

Πτυχιακή Εργασία



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανικών Πληροφορικής



Τίτλος: Υπηρεσίες εντοπισμού θέσης (Location Based Services) βασισμένες στο GPS των Android μικροσυσκευών.

**Κακουλάκης Ιωάννης (ΑΜ: 2197)
Χατζημανώλης Αντώνιος (ΑΜ: 2410)**

Επιβλέπων καθηγητής : Μαλάμος Αθανάσιος

Επιτροπή Αξιολόγησης :

Ημερομηνία παρουσίασης:

Ευχαριστίες

Θα θέλαμε να ευχαριστήσουμε το κ.Μαλάμο Αθανάσιο τόσο για την ιδέα της πτυχιακής, όσο και για τη στήριξη που μας παρείχε σε όλη τη διάρκεια εκπόνησής της. Επίσης ευχαριστούμε τον κ.Καπετανάκη Κωνσταντίνο, για τις συμβουλές και τη καθοδήγηση σε όποιες δυσκολίες αντιμετωπίσαμε και την άμεση ανταπόκριση σε ότι άλλο χρειαστήκαμε. Τέλος ευχαριστούμε όλο το προσωπικό του εργαστηρίου Πολυμέσων του ΤΕΙ Κρήτης, για την εξασφάλιση της υλικοτεχνικής υποδομής στην οποία στηρίχθηκε η παρούσα Πτυχιακή Εργασία.

Σύνοψη

Το Android, είναι αρκετά διαδεδομένο λειτουργικό σύστημα σε κινητές συσκευές, με τεράστια επιτυχία στην ανάπτυξη και διάθεση εφαρμογών και επίσης δίνει τη δυνατότητα άριστης συνεργασίας με το υλικό μέρος της συσκευής, όπως για παράδειγμα κάμερα, δέκτη GPS κ.α.

Η πτυχιακή μας εργασία έχει ως στόχο, τη σχεδίαση και ανάπτυξη μίας εφαρμογής Android, η οποία θα παίρνει τη τοποθεσία του χρήστη και θα εμφανίζει πληροφορίες σχετικές με τη περιοχή στην οποία βρίσκεται. Για να γίνει αυτό, απαιτείται η ανάπτυξη μίας βάσης δεδομένων και μίας web-based εφαρμογής, που αποτελεί το Administration μέρος.

Οι τεχνολογίες που θα χρησιμοποιήσουμε για να επιτευχθούν τα παραπάνω, είναι Android, PHP, HTML5, CSS, Javascript και λειτουργία ενός Apache HTTP Server και μίας MySQL βάσης δεδομένων μέσω της εφαρμογής Xampp.

Abstract

Android, is a fairly widespread Operating System on mobile devices, with great success in the development and distribution of applications and also allows excellent collaboration with the hardware elements of the devices, such as camera, GPS receiver etc.

The target of this Thesis is, design and development of an Android application, which will determines the location of the user and displays information about the area in which it is located. The development of a Data Base and a web-based application, which will be the Administration part, is required to achieve our purpose.

The technologies we use to achieve this is, Android, PHP, HTML5, CSS, Javascript and operation of an Apache HTTP Server and a MySQL database which runs on the Xampp application.

Περιεχόμενα

Εξώφυλλο	1
Ευχαριστίες.....	2
Σύνοψη	3
Abstract.....	4
Πίνακας Περιεχομένων.....	5
Πίνακας Εικόνων	7
1. Εισαγωγή	10
1.1 Περίληψη	10
2. Μεθοδολογία	11
2.1 GPS (Global Positioning System)	11
2.2 Φορητές συσκευές GPS.....	14
2.3 Geo-fence (γεωγραφικό πλέγμα)	15
2.4 Google Maps	17
2.5 Υπηρεσίες Εντοπισμού Θέσης (Location-based services)	19
3. Σχέδιο Δράσης	21
3.1 Ανάπτυξη του Administration.....	21
3.1.1 Τι είναι η HTML5.....	21
3.1.2 Τι είναι CSS	23
3.1.3 Τι είναι Βάση Δεδομένων	24
3.1.4 Τι είναι η MySQL.....	25
3.1.5 Τι είναι η PHP.....	26
3.1.6 Apache HTTP Server.....	27
3.1.7 Τι είναι Xampp	28
3.1.8 PhpMyAdmin	29
3.1.9 Τι είναι το Dreamweaver	30
3.1.10 Javascript	31
3.2 Ανάπτυξη εφαρμογής Android.....	32
3.2.1 Google	32
3.2.2 Google Play	33
3.2.3 Google Play Developers	35
3.2.4 Android	36
3.2.5 Ανάπτυξη Εφαρμογών σε Android Περιβάλλον.....	37
3.2.6 Android Versions	38
3.2.7 Eclipse	47
3.2.8 Android SDK	48
3.2.9 Αρχείο Android Manifest.....	49
3.2.10 To Format JSON	50
4. Κύριο Μέρος	51
4.1 Ανάλυση Προβλήματος	51
4.2 Δημιουργία της Βάσης Δεδομένων.....	52
4.3 Administration της Εφαρμογής	60
4.4 Εφαρμογή Android	76
4.4.1 Δημιουργία ενός Android Project	76
4.4.2 SDK Manager	79
4.4.3 Δημιουργία και τρέξιμο ενός Emulator	80

4.4.4 Στάδια λειτουργίας της Android Εφαρμογής	82
4.4.5 Export μιας Android Εφαρμογής.....	85
5. Αποτελέσματα	88
5.1 Συμπεράσματα.....	88
5.2 Μελλοντική Δουλειά	88
6. Βιβλιογραφία	89
Παράρτημα Α - Κώδικας	90
Παράρτημα Β - Διαφάνειες Παρουσίασης	126

Πίνακας Εικόνων

Εικόνα 2.1: Λογότυπο GPS	11
Εικόνα 2.2: Χρήση GPS Navigation σε Smartphone	14
Εικόνα 2.3: Εφαρμογή Google Maps για κινητές συσκευές	17
Εικόνα 2.4: Ιστοσελίδα υπηρεσίας Google Maps	18
Εικόνα 3.1: Λογότυπο HTML5	21
Εικόνα 3.2: Λογότυπο CSS.....	23
Εικόνα 3.3: Σχηματική περιγραφή Βάσης Δεδομένων & DBMS	24
Εικόνα 3.4: Λογότυπο MySQL	25
Εικόνα 3.5: Λογότυπο PHP	26
Εικόνα 3.6: Λογότυπο Apache HTTP Server.....	27
Εικόνα 3.7: Λογότυπο Xampp	28
Εικόνα 3.8: Λογότυπο phpMyAdmin	29
Εικόνα 3.9: Κεντρική σελίδα phpMyAdmin.....	29
Εικόνα 3.10: Λογότυπο Dreamweaver CC	30
Εικόνα 3.11: Λογότυπο JavaScript	31
Εικόνα 3.12: Λογότυπο Google.....	32
Εικόνα 3.13: Λογότυπο Google Play	33
Εικόνα 3.14: Λογότυπο Google Developers.....	35
Εικόνα 3.15: Λογότυπο Android	36
Εικόνα 3.16: Λογότυπο Android Cupcake 1.5	38
Εικόνα 3.17: Λογότυπο Android Donut 1.6	39
Εικόνα 3.18: Λογότυπο Android Eclair 2.0 – 2.1	40
Εικόνα 3.19: Λογότυπο Android Frogo 2.2 – 2.2.3	41
Εικόνα 3.20: Λογότυπο Android Gingerbread 2.3 - 2.3.7.....	42
Εικόνα 3.21: Λογότυπο Android Honeycomb 3.0 – 3.2.6.....	43
Εικόνα 3.22: Λογότυπο Android Ice Cream Sandwich 4.0 – 4.0.4	44
Εικόνα 3.23: Λογότυπο Android 4.1 – 4.3.1.....	45
Εικόνα 3.24: Λογότυπο Android KitKat 4.4 – 4.4.4	46
Εικόνα 3.25: Λογότυπο Eclipse 3.2	47
Εικόνα 3.26: Λογότυπο Android SDK	48
Εικόνα 3.27: Λογότυπο JSON.....	50
Εικόνα 4.1: Σχηματική απεικόνιση λειτουργίας της Εφαρμογής	51
Εικόνα 4.2: Συντόμευση Xampp Control Panel	52
Εικόνα 4.3: Εικονίδιο εκτέλεσης του Xampp	52
Εικόνα 4.4: Xampp Control Panel	53
Εικόνα 4.5: Πληκτρολόγηση του localhost στο browser	53
Εικόνα 4.6: Αρχική σελίδα Xampp.....	54
Εικόνα 4.7: Αρχική σελίδα phpMyAdmin.....	54
Εικόνα 4.8: Δημιουργία Βάσης Δεδομένων.....	55
Εικόνα 4.9: Προβολή πινάκων Βάσης Δεδομένων.....	56
Εικόνα 4.10: Προβολή πίνακα admins	56
Εικόνα 4.11: Προβολή πίνακα users	57
Εικόνα 4.12: Προβολή πίνακα spot	58
Εικόνα 4.13: Προβολή πίνακα task.....	58
Εικόνα 4.14: Συσχετίσεις μεταξύ πινάκων	59
Εικόνα 4.15: Σχηματική απεικόνιση Administration	61

Εικόνα 4.16: Σελίδα σύνδεσης.....	62
Εικόνα 4.17: Σελίδα αίτησης εγγραφής	63
Εικόνα 4.18: Αρχική σελίδα Administration	63
Εικόνα 4.19: Σελίδα προβολής όλων των Spots.....	65
Εικόνα 4.20: Σελίδα δημιουργίας νέου Spot	65
Εικόνα 4.21: Σελίδα ενημέρωσης Spot	66
Εικόνα 4.22: Σελίδα αναζήτησης Spot	67
Εικόνα 4.23: Όροι αναζήτησης Spot	67
Εικόνα 4.24: Σελίδα διαγραφής Spot.....	68
Εικόνα 4.25: Σελίδα προβολής Spots σε Χάρτη	68
Εικόνα 4.26: Σελίδα προβολής όλων των Tasks	69
Εικόνα 4.27: Σελίδα δημιουργίας νέου Task.....	69
Εικόνα 4.28: Σελίδα ενημέρωσης Task.....	70
Εικόνα 4.29: Σελίδα επιλογής πεδίου ενημέρωσης	70
Εικόνα 4.30: Σελίδα αναζήτησης Task.....	71
Εικόνα 4.31: Όροι αναζήτησης Task.....	71
Εικόνα 4.32: Σελίδα διαγραφής Task	72
Εικόνα 4.33: Σελίδα αναζήτησης συντεταγμένων	73
Εικόνα 4.34: Σελίδα αναζήτησης διεύθυνσης	73
Εικόνα 4.35: Σελίδα Google Maps	74
Εικόνα 4.36: Σελίδα εγγραφής & απεγγραφής Χρήστη	75
Εικόνα 4.37: Σελίδα εγγραφής & απεγγραφής Διαχειριστή.....	75
Εικόνα 4.38: Δημιουργία Android Project.....	76
Εικόνα 4.39: Εισαγωγή στοιχείων Android Project	76
Εικόνα 4.40: Επιλογές Android Project	77
Εικόνα 4.41: Ρύθμιση εικονιδίου Android Εφαρμογής	77
Εικόνα 4.42: Επιλογή τύπου του Activity	78
Εικόνα 4.43: Εκκίνηση SDK Manager	79
Εικόνα 4.44: Αποδοχή όρων στο SDK Manager	79
Εικόνα 4.45: Εκκίνηση Android Virtual Device Manager.....	80
Εικόνα 4.46: Ρύθμιση νέου Virtual Device	80
Εικόνα 4.47: Προβολή διαθέσιμων Virtual Devices.....	81
Εικόνα 4.48: Εκκίνηση Virtual Device	81
Εικόνα 4.49: Εντοπισμός του Application.....	82
Εικόνα 4.50: Log-in Window της Εφαρμογής	82
Εικόνα 4.51: Εισαγωγή λανθασμένων στοιχείων	83
Εικόνα 4.52: Εμφάνιση του κεντρικού Χάρτη.....	83
Εικόνα 4.53: Εντοπισμός θέσης	84
Εικόνα 4.54: Εμφάνιση πληροφοριών τοποθεσίας	84
Εικόνα 4.55: Εξαγωγή Android Εφαρμογής.....	85
Εικόνα 4.56: Επιλογές για εξαγωγή Android Εφαρμογής.....	85
Εικόνα 4.57: Εντοπισμός του Debug Keystore	86
Εικόνα 4.58: Εισαγωγή κωδικού του Debug Keystore	86
Εικόνα 4.59: Επιλογή φακέλου εξαγωγής APK	87

1. Εισαγωγή

1.1 Περίληψη

Στόχος μας είναι η ανάπτυξη μίας εφαρμογής, η οποία απευθύνεται σε χρήστες που χρησιμοποιούν κινητές συσκευές με λειτουργικό σύστημα Android. Μέσα από αυτή την εφαρμογή ο χρήστης θα μπορεί, ενεργοποιώντας τη λειτουργία εντοπισμού θέσης (GPS) της συσκευής του, να εντοπίζει τη θέση του στο χάρτη και να λαμβάνει πληροφορίες σχετικές με τη τοποθεσία αυτή. Για παράδειγμα, αν κάποιος βρίσκεται στο χώρο του ΤΕΙ στο Ηράκλειο, θα λάβει πληροφορίες σχετικές με το ΤΕΙ, όπως τηλέφωνο επικοινωνίας κτλ. Τις πληροφορίες σχετικά με τις τοποθεσίες αυτές, θα τις αντλεί η εφαρμογή μας από μία βάση δεδομένων με ειδικά διαμορφωμένους πίνακες.

Παράλληλα με την εφαρμογή αυτή, θα αναπτυχθεί και μία άλλη η οποία θα βασίζεται στη σύνδεση στο διαδίκτυο (web-based) και θα αποτελεί την εφαρμογή διαχείρισης (Administration). Μέσα από το Administration θα μπορούν οι διαχειριστές, να επικοινωνούν και να χειρίζονται τα ερωτήματα προς τη βάση δεδομένων, να δημιουργούν και να ενημερώνουν τα σημεία ενδιαφέροντος που θα φαίνονται στην Android εφαρμογή και να διαχειρίζονται εργασίες που σχετίζονται με αυτά τα σημεία. Παράλληλα μέσω του Administration, θα γίνονται αιτήσεις μέσω φορμών προς τους διαχειριστές, που αφορούν την εγγραφή και την απεγγραφή ενός χρήστη ή διαχειριστή στην εφαρμογή. Οι αιτήσεις αυτές θα λαμβάνονται από τους διαχειριστές οι οποίοι μέσω του Administration θα προσθέτουν νέους χρήστες ή διαχειριστές. Οι μεν απλοί χρήστες θα μπορούν να χρησιμοποιήσουν την Android εφαρμογή, οι δε διαχειριστές θα μπορούν να χρησιμοποιήσουν τη web-based εφαρμογή του Administration.

Τέλος η βάση δεδομένων θα είναι προσπελάσιμη και από τις δύο εφαρμογές, στο μέτρο όμως που θα αναλογεί σε κάθε μία από αυτές. Εκεί θα βρίσκονται οι πίνακες για την αποθήκευση των σημείων ενδιαφέροντος, την αποθήκευση των εργασιών που πρέπει να γίνουν σε κάθε σημείο ενδιαφέροντος (και αφορούν τους διαχειριστές), πίνακες σχετικά με τους απλούς χρήστες και τους διαχειριστές. Ενώ θα δημιουργηθούν και οι κατάλληλες συσχετίσεις μεταξύ των πεδίων των πινάκων, ώστε να έχουμε το βέλτιστο αποτέλεσμα.

2. Μεθοδολογία

2.1 GPS (Global Positioning System)



Εικόνα 2.1: Λογότυπο GPS

Το γνωστό σε όλους μας GPS (Global Positioning System) ή αλλιώς Παγκόσμιο Σύστημα Εντοπισμού Θέσης ξεκίνησε, ως ένα σύστημα πλοήγησης που αναπτύχθηκε από το Υπουργείο Άμυνας των Η.Π.Α. Το σύστημα αυτό αρχικά ονομάστηκε σε NAVSTAR GPS (Navigation Signal Timing and Ranging Global Positioning System) και ρυθμίζεται καθημερινά από τη Βάση Πολεμικής Αεροπορίας Σκρίβερ (Schriever) με υψηλό κόστος υλοποίησης αλλά και συντήρησης το οποίο ανέρχεται στα 750 εκατομμύρια δολάρια το χρόνο.

Η λειτουργία του βασίζεται σε ένα δίκτυο πολυάριθμων δορυφόρων όπου βρίσκονται σε μεσαίου ύψους τροχιές γύρω από το πλανήτη και εκπέμπουν μεγάλης ακρίβειας ραδιοσήματα. Οι δορυφόροι αυτοί χωρίζουν τη γη σε πλέγματα στα οποία υπάρχουν οι κατάλληλες συσκευές, τις οποίες ονομάζουμε δέκτες GPS, αναλαμβάνοντας να εκπέμπουν στους δέκτες κωδικοποιημένες πληροφορίες στη περιοχή των μικροκυμάτων. Με το τρόπο αυτό οι δέκτες μπορούν να καθορίζουν τη θέση τους, τη ταχύτητά τους, το υψόμετρο όπως επίσης και τη κατεύθυνση κίνησής τους με ακριβή τρόπο. Ουσιαστικά το σύστημα μπορεί να παρέχει στους χρήστες του το ακριβές στίγμα ενός σημείου οπουδήποτε στο κόσμο, ενώ σε συνδυασμό με ειδικό λογισμικό χαρτογράφησης μπορούν να απεικονιστούν γραφικά αυτές οι πληροφορίες. Η ακρίβεια των πληροφοριών που παρέχονται σχετικά με το προσδιορισμό θέσης είναι κατά μέσο όρο της τάξης των 15 τετραγωνικών μέτρων. Ο προσδιορισμός ενός σημείου στο χώρο τριών διαστάσεων και της ταχύτητας κίνησης του σε κάθε περιοχή στην ξηρά, τη θάλασσα και τον αέρα, αφορά κάθε χρονική στιγμή μέσα στο εικοσιτετράωρο κάτω από οποιοδήποτε συνθήκες καιρού και κάτω από οποιοδήποτε συνθήκες κίνησης στο δίκτυο των δορυφόρων.

Το Global Positioning System στη σημερινή του μορφή βασίζεται σε παλαιότερες μεθόδους που αξιοποιούσαν τη τεχνολογία των ηλεκτρομαγνητικών κυμάτων καθώς επίσης και την παρατήρηση ενός τεχνητού ουράνιου αντικειμένου. Οι σταθμοί βάσης οι οποίοι λαμβάνουν και δέχονται τα απαραίτητα ηλεκτρομαγνητικά κύματα δεν είναι πλέον επίγειοι, αλλά βρίσκονται σε δορυφόρους.

Το GPS αρχικά δημιουργήθηκε μόνο για στρατιωτική χρήση και ανήκε εξολοκλήρου στο Αμερικανικό Υπουργείο Εθνικής Άμυνας. Στην δεκαετία του 1960 το σύστημα δορυφορικής πλοήγησης, γνωστό μέχρι τότε με την ονομασία Transit System, χρησιμοποιήθηκε ειδικά από το αμερικανικό ναυτικό. Απαιτήθηκαν πολλές δεκαετίες, μέχρι και τα μέσα της δεκαετίας του 1990, ούτως ώστε το σύστημα GPS να εξελιχθεί, να γίνει πιο ακριβές στον εντοπισμό και να αρχίσει να διατίθεται για ελεύθερη χρήση στο ευρύ κοινό.

Σήμερα το GPS παρέχει ένα πλήθος εφαρμογών, όπως η χρήση του τόσο στο τομέα των μεταφορών όσο και για χαρτογραφικούς, τοπογραφικούς και άλλους επιστημονικούς λόγους. Τα πλέον γνωστά συστήματα πλοήγησης αυτοκινήτων που ήδη χρησιμοποιούνται περιλαμβάνουν απλές

λειτουργίες με ψηφιακούς χάρτες. Εκτός όμως από τα παραπάνω η μεγαλύτερη καινοτομία είναι η ενσωμάτωση των δεκτών GPS σε κινητές συσκευές όπως smartphones, tablets και φωτογραφικές μηχανές. Έτσι ως απλοί χρήστες έχουμε τη δυνατότητα να ελέγξουμε τη θέση μας ανά πάσα στιγμή πάνω σε ένα ψηφιακό χάρτη μέσω των κινητών μας συσκευών, ενώ μπορούμε επίσης να λάβουμε πληροφορίες για το πως θα πάμε στο προορισμό μας και ποιες διαδρομές έχουν τα λιγότερα κυκλοφοριακά προβλήματα.

Λειτουργικά τμήματα

Το σύστημα εντοπισμού θέσης GPS αποτελεί ένα παγκόσμιο δίκτυο, με τέτοιο μέγεθος εμβέλειας που είναι σε θέση να καλύπτει την ξηρά την θάλασσα αλλά και τον αέρα. Λόγω αυτής της εκτεταμένης περιοχής κάλυψής του, είναι απαραίτητος ο διαχωρισμός του σε επιμέρους τμήματα όπου πραγματοποιούνται οι λειτουργίες του αλλά και επίσης ο συντονισμός του. Αναλυτικά, τα τμήματα αυτά είναι:

α) Δορυφορικό Τμήμα.

Αποτελεί το κέντρο του παγκόσμιου συστήματος εντοπισμού θέσης και απαρτίζεται από ένα δίκτυο 32 δορυφόρων. Ο αριθμός των ενεργών δορυφόρων διατηρείται σταθερός αφού οι παλαιότεροι δορυφόροι οι οποίοι είχαν προβλήματα φθοράς, έχουν αντικατασταθεί με νέους ώστε να εξασφαλιστεί η σωστή και απρόσκοπτη λειτουργία του συστήματος. Οι δορυφόροι αυτοί καλύπτουν ομοιόμορφα με το σήμα τους ολόκληρο τον πλανήτη, γεγονός που δείχνει τη διαθεσιμότητά του Συστήματος Εντοπισμού Θέσης σε κάθε σημείο της Γης. Το δίκτυο των δορυφόρων που χρησιμοποιούνται αποτελείται από 24 ενεργούς δορυφόρους εντοπισμού θέσης και 8 επιπρόσθετους οι οποίοι χρησιμοποιούνται για να παρέχουν μεγαλύτερη ακρίβεια στους υπολογισμούς των δεκτών και άλλες απαραίτητες μετρήσεις.

Όλοι οι δορυφόροι βρίσκονται σε ύψος 20.200 χιλιομέτρων πάνω από την επιφάνεια της θάλασσας και η ακτίνα περιστροφής τους είναι 26.600 χιλιόμετρα με κλίση 55 μοιρών σε σχέση με τον ισημερινό, ενώ εκτελούν δύο περιστροφές γύρω από τη Γη κάθε 24ωρο. Οι τροχιές είναι επιλεγμένες έτσι ώστε τουλάχιστον 6 δορυφόροι να είναι πάντα ορατοί από σχεδόν κάθε σημείο της επιφάνειας της γης, έχουν ως κέντρο τους το κέντρο της γης και η ταχύτητα περιστροφής των δορυφόρων είναι 7.000 μίλια ανά ώρα. Η τροφοδοσία των δορυφόρων πραγματοποιείται με ηλεκτρική ενέργεια μέσω των φωτοβολταϊκών συστημάτων που διαθέτουν. Κάθε δορυφόρος εκπέμπει σήματα έντασης περίπου 20-50 Watts, πράγμα που σημαίνει ότι λόγω της χαμηλής του έντασης και της μεγάλης απόστασης των δορυφόρων από τη γη, είναι σημαντικό να υπάρχει άμεση οπτική επαφή του δέκτη με τον ουρανό κατά τη χρήση του συστήματος.

β) Επίγειο Τμήμα Ελέγχου.

Οι δορυφόροι, είναι πιθανό να αντιμετωπίσουν προβλήματα ανά πάσα στιγμή στη σωστή λειτουργία τους. Άρα μία κύρια εργασία που επιτελεί το Επίγειο Τμήμα Ελέγχου, είναι η εξασφάλιση της σωστής και χωρίς διακοπές λειτουργία του Συστήματος Εντοπισμού Θέσης. Οι έλεγχοι που πραγματοποιούνται σε αυτούς αφορούν στη σωστή τους ταχύτητα, υψόμετρο και επίσης στην επάρκεια της ηλεκτρικής τους ενέργειας. Παράλληλα, εφαρμόζονται όλες οι κατάλληλες ενέργειες που αφορούν στο σύστημα χρονομέτρησης των δορυφόρων, έτσι ώστε να αποτρέπεται η παροχή λάθος πληροφοριών στους χρήστες. Το τμήμα επίγειου ελέγχου αποτελείται τόσο από ένα

επανδρωμένο όσο και από τέσσερα μη επανδρωμένα κέντρα τα οποία είναι εγκατεστημένα σε ισάριθμες περιοχές του πλανήτη. Οι περιοχές αυτές είναι οι εξής:

α) Κολοράντο (ΗΠΑ), β) Χαβάη (Ανατολικός Ειρηνικός Ωκεανός), γ) Ascension Island (Ατλαντικός Ωκεανός), δ) Diego Garcia (Ινδικός Ωκεανός), ε) Kwajalein (Δυτικός Ειρηνικός Ωκεανός).

Όλοι οι σταθμοί είναι εξοπλισμένοι με υψηλής ακρίβειας δέκτες καθώς και Ατομικούς Χρονομετρητές κατασκευασμένους από Καίσιο.

Συνολικά το Επίγειο Τμήμα Ελέγχου της Υπηρεσίας Εντοπισμού Θέσης, εκτελεί και βασικές λειτουργίες απαραίτητες για τη λειτουργία της υπηρεσίας οι οποίες είναι αυτές της τηλεμετρίας και ιχνηλάτησης. Οι έλεγχοι αυτοί εστιάζονται στον έλεγχο της σωστής τροχιάς των δορυφόρων, της ακεραιότητας του συστήματος, στη διατήρηση ημερολογίων όσον αφορά τη κατάσταση των δορυφόρων και τέλος τον έλεγχο για τη σωστή λειτουργία των χρονομετρητών που διαθέτουν οι δορυφόροι και άλλοι πολλοί έλεγχοι.

γ) Τμήμα Τελικού Χρήστη.

Απαρτίζεται από τους χρήστες δεκτών GPS ανά την υφήλιο και πιο συγκεκριμένα οι δέκτες αυτοί μπορούν να χρησιμοποιηθούν τόσο κατά τη διάρκεια μιας απλής πεζοπορίας, όσο και σε οχήματα και συνήθως διαθέτουν αρκετά μικρές διαστάσεις. Για να προσφέρουν ακόμη περισσότερες πληροφορίες, οι δέκτες αυτοί συνδυάζονται με ειδικό λογισμικό το οποίο προβάλλει ένα χάρτη στην οθόνη της συσκευής GPS. Πρόκειται ουσιαστικά, για λογισμικό το οποίο λαμβάνει από τους δορυφόρους τις πληροφορίες για το στίγμα του σημείου πάνω στο οποίο βρίσκεται ο δέκτης και τις μετατρέπει σε κατανοητή μορφή, πληροφορώντας με αυτό τον τρόπο τον χρήστη για την ακριβή γεωγραφική του θέση. Επιπλέον ένα σημαντικό πεδίο χρήσης των δεκτών GPS είναι αυτό των Μέσων Μαζικής Μεταφοράς, όπως τη ναυτιλία, την αεροπορία, τα τρένα, αλλά και τα λεωφορεία. Στις περιπτώσεις αυτές, το σύστημα αυτό χρησιμοποιείται ως βοηθητικό μέσο, παράλληλα με τα άλλα μέσα που διατίθενται κατά περίπτωση και έτσι μπορούν οι χειριστές τους να υπολογίζουν τη ταχύτητα, το στίγμα, το ύψος και τη κατεύθυνση που διατηρούν.

2.2 Φορητές συσκευές GPS

Η μεγάλη εξάπλωση της χρήσης του GPS οφείλεται στη κυκλοφορία και διάθεση ολοένα και περισσότερο οικονομικά προσιτών, φορητών δεκτών GPS τόσο για χρήση από πεζούς όσο και χρήση σε οχήματα. Επίσης οφείλεται και στην εξάπλωση της χρήσης κινητών συσκευών όπως Smartphones, PDAs και Tablets, που ενσωματώνουν κάποιο δέκτη GPS, ως πλοηγούς. Ένας φορητός δέκτης αποτελείται από:

- α) Την εσωτερική δορυφορική κεραία, στην οποία ανατίθενται η λήψη του σήματος από τους GPS δορυφόρους με τους οποίους θα πρέπει να βρίσκεται σε οπτική επαφή, για βέλτιστα αποτελέσματα.
- β) Το δέκτη GPS, ο οποίος χρησιμοποιεί κυκλώματα που παρουσιάζουν πολύ μικρό θόρυβο και διαθέτει ειδικές τεχνικές επεξεργασίας σήματος έτσι ώστε να μπορεί να διαχωρίζει τα ασθενή σήματα από τους δορυφόρους, από τον ισχυρό τηλεπικοινωνιακό θόρυβο ο οποίος έχει τη μορφή τυχαίου σήματος.
- γ) Τον κυρίως μικροελεγκτή, την οθόνη απεικόνισης και το υπόλοιπο υλικό μέρος για την επικοινωνία με το χρήστη της συσκευής.



Εικόνα 2.2: Χρήση GPS Navigation σε Smartphone

2.3 Geo-fence (γεωγραφικό πλέγμα)

Ως γεωγραφικό πλέγμα ή αλλιώς Geo-fence ορίζουμε μια ψηφιακή περίμετρο η οποία αντιστοιχεί σε συγκεκριμένη γεωγραφική περιοχή. Ένας άλλος ορισμός σχετικά με το γεωγραφικό πλέγμα μπορεί να είναι μία προκαθορισμένη συνοριακή γραμμή όπου σχηματίζεται ανάμεσα σε συγκεκριμένα σημεία πάνω στο χάρτη, όπως για παράδειγμα η περιοχή του προαυλίου ενός σχολείου ή η συνοριακή γραμμή μίας γειτονιάς ή ακόμα και μεγαλύτερες περιοχές όπως πόλεις, χωριά κ.α.

Τα κατά περίπτωση οριζόμενα γεωγραφικά πλέγματα χρησιμοποιούνται από το 2004 όπου και αναπτύχθηκαν από το Dr. Vinay Rawlani στο Πανεπιστήμιο του Μισούρι-Κολούμπια, για χρήσεις πολλαπλών ψηφιακών χαρτογραφήσεων. Η μόνη προϋπόθεση για να μπορεί να χρησιμοποιηθεί η υπηρεσία αυτή από τους απλούς χρήστες, είναι να έχουν μία κινητή συσκευή η οποία να διαθέτει κάποιο δέκτη εντοπισμού, όπως GPS, για πρόσβαση σε Υπηρεσίες Βασισμένες στη Τοποθεσία (Location-Based Services). Όταν ο χρήστης λοιπόν της κινητής συσκευής εισέρχεται ή βρίσκεται ήδη σε μία περιοχή στην οποία έχει οριστεί κάποιο Geo-fence, η συσκευή του θα λάβει αυτόματα μία καθορισμένη ειδοποίηση. Η ειδοποίηση αυτή μπορεί να περιέχει πληροφορίες για τη συγκεκριμένη περιοχή που βρίσκεται ο χρήστης με τη συσκευή του, ενώ μπορεί να σταλεί είτε μέσω των δικτύων κινητής τηλεφωνίας είτε μέσω email.

Οι χρήσεις της υπηρεσίας Geofencing είναι πάρα πολλές και βρίσκουν εφαρμογή από κρίσιμα συστήματα έως και τις ανάγκες Marketing των επιχειρήσεων. Ένα παράδειγμα χρήσης της υπηρεσίας αυτής σε κρίσιμα συστήματα είναι αυτό της τηλεματικής υλικού και λογισμικού. Αυτό σημαίνει ότι επιτρέπει στους χρήστες της υπηρεσίας Geo-fence να σχεδιάζουν ζώνες γύρω από τις περιοχές εργασίας τους, στα σημεία των πελατών τους και σε περιοχές ασφαλείας. Αυτά τα Geo-fences όταν παραβιαστούν από οχήματα ή ανθρώπους εξοπλισμένα με κινητές συσκευές που κάνουν χρήση συστημάτων εντοπισμού θέσης, μπορούν να ενεργοποιήσουν μία προειδοποίηση σε κάποιο χρήστη ή διαχειριστή του συστήματος ασφαλείας με χρήση Geo-fences μέσω SMS ή Email. Άλλες εφαρμογές περιλαμβάνουν την αποστολή ειδοποίησης όταν ένα όχημα κλαπεί ή μπορεί να ειδοποιεί δασοφύλακες αν τα άγρια ζώα μένουν μέσα στις καθορισμένες περιοχές.

Η χρήση γεωγραφικών πλεγμάτων σαν μοντέλα στρατηγικής ασφάλειας, μπορεί να παρέχει ασφάλεια σε ασύρματα τοπικά δίκτυα. Αυτό γίνεται χρησιμοποιώντας προκαθορισμένα όρια π.χ. ένα γραφείο με όρια που έχουν καθοριστεί μέσω της τεχνολογίας εντοπισμού σε ένα ειδικά διαμορφωμένο εξυπηρετητή. Με αυτό το τρόπο το γραφείο αυτό γίνεται μία τοποθεσία μόνο για εξουσιοδοτημένους χρήστες και ασύρματες κινητές συσκευές. Τέλος, λόγω της αυξανόμενης δημοτικότητας της διαφήμισης σε κινητές συσκευές, τα Geo-fences χρησιμοποιούνται ώστε να αποστέλλονται στοχευμένες διαφημίσεις στις κινητές συσκευές των πελατών σύμφωνα με τη τοποθεσία τους μέσα στα προκαθορισμένα γεωγραφικά πλέγματα.

Εδώ παρατίθενται μερικές από τις πιο δημοφιλείς χρήσεις των Geo-fences:

Διαχείριση στόλου:

Όταν για παράδειγμα ένα φορτηγό ξεφεύγει από τη προκαθορισμένη διαδρομή, ο διαχειριστής λαμβάνει μία σχετική ειδοποίηση.

Διαχείριση ανθρώπινων πόρων:

Η έξυπνη κάρτα του εργαζόμενου μπορεί να στείλει μία ειδοποίηση στην ασφάλεια όταν εκείνος επιχειρήσει να εισέλθει σε μία μη εγκεκριμένη για εκείνον περιοχή.

Έλεγχος της συμμόρφωσης με κανονισμού:

Ένα δίκτυο έχει τη δυνατότητα να κρατάει αρχείο καταγραφής της θέσης κινητών συσκευών σε σχέση με τα geo-fences που έχουν καθοριστεί με σκοπό τη διασφάλιση της σωστής χρήσης των συσκευών αυτών και της συμμόρφωση με τους κανονισμούς.

Επιβολή νόμου:

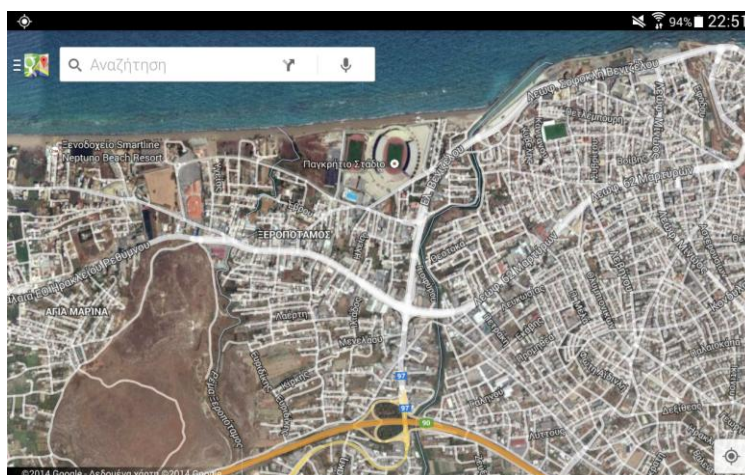
Ένα βραχιόλι τοποθετημένο στον αστράγαλο ενός ατόμου που του έχει επιβληθεί κατ'οίκον περιορισμός, μπορεί να ειδοποιήσει τις αρχές αν το άτομο αυτό επιχειρήσει να φύγει από το σημείο που έχει περιοριστεί.

