει μια ταξινόμηση σε κάποια συλλογή αντικειμένων. Οι comparators μπορούν να μεταβιβαστούν σε μια μέθοδο ταξινόμησης

(όπως η Collections.sort που φαίνεται στην Εικόνα) ώστε να έχουν τον ακριβή έλεγχο της σειράς ταξινόμησης. Οι comparators μπορούν επίσης να χρησιμοποιηθούν για τον έλεγχο της ταξινόμησης ορισμένων δομών δεδομένων (πχ ταξινομημένα σύνολα ή ταξινομημένοι χάρτες).

Η σειρά που επιβάλλεται από ένα συγκριτή c σε ένα σύνολο στοιχείων S λέγεται ότι είναι συνεπής με το ίσο αν και μόνο αν το $c.compare(e1, e2) == 0$ έχει την ίδια boolean τιμή με το $e1.equals(e2)$ για κάθε $e1$ και $e2$ στο S .

tourdetails.java

Για την Ταξινόμηση Περιοχών μιας Περιήγησης αναπτύχθηκε ο κώδικας της Εικόνας 5.22. Η λειτουργία του κώδικα είναι να παίρνει τις περιοχές της Περιήγησης που έχει επιλεχτεί για πλοήγηση και να τις ταξινομεί στις μικρότερες αποστάσεις μεταξύ τους. Δηλαδή αρχικά βρίσκει την πιο κοντινή στο σημείο που βρισκόμαστε και την αποθηκεύει ως το πρώτο μέρος προς επίσκεψη, στη συνέχεια βρίσκει την πιο κοντινή περιοχή του μέρους που ορίστηκε ως πρώτο στην περιήγηση και ορίζεται ως δεύτερο, και ούτω καθεξής μέχρι να τελειώσουν όλα τα μέρη της περιήγησης.

```

private String[] sortTourPlaces(int[] spt) {

    int k=0;
    myloc.setLatitude(tlatitude);
    myloc.setLongitude(tlongitude);
    int ls = spt.length;
    String[] ept = new String[ls];
    String[][] sv = new String[1][2];
    int j = ls;

    while ( k<ls ){

        String[][] distable = new String[j][3]; //emptying array
        j--;
        int m =0;

        for(int i=0; i<spt.length; i++){
            if ( spt[i] != 0 ) {
                distable[m][0] = Integer.toString(spt[i]);
                String type ="0";
                String coordinates ="0";

                for (int n =0; n<mTPlacesMapList.size() ;n++){
                    if( Integer.parseInt(mTPlacesMapList.get(n).get(KEY_PID)) == spt[i] ){
                        type = mTPlacesMapList.get(n).get(KEY_TYPE);
                        coordinates = mTPlacesMapList.get(n).get(KEY_COO);
                    }
                }

                double locCnlat = Double.parseDouble(latc);
                double locCnlng = Double.parseDouble(lngc);

                locCn.setLatitude(locCnlat);
                locCn.setLongitude(locCnlng);

                distable[m][1]= Double.toString( locCn.distanceTo(myloc) );
                distable[m][2]= locCn.getLatitude() +" "+locCn.getLongitude();
                m++;
            }
        }

        sv[0][0]= distable[0][0];
        sv[0][1]= distable[0][1];

        for (int i=1;i<distable.length; i++){
            if( Double.parseDouble(distable[i][1]) < Double.parseDouble(sv[0][1]) ){
                sv[0][0]=distable[i][0];
                sv[0][1]=distable[i][1];
            }
        }
        ept[k]=sv[0][0];
        k++;
        for(int i=0;i<spt.length;i++){
            if ( spt[i] == Integer.parseInt( sv[0][0] ) ){
                spt[i]=0;
            }
        }

        for (int i=0; i<distable.length; i++) {
            if (distable[i][0] == sv[0][0] ) {
                String[] coo2 = distable[i][2].split( regularExpression: ";" );

                myloc.setLatitude( Double.parseDouble(coo2[0]) );
                myloc.setLongitude( Double.parseDouble(coo2[1]) );
            }
        }
    }
    return ept;
}

```

Σχήμα 5.22: Ταξινόμηση Περιοχών Περιήγησης

5.3 Σύνοψη Κεφαλαίου

Στο παρόν κεφάλαιο έγινε μία ανάλυση του τρόπου λειτουργίας της Εφαρμογής. Παρουσιάστηκε το γραφικό της περιβάλλον και όλες οι ενέργειες που μπορούν να εκτελέσουν οι χρήστες. Τέλος περιγράφηκαν μερικά σημαντικά σημεία του κώδικα μαζί με τα μετρικά της.

