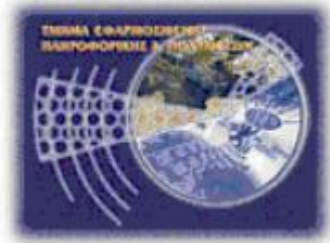




**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης**

**Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Μηχανικών Πληροφορικής**



## **Πτυχιακή εργασία**

**Ανάπτυξη εφαρμογής Android για την  
παροχή υπηρεσιών δρομολόγησης και  
καθοδήγησης στους χρήστες για τις  
καθημερινές τους δραστηριότητες**

**Αρβανίτης Κώστας, Α.Μ. 2577**

**Καλυκάκης Λευτέρης, Α.Μ. 2942**

**Επιβλέπων Καθηγητής: Παπαδάκης Νίκος**



## Σύνοψη

Με την μεγάλη εξάπλωση των κινητών τηλεφώνων ένα μεγάλο μέρος του πληθυσμού έχει έρθει σε επαφή με ένα πλήθος νέων τεχνολογιών, οι οποίες παλιότερα ήταν διαθέσιμες μόνο μέσω της χρήσης εξειδικευμένων ή μη φορητών συσκευών. Δύο από αυτές τις τεχνολογίες είναι ο αυτόματος προσδιορισμός της γεωγραφικής θέσης της συσκευής και η πρόσβαση σε δορυφορικούς χάρτες.

Σε αυτή την εργασία θα παρουσιαστεί η ανάπτυξη μίας εφαρμογής που χρησιμοποιεί τις παραπάνω τεχνολογίες για να δίνει στον χρήστη τη δυνατότητα να οργανώσει καλύτερα τον χρόνο του. Η εφαρμογή τρέχει σε κινητά με λειτουργικό σύστημα Android και χρησιμοποιεί τους χάρτες που προσφέρονται από την εταιρεία Google.

## **Abstract**

With the wide spread of mobile phones much of the population has come into contact with a number of new technologies that were previously only available through the use of specialized or non-portable devices. Two of these technologies is the automatic determination of the geographical position of the device and access to satellite maps.

This paper will present the development of an application that uses these technologies to give the user the ability to better organize his time. The application runs on mobile with Android operating system and uses the maps offered by the company Google.

## Περιεχόμενα

Σύνοψη .....	3
Abstract .....	4
Περιεχόμενα.....	5
1 Εισαγωγή .....	8
1.1 Περίληψη.....	8
1.2 Κίνητρο για την διεξαγωγή της εργασίας.....	8
1.3 Σκοπός και στόχοι εργασίας.....	8
1.4 Δομή εργασίας .....	8
2 Έξυπνα τηλέφωνα και εφαρμογές για κινητά.....	9
2.1 Γενικά.....	9
2.2 Τι είναι έξυπνο τηλέφωνο .....	9
2.3 Υλικό των έξυπνων κινητών .....	10
2.3.1 Γενικά .....	10
2.3.2 Επεξεργαστής .....	11
2.3.3 Μνήμη .....	11
2.3.4 Είσοδος.....	11
2.3.5 Οθόνη .....	11
2.3.6 Ασύρματη δικτύωση .....	12
2.3.7 Φωτογραφική μηχανή.....	12
2.3.8 Υλικό εύρεσης θέσης.....	12
2.3.9 Συνδεσιμότητα .....	12
3 Το λειτουργικό Android.....	13
3.1 Εισαγωγή .....	13
3.2 Ιστορική αναδρομή .....	13
3.3 Βασικά Χαρακτηριστικά .....	14
3.4 Οι εκδόσεις του Android .....	15
3.4.1 Android 1.0 και Android 1.1 .....	16
3.4.2 Android 1.5 Cupcake .....	16
3.4.3 Android 1.6 Donut .....	16
3.4.4 Android 2.0 Eclair .....	17
3.4.5 Android 2.2-2.2.3 Froyo .....	17
3.4.6 Android 2.3-2.3.7 Gingerbread .....	18
3.4.7 Android 3.0-3.2 Honeycomb .....	19
3.4.8 Android 4.0-4.0.2 Ice Cream Sandwich .....	19
3.4.9 Android 4.1-4.3.1 Jelly Bean.....	19

3.4.10	Android 4.4-4.4.4 Kit Kat .....	20
3.4.11	Android 5.0 "Lollipop" .....	20
	Τον Ιούνιο του 2014 ανακοινώθηκε από την Google η νέα έκδοση του Android αλλά θα είναι διαθέσιμη τον τον Νοέμβριο του 2014 για επιλεγμένες συσκευές που τρέχουν Android, συμπεριλαμβανομένων τις συσκευές Nexus.....	20
3.5	Επίπεδα του λειτουργικού .....	21
3.5.1	Linux Kernel .....	21
3.5.2	Libraries .....	21
3.5.3	Android Runtime .....	21
3.5.4	Application Framework .....	21
3.5.5	Applications .....	22
3.6	Ασφάλεια στο Android .....	22
3.7	Firmware σε συσκευές Android .....	23
3.7.1	Λογαριασμός root .....	23
3.8	Η γλώσσα προγραμματισμού Java .....	24
3.8.1	Java .....	24
3.8.2	Τα χαρακτηριστικά της Java .....	24
4	Περιγραφή εφαρμογής .....	41
4.1	Εισαγωγή .....	41
4.2	Γενική εικόνα της εφαρμογής .....	41
4.3	Απαραίτητα εργαλεία.....	41
4.3.1	Android SDK.....	41
4.3.2	Εγκατάσταση πακέτων SDK.....	42
4.3.3	Επιπλέον βιβλιοθήκες .....	44
5	Οδηγίες χρήσης .....	46
5.1	Εισαγωγή .....	46
5.2	Είσοδος.....	46
5.3	Δημιουργία λογαριασμού .....	46
5.4	Βασική οθόνη χάρτη.....	47
5.5	Επιλογή λειτουργιών.....	48
5.6	Εμφάνιση καλύτερης διαδρομής .....	49
6	Συμπεράσματα .....	51
6.1	Εφαρμογή και περιβάλλον Android .....	51
6.2	Μελλοντική επέκταση .....	51
7	Βιβλιογραφία.....	53
8	Παράρτημα Α – Κώδικας εφαρμογής.....	55
8.1	AndroidManifest.xml.....	55
8.2	Utils.java .....	56

8.3	AbstractMapNavActivity.java .....	57
8.4	LoginActivity.java.....	58
8.5	RegisterActivity.java .....	59
8.6	GlobalData.java .....	60
8.7	DefaultMapActivity.java .....	61
8.8	MovieActivity.java .....	69
8.9	LType.java.....	72
8.10	GlobalData.java .....	72
8.11	activity_login.xml.....	78
8.12	activity_register.xml .....	79
8.13	default_map.xml .....	82
8.14	activity_default_map.xml.....	83
8.15	activity_movie.xml.....	83

# 1 Εισαγωγή

## 1.1 Περίληψη

Αυτή η εργασία ασχολείται με την ανάπτυξη μίας εφαρμογής για κινητά τηλέφωνα με λειτουργικό σύστημα Android. Συγκεκριμένα θέλουμε να δείξουμε πόσο χρήσιμη μπορεί να είναι μία εφαρμογή που προσφέρει δυνατότητα εύρεσης διαφόρων τοποθεσιών και να δίνει οδηγίες στον χρήστη για τη βέλτιστη διαδρομή προς μία τοποθεσία.

## 1.2 Κίνητρο για την διεξαγωγή της εργασίας

Σχεδόν όλα τα σύγχρονα κινητά τηλέφωνα υποστηρίζουν λειτουργίες εύρεσης θέσης, προβολής χαρτών με τη θέση του χρήστη πάνω σε αυτούς και οδηγίες πλοήγησης προς συγκεκριμένους προορισμούς. Πρόκειται για μερικά πολύ ισχυρά εργαλεία, τα οποία θα μπορούσαν να συνδυαστούν με τέτοιο τρόπο ώστε να διευκολύνουν αρκετά την καθημερινότητα των χρηστών τους.

## 1.3 Σκοπός και στόχοι εργασίας

Σκοπός της εργασίας είναι να μελετήσει τις δυνατότητες του λειτουργικού Android στους τομείς που αναφέρθηκαν παραπάνω, δηλαδή στους διαδραστικούς ηλεκτρονικούς χάρτες και στον υπολογισμό διαδρομών ανάμεσα σε διαφορετικούς προορισμούς. Αυτή η μελέτη θα γίνει τόσο σε θεωρητικό επίπεδο όσο και πρακτικά μέσω της ανάπτυξης μίας εφαρμογής με αυτές τις δυνατότητες.

## 1.4 Δομή εργασίας

**Κεφάλαιο 1 – Περίληψη:** Πρόκειται για αυτό το κεφάλαιο, με γενικές πληροφορίες για την εργασία.

**Κεφάλαιο 2 – Έξυπνα τηλέφωνα και εφαρμογές:** Σε αυτό το κεφάλαιο παρουσιάζουμε κάποια βασικά στοιχεία για τα έξυπνα τηλέφωνα και τις εφαρμογές τους, καλύπτοντας τις δυνατότητες που δίνουν στους χρήστες.

**Κεφάλαιο 3 – Το λειτουργικό σύστημα Android:** Αυτό το κεφάλαιο δείχνει το λειτουργικό σύστημα Android και παρουσιάζει την δομή και τις δυνατότητές του.

**Κεφάλαιο 4 – Περιγραφή εφαρμογής:** Αυτό το κεφάλαιο περιέχει μία γενική περιγραφή των στοιχείων της εφαρμογής.

**Κεφάλαιο 5 – Οδηγίες χρήσης:** Αυτό το κεφάλαιο εξηγεί τον τρόπο χρήσης της εφαρμογής.

**Κεφάλαιο 6 – Συμπεράσματα:** Στο κεφάλαιο αυτό παρουσιάζουμε τα συμπεράσματα που προέκυψαν μέσω της διαδικασίας ανάπτυξης της εφαρμογής.



## 2 Έξυπνα τηλέφωνα και εφαρμογές για κινητά

### 2.1 Γενικά

Η εμφάνιση των smartphones, ή έξυπνων τηλεφώνων, όπως μεταφράζονται στα ελληνικά που τα προσφωνούν σε όλα τα μέσα μαζικής ενημέρωσης οδήγησε στην μεγάλη επανάσταση που έγινε στον χώρο των κινητών τηλεφώνων τα τελευταία χρόνια.

Ο λόγος ύπαρξης και εμφάνισης αυτών των συσκευών δόθηκε γενικά από την εξέλιξη της τεχνολογίας και πιο συγκεκριμένα του hardware, το οποίο πλέον στις μέρες μας είναι συγκρίσιμο με τους συμβατικούς υπολογιστές παλαιότερων χρόνων αλλά εξελίσσεται ραγδαία και συνεχώς. Έτσι αφού υπάρχει η «μαγιά» τότε υπάρχουν και νέες προοπτικές και δυνατότητες που μπορούν να προσφέρουν ώστε αυτές οι συσκευές να χρησιμοποιηθούν με πάρα πολλούς τρόπους σε σχέση με τις παραδοσιακές.



*Smart phones*

Η εξέλιξη αυτή οδήγησε ώστε η νοοτροπία τόσο των χρηστών αλλά και των εταιρειών ως προς τον τρόπο αντιμετώπισης τους απέναντι στα κινητά τηλέφωνα να αλλάξει και να αντιμετωπίζεται περισσότερο ως ένας φορητός υπολογιστής με την ιδιότητα και του τηλεφώνου αλλά σε μέγεθος κινητού τηλεφώνου. Όσο αφορά τις εταιρίες η ιδέα αυτή προσέφερε μεγάλο κέρδος καθώς σε ένα τέτοιο σύστημα μπορεί να υπάρχουν προγράμματα που γράφονται από τρίτους προγραμματιστές και κυρίως πως αυτά τα προγράμματα βελτιώνουν την ίδια τη συσκευή και την κάνουν πιο ελκυστική στους καταναλωτές.

### 2.2 Τι είναι έξυπνο τηλέφωνο

Το έξυπνο τηλέφωνο (smartphone) είναι ένα κινητό τηλέφωνο βασισμένο σε ένα λειτουργικό σύστημα κινητής τηλεφωνίας με περισσότερη προηγμένη υπολογιστική ικανότητα και συνδεσιμότητα σε σχέση με ένα απλό κινητό τηλέφωνο. Οι κλασικές συσκευές κινητής τηλεφωνίας αποτελούν τον «πρόγονο» τους και τα έξυπνα κινητά τηλέφωνα (smartphones), αποτελούν την φυσική εξέλιξη των προκατόχων είτε τον «απόγονο» τους. Δεν υπάρχει ένας γενικά ορισμός που θα μπορούσαμε να αναφέρουμε, αλλά γενικά θα λέγαμε πως ένα έξυπνο κινητό τηλέφωνο είναι ένα «σούπερ» τηλέφωνο που μπορεί ανάλογα με χαρακτηριστικά, δυνατότητες, λειτουργικό αλλά και επιθυμίες και ανάγκες του χρήστη να διευκολύνει την ζωή του κατόχου και να έχει μια συσκευή τηλεπικοινωνίας αλλά και εφαρμογών που ως τώρα μπορούσε να βρει μόνο εάν ήταν κάτοχος ηλεκτρονικού υπολογιστή, την επεξεργασία κειμένων, την λήψη και αποστολή μηνυμάτων ηλεκτρονικού ταχυδρομείου (e-mail), σερφάρισμα στο διαδίκτυο κ.λπ.

Γνωρίζοντας ότι, από την μία πλευρά, τα Personal Digital Assistants (PDA) είναι ηλεκτρονικές φορητές ατζέντες που μπορούσαν να επικοινωνήσουν με τον υπολογιστή

για ανταλλαγή στοιχείων και να διευκολύνουν στις ανάγκες του κατόχου τους χωρίς την μεταφεριστικότητα του ηλεκτρονικού τους υπολογιστή αλλά από την άλλη πλευρά την ραγδαία ανάγκη του ανθρώπου να κατέχει κινητό τηλέφωνο οδηγηθήκαμε στον συγκερασμό αυτών των δύο και είχαμε την καινοτομία στον κομό της τεχνολογίας με τα έξυπνα κινητά τηλέφωνα . Φυσικά, οι ανάγκες και οι απαιτήσεις του αγορίστικού κοινού αυξηθήκαν και έτσι σε μεταγενέστερα μοντέλα προστέθηκαν οι low-end compact ψηφιακές φωτογραφικές μηχανές, οι λειτουργίες των φορητών media players, οι βιντεοκάμερες τσέπης, καθώς και μονάδες πλοήγησης GPS, με αποτέλεσμα να διαμορφωθεί μια πολυχρηστική συσκευή.

Οι web browsers που εμφανίζουν τυποποιημένες ιστοσελίδες καθώς και βελτιστοποιημένες ιστοσελίδες, όπως επίσης οθόνες αφής υψηλής ανάλυσης αποτελούν δυνατότητες που περιλαμβάνουν πολλά σύγχρονα smartphones. Η κάλυψη και η προσβασιμότητα μέσω Wi-Fi και μέσω κινητών ευρυζωνικών υπηρεσιών σε δεδομένα που απαιτούν υψηλή ταχύτητα που παρέχουν τα smartphones οδήγησε ώστε να κατακλύσουν σε πωλήσεις την αγορά τα τελευταία χρόνια τόσο οι ίδιες αυτές συσκευές αλλά και οι εφαρμογές που προγραμματίζονται πάνω σε αυτά και τα αξεσουάρ τους.



*SmartPhone and Web*

Επιπρόσθετες δυνατότητες αποτελεί η ύπαρξη συστήματος εντοπισμού θέσης (π.χ. A-GPS). Πρόσφατα διάφορα έξυπνα κινητά παρέχονται με εσωτερικούς ιδιαίτερους αισθητήρες κίνησης (Accelerometer), προσφέροντας την δυνατότητα για πιο εξελιγμένο και φυσικό χειρισμό της συσκευής.

Από πλευρά κοινωνιολογίας, μπορούμε να αναφέρουμε ότι η δυνατότητα συνεχούς πρόσβασης στο διαδίκτυο (mobile internet) που προσφέρεται μέσω των συσκευών έξυπνων κινητών, έχουν αλλάξει τα δεδομένα στην διασκέδαση και την πληροφόρηση.

## **2.3 Υλικό των έξυπνων κινητών**

### **2.3.1 Γενικά**

Στην αγορά αυτή τη στιγμή κυκλοφορούν πολλά διαφορετικά μοντέλα smartphones τα οποία έχουν τεράστιες διαφορές μεταξύ τους σε χαρακτηριστικά, δυνατότητες σε επίπεδο απαιτήσεων που μπορούν να καλύψουν. Κατά συνέπεια μπορούμε να κάνουμε κάποιες γενικές παρατηρήσεις που κάποιος ενδιαφέρεται να γίνει ιδιοκτήτης ενός smartphone και θα έπρεπε να γνωρίζει ποια χαρακτηριστικά στο hardware τους είναι σημαντικά για αυτόν.

### 2.3.2 Επεξεργαστής

Το multitasking δηλαδή η παράλληλη εξυπηρέτηση ολοένα και περισσότερων λειτουργιών οδήγησε στην ανάγκη ώστε τα περισσότερα έξυπνα κινητά τηλέφωνα να διαθέτουν αρκετά ισχυρούς επεξεργαστές. Σε πολλά κινητά συναντάμε πλέον επεξεργαστές με δύο ή τέσσερις πυρήνες, ενώ η τάση είναι να αυξάνονται και οι συχνότητες λειτουργίας τους, οι οποίες κυμαίνονται σήμερα από 800 MHz έως και 1,8 GHz.

### 2.3.3 Μνήμη

Η ανάγκη για συσκευές με περισσότερη μνήμη αποθήκευσης δημιουργήθηκε εξαιτίας των απαιτητικών λειτουργιών των smartphones. Η μνήμη αυτή ποικίλει από συσκευή σε συσκευή και μπορεί να φτάσει σε μεγέθη εφάμιλλα ενός επιτραπέζιου υπολογιστή, σήμερα μέχρι και 2GB.

Οι εφαρμογές που μπορούν να τρέξουν σε μια τέτοια συσκευή δεν αφορούν μόνο αναπαραγωγή πολυμέσων αλλά την επεξεργασία των περισσότερων μορφών δεδομένων(αρχεία κειμένου, εικόνας, ήχου) καθώς και ενασχόληση με παιχνίδια.

### 2.3.4 Είσοδος

Η επαναστατική διαφορά ενός «απλού» κινητού με ένα «έξυπνο» είναι ότι ο χειρισμός του τηλεφώνου και η είσοδος δεδομένων σε αυτό γίνεται μέσω αφής. Οι περισσότερες λειτουργίες γίνονται με το άγγιγμα, τα πλήκτρα έχουν καταργηθεί σχεδόν εξ ολοκλήρου με την μόνη εξαίρεση τα υποτυπώδη πλήκτρα, όπως αυτό που οδηγεί στην αρχική οθόνη του κινητού.