2.4 Google Maps

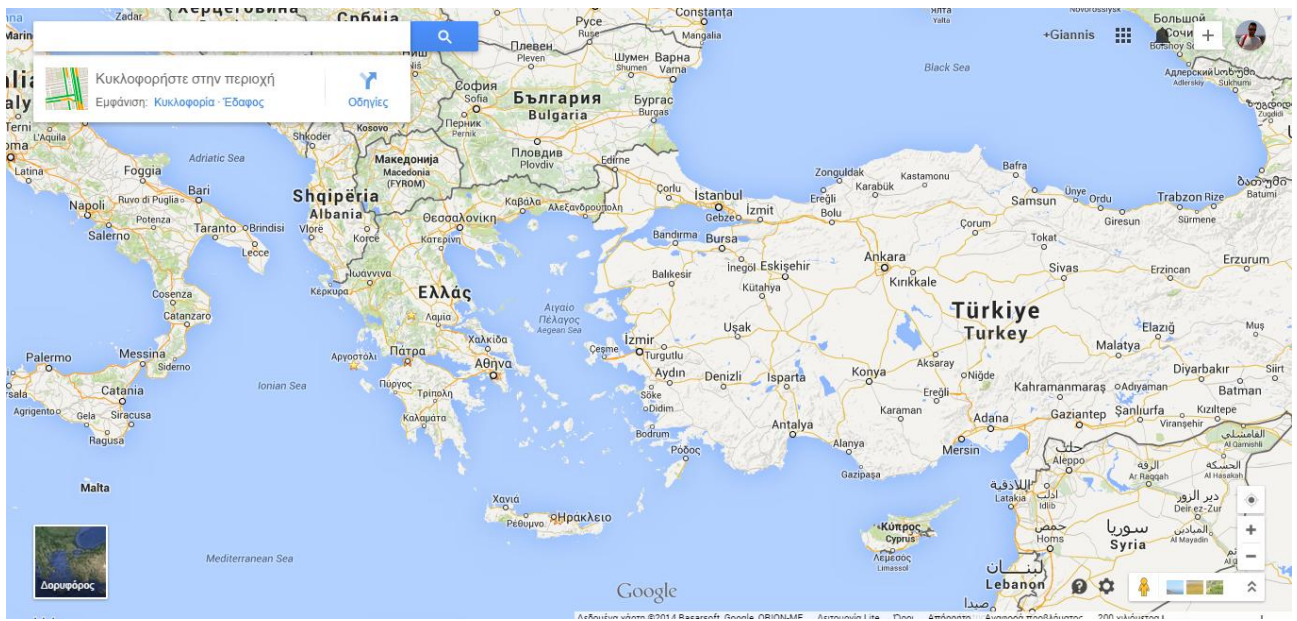
Το Google Maps είναι μία υπηρεσία ψηφιακής χαρτογράφησης η οποία παρέχεται δωρεάν στο διαδίκτυο. Η υπηρεσία αυτή υπάρχει σε μορφή ιστοτόπου στο <https://maps.google.com> και σε μορφή δωρεάν εφαρμογής για Android συσκευές με όνομα “Google Maps”. Σύμφωνα με έρευνες η εφαρμογή “Google Maps” για κινητές συσκευές έχει το μεγαλύτερο αριθμό λήψεων με το ποσοστό των χρηστών που έχει χρησιμοποιήσει την εφαρμογή τουλάχιστον μία φορά έως και τον Άγουστο του 2013 να είναι πάνω από το 54% παγκοσμίως. Η χαρτογράφηση και η ανάπτυξη της εφαρμογής αυτής έγινε αποκλειστικά από τη Google προσφέροντας εκτός από ένα απλό ψηφιακό χάρτη διάφορες υπηρεσίες. Οι υπηρεσίες αυτές περιλαμβάνουν δυνατότητα επιλογής από το χρήστη το πως ακριβώς θα του εμφανιστεί ο χάρτης, αλλά και δυνατότητα να προγραμματίσει κάποια διαδρομή.

Σε ότι αφορά τους τρόπους εμφάνισης του χάρτη, ο χρήστης έχει την επιλογή να βλέπει τη γη όπως αν ήταν σε κάποιο δορυφόρο, τη δυνατότητα να βλέπει έναν οδικό χάρτη περιλαμβάνοντας και τοποθεσίες ενδιαφέροντος αλλά και με μία νέα υπηρεσία με όνομα “Google Street View” θα μπορεί να “περπατήσει” εικονικά σε οποιαδήποτε περιοχή έχει εικονογραφηθεί από κάποια ειδικά εξοπλισμένα οχήματα της Google αλλά και να κινηθεί μέσα σε κάποια συγκεκριμένα κτήρια ιστορικού ή γενικού ενδιαφέροντος ανά το κόσμο. Τα μέρη του κόσμου όπου έχουν τη δυνατότητα αυτή γίνονται όλο και περισσότερα συμπεριλαμβανομένης και της Ελλάδας. Για τη περίπτωση της υπηρεσίας “Google Street View” αλλά και της δορυφορικής προβολής του χάρτη, γίνεται συχνή ανανέωση των εικονογραφήσεων ώστε να είναι όσο το δυνατόν πιο πρόσφατες ενώ η παλαιότερη εικονογράφηση δε ξεπερνάει τα 3 χρόνια.

Με την υπηρεσία “οδηγίες” μπορεί κάποιος να επιλέξει οποιοδήποτε σημείο πάνω στο ψηφιακό χάρτη και να λάβει οδηγίες για το πως θα φτάσει εκεί είτε περπατώντας, είτε χρησιμοποιώντας τα μέσα μαζικής μεταφοράς, είτε το αυτοκίνητό του, είτε το ποδήλατό του. Αυτές οι υπηρεσίες είναι διαθέσιμες και μέσω του ιστοτόπου αλλά και μέσω της εφαρμογής για κινητές συσκευές. Ειδικά στη περίπτωση των κινητών συσκευών μπορεί ο χρήστης να ενεργοποιήσει τη δυνατότητα εντοπισμού της θέσης του και να λάβει οδηγίες για ένα συγκεκριμένο σημείο που τον ενδιαφέρει με σημείο αφετηρίας το σημείο που βρίσκεται. Άρα οι χρήστες των κινητών συσκευών μπορούν να βρουν ανά πάσα στιγμή γεωγραφικά σε ποιο σημείο βρίσκονται και να μάθουν πως θα πάνε σε κάποια συγκεκριμένη διεύθυνση και να δουν πως είναι στη πραγματικότητα εκεί πριν ακόμα φτάσουν.



Εικόνα 2.3: Εφαρμογή Google Maps για κινητές συσκευές



Εικόνα 2.4: Ιστοσελίδα υπηρεσίας Google Maps

2.5 Υπηρεσίες Εντοπισμού Θέσης (Location-based services)

Οι υπηρεσίες εντοπισμού θέσης (στο εξής LBS από την Αγγλική της ονομασία) είναι μία γενική κλάση υπηρεσιών της πληροφορικής, σε προγραμματιστικό επίπεδο, η κύρια λειτουργία της οποίας είναι η χρήση δεδομένων τοποθεσίας με σκοπό τη διαχείριση κάποιων χαρακτηριστικών. Για να εντοπιστεί η τοποθεσία της κινητής συσκευής θα πρέπει να μπορεί εκείνη να έρθει σε επαφή με τον παροχέα της υπηρεσίας εντοπισμού θέσης.

Αν μία συσκευή δε διαθέτει κάποιο δέκτη GPS, τότε η επικοινωνία με το παροχέα της υπηρεσίας γίνεται μέσω του κοντινότερου επίγειου σταθμού βάσης του δικτύου κινητής τηλεφωνίας. Επειδή τα σήματα σε αυτή τη περίπτωση στέλνονται μέσω δικτύου που η κύρια λειτουργία του είναι να μεταφέρει σήματα που περιέχουν φωνή, είναι λογικό να παρουσιάζεται μία σημαντική καθυστέρηση στον εντοπισμό θέσης. Τα LBS χρησιμοποιούν ένα σταθμό βάσης, ενώ παρουσιάζουν και μία ακτίνα όπου ο εντοπισμός θέσης δε θα είναι ακριβής γύρω από αυτό το σταθμό βάσης. Αυτή η τεχνική χρησιμοποιείται ακόμα από τις υπηρεσίες της αστυνομίας για να εντοπίζει κινητά τηλέφωνα ως μέτρο ασφαλείας, όταν αυτά για παράδειγμα έχουν χρησιμοποιηθεί για κάποια αξιόποινη πράξη. Τα πιο εξελιγμένα κινητά τηλέφωνα και tablets σήμερα διαθέτουν ένα ολοκληρωμένο δέκτη GPS.

Ένας ακόμα δημοφιλής τρόπος επικοινωνίας με κάποιο παροχέα LBS είναι μέσω του παγκόσμιου συστήματος για κινητές τηλεπικοινωνίες (GSM). Αυτό λειτουργεί εντοπίζοντας τη τοποθεσία της κινητής συσκευής σε σχέση με το σε ποια κυψέλη του δικτύου βρίσκεται η συσκευή άρα ποιος σταθμό βάσης είναι πιο κοντά, ενώ η τοποθεσία υπολογίζεται με τεχνικές όπως η χρονική καθυστέρηση λήψης σήματος από το σταθμό βάσης υπολογίζοντας έτσι την απόσταση από εκείνο.

Στη περίπτωση όπου η κινητή συσκευή διαθέτει δέκτη GPS τα πράγματα είναι πολύ πιο απλά καθώς ο εντοπισμός της συσκευής του γίνεται μέσω του δικτύου δορυφόρων θεσιθεσίας. Τα κύρια πλεονεκτήματα αυτής της τεχνικής είναι η μεγάλη ακρίβεια στον εντοπισμό, οι διάφοροι υπολογισμοί όπου διατίθενται στο χρήστη όπως για παράδειγμα η ταχύτητα, το υψόμετρο και η κατεύθυνση της κίνησής του καθώς και η ταχύτητα με την οποία γίνεται ο εντοπισμός της συσκευής από τους δορυφόρους σε σχέση με τις υπόλοιπες τεχνικές.

Λόγω του ότι το LBS είναι μία υπηρεσία παροχής πληροφοριών τοποθεσίας έχει ένα πλήθος εφαρμογών στα μέσα κοινωνικής δικτύωσης ως μία υπηρεσία ψυχαγωγίας όπου είναι προσβάσιμη από κινητές συσκευές μέσω των δικτύων κινητής τηλεφωνίας και η οποία συλλέγει και χρησιμοποιεί πληροφορίες σχετικές με τη γεωγραφική θέση της κινητής συσκευής. Αυτού του τύπου οι υπηρεσίες γίνονται όλο και πιο σημαντικές λόγω της εξάπλωσης της χρήσης συσκευών tablets και smartphones. Αυτό συμβαίνει γιατί ο καθένας από εμάς μπορεί να χρησιμοποιήσει το smartphone ή το tablet του σαν συσκευή GPS για να εντοπίσει κάποιο μέρος σε μία πόλη που επισκέπτεται για πρώτη φορά ή ακόμα και να κοινοποιήσει σε κάποιο κοινωνικό δίκτυο τη παρουσία του για παράδειγμα σε κάποιο σημείο ψυχαγωγικού ενδιαφέροντος ώστε να το δουν οι φίλοι του.

Τα παραπάνω παραδείγματα δεν αποτελούν κάτι παραπάνω από δύο μόνο τομείς εφαρμογής των υπηρεσιών βασισμένων στη τοποθεσία. Αυτό σημαίνει ότι τα LBS έχουν εφαρμογή σε αμέτρητους τομείς της σύγχρονης ζωής όπως στην υγεία, στην αναζήτηση αντικειμένων μέσα σε ένα χώρο, στη ψυχαγωγία και στον εντοπισμό τοποθεσιών (όπως αναφέρθηκε στη προηγούμενη παράγραφο), στην εργασία, στη προσωπική ζωή, στην ενημέρωση, στην ασφάλεια αντικειμένων και προσώπων και σε πολλούς άλλους τομείς. Τα LBS περιλαμβάνουν υπηρεσίες αναγνώρισης της τοποθεσίας ενός προσώπου ή ενός αντικειμένου, όπως για παράδειγμα μπορεί κάποιος να

χρησιμοποιήσει το smartphone του και ενεργοποιώντας τη πρόσβαση στις υπηρεσίες εντοπισμού θέσης να μπορεί να βρει πού βρίσκεται το πιο κοντινό σε εκείνον ATM, βάσει της τοποθεσίας που βρίσκεται τη συγκεκριμένη χρονική στιγμή. Επίσης μπορεί κάποιος να εντοπίσει τη τοποθεσία κάποιου φίλου ή συνεργάτη του, ενώ και οι δύο χρησιμοποιούν συσκευές με δυνατότητα εντοπισμού θέσης. Μία ακόμα υπηρεσία που παρέχεται από τα LBS είναι αυτή της δυνατότητας παρακολούθησης της διαδρομής ενός δέματος ή ενός οχήματος, ειδικά το δεύτερο πολύ σημαντικό σε περιπτώσεις εταιριών με στόλο οχημάτων όπου θα πρέπει να ακολουθούν συγκεκριμένη διαδρομή. Ένας σημαντικός και συνεχώς εξελισσόμενος τομέας του Marketing των επιχειρήσεων είναι οι διαφημίσεις σε κινητές συσκευές, από τα LBS δε θα μπορούσε να λείπει η παροχή αυτής της υπηρεσίας καθώς ανάλογα με τη τοποθεσία του χρήστη της κινητής συσκευής θα μπορούν να του παρουσιάζονται ανάλογες διαφημίσεις π.χ. για εστιατόρια, τράπεζες, καταστήματα και άλλες υπηρεσίες που βρίσκονται κοντά σε εκείνον. Άλλες χρήσιμες υπηρεσίες των LBS είναι η πληροφορίες σχετικά με τις καιρικές συνθήκες που επικρατούν στη συγκεκριμένη τοποθεσία όπως και προβλέψεις για το επόμενο διάστημα, ενημέρωση σχετικά με τη κίνηση στους δρόμους και τις οδικές αρτηρίες που παρουσιάζουν τη μεγαλύτερη κίνηση ώστε να τις αποφύγει ο χρήστης, εμφάνιση των κοντινότερων νοσοκομείων, πρατηρίων βενζίνης και φαρμακείων, όπως επίσης και ειδησεογραφική ενημέρωση βασισμένη στη τοποθεσία του χρήστη.

Όλα τα παραπάνω αποτελούν παραδείγματα των ονομαζόμενων τηλεπικοινωνιών σύγκλισης, όμως αυτή η τεχνολογία δεν είναι συμβατή με τα Συστήματα Εντοπισμού Θέσης Πραγματικού Χρόνου (Real Time Locating Systems – RTLS) και τις σχετικές με αυτό υπηρεσίες, καθώς όλες οι παραπάνω πληροφορίες μπορεί να είναι διαθέσιμες πολύ καιρό νωρίτερα και δε παρέχει ενημέρωση πραγματικού χρόνου σχετικά με τις κοντινές προς το χρήστη τοποθεσίες. Πέρα από αυτό όμως μία συσκευή μπορεί από μόνη της να εντοπιστεί και να παρέχει τη δυνατότητα παρακολούθησής της σε πραγματικό χρόνο.

3. Σχέδιο Δράσης

3.1 Ανάπτυξη του Administration

3.1.1 Τι είναι η HTML5



Εικόνα 3.1: Λογότυπο HTML5

Η HTML είναι ακρωνύμιο του αγγλικού HyperText Markup Language δηλαδή, Γλώσσα Σήμανσης Υπερκειμένου και είναι η κύρια γλώσσα με την οποία δημιουργούνται οι ιστοσελίδες, ενώ τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων. Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες (html tags) και περικλείονται μέσα σε σύμβολα "<" και ">" (για παράδειγμα <body>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <html> και </html>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες, φόρμες κλπ τα λεγόμενα δηλαδή στοιχεία της HTML που αναφέρθηκαν παραπάνω.

Τα κύρια tags ενός HTML εγγράφου είναι τα εξής:

<html> --> δήλωση της γλώσσας σήμανσης που θα χρησιμοποιηθεί
<head> --> αποτελεί την κεφαλίδα και ορίζουμε απαραίτητα χαρακτηριστικά του εγγράφου, όπως σενάρια κώδικα χωρίς μεταγλώτιση (scripts), ενημέρωση του browser για το που θα βρει κάποια stylesheets όπως CSS που θα δούμε παρακάτω, ορισμός τίτλου όπως φαίνεται στη μεθεπόμενη γραμμή κώδικα, κωδικοποίηση κειμένου και άλλα.
<meta charset="utf-8"> -->ορισμός κωδικοποίησης κειμένου
<title>Example</title> -->ορισμός κειμένου που θα εμφανίζεται στη γραμμή τίτλου του προγράμματος περιήγησης (web browser), εδώ: "example"
</head> --> κλείσιμο tag κεφαλίδας
<body>--> κύριο σώμα του εγγράφου, εδώ βάζουμε κείμενο, εικόνες, βίντεο, φόρμες, ήχους και γενικά το περιεχόμενο του HTML εγγράφου, που θα μεταφράσει ο web browser σε αναγνώσιμη ιστοσελίδα
</body>--> κλείσιμο tag κυρίου σώματος εγγράφου
</html> --> κλείσιμο tag εγγράφου HTML.

Η HTML5 είναι η επόμενη έκδοση της HTML 4.01 και ο στόχος της δημιουργίας της είναι η δυνατότητα υποστήριξης και ενσωμάτωσης στη γλώσσα των πιο πρόσφατων πολυμεσικών πρόσθετων. Για αυτό το λόγο στην HTML5 υπάρχει η δυνατότητα πρόσθεσης των tags: <video>, <audio> και <canvas> ενώ σχεδιάστηκαν και τα tags: <section>, <article>, <header>, <nav> έτσι ώστε να εμπλουτιστεί το σημασιολογικό περιεχόμενο των εγγράφων. Επίσης έχει δοθεί ιδιαίτερη έμφαση στη σημαντικότητα της ενσωμάτωσης στην HTML σελίδα scripts από το Document Object Model (DOM), όπως είναι η JavaScript. Έτσι πλέον γίνεται υποστήριξη της πιο εύκολης

ενσωμάτωσης κώδικα JavaScript στην HTML σελίδα, και μάλιστα γίνεται και καλύτερη διαχείρισή του από το browser μέσω της HTML5. Τέλος ένα πολύ σημαντικό χαρακτηριστικό της HTML5 είναι η υποστήριξη της καλύτερης συμπεριφοράς της σελίδας σε περιπτώσεις κινητών συσκευών όπως tablets και smartphones και καλύτερη υποστήριξη των σεναρίων DOM σε αυτές τις συσκευές. Έτσι μέσα από την HTML5 γίνεται να σχεδιάσει κάποιος πως ακριβώς θα εμφανίζεται η σελίδα του τόσο σε κανονικές οθόνες αλλά και σε κινητές συσκευές.

Η λειτουργία ενός web browser είναι να διαβάζει τα έγγραφα HTML και να τα συνθέτει, μέσω των ετικετών που περιέχουν, σε σελίδες κατανοητές από τον απλό χρήστη ώστε εκείνος να μπορεί να διαβάσει ή να ακούσει το περιεχόμενό τους. Ο browser όπως είναι φυσικό δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας. Άρα πιο απλά η HTML είναι μία γλώσσα με την οποία δημιουργεί ο σχεδιαστής ιστοσελίδων μία σελίδα γεμάτη από οδηγίες προς το web browser του αναγνώστη της, ο οποίος browser αναλαμβάνει να τις μεταφράσει και να μετατρέψει τη σελίδα οδηγιών σε ιστοσελίδα κατανοητή από τον αναγνώστη, ενσωματώνοντας παράλληλα όλα όσα έχει προσθέσει ο σχεδιαστής της.

Η HTML επιτρέπει την ενσωμάτωση διάφορων αντικειμένων μέσα στη σελίδα, όπως για παράδειγμα μία σελίδα με εικόνες, κείμενο, ήχους, βίντεο και διαδραστικές φόρμες. Στη τελευταία περίπτωση των διαδραστικών φορμών, η HTML μπορεί να συνεργαστεί άριστα με κώδικα σε PHP για την επικοινωνία με μία βάση δεδομένων, οπότε δίνετε έτσι μία ολοκληρωμένη λύση επικοινωνίας με μία βάση δεδομένων μέσω φόρμας. Εκτός από τα αντικείμενα όπου επιτρέπει η HTML, επιτρέπεται να προστεθούν κάποια tags σε ότι αφορά τη μορφοποίηση του εμφανιζόμενου κειμένου. Έτσι δηλαδή μπορεί ο σχεδιαστής να προσθέτει κείμενο ορίζοντας μέσα από την HTML αν το κείμενο αυτό είναι κεφαλίδα, τι γραμματοσειρά θα έχει, τι μέγεθος θα έχει και αν είναι έντονο, πλάγιο ή υπογραμμισμένο. Επίσης μπορεί ο σχεδιαστής να ορίσει λίστες, συνδέσμους, πίνακες, παραθέσεις, παραγράφους, στοίχιση κειμένου και χρώμα κειμένου και πολλές άλλες επιλογές.

Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML. Ο σχεδιαστής έχει τη δυνατότητα για παράδειγμα με τη χρήση JavaScript να εισάγει μέσα στη σελίδα του ένα διαδραστικό ψηφιακό χάρτη. Ένα επίσης σημαντικό χαρακτηριστικό όπου μπορεί να προσθέσει κάποιος σε μία σελίδα, είναι αυτό της υποδιαίρεσης της σελίδας μέσω των tags <div> (από το division – διαίρεση). Με αυτό το τρόπο μπορεί ο σχεδιαστής να χωρίσει τη σελίδα σε διάφορα μέρη όπως, επικεφαλίδα, κυρίως μέρος, υποσέλιδο, περιοχή εικόνων κτλ και να μπορεί κάθε μία περιοχή να την διαχειρίζεται ανεξάρτητα από τις υπόλοιπες. Αν για παράδειγμα μία από τις περιοχές αυτές περιέχει κείμενο, τότε μπορεί κάποιος να ορίσει συγκεκριμένα χαρακτηριστικά κειμένου για την κάθε περιοχή, δηλαδή συγκεκριμένη στοίχιση, γραμματοσειρά και άλλα πολλά.

3.1.2 Τι είναι CSS



Εικόνα 3.2: Λογότυπο CSS

Η CSS (Cascading Style Sheets – Διαδοχικά Φύλλα Στυλ) είναι μία γλώσσα που χρησιμοποιείται αποκλειστικά για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί σε HTML. Η γλώσσα αυτή δημιουργήθηκε για να αναπτύσσει στιλιστικά μία ιστοσελίδα, δηλαδή μπορεί κάποιος να διαμορφώσει τη σελίδα ως προς το χρώμα, το στυλ, τη στοίχιση, παρέχοντάς του περισσότερες επιλογές στο τομέα της διαχείρισης εμφάνισης της ιστοσελίδας, σε σχέση με ένα απλό HTML έγγραφο και της χρήσης των tags.

Ο σχεδιαστής του ιστοτόπου, για να συμπεριλάβει τη μορφοποίηση των σελίδων του μέσω της CSS, θα πρέπει πρώτα να έχει φτιάξει ένα αρχείο στο οποίο θα ορίζει όλα τα στυλ και τις μορφοποιήσεις του ιστοτόπου του με κατάληξη “.css”. Αυτό το αρχείο θα το συμπεριλάβει μέσα σε όλες τις σελίδες όπου θέλει να χρησιμοποιήσει κάποιο ορισμένο από τον ίδιο στυλ μορφοποίησης μέσω του CSS αρχείου, στη περιοχή του tag HEAD. Με αυτό το τρόπο θα ορίσει στο browser, σε ποιο ακριβώς φάκελο θα πρέπει να ανατρέξει για να βρει το αρχείο CSS. Ο web browser έτσι αναφέρεται στο στυλ μορφοποίησης CSS για να ορίσει την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού της κάθε σελίδας. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου.

3.1.3 Τι είναι Βάση δεδομένων

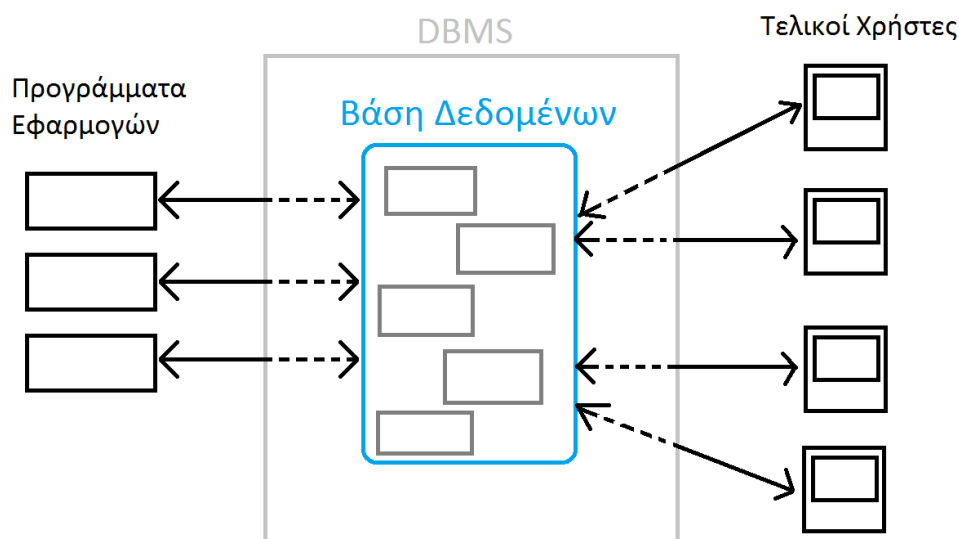
Μία βάση δεδομένων (database), στην επιστήμη των υπολογιστών, είναι μία διακριτή συλλογή σχετιζόμενων δεδομένων τα οποία είναι συστηματικά μορφοποιημένα και οργανωμένα σε πίνακες. Τα δεδομένα ομαδοποιούνται σε πίνακες με τρόπο τέτοιο ώστε να διευκολύνονται οι διαδικασίες ανάκτησης, αναζήτησης, διαγραφής και ενημέρωσής τους από το λογισμικό όπου διαχειρίζεται τη βάση δεδομένων. Η ανάκτηση δεδομένων από τη βάση γίνεται μετά από αναζήτησή τους στους πίνακες της βάσης και η αναζήτηση με τη σειρά της μετά από εντολή προς το λογισμικό που τις διαχειρίζεται. Η εντολή αυτή ονομάζεται ερώτημα προς τη βάση δεδομένων.

Το όνομα του λογισμικού όπου διαχειρίζεται μία βάση δεδομένων είναι: Σύστημα Διαχείρισης Βάσεων Δεδομένων γνωστό και από τα αγγλικά αρχικά του ως DBMS.

Τα DBMSs είναι εφαρμογές λογισμικού οι οποίες είναι σχεδιασμένες να έρχονται σε επαφή με το χρήστη, με άλλες εφαρμογές και με την ίδια τη βάση δεδομένων. Έτσι η κύρια αρμοδιότητα ενός DBMS είναι να βρίσκεται ανάμεσα στο χρήστη ή τις άλλες εφαρμογές και τη βάση δεδομένων, αναλαμβάνοντας τη δημιουργία, ανανέωση, διαγραφή, ενημέρωση, δημιουργία ερωτημάτων, δημιουργία συσχετίσεων και διαχείριση της βάσης δεδομένων. Πολύ γνωστά DBMSs είναι οι: MySQL, PostgreSQL, Microsoft SQL Server, Oracle, SAP, IBM DB2.

Μία βάση δεδομένων δεν έχει τη δυνατότητα της μεταφερσιμότητας ανάμεσα σε διαφορετικά DBMSs, δηλαδή δε μπορεί η ίδια βάση δεδομένων να χρησιμοποιηθεί από διαφορετικά DBMSs. Το αντίθετο όμως είναι δυνατόν να συμβεί, αφού διαφορετικά DBMSs μπορούν, με τη χρήση των προτύπων SQL, ODBC, JDBC, να επιτρέψουν σε μία εφαρμογή να επικοινωνεί με περισσότερα από ένα DBMSs. Τέλος τα DBMSs κατηγοριοποιούνται βάση του μοντέλου βάσης δεδομένων που υποστηρίζουν, με το πιο δημοφιλή από τη δεκαετία του 1980, σχεσιακό μοντέλο που παρουσιάστηκε από τη γλώσσα SQL, να υποστηρίζεται από τα πιο δημοφιλή DBMSs. Με τον όρο σχεσιακό μοντέλο βάσης δεδομένων αναφερόμαστε σε μία βάση δεδομένων οργανωμένη σε συσχετισμένους πίνακες που παρέχει ένα μηχανισμό για ανάγνωση, εγγραφή, τροποποίηση, ή και πιο πολύπλοκες διαδικασίες πάνω στα δεδομένα.

Ένας ηλεκτρονικός τηλεφωνικός κατάλογος, για παράδειγμα, θεωρείται βάση δεδομένων, καθώς μπορεί να αποθηκεύσει και να οργανώσει σχετιζόμενα τμήματα πληροφορίας, όπως είναι το όνομα και ο αριθμός τηλεφώνου. Έτσι ο πίνακας με όνομα “τηλεφωνικός κατάλογος”, θα περιέχει τα πεδία “Όνομα”, “Επώνυμο”, “Διεύθυνση”, “Επάγγελμα” όπου θα είναι δεδομένα τύπου κειμένου, ενώ θα περιέχει και ένα πεδίο με όνομα “Τηλεφωνικός Αριθμός” με τύπο δεδομένων αριθμού.



Εικόνα 3.3: Σχηματική περιγραφή Βάσης Δεδομένων & DBMS

3.1.4 Τι είναι η MySQL



Εικόνα 3.4: Λογότυπο MySQL

Η MySQL είναι ένα σύστημα διαχείρισης βάσεων δεδομένων και αφορά βάσεις δεδομένων που ακολουθούν το σχεσιακό μοντέλο. Είναι η 2η πιο δημοφιλής RDBMS (Relational DBMS) και μετράει 11 εκατομμύρια εγκαταστάσεις. Η MySQL είναι ανοιχτού κώδικα και ο κώδικας ανάπτυξης της είναι διαθέσιμος στο κοινό μέσω της GNU General Public License, καθώς και μέσω ορισμένων ιδιωτικών συμφωνιών. Ανήκει και χρηματοδοτείται από μία και μοναδική κερδοσκοπική εταιρία, τη σουηδική MySQL AB, η οποία σήμερα ανήκει στην Oracle.

Η MySQL τρέχει σε έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων. Το πρόγραμμα αυτό αναπτύχθηκε χωρίς Γραφικό Περιβάλλον Χρήστη (GUI) και μπορεί κάποιος να διαχειριστεί τις βάσεις δεδομένων είτε μέσω εργαλείων γραμμής εντολών (command line), είτε με χρήση κάποιων ειδικών “front-end” εφαρμογών, web εφαρμογών δηλαδή όπου δημιουργήθηκαν για δημιουργία και διαχείριση βάσεων δεδομένων. Τα επίσημα “front-end” εργαλεία που αναπτύχθηκαν από την Oracle είναι δωρεάν διαθέσιμα με όνομα “MySQL Workbench”.

Πολλές διαδικτυακές εφαρμογές χρησιμοποιούν τη MySQL, όπως η Wikipedia και η Google. Τα δεδομένα στη MySQL αποθηκεύονται σε πίνακες. Κάθε πίνακας είναι μια συλλογή από δεδομένα σχετικά μεταξύ τους που περιέχονται στις στήλες και τις γραμμές του. Συνήθως η MySQL χρησιμοποιείται για να δεχτεί ερωτήματα (queries) βάσεων δεδομένων και να επιστρέψει με ταχύτητα τα αποτελέσματα. Σημαντικό γεγονός για το οποίο η MySQL χρησιμοποιείται ευρέως είναι η άριστη συνεργασία με την PHP γλώσσα scripting. Έτσι σε μία HTML ιστοσελίδα, μπορεί να χρησιμοποιηθεί η PHP για να γίνει επικοινωνία, για παράδειγμα, μίας φόρμας της HTML με τη MySQL βάση δεδομένων.

3.1.5 Τι είναι η PHP



Εικόνα 3.5: Λογότυπο PHP

Η PHP είναι μια γλώσσα για δημιουργία scripts που χρησιμοποιείται για την ανάπτυξη δυναμικών ιστοσελίδων στο web. Μέσα σε μία HTML σελίδα δίνετε η δυνατότητα στο προγραμματιστή να εισάγει ένα script σε PHP και να εκτελείται κάθε φορά που κάποιος επισκέπτεται τη σελίδα αυτή. Αυτή η γλώσσα συγγραφής scripts εκτελείται από το web server (server-side scripting), έτσι ο κώδικας του script μεταφράζεται από το server και είτε δημιουργείται κώδικας σε HTML ή κάποια άλλη έξοδος προς το χρήστη.

Μία βασική ιδιότητα της PHP είναι αυτή της δυνατότητας ανάπτυξης scripts τα οποία επιτρέπουν την επικοινωνία της HTML σελίδας με μία βάση δεδομένων. Αυτό σημαίνει ότι μέσω της PHP από μία HTML σελίδα μπορεί κάποιος να εκτελεί ερωτήματα προς τη βάση δεδομένων και πάλι μέσω της PHP να γίνεται η παρουσίαση των αποτελεσμάτων του ερωτήματος από την HTML σελίδα προς το χρήστη. Με αυτό το τρόπο έχει επιτευχθεί η δημιουργία μίας δυναμικής ιστοσελίδας που θα επικοινωνεί με μία βάση δεδομένων. Επιπλέον επιτρέπει την ανάπτυξη scripts με σκοπό τη διαχείριση του περιεχομένου της βάσης δεδομένων αποκλειστικά μέσω της PHP. Όταν χρειαστεί δηλαδή να γίνει οποιαδήποτε μεταβολή στα περιεχόμενα της βάσης δεδομένων, δημιουργία εγγραφής, διαγραφή εγγραφής, δημιουργία πίνακα κτλ αυτό να γίνεται μέσω scripts σε PHP.

Η PHP διαθέτει εντολές για δημιουργία συνθηκών (if, then, else) και για τη δημιουργία ελεγχόμενων βρόγχων (for, while κτλ), ενώ χειρίζεται μεταβλητές και επιτρέπει την επικοινωνία με φόρμες συμπλήρωσης δεδομένων που έχουν δημιουργηθεί σε HTML με σκοπό τη διαχείριση των δεδομένων που εισάγονται και εξάγονται από αυτές. Οι επεκτάσεις των αρχείων PHP είναι .php, .php4, .phtml.

Όπως αναφέρθηκε και παραπάνω επειδή η PHP είναι μία server-side γλώσσα scripting χρειάζεται πρώτα να μεταγλωττιστεί από το web server και τα αποτελέσματα της μεταγλώττισης επιστρέφονται στη σελίδα από όπου στάλθηκαν. Σε αντίθεση με μία σελίδα HTML όπου ο κώδικας εκτελείται άμεσα από το web browser, στη περίπτωση της PHP προστίθενται ένα βήμα παραπάνω, αυτό της μεταγλώττισης, οπότε αυξάνεται ελάχιστα ο χρόνος εκτέλεσης του script.

Έτσι η ενσωμάτωση κώδικα PHP σε ένα αρχείο με κατάληξη .html δε θα λειτουργήσει και θα εμφανιστεί στο browser ο κώδικας χωρίς καμία επεξεργασία. Για να εμφανιστεί σωστά η σελίδα θα πρέπει να έχει ρυθμιστεί κατάλληλα ο server. Επίσης ακόμα και όταν ένα αρχείο έχει επέκταση .php θα πρέπει ο server να έχει ρυθμιστεί ώστε να μεταγλωττίζει το κώδικα PHP σε HTML όπου καταλαβαίνει το πρόγραμμα του πελάτη. Οι περισσότεροι από τους web servers υποστηρίζουν την εκτέλεση των PHP scripts, ενώ ο περισσότερο χρησιμοποιούμενος web server είναι ο Apache καθώς μπορεί να εκτελεστεί στα πιο διαδεδομένα λειτουργικά συστήματα Windows, Linux και Mac OS.