Κεφάλαιο 6

Επίλογος

Στο κεφάλαιο αυτό γίνεται μια σύνοψη της Ιστοσελίδας και της Εφαρμογής που σχεδιάστηκε και υλοποιήθηκε σε αυτή τη διπλωματική εργασία. Παρουσιάζεται μία ανακεφαλαίωση των όσων παρουσιάστηκαν και αναλύθηκαν, καταγράφονται τα συμπεράσματα που εξάχθηκαν κατά την υλοποίηση και περιγράφονται οι μελλοντικές επεκτάσεις που μπορούν να υπάρξουν για τη βελτιστοποίηση του συστήματος. Τέλος, γίνεται αναφορά στις συνολικές μετρικές και παρέχεται η ανάλυση SWOT (Strengths, Weaknesses, Opportunities, Threats) του συστήματος που αναπτύχθηκε.

6.1 Σύνοψη Συστήματος

Στην παρούσα εργασία, έγινε ο σχεδιασμός και η ανάπτυξη μίας Εφαρμογής Android με υποστηρικτική Ιστοσελίδα με σκοπό τη δημιουργία ταξιδιωτικού οδηγού της Δυτικής Μακεδονίας. Το παρόν σύστημα έχει ως σκοπό τη διευκόλυνση την ανάδειξη περιοχών και περιηγήσεων της Περιφέρειας Δυτικής Μακεδονίας και την προώθηση του τουρισμού της.

Το σύστημα αποτελείται από μία βάση δεδομένων μία ιστοσελίδα διαχείρισης και μία εφαρμογή Android. Η βάση δεδομένων διατηρεί πληροφορίες για τους χρήστες, τις περιοχές και τις περιηγήσεις. Η ιστοσελίδα αποτελεί το γραφικό περιβάλλον του χρήστη, στο οποίο ανάλογα με τα δικαιώματα πρόσβασής του μπορεί να διαχειριστεί από περιοχές και περιηγήσεις μέχρι και τους ίδιους τους χρήστες. Η εφαρμογή περιέχει λειτουργίες όπως αναζήτηση κάποιας διεύθυνσης, εμφάνιση των αποθηκευμένων περιοχών σε χάρτη, εμφάνιση όλων των περιοχών και περιηγήσεων, αλλά και των κοντινότερων, σε λίστα από όπου μπορούν να μάθουν παραπάνω πληροφορίες για αυτές και τέλος η πλοήγηση του χρήστη σε συγκεκριμένες περιοχές, ή

σε περιοχές μίας περιήγησης.

Οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για τον ιστοχώρο είναι οι: HTML5, CSS, PHP, JavaScript, AJAX και το framework της Bootstrap. Η MySQL βάση δεδομένων περιλαμβάνει τρεις πίνακες, των χρηστών, των περιοχών και των περιηγήσεων. Για μεγαλύτερη ασφάλεια, η σύνδεση με το σύστημα και η διαχείριση των δεδομένων της βάσης γίνεται με τη χρήση προκατασκευασμένων εντολών PDO. Για επιπλέον ασφάλεια, χρησιμοποιούνται κρυπτογραφημένοι κωδικοί πρόσβασης, γίνεται φιλτράρισμα και επικύρωση των τύπων των δεδομένων που εισάγονται από το χρήστη από τις διάφορες φόρμες επικοινωνίας, όπως επίσης χρησιμοποιούνται SESSIONS και ασφαλή σύνδεση HTTPS. Για την εφαρμογή χρησιμοποιήθηκαν οι γλώσσες JAVA και XML. Η αποστολή αρχείων μεταξύ εφαρμογής και ιστοσελίδας γίνεται μέσω JSON.