*Ενσωματωμένο πληκτρολόγιο σε smartphone*

Η έννοια και η ανάγκη πλέον του πληκτρολογίου έχει σχεδόν εξαλειφθεί ως υποχρεωτική και αποτελεί πλέον προαιρετική καθώς σε κάποιες συσκευές υπάρχει ένα πλήρες συρόμενο πληκτρολόγιο πίσω από την οθόνη του κινητού ή για κάποιες άλλες προσφέρεται ένα αποσπώμενο πληκτρολόγιο σαν επιπλέον αξεσουάρ.

### 2.3.5 Οθόνη

Μια σημαντικότερη διαφορά ενός απλού κινητού με ένα έξυπνο κινητό είναι η οθόνη. Το εικονικό πληκτρολόγιο (αφής), η αναπαραγωγή των πολυμέσων, οι λίστες με τα μηνύματα, οι χάρτες και τα παράθυρα του browser φιλοξενούνται στην διαγώνια της οθόνης, η οποία αποκτά πλέον σημασία. Οι οθόνες στα κινητά τηλέφωνα ποικίλουν

και μπορούν να είναι από 2,8 ίντσες μέχρι και 10 ίντσες αναλόγως τη συσκευή. Επίσης, ιδιαίτερη σημασία έχει η υποστηριζόμενη ανάλυση (συνήθως είναι 480x800).

Τεχνολογίες όπως το multi-touch (εκτέλεση λειτουργίας με παραπάνω από ένα ταυτόχρονο άγγιγμα της οθόνης ) δημιουργούν νέες δυνατότητες και λειτουργίες σε ένα έξυπνο κινητό τηλέφωνο αφού πλέον η οθόνη δεν σημαίνει μόνο αναπαραγωγή περιεχομένου αλλά και είσοδος με την αφή.

### 2.3.6 Ασύρματη δικτύωση

Η ευκολία πρόσβασης στο διαδίκτυο είτε μέσω τηλεπικοινωνιακών δικτύων (3G, 4G) είτε μέσω ασύρματων δικτύων Wi-Fi είναι μια άλλη σημαντική διαφορά. Ο συνδυασμός μιας πληθώρας εφαρμογών με την γρήγορη πλοήγηση στο διαδίκτυο, ανάλογα βέβαια με το λειτουργικό που διαθέτει η συσκευή, προσφέρει άμεση πρόσβαση σε σελίδες κοινωνικής δικτύωσης αλλά και γενικότερα προσφέρει πρόσβαση σε όλες τις γνωστές σελίδες αναπαραγωγής πολυμέσων και διαδικτύου.

### 2.3.7 Φωτογραφική μηχανή

Φωτογραφίες υψηλής ανάλυσης και ποιότητας, βίντεο υψηλής ευκρίνειας, σε συνδυασμό με εφαρμογές επεξεργασίας βίντεο και εικόνας ανήκουν πλέον στα χαρακτηριστικά των περισσότερων έξυπνων κινητών τηλεφώνων ακόμα και των πιο οικονομικών. Στο σημείο αυτό, θα μπορούσαμε να σχολιάσουμε ότι η συνεχής βελτίωση των συσκευών και παράλληλα αύξησης δυνατοτήτων του έχει ως αποτέλεσμα πολλές φορές οι δυνατότητες μια απλή φωτογραφική μηχανή να ξεπερνιούνται από τις αντίστοιχες των ενσωματωμένων ψηφιακών μηχανών. Πολλές έξυπνες συσκευές προσφέρουν την δυνατότητα στον χρήστη πραγματοποίησης τηλεδιασκέψεων είτε παρακολούθησης βίντεο μέσω μίας δευτερεύουσας κάμερας χαμηλότερης ανάλυσης στην πλευρά της οθόνης που έχουν ενσωματώσει.

### 2.3.8 Υλικό εύρεσης θέσης

Η υποστήριξη λειτουργιών πλοήγησης με χάρτες (Google Maps) είτε με GPS (σύστημα εντοπισμού θέσης), είτε με A-GPS (σύστημα εύρεσης θέσης με την βοήθεια δικτύου κινητής τηλεφωνίας) είναι μια επιπλέον λειτουργία που παρέχετε από τα περισσότερα έξυπνα κινητά τηλέφωνα.

### 2.3.9 Συνδεσιμότητα

Μια άλλη σημαντική παράμετρος που διαφοροποιεί τις συσκευές αυτές είναι η πληθώρα επιλογών συνδεσιμότητας (Bluetooth, καλώδιο σύνδεσης USB, Tv-out) με άλλες ηλεκτρονικές συσκευές, πράγμα που διευκολύνει τη διαχείριση και αναπαραγωγή του περιεχομένου τους.

## 3 Το λειτουργικό Android

### 3.1 Εισαγωγή

Το Android είναι ένα λειτουργικό σύστημα που ενσωματώνεται σε συσκευές κινητής τηλεφωνίας, τα οποία διαθέτουν οθόνη αφής, τρέχουν τον πυρήνα (kernel) του λειτουργικού Linux και ακόμη, επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με τη χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας τη συσκευή μέσω βιβλιοθηκών λογισμικού της Google.



*Η οικογένεια των Smartphones!*

Στις μέρες μας, υπάρχουν πάρα πολλές συσκευές με Android, όπου βέβαια η κάθε μία έχει τα δικά της διαφορετικά χαρακτηριστικά και κατασκευάζονται από διαφορετικές εταιρίες όπου η κάθε μια δίνει διαφορετική βαρύτητα σε διαφορετικά χαρακτηριστικά. Μερικές από τις εταιρίες που χρησιμοποιούν το λειτουργικό Android για τα smartphones τους είναι η LG, Samsung, HTC, Sony Ericsson, Motorola.

### 3.2 Ιστορική αναδρομή

Οι Rubin, Miner, Sears και White δημιούργησαν το Android, το 2003 στην Καλιφόρνια. Η δυνατότητα εντοπισμού και αναγνώρισης της τοποθεσίας του ιδιοκτήτη ενός έξυπνου τηλεφώνου οποιαδήποτε στιγμή και αν χρειαστεί ήταν η ανάγκη-αφορμή που οδήγησε στην δημιουργία του λειτουργικού αυτού. Στην αρχή δούλευαν κρυφά από την δημοσιότητα της κοινωνίας της πληροφορικής όσπου το 2005, ο Andy Rubin αποφάσισε να συνεργαστεί με την Google ώστε να ραχρησιμοποιηθεί ως την κύρια μηχανή αναζήτησης για το T-Mobile Sidekick. Έτσι, πραγματοποιήθηκε μια συνάντηση του Andy Rubin με τον Larry Page, ο οποίος είναι ένας από τους δύο υδρυτές της Google και σε αυτή την συνάντηση έγινε η παρουσίαση του Android ως ένα παγκόσμιο ανοιχτό λειτουργικό σύστημα για έξυπνα τηλέφωνα. Η συνάντηση αυτή είχε ως αποτέλεσμα Η Google να εξαγοράσει το Android και η εποχή της μεγάλης ανάπτυξης του ξεκίνησε. Η Google συνέχισε να κρατάει μυστικό το λειτουργικό σύστημα καθώς δούλευε μυστικά πάνω σε αυτό και κατωχείρωνε πατέντες και έκανε επαγγελματικές συναντήσεις ψάχνοντας για συνεργάτες και επενδυτές μέχρι το 2007 όπου η εποχή του απλού κινητού τηλεφώνου ως αναγκαιο μέσο επικοινωνίας αλλά σε συνδιασμό με τις δυνατότητες που μπορούσε να προσφέρει ένα έξυπνο κινητό τηλέφωνο θα μπορούσε να κατακλίσει την αγορά και να ανταγωνιστεί για πολλούς χρήστες και την ανάγκη αγοράς ενός υπολογιστή, πράγμα που επιβεβαιώνεται καθημερινά αν παρατηρηθούν οι πωλησεις των εκάστωτε προϊόντων.



*Υπάρχει μία παραδοσιακή αντιμαχία ανάμεσα στο Android και την Apple*

Παράλληλα, η συγκοιρία του αιώνιου αντιπάλου, εννοώντας την Apple, η οποία το 2007 το καλοκαίρι, να λανσάρει στην αγορά το iPhone οδηγεί την ανάγκη απάντησης της Google. Ο επιχειρηματικός και τεχνολογικός κόσμος νομίζει ότι θα απαντήσει με το gPhone, πράγμα το οποίο δεν έγινε φυσικά. Το φθινόπωρο του 2007, ανακοινώνεται το μυστικό και οι μυστικές συννομοσιες στον κοσμο της τεχνολογίας και στην αγορά αλλά όχι έτσι απλά. Πιο συγκεκριμένα, δημιουργείται ένας συνεταιρισμός εταιριων, η Open Handset Alliance, όπου αποτελείται απο εκπροσώπους τόσο της Google όσο και της Texas Instruments, της Intel, της T-Mobile και της Sprint Nextel, της HTC, της Samsung, της Qualcomm (κατασκευαστές επεξεργαστών) και άλλους με σκοπό να μετατρέψουν το Android σε μια πλατφόρμα ανοιχτού κώδικα που θα έχει ενσωματωμένο το λογισμικό Linux και σιγά-σιγά να κυκλοφορήσουν στην αγορά τα πρώτα να κυκλοφορήσουν τα πρώτα smartphones με Android. Εταιρίες όπως η HTC, η Motorola και η LG, ανακοινώνουν ότι πρόκειται να υπάρξουν στην αγορά τα smartphones, όπου θα διαθέτουν λειτουργικό σύστημα Android αλλά θα είναι κατάλληλα προσαρμοσμένα στις ανάγκες του αγοραστικού τους κοινού τόσο όσο αφορά τον καλοπισμό σε διάφορα σχήματα και μεγέθη όσο και στις δυνατότητες που μπορεί να του προσφέρει μέσω των πολλαπλών εφαρμογών που θα μπορούν να υλοποιηθούν και να υποστηριχθούν από αυτά προσαρμοσμένες σε οποιαδήποτε ανάγκη και αν έχει.

Η HTC κυκλοφορεί ένα χρόνο μετά το πρώτο smartphone με λειτουργικό Android το οποίο ονομαζόταν HTC Dream. Στην συνέχεια ακολούθησαν και οι υπόλοιπες εταιρείες καθώς και δεκάδες άλλες που εμπιστευτικάν το Android ως λειτουργικό και κατάφεραν να το κάνουν το κυρίαρχο λειτουργικό αυτή τη στιγμή στην αγορά. Η συνεργασία με την Samsung, η οποία είναι η πρώτη εταιρεία στις πωλήσεις κινητών παγκοσμίως αυτή την στιγμή κατάφερε και εδραίωσε το Android και η εξάπλωση του οφείλεται κατα κυριο λόγο σε αυτήν. Από εκεί και μετά η ιστορία είναι λίγο πολύ γνωστή σε όλους μας.

### **3.3 Βασικά Χαρακτηριστικά**

Οι συσκευές που χρησιμοποιούν Android είναι multimedia, δηλαδή παρέχουν την δυνατότητα να αναπαράγουν πολλαπλά μέσα, καθώς και multitasking δηλαδή δίνουν τη δυνατότητα εκτέλεσης πολλών εφαρμογών ταυτόχρονα, όπως παραδειγματος χάριν οι χρήστες των συσκευών να σερφάρουν στο ίντερνετ ενώ παράλληλα να ακούν τραγούδια και ταυτόχρονα να απαντούν σε ένα SMS χωρίς να χρειαστεί να κλείσουν καμία εφαρμογή ή να χάσουν τη σελίδα που επισκεφτονται στο διαδουκτιο).



Το Android παρέχει την δυνατότητα κάλυψης ανάγκών ενός απαιτητικού χρήστη σε επίπεδο υψηλό όσο αφορά το web browsing καθώς είναι ταχύτατο, μπορεί να υποστηρίξει τόσο flash όσο και την χρήση όποιου browser προτιμά. Επίσης, όλες οι συσκευές Android διαθέτουν GPS και Wi-fi, ανεξάρτητως κόστους της συσκευής και δικαιολογώντας έτσι τον βασικό λόγο «γεννησης» του λειτουργικού συστήματος Android, που είναι η εύκολη γρήγορη και ανεμπόδιστη πρόσβαση στο διαδίκτυο σε συνδιασμό με πάμπολλές εφαρμογές (apps), όπως αναζήτηση, chat, e-mail και χάρτες και επιτρέπουν στο χρήστη να μένει διαρκώς δικτυωμένος και ενημερωμένος.

Ακόμη, το Android δίνει τη δυνατότητα στους χρήστες να προσθέσουν widgets, δηλαδή εικονίδια για την ταχύτερη πρόσβαση στα προγράμματα, τα οποία τοποθετούνται στη home screen του κινητού (launcher).

Βασικό χαρακτηριστικό του Android, επίσης, είναι η ευχαίρεια επιλογής μεταξύ πολλών εφαρμογών που διατηρούν τη συνεχή σύνδεση με κοινωνικά δίκτυα όπως Facebook, MySpace, Twitter και άλλα.

Επιπλέον, εξαιρετικά χρήσιμη είναι η notification bar, καθώς με ένα απλό drag οι χρήστες μπορούν να δούν όλες τις ειδοποιήσεις σχετικά με συσκευή τους, αλλά και τα προγράμματα (applications) που έχουν εγκαταστήσει εάν έχουν π.χ. ενημερώσεις είτε οτιδήποτε άλλο.

Σχετικά με το hardware, οι εξελίξεις στην τεχνολογία είναι ραγδαίες καθώς οι διπύρηννοι και τετραπύρηννοι επεξεργαστές και οι διακεκριμένες GPU είναι πλέον γεγονός.

### 3.4 Οι εκδόσεις του Android

Το Android έχει δημιουργηθεί πάνω στο λειτουργικό σύστημα Linux αλλά προσαρμοσμένο πάνω σε κινητές συσκευές. Ξεκινώντας από το 2008 με την κυκλοφορία του πρώτου Android smartphone μέχρι σήμερα έχουν βγει αρκετές εκδόσεις όπως είναι φυσικό και υπάρχει το χαρακτηριστικό γνώρισμα ότι όλες έχουν πάντα ονόματα κάποιου γλυκού είτε επιδόρπιου. Η ονομασία των εκδόσεων αυτών πχ Frogo, Gingerbread, Honeycomb, Ice Cream Sandwich, Jelly Bean, KitKat, κλπ.



Χρονολογική σειρά των εκδόσεων του Android

Ας δούμε τις κυριότερες εκδόσεις Android που κυκλοφόρησαν από το 2008 μέχρι σήμερα με χρονολογική σειρά.

#### 3.4.1 Android 1.0 και Android 1.1

Με την κυκλοφορία του πρώτου Android smartphone, του HTC Dream έχουμε και την κυκλοφορία της έκδοσης 1.0. Ενσωματωμένες εφαρμογές που υπήρχαν ήταν το ξυπνητήρι, ο περιηγητής, η αριθμομηχανή, η κάμερα, το email, οι χάρτες και η μουσική καθώς και κάποιες άλλες.

#### 3.4.2 Android 1.5 Cupcake

Την άνοιξη του 2009 κυκλοφόρησε. Και ουσιαστικά έχουμε και την αρχή των ονοματων με γλυκά.

Ήταν η έκδοση με την οποία έχουμε την υποστήριξη των widgets και ως νέα χαρακτηριστικά του ήταν η εγγραφή video και playback σε μορφή MPEG-4 και 3GP και τα εφέ κίνησης κατά την περιήγηση στις διαφορετικές οθόνες. Η έκδοση 1.5 (CupCake) παρέχει τη δυνατότητα αυτόματης σύνδεσης ακουστικών headset σε συγκεκριμένη απόσταση, ανέβασμα εικόνων στο Picasa και βίντεο στο YouTube κατευθείαν από την κινητή συσκευή του χρήστη, ενώ παρέχει εικονικό πληκτρολόγιο με πρόβλεψη λέξεων και νέα widgets για την αρχική οθόνη. Διαθέτει κάμερα 5MP με αυτόματη εστίαση και κάρτα μνήμης micro SD. Τέλος, το CupCake (1.5) «τρέχει» το HTC Hero με οθόνη αφής TFT-LCD, 3,2", ανάλυσης 320x480p (HVGA).

#### 3.4.3 Android 1.6 Donut

Το Σεπτέμβριο του 2009 κυκλοφόρησε. Εννοείτε ότι συμπεριελάμβανε νέες βελτιώσεις όπως ευκολότερη αναζήτηση και δυνατότητα προεπισκόπησης εφαρμογών σε όσες συσκευές είχαν Google Play, δείκτης χρήσης της μπαταριάς και αυτόματη περιστροφή οθόνης. Με το Donut δίνεται έμφαση στην αναζήτηση από την αρχική οθόνη με bookmarks, ιστορικό, επαφές κ.ά. αλλά και στη φωνητική αναζήτηση, ενώ υποστηρίζονται και οθόνες αναλύσεων WVGA. Τηλέφωνα της έκδοσης αυτής είναι το LG GT 540, το οποίο υποστηρίζει οθόνη αναλύσεων WVGA και επιπλέον προσφέρει ευκολία στην εύρεση των επαφών, διαθέτει qwerty πληκτρολόγιο για γρήγορη αποστολή SMS και 3G και Wi-Fi για να είναι ο χρήστης συνδεδεμένος στο διαδύτιο όπου και να βρίσκεται, και το Sony Ericsson X10 με επεξεργαστή Snapdragon 1GHz της Qualcomm και οθόνη αφής 4", 854x480pixels. Στην πίσω όψη βρίσκεται η κάμερα 8,1MP και το LED Flash κάτω ακριβώς από τον φακό της.





Γραφικό περιβάλλον του Android version 1

#### 3.4.4 Android 2.0 Eclair

Τον Οκτώβριο του 2009 κυκλοφόρησε, μόλις ένα μήνα μετά το Donut. Συμπεριελάμβανε αρκετές μικρές βελτιώσεις, όπως Bluetooth 2.1, κινούμενο φόντο στην οθόνη του home, και πληκτρολόγιο με έξυπνο λεξιλόγιο που μαθαίνει ανάλογα με την χρήση των λέξεων.

Με το όνομα Éclair έχουμε και επόμενες επίσης γλυκές εκδόσεις, τις Éclair 2.0 και Éclair 2.1 που προχωρούν ακόμη πιο πολύ, διαθέτοντας υποστήριξη HTML5, νέο browser UI, Google Maps 3.1.2, ψηφιακό ζουμ, ενσωματωμένη υποστήριξη για flash στην κάμερα, βελτιωμένο εικονικό πληκτρολόγιο, δυνατότητα αντίληψης multi-touch, live wallpapers, και bluetooth 2.1.