Σήμερα ο μεγαλύτερος αριθμός ιστοσελίδων δυναμικού περιεχομένου έχουν αναπτυχθεί χρησιμοποιώντας τη γλώσσα scripting PHP, ενώ οι κύριοι ανταγωνιστές της είναι η Microsoft Active Server Pages (ASP .NET), η Java Server Pages (JSP) και η Perl, με τη PHP να υπερτερεί στα εξής σημεία: η PHP είναι ανοιχτού κώδικα οπότε υπάρχει διαθέσιμος ο κώδικας προέλευσης, χαμηλό κόστος, ευκολία χρήσης και εκμάθησης από μη έμπειρους προγραμματιστές, δυνατότητα επικοινωνίας με διάφορα συστήματα βάσεων δεδομένων, υψηλή ταχύτητα εκτέλεσης, ενσωματωμένες βιβλιοθήκες, υψηλή απόδοση, μεταφερσιμότητα.

3.1.6 Apache HTTP Server



Εικόνα 3.6: Λογότυπο Apache HTTP Server

Ο Apache HTTP Server είναι όπως λέει και ο τίτλος του ένας εξυπηρετητής του παγκόσμιου ιστού. Γενικά εξυπηρετητής παγκόσμιου ιστού (web server) είναι ένας διακομιστής μέσα στον οποίο αποθηκεύονται ιστοσελίδες του παγκόσμιου ιστού μαζί με όλα τα απαραίτητα δομικά τους στοιχεία, εικόνες, βίντεο, ήχους κτλ. Όταν ένας χρήστης επισκέπτεται ένα ιστότοπο μέσω του browser που χρησιμοποιεί (πρόγραμμα πλοήγησης web), ο δεύτερος αναλαμβάνει να επικοινωνήσει μέσω του πρωτοκόλλου HTTP με το web server στον οποίο είναι αποθηκευμένος αυτός ο ιστότοπος. Μόλις επικοινωνήσει ο browser με το web server αυτό, ο server του απαντάει στέλλοντας του την αρχική σελίδα αυτού του ιστότοπου (αν αυτή ζήτησε ο χρήστης), έτσι ο browser αναλαμβάνει να τη μεταφράσει και να τη παρουσιάσει στο χρήστη.

Ένας από τους λόγους που ο Apache είναι ένας από τους δημοφιλέστερους web servers, είναι γιατί λειτουργεί σε διάφορες πλατφόρμες όπως Windows, Linux, και Mac OS. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation). Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. Oracle, MySQL. Ενώ μπορεί να εκτελείται και μέσω των πιο γνωστών εργαλείων εξομοίωσης web server για ανάπτυξη και εκτέλεση ιστοσελίδων σε τοπικό επίπεδο π.χ. xampp, wampp και άλλων.

Η πρώτη του έκδοση, κυκλοφόρησε το 1993 και έπαιξε σημαντικό ρόλο στην επέκταση του παγκόσμιου ιστού. Ήταν η πρώτη εναλλακτική επιλογή που παρουσιάστηκε μετά τη κυκλοφορία του εξυπηρετητή HTTP της εταιρείας Netscape και από τότε έχει εξελιχθεί στο σημείο να ανταγωνίζεται άλλους εξυπηρετητές βασισμένους στο Unix. Ο Apache HTTP Server συμπεριλαμβανόταν από το 1996 στη λίστα των πιο δημοφιλών HTTP εξυπηρετητών, όμως από τον Μάρτιο του 2006 έχει μειωθεί το ποσοστό της εγκατάστασής του κυρίως λόγω της ύπαρξης του Microsoft Internet Information Services και την πλατφόρμα .NET . Τον Οκτώβριο του 2007 το μερίδιο του ήταν 47.73% από όλους τους ιστοτόπους.

3.1.7 Τι είναι XAMPP



Εικόνα 3.7: Λογότυπο Xampp

Το XAMPP είναι ένα πακέτο ελεύθερων εφαρμογών καθώς και λογισμικού στο οποίο μπορεί κάποιος να επέμβει στο κώδικά του (ανοικτού κώδικα). Επίσης, το λογισμικό που περιέχει μπορεί να εγκατασταθεί και να λειτουργήσει σε όλα τα γνωστά λειτουργικά συστήματα. Οι εφαρμογές αυτές είναι ο εξυπηρετητής ιστοσελίδων HTTP Apache, η βάση δεδομένων MySQL, η πλατφόρμα ηλεκτρονικού ταχυδρομείου Mercury και ένας διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl. Περιλαμβάνει ένα πρόγραμμα FTP client το Filezilla και ένα πρόγραμμα που υποστηρίζει τα Java Servlets με όνομα Tomcat. Επίσημα οι σχεδιαστές του XAMPP προόριζαν το λογισμικό ως εργαλείο ανάπτυξης και δοκιμής ιστοσελίδων τοπικά στον υπολογιστή χωρίς να είναι απαραίτητη η σύνδεση στο διαδίκτυο. Αποτελεί δηλαδή λογισμικό προσομοίωσης web server για τις ιστοσελίδες που θέλει κάποιος να φιλοξενήσει και να δοκιμάσει πριν τις καταχωρήσει σε κάποιο server του διαδικτύου ώστε να είναι δημόσια προσβάσιμες. Η άδεια χρήσης κάτω από την οποία δημιουργήθηκε το Xampp, είναι η GNU (General Public License) και συμπεριφέρεται σαν ένας κανονικός web server, χωρίς κόστος. Τέλος το Xampp αποτελεί ένα πακέτο εργαλείων το οποίο είναι διαθέσιμο για εγκατάσταση σε διάφορα λειτουργικά συστήματα όπως Windows, Linux, Solaris, και Mac OS.

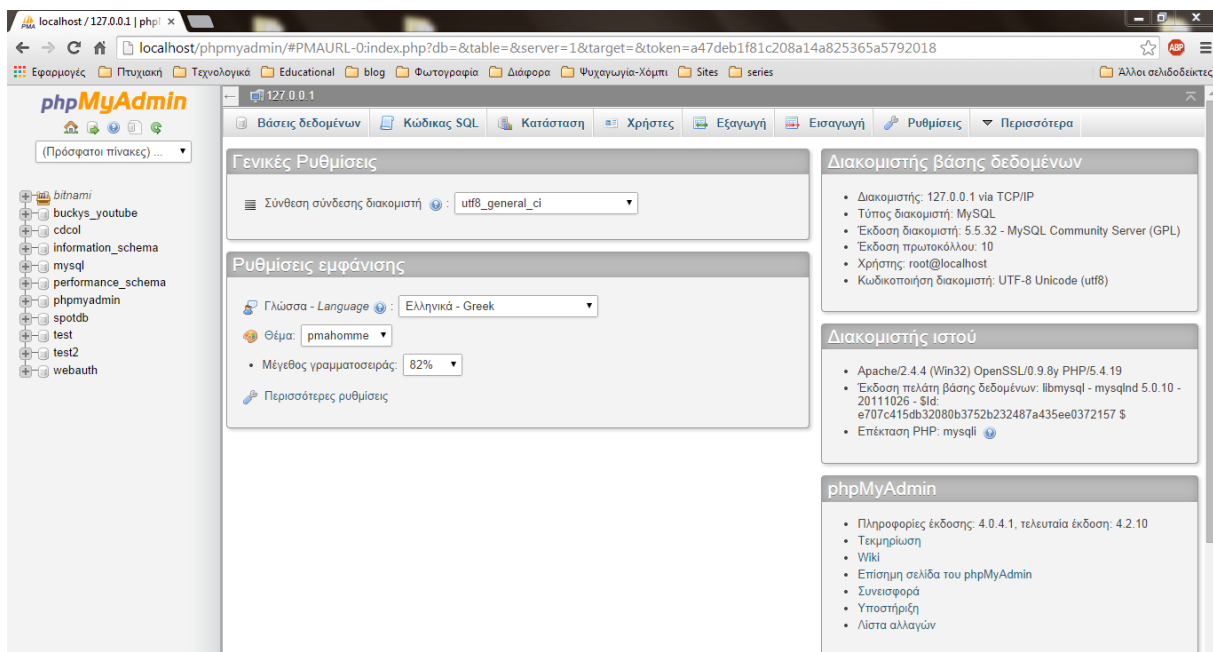
3.1.8 PhpMyAdmin



Εικόνα 3.8: Λογότυπο phpMyAdmin

Το PhpMyAdmin είναι ένα εργαλείο γραμμένο σε PHP το οποίο επιτρέπει τη διαχείριση των βάσεων δεδομένων που διαθέτει κάποιος μέσω web, απλά μέσα από ένα web browser. Το phpMyAdmin μπορεί να χειρίζεται πλήρως βάσεις δεδομένων, πίνακες, πεδία πινάκων αλλά και ένα ολόκληρο MySQL Server. Υποστηρίζει 54 γλώσσες, μεταξύ των οποίων και τα ελληνικά και είναι λογισμικό ανοιχτού κώδικα.

Όπως φαίνεται και στην εικόνα παρακάτω, η κεντρική σελίδα του phpMyAdmin διαθέτει τρεις στήλες. Η αριστερή περιέχει τις βάσεις δεδομένων όπου μπορεί να διαχειριστεί ο χρήστης, στη κεντρική στήλη υπάρχουν οι επιλογές που έχει σχετικά με τις βάσεις δεδομένων. Για παράδειγμα, μπορεί κάποιος μέσα από το κεντρικό μενού να εισάγει ή να εξάγει μία βάση δεδομένων, να δημιουργήσει κώδικα σε SQL, να κάνει ρυθμίσεις σχετικά με τους χρήστες που θα έχουν πρόσβαση στις βάσεις και άλλα πολλά. Στη δεξιά στήλη εμφανίζονται πληροφορίες σχετικά με το Server της βάσης δεδομένων, σχετικά με το Web Server και ότι έχει σχέση με την έκδοση του phpMyAdmin και τη τεκμηρίωσή του.



Εικόνα 3.9: Κεντρική σελίδα phpMyAdmin

3.1.9 Τι είναι το Dreamweaver



Εικόνα 3.10: Λογότυπο Dreamweaver CC

Το dreamweaver είναι λογισμικό δημιουργίας και επεξεργασίας ιστοσελίδων και αναπτύχθηκε από την Adobe. Μέσα από το πρόγραμμα αυτό μπορούμε να δημιουργήσουμε φόρμες (forms), πλαίσια (frames), πίνακες (tables) και άλλα αντικείμενα της HTML. Εκτός από HTML μπορεί κάποιος να κάνει ανάπτυξη κώδικα σε: CSS, LESS, Sass, Scss, JavaScript, PHP, XML, SVG όπως και να δημιουργήσει δικά του πρότυπα (templates) σε HTML και PHP. Το Dreamweaver παρέχει δυνατότητες που αποσκοπούν στη δημιουργία δυναμικής HTML (DHMTL) και επιτρέπει κίνηση γραμμής χρόνου, τη τοποθέτηση αντικειμένων με μία μόνο κίνηση, δημιουργία επιπέδων (layers) και συγγραφή scripts. Το Dreamweaver περιέχει δικές του συμπεριφορές (behaviors), που είναι έτοιμα scripts τα οποία μπορούμε να προσθέσουμε πολύ εύκολα σ' ένα αντικείμενο, καθώς και παρέχει τη δυνατότητα αυτοματοποιημένων εντολών εισαγωγής αντικειμένων.

3.1.10 Javascript



Εικόνα 3.11: Λογότυπο JavaScript

Η Javascript είναι μία γλώσσα ανάπτυξης scripts που προορίζονται για την ενσωμάτωση δυναμικού περιεχομένου σε ιστοσελίδες. Η σύνταξη της είναι επηρεασμένη από τη C και τα scripts της εκτελούνται στη πλευρά του πελάτη (client-side). Η Javascript υποστηρίζει πολλά προγραμματιστικά παραδείγματα (multi-paradigm) όπως είναι το αντικειμενοστραφές, το προστακτικό και το συναρτησιακό στυλ προγραμματισμού. Εκτός από την ανάπτυξη δυναμικού περιεχομένου σε ιστοσελίδες, η Javascript έχει εφαρμογή και σε τομείς εκτός ιστοσελίδων, όπως είναι τα γνωστά σε όλους έγγραφα PDF, τα widgets (μικρές εφαρμογές) στην επιφάνεια εργασίας και άλλες. Μερικά πολύ γνωστά παραδείγματα χρήσης της Javascript σε server-side εφαρμογές είναι αυτό της ανάπτυξης εικονικών μηχανών, αλλιώς virtual machines, όπως και η χρήση της στη πλατφόρμα ανάπτυξης κλιμακωτών διαδικτυακών εφαρμογών το Node.js.

Η ονομασία αυτής της γλώσσας περιέχει το όνομα της αντικειμενοστραφούς γλώσσας προγραμματισμού Java, από την οποία αντιγράφει ονόματα και συμβάσεις ονοματοδοσίας. Κατά τα άλλα όμως αυτές οι δύο γλώσσες δεν μοιάζουν και δε πρέπει να συγχέονται αφού διαθέτουν πολύ διαφορετική σύνταξη και η λογική για την οποία σχεδιάστηκαν να εξυπηρετούν είναι πολύ διαφορετική για τη κάθε γλώσσα, όπως και οι εφαρμογές τους. Αντίθετα ο σχεδιασμός της Javascript βασίζεται στις γλώσσες προγραμματισμού Self και Scheme.

Τέλος ένα σημαντικό χαρακτηριστικό της Javascript είναι η άψογη συνεργασία με την HTML5. Αυτό σημαίνει ότι κάποιος μπορεί να χρησιμοποιήσει την HTML5 για την ανάπτυξη μίας ιστοσελίδας και να ενσωματώσει με χρήση της Javascript δυναμικό περιεχόμενο σε αυτή. Ένα τέτοιο παράδειγμα είναι η δημιουργία μίας HTML σελίδας όπου θα περιέχει ένα ψηφιακό χάρτη της Google (Google Map).

3.2 Ανάπτυξη εφαρμογής Android

3.2.1 Google



Εικόνα 3.12: Λογότυπο Google

Η παγκοσμίως γνωστή εταιρεία που ονομάζεται Google ξεκίνησε την λειτουργία της στις 27 Σεπτεμβρίου του 1998. Ο στόχος της εταιρείας είναι να οργανώσει όλες τις πληροφορίες του κόσμου καθιστώντας τις παγκοσμίως διαθέσιμες. Η Google ξεκίνησε σαν ένα project κολεγιακού επιπέδου από τον Λάρι Πέιτζ (Larry Page) και τον Σεργκέι Μπριν (Sergey Brin) το 1996 για μια μηχανή αναζήτησης, και σήμερα αποτελεί μια από τις μεγαλύτερες εταιρείες διαδικτυακών υπηρεσιών και έναν κολοσό στον χώρο της ψηφιακής αγοράς εφαρμογών.

Επιπλέον η Google, έχει επιδείξει ένα ενδιαφέρον τόσο στην εξάπλωση και διάδοση τόσο της κεντρικής ιδέας της, του εμπορικού ονόματός (brand name) της, της πλατφόρμας αναζήτησης όσο και στην διαφήμιση του πακέτου εργαλείων της στην αγορά ασύρματης επικοινωνίας και συνεπώς της κινητής τηλεφωνίας. Το επιχειρηματικό μοντέλο της εταιρείας έχει τρομακτική επιτυχία στο Internet και συνεχίζει με αύξουσα πορεία. Ένας από τους στόχους της Google ήταν να μεταφέρει τις υπηρεσίες που απολάμβαναν οι χρήστες του internet στους συνδρομητές κινητής τηλεφωνίας. Οι αρχικές απόπειρες της εταιρείας στην αγορά των κινητών τηλεφώνων στιγματίστηκαν από πολλά προβλήματα. Με τη φιλοσοφία της να επικεντρώνεται στο χρήστη και τη σχεδίαση της, η Google ηγείται ενός κινήματος που επιχειρεί να μετατρέψει την υπάρχουσα και ιδιαίτερα κλειστή και φυλασσόμενη ασύρματη αγορά σε μία αγορά όπου οι χρήστες τηλεφώνων μπορούν να μετακινούνται από τον ένα φορέα στον άλλο και να έχουν ελεύθερη πρόσβαση τόσο σε εφαρμογές όσο και σε υπηρεσίες. Επίσης, η Google ακολουθεί μία ευρεία προσέγγιση εξετάζοντας την ασύρματη υποδομή απ' την οπτική της Αμερικανικής επιτροπής επικοινωνιών (FCC) για τις απαιτήσεις των κατασκευαστών τηλεφωνικών συσκευών, τις ανάγκες των προγραμματιστών εφαρμογών (developers) και τις επιθυμίες των πάροχων κινητής τηλεφωνίας. Στη συνέχεια, η Google στάθηκε στο πλευρό άλλων ομοϊδεατών μελών της ασύρματης κοινότητας θέτοντας το εξής ερώτημα: Τι θα έπρεπε να πραγματοποιηθεί, ώστε να κατασκευάσουμε ένα καλύτερο κινητό τηλέφωνο.

Όσον αφορά τις εφαρμογές της Google, εκείνες αναπτύσσονται κατά κύριο λόγο σε γλώσσα προγραμματισμού JAVA χρησιμοποιώντας το λογισμικό Android Development Kit (ADT). Παρόλα αυτά υπάρχουν κι άλλα εργαλεία που είναι διαθέσιμα για τη δημιουργία και την ανάπτυξη των εφαρμογών όπως το ενσωματωμένο Software Development Kit (SDK) για εφαρμογές ή επεκτάσεις τους σε γλώσσα C ή C++. Τέλος το Google App Inventor, είναι ένα γραφικό περιβάλλον που παρέχει τη δυνατότητα δημιουργίας προγράμματος για το Android ακόμα και σε αρχάριους προγραμματιστές.

3.2.2 Google Play



Εικόνα 3.13: Λογότυπο Google Play

Το Google Play, όπου παλιότερα ονομαζόταν Android Market, είναι μία πλατφόρμα διαμοιρασμού ψηφιακού περιεχομένου διαχειριζόμενη από τη Google. Για τις συσκευές με λειτουργικό σύστημα Android, αυτή η εφαρμογή, αποτελεί το επίσημο κατάστημα λογισμικού και επιτρέπει στους χρήστες να αναζητήσουν και να κατεβάσουν στη συσκευή τους εφαρμογές οι οποίες έχουν αναπτυχθεί με το Android SDK και έχουν δημοσιευθεί και διανεμηθεί μέσω του Google.

Το Android Market ανακοινώθηκε από τη Google στις 28 Αυγούστου 2008 και έγινε διαθέσιμο για τους χρήστες 22 Οκτωβρίου. Η υποστήριξη για εφαρμογές επί πληρωμή ξεκίνησε στις 13 Φεβρουαρίου 2009 για προγραμματιστές των Ηνωμένων Πολιτειών και Ηνωμένου Βασιλείου, ενώ στις 30 Οκτωβρίου 2010 προστέθηκαν άλλες 29 χώρες. Το Δεκέμβριο του 2010 προστέθηκε ένα φίλτρο περιεχομένου στο Android Market, ενώ μειώθηκε δραματικά ο χρόνος που χρειαζόταν από τη στιγμή της πληρωμής του χρήστη για μία εφαρμογή έως τη διάθεσή της σε εκείνον, από 24-48 ώρες σε μόλις 15 λεπτά.

Μέσα στο 2011 έγιναν πολλές και σημαντικές αλλαγές για το τότε Android Market, όπως αυτή της δυνατότητας να κατεβάζει κάποιος χρήστης εφαρμογές μπαίνοντας μέσω του υπολογιστή του στο Web Site της πλατφόρμας λογισμικού και επιλέγοντας τη συσκευή Android που επιθυμεί να την εγκαταστήσει μόλις αυτή συνδεθεί στο Internet. Επίσης σημαντική είναι η πρόσθεση λιστών μέσα στο Android Market με τις κατηγορίες εφαρμογών “Κορυφαίες Πωλήσεις”, “Κορυφαίοι Προγραμματιστές”, “Τάσεις” και “Προτάσεις Συντακτών” έτσι ώστε μπαίνοντας στο Android Market να μπορεί ο χρήστης να βλέπει ποιες εφαρμογές και ποιοι προγραμματιστές είναι πιο ψηλά στις προτιμήσεις των υπόλοιπων χρηστών αλλά και των συντακτών ενώ παράλληλα προστέθηκαν περισσότερα φίλτρα αναζήτησης.

Τέλος στις 6 Μαρτίου 2012 έγινε η μετονομασία του Android Market σε Google Play Store ενώ άλλαξε και ο σχεδιασμός του για να γίνει σχεδόν όπως είναι σήμερα.

Τα εκτελέσιμα αρχεία των εφαρμογών στο Google Play Store έχουν κατάληξη .apk. Ενώ όμως στην αρχή περιορίζονταν το μέγεθος των αρχείων .apk που ανέβαζαν οι προγραμματιστές στο ηλεκτρονικό κατάστημα σε 50 Mbytes, από εκείνη τη χρονιά οι προγραμματιστές έχουν τη δυνατότητα να προσθέσουν δύο αρχεία των 100Mbytes σε ότι αφορά τα .apk ενώ μπορούν να προσθέσουν και άλλα 2 αρχεία των 2Gbytes το καθένα, άρα συνολικά 4,146Gbytes

Εκτός από το διαμοιρασμό εφαρμογών, το Google Play πλέον προτείνεται και ως ψηφιακό κατάστημα πολυμέσων προσφέροντας μουσική, περιοδικά, βιβλία, ταινίες και τηλεοπτικά προγράμματα. Οι χρήστες επίσης μπορούν να αγοράσουν και συσκευές μέσω αυτής της πλατφόρμας όπως Chromebooks, Google Nexus κινητές συσκευές, Chromecasts και αξεσουάρς.

Τα ονόματα των υπηρεσιών που λειτουργούν στη πλατφόρμα Google Play είναι Google Play Music, Google Play Books, Google Play Newsstand, Google Play Movies & TV and Google Play Games.

Οι εφαρμογές που διατίθενται από το Google Play χωρίζονται σε δύο βασικές κατηγορίες, αυτές που μπορεί να κατεβάσει ο χρήστης στη συσκευή του δωρεάν και αυτές που θα πρέπει να καταβάλει κάποιο ποσό. Για να κατεβάσει κάποιος τις εφαρμογές που τον ενδιαφέρουν μπορεί είτε να επισκεφτεί την εφαρμογή του Google Play για συσκευές Android ή για Google TV ή μπορεί να επισκεφτεί και να κατεβάσει τις εφαρμογές που τον ενδιαφέρουν από το διαδικτυακό τόπο του Google

Play. Πολλές από τις εφαρμογές μπορεί να στοχεύουν σε συγκεκριμένους χρήστες με βάση τα ιδιαίτερα χαρακτηριστικά των συσκευών τους, για παράδειγμα μερικές εφαρμογές χρησιμοποιούν τον αισθητήρα κίνησης για να κινηθεί κάποιο αντικείμενο μέσα στην εφαρμογή ενώ άλλες εφαρμογές απαιτούν τη χρήση του δέκτη GPS της συσκευής όπως συμβαίνει στη περίπτωση μας, για εντοπισμό θέσης, υπολογισμό ύψους και ταχύτητας κίνησης.

Το κατάστημα Google Play έχει φτάσει τα 1,3 εκατομμύρια δημοσιευμένες εφαρμογές και 50 δισεκατομμύρια κατεβάσματα. Οι δωρεάν εφαρμογές είναι διαθέσιμες παγκοσμίως εκτός από χώρες που έχουν τεθεί σε εμπάργκο από τις Η.Π.Α., ενώ οι επί πληρωμή εφαρμογές είναι διαθέσιμες σε 135 χώρες. Σύμφωνα με την “AppBrain Stats” από τις 1,3 εκατομμύρια συνολικά διαθέσιμες εφαρμογές, το 1 εκατομμύριο διατίθενται δωρεάν ενώ μόνο 200 χιλιάδες είναι επί πληρωμή.

Σύμφωνα με μετρήσεις, προγραμματιστές από 45 χώρες έχουν τη δυνατότητα να διαμοιραστούν τις εφαρμογές τους στο Google Play επί πληρωμή. Για να μπορέσει όμως κάποιος προγραμματιστής να καταστήσει την εφαρμογή του διαθέσιμη στο ηλεκτρονικό κατάστημα της Google, θα πρέπει πρώτα να καταβάλει το ποσό των 25\$. Στη περίπτωση όπου μία εφαρμογή διατίθεται επί πληρωμή, το 70% του χρηματικού ποσού πηγαίνει στο προγραμματιστή, ενώ το υπόλοιπο 30% πηγαίνει στα έξοδα διανομής και στα λειτουργικά τέλη. Το κέρδος που μπορεί κάποιος να έχει από μία εφαρμογή του στο Google Play τα λαμβάνει μέσω της υπηρεσίας Google Wallet ή μέσω του AdSense σε μερικές χώρες.

Το Google Play Store περιέχει φίλτρα που παρουσιάζουν μόνο τις εφαρμογές που είναι συμβατές με τη συσκευή του χρήστη έτσι ώστε να αποφευχθούν προβλήματα συμβατότητας. Επίσης οι προγραμματιστές των εφαρμογών έχουν τη δυνατότητα να περιορίζουν τις εφαρμογές που έχουν δημιουργήσει σε συγκεκριμένες χώρες. Σαν εφαρμογή όμως δεν είναι ανοιχτού κώδικα και μπορεί να εγκατασταθεί μόνο σε συσκευές όπου είναι συμβατές με τις απαιτήσεις της Google αφού αποδεχτεί τη δωρεάν χορήγηση άδειας εγκατάστασης και χρήσης από τη Google.

Ως εκ τούτου οι χρήστες κάποιων κινητών συσκευών τύπου tablet όπως για παράδειγμα το Kindle Fire της Amazon δε μπορούν να εγκαταστήσουν το Google Play Store αλλά χρησιμοποιούν το ήδη εγκατεστημένο λογισμικό ψηφιακού καταστήματος που τους χορηγεί η εταιρεία κατασκευής του tablet.

3.2.3 Google Play Developers



Εικόνα 3.14: Λογότυπο Google Developers

Πλέον στο Google Play είναι αρκετά εύκολο, οποιοσδήποτε προγραμματιστής Android εφαρμογών να ανεβάσει και να προωθήσει το application του σε αυτό. Η διαδικασία για να ανεβάσει ένας προγραμματιστής την εφαρμογή του είναι σχετικά εύκολη και χρειάζεται ένα ενδεικτικό ποσό καταβολής για μία και μόνο φορά. Το Google Play δίνει διάφορες δυνατότητες τόσο στους προγραμματιστές των εφαρμογών όσο και στους χρήστες παρέχοντας ένα σύστημα αξιολόγησης της εφαρμογής, εικόνες της εφαρμογής και άλλα στοιχεία όπως την έκδοση αυτής αλλά και το μέγεθός της. Επίσης, υπάρχουν πολλά στατιστικά όπως π.χ πόσοι χρήστες κατέβασαν την εφαρμογή σε συγκεκριμένο χρονικό διάστημα. Πιο συγκεκριμένα για να δημοσιεύσουμε την εφαρμογή μας στο Google Play θα πρέπει το application μας να τηρεί κάποιους βασικούς κανόνες που έχει θέσει η Google όπως για παράδειγμα.

- Η εφαρμογή μας δηλαδή το .apk αρχείο να μην υπερβαίνει τα 100 Mbytes.
- Πρέπει να διαγράψουμε τυχόν Log κλήσεις από τον κώδικα μας και να μην υπάρχει το χαρακτηριστικό android: debuggable στο manifest αρχείο.
- Να κάνουμε έναν ενδελεχή έλεγχο της εφαρμογής σε τουλάχιστον μία κινητή συσκευή πριν την ανεβάσουμε στο Google Play.
- Να βεβαιωθούμε ότι δουλεύουν τα path μας και οι κλήσεις που κάνει η εφαρμογή σε κάποιον απομακρυσμένο server.
- Να παρέχουμε τιμές για τις μεταβλητές android:versionCode και android:versionName στο manifest αρχείο μας.
- Να βεβαιωθούμε ότι χρησιμοποιούμε τα σωστά API keys στον κώδικα μας (κάθε API key πρέπει να αντιστοιχεί στο package name της εφαρμογής μας).

Εάν η εφαρμογή μας δεν τηρεί σωστά τον τελευταίο κανονισμό σχετικά με το API Key, τότε δεν έχουμε πρόσβαση στην υπηρεσία δικαιωμάτων της Google (Google Play Services) και επομένως δεν λειτουργεί σωστά.

3.2.4 Android



Εικόνα 3.15: Λογότυπο Android

Το Android είναι ένα λειτουργικό σύστημα για κινητές συσκευές το οποίο τρέχει σε πυρήνα βασισμένο στο Linux και το λογότυπο του είναι ένα ρομπότ σε πράσινο χρώμα το οποίο σχεδιάστηκε από τη γραφίστρια Irina Blok. Αρχικά αναπτύχθηκε από την Google και στη συνέχεια από την Open Handset Alliance (OHA) η οποία είναι μια κοινοπραξία 48 τηλεπικοινωνιακών εταιριών αλλά και λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφοσιωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων πάνω στις συσκευές κινητής τηλεφωνίας. Ακόμη επιτρέπουν στους προγραμματιστές να συνθέτουν κώδικα με την χρήση της ευρέως διαδεδομένης και χρήσιμης γλώσσας προγραμματισμού Java και ελέγχοντας παράλληλα τη συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων αποκλειστικά από τη Google.

Το λειτουργικό σύστημα Android μπορεί να συνεργαστεί με κάμερες στατικής ή κινούμενης εικόνας, οθόνες αφής, GPS και αισθητήρες, κάτι το οποίο μας δίνει τεράστιες δυνατότητες όσον αφορά την λειτουργικότητά του. Επίσης αποτελεί ένα σύνολο από εφαρμογές που συμπεριλαμβάνονται στις συσκευές τύπου Android για καθημερινές διεργασίες, όπως περιήγηση στο Web αλλά και χρήση του email. Αφού απευθύνεται σε ένα ανθεκτικό και ανοιχτό περιβάλλον ανάπτυξης για ασύρματες εφαρμογές, το Android είναι μία ταχύτατα αναπτυσσόμενη πλατφόρμα για την ανάπτυξη εφαρμογών σε κινητές συσκευές. Πρέπει να τονιστεί πως η πλατφόρμα σχεδιάστηκε για την ενθάρρυνση μιας ελεύθερης και ανοιχτής αγοράς, από την οποία θα μπορούν να απολαμβάνουν όλοι οι χρήστες εφαρμογές κινητών και που θα μπορούν να αναπτύσσουν όλοι οι προγραμματιστές.

3.2.5 Ανάπτυξη Εφαρμογών σε Android Περιβάλλον

Το Android προσφέρει στους προγραμματιστές την δυνατότητα να κατασκευάσουν πλούσιες χρήσιμες και καινοτόμες εφαρμογές, λόγω της ανοικτής πλατφόρμας ανάπτυξης που παρέχει. Οι προγραμματιστές αφήνονται ελεύθεροι να εκμεταλλευτούν πλήρως το υλικό μέρος της κινητής συσκευής, για παράδειγμα μπορούν να σχεδιάσουν εφαρμογές που θα εκμεταλλεύονται τη πρόσβαση στις υπηρεσίες εντοπισμού θέσης, να εκτελούν διεργασίες αδιαφανείς προς το χρήστη και πολλά άλλα. Επίσης ο προγραμματιστής έχει τη δυνατότητα να χρησιμοποιήσει APIs που υπάρχουν ήδη για την ανάπτυξη άλλων εφαρμογών. Το Android δίνει τη δυνατότητα σε μία εφαρμογή να χρησιμοποιήσει τις δυνατότητες μιας άλλης και επίσης δίνει την δυνατότητα στο χρήστη να αλλάξει τα συστατικά κάθε εφαρμογής. Επίσης το περιβάλλον ανάπτυξης λογισμικού περιλαμβάνει ένα προσομοιωτή συσκευής (emulator), εργαλεία για διόρθωση σφαλμάτων (debugging), μνήμη και εργαλεία ανάλυσης της απόδοσης του εκτελέσιμου λογισμικού καθώς και ένα επιπρόσθετο για το Eclipse IDE.

Η ανάπτυξη εφαρμογών Android γίνεται δωρεάν. Δεν απαιτείται η καταβολή χρηματικών ποσών για άδειες χρήσης και πνευματικά δικαιώματα ώστε να προγραμματίσει κάποιος σ' αυτήν την πλατφόρμα. Ακόμη, είναι δωρεάν για όποιον θέλει να γίνει μέλος της κοινότητας προγραμματιστών. Όπως επίσης δεν καταβάλλεται κάποιο ποσό για τις δοκιμές. Τέλος κανένα ποσό δεν δίνεται για υπογραφές ή πιστοποιήσεις.

Η πλατφόρμα Android παρέχεται μέσω της διαδικασίας ανοικτής πηγής (open source). Οι προγραμματιστές απολαμβάνουν ανεμπόδιστη πρόσβαση στα χαρακτηριστικά των συσκευών όταν αναπτύσσουν εφαρμογές. Οι σχεδιαστές ακολούθησαν μία περιεκτική προσέγγιση στη σχεδίαση της πλατφόρμας Android. Ξεκίνησαν μ' ένα ασφαλές λειτουργικό σύστημα και κατασκεύασαν ένα στιβαρό πλαίσιο λογισμικού το οποίο επιτρέπει την πλούσια ανάπτυξη εφαρμογών, όπως επίσης και την ακόμη ευκολότερη διάδοση αυτών.

Κάθε νέα έκδοση του Android SDK παρέχει πολλές ουσιαστικές βελτιώσεις στην πλατφόρμα. Σε πρόσφατες αναθεωρήσεις, η πλατφόρμα Android έχει δεχθεί μία περαιτέρω διόρθωση τόσο στην οπτική πλευρά του, όσο και στις επιδόσεις του. Αν και οι περισσότερες αναβαθμίσεις και βελτιώσεις είναι καλοδεχούμενες και απαραίτητες, οι νέες εκδόσεις SDK συχνά προκαλούν κάποια αναστάτωση στην κοινότητα προγραμματιστών του Android. Πολλές δημοσιευμένες εφαρμογές απαιτούν νέες δοκιμές και νέες υποβολές στην αγορά του Android ώστε να συμμορφώνονται με τις προδιαγραφές των νέων SDK, οι οποίες γρήγορα επεκτείνονται στην πράξη σ' όλα τα τηλέφωνα Android με τη μορφή αναβάθμισης του λογισμικού υλικού (firmware) και παραγκωνίζουν τις παλαιότερες εφαρμογές. Ορισμένες παλαιότερες συσκευές Android δεν μπορούν να εκτελέσουν τις νεότερες εκδόσεις της πλατφόρμας. Αυτό σημαίνει ότι οι προγραμματιστές Android πρέπει συχνά να λαμβάνουν υπόψη τους πολλές διαφορετικές εκδόσεις SDK αν θέλουν να καλύπτουν το μεγαλύτερο μέρος των χρηστών. Το θετικό είναι πως η ανάπτυξη με τα εργαλεία του Android διευκολύνει τη διαδικασία ακόμη περισσότερο.

Αντίθετα από άλλες πλατφόρμες ανάπτυξης εφαρμογών κινητών τηλεφώνων, στην πλατφόρμα Android δεν γίνεται καμία διάκριση μεταξύ των εσωτερικών, έμφυτων εφαρμογών και των εφαρμογών που δημιουργούν ανεξάρτητοι προγραμματιστές. Εφόσον η εφαρμογή έχει λάβει τα απαραίτητα δικαιώματα, όλες οι εφαρμογές έχουν την ίδια πρόσβαση σε βασικές βιβλιοθήκες και στο υλικό μέρος της συσκευής. Οι κινητές συσκευές με Android περιλαμβάνουν ένα πακέτο από εσωτερικές εφαρμογές, όπως το πρόγραμμα περιήγησης στο web και το πρόγραμμα διαχείρισης επαφών. Τρίτες εφαρμογές μπορούν να ενοποιηθούν μ' αυτές τις βασικές εφαρμογές, να τις επεκτείνουν, έτσι ώστε να παρέχουν μια πλούσια εμπειρία χρήστη ή να τις αντικαταστήσουν εντελώς με εναλλακτικές εφαρμογές.