6.2 Μετρικά Κώδικα Συστήματος

Σε αυτή την ενότητα παραθέτονται τα στοιχεία σχετικά με τους κώδικες που αναπτύχθηκαν για τον προγραμματισμό της Ιστοσελίδας και της Εφαρμογής.

6.2.1 Μετρικά Κώδικα Ιστοχώρου

Για τις ανάγκες και τη σωστή λειτουργία του Ιστοχώρου δημιουργήθηκαν 38 αρχεία πηγαίου κώδικα PHP και χρησιμοποιήθηκαν 2 εικονίδια και από 1 αρχείο CSS και JavaScript. Υπάρχουν και άλλα αρχεία τέτοιων τύπων, τα οποία όμως δεν αναφέρονται γιατί είναι από έτοιμες δομές (framework) και βιβλιοθήκες που χρησιμοποιήθηκαν. Οι μετρικές του συστήματος περιγράφονται στον Πίνακα 6.1.

Μετρικές	Αριθμός
Φάκελοι	274
Αρχεία πηγαίου κώδικα	40
Συνολικός αριθμός αρχείων	2020
Συνολικό μέγεθος αρχείων	40MB

Πίνακας 6.1: Μετρικές Ιστοχώρου

6.2.2 Μετρικά Κώδικα Εφαρμογής

Για τις ανάγκες και τη λειτουργία της Εφαρμογής δημιουργήθηκαν συνολικά 48 αρχεία εκ των οποίων τα 17 είναι JAVA και τα άλλα 31 είναι αρχεία XML. Επίσης χρησιμοποιήθηκαν 12 εικονίδια. Τα υπόλοιπα αρχεία που αποτελούν την εφαρμογή δεν αναφέρονται γιατί δημιουργούνται αυτόματα από το Android Studio. Οι μετρικές της Εφαρμογής περιγράφονται στον Πίνακα 6.2.

Μετρικές	Αριθμός
Φάκελοι	1055
Αρχεία πηγαίου κώδικα	49
Συνολικός αριθμός αρχείων	4684
Συνολικό μέγεθος αρχείων	123MB

Πίνακας 6.2: Μετρικές Εφαρμογής

6.3 Συμπεράσματα

Τα συμπεράσματα που μπορούν να εξαχθούν είναι πολλά και ενδιαφέροντα και σχετίζονται άμεσα με τα οφέλη χρήσης του συστήματος. Τα οφέλη από τον επιτυχή σχεδιασμό και την υλοποίηση μιας εφαρμογής ταξιδιωτικού περιεχομένου στοχευμένη στην Δυτική Μακεδονία, αναμένονται πολλά και σημαντικά, τόσο για τους άμεσα επηρεαζόμενους, δηλαδή τους κατοίκους της περιοχής αφού βοηθάει στην ανάπτυξη της οικονομίας και στην ανάδειξη του τόπου τους, όσο και για τους ταξιδιώτες που διευκολύνει και απλοποιεί τον τρόπο να βρουν πληροφορίες για σημεία ενδιαφέροντος και αξιοθέατα που είναι λιγότερο γνωστά αλλά εξίσου σημαντικά. Είναι γεγονός ότι, λόγω της ραγδαίας εξέλιξης της τεχνολογίας και της άμεσης επιρροής της στην καθημερινότητα των ανθρώπων, συστήματα όπως αυτό, μπορούν να συνδράμουν σημαντικά στην διευκόλυνση και απλούστευση διαδικασιών που κατά τα άλλα είναι χρονοβόρες και περίπλοκες.

6.4 Ανάλυση SWOT

Η ανάλυση SWOT είναι ένα χρήσιμο εργαλείο, το οποίο εξετάζει τα Δυνατά Σημεία (Strengths), τις Αδυναμίες (Weaknesses), τις Ευκαιρίες (Opportunities) αλλά και τις Απειλές (Threats) του συστήματος. Αυτό διευκολύνει τον καθορισμό των απαιτήσεων και των μελλοντικών επεκτάσεων του..