Το Motorola Droid είναι ένας εκπρόσωπος της έκδοσης 2.0 (Éclair), με επεξεργαστή Arm Cortex A8 550MHz. Ο σχεδιασμός του είναι slide με πλήρες qwerty πληκτρολόγιο και οθόνη αφής 3,7", ανάλυσης 480x854p. Επίσης, έχουμε το Samsung galaxy S, με οθόνη αφής 4.0" Super AMOLED που «τρέχει» Android 2.1 (Éclair) με επεξεργαστή 1GHz, εσωτερική μνήμη 8GB, Wi-Fi και bluetooth 3.0, παρέχει ταχύτητα, χώρο και απαιτούμενες δυνατότητες σύνδεσης. Επιπλέον, έχουμε το HTC Wildfire με οθόνη αφής 3,2" QVGA, το οποίο υποστηρίζει Adobe flash με άνετη πλοήγηση στο διαδίκτυο και παρέχει λήψη online βίντεο και παιχνιδιών. Διαθέτει κάμερα στα 5MP με αυτόματη εστίαση και LED φλας.

#### 3.4.5 Android 2.2-2.2.3 Froyo

Το Μάιο του 2010 κυκλοφόρησε και το ακρόνυμο της έκδοσης αποτελεί συντόμηση της φράσης "Frozen Yogurt" (παγωμένο γιαούρτι). Είναι η πρώτη έκδοση του Android που υποστήριζε Adobe Flash. Μερικές από τις βελτιώσεις ήταν σύνδεση μέσω USB και Wi-Fi hotspot, η γρήγορη εναλλαγή γλώσσας κατά την πληκτρολόγηση και η δυνατότητα απενεργοποίησης της λειτουργίας δικτύου δεδομένων.

Η έκδοση 2.2 (FroYo-Frozen Yogurt) αναβάθμισε αισθητά την ταχύτητα του OS, αλλά και τη γενικότερη απόδοση, παρέχει υποστήριξη Adobe flash 10.1 και επιλογή εγκατάστασης εφαρμογών στην κάρτα μνήμης, διαθέτει Market με δυνατότητα αυτόματων updates, ενσωματώνει τον Chrome V8 JavaScript στα browsers applications. Στη FroYo

βρίσκουμε και τη δυνατότητα χρήσης της συσκευής για διαμοιρασμό ίντερνετ μέσω Wi-Fi σε άλλες συσκευές (tethering).

Εκπρόσωπος της έκδοσης αυτής είναι το LG Optimus 2x που διαθέτει οθόνη αφής 4", λειτουργικό Android Frogo και διπύρνηο επεξεργαστή Nvidia Tegra 2 με ισχυρή κάμερα 8MP, δυνατότητα εγγραφής και αναπαραγωγής βίντεο full HD και συνδεσιμότητα HDMI και DLNA. Ένας άλλος εκπρόσωπος είναι το HTC Desire Z με Android Frogo και οθόνη αφής 3,7", ανάλυσης WVGA (480x800p), συρόμενο Qwerty πληκτρολόγιο για γρήγορα e-mail, 5MP κάμερα και δέκτη GPS. Χάρη στον ενσωματωμένο Adobe Flash Player και προβάλλει κάθε ιστοσελίδα.

### 3.4.6 Android 2.3-2.3.7 Gingerbread

Τον Δεκέμβριο του 2010 Κυκλοφόρησε η πιο πολυπληθής έκδοση του Android. Ήταν πολύ πιο γρήγορο και εύχρηστο από τις προηγούμενες εκδόσεις και έδινε στους δημιουργούς εφαρμογών μεγαλύτερες δυνατότητες. Οι βελτιώσεις περιλάμβαναν υποστήριξη πολλών καμερών στην συσκευή όπως και μεγαλύτερης ανάλυσης οθόνη.

Η GingerBread (2.3), που κυκλοφόρησε το Δεκέμβριο του 2010, υποστηρίζει πλέον πολύ μεγάλα μεγέθη οθονών και αναλύσεων, διαθέτει επανασχεδιασμένο multi-touch πληκτρολόγιο, προεγκατεστημένη υποστήριξη για τηλεφωνικές κλήσεις μέσω ίντερνετ (VoIP), download manager για κατέβασμα μεγάλων αρχείων, λειτουργίες copy-paste σε όλο το λειτουργικό, καθώς και προεγκατεστημένη υποστήριξη για πολλαπλές κάμερες.

Μεγάλος εκπρόσωπος της αποτελεί το Sony Xperia Play, συσκευή προσανατολισμένη στο gaming, «τρέχει» GingerBread (2.3) με επεξεργαστή Scorpion ARM7 και ταχύτητα στα 1GHz. Η οθόνη του είναι 4 ιντσών με ανάλυση 480x854p και η εσωτερική του μνήμη 400MB. Άλλοι εκπρόσωποι είναι τα HTC Cha Cha και HTC Salsa με GingerBread (2.3) που ενσωματώνουν ένα εξειδικευμένο Facebook πλήκτρο, για πρόσβαση με ένα άγγιγμα, στην υπηρεσία του Facebook μέσα από την εμπειρία HTC Sense.



Γραφικό περιβάλλον Android version 2

### 3.4.7 Android 3.0-3.2 Honeycomb

Τον Φεβρουάριο του 2011 κυκλοφόρησε και ήταν διαθέσιμη μόνο για tablets. Οι βελτιώσεις της περιλαμβάνουν γρήγορη πρόσβαση σε χαρακτηριστικά της κάμερας, καλύτερο πληκτρολόγιο κατάλληλο για μεγάλες οθόνες, εκτέλεση πολλαπλών λειτουργιών και εύκολη μετάβαση από την μια στην άλλη. Η έκδοση Honeycomb (3.1) προσέθεσε την επιλογή να μεταφέρεται περιεχόμενο απευθείας από συσκευές USB, ενώ τέλος η έκδοση 3.2 προσέθεσε διάφορες δυνατότητες και ευκολίες για χρήστες και developers όπως τη μεταφορά αρχείων από κάρτες SD και δυνατότητα Zoom to Fill.

Αυτή τη στιγμή στην αγορά υπάρχουν αρκετές ταμπλέτες που τρέχουν Android Honeycomb όπως το Motorola Xoom, το Samsung Galaxy Tab 10.1, το επερχόμενο tablet της Sony, Sony Tablet S, το Asus Eee Pad Transformer και το Toshiba AT200 το οποίο μάλιστα είναι εξοπλισμένο με την τελευταία έκδοση του Android Honeycomb 3.2. Η πρώτη συσκευή που έτρεχε σε Honeycomb ήταν το tablet Motorola Xoom.



Γραφικό περιβάλλον Android 3.

### 3.4.8 Android 4.0-4.0.2 Ice Cream Sandwich

Τον Οκτώβρη του 2011 κυκλοφόρησε και έφερε πληθώρα αλλαγών στο λειτουργικό σύστημα. Η δυνατότητα χρήσης “μαλακών” κουμπιών δηλαδή των κουμπιών πάνω στην οθόνη (πίσω, αρχική, κλπ) είναι πλέον πραγματικότητα καθώς μέχρι τότε όλα τα κινητά είχαν εξωτερικά κουμπιά για αυτές τις λειτουργίες. Μερικές άλλες δυνατότητες ήταν καλύτερη χρήση των φωνητικών εντολών, το Face Unlock, βελτίωση της ταχύτητας απόκρισης και αναδιαμόρφωση του περιβάλλοντος χρήσης. Αυτή η έκδοση διαθέτει καλύτερο web browser με tabs, ανανεωμένο γραφικό περιβάλλον με αρκετά 3D στοιχεία και ανανεωμένο εικονικό πληκτρολόγιο. Δίνει ειδική έκδοση του Gmail για tablets, δυνατότητα βιντεοκλήσεων μέσω εφαρμογής Google Talk, ανανεωμένη έκδοση Google Maps και βελτιστοποιημένη εφαρμογή για ανάγνωση Google e-books.

### 3.4.9 Android 4.1-4.3.1 Jelly Bean

Τον Ιούνιο του 2012 κυκλοφόρησε και αποτελεί την καλύτερη έκδοση του Android μέχρι σήμερα. Το περιβάλλον χρήσης και η απόκρισή του είναι πιο γρήγορη και καλοφτιαγμένη από ποτέ ενώ περιλαμβάνει πάρα πολλές μικρές βελτιώσεις σε όλο το σύστημα, όπως για παράδειγμα στην κάμερα και στην χρήση φωνής για υπαγόρευση κειμένου. Η έκδοση 4.1 Jelly Bean του OS σκαρφάωσε στο 28,4% από 25% τον προηγούμενο μήνα και ξεπέρασε το ποσοστό του Ice Cream Sandwich, το οποίο ανέρχεται

πλέον στο 27,5%. Παρόλα αυτά, η έκδοση 2.3 Gingerbread του λειτουργικού παραμένει στην πρώτη θέση και είναι εγκατεστημένη στο 38,5% των Android συσκευών.

Το Jelly Bean εφαρμόστηκε για πρώτη φορά στο tablet Google Nexus 7 ενώ η έκδοση Android 4.2 πρωτοεμφανίστηκε στα Nexus 4 και Nexus 10.

#### 3.4.10 Android 4.4-4.4.4 Kit Kat

Τον Σεπτέμβριο του 2013 κυκλοφόρησε. Αν και αρχικά ήταν να ονομαστεί "Key Lime Pie" ("KLP") κωδική ονομασία, το όνομα άλλαξε καθώς πολύ λίγοι άνθρωποι ξέρουν πραγματικά τη γεύση αυτής της πίτας.

Το KitKat έκανε το ντεμπούτο του στο Nexus της Google 5 και έχει βελτιστοποιηθεί για να τρέχει σε ένα μεγαλύτερο εύρος συσκευών από τις προηγούμενες εκδόσεις του Android, αφού έχει ως συνιστώμενη ελάχιστη μνήμη RAM 512 .

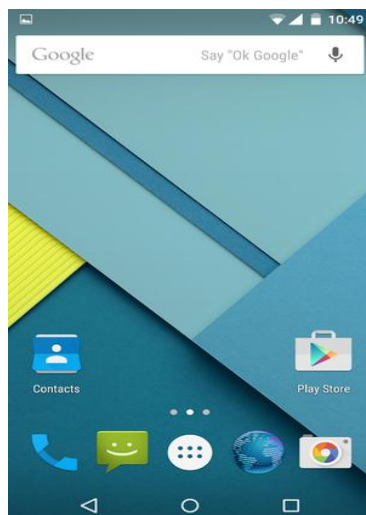


Γραφικό περιβάλλον 4

#### 3.4.11 Android 5.0 "Lollipop"

Τον Ιούνιο του 2014 ανακοινώθηκε από την Google η νέα έκδοση του Android αλλά θα είναι διαθέσιμη τον τον Νοέμβριο του 2014 για επιλεγμένες συσκευές που τρέχουν Android, συμπεριλαμβανομένων τις συσκευές Nexus.

Η Lollipop ως βασική αλλαγή που θα παρουσιάσει είναι ένα επανασχεδιασμένο περιβάλλον εργασίας χρήστη χτισμένο γύρω από μία διαδραστική σχεδιαστική γλώσσα που αποκαλείται ως «υλικό σχεδιασμού». Επιπλέον βελτιώσεις του συστήματος θα είναι ότι το notifications system θα επιτρέπει κοινοποιήσεις που θα μπορούν να προσπελαστούν από την lockscreen, και να εμφανίζονται μαζί με εφαρμογές ως banner πάνω την κορυφή της οθόνης. Εσωτερικές αλλαγές που θα γίνουν επίσης στην πλατφόρμα, πιο συγκεκριμένα, το Android Runtime (ART) θα αντικαταστήσει το Dalvik με μία πιο βελτιωμένη έκδοση την γνωστή ως Project Volta, η οποία θα παρέχει καλύτερη απόδοση των εφαρμογών πράγμα το οποίο σημαίνει ότι θα υπάρξουν και αλλαγές που αποσκοπούν στη βελτίωση και βελτιστοποίηση της χρήσης της μπαταρίας.



Γραφικό περιβάλλον 5

Όπως παρατηρούμε και στις εικόνες κατά το πέρας των εκδόσεων, οι διαφορές του γραφικού περιβάλλοντος είναι αισθητές πόσο μάλιστα των δυνατοτήτων που μπορεί να παρέχει σε ένα χρήστη.

### 3.5 Επίπεδα του λειτουργικού

Το Android είναι στο περίπου χωρισμένο σε 4 επίπεδα.

#### 3.5.1 Linux Kernel

Ο Linux Kernel είναι ο πυρήνας στον οποίο βασίζεται το Android και βρίσκεται στο χαμηλότερο επίπεδο. Παρέχει τους Drivers τους οποίους χρειάζεται για να τρέξει το σύστημα, όπως της οθόνης, της κάμερας κ.α.

#### 3.5.2 Libraries

Οι βιβλιοθήκες(Libraries) είναι όλος ο κώδικας που περιέχει το Android OS και βοηθαεί ώστε να υπάρξει η συσκευή και να παρέχει όλα τα χαρακτηριστικά και δυνατότητες που επιθυμεί ο χρήστης, π.χ η Webkit βιβλιοθήκη παρέχει λειτουργίες για το διαδικτυακό σερφάρισμα, η SQLite βιβλιοθήκη παρέχει υποστήριξη έτσι ώστε μια εφαρμογή να χρησιμοποιήσει την αποθήκευση δεδομένων κ.α.

#### 3.5.3 Android Runtime

Το Android Runtime παρέχει ένα σύνολο βασικών βιβλιοθηκών που επιτρέπουν στους προγραμματιστές να γράψουν εφαρμογές χρησιμοποιώντας JAVA και βρίσκεται στο ίδιο επίπεδο με τις βιβλιοθήκες. Επίσης περιλαμβάνει την Dalvik virtual machine, που επιτρέπει κάθε εφαρμογή να τρέξει την δικιά της εργασία, μαζί με την δικιά της ξεχωριστή Dalvik virtual machine. Η Dalvik είναι μια εξειδικευμένη virtual machine, ειδικά διαμορφωμένη για κινητές συσκευές που έχουν περιορισμένη μνήμη και ισχύ.

#### 3.5.4 Application Framework

Το Application Framework παρέχει διάφορες δυνατότητες του Android στους προγραμματιστές των εφαρμογών ώστε να τις χρησιμοποιήσουν στις εφαρμογές τους.

### 3.5.5 Applications

Το Applications είναι το πιο υψηλό επίπεδο, είναι εφαρμογές που έρχονται μαζί με την Android συσκευή του χρήστη (όπως τηλέφωνο, επαφές, μουσική κ.α.), όπως επίσης εφαρμογές που κάνει εγκατάσταση σύμφωνα με τις ανάγκες του. Οποιαδήποτε εφαρμογή που εγκαθιστά ο χρήστης εκτελείται σε αυτό το επίπεδο.

## 3.6 Ασφάλεια στο Android

Η ασφάλεια του συστήματος παίζει καθοριστικό ρόλο καθώς υπάρχουν καθημερινά όλο και περισσότερες εφαρμογές που έχουν άμεση σχέση είτε θεματολογία σχετικά με τα προσωπικά και ευαίσθητα δεδομένα του κάθε χρήστη και αυτό είναι λογικό ότι εγκυμονεί κινδύνους.

Δυστηχώς οι απλοι χρήστες καθώς είναι ενθουσιασμένοι με τις δυνατότητες που μπορεί να τους παρέχει το κινητό τους όμως δεν γνωρίζουν βασικές συνέπιες που μπορούν να διεξαχθουν όπως ότι με την μόλυνση του κινητου τους από ένα κακόβουλο πρόγραμμα μπορεί το αγαπημένο τους smartphone να στείλει γραπτά μηνύματα σε αριθμούς με υψηλές χρεώσεις και με τον τρόπο αυτό να κλέβει χρήματα από τον λογαριασμό τους είτε ότι μπορεί να μετατρέπει σε συσκευή – κατάσκοπο, δίνοντας σε απατεώνες πληροφορίες προσωπικών δεδομένων όπως τον ιδιοκτήτη του, τις κλήσεις του, τα μηνύματα του, τους κωδικους πρόσβασης του σε κοινωνικά δίκτυα, τους λογαριασμούς e-pay του, κλπ.



*Security of smartphones*

Από την στιγμή που στην εποχή μας οι περισσότεροι καταναλωτές και οι περισσότερες εταιρείες επενδύουν σε smartphones και tablets με λειτουργικό σύστημα Android, έχουν πραγματοποιηθεί έρευνες σχετικά με το θέμα της ασφαλειας.

Σε μία σχετική έρευνα παρατηρήθηκε ότι σε ποσοστό πάνω από το 50% των χρηστών και κατόχων παραλληλα Android smartphones και tablets στην Ευρώπη δε χρησιμοποιούν λογισμικό ασφαλείας για την προστασία των συσκευών τους είτε αγνόουν την ανάγκη κάλυψης των δεδομένων τους απο ψηφιακούς εγκληματίες που χρησιμοποιούν σκουλικάκια για να διωσδισουν στα προσωπικα δεδομενα άλλων απλών χρηστών και να κλέψουν την ιδιωτικότητα των εκάστωτε θυμάτων με αυτό τον τρόπο.

Σύμφωνα με μία πρόσφατη έρευνα της βιομηχανίας που πραγματοποιήθηκε το 2013, επισημάνθηκε ότι κατα ποσοστο 60% αύξηση των μολύνσεων τύπου malware των συστημάτων με Android γιατί οι συσκευές είναι ευάλωτες και εκτεθειμένες σε κινδύνους από γνωστά exploits καθώς πάνω από το 90% των χρηστων δεν διαθέτουν την πιο πρόσφατη έκδοση του Android στην φοριτή συσκευή του.



Η υπηρεσία cloud της Kaspersky Lab, Kaspersky Security Network, η οποία συγκεντρώνει στοιχεία για επείγουσες ψηφιακές απειλές, μας ενημερώνει ότι το 99% των σημερινών δειγμάτων malware που στοχεύουν φορητές συσκευές έχουν αναπτυχθεί για την πλατφόρμα Android και αυτή η απειλή αυξάνεται ραγδαίως καθώς οι ειδικοί της εντοπίσαν 35.000 κακόβουλα δείγματα συνολικά για το 2012, ενώ μόνο το πρώτο εξάμηνο του 2013 τα δείγματα ξεπερνούσαν τα 47.000.