3.2.6 Android Versions

Android Cupcake 1.5



Εικόνα 3.16: Λογότυπο Android Cupcake 1.5

Κυριότερα χαρακτηριστικά :

- Ικανότητα παρακολούθησης και εγγραφής video μέσω βιντεοκάμερας.
- Καινούρια widgets και φάκελοι που μπορούν να δημοσιευτούν στην αρχική οθόνη.
- Δημοσίευση πολυμέσων σε κοινωνικά δίκτυα και σε υπηρεσίες όπως YouTube, Picasa απευθείας από το τηλέφωνο.
- Κινούμενες μεταβάσεις οθόνης.
- Υποστηρίζει τα πρότυπα Bluetooth, AVRCP και A2DP.
- Δυνατότητα απομακρυσμένης σύνδεσης με συσκευές που υποστηρίζουν το πρότυπο Bluetooth ενώ βρίσκονται στο πεδίο εμβέλειάς τους.

Android Donut 1.6



Εικόνα 3.17: Λογότυπο Android Donut 1.6

Κυριότερα χαρακτηριστικά :

- Ενημέρωση σχετικά με την υποστήριξη τεχνολογιών για CDMA/EVDO, 802.1x, VPNs και με μηχανή μετατροπής κειμένου σε ομιλία (text-to-speech).
- Υποστήριξη για ανάλυση οθονών WVGA.
- Βελτιωμένο Android Market.
- Ενημέρωση της δυνατότητας για φωνητική αναζήτηση, με ταχύτερη και αποδοτικότερη απόκριση, συμπεριλαμβανομένης της δυνατότητας να καλούμε επαφές.
- Ενημέρωση εφαρμογής αναζήτησης με την δυνατότητα εύρεσης σελιδοδεικτών και ιστορικού επαφών.
- Δυνατότητα διαγραφής πολλαπλών φωτογραφιών από το άλμπουμ.



Εικόνα 3.18: Λογότυπο Android Eclair 2.0 – 2.1

Κυριότερα χαρακτηριστικά :

- Υποστήριξη περισσότερων αναλύσεων σε οθόνες συσκευών.
- Ανανεωμένο και βελτιωμένο User Interface
- Βελτιωμένοι χάρτες για την εφαρμογή Google Maps.
- Υποστήριξη Microsoft Exchange.
- Bluetooth 2.1.
- Η κλάση Motion Event βελτιώθηκε έτσι ώστε οι κατασκευαστές να έχουν καλύτερη αντίληψη στα γεγονότα πολλαπλής αφής.
- Βέλτιστη ταχύτητα υλικού.
- Καινούριο UI για την μηχανή αναζήτησης.
- Υποστήριξη του προτύπου HTML5.



Εικόνα 3.19: Λογότυπο Android Froyo 2.2 – 2.2.3

Κυριότερα χαρακτηριστικά :

- Ενσωμάτωση της μηχανής αναζήτησης Javascript για το Chrome.
- Βελτίωση της υποστήριξης για το Microsoft Exchange.
- Βελτιστοποιήσεις στην ταχύτητα του λειτουργικού συστήματος, στη μνήμη και στην απόδοση.
- Ανανεωμένος εκκινητής εφαρμογών (application launcher), με συντομεύσεις για τις εφαρμογές τηλεφώνου και εφαρμογές αναζήτησης.
- Λειτουργία δυναμικής ζώνης WiFi hotspot.
- Υποστήριξη Adobe Flash 10.1.
- Δυνατότητα κλήσης μέσω φωνητικών εντολών και αποστολής επαφών με χρήση του Bluetooth.
- Ανανεωμένη εφαρμογή Αγοράς (Android Market) με αυτόματη ανανέωση.

Android Gingerbread 2.3 – 2.3.7



Εικόνα 3.20: Λογότυπο Android Gingerbread 2.3 - 2.3.7

Κυριότερα χαρακτηριστικά :

- Βελτιωμένο περιβάλλον χρήστη με σκοπό τη διασφάλιση απλότητας και ταχύτητας.
- Δυνατότητα επιλογής λέξεων και υποστήριξη αντιγραφής-επικόλλησης με ένα άγγιγμα.
- Βελτιωμένη ενεργειακή διαχείριση (energy saving).
- Υποστήριξη NFC (Near Field Communication).
- Υποστήριξη κλήσεων με βίντεο.
- Υποστήριξη του WebM για αναπαραγωγή βίντεο.

Android Honeycomb 3.0 – 3.2.6



Εικόνα 3.21: Λογότυπο Android Honeycomb 3.0 – 3.2.6

Κυριότερα χαρακτηριστικά :

- Υποστήριξη σε διπύρηνους και τετραπύρηνους επεξεργαστές.
- Βελτιωμένη υποστήριξη σε tablets.
- Video chat μέσω Google Talk.
- Google eBooks.
- Ανάπτυξη σεναρίων κώδικα (scripting) με υποστήριξη γραφικών τριών διαστάσεων, με χρήση της γλώσσας Renderscript.

Android Ice Cream Sandwich 4.0 – 4.0.4



Εικόνα 3.22: Λογότυπο Android Ice Cream Sandwich 4.0 – 4.0.4

Κυριότερα χαρακτηριστικά :

- Χρησιμοποιείται σε πληθώρα συσκευών όπως Smartphones, tablets, laptops, netBooks κλπ.
- Ολογραφικό 3D UI.
- Ενισχυμένο multitasking και νέα APIs.
- Πλήρως open source .
- Πλούσια επικοινωνία με νέο ημερολόγιο API, Android Beam για NFC βάση, Wi-Fi άμεση υποστήριξη, και Bluetooth για υποστήριξη συσκευών Υγείας.
- Βαθιά διαδραστικότητα και παραμετροποίηση, ηχητικός έλεγχος και βελτίωση της διαχείρισης του προγράμματος εκκίνησης.
- Χρήση επιταχυντή υλικού (Hardware-accelerated) 2D σχεδίασης, νέο grid-based layout, βελτίωση εφαρμογής πληκτρολογίου και ποντικιού, spell-checker API, και γραφίδα υποστήριξης εισόδου.

Android Jelly Bean 4.1 – 4.3.1



Android 4.1 Jelly Bean

Εικόνα 3.23: Λογότυπο Android 4.1 – 4.3.1

Κυριότερα χαρακτηριστικά :

- Δυνατότητα αυτόματης αναγνώρισης των τεχνικών χαρακτηριστικών κάθε συσκευής.
- Νέο ακόμη πιο προσιτό User Interface.
- Vsync for apps.
- Android Beam, βασισμένο στο NFC.
- Δυνατότητα εύρεσης ασύρματων δικτύων (Wi-fi) με χρήση της τεχνολογίας multicast DNS-based services.
- Υπηρεσία εύρεσης υπηρεσιών Wifi-Direct και διαχείριση εύρους δικτύου.
- Πολυκάναλος ήχος, υποστήριξη κωδικοποίησης/αποκωδικοποίησης ήχου AAC 5.1 αλλά και πρόσβαση σε codecs.
- Renderscript Computation.
- Android Browser και WebView.
- Google Cloud Messaging for Android , Google Play Services.

Android KitKat 4.4 – 4.4.4

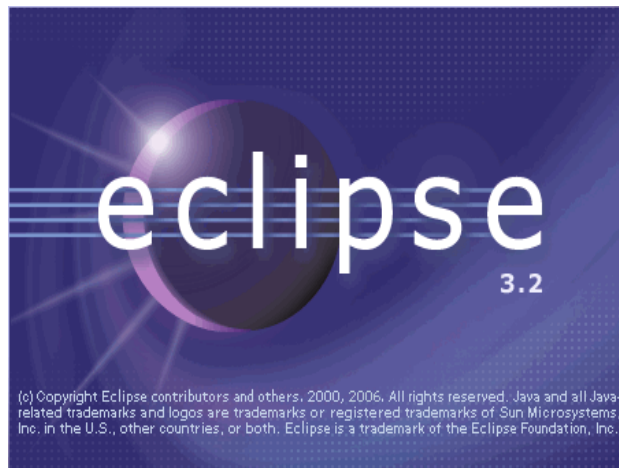


Εικόνα 3.24: Λογότυπο Android KitKat 4.4 – 4.4.4

Κυριότερα χαρακτηριστικά :

- Νέο framework για UI μεταβάσεις.
- Δυνατότητα ασύρματης εκτύπωσης.
- Βελτιστοποίηση στην απόδοση για παλαιότερες συσκευές με χαμηλότερα τεχνικά χαρακτηριστικά.

3.2.7 Eclipse



Εικόνα 3.25: Λογότυπο Eclipse 3.2

Το Eclipse αποτελεί ένα ολοκληρωμένο περιβάλλον ανάπτυξης προγραμματισμού (IDE) μέσα από το οποίο μπορούμε να γράψουμε και να εκτελέσουμε κώδικα. Επομένως μας δίνει τις απαραίτητες δυνατότητες καθώς και όλα τα υπόλοιπα προγράμματα που χρειάζονται για να εκτελέσουμε κώδικα σε Java, COBOL, Fortran, JavaScript, Perl, PHP, Python, Ruby, Scala, Groovy, Erlang και C/C++.

Επίσης είναι τόσο ελεύθερης διανομής (freeware) όσο και ανοικτού κώδικα (open source) με αποτέλεσμα να αποτελεί το πλέον ανερχόμενο περιβάλλον ανάπτυξης για κώδικα, καθώς χρησιμοποιείται από ολόένα και περισσότερους χρήστες αλλά και εταιρείες. Περιέχει μια βάση εργασίας (workspace) και ένα σύστημα για επεκτασιμότητα (plugins). Πιο συγκεκριμένα, το Eclipse παρέχει plugins για μια ευρεία ποικιλία χαρακτηριστικών όπως για UML διαγράμματα, plugin για τον DB Explorer και πολλά άλλα που ο χρήστης μπορεί να επιλέξει ανάλογα την περίπτωση.

3.2.8 Android SDK



Εικόνα 3.26: Λογότυπο Android SDK

Η ανάπτυξη εφαρμογών για τις συσκευές Android διευκολύνεται από μια ομάδα εργαλείων που παρέχονται στο SDK (Software Development Kit) το οποίο ουσιαστικά μας παρέχει τα επιπλέον εργαλεία ώστε να μπορούμε να γράψουμε κώδικα συγκεκριμένα για την κατασκευή μιας εφαρμογής Android. Η σύνδεση του Android SDK με το γραφικό μας περιβάλλον (Eclipse) γίνεται μέσω μίας επέκτασης (Android Development Tools) που εγκαθιστάμε στο Eclipse, ώστε να μπορέσουμε να μεταγλωττίσουμε την εφαρμογή μας και έπειτα να την τρέξουμε.

Καθώς το λειτουργικό σύστημα Android κυκλοφορεί σε διάφορες εκδόσεις γίνεται αντιληπτό ότι κάθε επιμέρους έκδοση θα διαθέτει διαφορετικά προγραμματιστικά εργαλεία. Επομένως μέσω του ADT Plugin μπορούμε να εγκαταστήσουμε τα εργαλεία για την υλοποίηση εφαρμογής σε οποιαδήποτε έκδοση. Για παράδειγμα αν θέλουμε η εφαρμογή μας να είναι συμβατή σε παλαιότερες εκδόσεις του Android τότε θα πρέπει να εγκαταστήσουμε τα αντίστοιχα εργαλεία και να δοκιμάσουμε την εφαρμογή μας για τις συγκεκριμένες εκδόσεις.

Στο SDK περιλαμβάνονται πολλά και χρήσιμα πράγματα όπως για παράδειγμα τα android APIs δηλαδή το σύνολο των βιβλιοθηκών που μας χρειάζονται, εργαλεία ανάπτυξης (development tools) , το android emulator με το οποίο μπορούμε να κάνουμε τις δοκιμές μας χωρίς την χρήση πραγματικής συσκευής, sample code και documentation για την λεπτομερή επεξήγηση κάθε κλάσης που μπορεί να χρησιμοποιήσει ο προγραμματιστής.

3.2.9 Αρχείο Android Manifest

Το αρχείο manifest της εφαρμογής Android είναι ένα ειδικά μορφοποιημένο αρχείο XML, το οποίο πρέπει να συνοδεύει κάθε εφαρμογή Android. Τα διάφορα components και οι ρυθμίσεις μιας Android εφαρμογής περιγράφονται στο αρχείο AndroidManifest.xml. Όλες οι δραστηριότητες (activities), οι υπηρεσίες και άλλα στοιχεία παρόχων περιεχομένου της εφαρμογής πρέπει να δηλώνονται οπωσδήποτε σε αυτό το αρχείο. Το manifest αρχείο θα πρέπει επίσης να περιέχει τα απαιτούμενα δικαιώματα για την εφαρμογή. Για παράδειγμα, αν η εφαρμογή απαιτεί πρόσβαση στο Internet, θα πρέπει να προσδιορίζονται μέσα σε αυτό το αρχείο. Εδώ ορίζεται επίσης το όνομα της εφαρμογής και πληροφορίες για την έκδοσή της, αλλά και τα στοιχεία, στα οποία βασίζεται η εφαρμογή και άλλες πληροφορίες για τη ρύθμιση των παραμέτρων της, όπως επίσης και η δήλωση του API key που διαθέτουμε. Τα δεδομένα που υπάρχουν μέσα στο XML αυτό χρησιμοποιούνται από την εφαρμογή μας για κάποιες συγκεκριμένες λειτουργίες όπως :

- Λεπτομέρειες της εφαρμογής, όπως το όνομα και η περιγραφή της.
- Εγκατάσταση και αναβάθμιση του πακέτου της εφαρμογής.
- Εκκίνηση δραστηριοτήτων της εφαρμογής (activities).
- Διαχείριση δικαιωμάτων εφαρμογής (permissions).
- Προσδιορισμός απαιτήσεων συστήματος για την εφαρμογή, για παράδειγμα ποιά SDK υποστηρίζονται (Min,Max SDK Target) και ποιές ρυθμίσεις υλικού απαιτούνται.
- Καθορισμός διαφόρων άλλων ρυθμίσεων που απαιτεί το application.
- Ταυτοποίηση του API Key μας έτσι ώστε να πάρουμε την έγκριση για τα Google Play Services.

3.2.10 To Format JSON



Εικόνα 3.27: Λογότυπο JSON

Το JSON είναι ένα ανοιχτό (open source) πρότυπο, που χρησιμοποιεί εύκολο και αναγνώσιμο κείμενο για την μετάδοση αντικειμένων δεδομένων το οποίο με τη σειρά του αποτελείται από ζεύγη χαρακτηριστικών-τιμών. Αποτελεί μια εναλλακτική λύση για την XML γλώσσα προγραμματισμού και χρησιμοποιείται κυρίως για τη μετάδοση δεδομένων μεταξύ εφαρμογών και webserVICES. Προτείνεται η χρησιμοποίηση JSON αντί XML, διότι είναι πιο ελαφρύ σε δεδομένα και κώδικα, δεν περιέχει συγκεκριμένες προκαταχωρημένες λέξεις και επίσης διαθέτει πίνακες κόνοντας το έτσι πιο κατανοητό και προσιτό στον άνθρωπο.

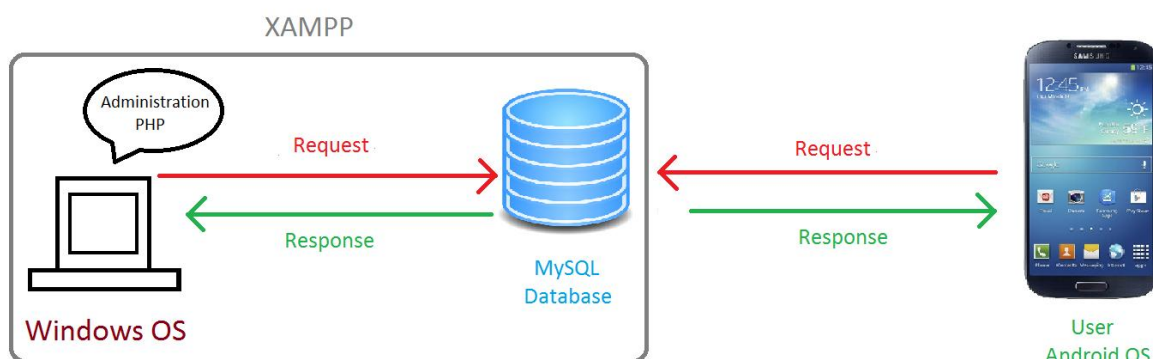
Το πρότυπο JSON μπορεί να χρησιμοποιηθεί σε ένα τεράστιο εύρος γλωσσών προγραμματισμού και όχι απαραίτητα μονάχα στην Javascript. Στο ζεύγος όνομα-τιμή το όνομα πρέπει πάντα να είναι τύπου string, ενώ η τιμή (value) μπορεί να είναι τύπου: string, boolean, αντικείμενο, αριθμός, πίνακας, ή και άδεια τιμή (null). Κάθε αντικείμενο τύπου JSON είναι μια σειρά από ζεύγη ονομάτων-τιμών και επίσης το αντικείμενο ξεκινά με ένα αριστερό υποστήριγμα ' { ' και τελειώνει με ένα δεξί υποστήριγμα ' } '. Κάθε όνομα στο ζεύγος ακολουθείται από μία άνω κάτω τελεία ' : ' και τα ζεύγη ονομάτων τιμών χωρίζονται με , (κόμμα).

4. Κύριο Μέρος

4.1 Ανάλυση Προβλήματος

Για την υλοποίηση της πτυχιακής μας εργασίας δημιουργήθηκαν τρία μέρη, η βάση δεδομένων, το Administration κομμάτι και το Android Application, που απευθύνεται στο χρήστη. Πιο συγκεκριμένα στο κομμάτι της διαχείρισης, πραγματοποιείται η εγγραφή των χρηστών και των διαχειριστών, ώστε να έχουν τη δυνατότητα πρόσβασης στο Android Application και τη διαχείριση της βάσης δεδομένων, αντίστοιχα.

Το μέρος του χρήστη και του διαχειριστή, επικοινωνούν και τα δύο με τη βάση δεδομένων, η οποία περιέχει όλα τα χρήσιμα στοιχεία που απαιτούνται τόσο για τον έλεγχο πρόσβασης του χρήστη, όσο και για τη διαχείριση των σημείων ενδιαφέροντος (spots). Τέλος, εφόσον έχει ολοκληρωθεί η διαδικασία πρόσβασης του χρήστη στην Android εφαρμογή, γίνεται εντοπισμός της θέσης του και έτσι έχει τη δυνατότητα να δει χρήσιμες πληροφορίες σχετικές με τη περιοχή όπου βρίσκεται.



Εικόνα 4.1: Σχηματική απεικόνιση λειτουργίας της Εφαρμογής

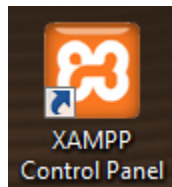
4.2 Δημιουργία της Βάσης Δεδομένων

Ένα από τα βασικά και απαραίτητα στοιχεία της εφαρμογής μας είναι η δημιουργία της βάσης δεδομένων που θα χρησιμοποιήσουμε τόσο στην εφαρμογή Android όσο και στο Administration. Για αυτό το λόγο λοιπόν δόθηκε ιδιαίτερη έμφαση και προσοχή στο σχεδιασμό και στην ανάπτυξη της, ώστε να είμαστε σίγουροι ότι χρησιμοποιήθηκαν οι σωστές ρυθμίσεις για να μπορέσει να βασιστεί ολόκληρη η εφαρμογή πάνω σε αυτή. Μετά από έρευνα και πολλές δοκιμές καταλήξαμε στη χρήση της βάσης δεδομένων σε MySQL, όπου περιέχεται μέσα στο πακέτο εφαρμογών Xampp. Στη περίπτωση του Xampp η διαχείριση των βάσεων δεδομένων γίνεται μέσω του εργαλείου phpMyAdmin το οποίο επίσης περιέχεται, προσφέροντας ένα πολύ απλό και ασφαλή τρόπο για τη δημιουργία και διαχείριση της βάσης δεδομένων. Τέλος η βάση δεδομένων φιλοξενείται από τον MySQL Community Server (GPL) όπου παρέχεται από το Xampp, ενώ η επικοινωνία με το Server γίνεται μέσω του πρωτοκόλλου TCP/IP και εκτελείτε σε τοπικό επίπεδο. Η τρέχουσα έκδοση για κάθε ένα από τα χρησιμοποιούμενα στοιχεία της εφαρμογής είναι:

- Xampp, έκδοση 3.2.1
- MySQL, έκδοση 5.0.10
- PhpMyAdmin, έκδοση 4.0.4.1
- MySQL Community Server (GPL), έκδοση 5.5.32
- TCP/IP πρωτόκολλο, έκδοση 10
- Apache HTTP Server, έκδοση 2.4.4 (Win32)
- OpenSSL, έκδοση 0.9.8y
- PHP, έκδοση 5.4.19
- Έκδοση Λειτουργικού Συστήματος Windows 7, 32bit

Για τη δημιουργία της βάσης δεδομένων θα πρέπει πρώτα να εγκαταστήσουμε το πακέτο εφαρμογών Xampp το οποίο διανέμεται δωρεάν και έπειτα να επιλέξουμε την εκκίνησή του.

- Για την εκκίνηση του Xampp πατάμε διπλό κλικ στο εικονίδιο της επιφάνειας εργασίας ή του μενού έναρξης:



Εικόνα 4.2: Συντόμευση Xampp Control Panel

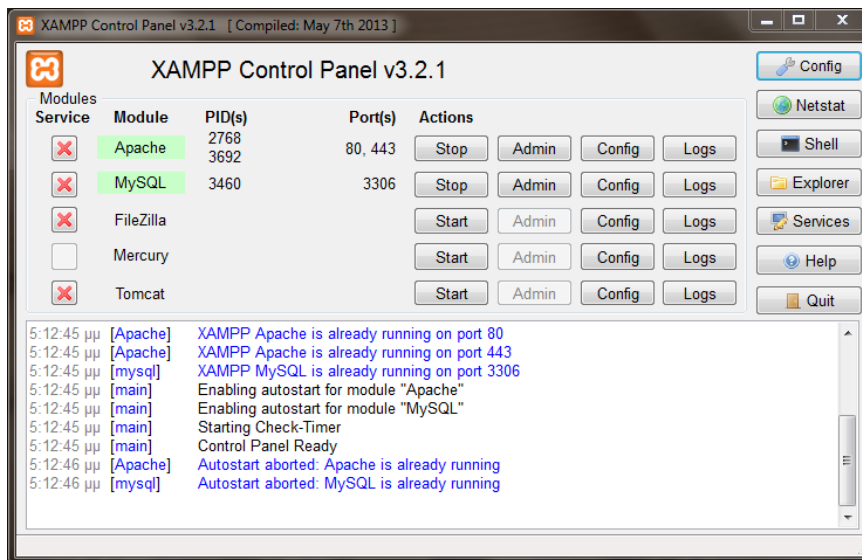
- Για να βεβαιωθούμε ότι η εφαρμογή έχει ξεκινήσει να λειτουργεί θα πρέπει να ανατρέξουμε στο κάτω δεξιά μέρος της οθόνης μας και να ψάξουμε για το παρακάτω εικονίδιο που φαίνεται στα αριστερά με πορτοκαλί χρώμα:



Εικόνα 4.3: Εικονίδιο εκτέλεσης του Xampp

Πολλές φορές, θα χρειαστεί να πατήσουμε στο βελάκι που δείχνει προς τα πάνω και να το εντοπίσουμε στη λίστα των εφαρμογών όπου θα εμφανιστούν μέσα σε ένα τετράγωνο.

- Στη συνέχεια θα πρέπει να πατήσουμε διπλό κλικ σε αυτό το εικονίδιο για να δούμε ποια από τα εργαλεία του Xampp έχουν εκκινηθεί μέσα από το Control Panel του:



Εικόνα 4.4: Xampp Control Panel

Τα στοιχεία του Xampp όπου θα χρειαστούμε και πρέπει να είναι ενεργά είναι στη στήλη των Modules τα Apache και MySQL. Αυτά τα στοιχεία ξέρουμε ότι είναι ενεργά αν υπάρχουν τιμές στη στήλη PID(s) και Port(s) και αν στη περιοχή με το λευκό φόντο κάτω εμφανίζονται οι παραπάνω αναφορές με χρώμα μαύρο και μπλε. Αν εμφανιστούν διαφορετικές αναφορές με κόκκινο χρώμα σημαίνει ότι κάποια υπηρεσία δεν ξεκίνησε και πρέπει να πατήσουμε εμείς χειροκίνητα το πλήκτρο start που θα εμφανιστεί στη στήλη Actions.

Μετά από την ολοκλήρωση των παραπάνω βημάτων είμαστε έτοιμοι να μπούμε στο phpMyAdmin με σκοπό να δημιουργήσουμε τη βάση δεδομένων. Για να μπούμε λοιπόν στο phpMyAdmin ακολουθούμε τα παρακάτω βήματα:

- Ανοίγουμε ένα web browser, στη περίπτωσή μας τον Chrome της Google και πληκτρολογούμε το παρακάτω στη γραμμή διεύθυνσης, ώστε να συνδεθούμε στην αρχική σελίδα του Xampp με σύνδεση κρυπτογράφησης OpenSSL.



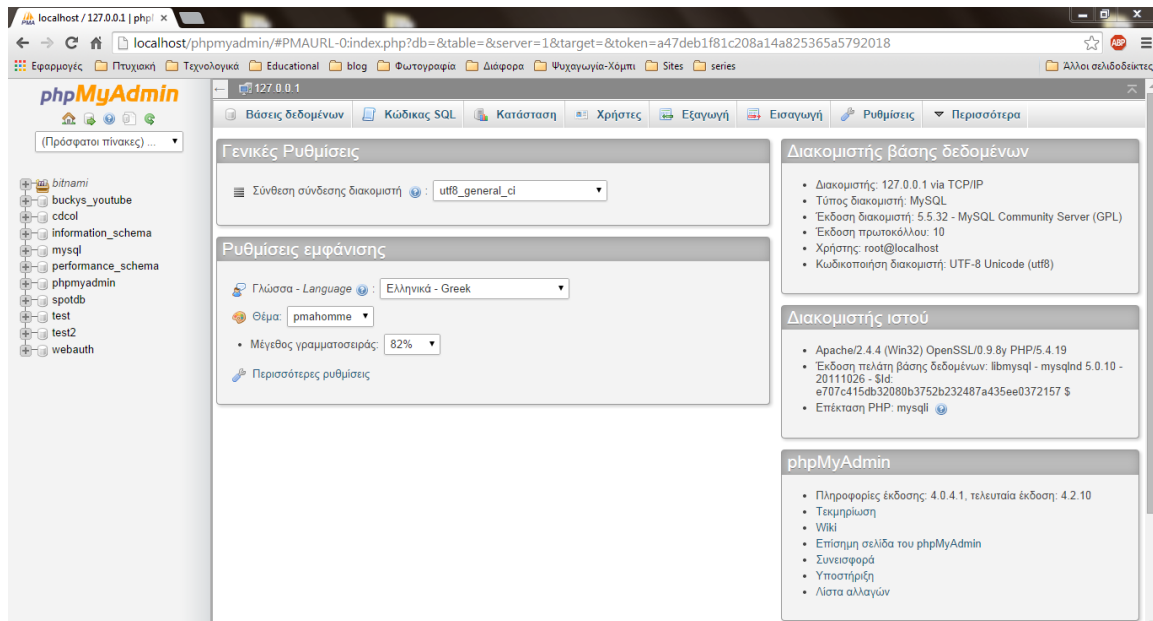
Εικόνα 4.5: Πληκτρολόγηση του localhost στο browser

➤ Αν πατήσουμε Enter θα μπούμε στην αρχική σελίδα του Xampp



Εικόνα 4.6: Αρχική σελίδα Xampp

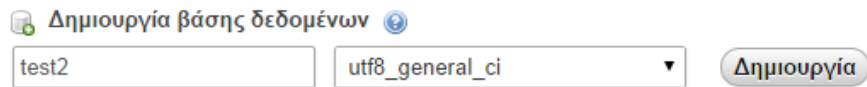
➤ Το επόμενο βήμα είναι να πατήσουμε στο μενού από αριστερά χαμηλά με όνομα Tools το υπό-μενού phpMyAdmin για να εμφανιστεί το παρακάτω:



Εικόνα 4.7: Αρχική σελίδα phpMyAdmin

- Στη συνέχεια πηγαίνοντας στο μενού που βρίσκεται στη πάνω πλευρά της οθόνης, στην επιλογή Βάσεις δεδομένων θα έχουμε τη δυνατότητα να δούμε τις βάσεις που βρίσκονται εκείνη τη στιγμή στο Server και να δημιουργήσουμε μία δική μας από την επιλογή δημιουργία με τα χαρακτηριστικά που φαίνονται:

Βάσεις δεδομένων



Δημιουργία βάσης δεδομένων ⓘ

test2 utf8_general_ci Δημιουργία

Εικόνα 4.8: Δημιουργία Βάσης Δεδομένων

- Μόλις δημιουργηθεί η βάση δεδομένων θα πρέπει να εισάγουμε τέσσερις πίνακες ακολουθώντας τα αυτοματοποιημένα μενού του phpMyAdmin. Οι πίνακες που θα χρειαστούμε καθώς και τα ονόματα τους φαίνονται εδώ:

Πίνακας	Ενέργεια	Εγγραφές	Τύπος	Σύνθεση	Μέγεθος	Περίσσεια
admins	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~3	InnoDB	utf8_general_ci	16 KB	-
spot	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~2	InnoDB	utf8_general_ci	16 KB	-
task	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~2	InnoDB	utf8_general_ci	32 KB	-
users	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~2	InnoDB	utf8_general_ci	16 KB	-
4 πίνακες	Σύνολο	9	InnoDB	utf8_general_ci	80 KB	0 B

Εικόνα 4.9: Προβολή πινάκων Βάσης Δεδομένων

- Για κάθε πίνακα που δημιουργήσαμε ακολουθούν τα ονόματα των πεδίων τους, η μορφή των δεδομένων όπου περιμένουν να μπουν στο κάθε πεδίο μαζί με το μέγεθός τους και ποιο πεδίο θα διαθέτει δεδομένα μοναδικά για κάθε πίνακα (πρωτεύον κλειδί) εμφανιζόμενο με υπογράμμιση.

Πίνακας διαχειριστών (administrators) με όνομα admins:

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα	Ενέργεια
1	id	int(100)			Όχι	Καμία	AUTO_INCREMENT	Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Περισσότερα
2	Username	varchar(30)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Περισσότερα
3	Password	char(40)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Περισσότερα
4	Email	varchar(80)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Περισσότερα

Χρήση χώρου	Στατιστικά εγγραφών
Δεδομένα 16 KB	Μορφοποίηση compact
Ευρετήριο 0 B	Σύνθεση utf8_general_ci
Σύνολο 16 KB	Επόμενη αυτόματη αριθμηση 18
	Δημιουργία 03 Οκτ 2014 στις 14:28:34

Εικόνα 4.10: Προβολή πίνακα admins

Πίνακας απλών χρηστών με όνομα users:

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα	Ενέργεια
1	Username	varchar(30)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα
2	Password	char(40)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα
3	Email	varchar(80)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα

Χρήση χώρου		Στατιστικά εγγραφών	
Δεδομένα	16 KB	Μορφοποίηση	compact
Ευρετήριο	0 B	Σύνθεση	utf8_general_ci
Σύνολο	16 KB	Δημιουργία	08 οκτ 2014 στις 11:27:16

Εικόνα 4.11: Προβολή πίνακα users

Πίνακας αποθήκευσης σημείων ενδιαφέροντος με όνομα spot:

The screenshot shows a database management interface for a table named 'spot'. The table has three columns: 'Username', 'Password', and 'Email'. The 'Username' column is of type varchar(30), 'Password' is char(40), and 'Email' is varchar(80). All columns use the utf8_general_ci collation. The 'Username' column is marked as 'Κενό' (Empty) and 'Προεπιλογή' (Default) with the value 'Καμία' (None). The 'Password' and 'Email' columns are also marked as 'Κενό' and 'Προεπιλογή' with the value 'Καμία'. The table is currently empty. The interface includes a toolbar with various actions like 'Περιήγηση', 'Δομή', 'Κώδικας SQL', 'Αναζήτηση', 'Προσθήκη', 'Εξαγωγή', 'Εισαγωγή', 'Λειτουργίες', and 'Περισσότερα'. Below the table structure, there is a 'Πληροφορία' (Information) section with a table showing usage and statistics.

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα	Ενέργεια
1	Username	varchar(30)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα
2	Password	char(40)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα
3	Email	varchar(80)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα

Χρήση χώρου	Στατιστικά εγγραφών
Δεδομένα 16 KB	Μορφοποίηση compact
Ευρετήριο 0 B	Σύνθεση utf8_general_ci
Σύνολο 16 KB	Δημιουργία 08 Οκτ 2014 στις 11:27:16

Εικόνα 4.12: Προβολή πίνακα spot

Πίνακας εργασιών με όνομα task:

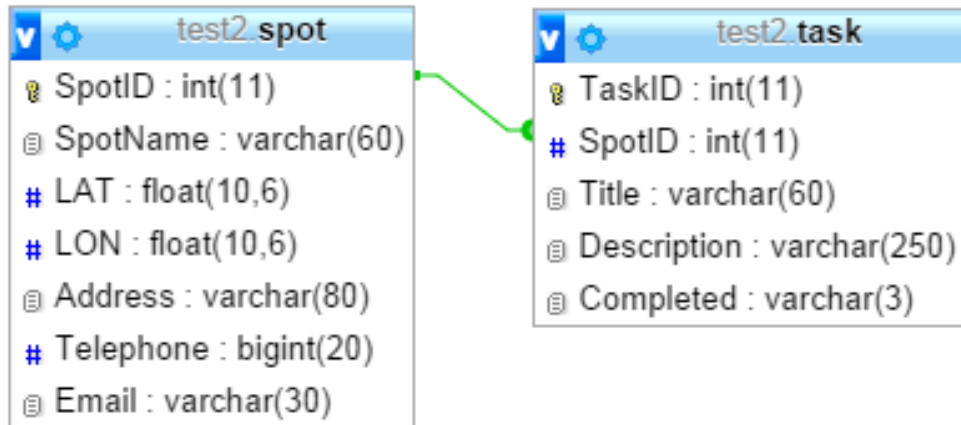
The screenshot shows a database management interface for a table named 'task'. The table has three columns: 'Username', 'Password', and 'Email'. The 'Username' column is of type varchar(30), 'Password' is char(40), and 'Email' is varchar(80). All columns use the utf8_general_ci collation. The 'Username' column is marked as 'Κενό' (Empty) and 'Προεπιλογή' (Default) with the value 'Καμία' (None). The 'Password' and 'Email' columns are also marked as 'Κενό' and 'Προεπιλογή' with the value 'Καμία'. The table is currently empty. The interface includes a toolbar with various actions like 'Περιήγηση', 'Δομή', 'Κώδικας SQL', 'Αναζήτηση', 'Προσθήκη', 'Εξαγωγή', 'Εισαγωγή', 'Λειτουργίες', and 'Περισσότερα'. Below the table structure, there is a 'Πληροφορία' (Information) section with a table showing usage and statistics.

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα	Ενέργεια
1	Username	varchar(30)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα
2	Password	char(40)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα
3	Email	varchar(80)	utf8_general_ci		Όχι	Καμία		Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Περισσότερα

Χρήση χώρου	Στατιστικά εγγραφών
Δεδομένα 16 KB	Μορφοποίηση compact
Ευρετήριο 0 B	Σύνθεση utf8_general_ci
Σύνολο 16 KB	Δημιουργία 08 Οκτ 2014 στις 11:27:16

Εικόνα 4.13: Προβολή πίνακα task

- Μετά από τον ορισμό των πινάκων και των πεδίων τους, ακολουθεί η δημιουργία των συσχετίσεων μεταξύ των πεδίων στους πίνακες. Για να γίνει αυτό πρέπει να μεταβούμε στο μενού Περισσότερα και στο υπό-μενού Σχεδιαστής. Όπως φαίνεται η συσχέτιση που έχει δημιουργηθεί είναι αυτή του πεδίου SpotID του πίνακα spot και του SpotID του πίνακα task. Έτσι όταν εισάγουμε μία νέα εγγραφή στο πίνακα task θα πρέπει να συμπληρώσουμε το πεδίο SpotID, το πεδίο δηλαδή που ορίζει σε ποιο spot της βάσης δεδομένων αντιστοιχεί η συγκεκριμένη εγγραφή του πίνακα task. Άρα το πεδίο SpotID εφόσον στο πίνακα spot αποτελεί πρωτεύον κλειδί και το ίδιο πεδίο στο πίνακα task είναι ένα απλό πεδίο, το SpotID αποτελεί ξένο κλειδί στη περίπτωση του δεύτερου πίνακα. Το είδος της σχέσης που σχηματίζεται εδώ είναι ένα-προς-πολλά. Ο λόγος είναι ότι ένα task μπορεί να ανήκει μόνο σε ένα spot, όμως ένα spot μπορεί να έχει πολλά διαφορετικά tasks.