6.4.1 Δυνατά Σημεία

Το παρόν σύστημα παρουσιάζει αρκετά από τα πλεονεκτήματα που μπορεί να φέρει ένα τέτοιου είδους σύστημα, τα οποία έχουν αναφερθεί και αναλυθεί κατά τη συγγραφή αυτής της διπλωματικής εργασίας.

Αρχικά, η Ιστοσελίδα μπορεί να χρησιμοποιηθεί από οποιονδήποτε υπολογιστή ή άλλη φορητή συσκευή αρκεί αυτή να έχει πρόσβαση στο διαδίκτυο. Επίσης είναι συμβατή με όλα τα λειτουργικά συστήματα, αφού εκτελείται μέσω του περιηγητή διαδικτύου. Έχει ελάχιστες απαιτήσεις υπολογιστικών πόρων και λειτουργεί σε διάφορα μεγέθη οθονών, επιτρέποντας την προβολή του ακόμα και σε κινητά τηλέφωνα.

Στη συνέχεια, τόσο η Εφαρμογή όσο και η Ιστοσελίδα μπορούν να χρησιμοποιηθούν από οποιαδήποτε τοποθεσία και οποιαδήποτε χρονική στιγμή. Το περιβάλλον τους είναι απλό και φιλικό προς τον χρήστη και δεν απαιτεί από ιδιαίτερες γνώσεις για να τη χρήση του. Επιπλέον δεν υπάρχει καμία οικονομική επιβάρυνση μιας και τα δύο παρέχονται δωρεάν.

Τέλος η εφαρμογή μπορεί να εκτελεστεί από οποιαδήποτε σύγχρονη έξυπνη συσκευή με που δουλεύει με λογισμικό android.

6.4.2 Αδυναμίες

Η κύρια αδυναμία του συστήματος είναι το γεγονός ότι δεν έχει δοκιμαστεί ακόμα σε πραγματικές συνθήκες χρήσης και σε πολύ μεγάλο όγκο δεδομένων. Επίσης δεν έχει δοκιμαστεί με μεγάλο αριθμό χρηστών να είναι συνδεδεμένοι ταυτόχρονα.

6.4.3 Ευκαιρίες

Η συνεχής μείωση του χρόνου και του κόστους μεταφοράς δημιουργεί όλο και περισσότερες ευκαιρίες για τη χρήση του συστήματος γιατί γίνεται όλο και πιο εύκολο να ταξιδεύεις από το ένα μέρος στο άλλο. Έτσι η ζήτηση για νέες και "ανεξερεύνητες" εναλλακτικές περιοχές συνεχώς αυξάνεται.

Επίσης, όσοι οι ρυθμοί της καθημερινότητας αυξάνονται, όσοι ταξιδεύουν θέλουν να χάνουν όσο λιγότερο χρόνο γίνεται στον προγραμματισμό του ταξιδιού και να αφιερώνουν περισσότερο χρόνο στο ίδιο το ταξίδι.

Τέλος, μια ακόμα ευκαιρία που προσφέρεται είναι ότι το σύστημα είναι ευέλικτο και μπορεί με ελάχιστες αλλαγές να χρησιμοποιηθεί για οποιαδήποτε περιοχή εκτός της Δυτικής Μακεδονίας.

6.4.4 Απειλές

Η βασική απειλή που παρουσιάζεται είναι ο ανταγωνισμός, καθώς υπάρχουν πολλές παρόμοιες εφαρμογές, με πολλές παρόμοιες δυνατότητες, ακόμα και διάσημες εφαρμογές παρεμφερές πεδίου που όμως έχουν αρχίσει να υποστηρίζουν τέτοιου είδους υπηρεσίες.

6.5 Μελλοντικές Επεκτάσεις

Στο σύστημα που αναπτύχθηκε, όπως και σε κάθε σύστημα, πάντα υπάρχει χώρος για βελτίωση και μελλοντικές επεκτάσεις, άλλοτε μείζονος και άλλοτε ήσσονος σημασίας. Παρακάτω αναφέρονται μερικές που αφορούν την παρούσα διπλωματική.