### **3.7 Firmware σε συσκευές Android**

Το firmware είναι το «μυαλό», αφού ελέγχει τα κυκλώματα του τηλεφώνου για να λειτουργήσουν τα μέρη του τηλεφώνου καθώς στέλνει σε αυτά εντολές. Ελέγχει το hardware της συσκευής. Είναι το BIOS της κινητής συσκευής. Βγαίνουν διάφορες αναβαθμίσεις για να διορθώσουν Bugs και να προσθέσουν καινούργιες ιδιότητες. Δίνονται κωδικές ονομασίες αποτελούμενες από νούμερα και γράμματα, υποδηλώνοντας συσκευή, ημερομηνία έκδοσης και πολλά άλλα. Δεν είναι απίθανο να παρατηρηθεί σε 2 ίδιες συσκευές να έχουν διαφορετικό Firmware αλλά να βασίζονται στην ίδια έκδοση του Android.

#### **3.7.1 Λογαριασμός root**

Root λέμε το λογαριασμό που έχει πρόσβαση σε ΟΛΑ τα αρχεία ενός συστήματος Linux (το Android βασίζεται στο Linux) και μπορεί να κάνει αλλαγές που ένας κανονικός χρήστης δε μπορεί. Με αυτόν τον τρόπο μπορεί με λίγα λόγια ο χρήστης να έχει τον πλήρη έλεγχο του συστήματος, να διορθώνει κάποιο πρόβλημα και να αλλάζει δικαιώματα σε αρχεία. Φυσικά το root έρχεται με τους κινδύνους ενός administrator account, διότι αν ο χρήστης που είναι root είναι άσχετος μπορεί να σβήσει πράγματα που δεν πρέπει αφού έχει τον πλήρη έλεγχο.

Πράγματα που μπορεί να κάνει ένας χρήστης ως root στη συσκευή του περιλαμβάνουν : εγκατάσταση extra εφαρμογών που χρειάζονται πρόσβαση στα αρχεία του συστήματος (π.χ. εφαρμογές για πλήρες backup), αφαίρεση εφαρμογών του συστήματος που δε θέλει να έχει, μεταφορά εφαρμογών στην κάρτα SD, διόρθωση κάποιων προβλημάτων που έχουν άμεση σχέση με τα αρχεία του συστήματος (εξαρτάται τη συσκευή), διαφορετικά εικονίδια και τροποποίηση κατά βούληση ολόκληρου του γραφικού περιβάλλοντος, καλύτερη διαχείριση μνήμης, πρόσβαση σε εφαρμογές που είναι φραγμένες στην Ελλάδα, πχ Gmail/Google Earth/Google Maps με τις αναβαθμίσεις τους, development, δημιουργία και φόρτωση custom ROMs, reboot εκ των έσω, overclock undervolt, etc, tethering σε συσκευές που δεν έχουν, όχι πια ενοχλητικές διαφημίσεις στα free apps, εγκατάσταση παιχνιδιών που προορίζονται αποκλειστικά για Tegra Android συσκευές, σε όλες τις Android συσκευές που έχουν υψηλά χρονισμένο επεξεργαστή(800mhz +), επανάκτηση αρχείων που σβήστηκαν κατά λάθος, ξεκλείδωμα από τον πάροχο, και πολλά άλλα.

Αναλόγως την συσκευή, σε κάποιες συγκεκριμένες η απόκτηση πρόσβασης σε επίπεδο root προκαλεί αυτόματα την επαναφορά εργοστασιακών. Στη συντριπτική πλειοψηφία των Android συσκευών είναι μια ακίνδυνη διαδικασία και μάλιστα σε μερικές γίνεται με την εγκατάσταση μιας απλής εφαρμογής.

## 3.8 Η γλώσσα προγραμματισμού Java

Η ανάπτυξη εφαρμογών στο Android γίνεται στη γλώσσα Java.



*Java + Android*

### 3.8.1 Java

Η Java είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού που σχεδιάστηκε από την εταιρεία πληροφορικής Sun Microsystems. Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι ότι τα προγράμματα που είναι γραμμένα σε Java τρέχουν ακριβώς το ίδιο σε Windows, Linux, Unix και Macintosh (χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα δηλαδή προσφέρει η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας.

### 3.8.2 Τα χαρακτηριστικά της Java

Ένα από τα βασικά πλεονεκτήματα της Java εν συγκρίσει με των υπολοίπων περισσότερων γλωσσών είναι η ανεξαρτησία που διαθέτει όπως και προαναφεραμε εισαγωγικά προλίγου. Για να επιτευχθεί ο στόχος αυτός όμως, χρειάστηκε να βρεθεί κάποιος τρόπος έτσι ώστε τα προγράμματα που ήταν γραμμένα σε Java να μπορούν να είναι «κατανοητά» από κάθε υπολογιστή ανεξάρτητα από το είδος του επεξεργαστή (Intel x86, IBM, Sun SPARC, Motorola) αλλά και του λειτουργικού συστήματος (Windows, Unix, Linux, BSD, MacOS). Ο λόγος που συμβαίνει αυτό είναι ότι κάθε κεντρική μονάδα επεξεργασίας κατανοεί διαφορετικό κώδικα μηχανής. Κατά συνέπεια ο συμβολικός κώδικας (assembly) που μεταφράζεται και εκτελείται σε Windows είναι διαφορετικός από αυτόν που μεταφράζεται και εκτελείται σε έναν υπολογιστή Macintosh. Η λύση προήλθε με την ανάπτυξη της Εικονικής Μηχανής (VirtualMachine ή VM ή EM στα ελληνικά). Σύμφωνα με την Εικονική Μηχανή όταν γραφτεί κάποιο πρόγραμμα σε Java, στη συνέχεια μεταγλωττίζεται μέσω του μεταγλωττιστή javac, ο οποίος παράγει έναν αριθμό από αρχεία .class (κώδικας byte ή bytecode). Ο κώδικας byte είναι η μορφή που παίρνει ο πηγαίος κώδικας της Java όταν μεταγλωττιστεί. Όταν χρειαστεί να εκτελεστεί η εφαρμογή σε ένα μηχάνημα, το JavaVirtualMachine που πρέπει να είναι εγκατεστημένο σε αυτό θα αναλάβει να διαβάσει τα αρχεία .class. Στη συνέχεια τα μεταφράζει σε γλώσσα μηχανής που να μπορούν να υποστηριχτεί από το λειτουργικό σύστημα και τον επεξεργαστή, έτσι ώστε να εκτελεστεί (να σημειωθεί εδώ ότι αυτό συμβαίνει με την παραδοσιακή Εικονική Μηχανή (VirtualMachine). Οι πιο σύγχρονες εφαρμογές της Εικονικής Μηχανής μπορούν και μεταγλωττίζουν εκ των προτέρων τμήματα bytecode απευθείας σε κώδικα μηχανής (εγγενή κώδικα ή nativecode) με αποτέλεσμα να βελτιώνεται η ταχύτητα. Χωρίς αυτό δε θα ήταν δυνατή η εκτέλεση λογισμικού γραμμένου σε Java. Αξίζει να σημειωθεί ότι η JVM είναι



λογισμικό που εξαρτάται από την πλατφόρμα, δηλαδή για κάθε είδος λειτουργικού συστήματος και αρχιτεκτονικής επεξεργαστή υπάρχει διαφορετική έκδοση του. Έτσι υπάρχουν διαφορετικές JVM για Windows, Linux, Unix, Macintosh, κινητά τηλέφωνα, παιχνιδιομηχανές κλπ.

Οτιδήποτε θέλει να κάνει ο προγραμματιστής (ή ο χρήστης) γίνεται μέσω της εικονικής μηχανής. Εξαιτίας αυτού υπάρχει μεγαλύτερη ασφάλεια στο σύστημα γιατί η εικονική μηχανή είναι υπεύθυνη για την επικοινωνία χρήστη - υπολογιστή. Γενικά ισχύει ότι ο προγραμματιστής δεν μπορεί να γράψει κώδικα ο οποίος θα έχει καταστροφικά αποτελέσματα για τον υπολογιστή γιατί η εικονική μηχανή θα τον ανιχνεύσει και δε θα επιτρέψει να εκτελεστεί. Παρομοίως, στην άλλη μεριά ούτε ο χρήστης μπορεί να κατεβάσει «κακό» κώδικα από το δίκτυο και να τον εκτελέσει. Αυτό είναι ιδιαίτερα χρήσιμο για μεγάλα καταναμημένα συστήματα όπου πολλοί χρήστες χρησιμοποιούν το ίδιο πρόγραμμα συγχρόνως.

Η ύπαρξη του συλλέκτη απορριμμάτων (GarbageCollector) αποτελεί ακόμα μία ιδέα που βρίσκεται πίσω από τη Java. Συλλογή απορριμμάτων είναι μία κοινή ονομασία που χρησιμοποιείται στον τομέα της πληροφορικής για να δηλώσει την ελευθέρωση τμημάτων μνήμης από δεδομένα που δε χρειάζονται και δε χρησιμοποιούνται άλλο. Στη Java η απελευθέρωση μνήμης είναι αυτόματη και γίνεται μέσω του συλλέκτη απορριμμάτων. Η εικονική μηχανή είναι και πάλι υπεύθυνη για αυτό και μόλις «καταλάβει» ότι ο σωρός (heap) της μνήμης (στη Java η συντριπτική πλειοψηφία των αντικειμένων αποθηκεύονται στο σωρό σε αντίθεση με τη C++ όπου αποθηκεύονται κυρίως στη στοίβα) κοντεύει να γεμίσει ενεργοποιεί το συλλέκτη απορριμμάτων. Συνεπώς ο προγραμματιστής δε χρειάζεται να ανησυχεί για το πότε και αν θα ελευθερώσει ένα συγκεκριμένο τμήμα της μνήμης, ούτε και για σφάλματα δεικτών. Αυτό είναι ιδιαίτερα σημαντικό γιατί είναι κοινά τα σφάλματα προγραμμάτων που οφείλονται σε λανθασμένο χειρισμό της μνήμης.

Παρόλο που η εικονική μηχανή προσφέρει όλα αυτά τα πλεονεκτήματα, η Java αρχικά ήταν πιο αργή σε σχέση με άλλες προγραμματιστικές γλώσσες υψηλού επιπέδου (high-level) όπως η C και η C++. Στο παρελθόν εμπειρικές μετρήσεις είχαν δείξει ότι η C++ μπορούσε να είναι αρκετές φορές γρηγορότερη από την Java. Παρόλα αυτά γίνονται προσπάθειες από τη Sun για τη βελτιστοποίηση της εικονικής μηχανής, ενώ υπάρχουν και άλλες υλοποιήσεις της εικονικής μηχανής από διάφορες εταιρίες (όπως της IBM), οι οποίες μπορεί σε κάποια σημεία να προσφέρουν καλύτερα και σε κάποια άλλα χειρότερα αποτελέσματα. Επιπλέον με την καθιέρωση των μεταγλωττιστών JIT (JustInTime), οι οποίοι μετατρέπουν τον κώδικα byte απευθείας σε γλώσσα μηχανής, η διαφορά ταχύτητας από τη C++ έχει μικρύνει κατά πολύ.

## 3. Google Maps

### 3.1 Εισαγωγή

Η υπηρεσία Google Maps είναι μια διαδικτυακή υπηρεσία χαρτογράφησης που παρέχεται από την Google και τρέχει σε Desktop και Smartphones. Η υπηρεσία επίσης προσφέρει ένα εύρη φάσμα εφαρμογών όπως δορυφορικές εικόνες, οδικούς χάρτες, φωτογραφίες οδών καθώς και λειτουργίες όπως σχεδιασμό διαδρομής, πλοήγησης με το αυτοκίνητο, τα πόδια και το ποδήλατο. Επίσης υποστηρίζει τη χρήση χαρτών από ιστοσελίδες τρίτων μέσω του Google Maps API και πολλά άλλα όπως εγγραφή τοποθεσίας επιχείρησης και οργανισμών στους χάρτες της. Η υπηρεσία/βάση δεδομένων δεν ενημερώνεται σε πραγματικό χρόνο αλλά γίνεται συνεχής αναβάθμιση των εικόνων και καμία εικόνα δεν είναι παλαιότερη των τριών χρόνων.

Η εφαρμογή Google Maps χρησιμοποιεί παρόμοια προβολή με τη προβολή Mercator και δεν μπορεί να δείξει περιοχές κοντά στο Νότιο και Βόρειο Πόλο. Ένα προϊόν το οποίο επίσης ανήκει στη Google είναι το πρόγραμμα Google Earth το οποίο προσφέρει στο χρήστη πολλές πληροφορίες μέσω μιας εικονικής υδρογείου που συμπεριλαμβάνει και την προβολή πολιικών περιοχών.

Η εφαρμογή Google Maps για κινητά είναι η πιο δημοφιλής και ευρέως διαδεδομένη εφαρμογή για smartphones με ένα εντυπωσιακό ποσοστό της τάξεως του 54% των χρηστών smartphones να το χρησιμοποιούν τουλάχιστον μία φορά και πιο συγκεκριμένα κατά τον μήνα Αύγουστο, όπως έδειξε και μια έρευνα που έγινε το 2013.

### 3.2 Satellite view

Η υπηρεσία Google Maps παρέχει αεροφωτογραφίες και δορυφορικές εικόνες υψηλής ανάλυσης για τις περισσότερες αστικές περιοχές ανά το παγκόσμιο χαρτη που γνωρίζουμε. Οι περισσότερες δορυφορικές εικόνες της Γής είναι το πολύ 3 χρόνων και ενημερώνονται συνεχώς με αποτέλεσμα να παρουσιάζει κάθε νεότερο χαρακτηριστικό που υπάρχει καθώς και ενημερώσεις που έχουν γίνει σε διάφορες υποδομές. Οι κυβερνήσεις πολλών χωρών έχουν διαμαρτυρηθεί για το ενδεχόμενο ότι πιθανοί τρομοκράτες μπορούν να χρησιμοποιούν τις δορυφορικές εικόνες στο σχεδιασμό επιθέσεων με αποτέλεσμα η Google να έχει θολώσει ορισμένους τομείς για θέματα ασφάλεια κυρίως για τις Ηνωμένες Πολιτείες, συμπεριλαμβανομένων του Ναυτικού παρατηρητηρίου των Η.Π.Αμερικής, περιοχή όπου βρίσκεται και η επίσημη κατοικία του Αντιπροέδρου και στο παρελθόν, το Καπιτώλιο των Ηνωμένων Πολιτειών και ο Λευκός Οίκος.

Δεν παρέχονται όλες οι περιοχές με ευκρίνεια. Λιγότερο πυκνοκατοικημένες περιοχές παίρνουν λιγότερες λεπτομέρειες. Η μηχανή χαρτογράφησης της Google

προκάλεσε ένα κύμα ενδιαφέροντος σε δορυφορικές φωτογραφίες, λόγω της εύκολης χρήσης και περιήγησης των διαθέσιμων χαρτών της. Δημιουργήθηκαν ιστοσελίδες οι οποίες διαθέτουν δορυφορικές εικόνες από ενδιαφέρουσα φυσικά και τεχνητά αξιοθέατα, όπως ορατή γραφή στις δορυφορικές εικόνες, διάσημα γήπεδα και μοναδικούς γεωλογικούς σχηματισμούς. Αν και η Google χρησιμοποιεί τον όρο Satellite View, οι περισσότερες από τις εικόνες υψηλής ανάλυσης των πόλεων είναι αεροφωτογραφίες που λαμβάνονται από αεροσκάφη τα οποία πετούν σε ύψος 800-1500 μέτρων και όχι από δορυφόρους ενώ οι υπόλοιπες άλλες εικόνες είναι στην πραγματικότητα από δορυφόρους.

### 3.3 Directions-οδηγίες

Η εφαρμογή Google Maps παρέχει δρομολόγια από ένα Α μέρος σε ένα Β μέρος υπό τον τίτλο "Λήψη Οδηγιών». Με μία γκάμα τρόπων μεταφοράς αναλόγως περιοχής προσφέρει οδηγίες για αυτοκίνητο, μέσα μαζικής μεταφοράς (βλέπε Google Transit παρακάτω), περπάτημα και ποδηλασία. Σε συνδυασμό με την υπηρεσία Google Street View προβλήματα όπως παρκάρισμα, αναστροφές, αλλαγές λωρίδας και μονόδρομοι μπορούν να λυθούν πριν να καθώς ο οδηγός μπορεί να τα γνωρίζει πριν καν εισέλθει στο αυτοκίνητο του.

Οι περισσότερες χώρες της ηπειρωτικής Ευρασίας και της Αφρικής είναι ευρέως καταγραμμένες, όπως το Ηνωμένο Βασίλειο, την Ιρλανδία, τις Καναρίους Νήσους, τη Μάλτα και τη Σρι Λάνκα. Ενώ η Κύπρος, το Χονγκ Κονγκ, το Ισραήλ (συμπεριλαμβανομένων των τμημάτων της Δυτικής Όχθης), την Ιορδανία, τον Λίβανο, και το Μακάο έχει τις κατευθύνσεις διαθέσιμες χωρίς σύνδεση με άλλα κράτη. Μόνο σε δημόσιες υπηρεσίες μετακίνησης είναι διαθέσιμες οδηγίες στη Β. Κορέα. Όλες οι χώρες της ηπειρωτικής Βόρειας και Κεντρικής Αμερικής καλύπτονται συνεχόμενα. Όλες οι κατοικημένες χώρες και εδάφη της Καραϊβικής καλύπτονται, εκτός από Τρινιντάντ και Τομπάγκο, αν και σε γενικές γραμμές δεν υπάρχουν συνδέσεις μεταξύ των νησιών.

83620 Feldkirchen-Westerham, Germany

Suggested routes:

- S12078 and A99** 72.8 km - about 49 mins
- A99** 84.1 km - about 53 mins
- A995** 51.4 km - about 51 mins

RO2 2, 83620 Feldkirchen-Westerham, Germany

72.8 km - about 49 mins

1. Head **northwest** on **Münchener Str/S12078** toward **Glaner Str/RO2** 20.3 km  
Continue to follow S12078
2. Slight **right** to merge onto **A99** 50.4 km toward **A9/Nürnberg**
3. Take **exit 5** toward **Bodenseestraße/B2** 0.4 km
4. Turn **right** onto **Bodenseestraße/B2** 0.9 km
5. Turn **left** to stay on **Bodenseestraße/B2** (signs for **Neuaußing**) 0.8 km  
Destination will be on the right.

B2 290-308, 81249 Munich, Germany

Map data ©2011 Tele Atlas

The image shows a Google Maps interface with a route highlighted in blue. The route starts at Feldkirchen-Westerham and ends in Munich. The directions panel on the left provides a step-by-step guide, including road names and distances. The map on the right shows the geographical context, with major roads and landmarks labeled.