Εικόνα 4.14: Συσχετίσεις μεταξύ πινάκων

4.3 Administration της Εφαρμογής

Το Administration αποτελεί, όπως λέει και ο τίτλος του, το σημείο από το οποίο θα γίνεται η κεντρική διαχείριση όλων των δεδομένων της εφαρμογής. Άρα, μέσα από το Administration θα μπορούν οι διαχειριστές να ελέγχουν και να επεξεργάζονται τα περιεχόμενα της βάσης δεδομένων, να προσθέτουν νέες εγγραφές στους πίνακες και να κάνουν ενημέρωση στις υπάρχουσες, καθώς και να δημιουργούν νέους χρήστες για την εφαρμογή-πελάτη σε Android. Η εφαρμογή της διαχείρισης εκτελείται μέσω web browser, ενώ θα μπορεί να λειτουργεί εξίσου καλά τόσο σε web browsers για σταθερούς υπολογιστές, ανεξαρτήτου λειτουργικού συστήματος, όσο και σε web browsers για κινητές συσκευές.

Για την ανάπτυξή του, χρησιμοποιήθηκαν τεχνολογίες όπως η HTML 5 με χρήση CSS, για τη σωστή και γρήγορη εμφάνιση της κάθε σελίδας της εφαρμογής, σε web browsers υπολογιστών και κινητών συσκευών. Για την επικοινωνία της σελίδας HTML και κατ'επέκταση των διαχειριστών της εφαρμογής, με τη βάση δεδομένων χρησιμοποιήθηκε η server-side γλώσσα scripting, PHP. Οι σελίδες της εφαρμογής φιλοξενούνται στον Apache HTTP Server, ο οποίος περιέχεται στο πακέτο εφαρμογών Xampp, που εκτελείται καθ' όλη τη διάρκεια λειτουργίας της εφαρμογής αυτής. Επίσης για την ενσωμάτωση των χαρτών Google Maps και των διάφορων εφαρμογών όπου στηρίχθηκαν σε αυτούς, χρησιμοποιήθηκε κώδικας σε JavaScript. Για τη δοκιμή και την τελική ανάπτυξη του κώδικα της εφαρμογής χρησιμοποιήθηκε το Adobe Dreamweaver CC καθώς και το notepad ++, ενώ η δοκιμή της τελικής εφαρμογής έγινε σε Ηλεκτρονικό Υπολογιστή, σε συσκευή android tablet και σε τρία smartphones με λειτουργικό σύστημα Android, με διαφορετικό μέγεθος οθόνης το κάθε ένα.

Στην ανάπτυξη του κώδικα κάθε σελίδας, έχουμε λάβει υπόψιν ρυθμίσεις που αφορούν την ασφάλεια των δεδομένων μέσα στη βάση δεδομένων, αλλά και τη διασφάλιση ότι δε θα υπάρχουν διπλότυπες εγγραφές. Με αυτό το τρόπο παρέχουμε ένα ολοκληρωμένο περιβάλλον διαχείρισης της βάσης δεδομένων, χωρίς κάποιος από τους διαχειριστές να χρειάζεται να χρησιμοποιήσει το phpMyAdmin για να κάνει οποιαδήποτε αλλαγή στους πίνακες της βάσης δεδομένων. Όλα τα ερωτήματα προς τη βάση δεδομένων γίνονται εύκολα, απλά, με ασφάλεια και αυτόματα, με το πάτημα ενός κουμπιού, χωρίς να χρειάζεται κάποιος να γνωρίζει πως λειτουργεί από τη πλευρά του κώδικα η εφαρμογή και φυσικά χωρίς να χρειάζεται γνώσεις για δημιουργία ερωτημάτων σε MySQL.

Για να μπορεί κάποιος να αποκτήσει πρόσβαση στο Administration, θα πρέπει τα στοιχεία του να βρίσκονται στη βάση δεδομένων, στο πίνακα admins. Όταν κάποιος δε βρίσκεται σε αυτό το πίνακα, τότε δε μπορεί να έχει πρόσβαση στο Administration. Το μόνο που θα μπορεί να κάνει, είναι να πατήσει το κουμπί “sign up” και να στείλει μία αίτηση με τα στοιχεία που θέλει να χρησιμοποιήσει, δηλαδή το όνομα χρήστη, τη διεύθυνση email, το κωδικό πρόσβασης και το αν θέλει να εγγραφεί σαν απλός χρήστης ή σαν διαχειριστής. Αυτόματα όμως με τη συμπλήρωση της αίτησης, γίνεται έλεγχος αν το όνομα χρήστη και η διεύθυνση email βρίσκονται ήδη στη βάση δεδομένων. Αν ένα και τα δύο βρίσκονται ήδη στη βάση δε μπορεί να συνεχιστεί η διαδικασία και πρέπει να διαλέξει διαφορετικά στοιχεία. Η αίτηση αυτή θα έρθει αυτόματα στο email των διαχειριστών οι οποίοι θα προσθέσουν χειροκίνητα το χρήστη – διαχειριστή. Για λόγους ασφαλείας, στη περίπτωση συμπλήρωσης της παραπάνω φόρμας, όπως και στις περιπτώσεις των φορμών για δημιουργία χρήστη και διαχειριστή, γίνεται έλεγχος αν οι κωδικοί πρόσβασης είναι μεταξύ 6 και 16 ψηφίων και αν οι κωδικοί και στα δύο πεδία συμπλήρωσης τους είναι ίδιοι. Αν ένα από αυτά τα δύο δεν ισχύει, τότε δε μπορεί να συνεχιστεί η αποστολή αίτησης για εγγραφή ή η εγγραφή χρήστη – διαχειριστή.

Επίσης, έχουν ληφθεί υπόψιν πολλά από τα λάθη που μπορεί να κάνει κάποιος, συμπληρώνοντας μία φόρμα. Έτσι, αν ο διαχειριστής πληκτρολογήσει πάρα πολλά κενά κατά λάθος, τότε αυτά θα μειωθούν σε ένα κενό ανά λέξη. Επιπροσθέτως αν πριν από το όνομα ενός spot, για παράδειγμα, κάποιος πληκτρολογήσει ένα κενό, κατά λάθος, τότε θα συνεχιστεί η αποθήκευση του ονόματος στη βάση, αλλά χωρίς το κενό μπροστά από το όνομα. Στη περίπτωση όπου πρέπει να εισαχθεί ο τηλεφωνικός αριθμός ενός νέου spot, αν μεταξύ των αριθμών πληκτρολογηθεί κενό, τότε θα αποθηκευτεί στη βάση δεδομένων χωρίς το κενό ανάμεσα στους αριθμούς.

Στις περιπτώσεις όπου ο διαχειριστής πρέπει να συμπληρώσει κάποια φόρμα, όλα τα πεδία της φόρμας, των οποίων ο τίτλος έχει το σύμβολο “*”, πρέπει να συμπληρωθούν υποχρεωτικά. Σε

αντίθετη περίπτωση, εμφανίζεται μήνυμα σφάλματος και τον προτρέπουμε να μεταβεί πίσω στη σελίδα που ήταν και να προσπαθήσει ξανά. Αν σε υποχρεωτικά πεδία της φόρμας, αντί να πληκτρολογήσει κάποιο χαρακτήρα, πληκτρολογήσει μόνο το χαρακτήρα κενό, τότε εμφανίζεται μήνυμα λάθους και δε μπορεί να συνεχίσει αν δε το διορθώσει.

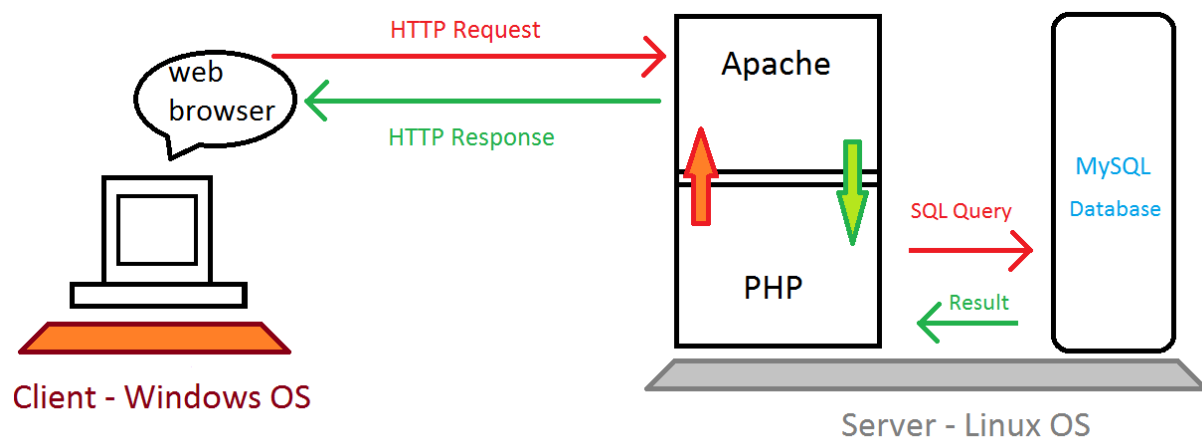
Αν κάποιο πεδίο δε συμπληρωθεί με το σωστό τύπο δεδομένων, τότε εμφανίζεται το ανάλογο μήνυμα. Δηλαδή αν στο πεδίο της διεύθυνσης Email πληκτρολογήσουμε το τηλεφωνικό αριθμό, θα εμφανιστεί μήνυμα λάθους.

Με σκοπό τον έλεγχο των διπλότυπων εγγραφών στη βάση δεδομένων, γίνονται έλεγχοι κάθε φορά που πάει κάποιος διαχειριστής να προσθέσει μία εγγραφή στη βάση δεδομένων, είτε αυτή είναι ένα spot, ένα task, ένας χρήστης ή ένας διαχειριστής. Χρησιμοποιούνται και διάφορες συναρτήσεις ώστε να αποτρέψουν τις εξής περιπτώσεις: όταν σε μία νέα εγγραφή στο πίνακα spot, για παράδειγμα, ένας διαχειριστής προσπαθήσει να βάλει ένα όνομα με κεφαλαία γράμματα ή με μικρά ή σε συνδυασμό κεφαλαία – μικρά γράμματα και στη βάση υπάρχει ήδη το ίδιο όνομα spot, σε οποιαδήποτε συνδυασμό κεφαλαίων – μικρών χαρακτήρων, τότε θα αποτραπεί η προσθήκη αυτού του ονόματος για αποφυγή διπλότυπου. Για τον ίδιο ακριβώς λόγο, γίνεται η διαγραφή των επιπλέον κενών και του κενού στην αρχή των δεδομένων που εισάγει κάποιος, μέσω των κατάλληλων συναρτήσεων της PHP, ώστε να γίνει αποφυγή των διπλότυπων.

Ειδικές συναρτήσεις της PHP και έλεγχοι εκτελούνται επίσης, πριν την εισαγωγή των στοιχείων που πληκτρολογεί κάποιος στα πεδία των φορμών. Έτσι εκτελείτε έλεγχος για το αν οι συμβολοσειρές σε κάθε πεδίο της φόρμας ακολουθούν τους κανόνες σύνταξης της βάσης δεδομένων. Για αυτό το λόγο, με τη χρήση ειδικής συνάρτησης, γίνεται εισαγωγή κάθετων ανάμεσα στις λέξεις, στα κενά και στα σύμβολα που πληκτρολογούνται.

Τέλος, σε κάθε φόρμα υπάρχει το κουμπί Reset στο κάτω μέρος, όπου βοηθάει στη διαγραφή όλων των δεδομένων της φόρμας αν κάτι έχει γίνει λάθος και πρέπει να τη συμπληρώσει κάποιος από την αρχή. Η γλώσσα εισαγωγής είναι τα Αγγλικά και Ελληνικά με λατινικούς χαρακτήρες.

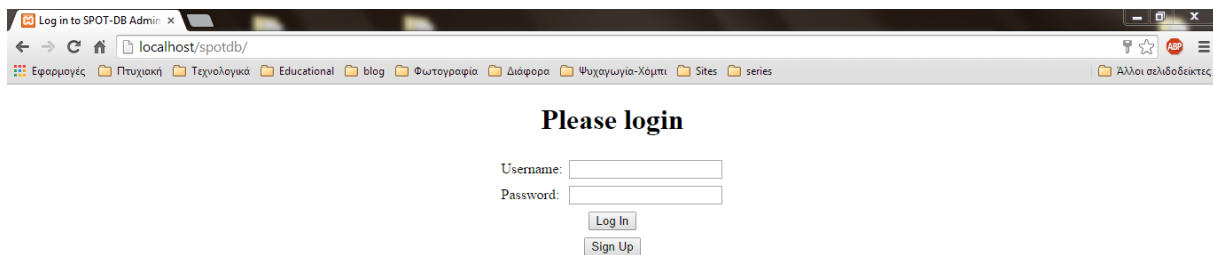
Όλες οι σελίδες του Administration, βρίσκονται στο φάκελο “spotdb” του φακέλου “htdocs”, που βρίσκεται μέσα στο φάκελο εγκατάστασης του Xampp. Επειδή η σελίδα σύνδεσης έχει όνομα index.html, το μόνο που χρειάζεται, όπως θα δούμε και παρακάτω, είναι απλά να πληκτρολογήσουμε στο web browser, το όνομα του φακέλου και αυτόματα θα φορτωθεί στο web browser η σελίδα σύνδεσης.



Εικόνα 4.15: Σχηματική απεικόνιση Administration

Εκκίνηση του Administration και λειτουργία σε Ηλεκτρονικό Υπολογιστή:

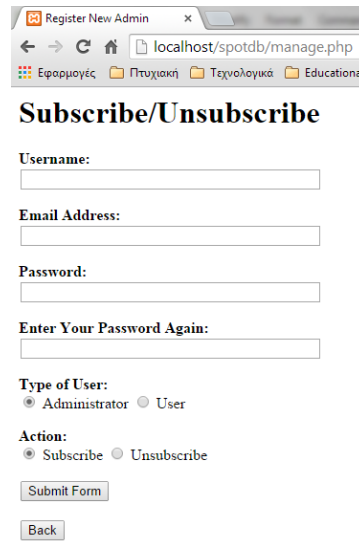
- Το πρώτο βήμα για να θέσουμε σε λειτουργία την εφαρμογή διαχείρισης, είναι να ξεκινήσουμε την εφαρμογή Xampp, με τον ίδιο ακριβώς τρόπο όπου τη ξεκινήσαμε στη περίπτωση της βάσης δεδομένων, κάνοντας ακριβώς τους ίδιους ελέγχους εκκίνησης στοιχείων. Μας ενδιαφέρει δηλαδή να έχουν ξεκινήσει οι υπηρεσίες: Apache και MySQL.
- Στη συνέχεια θα πάμε στο web browser που χρησιμοποιούμε, στη περίπτωση μας τον Chrome και θα πληκτρολογήσουμε στη γραμμή διεύθυνσης το “localhost/spotdb”, όπως φαίνεται και παρακάτω. Πληκτρολογώντας αυτό θα μας μεταφέρει στο αρχείο index.html όπου βρίσκεται μέσα στο φάκελο spotdb του Apache HTTP Server. Στη περίπτωση όπου θέλουμε να εκτελέσουμε το Administration από άλλο υπολογιστή (όχι από εκείνον που εκτελείται το Xampp δηλαδή) ή από κάποια κινητή συσκευή θα πρέπει αντί localhost, να πληκτρολογήσουμε τη διεύθυνση IP του υπολογιστή στον οποίο τρέχει το Xampp, άρα και ο Apache HTTP Server. Για παράδειγμα αν ο υπολογιστής όπου εκτελείται ο Server έχει διεύθυνση IP: 192.168.2.4, τότε στο web browser μίας άλλης συσκευής, εντός του τοπικού δικτύου, θα πληκτρολογήσω “ 192.168.2.4/spotdb”.



Εικόνα 4.16: Σελίδα σύνδεσης

Όπως βλέπουμε η πρώτη σελίδα που εμφανίζεται είναι αυτή όπου καλεί κάποιον, που ισχυρίζεται ότι είναι διαχειριστής, να εισέλθει στην εφαρμογή πληκτρολογώντας το όνομα χρήστη και το κωδικό πρόσβασης του και πατώντας το κουμπί “Log-in”. Εφόσον κάθε εγγεγραμμένος administrator βρίσκεται σε συγκεκριμένο πίνακα της βάσης δεδομένων, όταν κάποιος πληκτρολογήσει τα στοιχεία του και πατήσει το “Log-in”, έχουμε δημιουργήσει ερωτήματα μέσω της PHP και της MySQL προς τη βάση δεδομένων, όπου γίνεται αναζήτηση για το αν ο συνδυασμός αυτός του ονόματος και του κωδικού υπάρχουν στη βάση. Αν υπάρχουν στη βάση τότε πιστοποιείται ότι ο χρήστης αυτός είναι administrator και θα του εμφανιστεί η κεντρική σελίδα του Administration. Αν δεν βρίσκεται στο πίνακα των admins τότε δε θα μπορεί να συνεχίσει και θα πρέπει να πάει πίσω και να προσπαθήσει ξανά να εισέλθει ή να κάνει αίτηση εγγραφής, πατώντας το κουμπί “Sign up”. Στη περίπτωση της εφαρμογής αυτής όμως, οι administrators είναι μόνο δύο και μόνο αυτοί έχουν το δικαίωμα να διαγράψουν ή να προσθέτουν χρήστες και άλλους διαχειριστές στην εφαρμογή.

- Αν λοιπόν ο χρήστης δε πιστοποιηθεί ότι είναι εγγεγραμμένος έχει τη δυνατότητα να συμπληρώσει μία αίτηση μέσω της οποίας θα σταλεί email στους διαχειριστές, όπου με τη σειρά τους θα προχωρήσουν στην εγγραφή του χρήστη ή όχι.

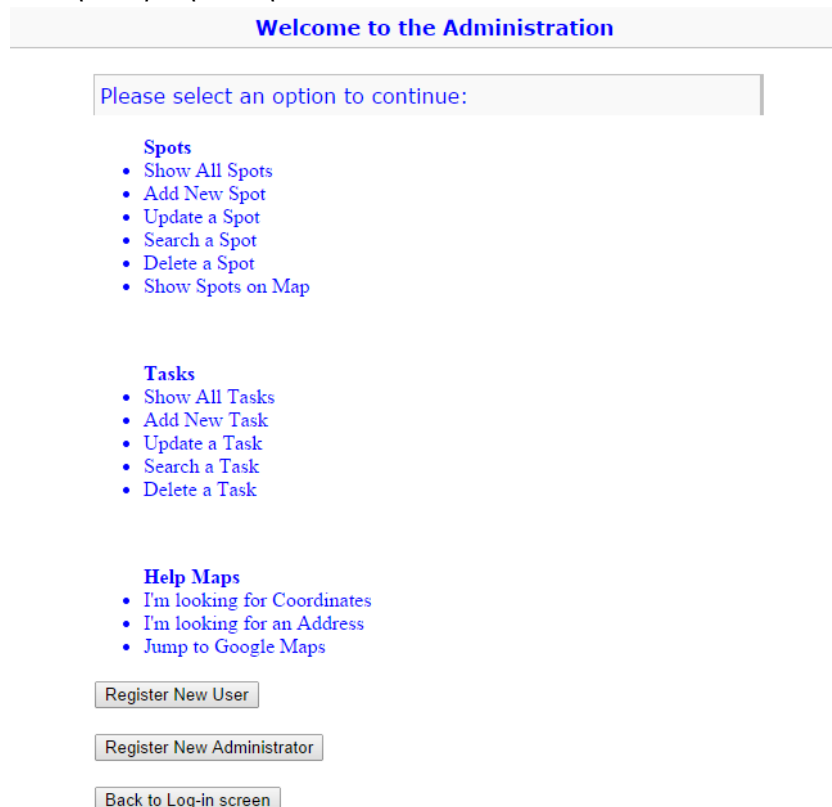


The screenshot shows a web browser window with the address bar displaying 'localhost/spotdb/manage.php'. The page title is 'Register New Admin'. The main heading is 'Subscribe/Unsubscribe'. The form contains the following fields and options:

- Username:** A text input field.
- Email Address:** A text input field.
- Password:** A text input field.
- Enter Your Password Again:** A text input field.
- Type of User:** Radio buttons for 'Administrator' (selected) and 'User'.
- Action:** Radio buttons for 'Subscribe' (selected) and 'Unsubscribe'.
- Submit Form:** A button.
- Back:** A button.

Εικόνα 4.17: Σελίδα αίτησης εγγραφής

- Στη περίπτωση όπου ο χρήστης βρίσκεται στο πίνακα των administrators τότε θα μεταφερθεί αυτόματα στη κεντρική οθόνη του Administration.



The screenshot shows the Administration page with the following content:

- Welcome to the Administration** (blue text)
- Please select an option to continue:** (blue text)
- Spots**
 - Show All Spots
 - Add New Spot
 - Update a Spot
 - Search a Spot
 - Delete a Spot
 - Show Spots on Map
- Tasks**
 - Show All Tasks
 - Add New Task
 - Update a Task
 - Search a Task
 - Delete a Task
- Help Maps**
 - I'm looking for Coordinates
 - I'm looking for an Address
 - Jump to Google Maps
- Register New User** (button)
- Register New Administrator** (button)
- Back to Log-in screen** (button)

Εικόνα 4.18: Αρχική σελίδα Administration

Όπως φαίνεται στη παραπάνω εικόνα, μέσα από τη κεντρική σελίδα του Administration ο διαχειριστής μπορεί να διακρίνει τρεις ομάδες επιλογών και τρεις ακόμα επιλογές σε μορφή κουμπιών στο κάτω μέρος.

- Η πρώτη ομάδα αφορά δραστηριότητες και δημιουργία ερωτημάτων σχετικά με τη διαχείριση σημείων ενδιαφέροντος, άρα έχουμε να κάνουμε με το πίνακα spot.
- Η δεύτερη ομάδα επιλογών αφορά τον πίνακα task και τις δραστηριότητες όπου ανακοινώνονται για το κάθε spot.
- Η τρίτη ομάδα αποτελείται από βοηθητικές επιλογές, με σκοπό να διευκολύνει το διαχειριστή στη δημιουργία και επεξεργασία των σημείων ενδιαφέροντος με χρήση χαρτών Google Maps.
- Η ομάδα επιλογών σε μορφή κουμπιών όπου βρίσκονται στο τέλος της σελίδας, αφορά τη δημιουργία ή διαγραφή κάποιου administrator και τη δημιουργία ή διαγραφή κάποιου απλού χρήστη. Άρα, μέσω αυτών των κουμπιών γίνεται η προσθαφαίρεση από τους διαχειριστές, εγγραφών στους πίνακες admins και users αντίστοιχα.
- Το τελευταίο κουμπί όπου εμφανίζεται στη κεντρική σελίδα, είναι αυτό με τίτλο “Back to login screen” και είναι ένα είδος αποσύνδεσης χρήστη και επαναφορά στη σελίδα σύνδεσης διαχειριστή.

Κάτι που θα παρατηρήσει κάποιος, είναι η εργαλειοθήκη όπου βρίσκεται στην αριστερή πλευρά της κάθε σελίδας, εκτός της κεντρικής και παρέχει στο διαχειριστή τη δυνατότητα να μεταβεί στις βασικές σελίδες του Administration και στη κεντρική ανα πάσα στιγμή. Ακολουθούν εικόνες από όλες τις παραπάνω επιλογές με τη σειρά όπου είναι στην παραπάνω εικόνα της κεντρικής σελίδας, όπως θα τις έβλεπε ένας administrator με δικαίωμα σύνδεσης.

Spots

Τίτλος σελίδας: Show All Spots

Show all the spots of the database

Results:

Number of spots: 2

1. Spot's ID: 67
Spot's Name: Tei of Crete
Latitude: 35.316158
Longitude: 25.102825
Address: Heraklion, Estavromenos, 71004
Telephone: 2810379308
Email Address: info@staff.teicrete.gr

2. Spot's ID: 743
Spot's Name: Heraklion International Airport
Latitude: 35.339976
Longitude: 25.181631
Address: New Alikarnassos, 71601
Telephone: 2810397129
Email Address: info@heraklionairport.net

Εικόνα 4.19: Σελίδα προβολής όλων των Spots

Εδώ βλέπουμε όλες τις εγγραφές του πίνακα spot με λεπτομέρειες όπως είναι το ID, το όνομα, το γεωγραφικό πλάτος, το γεωγραφικό μήκος, η διεύθυνση με το ταχυδρομικό κώδικα, ο τηλεφωνικός αριθμός και η διεύθυνση E-mail.

Τίτλος σελίδας: Add New Spot

Add new spot in database

Please insert the data of the new spot.
*(required fields)

*Spot's name:
*Spot's Latitude:
*Spot's Longitude:
*Address, City, State:
Telephone:
*Email:

Εικόνα 4.20: Σελίδα δημιουργίας νέου Spot

Σε αυτή τη σελίδα μπορεί ο διαχειριστής να προσθέσει μία νέα εγγραφή στο πίνακα spot συμπληρώνοντας όλα τα πεδία με αστερίσκο δίπλα στο τίτλο τους, τα οποία είναι υποχρεωτικά. Στο κώδικα της σελίδας, γίνεται μέσω της PHP έλεγχος αν υπάρχει άλλη εγγραφή στο πίνακα spot, με το ίδιο όνομα και με τις ίδιες συντεταγμένες, αν υπάρχει τότε η διαδικασία σταματάει και θα πρέπει να αλλαχθούν αυτά τα στοιχεία.

Τίτλος σελίδας: Update a Spot

Update a spot in the database

Please complete the search form before you update the spot in order to note the **Spot's ID** and keep the results in another tab in order to copy the fields that won't change.

In order to update a Spot you need to complete the form below.
*(required fields)

Edit a spot by it's <i>ID</i>	
*Spot's ID	<input type="text"/>
*Spot's Name	<input type="text"/>
*Latitude	<input type="text"/>
*Longitude	<input type="text"/>
*Address, City, State	<input type="text"/>
Telephone	<input type="text"/>
*Email Address	<input type="text"/>

Εικόνα 4.21: Σελίδα ενημέρωσης Spot

Μέσω αυτής της σελίδας, ο διαχειριστής μπορεί να κάνει ενημέρωση στα στοιχεία μίας εγγραφής στο πίνακα spot. Για να γίνει αυτό θα πρέπει πρώτα να γνωρίζει το ID του spot όπου θέλει να αλλάξει. Αν οι εγγραφές του spot είναι λίγες τότε ανοίγοντας τη σελίδα “Show All Spots”, μπορεί να βρει το ID του spot που τον ενδιαφέρει και να συμπληρώσει τη παραπάνω φόρμα επεξεργασίας του spot. Διαφορετικά αν υπάρχουν πολλά spots θα πρέπει να ολοκληρώσει πρώτα την αναζήτηση, τη διαδικασία της οποίας περιγράφουμε αμέσως μετά και έτσι θα βρει το ID που τον ενδιαφέρει.

Τίτλος σελίδας: Search a Spot

Search a spot in the database

The screenshot shows a web application interface for searching spots. On the left, there is a navigation menu with three sections: 'Home', 'Spots', and 'Help Maps'. The 'Spots' section is active and contains links for 'Show All Spots', 'Add New Spot', 'Update a Spot', 'Search a Spot', 'Delete a Spot', and 'Show Spots on Map'. The 'Help Maps' section contains links for 'I'm looking for Coordinates', 'I'm looking for an Address', and 'Jump to Google Maps'. The main content area features a search form with the following elements: a header 'Please complete the search form', a 'Choose Search type:' dropdown menu with 'Spot's ID' selected, an 'Enter search Term:' text input field, and 'Search' and 'Reset' buttons.

Εικόνα 4.22: Σελίδα αναζήτησης Spot

Ο διαχειριστής έχει τη δυνατότητα να επιλέξει τον όρο αναζήτησης όπου γνωρίζει και να του εμφανιστούν όλα τα στοιχεία της συγκεκριμένης εγγραφής. Δεν είναι απαραίτητο, για παράδειγμα, να γνωρίζει όλο το τίτλο του spot, αρκεί να γνωρίζει μέρος του και έτσι θα μπορεί να γίνει η αναζήτηση. Στην επόμενη εικόνα φαίνονται οι επιλογές, σε όρους αναζήτησης όπου έχει ο διαχειριστής.

Όροι αναζήτησης:

This screenshot shows the search form with the 'Choose Search type:' dropdown menu expanded. The menu lists the following search criteria: 'Spot's ID', 'Spot's ID', 'Spot's Name', 'Latitude', 'Longitude', 'Address, City, State', 'Telephone', and 'Email Address'. The 'Email Address' option is currently selected and highlighted in blue. The rest of the form, including the text input and 'Search'/'Reset' buttons, remains visible in the background.

Εικόνα 4.23: Όροι αναζήτησης Spot

Τίτλος σελίδας: Delete a Spot

Delete a spot from the database

Please complete the search form before you delete the spot in order to note the **Spot's ID** and the **Spot's Name** as it is in the database.

In order to delete a Spot you need to complete the form below. Both fields have to be completed.

Insert the Spot's ID:

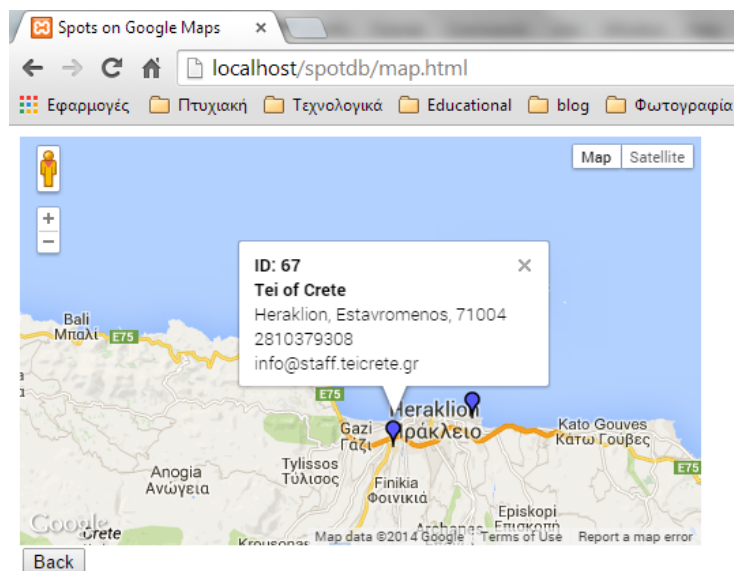
Insert Spot's Name:

Delete Reset

Εικόνα 4.24: Σελίδα διαγραφής Spot

Όπως και στη περίπτωση της ενημέρωσης μία εγγραφής, έτσι και εδώ είναι απαραίτητο να γνωρίζει ο διαχειριστής το ID του spot που θα διαγράψει, αλλά αυτή τη φορά για λόγους ασφαλείας, θα πρέπει να ξέρει και μέρος του ονόματος του spot. Τα παραπάνω στοιχεία μπορεί να τα βρει είτε από την αναζήτηση, είτε από το μενού “Show All Spots”.

Τίτλος σελίδας: Show Spots on Map



Εικόνα 4.25: Σελίδα προβολής Spots σε Χάρτη

Με αυτό το τρόπο βλέπουμε τις εγγραφές του πίνακα spot, σε μορφή σημείων σε χάρτη και αν πατήσουμε πάνω σε κάθε σημείο εμφανίζονται πληροφορίες όπως βρίσκονται στη Βάση Δεδομένων.

Tasks

Τίτλος σελίδας: Show All Tasks

Show all the tasks of the database

Results:

Number of tasks: 2

1. Task's ID: 1
Spot's ID: 67
Spot's Name: Tel of Crete
Task's Title: Fix Net problem on Pc
Task's Description: Lab 12, Pc no. 2 internet access problem.
Completed Yes/No: No

2. Task's ID: 3
Spot's ID: 743
Spot's Name: Heraklion International Airport
Task's Title: New website
Task's Description: New website with wordpress theme
Completed Yes/No: No

Home

Spots

- Show All Spots
- Add New Spot
- Update a Spot
- Search a Spot
- Delete a Spot
- Show Spots on Map

Tasks

- Show All Tasks
- Add New Task
- Update a Task
- Search a Task
- Delete a Task

Help Maps

- I'm looking for Coordinates
- I'm looking for an Address
- Jump to Google Maps

Εικόνα 4.26: Σελίδα προβολής όλων των Tasks

Με τον ίδιο τρόπο όπως στα spots, εμφανίζονται εδώ όλες οι εργασίες όπου έχουν εισαχθεί στη βάση δεδομένων, μαζί με το ID και το όνομα του spot που αφορά το κάθε task.

Τίτλος σελίδας: Add New Task

Add new task in database

Please insert the data of the new task.
*(required fields)

Spot's ID:

*Task's Title:

*Task's Description:

Home

Spots

- Show All Spots
- Add New Spot
- Update a Spot
- Search a Spot
- Delete a Spot
- Show Spots on Map

Tasks

- Show All Tasks
- Add New Task
- Update a Task
- Search a Task
- Delete a Task

Help Maps

- I'm looking for Coordinates
- I'm looking for an Address
- Jump to Google Maps

Εικόνα 4.27: Σελίδα δημιουργίας νέου Task

Εισαγωγή νέας εργασίας στη βάση δεδομένων. Τα δύο από τα τρία πεδία αυτά είναι υποχρεωτικά να συμπληρωθούν. Το ID του spot μπορεί να μη το γνωρίζει ο διαχειριστής και μπορεί να το συμπληρώσει αργότερα, κάνοντας απλά update στο task, όπως φαίνεται παρακάτω.

Τίτλος σελίδας: Update a Task

Update a task in the database

• Home

Spots

- Show All Spots
- Add New Spot
- Update a Spot
- Search a Spot
- Delete a Spot
- Show Spots on Map

Tasks

- Show All Tasks
- Add New Task
- Update a Task
- Search a Task
- Delete a Task

Help Maps

- I'm looking for Coordinates
- I'm looking for an Address
- Jump to Google Maps

In order to continue you need to know the Task's ID you want to update

Insert the Task's ID:

You can change only one field each time

Spot's ID ▼

Insert the new data of the field:

Update | Reset

Εικόνα 4.28: Σελίδα ενημέρωσης Task

Εδώ μπορούμε να αλλάξουμε τα στοιχεία μίας εργασίας, πληκτρολογώντας το ID της και συμπληρώνοντας ένα πεδίο κάθε φορά, όπως φαίνεται στην εικόνα:

Επιλογή πεδίου για αλλαγή:

In order to continue you need to know the Task's ID you want to update

Insert the Task's ID:

You can change only one field each time

Spot's ID ▼

Insert the new data of the field:

Update | Reset

Εικόνα 4.29: Σελίδα επιλογής πεδίου ενημέρωσης

Τίτλος σελίδας: Search a Task

Search a task in the database

The screenshot shows a web page titled "Search a Task" with a search form. The form has a header "Please complete the search form". Below it is a section "Choose Search type:" with a dropdown menu currently showing "Task's ID". Underneath is a section "Enter search Term:" with a text input field. At the bottom of the form are "Search" and "Reset" buttons. On the left side, there is a navigation menu with three sections: "Home", "Spots", and "Tasks". The "Spots" section includes links for "Show All Spots", "Add New Spot", "Update a Spot", "Search a Spot", "Delete a Spot", and "Show Spots on Map". The "Tasks" section includes links for "Show All Tasks", "Add New Task", "Update a Task", "Search a Task", and "Delete a Task". The "Help Maps" section includes links for "I'm looking for Coordinates", "I'm looking for an Address", and "Jump to Google Maps".