6.5.1 Προσθήκη Περισσότερων Περιοχών

Μια βασική επέκταση της παρούσας εφαρμογής μπορεί να γίνει προσθέτοντας περισσότερες περιοχές της Ελλάδας αλλά και του εξωτερικού, ώστε να μην είναι πλέον επικεντρωμένη στη περιφέρεια Δυτικής Μακεδονίας.

6.5.2 Ανάπτυξη σε Διαφορετική Πλατφόρμα

Μια ακόμα σημαντική επέκταση θα ήταν και η υλοποίηση της εφαρμογής για διάφορες πλατφόρμες, άλλες από αυτές του Android, όπως iOS, Windows Phone κλπ. Έτσι θα απευθύνεται σε ακόμα μεγαλύτερη μερίδα του πληθυσμού.

6.5.3 Προσθήκη Αξιολογήσεων Χρηστών

Επίσης μια λεπτομέρεια που θα μπορούσε να προστεθεί είναι η δυνατότητα οι χρήστες της εφαρμογής να εισάγουν κριτικές και αξιολογήσεις (reviews/ratings) για τις περιοχές που επισκέφτηκαν, και να μπορούν να ανεβάζουν φωτογραφίες που τράβηξαν οι ίδιοι από το κάθε μέρος. Έτσι θα μπορούν να ενημερώνονται και οι υπόλοιποι χρήστες που σκοπεύουν να τις επισκεφτούν, για μια δεύτερη γνώμη από επισκέπτη σε επισκέπτη.

6.5.4 Επαυξημένη Πραγματικότητα

Άλλη πιθανή μελλοντική επέκταση είναι η προσθήκη λειτουργιών επαυξημένης πραγματικότητας (Augmented Reality). Η εφαρμογή χρησιμοποιώντας την κάμερα του κινητού και επαυξημένης πραγματικότητας στοιχεία θα μπορεί να κατευθυνθεί το χρήστη σε κοντινά αξιοθέατα καθώς του παρέχονται πληροφορίες για αυτά. Έτσι θα υπάρχει ένας πιο διασκεδαστικό τρόπος πλοήγησης.

6.5.5 Εικονική Πραγματικότητα

Ακόμα θα μπορούσε να προστεθεί η εμπειρία εικονικής πραγματικότητας (Virtual Reality) με την συνοδεία των ειδικών γυαλιών. Με αυτόν τον τρόπο ο χρήστης θα μπορεί από οπουδήποτε να βλέπει και να βιώνει μια περιοχή ή μια διαδρομή σαν να ήταν ο ίδιος εκεί.

6.5.6 Ηχητικές Περιγραφές

Τέλος μια ενδιαφέρουσα προσθήκη θα ήταν αυτή των ηχητικών περιγραφών και ξεναγήσεων. Έτσι αντικαθίσταται σχεδόν πιστά ο ξεναγός. Πλέον δεν είναι αναγκαίο ο χρήστης να κοιτάει τη κινητή συσκευή του, αλλά να απολαμβάνει τη διαδρομή. Με

αυτόν τον τρόπο η εφαρμογή γίνεται προσβάσιμη και από άτομα με προβλήματα όρασης.

6.6 Σύνοψη Κεφαλαίου

Στο αυτό το τελευταίο κεφάλαιο, συνοψίστηκαν ο τρόπος σχεδίασης και υλοποίησης του συστήματος, έγινε αναφορά στα γενικά συμπεράσματα που προέκυψαν και στην ανάλυση των δυνατών και αδύνατων σημείων του αλλά και των αδυναμιών και των ευκαιριών του. Τέλος έγινε κλείσιμο της εργασίας με την εξέλιξη και τις επεκτάσεις που θα μπορούσε να έχει το παρόν σύστημα στο μέλλον.