Google maps directions

Επιπλέον, η Αμερικανική Σαμόα, την Αυστραλία, οι Αζόρες, το Μπρουνέι, το Πράσινο Ακρωτήριο, Κομόρες, Νήσοι Κουκ, οι Νήσοι Φερόε, οι Ηνωμένες Πολιτείες της Μικρονησίας, Φίτζι, Γαλλική Πολυνησία, Γκουάμ, Χαβάη, την Ισλανδία, την Ιαπωνία, τη Μαδαγασκάρη, τις Μαλδίβες, ο Μαυρίκιος, το Μαγιότ, τη Νέα Καληδονία, τη Νέα

Ζηλανδία, το Νιούε, το Παλάου, οι Φιλιππίνες, η Ρεϋνιόν, το Σάο Τομέ και το Πρίνσιπε, τις Σεϋχέλλες, τη Σαμόα, την Ταϊβάν, το Τιμόρ-Λέστε, το Τόνγκα, το Βανουάτου, το Ουόλις και η Φουτούνα καλύπτονται ως αυτόνομες περιοχές, όπως είναι το Νουκ στη Γροιλανδία, τα τμήματα της Ινδονησίας (Μπαλί, Βόρνεο, Τζαβα, Λομποκ, Ματούρα, Σουλαγεσι και τη Σουμάτρα), το Σαμπαχ και το Σάραγουακ της Μαλαισίας, το Σαϊπάν στην Βόρειες Μαριάνες, τμήματα της Παπούα Νέα Γουινέα, τα μέρη των Νήσων Σολομώντος και Δόκοτρα στην Υεμένη. Καλύπτονται όλες οι χώρες της ηπειρωτικής Νότιας Αμερικής. Η Αργεντινή, η Βολιβία, η Βραζιλία, η Χιλή, το Εκουαδόρ, η Παραγουάη, το Περού και η Ουρουγουάη καλύπτονται συνεχόμενα, ενώ η Κολομβία, η Γαλλική Γουιάνα, η Γουιάνα, το Σουρινάμ και η Βενεζουέλα δεν είναι συνδεδεμένα με άλλα κράτη.

### 3.4 Implementation-υλοποίηση

Όπως και πολλές άλλες διαδικτυακές εφαρμογές της Google, η Google Maps χρησιμοποιεί εκτενώς JavaScript . Καθώς ο χρήστης σέρνει το χάρτη, τα τετράγωνα του χάρτη κατεβάζονται από το διακομιστή και εισάγονται στη σελίδα. Όταν ένας χρήστης ψάχνει για μια επιχείρηση, τα αποτελέσματα φορτώνονται στο παρασκήνιο για την εισαγωγή στη πλαϊνή εργαλειοθήκη και χάρτη χωρίς να ξαναφορτώνεται η σελίδα. Τοποθεσίες μπορούν να σημειώνονται από το χρήστη χρησιμοποιώντας μια κόκκινη καρφίτσα στο χάρτη (που αποτελείται από πολλές μερικώς διαφανές εικόνες τύπου png ). Για να επιτευχθεί αυτό χρησιμοποιείται ένα κρυφό IFrame με υποβολή εντύπου για να μπορεί να διατηρεί ιστορικό περιήγησης. Το site χρησιμοποιεί επίσης JSON για τη μεταφορά δεδομένων, αντί για XML, για λόγους απόδοσης. Αυτές οι τεχνικές και οι δύο εμπίπτουν στην ευρεία ομπρέλα Ajax. Το αποτέλεσμα ονομάζεται γλιστερός χάρτης και εφαρμόζεται και σε άλλες εφαρμογές όπως το OpenLayers.

Τον Οκτώβριο του 2011, η Google ανακοίνωσε το MapsGL, μια WebGL έκδοση των Χαρτών με καλύτερες αποδόσεις και την ομαλότερη μετάβαση. Η προηγούμενη έκδοση Google Street View για την Google Maps χρειάζεται Adobe Flash. Αντίθετα η καινούργια έκδοση Google Maps από το 2013 μέχρι και σήμερα δεν χρειάζεται Flash , όπως συνέβηκε και προηγούμενως με πολλές εφαρμογές για smartphones.

### 3.5 Extensibility and customization - Επεκτασιμότητα και παραμετροποίηση

Επειδή η εφαρμογή Google Maps είναι κωδικοποιημένη σχεδόν εξ ολοκλήρου σε JavaScript και XML, ορισμένοι χρήστες εφάρμοζαν αντίστροφη μηχανική στο εργαλείο και παρήγαγαν εκδόσεις στις οποίες η ιστοσελίδα μπορεί να αναβαθμιστεί για να μπορεί ένας χρήστης να εισαγάγει δεδομένα ή να επεκταθεί σε ορισμένες περιοχές. Χρησιμοποιώντας τη μηχανή της Google και χάρτες / δορυφορικές εικόνες που φιλοξενούνται από την Google, τέτοια εργαλεία μπορούν να εισάγουν προσαρμοσμένα εικονίδια, τοποθεσία, συντεταγμένες της θέσης στη εφαρμογή Google Maps. Το εργαλείο τοποθέτησης σεναρίων Greasemonkey παρέχει ένα μεγάλο αριθμό των σεναρίων χρήστη για να προσαρμόσετε τα δεδομένα στην εφαρμογή Google Maps.

Οι συνδυασμοί με ιστοσελίδες ανταλλαγής φωτογραφιών, όπως το Flickr, το οποίο χρησιμοποιείται για τη δημιουργία "Χαρτών απομνημονευμάτων". Χρησιμοποιώντας αντίγραφα των δορυφορικών φωτογραφιών , οι χρήστες έχουν επωφεληθεί από τις δυνατότητες σχολιασμού στην εικόνα για να παρέχουν προσωπικές ιστορίες και πληροφορίες όσον αφορά συγκεκριμένα σημεία ή κτήρια της περιοχής.

### 3.6 Google Maps API

Μετά την επιτυχία αντιστρόφως μηχανοποιημένων σελίδων όπως [chicagocrime.org](http://chicagocrime.org) και [housingmaps.com](http://housingmaps.com), η Google ξεκίνησε το Google Maps API, τον Ιούνιο του 2005 για να επιτρέψει στους προγραμματιστές να ενσωματώσουν το Google Maps στις ιστοσελίδες τους. Πρόκειται για μια δωρεάν υπηρεσία η οποία επί του παρόντος δεν περιέχει διαφημίσεις, αλλά, η Google περιέχει στους όρους χρήσης ότι διατηρεί το δικαίωμα να προβάλλουν διαφημίσεις στο μέλλον.

Με τη χρήση του Google Maps API, είναι δυνατόν να ενσωματώσετε το Google Maps σε μια ιστοσελίδα τρίτου, επί του οποίου σε συγκεκριμένο χώρο συγκεκριμένα δεδομένα μπορούν να επικαλύπτονται. Αν και αρχικά προοριζόταν μόνο για JavaScript API, το Google Maps API επεκτάθηκε για να συμπεριλάβει ένα API για Adobe Flash εφαρμογές (αλλά αυτό έχει αποδοκιμαστεί), μια υπηρεσία για την ανάκτηση στατικών εικόνων χάρτη και διαδικτυακών υπηρεσιών για την εκτέλεση γεωκωδικοποίησης, δημιουργία οδηγιών από ένα σημείο σε ένα άλλο και την απόκτηση σημείων ανύψωσης. Πάνω από 1,000,000 ιστοσελίδες χρησιμοποιούν το Google Maps API, καθιστώντας αυτή ως την πιο ευρέως χρησιμοποιημένη διαδικτυακή εφαρμογή ανάπτυξης API.

Το Google Maps API είναι δωρεάν για εμπορική χρήση, με την προϋπόθεση ότι η περιοχή στην οποία χρησιμοποιείται είναι προσβάσιμη στο κοινό και δεν χρεώνει για την πρόσβαση, και δεν έχουν περισσότερες από 25000 επισκέψεις στο χάρτη σε μια ημέρα. Για τα Sites που δεν πληρούν αυτές τις απαιτήσεις μπορούν να αγοράσουν το Google Maps API για επιχειρήσεις.

Η επιτυχία του Google Maps API έχει γεννήσει μια σειρά ανταγωνιστικών εναλλακτικών λύσεων, συμπεριλαμβανομένης της Yahoo! Maps API, Bing Maps Platform, MapQuest API και OpenLayers. Τον Σεπτέμβριο του 2011, η Google ανακοίνωσε ότι θα διακόψει μια σειρά των προϊόντων της, συμπεριλαμβανομένων των Google Maps API για το Adobe Flash.

### 3.6 Google Maps for Mobile

Τον Οκτώβριο του 2005, η Google παρουσίασε μια εφαρμογή Java που ονομάζεται Google Maps for Mobile, που προοριζόταν να τρέχει σε οποιοδήποτε τηλέφωνο ή σε κινητή συσκευή με δυνατότητες Java. Πολλά από τα χαρακτηριστικά της διαδικτυακής σελίδας, παρέχονται στην σημερινή μορφή της εφαρμογής Google Maps.

Στις 28 Νοεμβρίου, 2007, η εφαρμογή Google Maps for Mobile 2.0 κυκλοφόρησε. Η υπηρεσία "Βρες τη τοποθεσία μου" μπορούσε να λειτουργήσει με ή χωρίς δέκτη GPS. Η τοποθεσία της συσκευής που καθόριζε το GPS ή και Assisted GPS εάν υπάρχει, συμπληρώθηκε με την εύρεση της θέσης της συσκευής μέσω γεωμετρικού τριγωνισμού ως προς τα πλησιέστερα ασύρματα τηλεφωνικά δίκτυα. Το λογισμικό αναζητά τη θέση του χώρου της κεραίας χρησιμοποιώντας μια βάση δεδομένων γνωστών ασύρματων δικτύων και τοποθεσιών. Χρησιμοποιώντας γεωμετρικό τριγωνισμό με δεδομένα τις διαφορετικές δυνάμεις σήματος από διαφορετικούς πομπούς των κεραιών και στη συνέχεια, χρησιμοποιώντας τη θέση τους (προέρχονται από την ηλεκτρονική βάση δεδομένων κεραιών), Η τοποθεσία μου προσδιορίζει την τρέχουσα θέση του χρήστη. Η ασύρματη μέθοδος εντοπισμού του δικτύου υπολογίζεται με την ανακάλυψη των κοντινότερων WiFi hotspots και τη θέση τους (προέρχονται από την ηλεκτρονική βάση δεδομένων WiFi, κατά τον ίδιο τρόπο με τη βάση δεδομένων κεραιών) για να ανακαλύψουν περαιτέρω τη θέση του χρήστη. Η σειρά με την οποία αυτές υπερσχύουν είναι: υπηρεσίες βασισμένες σε GPS, υπηρεσίες βασισμένες σε WLAN- WiFi και υπηρεσίες βασισμένες σε κεραίες-πομπούς.



Γραφικό

περιβαλον εφαρμογης σε κινητά

Το Δεκέμβριο του 15, 2008, αυτή η υπηρεσία ήταν διαθέσιμη για τις ακόλουθες πλατφόρμες: Android, iOS, PlayStation Vita system software, Windows Mobile, Symbian OS, BlackBerry, OS, Palm OS, Palm webOS.

Στις 4 Νοεμβρίου 2009, το Google Maps Navigation κυκλοφόρησε σε συνδυασμό με το Google Android OS 2.0 Eclair για το Motorola Droid, προσθέτοντας φωνητικές εντολές, τις εκθέσεις της κυκλοφορίας, και θέα δρόμου (Street View). Αν και η αρχική έκδοση ήταν περιορισμένη στις Ηνωμένες Πολιτείες, η υπηρεσία ξεκίνησε στο Ηνωμένο Βασίλειο στις 20 Απριλίου 2010 και σε μεγάλα τμήματα της ηπειρωτικής Ευρώπης στις 9 Ιουνίου, 2010 (όπως η Αυστρία, το Βέλγιο, τον Καναδά, τη Δανία, τη Γαλλία, τη Γερμανία, την Ιταλία, τις Κάτω Χώρες, την Πορτογαλία, την Ισπανία και την Ελβετία ).

Τον Μάρτιο του 2011 Αντιπρόεδρος της Google Marissa Mayer δήλωσε ότι η Google παρέχει υπηρεσίες χάρτη για 150 εκατομμύρια χρήστες.

Παρέχει διαθέσιμη φωνητική καθοδήγηση και ζωντανές πληροφορίες για την κυκλοφορία στις πόλεις της Ινδίας της Μπανγκαλόρ, Βομβάη, Νέο Δελχί, Τσεναϊ, Πούνε και το Χυντεραμπαντ από 5 Σεπτεμβρίου του 2012.

Τον Ιούνιο του 2012, η Apple ανακοίνωσε ότι θα αντικαταστήσει την εφαρμογή Google Maps με τη δική της υπηρεσία χαρτών από το iOS 6. Η εφαρμογή και χάρτες που αντικατέστησαν το Google Maps ήρθε κάτω από σφοδρή κριτική για διάφορα προβλήματα και λάθη. Ωστόσο, στις 13 Δεκεμβρίου, 2012, η Google ανακοίνωσε τη διαθεσιμότητα των Google Maps στο App Store της Apple, ξεκινώντας με την έκδοση για iPhone. Μόλις μερικές ώρες μετά η εφαρμογή Google Maps iOS απελευθερώθηκε, έγινε η κορυφαία δωρεάν εφαρμογή στο App Store. Αν και η εφαρμογή Google Maps ως εφαρμογή δεν απολαμβάνει πλέον τη πλήρη ενσωμάτωση στο iOS, δεδομένου ότι υπήρχε ως προεγκατεστημένη εφαρμογή υπηρεσιών στο σύστημα, η νέα έκδοση προσφέρει μερικά χαρακτηριστικά για τα οποία έχουν ακουστεί πολλές καλές κριτικές συμπεριλαμβανομένου ότι είναι πιο εύκολο στη χρήση και πιο διαισθητικό ακόμη και από την υπηρεσία βασισμένη σε Android συσκευές .

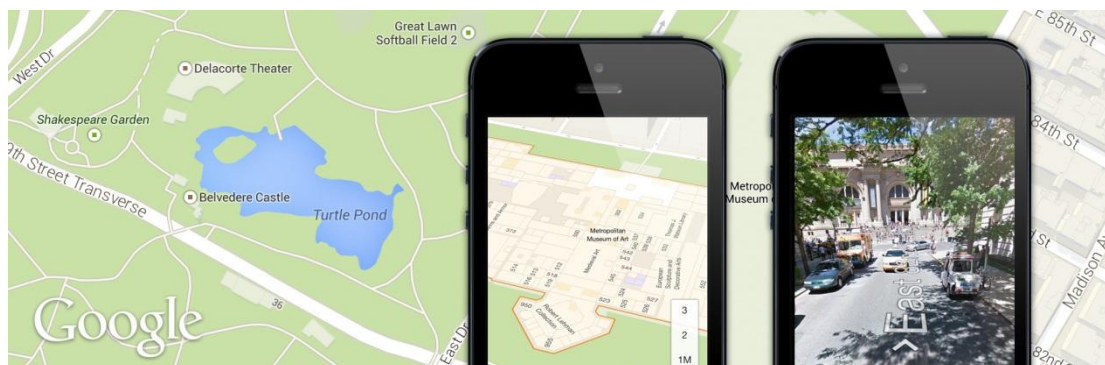
Στις 6 Δεκ 2012 είχε ανακοινωθεί ότι η εφαρμογή Google Maps θα κάνει το δρόμο του για το Wii U, κονσόλα παιχνιδιών όγδοης γενιάς της Nintendo. Προσβασιμότητα στο



Google Maps για το Wii U θα έρθει μέσω μια εφαρμογής την οποία ο χρήστης πρέπει να κατεβάσει.

### 3.6.1 Google Maps Android 2.0

Τα κινητά τηλέφωνα χρησιμοποιούνται όλο και περισσότερο για τη βοήθεια στην πλοήγηση. Το Google Maps Navigation για το Android 2.0 είναι δωρεάν. Οι δυνατότητες που παρέχονται στην εφαρμογή είναι: αναζήτηση σε αγγλικά, αναζήτηση με φωνή, προβολή κυκλοφορίας, αναζήτηση κατά μήκος της διαδρομής, δορυφορική άποψη, Street View, βάση λειτουργίας σε αυτοκίνητο.



Οι χάρτες και οι σχετικές πληροφορίες που δεν περιλαμβάνονται στη προεγκατεστημένη εφαρμογή Google Maps για Android μπορούν να αποκτηθούν εύκολα με μια απλή σύνδεση στο διαδίκτυο 'όπως και για την εφαρμογή Google Maps του iPhone. Ένας αυτόματος χάρτης με δυνατότητα προσωρινής αποθήκευσης, αποθηκεύει προσωρινά πρόσφατες περιοχές, μειώνοντας την ποσότητα των δεδομένων χρειάζονται για κατέβασμα. Η εφαρμογή επίσης διαθέτει ένα κουμπί το οποίο αναγράφει "Download Area Map" στην ενότητα Labs της Google Maps που επιτρέπει στο χρήστη να κατεβάσει τον βασικό οδικό χάρτη και τα δεδομένα ορόσημο για μια έκταση 10 τετραγωνικών μιλίων (26 τετραγωνικά χιλιόμετρα) γύρω ένα σημείο που μπορεί να καθοριστεί από το χρήστη. Αλλά ακόμα και μετά από τη λήψη ενός χάρτη μιας συγκεκριμένης περιοχής από το διαδίκτυο, μια σύνδεση δεδομένων εξακολουθεί να είναι απαραίτητη για να "δει δορυφόρο και 3D κτίρια, αναζήτηση για συγκεκριμένα μέρη και για να πάρει οδηγίες». Ο χρήστης μπορεί να κατεβάσει πολλές μικρές και περιορισμένες περιοχές αρκετές φορές για να καλύψει τη συνολική επιθυμητή περιοχή. Η Google δείχνει ότι οι χρήστες μπορούν να κάνουν χρήση αυτής της δυνατότητας για να κατεβάσουν το χάρτη για μια ξένη πόλη πριν από το ταξίδι για να την επισκεφτείτε, για να αποφευχθεί η ανάγκη για δαπανηρές διεθνείς τηλεφορτώσεις δεδομένων .

### 3.7 Offline version

Από την έκδοση 6.9 η εφαρμογή Google Maps προσφέρει offline πρόσβαση για λήψη χαρτών σε ορισμένες χώρες, ένα χαρακτηριστικό που προσφέρεται μόνο στα Labs της προηγούμενης έκδοσης .Οι χάρτες για λήψη είναι διαθέσιμοι σε 150 χώρες . Για παράδειγμα, μια χώρα που δεν είναι διαθέσιμοι οι χάρτες χωρίς σύνδεση είναι η Κολομβία.