Εικόνα 4.30: Σελίδα αναζήτησης Task

Αναζήτηση μίας εργασίας στη βάση δεδομένων επιλέγοντας τον όρο της αναζήτησης.

Όροι αναζήτησης:

This is a close-up view of the search form. The "Choose Search type:" dropdown menu is open, showing five options: "Task's ID", "Spot's ID", "Task's Title", "Description", and "Completed Yes/No". The "Task's ID" option is currently selected. Below the dropdown is a text input field for the search term, and at the bottom are "Search" and "Reset" buttons.

Εικόνα 4.31: Όροι αναζήτησης Task

Τίτλος σελίδας: Delete a Task

Delete a task from the database

- Home

Spots

- Show All Spots
- Add New Spot
- Update a Spot
- Search a Spot
- Delete a Spot
- Show Spots on Map

Tasks

- Show All Tasks
- Add New Task
- Update a Task
- Search a Task
- Delete a Task

Help Maps

- I'm looking for Coordinates
- I'm looking for an Address
- Jump to Google Maps

Please complete the search form before you delete the task in order to note the **Task's ID**

In order to delete a Task you need to complete the form below.

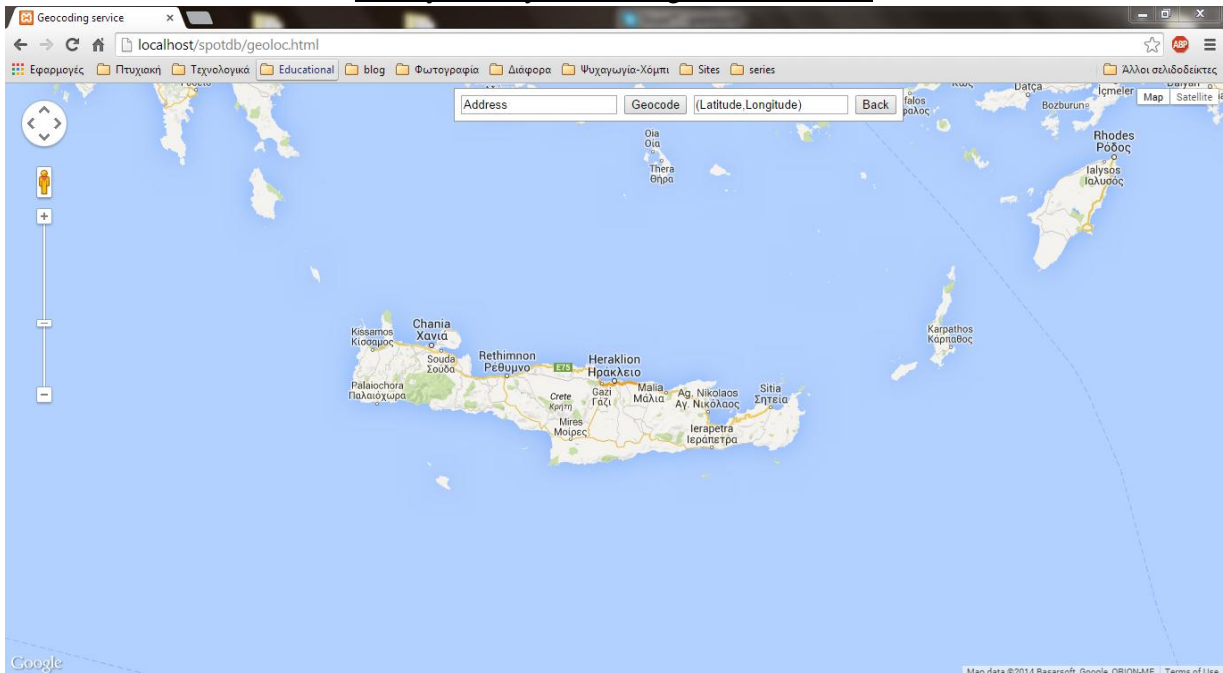
Insert the Task's ID:

Εικόνα 4.32: Σελίδα διαγραφής Task

Διαγραφή μίας εργασίας συμπληρώνοντας μόνο το ID της.

Help Maps

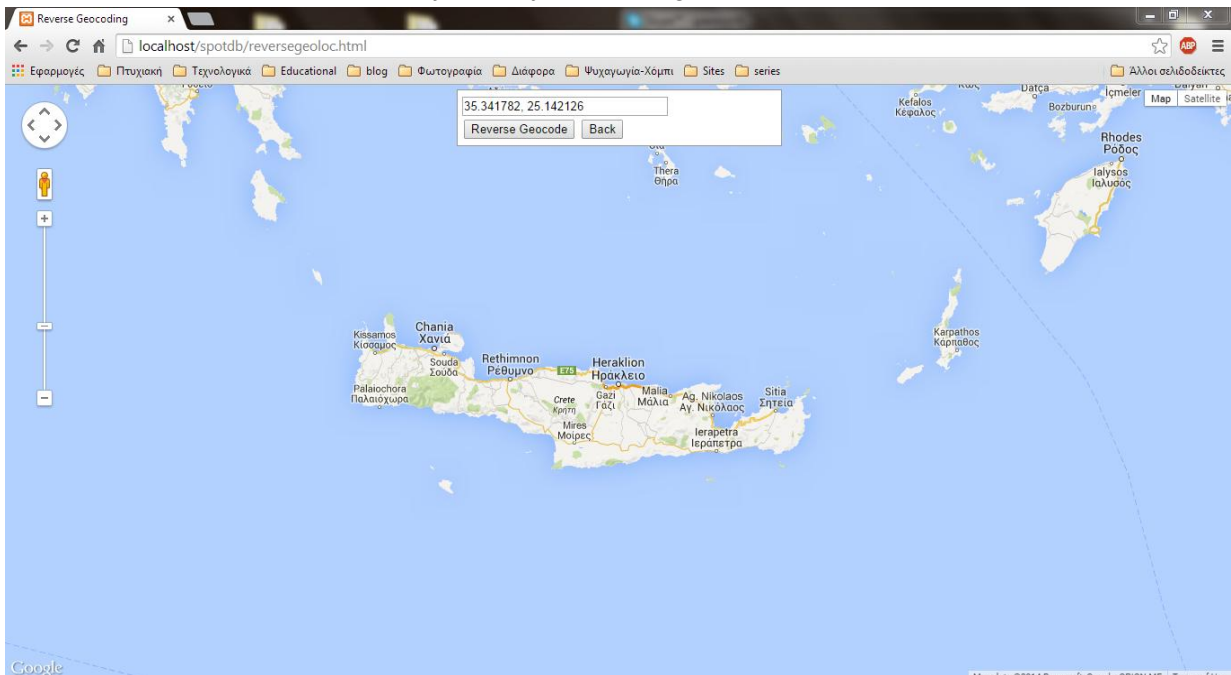
Τίτλος σελίδας: I'm looking for Coordinates



Εικόνα 4.33: Σελίδα αναζήτησης συντεταγμένων

Αν ο administrator δε γνωρίζει τις συντεταγμένες ενός σημείου στο χάρτη, τότε μπορεί να ανοίξει αυτή τη σελίδα σε μία νέα καρτέλα του browser και να συμπληρώσει το πεδίο “Address”. Όταν πατήσει το κουμπί “Geocode”, στο πεδίο δεξιά από αυτό θα εμφανιστούν οι συντεταγμένες, ενώ παράλληλα στο χάρτη θα εμφανιστεί το σημείο του οποίου οι συντεταγμένες επιστράφηκαν.

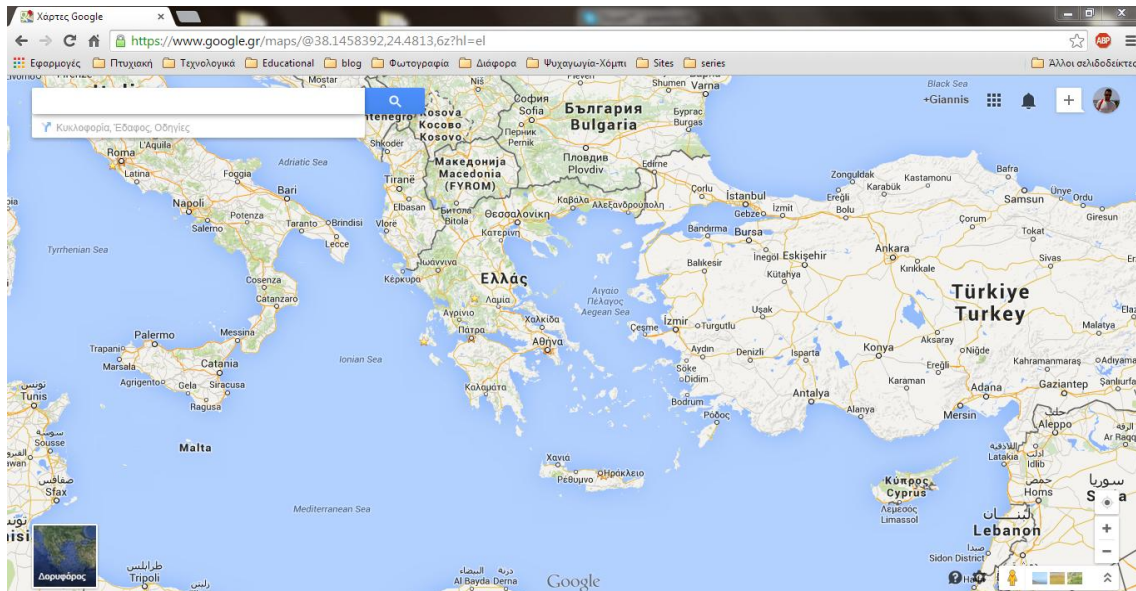
Τίτλος σελίδας: I'm looking for an Address



Εικόνα 4.34: Σελίδα αναζήτησης διεύθυνσης

Αυτή τη σελίδα επισκέπτεται ο διαχειριστής, αν γνωρίζει τις συντεταγμένες ενός σημείου και θέλει να βρει ποια είναι η διεύθυνσή του και το σημείο του στο χάρτη. Τις συντεταγμένες τις πληκτρολογεί στο πεδίο που φαίνεται στο πάνω μέρος της οθόνης, με το τρόπο που επίσης φαίνεται στο παραπάνω στιγμιότυπο.

Τίτλος σελίδας: Jump to Google Maps

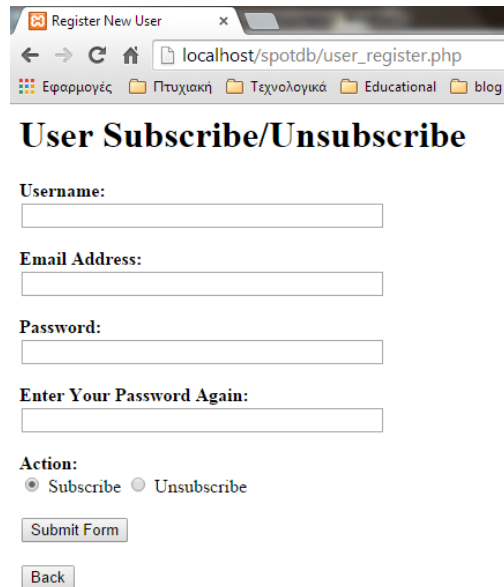


Εικόνα 4.35: Σελίδα Google Maps

Εδώ βλέπουμε την εφαρμογή Google Maps, μέσω της οποίας ο διαχειριστής μπορεί να βρει τη διεύθυνση, τις συντεταγμένες αλλά και το σημείο στο χάρτη ενός μέρους για το οποίο γνωρίζει μόνο σε ποιο σημείο βρίσκεται στο χάρτη και καμία άλλη πληροφορία. Πατώντας παρατεταμένα το αριστερό κλικ πάνω στη τοποθεσία αυτή, εμφανίζεται η διεύθυνση, οι συντεταγμένες και άλλες πληροφορίες της τοποθεσίας, όπου μπορεί να τις χρησιμοποιήσει ο διαχειριστής. Προτείνεται τόσο αυτός, όσο και οι παραπάνω χάρτες, να ανοίγονται σε νέο παράθυρο ή νέα καρτέλα για παράλληλη χρήση με το Administration.

Κουμπιά

Τίτλος σελίδας: Register New User



Register New User

localhost/spotdb/user_register.php

Εφαρμογές Πτυχιακή Τεχνολογικά Educational blog

User Subscribe/Unsubscribe

Username:

Email Address:

Password:

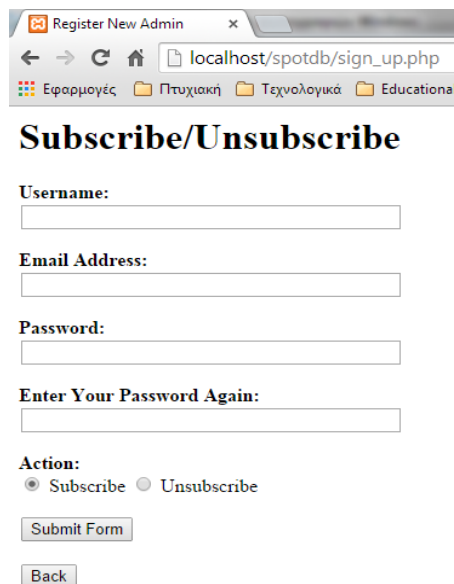
Enter Your Password Again:

Action:
 Subscribe Unsubscribe

Εικόνα 4.36: Σελίδα εγγραφής & απεγγραφής Χρήστη

Σελίδα εγγραφής ή διαγραφής ενός απλού χρήστη στη βάση δεδομένων και το πίνακα users. Γίνεται έλεγχος αν υπάρχει ήδη ο χρήστης αυτός στη βάση δεδομένων, βάσει του ονόματος χρήστη αλλά και της διεύθυνσης Email. Ο κωδικός πρόσβασης πρέπει να είναι μεταξύ 6 και 16 ψηφία.

Τίτλος σελίδας: Register New Administrator



Register New Admin

localhost/spotdb/sign_up.php

Εφαρμογές Πτυχιακή Τεχνολογικά Educational

Subscribe/Unsubscribe

Username:

Email Address:

Password:

Enter Your Password Again:

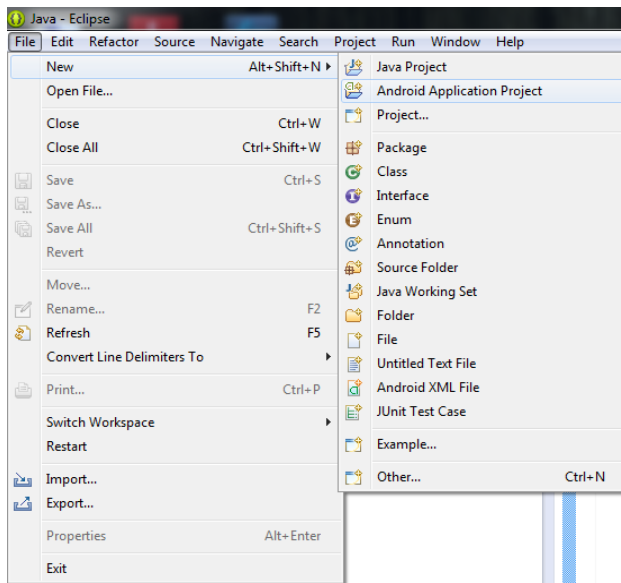
Action:
 Subscribe Unsubscribe

Εικόνα 4.37: Σελίδα εγγραφής & απεγγραφής Διαχειριστή

Σελίδα εγγραφής ή διαγραφής ενός διαχειριστή στη βάση δεδομένων και το πίνακα admins. Γίνεται έλεγχος αν υπάρχει ήδη διαχειριστής αυτός στη βάση δεδομένων, βάσει του ονόματος χρήστη αλλά και της διεύθυνσης Email. Ο κωδικός πρόσβασης πρέπει να είναι μεταξύ 6 και 16 ψηφία.

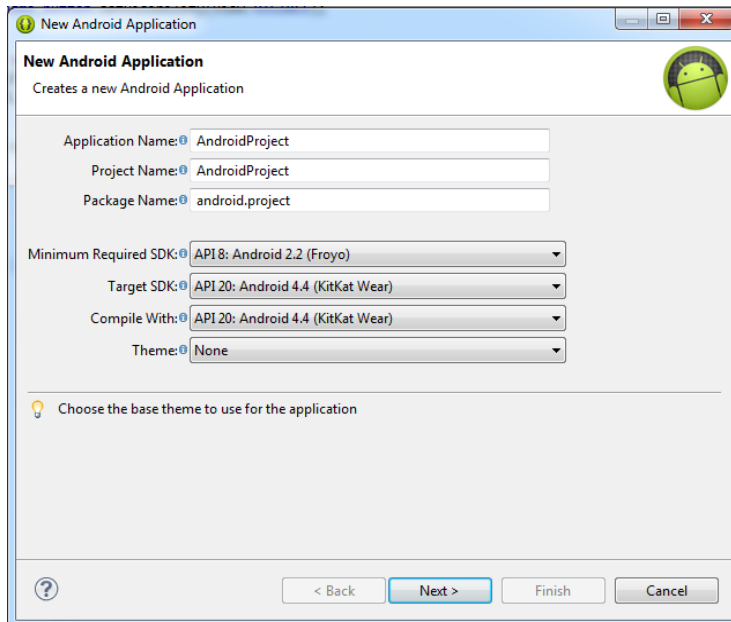
4.4 Εφαρμογή Android

4.4.1 Δημιουργία ενός Android Project



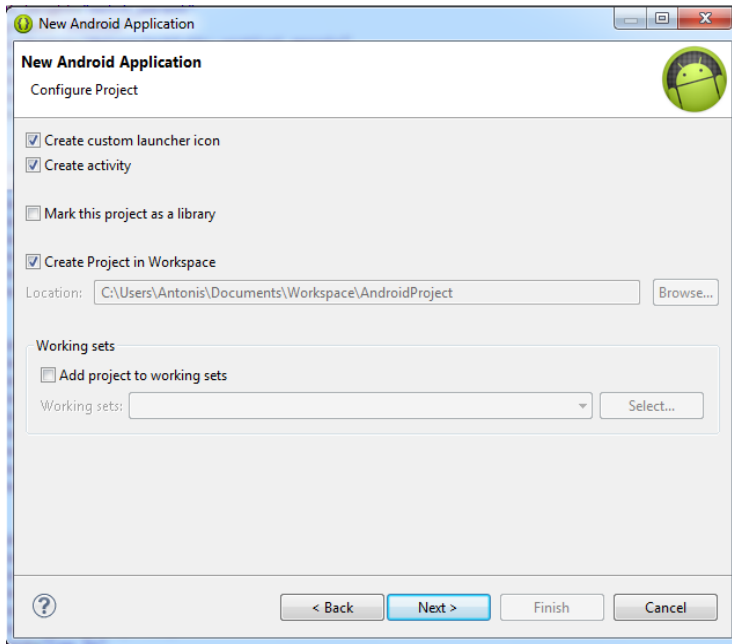
Αρχικά ανοίγουμε το Eclipse στο οποίο γίνεται η ανάπτυξη της εφαρμογής, και μετά δημιουργούμε ένα android project όπως φαίνεται στο σχήμα.

Εικόνα 4.38: Δημιουργία Android Project



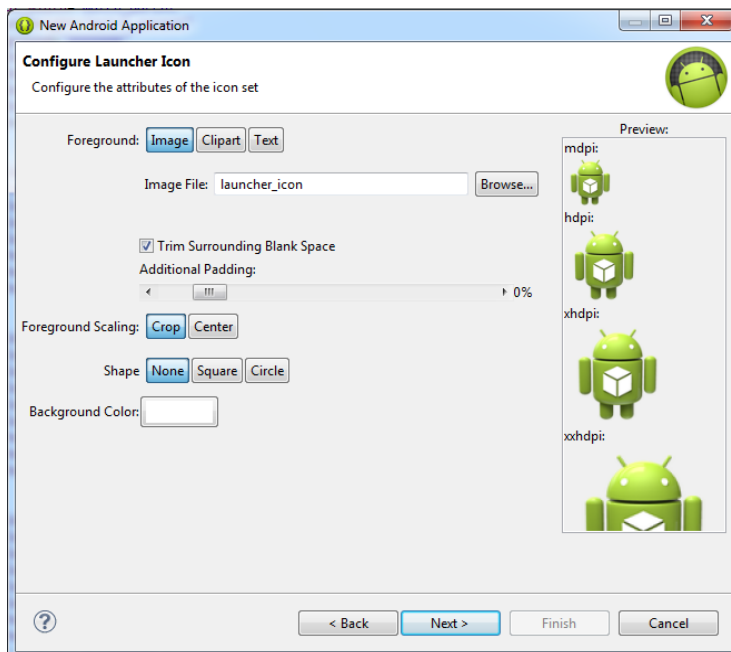
Στην συνέχεια ορίζουμε το όνομα του project, το package name, την ελάχιστη και μέγιστη έκδοση API για να είμαστε σίγουροι πως η εφαρμογή μας λειτουργεί σε μεγάλο εύρος κινητών συσκευών, και επίσης το API με το οποίο θέλουμε να κάνουμε compile το application μας.

Εικόνα 4.39: Εισαγωγή στοιχείων Android Project



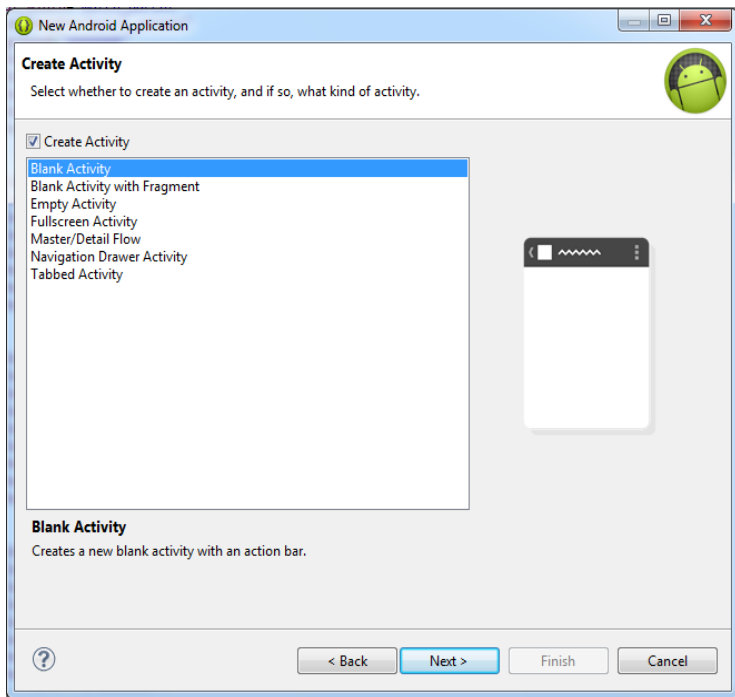
Τέλος "τσεκάρουμε" τα κουτάκια με τη δημιουργία του project μας στο workspace και του activity, όπως επίσης και το είδος του activity που επιθυμούμε όπως φαίνεται στα παρακάτω σχήματα.

Εικόνα 4.40: Επιλογές Android Project



Καθορισμός μεγέθους και διάταξης του Εικονιδίου Εκκίνησης

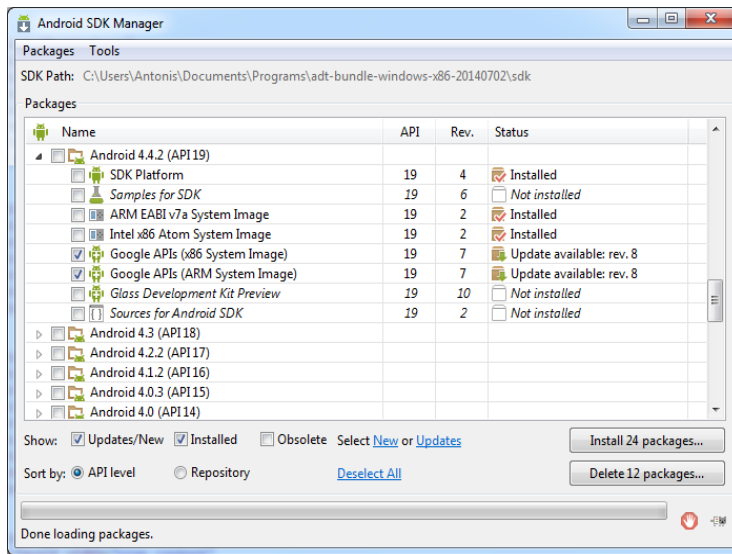
Εικόνα 4.41: Ρύθμιση εικονιδίου Android Εφαρμογής



Καθορισμός Τύπου του Activity

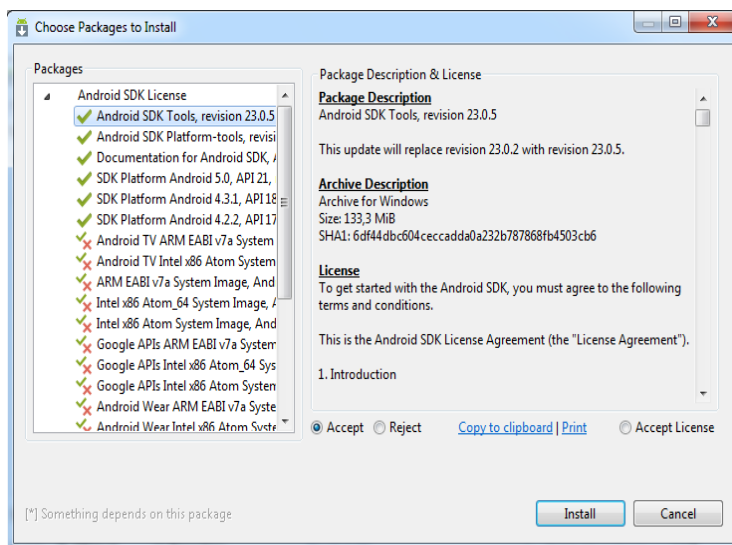
Εικόνα 4.42: Επιλογή τύπου του Activity

4.4.2 SDK Manager



Μόλις δημιουργήσουμε επιτυχώς το project μας, πρέπει να ελέγξουμε από το SDK Manager του Eclipse μας, εάν διαθέτουμε τις πιο πρόσφατες εκδόσεις API όπως επίσης και όλα τα κατάλληλα εργαλεία που μας παρέχει το ADT π.χ Android SDK Platform tools & Build tools, SDK Platform, Android Support Repository & Library και τέλος Google Repository, Google Play Services.

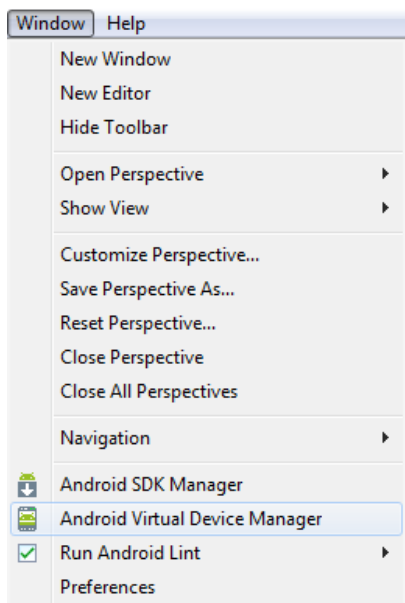
Εικόνα 4.43: Εκκίνηση SDK Manager



Στη συνέχεια αποδεχόμαστε τους όρους χρήσης εγκαθιστούμε τις ενημερώσεις και είμαστε έτοιμοι να ξεκινήσουμε την υλοποίηση.

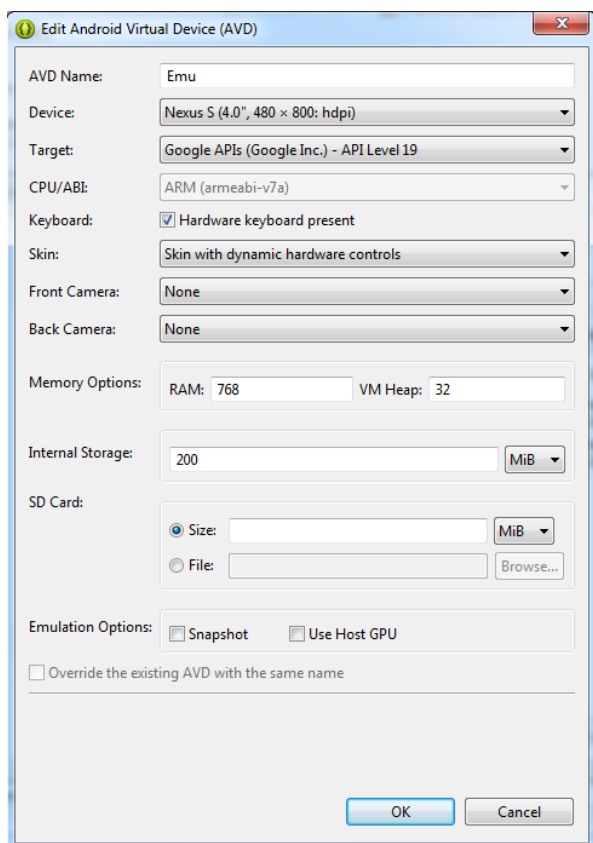
Εικόνα 4.44: Αποδοχή όρων στο SDK Manager

4.4.3 Δημιουργία και τρέξιμο ενός Emulator

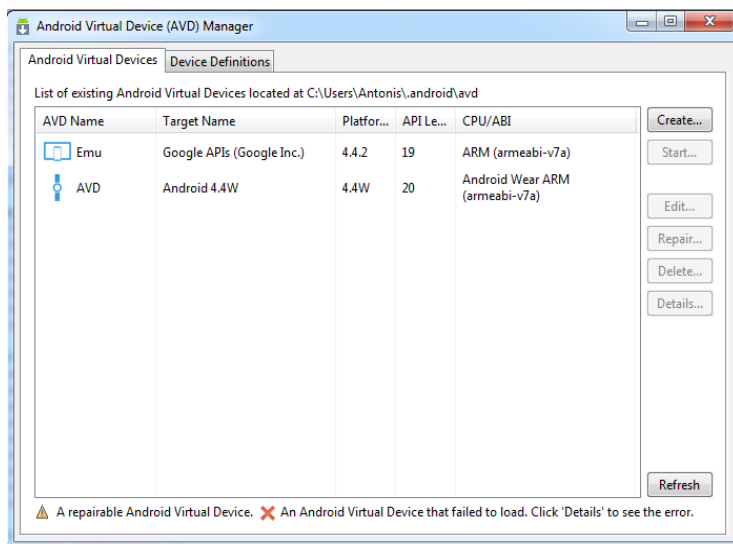


Για να φτιάξουμε τον δικό μας emulator με τον οποίο θα δοκιμάσουμε το application, πρέπει πρώτα από όλα να πάμε στην καρτέλα windows του eclipse και αφού ανοίξουμε το Android Virtual Device Manager, δημιουργούμε έναν εξομοιωτή με τα δικά μας τεχνικά χαρακτηριστικά όπως φαίνεται παρακάτω.

Εικόνα 4.45: Εκκίνηση Android Virtual Device Manager

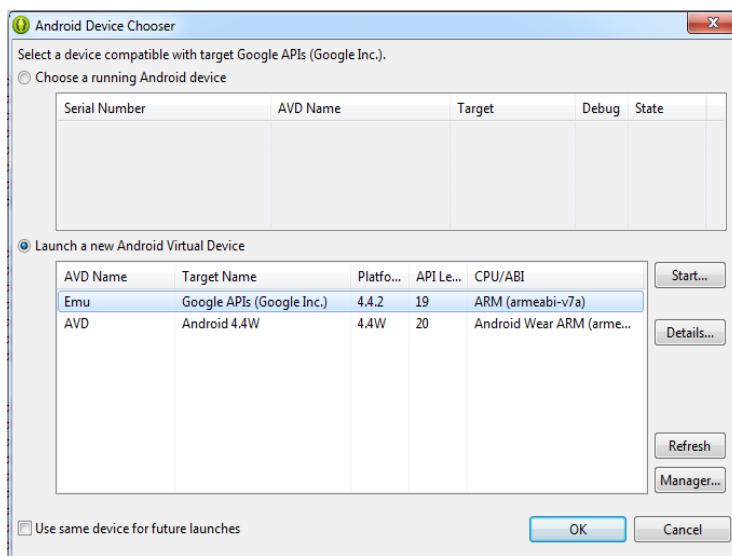


Εικόνα 4.46: Ρύθμιση νέου Virtual Device



Στο τέλος και αφού φτιάξουμε τον εξομοιωτή μας, μπορούμε να δούμε ποιά άλλα emulators έχουμε διαθέσιμα, να διαγράψουμε όποιο δεν χρειαζόμαστε και ακόμη να αλλάξουμε τα χαρακτηριστικά τους με την επιλογή Edit.

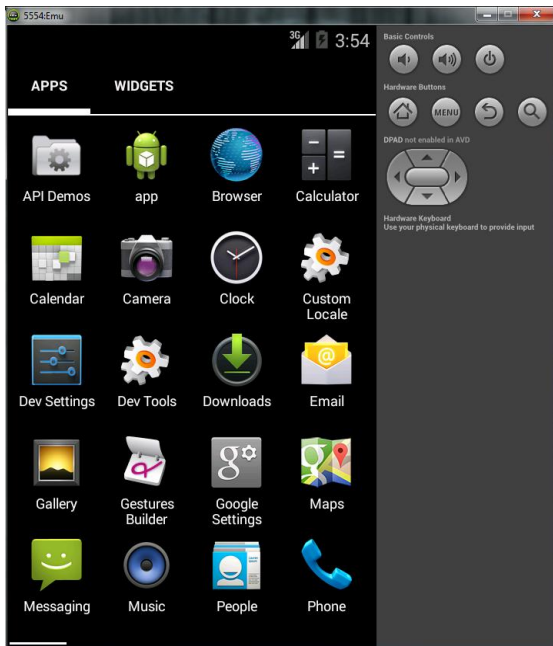
Εικόνα 4.47: Προβολή διαθέσιμων Virtual Devices



Εφόσον έχει δημιουργηθεί επιτυχώς ο emulator, μπορούμε να δοκιμάσουμε οποιοδήποτε project θέλουμε με το να ανατρέξουμε πολύ απλά στην επιλογή Run As → Android Application και στη συνέχεια να επιλέξουμε ένα emulator ή και μια κανονική συσκευή για τη δοκιμή μας όπως παρατηρούμε στο επόμενο σχήμα.

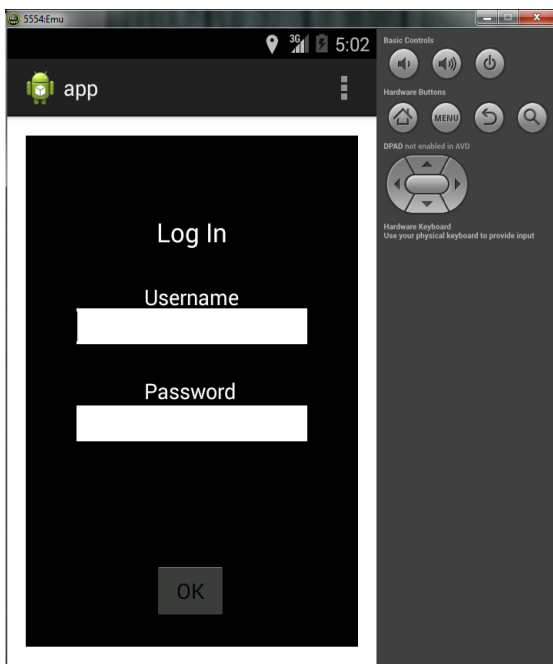
Εικόνα 4.48: Εκκίνηση Virtual Device

4.4.4 Στάδια λειτουργίας της Android Εφαρμογής



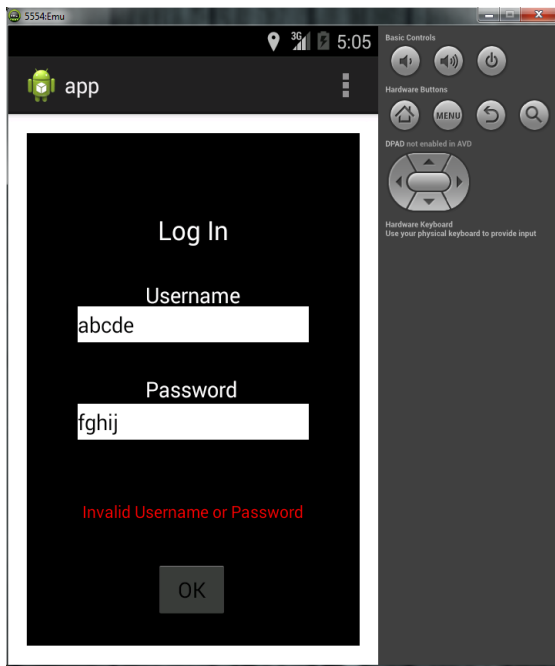
Αρχικά εντοπίζουμε το application μας με το όνομα app στην κινητή συσκευή ή στο emulator.