Βιβλιογραφία

- [1] About - notepad++, Δεκέμβριος 2017. <https://notepad-plus-plus.org/>.
- [2] About - phpmyadmin, Δεκέμβριος 2017. <https://www.phpmyadmin.net/>.
- [3] Ricardo Anacleto, Lino Figueiredo, Ana Almeida, and Paulo Novais. Mobile application to provide personalized sightseeing tours. *Journal of Network and Computer Applications*, 41:56–64, 2014.
- [4] H Anegg and M Umlauf. Lol@: usability of a location based umts application. *e & i Elektrotechnik und Informationstechnik*, 120(2):61–65, 2003.
- [5] Bootstrap (front-end framework), Δεκέμβριος 2017. [https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)).
- [6] Damianos Gavalas and Michael Kenteris. A web-based pervasive recommendation system for mobile tourist guides. *Personal and Ubiquitous Computing*, 15(7):759–770, 2011.
- [7] Damianos Gavalas, Charalampos Konstantopoulos, Basilis Mamalis, and Grammati Pantziou. Mobility prediction in mobile ad-hoc networks. In *Next generation mobile networks and ubiquitous computing*, pages 226–240. IGI Global, 2011.
- [8] Getting started - what's ajax?, Δεκέμβριος 2017. https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started.
- [9] Gimp, Δεκέμβριος 2017. https://en.wikipedia.org/wiki/GIMP#cite_note-what-is-gimp-7.
- [10] Michael F Goodchild. Geographic information systems and science: today and tomorrow. *Annals of GIS*, 15(1):3–9, 2009.
- [11] Google maps api, Δεκέμβριος 2017. https://www.w3schools.com/graphics/google_maps_intro.asp.
- [12] A Yair Grinberger, Noam Shoval, and Bob McKercher. Typologies of tourists' time-space consumption: a new approach using gps data and gis tools. *Tourism Geographies*, 16(1):105–123, 2014.
- [13] History of mysql, Δεκέμβριος 2017. <https://dev.mysql.com/doc/refman/5.7/en/history.html>.
- [14] Html, Δεκέμβριος 2017. <https://developer.mozilla.org/en-US/docs/Learn/HTML>.
- [15] Introducing winscp, Δεκέμβριος 2017. <https://winscp.net/eng/docs/introduction>.

- [16] Javascript, Δεκέμβριος 2017. <https://developer.mozilla.org/en-US/docs/Learn/JavaScript>.
- [17] Learn to style html using css, Δεκέμβριος 2017. <https://developer.mozilla.org/en-US/docs/Learn/CSS>.
- [18] Hassan I Mathkour. A gps-based mobile dynamic service locator system. *Applied computing and informatics*, 9(2):95–106, 2011.
- [19] Meet android studio, Δεκέμβριος 2017. <https://developer.android.com/studio/intro/index.html>.
- [20] Mysql workbench, Δεκέμβριος 2017. <https://www.mysql.com/products/workbench/>.
- [21] José M Noguera, Manuel J Barranco, Rafael J Segura, and Luis MartíNez. A mobile 3d-gis hybrid recommender system for tourism. *Information Sciences*, 215:37–52, 2012.
- [22] Maziyar Shariat Panahi, Peter Woods, and Harold Thwaites. Designing and developing a location-based mobile tourism application by using cloud-based platform. In *Technology, Informatics, Management, Engineering, and Environment (TIME-E), 2013 International Conference on*, pages 151–156. IEEE, 2013.
- [23] Protocols & communication formats[xml/json] & apis, May 2018. https://www.uni-weimar.de/kunst-und-gestaltung/wiki/images/Vortrag_Protocols.pdf.
- [24] Putty, Δεκέμβριος 2017. <http://www.putty.org/>.
- [25] Douglas B Richardson, Nora D Volkow, Mei-Po Kwan, Robert M Kaplan, Michael F Goodchild, and Robert T Croyle. Spatial turn in health research. *Science*, 339(6126):1390–1392, 2013.
- [26] MC Rodriguez-Sanchez, Juan Martinez-Romo, Susana Borromeo, and JA Hernandez-Tamames. Gat: Platform for automatic context-aware mobile services for m-tourism. *Expert Systems with applications*, 40(10):4154–4163, 2013.
- [27] Securing passwords with bcrypt hashing function, May 2018. <https://thehackernews.com/2014/04/securing-passwords-with-bcrypt-hashing.html>.
- [28] Noam Shoval, Michal Isaacson, and Prem Chhetri. Gps, smartphones, and the future of tourism research. *The Wiley Blackwell companion to tourism*, pages 251–261, 2013.
- [29] Web development, December 2017. https://en.wikipedia.org/wiki/Web_development.
- [30] What is jquery?, Δεκέμβριος 2017. <https://jquery.com/>.
- [31] What is php?, Δεκέμβριος 2017. <http://php.net/manual/en/intro-what-is.php>.
- [32] Wan-Shiou Yang and San-Yih Hwang. itravel: A recommender system in mobile peer-to-peer environment. *Journal of Systems and Software*, 86(1):12–20, 2013.