### 3.8 Google Maps parameters


Στο Google Maps, οι παράμετροι διεύθυνσης URL είναι μερικές φορές τα δεδομένα που βασίζονται στα όριά τους και το περιβάλλον εργασίας χρήστη που υποβλήθηκε από το διαδίκτυο. Μπορεί να είναι αυτά ή μπορεί να μην αντικατοπτρίζουν τα όρια αυτά.

Ειδικότερα, το επίπεδο ζουμ (συμβολίζεται με την παράμετρο Z) ποικίλλει από περιοχή σε περιοχή. Σε λιγότερο πυκνοκατοικημένες περιοχές, τα υποστηριζόμενα επίπεδα ζουμ μπορεί να σταματήσουν στα περίπου 18x. Σε παλαιότερες εκδόσεις του API, προσδιορίζοντας τις υψηλότερες τιμές ζουμ μπορεί να οδηγήσει σε μία εικόνα που δεν εμφανίζεται. Στις δυτικές πόλεις, το υποστηριζόμενο επίπεδο ζουμ σταματά γενικά σε περίπου 20x. Σε ορισμένες μεμονωμένες περιπτώσεις, τα επίπεδα ζουμ υποστηρίζουν μέχρι και 23x ή μεγαλύτερο, καθώς σε αυτές φαίνονται οι ελέφαντες ή οι άνθρωποι σε ένα πηγάδι στο Τσαντ, την Αφρική. Διαφορετικές εκδόσεις των API και διαδικτύου μπορεί να υποστηρίζουν ή μπορεί και να μην υποστηρίζουν πλήρως αυτά τα υψηλά επίπεδα.

Από τον Οκτώβριο του 2010, η Google Map Viewer ενημερώνει συνεχώς τη πλευρική εργαλειοθήκη ζουμ για να επιτρέψει στο χρήστη να κάνει ζουμ σε όλη τη διαδρομή, όταν επικεντρώνεται σε τομείς που υποστηρίζουν υψηλότερα επίπεδα ζουμ.

## URL parameters

Telling Googlebot about each parameter's purpose helps us crawl your site more efficiently, saving you bandwidth and boosting the overall number of unique pages Googlebot can crawl. [Learn more.](#)

 Only use this feature if you feel confident about how parameters work for your site. Telling Googlebot to exclude URLs with certain parameters could result in large numbers of your pages disappearing from our index.

Add parameter

Parameter	URLs monitored	Last configured	Effect on content	What Googlebot should crawl
utm_source	21,352 -	-	-	Let Googlebot decide (default) <a href="#">Edit / Reset</a>
utm_medium	20,976 -	-	-	Let Googlebot decide (default) <a href="#">Edit / Reset</a>
utm_campaign	20,216 -	-	-	Let Googlebot decide (default) <a href="#">Edit / Reset</a>
p	2,385 -	-	-	Let Googlebot decide (default) <a href="#">Edit / Reset</a>

Url Parameters

## 3.9 Ιστορικά

### 3.9.1 Απόκτηση

Το Google Maps ξεκίνησε ως ένα C++ πρόγραμμα που σχεδιάστηκε από δύο Δανούς αδελφούς Lars και Jens Rasmussen σε μια εταιρία που εδρεύει στο Σύννεϋ - Αυστραλία την Where 2 Technologies. Στην αρχή είχε σχεδιαστεί έτσι ώστε να γίνεται λήψη ξεχωριστά από τους χρήστες, αλλά η εταιρεία σκέφτηκε αργότερα η ιδέα για ένα καθαρά διαδικτυακά βασισμένο προϊόν στη διαχείριση της Google, δηλαδή την αλλαγή του τρόπου διανομής. Τον Οκτώβριο του 2004 η εταιρεία εξαγοράστηκε από την Google Inc, όπου μετατράπηκε στην εφαρμογή που γνωρίζουμε όλοι σήμερα ως Google Maps

### 3.9.2 Ιστορική Εξέλιξη

#### 3.9.2.1 2005

Η εφαρμογή ανακοινώθηκε για πρώτη φορά στο blog της Google στις 8 Φεβρουαρίου, 2005, και βρισκόταν στα χέρια της Google. Αρχικά υποστηριζόταν μόνο για χρήστες του Internet Explorer και Mozilla περιηγητές διαδικτύου. Η υποστήριξη για Opera και Safari προστέθηκε στις 25 Φεβ 2005, αλλά προς το παρόν Opera έχει αφαιρεθεί από τις



απαιτήσεις λίστας συστήματος . Βρισκόταν σε δοκιμαστική φάση (beta) για έξι μήνες πριν γίνει μέρος της Google Local στις 6 Οκτωβρίου 2005. Τον Απρίλιο του 2005, η Google δημιούργησε το Google Ride Finder χρησιμοποιώντας το Google Maps. Τον Ιούνιο του 2005, η Google κυκλοφόρησε το Google Maps API. Τον Ιούλιο του 2005, η Google ξεκίνησε το Google Maps και το Google Local υπηρεσίες για την Ιαπωνία, συμπεριλαμβανομένων των οδικών χαρτών. Στις 22 Ιουλίου του 2005, η Google κυκλοφόρησε το "Hybrid View". Μαζί με αυτή την αλλαγή, τα δορυφορικά δεδομένα εικόνες μετατράπηκαν από την προβολή plate carrée σε Mercator προβολή, γεγονός που κάνει δυνατή μια λιγότερο παραμορφωμένη εικόνα στα γεωγραφικά πλάτη με πιο θερμά κλίματα. Τον Ιούλιο του 2005, προς τιμήν της τριακοστής έκτης επετείου της απόβασης Apollo στη σελήνη, το Google Moon ξεκίνησε. Τον Σεπτέμβριο του 2005, στον απόηχο του τυφώνα Κατρίνα, το Google Maps θα ενημερώνει καθημερινά και γρήγορα δορυφορικές εικόνες της Νέας Ορλεάνης για να επιτρέψει στους χρήστες να δουν την έκταση των πλημμυρών σε διάφορα μέρη της πόλης. (Παραδόξως το Μάρτιο του 2007, εικόνες που δείχνουν τις ζημιές του τυφώνα αντικαταστάθηκαν με εικόνες πριν από την καταιγίδα! Η αντικατάσταση αυτή όμως δεν έγινε στο Google Earth, το οποίο εξακολουθεί να χρησιμοποιεί εικόνες έπειτα του τυφώνα Κατρίνα )

### **3.9.2.2 2006**

Από τον Ιανουάριο του 2006, το Google Maps παρείχε οδικούς χάρτες για τις Ηνωμένες Πολιτείες, Πουέρτο Ρίκο, το Καναδά, το Ηνωμένο Βασίλειο, την Ιαπωνία, και ορισμένες πόλεις στη Δημοκρατία της Ιρλανδίας. Κάλυψη της περιοχής γύρω από το Τορίνο προστέθηκαν εγκαίρως για τους χειμερινούς Ολυμπιακούς αγώνες του 2006. Την 23η Ιανουαρίου 2006, το Google Maps έχει ενημερωθεί ώστε να χρησιμοποιεί την ίδια βάση δεδομένων δορυφορικών εικόνων, όπως το Google Earth. Στις 12 Μαρτίου 2006, η Google Mars ξεκίνησε, το οποίο διαθέτει ένα συρόμενο χάρτη και δορυφορικές εικόνες του πλανήτη Άρη. Τον Απρίλιο του 2006, η Google Local συγχωνεύθηκε με το κεντρικό site Google Maps. Στις 3 Απριλίου 2006, η δεύτερη έκδοση του Maps API κυκλοφόρησε. Στις 11 Ιουνίου 2006, η Google πρόσθεσε δυνατότητες γεωκωδικοποίησης στο API, ικανοποιώντας έτσι την μεγαλύτερη σε απαίτηση ανάγκη από προγραμματιστές για την υπηρεσία αυτή. Στις 14 Ιουνίου του 2006, το Google Maps για επιχειρήσεις εγκαινιάστηκε επίσημα. Ως εμπορική υπηρεσία, διαθέτει intranet και υλοποιήσεις ελεύθερες από διαφημίσεις. Επίσης τον Ιούνιο, τρισδιάστατα μοντέλα κτιρίων προστέθηκαν στο Google Earth. Τον Ιούλιο του 2006 η Google ξεκίνησε την εισαγωγή επιχειρήσεων στα κύρια αποτελέσματα αναζήτησης Google συμπεριλαμβάνοντας τις καταχωρίσεις επιχειρήσεων από το Google Maps. Στις 9 Δεκεμβρίου η Google ενσωματώνει την Plusbox στα κύρια αποτελέσματα της αναζήτησης, στις 19 Δεκεμβρίου η Google πρόσθεσε ένα χαρακτηριστικό που επιτρέπει στο χρήστη να προσθέτει πολλαπλούς προορισμούς στη πλοήγηση .

### **3.9.2.3 2007**

Αρχίζοντας το Φεβρουάριο του 2007, εμφανίζονται σημαντικά κτίρια και οι στάσεις του μετρό στη εφαρμογή Google Maps σε περιοχές της Νέας Υόρκης, Ουάσιγκτον, στο Λονδίνο, το Σαν Φρανσίσκο και μερικές άλλες πόλεις. Στις 29 Ιανουαρίου 2007 τα αποτελέσματα στην αναζήτηση Google αναβαθμίστηκαν και περισσότερα στοιχεία περιλαμβάνονται στην κεντρική σελίδα αποτελεσμάτων της Google. Στις 28 Φεβρουαρίου 2007, εγκαινιάστηκε επίσημα η Google Traffic info και περιλαμβάνει αυτόματα σε πραγματικό χρόνο συνθήκες ροής κυκλοφορίας με τους χάρτες των 30 μεγαλύτερων πόλεων των Ηνωμένων Πολιτειών της Αμερικής. Στις 8 Μαρτίου του 2007, το Local Business Center αναβαθμίστηκε. Στις 16 Μάη 2007 η Google ξεδιπλώνει όλα τα αποτελέσματα αναζήτησης, προβάλλοντας περισσότερες πληροφορίες χάρτη στην κύρια σελίδα αποτελεσμάτων του Google. Στις 18 Μαΐου 2007 η Google πρόσθεσε δυνατότητες

αναζήτησης γειτονιάς. Στις 29 Μαΐου 2007, το Google Driving Directions Support προστέθηκε στο Google Maps API. Στις 29 Μαΐου 2007, το Street View προστέθηκε, δίνοντας θέα 360 μοιρών στο επίπεδο του εδάφους σε δρόμους και σε ορισμένες μεγάλες πόλεις στις Ηνωμένες Πολιτείες. Στις 19 Ιουνίου, 2007, κριτικές επιτρέπεται να προστεθούν άμεσα στις επιχειρήσεις στο Google Maps. Στις 28 Ιουνίου 2007, συρόμενες οδηγίες οδήγησης εισήχθησαν στο Google Maps. Στις 31 Ιουλίου 2007, ανακοινώθηκε η υποστήριξη για την microformat hCard. Στις 21 Αυγούστου 2007, η Google ανακοίνωσε έναν απλό τρόπο για να ενσωματώνετε το Google Maps σε άλλες ιστοσελίδες. Στις 13 Σεπτεμβρίου, 2007, 54 νέες χώρες προστέθηκαν στο Google Maps στη Λατινική Αμερική και την Ασία, στις 3 Οκτωβρίου του 2007, το Google Transit ενσωματώθηκε στο Google Maps κάνοντας τη δρομολόγηση μαζικών μέσων μεταφοράς στο Google Maps. Στις 27 Οκτωβρίου 2007, το Google Maps ξεκίνησε τη χαρτογράφηση του Geoweb και δείχνει τα αποτελέσματα στο Google Maps. Στις 27 Οκτωβρίου 2007, το Google Maps πρόσθεσε μια εύχρηστη διεπαφή για κουπόνια επιχειρήσεων στους καταλόγους των επιχειρήσεων. Στις 27 Νοεμβρίου 2007, προστέθηκε θέα εδάφους που δείχνει βασικά τοπογραφικά χαρακτηριστικά. Το κουμπί για το υβριδική θέα και αντικαταστάθηκε με ένα κουτάκι επιλογής κάτω από το κουμπί "Δορυφόρος" για την εναλλαγή μεταξύ υβριδικής και δορυφορικής θέας.

### **3.9.2.4 2008**

Στις 22 του Γενάρη του 2008, η Google επεκτείνει το Local OneBox από τρεις καταχωρήσεις επιχειρήσεων έως δέκα. Στις 20 Φεβρουαρίου, 2008, η Google Maps επιτρέπει στις αναζητήσεις να φαίνεται βαθμολογία χρηστών και διεύθυνση. Στις 18 Μαρτίου 2008, η Google επιτρέπεται στους τελικούς χρήστες να επεξεργάζονται τις καταχωρήσεις των επιχειρήσεων και να προσθέτουν νέες θέσεις. Στις 19 Μαρτίου 2008, η Google πρόσθεσε απεριόριστες επιλογές κατηγορίας στο Local Business Center. Στις 2 Απριλίου 2008, η Google πρόσθεσε ισοϋψείς γραμμές στη θέα εδάφους. Τον Απρίλιο του 2008, ένα κουμπί για να δείτε τις πρόσφατες αποθηκευμένες τοποθεσίες προστέθηκε στα δεξιά του πεδίου αναζήτησης. Το Μάιο του 2008, προστίθεται ένα κουμπί "Περισσότερα" παράλληλα με τα κουμπιά "Χάρτης", "Θέα Δορυφόρου", και «Θέα Εδάφους», επιτρέποντας την πρόσβαση σε γεωγραφικά σχετικές φωτογραφίες στη βάση φωτογραφιών Panoramio και άρθρα σχετικά με Wikipedia. Στις 15 Μαΐου του 2008, το Google Maps μεταφέρθηκε στο Adobe Flash και ActionScript 3 ως θεμέλιο για τη πλούσια γκάμα εφαρμογών. Στις 15 Ιουλίου 2008, οι οδηγίες για περπάτημα προστέθηκαν. Στις 4 Αυγούστου 2008, το Street View ξεκίνησε στην Ιαπωνία και την Αυστραλία. Στις 5 Αυγούστου 2008, έχει επανασχεδιαστεί η διεπαφή χρήστη. Στις 29 Αυγούστου 2008, η Google υπέγραψε συμφωνία βάσει της οποίας η GeoEye θα τους προμηθεύσει με εικόνες από δορυφόρο, και εισήγαγε το εργαλείο Map Maker, το οποίο επιτρέπει σε κάθε χρήστη να βελτιώσει τα δεδομένα χάρτη όταν γίνει έλεγχος από όλους. Στις 9 Σεπτεμβρίου 2008, προστέθηκε δυνατότητα αναζήτησης επιχειρήσεων αντίστροφα. Στις 23 Σεπτεμβρίου 2008, προστέθηκαν πληροφορίες για τη New York City Metropolitan Transit Authority. Στις 7 Οκτωβρίου, 2008, η GeoEye πήρε την πρώτη εικόνα της, μια προοπτική άποψη από το Kutztown University στην Πενσυλβανία. Στις 26 Οκτωβρίου 2008, αντίστροφη γεωκωδικοποίηση προστέθηκε στο Maps API. Στις 11 Νοεμβρίου, 2008, εισήχθη το Street View στην Ισπανία, την Ιταλία και τη Γαλλία. Στις 23 Νοέμβριου 2008, Adobe AIR υποστήριξη για το Maps API για το Adobe Flash προστέθηκε. Στις 25 Νοεμβρίου 2008, ένα νέο περιβάλλον εργασίας χρήστη για το Street View εισήχθη. Στις 27 Νοεμβρίου 2008, χάρτες, πληροφορίες για τοπικές επιχειρήσεις, και τοπικές στάσεις για την Κίνα εισήχθησαν. Στις 9 Δεκεμβρίου 2008, εισήχθηκε 2x Street View κάλυψη.

### **3.9.2.5 2009**

Στις 19, Μάρτη 2009 το Street View ξεκίνησε στο Ηνωμένο Βασίλειο και τις Κάτω Χώρες. Το Μάιο του 2009, ένα νέο λογότυπο του Google Maps εισήχθη. Στις αρχές Οκτωβρίου 2009, η Google αντικατέστησε τη Tele Atlas ως κύριο προμηθευτή τους γεωχωρικών δεδομένων στην αμερικανική έκδοση του Maps για να χρησιμοποιούν δικά τους δεδομένα. Τον Οκτώβριο του 2009, οι σιδηρόδρομοι είχαν επαναληφθεί, το οποίο διαθέτει μια ελαφρώς νέα εμφάνιση και ενημέρωση, και γίνεται ταυτόχρονη αφαίρεση παλαιότερων γραμμών. Επίσης, τον ίδιο μήνα, χάρτες σε διάφορες περιοχές είχαν αλλάξει ώστε να συμπεριλαμβάνουν δρόμους που δεν υπάρχουν, καθώς και γραμμές τραίνου που εμφανίζονται στο χάρτη αλλά δεν υπάρχουν.

### **3.9.2.6 2010**

Στις 11 Φεβρουαρίου, 2010, η Google Maps Labs προστέθηκε. Στις 11 Μαρτίου, 2010, το Street View στο Χονγκ Κονγκ και το Μακάο ξεκίνησαν. Στις 25 Μαΐου 2010, προστέθηκε η δρομολόγηση για τα μέσα μαζικής μεταφοράς στη Δανία και προστέθηκε με την ενσωμάτωση με Rejseplanen. dk. Από τις 20 Δεκεμβρίου 2010 οι περιηγητές Internet Explorer 7.0 +, Firefox 3.6 +, Safari 3.1 + και Google Chrome υποστηρίζονται.

### **3.9.2.7 2011**

Στις 8 Απριλίου, η Google ανακοίνωσε ότι θα αρχίσουμε να χρεώνουμε για τη χρήση του API από εμπορικές περιοχές πάνω από ένα όριο. Επίσης θα εισαχθεί μία πριμοδοτημένη άδειας παροχής υπηρεσιών. Στις 19 Απριλίου, 2011, Map Maker προστέθηκε στο Google Maps για τις Ηνωμένες Πολιτείες Αμερικής, επιτρέποντας κάθε θεατή να επεξεργαστείτε και να προσθέσετε τις αλλαγές στο Google Maps. Αυτό παράλληλα παρέχει στη Google τις τοπικές ενημερώσεις χάρτη σχεδόν σε πραγματικό χρόνο αντί να περιμένουν για εταιρείες ψηφιακών δεδομένων χάρτη να κυκλοφορήσουν λιγότερα συχνές αναβαθμίσεις στο χάρτη.