Εικόνα 4.49: Εντοπισμός του Application



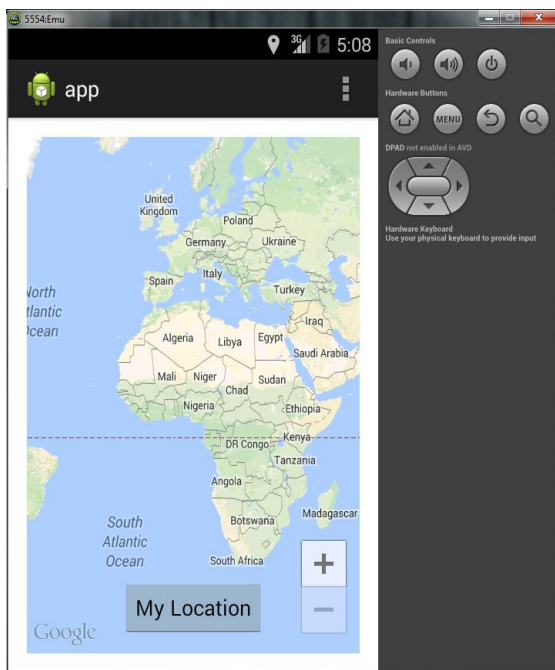
Εν συνεχεία και μόλις η εφαρμογή μας ανοίξει, εμφανίζεται στον χρήστη ένα log-in window στο οποίο αυτός πρέπει να συμπληρώσει επιτυχώς τα στοιχεία του για να του επιτραπεί η είσοδος.

Εικόνα 4.50: Log-in Window της Εφαρμογής



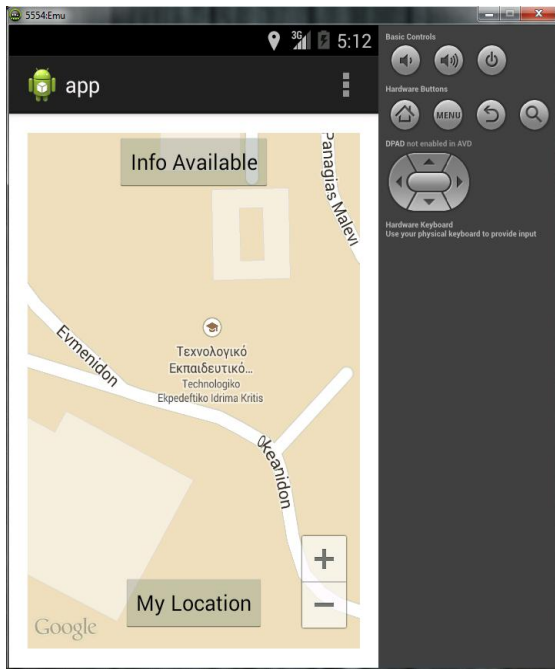
Εάν αυτός εισάγει λανθασμένα στοιχεία τότε εμφανίζεται και το αντίστοιχο κείμενο που του εξηγεί πως η είσοδος των στοιχείων του είναι λάθος.

Εικόνα 4.51: Εισαγωγή λανθασμένων στοιχείων



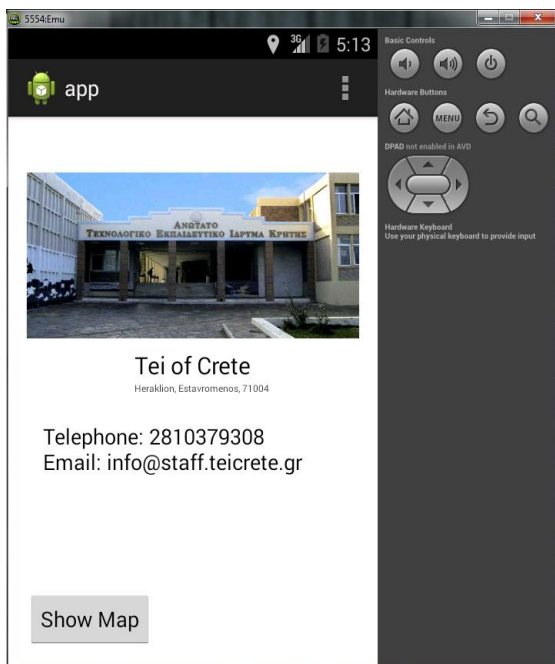
Όταν εισάγει σωστά τα username και password τότε εμφανίζεται στην οθόνη μας ένας χάρτης (Google Maps) όπως επίσης και ένα button το My Location.

Εικόνα 4.52: Εμφάνιση του κεντρικού Χάρτη



Πατώντας το συγκεκριμένο button, η εφαρμογή χρησιμοποιώντας το GPS της κινητής συσκευής εντοπίζει την τοποθεσία του χρήστη και κάνει αυτόματα zoom στο συγκεκριμένο location εμφανίζοντας παράλληλα ονόματα οδών, διάφορα σημεία της περιοχής κλπ.

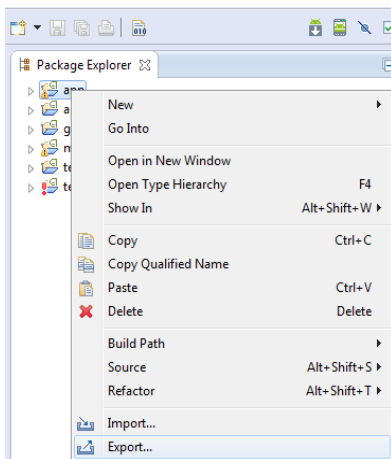
Εικόνα 4.53: Εντοπισμός θέσης



Πιο συγκεκριμένα, εφόσον ο χρήστης βρίσκεται εντός της περιμέτρου (γεωγραφικό πλέγμα) μιας περιοχής που έχουμε εισάγει εμείς στην βάση δεδομένων μας, τότε εμφανίζεται αυτόματα το button Info Available το οποίο όταν πατηθεί θα δώσει χρήσιμες πληροφορίες στον χρήστη για το συγκεκριμένο μέρος όπως το όνομα, οδός, τηλέφωνο επικοινωνίας, email, FAX κλπ.

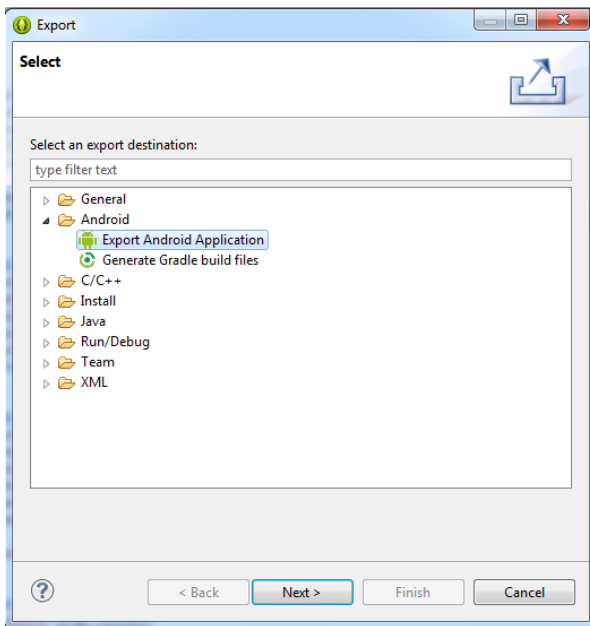
Εικόνα 4.54: Εμφάνιση πληροφοριών τοποθεσίας

4.4.5 Export μιας Android Εφαρμογής



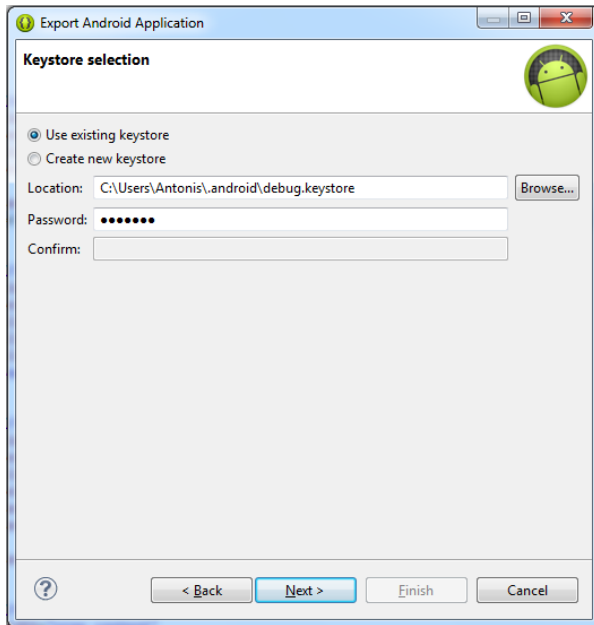
Εφόσον έχουμε δοκιμάσει και ελέγξει ότι η εφαρμογή μας δουλεύει σωστά, το μόνο που μένει να κάνουμε είναι να την περάσουμε στην κινητή συσκευή μας με τον εξής τρόπο. Αρχικά, πηγαίνουμε στο project μας και επιλέγουμε την επιλογή export.

Εικόνα 4.55: Εξαγωγή Android Εφαρμογής



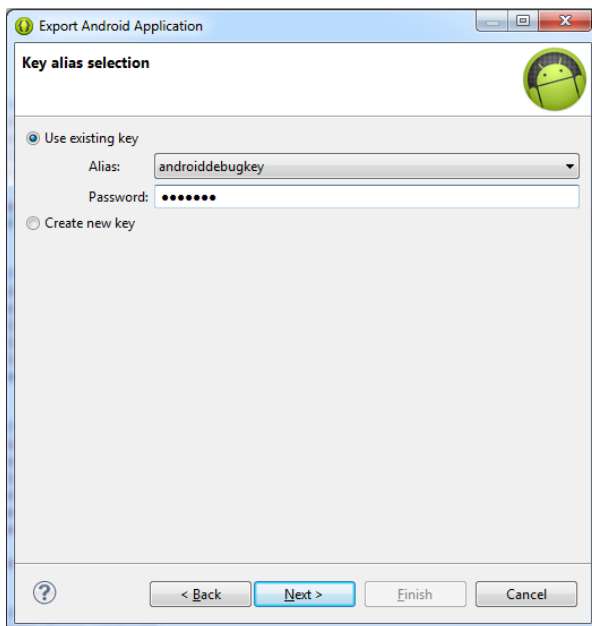
Στη συνέχεια επιλέγουμε στον φάκελο Android το Export Android Application.

Εικόνα 4.56: Επιλογές για εξαγωγή Android Εφαρμογής



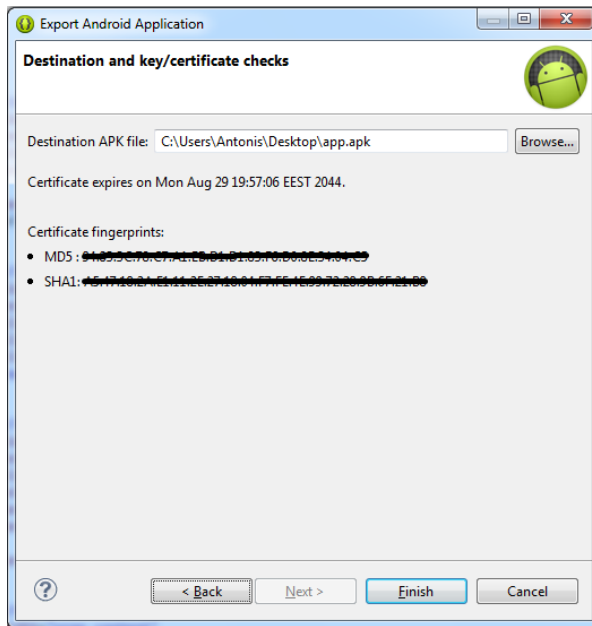
Μετά αφού τσεκάρουμε την επιλογή Use existing keystore, εντοπίζουμε το αρχείο debug.keystore (βρίσκεται στον .android φάκελο) στον υπολογιστή μας, εισάγουμε τον κωδικό 'android' και πατάμε επόμενο.

Εικόνα 4.57: Εντοπισμός του Debug Keystore



Στην επόμενη καρτέλα που εμφανίζεται τσεκάρουμε την επιλογή Use existing key και βάζουμε ξανά για κωδικό την λέξη 'android'.

Εικόνα 4.58: Εισαγωγή κωδικού του Debug Keystore



Τέλος ελέγχουμε τα προσωπικά μας στοιχεία (SHA1, MD5) και αφού επιλέξουμε την τοποθεσία που θα δημιουργηθεί το .apk αρχείο, πατάμε finish και περιμένουμε να τελειώσει η διαδικασία.

Εικόνα 4.59: Επιλογή φακέλου εξαγωγής APK

5. Αποτελέσματα

5.1 Συμπεράσματα

Αρχικά, ψάξαμε ποιό λογισμικό θα χρησιμοποιήσουμε για κάθε τμήμα της πτυχιακής μας ξεχωριστά και καταλήξαμε σε εργαλεία και εφαρμογές ανοιχτού κώδικα που συνεργάζονται άριστα μεταξύ τους. Για να εμπλουτίσουμε τις γνώσεις και τις δεξιότητες μας στη χρήση των παραπάνω εργαλείων, ανατρέξαμε σε πηγές όπως τα εγχειρίδια χρήσης τους όπως επίσης και άλλες χρήσιμες πηγές από το internet.

Πιο συγκεκριμένα, όσον αφορά τη βάση δεδομένων μας μετά από δοκιμές που πραγματοποιήσαμε σε συσχετίσεις και σε τύπους πεδίων, καταλήξαμε στο ιδανικό σχήμα για τις περιστάσεις της εφαρμογής μας. Δώσαμε επίσης έμφαση στην ασφάλεια των δεδομένων και στις σωστές παραμέτρους των πεδίων όπως επίσης και στο σωστό καθορισμό των πρωτευόντων και ξένων κλειδιών των πινάκων.

Όσον αφορά το προγραμματιστικό μέρος, καταλήξαμε σε ένα πακέτο έκδοσης του eclipse το οποίο περιείχε επιπλέον τα ADT - SDK και όλα τα χρήσιμα εργαλεία για την ανάπτυξη της εφαρμογής μας. Για την ανάπτυξη του κώδικα σε PHP, έγιναν δοκιμές στη χρήση διάφορων συναρτήσεων για να διασφαλίσουμε τα βέλτιστα αποτελέσματα.

Αυτή η πτυχιακή εργασία, αφορά απλούς χρήστες οι οποίοι θέλουν να χρησιμοποιούν μία εφαρμογή στο smartphone τους και να εντοπίζουν σημεία ενδιαφέροντος κοντά στη τοποθεσία που βρίσκονται. Μπορεί όμως με τις κατάλληλες αλλαγές, να αφορά και μία εταιρία η οποία παρέχει τεχνική υποστήριξη σε σπίτια και άλλες εταιρίες και θέλει το προσωπικό της, να χρησιμοποιεί smartphones και να εντοπίζει τις βλάβες που βρίσκονται κοντά στο σημείο που βρίσκονται.

5.2 Μελλοντική Δουλειά

Μερικά από τα μελλοντικά μας σχέδια για το Android Application είναι τα εξής:

- Να προσθέσουμε περισσότερα σημεία ενδιαφέροντος στη βάση δεδομένων μας.
- Να εμπλουτίσουμε τις πληροφορίες στις οποίες έχει ήδη πρόσβαση ο χρήστης με επιπλέον δεδομένα, όπως για παράδειγμα σχόλια και αξιολογήσεις χρηστών για μια συγκεκριμένη περιοχή.
- Περισσότερες λειτουργίες όπως πυξίδα και αναζήτηση τοποθεσιών πάνω στο χάρτη.
- Ομορφότερο και πιο εύχρηστο γραφικό περιβάλλον χρήστη (GUI).
- Προσθήκη ενός help button, σχετικά με βοήθεια και οδηγίες χρήσης της εφαρμογής.

Είναι σημαντικό να τονίσουμε όμως, ότι μετά τη προσθήκη των παραπάνω, πρέπει να μεριμνήσουμε ώστε η εφαρμογή μας να παραμένει εξίσου γρήγορη, λειτουργική και φιλική προς το χρήστη (User Friendly).

6. Βιβλιογραφία

Διαδίκτυο

1. <http://www.wikipedia.org>
2. <http://thenewboston.org>
3. <https://buckysroom.org>
4. <http://www.tutorialspoint.com>
5. <http://stackoverflow.com>
6. <http://www.w3schools.com>
7. <http://www.appcelerator.com/developers>
8. <http://dev.mysql.com>
9. <http://developer.android.com>
10. <http://www.androidhive.info>
11. <http://developer.android.com/tools/sdk/eclipse-adt.html>
12. <http://www.vogella.com>
13. <https://www.coursera.org>
14. <http://www.youtube.com> "android lessons for beginners"
15. <https://www.eclipse.org/>
16. <https://www.apachefriends.org>
17. <http://whatis.techtarget.com/>

Βιβλία

1. Ανάπτυξη web εφαρμογών με PHP και MySQL (4^η έκδοση) - Luke Welling, Laura Thomson
2. Μάθετε PHP, MySQL και Apache σε 24 ώρες - Julie C. Meloni
3. Οδηγός της Javascript (2^η έκδοση) – John Pollock
4. Εισαγωγή στα συστήματα Βάσεων Δεδομένων (Τόμος Α', 6^η έκδοση) – C.J.Date

Παράρτημα Α Σημαντικές γραμμές κώδικα

Administration

Αρχείο index.html

```
<!doctype html><html><head><meta charset="utf-8"><title>Log in to SPOT-DB Administration</title></head>

<body><center>

<h1> Please login </h1>

<form action="login.php" method="post">

<table border="0">

<tr>

<td>Username:</td>

<td><input type="text" name="user"></td>

</tr>

<tr>

<td>Password:</td>

<td><input type="password" name="pass"></td>

</tr>

<tr>

<td colspan="2" align="center"><input type="submit" value="Log In"></td>

</tr>

<tr>

<td colspan="2" align="center"><a href="manage.php"><input name="Register" type="button" id="Register" value="Sign Up"></a>

</td>

</tr>

</table>

</form>

</center></body></html>
```

Αρχείο login.php

```
<!doctype html><html><head><meta charset="utf-8"><title>Administation</title></head>

<body><center>

<h1> <center>Log-in </center></h1>

<?php

$Username=$_POST[user];

$Password=$_POST[pass];

//Συνάρτηση σύνδεσης με τη βάση δεδομένων

function doDB()

{

global $conn;

//σύνδεση στο server και επιλογή της βάσης δεδομένων. Θα χρειαστούμε το παρακάτω

$conn=mysql_connect("localhost","giannis","1234","test2") or die(mysql_error());

mysql_select_db("test2",$conn) or die(mysql_error());

}

function usernameChecker($Username,$Password)

{

//έλεγχος αν το username του χρήστη δεν υπάρχει ήδη στη βάση

global $conn,$check_un_result;

$check="select id from admins where Username='$Username' and Password='$Password'";

$check_un_result=mysql_query($check,$conn) or die(mysql_error());

}

//Έλεγχος αν συμπληρώθηκαν και τα δύο πεδία

if(($_POST[user]=="")||($_POST[pass]==""))

{

echo "You have not entered all the required details.<br />". "Please go back and try again.";

exit;

}

//σύνδεση στη βάση δεδομένων
```

```

doDB();

//έλεγχος οτι το username περιέχεται στη λίστα
usernameChecker($_POST[user],$_POST[pass]);

if(mysql_num_rows($check_un_result)<1)

{

//εμφάνιση μηνύματος αποτυχίας

echo "<p> Your Username and/or Password You've entered is wrong! Please go back and try again.</p>";

}

else

{

header('Location: home.html');

}

?>

</center></body></html>

```

Αρχείο manage.php

```

<?php

//Εγγραφή νέου χρήστη σελίδα 363 php σε 24 ώρες

//ορισμός συναρτήσεων

function doDB()

{

global $conn;

//σύνδεση στο server και επιλογή της βάσης δεδομένων. Θα χρειαστούμε το παρακάτω

$conn=mysql_connect("localhost","giannis","1234","test2") or die(mysql_error());

mysql_select_db("test2",$conn) or die(mysql_error());

}

function emailChecker($Email)

{

//έλεγχος αν η διεύθυνση email του χρήστη δεν υπάρχει ήδη στη βάση

global $conn,$check_result;

```

```

$check="select id from admins where Email='$Email'";

$check_result=mysql_query($check,$conn) or die(mysql_error());

}

function usernameChecker($Username)

{

//έλεγχος αν το username του χρήστη δεν υπάρχει ήδη στη βάση

global $conn,$check_un_result;

$check="select id from admins where Username='$Username'";

$check_un_result=mysql_query($check,$conn) or die(mysql_error());

}

//η ακόλουθη if εξακριβώνει αν ο χρήστης θέλει να δει τη φόρμα ή όχι

if($_POST[op] != "ds")

{

//αν ναι δημιουργία της ενότητας για την εμφάνιση της φόρμας

$display_block="<form method=POST action='\".$_SERVER[PHP_SELF]\"'>

<p><strong> Username: </strong><br>

<input type=text name=\"Username\" size=40 maxlength=150>

<p><strong> Email Address: </strong><br>

<input type=email name=\"Email\" size=40 maxlength=150>

<p><strong> Password: </strong><br>

<input type=password name=\"Password\" size=40 maxlength=150>

<p><strong> Enter Your Password Again: </strong><br>

<input type=password name=\"Password2\" size=40 maxlength=150>

<p><strong> Type of User: </strong><br>

<input type=radio name=\"type\" value=\"admin\" checked> Administrator

<input type=radio name=\"type\" value=\"user\"> User

<p><strong> Action: </strong><br>

<input type=radio name=\"action\" value=\"sub\" checked> Subscribe

<input type=radio name=\"action\" value=\"unsub\"> Unsubscribe

```



```

<input type="hidden" name="op" value="ds">

<p><input type="submit" name="submit" value="Submit Form"></p>

<p><a href="index.html" /><input type="button" name="back" value="Back"></a></p>

</form>;

}

else if($_POST[op]=="ds" && $_POST[action]=="sub")

{

//προσπάθεια εγγραφής νέου συνδρομητή στη λίστα και επικύρωση του username, email και κωδικού

if($_POST[Username]=="")||($_POST[Email]=="")||($_POST[Password]=="")||($_POST[Password2]=="")

{

echo "You have not entered all the required details.<br />". "Please go back and try again.";

exit;

}

//έλεγχος για το αν είναι οι ίδιοι οι passwords και αν είναι μεταξύ 6 και 16 ψηφία.

else if((strlen($_POST[Password]) < 6) || (strlen($_POST[Password]) > 16))

{

echo "Your password must be between 6 and 16 characters. Please go back and try again.";

exit;

}

else if($_POST[Password] != $_POST[Password2])

{

echo "Your passwords you've entered doesn't match. Please go back and try again.";

exit;

}

//έλεγχος για το αν είναι οι ίδιοι οι passwords και αν είναι μεταξύ 6 και 16 ψηφία.

//σύνδεση στη βάση δεδομένων

doDB();

//έλεγχος ότι το username περιέχεται στη λίστα

usernameChecker($_POST[Username]);

```

```

//έλεγχος ότι η διεύθυνση email περιέχεται στη λίστα

emailChecker($_POST[Email]);

//εύρεση του αριθμού των αποτελεσμάτων εκτέλεσης ενέργειας

if ((mysql_num_rows($check_result)<1) && (mysql_num_rows($check_un_result)<1))

{

//προσθήκη εγγραφής

//$sql="insert into admins values(',$_POST[Username]','$_POST[Password]','$_POST[Email]')";

//$result = mysql_query($sql,$conn) or die(mysql_error());

//$display_block = "<p> Thanks for signing up.</p>";

//αποστολή email

$name=$_POST[Username];

$email=$_POST[Email];

$password=$_POST[Password];

if ($_POST[type]=="admin")

{

$new="Administrator";

}

else if($_POST[type]=="user")

{

$new="User";

}

$to="giannisk88@gmail.com";

$subject="New registration";

$mailcontent= "Customer name: ".$name."\n".

"Customer email: ".$email."\n".

"Customer password: ".$password."\n".

"Customer type: \n".$new."\n";

//κλήση συνάρτησης mail() για αποστολή αλληλογραφίας

mail($to,$subject,$mailcontent);

```

```

$display_block = "<p> Your registration form was successfully sent for approval! Please check your email shortly.</p>";

}

else

{

//εμφάνιση μηνύματος αποτυχίας

$display_block="<p>The Username and/or Email you entered already exists! Please go back and try again.</p>";

}

}

else if($_POST[op]=="ds"&&$_POST[action]=="unsub")

{

//προσπάθεια διαγραφής από τη λίστα και επικύρωση της διεύθυνσης email

if($_POST[Username]=="")&&($_POST[Email]=="")&&($_POST>Password=="")&&($_POST>Password2=="")

{

header("Location: manage.php");

exit;

}

//σύνδεση στη βάση δεδομένων

doDB();

//έλεγχος ότι η διεύθυνση email και το username περιέχονται στη λίστα

emailChecker($_POST[Email]);

usernameChecker($_POST[Username]);

//εύρεση του αριθμού των αποτελεσμάτων και εκτέλεση ενέργειας

if ((mysql_num_rows($check_result)<1) && (mysql_num_rows($check_un_result)<1))

{

//εμφάνιση failure message

$display_block="<p>Couldn't find your username and/or email address!</p> <p>No action was taken.</p>";

}

else

{

```

```

//διαγραφή διεύθυνσης email και username από τη βάση

$хid=mysql_result($check_result,0,"id");

$хsql="delete from admins where id='$хid";

$хresult=mysql_query($хsql,$хconn) or die(mysql_error());

$хdisplay_block="<p> You're unsubscribed!</p>";

}

}

?>

<html><head><meta charset="utf-8"><title>Register New Admin/User</title></head>

<body><h1>Subscribe/Unsubscribe</h1>

<?php

echo "$хdisplay_block";

?>

</body></html>

```

Αρχείο home.html

```

<!doctype html><html><head><meta charset="utf-8"><title>Administration</title><link rel="stylesheet" type="text/css" href="style.css">

</head><body>

<div id="header" class="title"><strong><center>Welcome to the Administration</center></strong></div>

<div id="wraphome">

<div class="itemTitle">

Please select an option to continue:

</div>

<ul>

<b> Spots </b>

<li><a href="showallspots.php">Show All Spots</a></li>

<li><a href="insertnewspot.php">Add New Spot</a></li>

<li><a href="updatespot.php">Update a Spot</a></li>

<li><a href="showspot.php">Search a Spot</a></li>

<li><a href="deletespot.php">Delete a Spot</a></li>

```

```

<li><a href="map.html">Show Spots on Map</a></li>

<p>&nbsp;</p>

<b>Tasks </b>

<li><a href="showalltasks.php">Show All Tasks</a></li>

<li><a href="insertnewtask.php">Add New Task</a></li>

<li><a href="updatetask.php">Update a Task</a></li>

<li><a href="showtask.php">Show Tasks</a></li>

<li><a href="deletetask.php">Delete a Task</a></li>

<p>&nbsp;</p>

<b>Help Maps </b>

<li><a href="geoloc.html">I'm looking for Coordinates</a></li>

<li><a href="reversegeoloc.html">I'm looking for an Address</a></li>

<li><a href="https://maps.google.com">Jump to Google Maps</a></li>

</ul>

</div>

<div id="wraphome">

<p><a href="user_register.php" /><input type=button name="User_register" value="Register New User"></a></p>

<p><a href="sign_up.php" /><input type=button name="Admin_register" value="Register New Administrator"></a></p>

<p><a href="index.html" /><input type=button name="back" value="Back to Log-in screen"></a></p>

</div></body></html>

```

Αρχείο showallspots.php

```

<!doctype html><html><head><meta charset="utf-8"><title>Spot's database</title><link rel="stylesheet" type="text/css" href="style.css">

</head><body>

<div id="header"><center>

<h1>Show all the spots of the database</h1></center></div>

<div id="wrap">

<div id="left">

<?php require_once('sidebar.html');?>

```

```

</div>

<div id="content">

<div class="title">

<p align="center">Results: </p>

</div>

<div class="item" align="center">

<?php

$conn=mysql_connect("localhost","giannis","1234","test2");

mysql_select_db("test2",$conn);

$query="SELECT * FROM `spot` WHERE 1";

$result=mysql_query($query,$conn) or die(mysql_error());

$num_results=mysql_num_rows($result);

echo "<p> Number of spots: ".$num_results."</p>";

for($i=0 ; $i < $num_results ; $i++)

{

$row=mysql_fetch_assoc($result);

echo "<p><strong>".($i+1).".Spot's ID: </strong>";

echo stripslashes($row['SpotID']);

echo "<br /><strong>Spot's Name: </strong>";

echo htmlspecialchars(stripslashes($row['SpotName']));

echo "<br /><strong>Latitude: </strong>";

echo stripslashes($row['LAT']);

echo "<br /><strong>Longitude: </strong>";

echo stripslashes($row['LON']);

echo "<br /><strong>Address: </strong>";

echo htmlspecialchars(stripslashes($row['Address']));

echo "<br /><strong>Telephone: </strong>";

echo stripslashes($row['Telephone']);

echo "<br /><strong>Email Address: </strong>";

```

```

echo htmlspecialchars(stripslashes($row['Email']));

echo "</p>";

}

mysql_free_result($result);

mysql_close($conn);

?>

</div>

</div>

</div>

<div id="footer">

<p>&nbsp;</p>

</div></body></html>

```

Αργείο insertnewspot.php

```

<!doctype html><html><head><meta charset="utf-8"><title>Insert New Spot to Database</title><link rel="stylesheet" type="text/css"
href="style.css"></head><body>

<div id="header"><center><h1>Add new spot in database</h1></center></div>

<div id="wrap">

<div id="left">

<?php require_once('sidebar.html');?>

</div>

<div id="content">

<div class="title">

<p align="center">Please insert the data of the new spot. <br>

*(required fields)

</p>

</div>

<div class="item">

<form method=post action="insert_newspot.php">

<table border="0">

```

```
<tr>

<td>*Spot's name:</td>

<td><input type="text" name="Name" maxlength="50" size="20"></td>

</tr>

<tr>

<tr>

<td>*Spot's Latitude:</td>

<td><input type="text" name="lat" size="18"></td>

</tr>

<tr>

<td>*Spot's Longitude:</td>

<td><input type="text" name="lon" size="18"></td>

</tr>

<tr>

<td>*Address,City,State:</td>

<td><input type="text" name="Address" maxlength="50" size="20"></td>

</tr>

<tr>

<td>Telephone:</td>

<td><input type="tel" name="Tel" maxlength="15" size="15"></td>

</tr>

<tr>

<td>*Email:</td>

<td><input type="email" name="Email" maxlength="30" size="30"></td>

</tr>

<tr>

<td colspan="2" align="center">

<input type="submit" name="submit_button" value="Submit"/>

<input type="reset" name="reset" value="Reset"/>


```



```
</td>
```

```
</tr>
```

```
</table></form></div></div></div><div id="footer"><p>&nbsp;</p></div></body></html>
```

Αρχείο insert_newspot.php

```
<!--ΣΕ ΑΥΤΟ ΤΟ PHP FILE ΓΙΝΕΤΑΙ ΧΡΗΣΗ ΕΝΤΟΛΩΝ ΓΙΑ ΣΥΝΔΕΣΗ ΜΕ ΤΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ,  
ΤΟ ΠΕΡΑΣΜΑ ΣΤΟ ΠΙΝΑΚΑ ITEMS ΝΕΩΝ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΗΝ ΕΜΦΑΝΙΣΗ ΤΩΝ ΠΕΡΙΕΧΟΜΕΝΩΝ  
ΤΟΥ ΠΙΝΑΚΑ ΠΕΡΙΣΣΟΤΕΡΑ ΓΙΑ ΤΟ ΤΕΛΕΥΤΑΙΟ ΒΗΜΑ ΣΤΙΣ ΦΩΤΟΓΡΑΦΙΕΣ ΑΠΟ ΤΟ ΒΙΒΛΙΟ  
"ΑΝΑΠΤΥΞΗ WEB ΕΦΑΡΜΟΓΩΝ ΜΕ PHP ΚΑΙ ΣΕΛΙΔΕΣ 268,270 -->
```

```
<!doctype html><html><head><meta charset="utf-8"><title>Result</title><link rel="stylesheet" type="text/css" href="style.css">
```

```
</head><body>
```

```
<div id="header"><center><h1>Add new spot in database</h1></center></div>
```

```
<div id="wrap">
```

```
<div id="left">
```

```
<?php require_once('sidebar.html');?>
```

```
</div>
```

```
<div id="content">
```

```
<div class="title">
```

```
<p align="center">Results: </p>
```

```
</div>
```

```
<div class="item">
```

```
<?php
```

```
//Δημιουργία σύντομων ονομάτων μεταβλητών
```

```
$name=$_POST['Name'];
```

```
$lat=$_POST['lat'];
```

```
$lon=$_POST['lon'];
```

```
$address=$_POST['Address'];
```

```
$tel=$_POST['Tel'];
```

```
$email=$_POST['Email'];
```

```
////Έλεγχος συμπλήρωσης των υποχρεωτικών πεδίων της φόρμας.
```

```

if(!$name|| !$lat|| !$lon|| !$address|| !$email)

{

echo "You have not entered all the required details.<br />". "Please go back and try again.";

exit;

}

//Έλεγχος για το αν ο χρήστης στο πεδίο ονόματος πληκτρολόγησε το χαρακτήρα κενό. Αν το έκανε δε θα μπορεί να συνεχίσει στην
αποθήκευση του spot.

//πηγή: http://www.php.net/manual/en/function ctype-space.php<br>

//Σε περίπτωση που ο χρήστης πριν το όνομα βάλει πολλά κενά ή έστω ένα η trim κόβει τα κενά και μπαίνει στη βάση δεδομένων το όνομα
χωρίς κενά στην αρχή.

if(ctype_space($name))

{

echo "<br />The Spot's Name you've entered is not right because it contains only whitespaces. Please try again a different name.";

exit;

}

//Έλεγχος αν οι συμβολοσειρές σε κάθε πεδίο της φόρμας ακολουθούν τους κανόνες σύνταξης της βάσης δεδομένων.

//Για αυτό το λόγο γίνεται εισαγωγή κάθετων ανάμεσα στις λέξεις, στα κενά και στα σύμβολα που πληκτρολογούνται.

if(!get_magic_quotes_gpc())

{

$name=addslashes($name);

$lat=addslashes($lat);

$lon=addslashes($lon);

$address=addslashes($address);

$tel=addslashes($tel);

$email=addslashes($email);

}

//Σύνδεση στο σύστημα διαχείρισης βάσεων δεδομένων, με τα στοιχεία ενός από τους administrators και επιλογή της συγκεκριμένης βάσης
δεδομένων που θέλουμε.

$db=mysql_connect("localhost","giannis","1234","test2");

mysql_select_db("test2",$db);