Παραρτήματα

Παράρτημα Α΄

Οδηγίες Εγκατάστασης Ιστοχώρου

Στο παράρτημα αυτό παρουσιάζεται η διαδικασία εγκατάστασης του Ιστοχώρου που αναπτύχθηκε, ώστε να είναι δυνατή η χρήση και η λειτουργία του από οποιοδήποτε διακομιστή. Για τη σωστή εγκατάσταση και την απρόσκοπτη λειτουργία ενός εξυπηρετητή ιστού με δυνατότητα PHP + MySQL, όπως ο δικός μας, είναι αναγκαία η λήψη και εγκατάσταση των ελεύθερων λογισμικών: Apache HTTPD Web Server, PHP Windows Binaries, phpMyAdmin Database Manager. Μια εναλλακτική είναι η λήψη και εγκατάσταση του Xampp πακέτου, το οποίο περιέχει όλα τα παραπάνω, όπως και ένα πίνακα ελέγχου για τη διαχείρισή τους. Επίσης θα πρέπει να δοθεί προσοχή ώστε η έκδοση της PHP θα πρέπει να είναι από την 5η και πάνω ώστε να είναι δυνατή η υποστήριξη των PDO.

Στην συνέχεια τα επόμενα βήματα για την εγκατάσταση του συστήματος είναι:

1. Ο φάκελος "public_html" να γίνει αντιγραφή στον εξυπηρετητή (server).
2. Τα δικαιώματα των αρχείων θα πρέπει να είναι 755, τόσο του φακέλου όσο και των περιεχομένων του, εκτός από τον φάκελο "places" ο οποίος χρειάζεται πλήρη δικαιώματα (777).
3. Στο αρχείο connect.php να γίνει αντικατάσταση του host με το socket που αντιστοιχεί στη βάση δεδομένων του καινούριου server.
4. Στο phpMyAdmin εισαγωγή του αρχείου "places.sql", το οποίο αποτελεί το αρχείο αρχικοποίησης της βάσης δεδομένων.

Πλέον ο ιστότοπος είναι έτοιμος για χρήση. Ανοίγοντας έναν περιηγητή στη διεύθυνση του εξυπηρετητή που έχουμε τοποθετήσει τον φάκελο "public_html", εμφανίζεται

η σελίδα σύνδεσης. Εκεί είναι δυνατό να γίνει σύνδεση με κάποιον υπάρχων χρήστη(πχ τον διαχειριστή με όνομα χρήστη: admin και κωδικό πρόσβασης: Admin!11) ή να δημιουργηθεί καινούριος. Μετά τη σύνδεση μπορεί να αρχίσει η πλήρης χρήση του συστήματος.

Παράρτημα Β΄

Οδηγίες Εγκατάστασης Εφαρμογής

Στο παράρτημα αυτό παρουσιάζεται η διαδικασία εγκατάστασης της Εφαρμογής που αναπτύχθηκε, για να γίνει δυνατή η χρήση της.

Αρχικά ο χρήστης πρέπει να έχει μία έξυπνη κινητή συσκευή με λογισμικό Android τουλάχιστον στην έκδοση 4.2. Στην συνέχεια πρέπει να κατεβάσει και να εγκαταστήσει το apk (Android Application Package) αρχείο της εφαρμογής. Πλέον η εφαρμογή είναι εγκατεστημένη στη συσκευή. Με το πάτημα του εικονιδίου και την έγκριση της παραχώρησης των δικαιωμάτων που ζητούνται η εφαρμογή μπορεί να αρχίσει τη λειτουργία της.