### **3.9.2.8 2012**

Στις 31 Ιανουαρίου 2012, επειδή η Google παρείχε την εφαρμογή Google Maps, κρίθηκε ένοχη για κατάχρηση της δεσπόζουσας θέσης της εφαρμογής Google Maps και είχε διαταχθεί από το δικαστήριο να καταβάλει πρόστιμο και ζημιές στη Bottin Cartographes, μια γαλλική εταιρεία χαρτογράφησης. Στις 30 Μαΐου 2012, τμήματα της Google αντικαταστάθηκε από το Google+ Local, το οποίο τώρα ενσωματώνετε άμεσα με το Google+ υπηρεσία η οποία επιτρέπει στους χρήστες να δημοσιεύουν φωτογραφίες και σχόλια των θέσεων απευθείας στη σελίδα του από την υπηρεσία. Επιπλέον, Google+ Local και Maps, επίσης, διαθέτουν πλέον λεπτομερή σχόλια και τις βαθμολογίες από τη Zagat, η οποία εξαγοράστηκε από την Google το Σεπτέμβριο του 2011. Τον Ιούνιο του 2012, η Google ξεκίνησε τη χαρτογράφηση των ποταμών και των καναλιών της Βρετανίας σε συνεργασία με την Canal and River Trust. Η εταιρεία έχει δηλώσει ότι θα ενημερώσει το πρόγραμμα κατά τη διάρκεια του έτους για να επιτρέψει στους χρήστες να προγραμματίζουν τα ταξίδια που περιλαμβάνουν γέφυρες και μονοπάτια κατά μήκος των 2.000 χιλιομέτρων του ποταμού στο Ηνωμένο Βασίλειο. Στις 11 Οκτώβριου ανακοινώθηκε ότι η Google ενημέρωσε στις βάσεις δεδομένων της 250.000 μίλια των δρόμων στις ΗΠΑ. Τον Δεκέμβριο του 2012, η εφαρμογή Google Maps διατίθεται επίσημα στο App Store της Apple μετά την αφαίρεση του από τη κινητή έκδοση του λειτουργικού συστήματος του iOS 6. Ο Tim Cook αναγκάστηκε να κάνει μια δημόσια απολογία και να συστήσει άλλες παρόμοιες εφαρμογές.

### **3.9.2.9 2013**

Στις 29 Ιανουαρίου 2013, το Google Maps ενημερώθηκε με χάρτη της Βόρειου Κορέας. Στις 23 Απριλίου 2013, το Street View ξεκίνησε στην Ουγγαρία. Στις 27 Μαρτίου 2013, η Google ξεκίνησε το Google Maps Engine Lite, μια απλοποιημένη έκδοση του εμπορικού προϊόντος, η οποία προορίζεται να αντικαταστήσει τελικά τη λειτουργία My Maps. Την 23η Απριλίου 2013, το Street View ξεκίνησε στην Ουγγαρία και το Λεσότο, επεκτείνοντας την κάλυψη των φωτογραφιών 360 μοιρών Street View σε πενήντα χώρες. Κατά την ίδια χρονική περίοδο, η Google ολοκλήρωσε επίσης την μεγαλύτερη ενιαία ενημέρωση της βάσης δεδομένων Street View, με φωτογραφίες τάξεως πάνω από 350.000 μίλια (560.000 χιλιομέτρων) του δρόμου σε δεκατέσσερις χώρες. Από 3 Μαΐου, 2013, το Google Maps αναγνωρίζει την Παλαιστίνη ως χώρα, αντί του αναπροσανατολισμού προς τα παλαιστινιακά εδάφη. Η Google ανακοίνωσε στο blog του Google Maps στις 15 Μαΐου 2013 ότι μια νέα αναβαθμισμένη έκδοση του Google Maps είναι διαθέσιμη για χρήση από τους εν λόγω εγγεγραμμένους χρήστες του Google που ζητήσουν μια πρόσκληση. Το νέο Google Maps μπορείτε να δημιουργήσετε έναν προσαρμοσμένο χάρτη που είναι ειδικά για την συμπεριφορά του κάθε χρήστη, αποκαλύπτοντας στιγμιότυπα που βασίζονται στις πληροφορίες που έχει εισαχθεί, και παρέχοντας χρήσιμες πληροφορίες για την περιοχή, όπως εστιατόρια. Ένα νέο χαρακτηριστικό είναι μία λειτουργία που συγκεντρώνει όλες τις εικόνες του Google Maps σε μία θέση και περιέχει μια θέα της Γης που ενσωματώνει άμεσα την 3D εμπειρία από το Google Earth στους νέους χάρτες. Η νέα έκδοση είναι επίσης πιο στενά συνδεδεμένη στο Google+ και οι τοπικές επιχειρήσεις που εμφανίζονται βασίζονται στο εξατομικευμένο Google+ δίκτυο του κάθε χρήστη. Οι διαφημίσεις στους νέο Google Maps έχει επανασχεδιαστεί και είναι μικρά τμήματα διαφημίσεων που τοποθετούνται απευθείας πάνω στον ίδιο το χάρτη, παράλληλα με την επωνυμία της επιχείρησης. Τον Αύγουστο του 2013, το Google Maps αφαιρεί το Wikipedia Layer, το οποίο περιείχε συνδέσεις με το περιεχόμενο της Wikipedia σχετικά με τις θέσεις φαίνονται στους Χάρτες Google με τη χρήση της Wikipedia Geocodes

### **3.9.2.10 2014**

Στις 21 Μαρτίου 2014, η Google ξεδιπλώνει ένα νέο περιβάλλον Google Maps με επιπλέον χαρακτηριστικά. Δυστυχώς το προϊόν είναι άορατο σε όλους τους χρήστες που έχουν κλειδώμα των cookies, ή παραπέμπτες, και η προεπισκόπηση περιορίζεται μόνο σε χρήστες οι οποίοι διαθέτουν λογαριασμό και βρίσκονται εντός του λογαριασμού τους. Στις 5 Απρίλη 2014, το Street View ξεκίνησε από τα Νησιά Κουκ, επεκτείνοντας την κάλυψη των εικόνων 360 μοιρών Street View σε 58 χώρες. Κατά την ίδια χρονική περίοδο, η Google ολοκλήρωσε επίσης την μεγαλύτερη ενιαία ενημέρωση της βάσης δεδομένων του Street View, με φωτογραφίες της πάνω από 370.000 μίλια (600.000 χιλιόμετρα) των δρόμων. Στις 12 Απριλίου 2014, το Google Maps ενημερώθηκε ώστε να αντικατοπτρίζει τη κρίση στη Κριμαία 2014. Η Κριμαία εμφανίζεται ως Δημοκρατία της Κριμαίας στη Ρωσία και ως Αυτόνομη Δημοκρατία της Κριμαίας στην Ουκρανία. Όλες οι άλλες εκδοχές δείχνουν μια διακεκομμένη αμφισβητούμενη γραμμή ως σύνορα.

## **3.10 Εφαρμογές που χρησιμοποιούν το Google Maps**

### **3.10.1 Google Ditu**

Το Google Ditu κυκλοφόρησε για το κοινό στις 9 Φεβρουαρίου του 2007, και αντικατέστησε το παλιό Google Bendi. Αυτή είναι η κινεζική μεταφρασμένη έκδοση της υπηρεσίας Google Maps και του Google Local. Για να είναι συμβατό με τις απαιτήσεις της κινεζικής νομοθεσίας, η Google έπρεπε να αφαιρέσει ή να τροποποιήσει ορισμένες δυνατότητες του Google Maps στο Google Ditu: Το Google Ditu δεν επιτρέπει την επικάλυψη του περιεχόμενου που εξαρτάται από τις προτιμήσεις του χρήστη από το

Panoramio, το YouTube, Wikipedia και κάμερες ταχύτητας. Το Google Ditu δείχνει τις επίμαχες περιοχές των συνόρων μεταξύ της Κίνας και της Ινδίας ως τμήμα της Κίνας, ενώ στο Google Maps αυτές οι επίμαχες περιοχές εμφανίζονται μέσα σε διακεκομμένες γραμμές.

### 3.10.2 Google Moon

Προς τιμήν της 36ης επετείου της προσσελήνωσης του Apollo 11 στις 20 Ιουλίου 1969, η Google πήρε δημόσια εικόνες ενός τομέα της Σελήνης, ενσωμάτωσε το Google Maps interface, και δημιούργησε ένα εργαλείο που ονομάζεται Google Moon. Με προεπιλογή αυτό το εργαλείο, με μειωμένο σύνολο χαρακτηριστικών, εμφανίζει επίσης τα σημεία εκφόρτωσης όλων των διαστημοπλοίων Apollo να προσγειωθεί στο φεγγάρι. Περιελάμβανε επίσης ένα αυγό το Πάσχα, εμφανίζοντας ένα σχεδιασμό ελβετικού τυριού στο υψηλότερο επίπεδο ζουμ, το οποίο η Google έχει αφαιρέσει από τότε. Ένα πρόσφατο πρόγραμμα συνεργασίας μεταξύ της NASA Ames Research Center και το Google ενσωματώνει και τη βελτίωση των δεδομένων που χρησιμοποιείται για την Google Σελήνη. Αυτή είναι η Planetary Περιεχόμενο έργο. Google Moon συνδέθηκε από μια ειδική αναμνηστική έκδοση του λογότυπου της Google που εμφανίζονται στο επάνω μέρος της κεντρικής σελίδας αναζήτησης της Google για τις 20 Ιούλη του 2005.



Φωτογραφία από το Google moon.

### 3.10.3 Google Mars

Το Google Mars παρέχει μια προβολή εικόνων, όπως το Google Moon, καθώς και εικόνες από υπέρυθρη φωτογραφική και ανύψωση εδάφους του πλανήτη Άρη. Οι χρήστες μπορούν να επιλέξουν μεταξύ θέας ανύψωσης, προβολής εικόνων και εικόνες από υπέρυθρη φωτογραφική, με τον ίδιο τρόπο εναλλαγή μεταξύ χάρτη, δορυφορική, και σε υβριδική κατάσταση του Google Maps. Σε συνεργασία με επιστήμονες της NASA στην Εγκατάσταση Διαστημικών Πτήσεων στον Άρη η οποία βρίσκεται στο Arizona State University, η Google διαθέτει στο κοινό δεδομένα που συλλέχθηκαν από δύο αποστολές της NASA στο πλανήτη Άρη, την Mars Global Surveyor και Mars Odyssey 2001.

Τώρα, με το Google Earth 5 είναι δυνατή η πρόσβαση σε νέα βελτιωμένα δεδομένα στο Google Mars σε πολύ υψηλότερη ανάλυση, καθώς και τη θέα 360 μοιρών (παρόμοια με το Google Street View). και την προβολή πανοραμάτων από διάφορες μη επανδρωμένες αποστολές στον Άρη.

### **3.10.4 Google Sky**

Στις 27 Αυγούστου, 2007, η Google παρουσίασε το Google Sky, ένα online εργαλείο χαρτογράφησης χώρου που επιτρέπει στους χρήστες να περιηγηθείτε μέσα από ένα χάρτη του ορατού σύμπαντος, με φωτογραφίες που ελήφθησαν από το διαστημικό τηλεσκόπιο Hubble.

### **3.10.5 Google Ride Finder**

Η Google ξεκίνησε ένα πειραματικό εργαλείο βασισμένο στο Google Maps που ονομάζεται Ride Finder, αξιοποιώντας τις μονάδες GPS στα ταξί και λιμουζίνες τα οποία ήθελαν να συμμετέχουν στο πρόγραμμα. Το εργαλείο εμφανίζει την τρέχουσα θέση όλων των οχημάτων που υποστηρίζονται από τις συμμετέχουσες υπηρεσίες στις μεγάλες πόλεις των ΗΠΑ, όπως το Σικάγο και το Σαν Φρανσίσκο, σε ένα χάρτη από την υπηρεσία Google Maps. Το εργαλείο δυστυχώς απέτυχε και διακόπηκε το 2009.

### **3.10.6 Google Transit**

Τον Δεκέμβριο του 2005, η Google κυκλοφόρησε το Google Transit στο Google Labs το οποίο χαρτογραφούσε δρομολόγια μαζικών μέσων μεταφοράς, μια έννοια η οποία επινοήθηκε μερικώς από τον Chris Harelson και Avichal Garg. Το Google Transit ξεκίνησε αρχικά με υποστήριξη διαδρομών στο Πόρτλαντ, Όρεγκον, και τώρα περιλαμβάνει εκατοντάδες πόλεις στις Ηνωμένες Πολιτείες, τον Καναδά, την Ευρώπη, την Ασία, την Αφρική, την Αυστραλία, την Ινδία και τη Νέα Ζηλανδία. Η υπηρεσία υπολογίζει τη διαδρομή, το χρόνο και το κόστος διαδρομής, και να συγκρίνει το ταξίδι με χρήση αυτοκινήτου. Τον Οκτώβριο του 2007 Google Transit αποφοίτησε από το Google Labs ενσωματώθηκε πλήρως στο Google Maps.

Η κάλυψη του Google Transit είναι διαθέσιμη στο κοινό. Έχει εξαπλωθεί σε όλο τον κόσμο, σε εκατοντάδες πόλεις και ολόκληρες χώρες, όπως η Κίνα, η Ιαπωνία, η Ελβετία. Η κάλυψη των μεγάλων πόλεων στις Ηνωμένες Πολιτείες και τον Καναδά είναι σχεδόν πλήρης. Σε ορισμένους τομείς, όπως το Ηνωμένο Βασίλειο, το Google Transit καλύπτει μόνο ένα μέρος των υπηρεσιών μαζικής μεταφοράς. Σε άλλες χώρες και περιοχές, το Google Transit δεν παρέχει οδηγίες οδήγησης των δημόσιων συγκοινωνιών, αλλά εξακολουθεί να παρέχει στο χάρτη τη διαδρομή διέλευσης των δημοσίων συγκοινωνιών. Παραδείγματα περιλαμβάνουν το Παρίσι, το Βερολίνο, τη Πόλη του Μεξικού και πολλές άλλες πρωτεύουσες σε όλο τον κόσμο.

### **3.10.7 Google Biking Directions**



*Directions from Google Maps for Bicycling*

Στις 10 Μαρτίου, 2010, η Google πρόσθεσε τη δυνατότητα λήψης οδηγιών για ποδηλασία στο Google Maps. Οι βέλτιστες διαδρομές έχουν υπολογιστεί βασισμένες στην κίνηση, την αλλαγή υψόμετρου, τους ποδηλατοδρόμους, και προνομιούχων δρόμων για ποδηλασία. Ένα προαιρετικό στρώμα στο Google Maps δείχνει επίσης διάφορα είδη ποδηλασίας, από τη ποδηλασία μόνο σε διαδρομές ποδηλασίας μέχρι και ποδηλασία σε κανονικούς δρόμους. Αυτή η υπηρεσία είναι διαθέσιμη στις ΗΠΑ και τον Καναδά και είναι σε πειραματικό στάδιο σε άλλες χώρες, όπως τη Σιγκαπούρη.

### 3.10.8 Google My Maps

Τον Απρίλιο του 2007, το My Maps ήταν ένα νέο χαρακτηριστικό που προστέθηκε στους τοπικούς χάρτες αναζήτησης της Google. Η υπηρεσία My Maps επιτρέπει σε χρήστες και επιχειρήσεις να δημιουργούν το δικό τους χάρτη, με δείκτες τοποθεσίας, γραμμές και πολύγωνα σε ένα χάρτη. Το interface είναι μια απλή επικάλυψη στο χάρτη. Ένα σύνολο από ογδόντα τέσσερις προσχεδιασμένους δείκτες είναι διαθέσιμοι, που κυμαίνονται από τα μπαρ και εστιατόρια στην κάμερα ελέγχου κυκλοφορίας και σεισμογενείς περιοχές. Οι χάρτες οι οποίοι έχουν τροποποιηθεί χρησιμοποιώντας την υπηρεσία My Maps μπορούν να αποθηκευτούν για μετέπειτα προβολή αλλά επίσης δημοσιοποιούνται ή σημαδεύονται, με τον χρήστη που έκανε τις αλλαγές να αποθηκεύσει τη διεύθυνση URL που δημιουργείται αυτόματα με αναγνωριστικό 42 χαρακτήρων.

Κάθε στοιχείο που προστίθεται σε ένα χάρτη χρησιμοποιώντας το My Maps έχει επεξεργάσιμη ετικέτα. Αυτή η ετικέτα μπορεί να περιέχει κείμενο ή κείμενο εμπλουτισμένο με HTML. Στη δεύτερη περίπτωση μπορεί να περιέχει βίντεο και άλλα είδη περιεχόμενου που μπορεί να περιλαμβάνονται στο κείμενο εμπλουτισμένο με HTML.

Κατά την έναρξη του My Maps δεν υπήρχε δυνατότητα να ενσωματωθούν οι τροποποιημένοι χάρτες σε μια ιστοσελίδα ή blog. Μερικές ανεξάρτητες ιστοσελίδες είχαν δημιουργήσει τα εν λόγω εργαλεία για να επιτρέψει στους χρήστες να ενσωματώσουν χάρτες που δημιουργήθηκαν στο My Maps και να προσθέσουν επιπλέον λειτουργικότητα σε χάρτες τους. Αυτό έχει επιλυθεί με την έκδοση 2.78 του Google Maps

### 3.10.9 Google Map Maker

Σε περιοχές όπου είναι διαθέσιμο το εργαλείο Google Map Maker, για παράδειγμα, ένα μεγάλο μέρος της Ασίας, της Αφρικής, της Λατινικής Αμερικής της Ανατολικής Ευρώπης, καθώς, στις Ηνωμένες Πολιτείες και τον Καναδά, όποιος συνδέεται στο Google λογαριασμό τους μπορεί να βελτιώσει άμεσα το χάρτη με τον καθορισμό λανθασμένων οδηγιών οδήγησης, προσθέτοντας ποδηλασία, ή την προσθήκη καινούριου κτιρίου η



δρόμου. Γενικά τα λάθη στο Google Maps στην Αυστραλία, την Αυστρία, το Βέλγιο, τη Δανία, τη Γαλλία, το Λιχτενστάιν, Ολλανδία, Νέα Ζηλανδία, Νορβηγία, Νότια Αφρική, την Ελβετία και τις Ηνωμένες Πολιτείες μπορούν να καταγραφούν με μία έκθεση προβλήματος στο Google Maps και θα ενημερώνεται έπειτα από την Google.

Για τις περιοχές όπου η Google χρησιμοποιεί δεδομένα Tele Atlas, σφάλματα στο χάρτη μπορούν να αναφερθούν με τη χρήση χαρτών της Tele Atlas. Σε περίπτωση που εικόνες λείπουν, είναι ξεπερασμένες, μη ευθυγραμμισμένες ή γενικά ανακριβής, μπορεί ο οποιοσδήποτε χρήστης να ειδοποιήσει τη Google μέσω μιας αίτησης στην ιστοσελίδα τους.