```

```

//Έλεγχος της βάσης δεδομένων αν υπάρχει ήδη κάποιο spot με το ίδιο όνομα.

function spotChecker($newname,$lat,$lon)

{

global $db, $check_result;

//έλεγχος

$check="select SpotID from `spot` where `SpotName`='$newname' or `LAT`=$lat and `LON`=$lon";

$check_result=mysql_query($check,$db) or die(mysql_error());

}

/*Λήψη μέτρων σχετικά με τους κενούς χαρακτήρες και τα προβλήματα που προκαλούν στον έλεγχο διπλότυπων εγγραφών στα ονόματα.
Σημείωση: μπορούμε να βάλουμε και κεφαλαία γράμματα και να χτυπήσει error αν υπάρχει ήδη το ίδιο όνομα με μικρά γράμματα.
Εκτελούμε τη preg_replace για να κόψουμε το κενό που μπορεί κάποιος να βάλει μπροστά από το όνομα.
*/

//εντολή εξάλειψης περισσότερων των ένα κενών μεταξύ λέξεων και μείωσή τους σε ένα κενό ανά λέξη.

$tel=str_replace(" ","",$tel);

$newname=trim($name);

$newname = preg_replace('/\s|s+/', '', $newname);

//Εκτέλεση συνάρτησης ελέγχου για το αν υπάρχει ξανά αυτή η εγγραφή πηγή σελ.363 βιβλίο php,mysql και apache σε 24 wres

spotChecker($newname,$lat,$lon);

if(mysql_num_rows($check_result) < 1)

{

//Δημιουργία ερωτήματος στη βάση δεδομένων για εισαγωγή όλων των στοιχείων

$query="INSERT INTO spot values('','$newname','$lat','$lon','$address','$tel','$email')";

//εκτέλεση ερωτήματος και αποθήκευση αποτελεσμάτων στη μεταβλητή results

$result=mysql_query($query,$db) or die(mysql_error());

if ($result)

{

echo mysql_affected_rows($db)." spot inserted into database.";

$sql="SELECT * FROM spot";

$result=mysql_query($sql,$db) or die(mysql_error());

```

```

$number_of_rows=mysql_num_rows($result);

echo "The number of spots now are $number_of_rows\n";

}

else

{

echo "An error has occurred. The item was not added.";

}

}

else

{

//εμφάνιση μηνύματος αποτυχίας

echo "<p><strong>This Spot is already in the database.Try another spot.</strong></p>";

}

//τερματισμος συνδεσης

mysql_close($db);

?>

</div></div></div><div id="footer"><p>&nbsp;</p></div></body></html>

```

Αρχείο map.html

```

<!--https://developers.google.com/maps/articles/phpsqlajax_v3 -->

<!DOCTYPE html ><head>

<meta name="viewport" content="initial-scale=1.0, user-scalable=no" />

<meta http-equiv="content-type" content="text/html; charset=UTF-8"/>

<title>Spots on Google Maps</title>

<script type="text/javascript" src="http://maps.googleapis.com/maps/api/js?sensor=false"></script>

<script type="text/javascript">

//<![CDATA[

var customIcons = {

spot: {

icon: 'http://labs.google.com/ridefinder/images/mm_20_blue.png'

```

```

}

};

function load() {

var map = new google.maps.Map(document.getElementById("map"), {

center: new google.maps.LatLng(35.342393, 25.137817),

zoom: 11,

mapTypeId: 'roadmap'

});

var infoWindow = new google.maps.InfoWindow;

// Change this depending on the name of your PHP file

downloadUrl("xmlpoints.php", function(data) {

var xml = data.responseXML;

var markers = xml.documentElement.getElementsByTagName("marker");

for (var i = 0; i < markers.length; i++) {

var id = markers[i].getAttribute("id");

var name = markers[i].getAttribute("name");

var point = new google.maps.LatLng(

parseFloat(markers[i].getAttribute("lat")),

parseFloat(markers[i].getAttribute("lng")));

var address = markers[i].getAttribute("address");

var telephone = markers[i].getAttribute("tel");

var email = markers[i].getAttribute("email");

var html = "<b> ID: " + id + "<br/>" + name + "</b> <br/>" + address + "<br/>" + telephone + "<br/>" + email;

var icon = customIcons['spot'] || {};

var marker = new google.maps.Marker({

map: map,

position: point,

icon: icon.icon

});

```

```

bindInfoWindow(marker, map, infoWindow, html);

}

});

}

function bindInfoWindow(marker, map, infoWindow, html) {

google.maps.event.addListener(marker, 'click', function() {

infoWindow.setContent(html);

infoWindow.open(map, marker);

});

}

function downloadUrl(url, callback) {

var request = window.ActiveXObject ?

new ActiveXObject('Microsoft.XMLHTTP') :

new XMLHttpRequest;

request.onreadystatechange = function() {

if (request.readyState == 4) {

request.onreadystatechange = doNothing;

callback(request, request.status);

}

};

request.open('GET', url, true);

request.send(null);

}

function doNothing() {}

//]]>

</script></head>

<body onload="load()">

<div id="map" style="width: 500px; height: 300px"></div>

<div id="back"><a href="home.html" /><input type="button" value="Back"></a></div></body></html>

```

Αρχείο xmlpoints.php

```
<?php

//require("phpsqlajax_dbinfo.php");

// Start XML file, create parent node

$dom = new DOMDocument("1.0");

$node = $dom->createElement("markers");

$parnode = $dom->appendChild($node);

// Opens a connection to a MySQL server

$connection=mysql_connect ('localhost', "giannis", "1234");

if (!$connection) { die('Not connected : ' . mysql_error());}

// Set the active MySQL database

$db_selected = mysql_select_db("test2", $connection);

if (!$db_selected) {

die ('Can\'t use db : ' . mysql_error());

}

// Select all the rows in the markers table

$query = "SELECT * FROM spot WHERE 1";

$result = mysql_query($query);

if (!$result) {

die('Invalid query: ' . mysql_error());

}

header("Content-type: text/xml");

// Iterate through the rows, adding XML nodes for each

while ($row = @mysql_fetch_assoc($result)){

// ADD TO XML DOCUMENT NODE

$node = $dom->createElement("marker");

$newnode = $parnode->appendChild($node);

$newnode->setAttribute("id",$row['SpotID']);
```

```
$newnode->setAttribute("name",$row['SpotName']);  
  
$newnode->setAttribute("lat", $row['LAT']);  
  
$newnode->setAttribute("lng", $row['LON']);  
  
$newnode->setAttribute("address", $row['Address']);  
  
$newnode->setAttribute("tel", $row['Telephone']);  
  
$newnode->setAttribute("email", $row['Email']);  
  
}  
  
echo $dom->saveXML();  
  
?>
```

Αρχείο geoloc.html

```
<!--https://developers.google.com/maps/documentation/javascript/examples/geocoding-simple -->  
  
<!DOCTYPE html><html><head><meta name="viewport" content="initial-scale=1.0, user-scalable=no"><meta charset="utf-8">  
  
<title>Geocoding service</title><style>  
  
html, body, #map-canvas {  
  
height: 100%;  
  
margin: 0px;  
  
padding: 0px  
  
}  
  
#panel {  
  
position: absolute;  
  
top: 5px;  
  
left: 50%;  
  
margin-left: -180px;  
  
z-index: 5;  
  
background-color: #fff;  
  
padding: 5px;  
  
border: 1px solid #999;  
  
}  
  
</style>
```



```
<script src="https://maps.googleapis.com/maps/api/js?v=3.exp"></script>

<script>

var geocoder;

var map;

function initialize() {

geocoder = new google.maps.Geocoder();

var latlng = new google.maps.LatLng(35.341782, 25.142126);

var mapOptions = {

zoom: 8,

center: latlng

}

map = new google.maps.Map(document.getElementById('map-canvas'), mapOptions);

}

function codeAddress() {

var address = document.getElementById('address').value;

geocoder.geocode( { 'address': address }, function(results, status) {

if (status == google.maps.GeocoderStatus.OK) {

map.setCenter(results[0].geometry.location);

var marker = new google.maps.Marker({

map: map,

position: results[0].geometry.location

});

pos=results[0].geometry.location;

document.getElementById('coo').value=pos;

} else {

alert('Geocode was not successful for the following reason: ' + status);

}

});

}

}
```

```
google.maps.event.addDomListener(window, 'load', initialize);

</script>

</head>

<body>

<div id="panel">

<input id="address" type="text" value="Address">

<input type="button" value="Geocode" onclick="codeAddress()">

<input type="text" id="coo" value="(Latitude,Longitude)">

<a href="home.html" /><input type="button" value="Back" align="right"></a>

</div>

<div id="map-canvas"></div></body></html>
```

Android Εφαρμογή

Αρχείο MainActivity.java

```
package and.ant;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.sql.*;

import java.net.MalformedURLException;

import java.net.URL;

import java.net.URLConnection;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import org.apache.http.HttpEntity;

import org.apache.http.HttpResponse;

import org.apache.http.NameValuePair;

import org.apache.http.client.HttpClient;

import org.apache.http.client.methods.HttpPost;
```

```
import org.apache.http.impl.client.DefaultHttpClient;

import org.apache.http.message.BasicNameValuePair;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import android.app.Activity;

import android.app.Fragment;

import android.app.FragmentManager;

import android.app.FragmentTransaction;

import android.app.ProgressDialog;

import android.content.Context;

import android.content.Intent;

import android.location.Criteria;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.os.AsyncTask;

import android.os.Bundle;

import android.os.Debug;

//import android.support.v4.app.Fragment;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.Menu;

import android.view.View;
```

```
import android.view.ViewGroup;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import com.google.android.gms.common.ConnectionResult;

import com.google.android.gms.common.GooglePlayServicesClient;

import com.google.android.gms.location.LocationClient;

import com.google.android.gms.maps.CameraUpdate;

import com.google.android.gms.maps.CameraUpdateFactory;

import com.google.android.gms.maps.GoogleMap;

import com.google.android.gms.maps.MapFragment;

import com.google.android.gms.maps.model.LatLng;

public class MainActivity extends Activity {

    //      private LatLng loc = new LatLng(0.0,0.0);

    private LatLng LOCATION_CITY = new LatLng(0.0, 0.0);

    private GoogleMap map;

    LocationListener listener;

    LocationManager manager;

    Button findme_button;

    Button back_button;

    Button info_button;

// Creating JSON Parser object
```

```

JSONParser jParser = new JSONParser();

HashMap<String,Float> distancemap;

private static String url_LogIn = "http://192.168.2.3/LogIn.php";

private static String url_latlon = "http://192.168.2.3/latlon.php";

private static String url_info = "http://192.168.2.3/Info.php";

// JSON Node names

private static final String TAG_SUCCESS = "success";

private static String closer_id = "-1";

private float closer_distance =999999999.9f;

// products JSONArray

JSONArray spots = null;

JSONArray spot_info = null;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    map = ((MapFragment) getFragmentManager().findFragmentById(R.id.map)).getMap();

    manager = (LocationManager) this.getSystemService(Context.LOCATION_SERVICE);

    info_button = (Button) findViewById(R.id.info_button);

    info_button.setVisibility(View.INVISIBLE);

    back_button = (Button) findViewById(R.id.back);

    findme_button = (Button) findViewById(R.id.findme);

    //info.setVisibility(View.INVISIBLE);

    back_button.setVisibility(View.INVISIBLE);

    findme_button.setVisibility(View.INVISIBLE);

```

```
FragmentTransaction transaction = getFragmentManager().beginTransaction();

transaction.hide(getFragmentManager().findFragmentById(R.id.info));

transaction.hide(getFragmentManager().findFragmentById(R.id.map));

transaction.show(getFragmentManager().findFragmentById(R.id.Log_In));

transaction.commit();

listener = new LocationListener() {

    @Override

    public void onLocationChanged(Location location) {

        LOCATION_CITY = new LatLng(location.getLatitude(),location.getLongitude());

    }

    @Override

    public void onProviderDisabled(String arg0) {

        // TODO Auto-generated method stub

    }

    @Override

    public void onProviderEnabled(String arg0) {

        // TODO Auto-generated method stub

    }

    @Override

    public void onStatusChanged(String arg0, int arg1, Bundle arg2) {

        // TODO Auto-generated method stub

    }

};

manager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, listener);
```

```

manager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0, listener);

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

    // Inflate the menu; this adds items to the action bar if it is present.

    getMenuInflater().inflate(R.menu.main, menu);

    return true;

}

public void onClick_Location(View v){

    Info info=(Info)getFragmentManager().findFragmentById(R.id.info);

    info.ChangeImage("loading");

    Distance_Task task = new Distance_Task();

    task.execute();

    CameraUpdate update = CameraUpdateFactory.newLatLngZoom(LOCATION_CITY,15);

    map.animateCamera(update);

    System.out.println(LOCATION_CITY.latitude+" "+LOCATION_CITY.longitude);

    if(checkDistance(closer_distance)){

        info_button.setVisibility(View.VISIBLE);

    }

}

public void onClick_Info(View v){

    Info_Task task = new Info_Task();

    task.execute();

    FragmentTransaction transaction = getFragmentManager().beginTransaction();

    transaction.hide(getFragmentManager().findFragmentById(R.id.map));

```



```

transaction.show(getFragmentManager().findFragmentById(R.id.info));

transaction.commit();

back_button.setVisibility(View.VISIBLE);

info_button.setVisibility(View.INVISIBLE);

findme_button.setVisibility(View.INVISIBLE);

        }

public void onClick_Back(View v){

    FragmentTransaction transaction = getFragmentManager().beginTransaction();

    transaction.hide(getFragmentManager().findFragmentById(R.id.info));

    transaction.show(getFragmentManager().findFragmentById(R.id.map));

    transaction.commit();

    back_button.setVisibility(View.INVISIBLE);

    info_button.setVisibility(View.VISIBLE);

    findme_button.setVisibility(View.VISIBLE);

        }

public boolean checkDistance(float dis){

    if(dis<30000000000.0) return true;

    return false;

}

public float computeDistance(float lat, float lon){

    Location mylocation = new Location("MyLoc");

    mylocation.setLatitude(LOCATION_CITY.latitude);

    mylocation.setLongitude(LOCATION_CITY.longitude);

    Location target = new Location("Target");

    target.setLatitude(lat);

```

```

        target.setLongitude(lon);

        float distance = mylocation.distanceTo(target);

        return distance;

    }

    public void onClick_Log_in(View v){

        Login_Task dlTask = new Login_Task();

        dlTask.execute();

    }

    public void Login(){

        FragmentTransaction transaction = getFragmentManager().beginTransaction();

        transaction.show(getFragmentManager().findFragmentById(R.id.map));

        transaction.hide(getFragmentManager().findFragmentById(R.id.Log_In));

        transaction.commit();

        findme_button.setVisibility(View.VISIBLE);

    }

    public void Access_Denied(){

        LogIn Log_in=(LogIn)getFragmentManager().findFragmentById(R.id.Log_In);

        Log_in.Show_Error();

    }

    class Login_Task extends AsyncTask<String, Void, String> {

        protected String doInBackground(String... urls) {

            EditText username = (EditText)findViewById(R.id.textfield_user);

            String usernm="" +username.getText().toString()+"";

            EditText password = (EditText)findViewById(R.id.textfield_pass);

            String pwd="" +password.getText().toString()+"";

```

```

        List<NameValuePair> params = new ArrayList<NameValuePair>();

        params.add(new BasicNameValuePair("password", pwd));

        params.add(new BasicNameValuePair("username", usernm));

        // getting JSON string from URL

        JSONObject json = jParser.makeHttpRequest(url_LogIn, "GET", params);

        try {

int success = json.getInt(TAG_SUCCESS);

        if (success == 1) {

            runOnUiThread(new Runnable() {

                @Override

                public void run() {

                    Login();

                }

            });

        } else {

            runOnUiThread(new Runnable() {

                @Override

                public void run() {

                    Access_Denied();

                }

            });

        }

    } catch (JSONException e) {

```

```

        e.printStackTrace();
    }

    return null;

    }

    //This Method is called when Network-Request finished

    /* protected void onPostExecute(String serverData) {

        textView.setText(serverData)

    }*/

}

class Distance_Task extends AsyncTask<String, Void, String> {

    protected String doInBackground(String... urls) {

        List<NameValuePair> params = new ArrayList<NameValuePair>();

        JSONObject json = jParser.makeHttpRequest(url_latlon, "GET", params);

        // Check your log cat for JSON reponse

        Log.d("spots: ", json.toString());

        try {

            int success = json.getInt(TAG_SUCCESS);

            if (success == 1) {

                distancemap= new HashMap<String,Float>();

                // user found

                // Getting Array of user

                spots = json.getJSONArray("spots");

                System.out.println(spots);

                // looping through All users

                for (int i = 0; i < spots.length(); i++) {

```

```

JSONObject c = spots.getJSONObject(i);

float lat = Float.parseFloat(c.getString("LAT"));

float lon = Float.parseFloat(c.getString("LON"));

String spotid = c.getString("SpotID");

distancemap.put(spotid, computeDistance(lat, lon));

}

for(String s:distancemap.keySet()){

if(distancemap.get(s)<closer_distance){

        closer_distance=distancemap.get(s);

        closer_id=s;

}

}

System.out.println(closer_id);

} else {

System.out.println("wrong");

}

}

} catch (JSONException e) {

e.printStackTrace();

}

return null;

}

//This Method is called when Network-Request finished

/* protected void onPostExecute(String serverData) {

textView.setText(serverData)

```

```

    }*/

}

class Info_Task extends AsyncTask<String, Void, String> {

    protected String doInBackground(String... urls) {

        String spotid = closer_id;

        List<NameValuePair> params = new ArrayList<NameValuePair>();

        params.add(new BasicNameValuePair("spotid", spotid));

        // getting JSON string from URL

        JSONObject json = jParser.makeHttpRequest(url_info, "GET", params);

        Log.d("Info: ", json.toString());

        try {

            int success = json.getInt(TAG_SUCCESS);

            if (success == 1) {

                spot_info = json.getJSONArray("info");

                JSONObject c = spot_info.getJSONObject(0);

                final String spotname = c.getString("SpotName");

                final String address = c.getString("Address");

                final String telephone = c.getString("Telephone");

                final String email = c.getString("Email");

                runOnUiThread(new Runnable() {

                    @Override

                    public void run() {

                        UpdateInfo(spotname,address,telephone,email);

                    }

                });

            }

        } catch (JSONException e) {

            Log.e("Error: ", e.getMessage());

        }

    }

};

```

```

    } else {

    }

} catch (JSONException e) {

    e.printStackTrace();

}

return null;

}

//This Method is called when Network-Request finished

/* protected void onPostExecute(String serverData) {

    textView.setText(serverData)

}*/

}

private void UpdateInfo(String spotname, String address,

    String telephone, String email) {

String pic="";

if(closer_id.equals("67")){

    pic="tei";

}

else{

    pic="airport";

}

Info info=(Info)getFragmentManager().findFragmentById(R.id.info);

info.ChangeImage(pic);

info.ChangeAdress(address);

info.ChangeTitle(spotname);

```

```

        info.ChangeInfo("Telephone: "+telephone+"\n"+"Email: "+email);
    }
}

```

Αρχείο AndroidManifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="and.ant"
    android:versionCode="1"
    android:versionName="1.0" >
<uses-sdk
    android:minSdkVersion="11"
    android:targetSdkVersion="19" />
<permission
    android:name="com.example.mapdemo.permission.MAPS_RECEIVE"
    android:protectionLevel="signature" />
<uses-permission android:name="and.ant.permission.MAPS_RECEIVE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<!--
    The following two permissions are not required to use
    Google Maps Android API v2, but are recommended.
-->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

<uses-feature
    android:glEsVersion="0x00020000"
    android:required="true" />

<application
    android:allowBackup="true"
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name"
    android:theme="@style/AppTheme" >
    <meta-data
        android:name="com.google.android.maps.v2.API_KEY"
        android:value="AIzaSyCnxEjakbRJMCo8zs1ZjNVO8NBVRiKwCSQ" />
    <meta-data
        android:name="com.google.android.gms.version"
        android:value="@integer/google_play_services_version" />

    <activity
        android:name=".MainActivity"
        android:label="@string/app_name" >
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity
        android:name=".Info"
        android:label="@string/title_activity_info" >
    </activity>
    <activity
        android:name=".LogIn"
        android:label="@string/title_activity_log_in" >
    </activity>
</application>

</manifest>

```




Υπηρεσίες εντοπισμού θέσης (Location Based Services) βασισμένες στο GPS των Android μικροσυσκευών.

Κακουλάκης Ιωάννης 2197
Χατζημανώλης Αντώνιος 2410

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανικών Πληροφορικής
ΤΕΙ Κρήτης

Επιβλέπων Καθηγητής: Μαλάμος Αθανάσιος



Συνοπτική Περιγραφή

Η πτυχιακή μας εργασία έχει ως στόχο, τη σχεδίαση και ανάπτυξη μίας εφαρμογής Android, η οποία θα παίρνει τη τοποθεσία του χρήστη και θα εμφανίζει πληροφορίες σχετικές με τη περιοχή στην οποία βρίσκεται. Επιπλέον, απαιτείται η ανάπτυξη μίας βάσης δεδομένων και μίας web-based εφαρμογής, που αποτελεί το Administration μέρος.



Υπηρεσίες Εντοπισμού Θέσης (Location Based Services)

- Τεχνικές Εντοπισμού
 - GPS – GSM
- Πλεονεκτήματα GPS
 - Μεγάλη ακρίβεια στον εντοπισμό
 - Ταχύτητα εντοπισμού
 - Υπολογισμοί που διατίθενται στον χρήστη
πχ. ταχύτητα, υψόμετρο, κατεύθυνση κλπ.



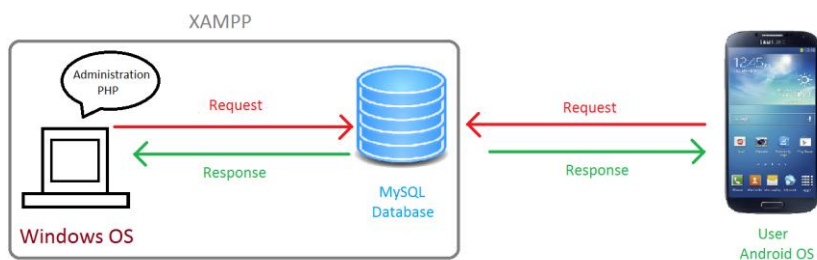
Υπηρεσίες Εντοπισμού Θέσης (Location Based Services)

- Παραδείγματα Χρήσης
 - Μέσα κοινωνικής δικτύωσης
 - Εντοπισμός πλησιέστερου πρατηρίου βενζίνης, νοσοκομείου, φαρμακείου, ATM
 - Παρακολούθηση διαδρομής οχήματος και ενημέρωση κίνησης στο δρόμο
 - Αποστολή διαφημίσεων μόλις ο χρήστης εισέλθει σε μια περιοχή (Geofence)
 - Καιρικές συνθήκες και πρόγνωση καιρού

Εργαλεία και Τεχνολογίες

- Xampp
 - Apache HTTP Server
 - MySQL Βάση Δεδομένων
 - PHP
 - phpMyAdmin
- HTML 5, CSS
- Google Maps API v3 – Javascript
- Android

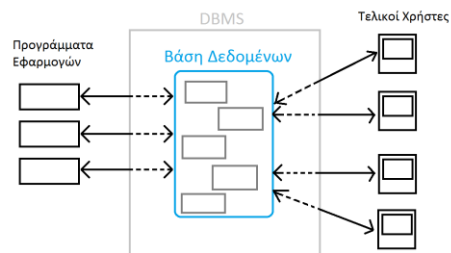
Λειτουργία & Δομή της Εφαρμογής



- I. Βάση Δεδομένων
- II. Administration – Web-Based Εφαρμογή
- III. Android Εφαρμογή

I. Βάση Δεδομένων

- Τι είναι MySQL:
 - Σύστημα Διαχείρισης Σχεσιακών Βάσεων Δεδομένων
 - Τρέχει με τη χρήση ενός εξυπηρετητή



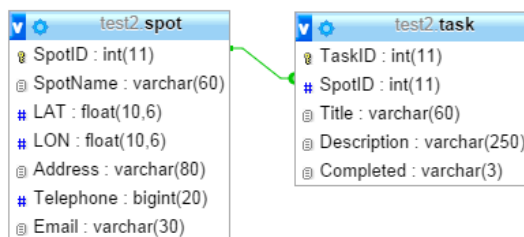
I. Βάση Δεδομένων

- Γιατί MySQL;
 - Ευκολία στην εγκατάσταση
 - Ευκολία στη χρήση, με γνώση μόνο των βασικών εντολών της MySQL
 - Είναι δωρεάν
 - Παρέχει ασφάλεια πολλών επιπέδων σε δεδομένα και σε κωδικούς
 - Επιτρέπει τον ορισμό δικαιωμάτων σε χρήστες
 - Δυνατότητα εκτέλεσης σε διάφορα λειτουργικά συστήματα
 - Μπορεί να αποθηκεύσει μεγάλο όγκο δεδομένων
 - Ταχύτερη σε σχέση με άλλα συστήματα Διαχείρισης

I. Βάση Δεδομένων

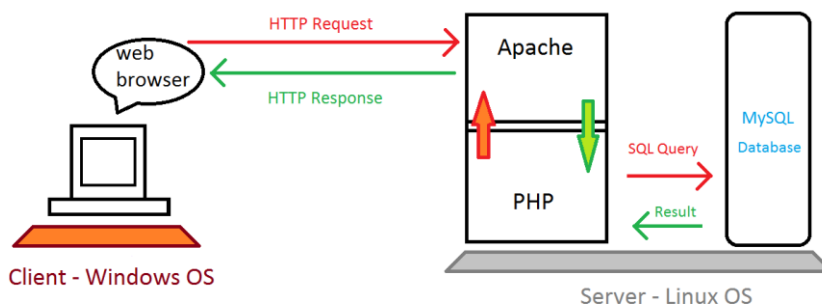
Πίνακας	Ενέργεια	Εγγραφές	Τύπος	Σύνθεση	Μέγεθος	Περίσσεια
admins	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~3	InnoDB	utf8_general_ci	16 KB	-
spot	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~2	InnoDB	utf8_general_ci	16 KB	-
task	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~2	InnoDB	utf8_general_ci	32 KB	-
users	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~2	InnoDB	utf8_general_ci	16 KB	-
4 πίνακες	Σύνολο		9	InnoDB	utf8_general_ci	80 KB 0 B

Τύπος σχέσης: ένα-προς-πολλά



I. Βάση Δεδομένων

Η σχέση της PHP με τη MySQL Βάσης Δεδομένων

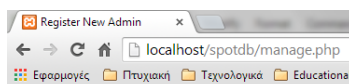


II. Administration

Please login

Username:

Password:



Subscribe/Unsubscribe

Username:

Email Address:

Password:

Enter Your Password Again:

Type of User:
 Administrator User

Action:
 Subscribe Unsubscribe

II. Administration

Welcome to the Administration

Please select an option to continue:

Spots

- Show All Spots
- Add New Spot
- Update a Spot
- Search a Spot
- Delete a Spot
- Show Spots on Map

Tasks

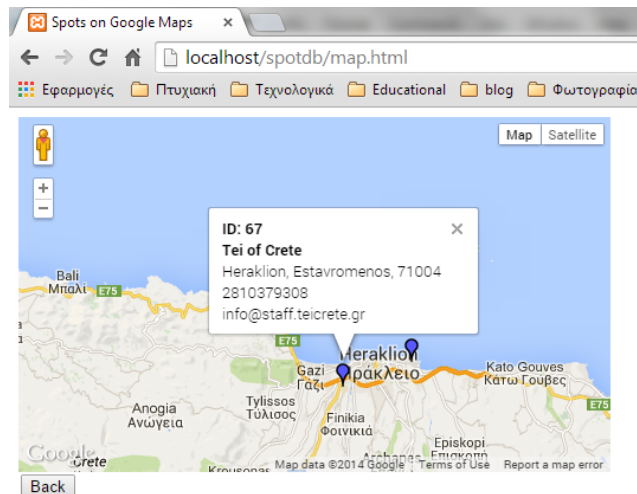
- Show All Tasks
- Add New Task
- Update a Task
- Search a Task
- Delete a Task

Help Maps

- I'm looking for Coordinates
- I'm looking for an Address
- Jump to Google Maps

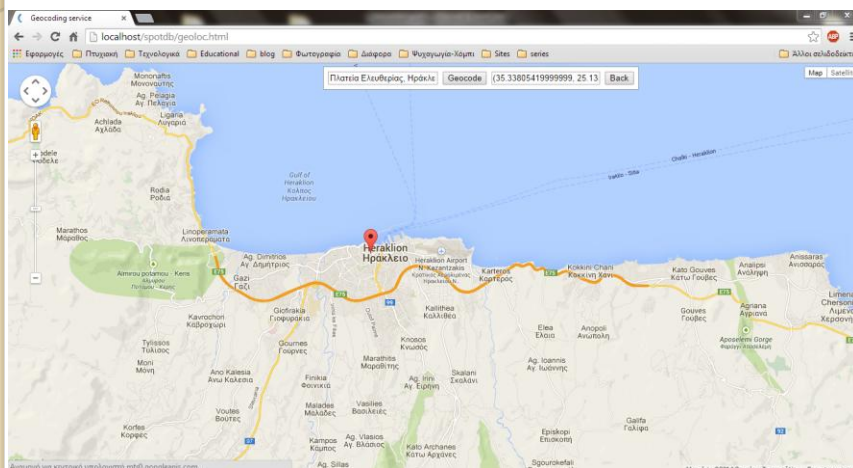
II. Administration

Εμφάνιση τοποθεσιών σε χάρτη



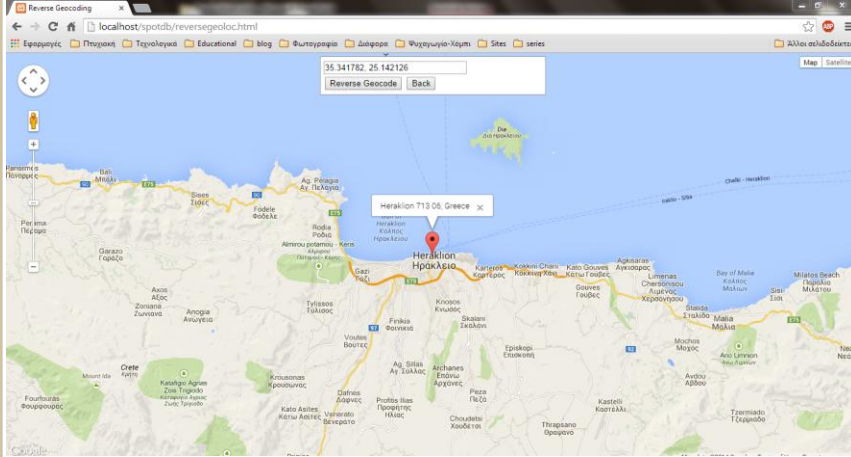
II. Administration

Παράδειγμα χρήσης υπηρεσίας Geocode:



II. Administration

Παράδειγμα χρήσης υπηρεσίας Reverse Geocode:



II. Administration

Έλεγχοι και ενέργειες:

- Αφαίρεση χαρακτήρων «κενό» στις φόρμες
- Έλεγχος κεφαλαίων-πεζών χαρακτήρων
- Έλεγχος και αποφυγή κενών πεδίων στις φόρμες
- Έλεγχος συμπλήρωσης υποχρεωτικών πεδίων
- Ταυτοποίηση στοιχείων εισόδου
- Έλεγχος και αποφυγή διπλότυπων εγγραφών στη Βάση Δεδομένων
- Εξασφάλιση ασφαλών κωδικών πρόσβασης
- Μορφοποίηση δεδομένων στις φόρμες για αποθήκευση στη Βάση Δεδομένων



II. Administration

Πλεονεκτήματα:

- Ολοκληρωμένη εφαρμογή διαχείρισης της Βάσης Δεδομένων
- Ασφαλής τρόπος διαχείρισης της Βάσης Δεδομένων
- Απομάκρυνση του Διαχειριστή από τη δημιουργία ερωτημάτων με κώδικα
- Παροχή βοηθητικών μέσων
- Γραφικός και φιλικός τρόπος με οδηγίες και συμβουλές
- Διασφάλιση ελέγχου και σωστής αποθήκευσης των δεδομένων στη Βάση



III. Android

Πλεονεκτήματα χρήσης:

- Ευρέως γνωστό και διαδεδομένο με τεράστια απήχηση στην αγορά εφαρμογών (Google Play Store)
- Εύχρηστο και ευέλικτο λειτουργικό σύστημα
- Ανοιχτού Κώδικα (Open Source)
- Συνεχείς ενημερώσεις και νέες διανομές

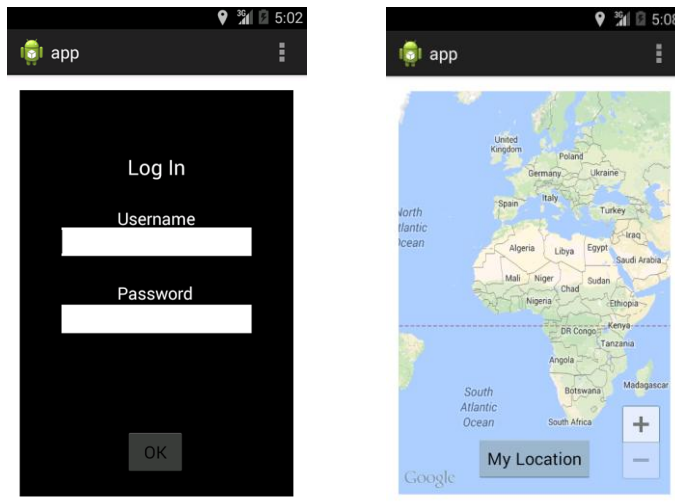
III.Android

Πλεονεκτήματα των Android Developers:

- Έλλειψη χρηματικού αντίτιμου για ανάπτυξη
- Χρήση ελεύθερων εργαλείων για ανάπτυξη
- Άριστη διαχείριση του υλικού μέρους μιας συσκευής
- Δυνατότητα συνεργασίας δύο εφαρμογών μεταξύ τους
- Άψογη συνεργασία με Google Maps, Google Play Services
- Emulator για δοκιμές
- Συχνή ενημέρωση των APIs, tools μέσω SDK Manager
- Μεγάλη υποστήριξη σε προγραμματιστές μέσω demos, documentation κλπ.

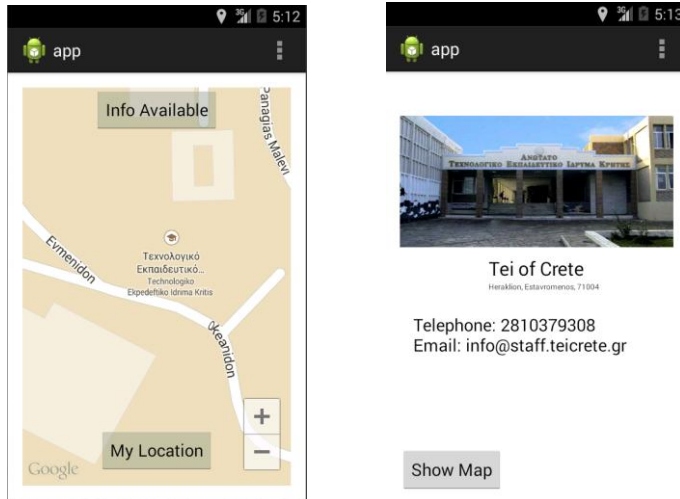
III.Android

Στάδια Λειτουργίας της εφαρμογής μας:



III. Android

Στάδια Λειτουργίας της εφαρμογής μας:



III. Android

Πλεονεκτήματα της Android εφαρμογής:

- Ιδιαίτερα γρήγορη και λειτουργική
- Απλή στη χρήση (User Friendly)
- Χαμηλή κατανάλωση ενέργειας
- Εντοπισμός χρήστη τόσο με **GPS** όσο και μέσω **Internet**
- Υπολογισμός απόστασης του χρήστη από μια περιοχή



Μελλοντική Δουλειά

- Προσθήκη περισσότερων σημείων ενδιαφέροντος στη βάση δεδομένων μας
- Θα εμπλουτίσουμε τις πληροφορίες στις οποίες έχει πρόσβαση ο χρήστης, όπως για παράδειγμα σχόλια και αξιολογήσεις χρηστών για μια περιοχή
- Περισσότερες λειτουργίες όπως πυξίδα και αναζήτηση τοποθεσιών πάνω στο χάρτη
- Ομορφότερο και πιο εύχρηστο γραφικό περιβάλλον χρήστη (GUI)
- Προσθήκη ενός `help button`, σχετικά με βοήθεια και οδηγίες χρήσης της εφαρμογής



Τέλος Παρουσίασης

Ερωτήσεις;