### **3.11 Σεβασμός εθνικής κυριαρχίας**

Υπάρχουν ορισμένες διαφορές σε ευθυγραμμίσεις συνόρων μεταξύ του Google Ditu και της εφαρμογής Google Maps. Στο Google Maps, τμήματα των συνόρων της Κίνας με την Ινδία και το Πακιστάν επισημαίνονται με διακεκομμένες γραμμές, υποδεικνύοντας περιοχές ή σύνορα διαφορετικά. Ωστόσο, η Google Ditu δείχνει τα σύνορα της Κίνας με την Ινδία και το Πακιστάν αυστηρά σύμφωνα με τις κινέζικες αξιώσεις χωρίς διακεκομμένες γραμμές. Για παράδειγμα, η περιοχή Νότια του Θιβέτ διεκδικείται από την Κίνα, αλλά διοικείται από την Ινδία ως ένα μεγάλο μέρος της Αρουντσάλ. Στο Google Ditu εσωτερικά των συνόρων σύμφωνα με τη κινέζικη αξίωση δείχνει ινδικούς αυτοκινητόδρομους να τελειώνουν απότομα στα σύνορα της Κίνας. Στο Google Ditu δείχνει, επίσης, την Ταϊβάν και νησιά στη θάλασσα της Νότιας Κίνας ως τμήμα της Κίνας. Από το Μάιο του 2009, ο οδικός χάρτης της Ταϊβάν στο Google Ditu παραλείπει επίσης σημαντικά κρατικά κτίρια, όπως το Προεδρικό Μέγαρο, τα πέντε Yuans, και το Ανώτατο Δικαστήριο.

Υπάρχουν ορισμένες διαφορές μεταξύ [ditu.google.cn](http://ditu.google.cn) και [ditu.google.com](http://ditu.google.com). Για παράδειγμα, η σελίδα [ditu.google.cn](http://ditu.google.cn) διαθέτει το εργαλείο My Maps. Από την άλλη πλευρά, ενώ στη πρώτη εμφανίζει σχεδόν όλο το κείμενο στα κινέζικα, η δεύτερη εμφανίζει κείμενο στην αγγλική γλώσσα. Αυτή η συμπεριφορά του να εμφανίζει κείμενο στην αγγλική γλώσσα δεν είναι συνεπής, αλλά διαλείπουσα - μερικές φορές είναι στην αγγλική γλώσσα, μερικές φορές είναι στα κινέζικα. Τα κριτήρια για την επιλογή γλώσσας εμφάνισης δεν είναι γνωστά στο κοινό.



## 4 Περιγραφή εφαρμογής

### 4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δούμε τα βήματα που ακολουθήσαμε ώστε να φτιάξουμε την εφαρμογή. Θα ξεκινήσουμε από μία γενική παρουσίαση της εφαρμογής, θα δούμε τα εργαλεία που χρησιμοποιήσαμε και κάποια χαρακτηριστικά κομμάτια από τον κώδικα της εφαρμογής.

### 4.2 Γενική εικόνα της εφαρμογής

Το app που κατασκευάσαμε έχει γενικά την δυνατότητα να δείχνει στον χρήστη μία σειρά από προεπιλεγμένες τοποθεσίες που συμβολίζουν κινηματογράφους, σούπερ μάρκετ και βενζινάδικα, να υπολογίζει τον χρόνο που θα του έπαιρνε για να φτάσει σε αυτές και να προβάλλει την καλύτερη διαδρομή.

Για αυτόν τον λόγο χρησιμοποιεί κατά κόρων την υπηρεσία Google Maps και το Google Directions. Το Google Maps μπορεί να χρησιμοποιηθεί μέσω μίας βιβλιοθήκης Java που είναι μέρος του πακέτου Google Play Services. Για το Google Directions δεν υπάρχει υλοποίηση για Android. Για αυτόν τον λόγο χρησιμοποιήσαμε μία Τρίτη βιβλιοθήκη η οποία πρακτικά κάνει κλήσεις στο REST API του Google directions.

Οι τοποθεσίες που αποθηκεύουμε είναι αρκετές σε πλήθος. Επιπλέον χρειάζεται να μπορούμε να κάνουμε έξυπνες αναζητήσεις σε αυτές βάση συγκεκριμένων κριτηρίων και να διατηρούνται ανάμεσα στις κλήσεις της εφαρμογής. Είναι ένα σενάριο χρήσης που ενδείκνυται για σχεσιακές βάσεις δεδομένων, και για αυτόν τον λόγο χρησιμοποιήσαμε την βάση SQLite που βρίσκεται ενσωματωμένη στο λειτουργικό Android και είναι διαθέσιμη σε κάθε εφαρμογή.

Για την επικοινωνία με τη βάση χρησιμοποιήσαμε το GreenDAO, μία βιβλιοθήκη ORM. Αυτή μας επιτρέπει να δηλώσουμε αρχικά τις ιδιότητες κάθε οντότητας και μετά να τις χρησιμοποιήσουμε σαν αντικείμενα Java, χωρίς να χρειάζεται να γράψουμε καθόλου κώδικα SQL. Πρόκειται για μία τεχνική που γλυτώνει αρκετό χρόνο κατά την ανάπτυξη μίας εφαρμογής και αν είναι απαραίτητο επιτρέπει την χρήση κανονικών ερωτημάτων SQL για τα πιο δύσκολα ερωτήματα.

### 4.3 Απαραίτητα εργαλεία

#### 4.3.1 Android SDK

Τα βασικά εργαλεία ανάπτυξης της εφαρμογής ονομάζονται Android SDK (Standard Development Kit) και δίνονται όλα συγκεντρωμένα από την εταιρεία Google. Αυτό περιλαμβάνει τις βιβλιοθήκες Java του Android που θα χρησιμοποιήσει η εφαρμογή, τον emulator για την δοκιμαστική εκτέλεση της εφαρμογής χωρίς να χρειάζεται να υπάρχει συσκευή (που παρεμπιπτόντως δεν τρέχει εφαρμογές με το Google Maps), τον compiler που παράγει τον κώδικα και μία πληθώρα διαφορετικών βιβλιοθηκών.

Το SDK δεν παρέχεται μόνο του αλλά έρχεται μαζί με ένα περιβάλλον ανάπτυξης. Αυτή τη στιγμή υπάρχουν δύο επιλογές, το eclipse και το Android Studio.

Το Android Studio είναι ένα περιβάλλον βασισμένο στο IntelliJ IDEA και είναι το περιβάλλον που αναπτύσσεται ενεργά από τη Google. Υποστηρίζει όλες τις δυνατότητες που παρέχει και το eclipse εκτός από το NDK.

## Android Studio vs. Eclipse ADT Comparison

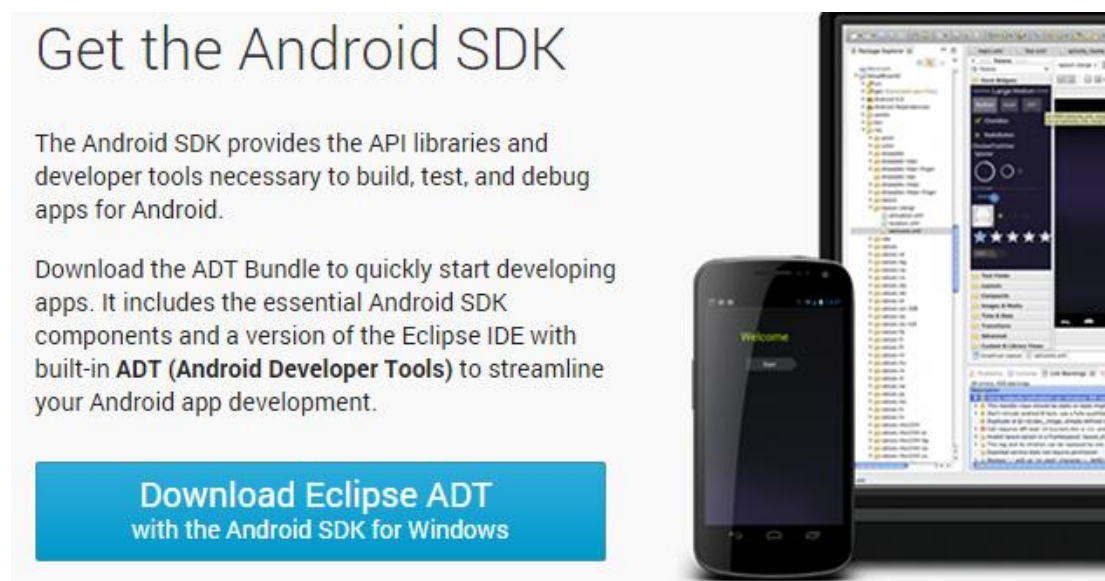
The following table lists some key differences between Android Studio Beta and [Eclipse with ADT](#).

Feature	Android Studio	ADT
Build system	Gradle <a href="#">☞</a>	Ant <a href="#">☞</a>
Maven-based build dependencies	Yes	No
Build variants and multiple-APK generation (great for Android Wear)	Yes	No
Advanced Android code completion and refactoring	Yes	No
Graphical layout editor	Yes	Yes
APK signing and keystore management	Yes	Yes
NDK support	Coming soon	Yes

Το eclipse ήταν παλιότερα το επίσημο εργαλείο συγγραφής κώδικα για Android αλλά έχει πλέον εγκαταλειφθεί. Τον Νοέμβριο τον 2014 η Google άρχισε να προτείνει επίσημα στους χρήστες την χρήση του, εκτός και αν τους ήταν απαραίτητη κάποια λειτουργία που δεν παρέχει.

Η έκδοση του ADT με το eclipse υπάρχει στη διεύθυνση:

<https://developer.android.com/sdk/index.html?hl=i>



Get the Android SDK

The Android SDK provides the API libraries and developer tools necessary to build, test, and debug apps for Android.

Download the ADT Bundle to quickly start developing apps. It includes the essential Android SDK components and a version of the Eclipse IDE with built-in **ADT (Android Developer Tools)** to streamline your Android app development.

[Download Eclipse ADT with the Android SDK for Windows](#)

Δυστυχώς είχαμε ξεκινήσει την ανάπτυξη της εφαρμογής μας πολύ νωρίτερα και η διαδικασία μετάβασης από το ένα περιβάλλον στο άλλο είναι επίπονη, οπότε συνεχίσαμε να χρησιμοποιούμε το eclipse.

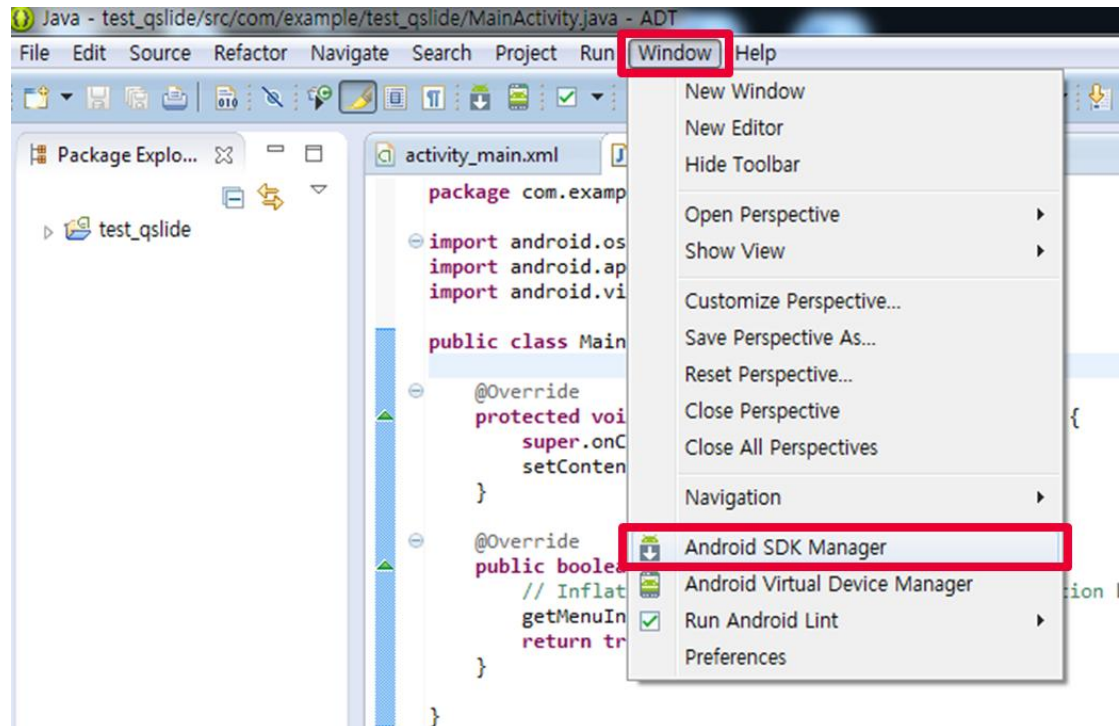
### 4.3.2 Εγκατάσταση πακέτων SDK

Όπως είδαμε στα προηγούμενα κεφάλαια το Android έχει πάρα πολλές διαφορετικές εκδόσεις. Κάθε μία από αυτές είναι και μία διαφορετική βιβλιοθήκη. Όταν γράφουμε μία εφαρμογή χρειάζεται να ορίσουμε την έκδοση του Android στην οποία θα

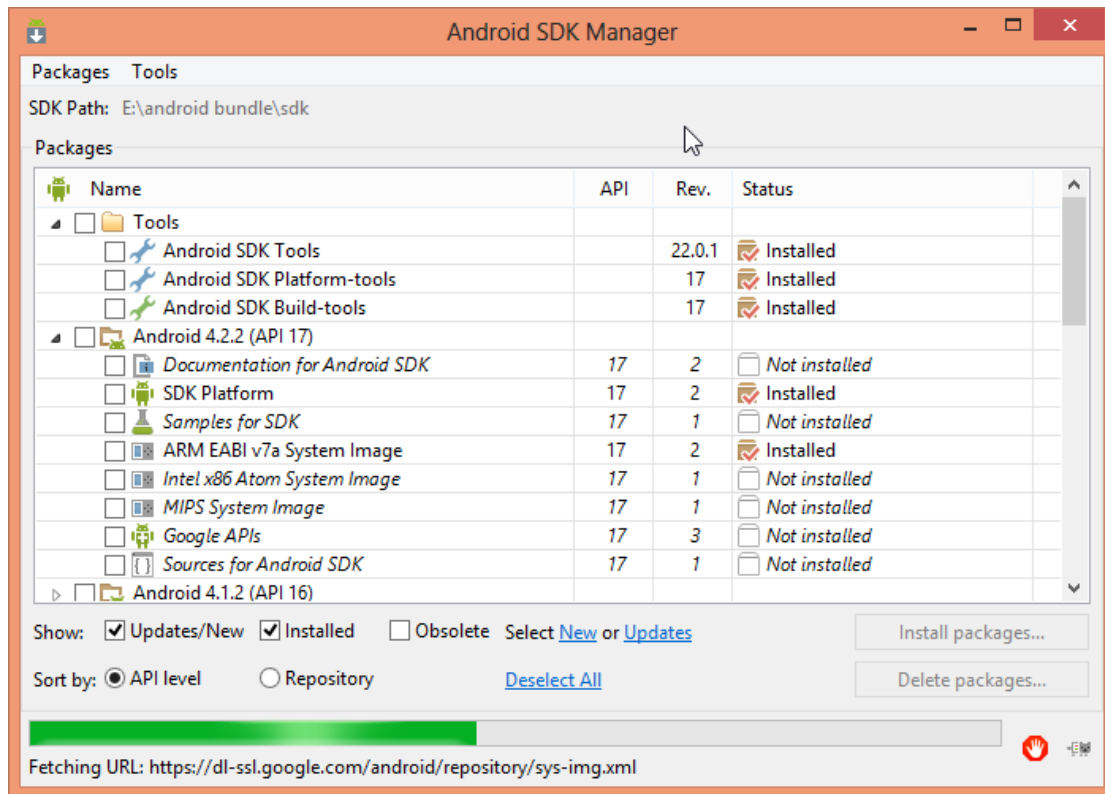
την κάνουμε compile και να έχουμε εγκαταστημένη την αντίστοιχη έκδοση των βιβλιοθηκών στον υπολογιστή μας.

Καθώς δεν είναι εφικτό να είναι ταυτόχρονα εγκαταστημένες όλες οι εκδόσεις του λειτουργικού, υπάρχει η δυνατότητα να διαχειριστούμε ποιες από αυτές θέλουμε να κατεβάσουμε. Αυτό γίνεται με τη χρήση του Android SDK Manager.

Μπορούμε να εκτελέσουμε το πρόγραμμα μέσα από το eclipse.



Μέσα σε αυτό βλέπουμε μία λίστα με τις διαφορετικές εκδόσεις των βιβλιοθηκών και διάφορα εργαλεία για την κάθε μία. Εμείς επιλέξαμε την έκδοση 5.0.



Αφού ολοκληρώσουμε τις επιλογές μας πατάμε install packages για να ξεκινήσει η διαδικασία εγκατάστασης.

### 4.3.3 Επιπλέον βιβλιοθήκες

Για την λειτουργία της εφαρμογής μας χρειάζεται προφανώς να έχουμε εγκαταστημένη τη Java και συγκεκριμένα το Java SE SDK, το οποίο είναι διαθέσιμο εδώ:

<http://www.oracle.com/technetwork/articles/javase/index-jsp-138363.html>

## Java SE Downloads



Java Platform (JDK) 8u25



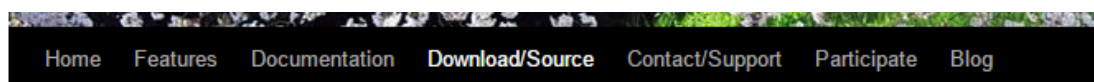
JDK 8u25 & NetBeans 8.0.1

Java Platform, Standard Edition	
<b>Java SE 8u25</b> This release includes important security fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release. <a href="#">Learn more</a> ▶	
<ul style="list-style-type: none"><li>Installation Instructions</li><li>Release Notes</li><li>Oracle License</li><li>Java SE Products</li><li>Third Party Licenses</li><li>Certified System Configurations</li><li>Readme Files<ul style="list-style-type: none"><li>JDK ReadMe</li><li>JRE ReadMe</li></ul></li></ul>	<b>JDK</b> <a href="#">DOWNLOAD</a> ↓
	<b>Server JRE</b> <a href="#">DOWNLOAD</a> ↓
	<b>JRE</b> <a href="#">DOWNLOAD</a> ↓

Επιπλέον χρησιμοποιήσαμε την βιβλιοθήκη GreenDAO για τον χειρισμό των αντικειμένων που αποθηκεύουμε στη βάση. Το GreenDAO είναι μία βιβλιοθήκη ανοικτού κώδικα, το οποίο είναι διαθέσιμο από τη διεύθυνση:

<http://greendao-orm.com/download-and-source/>

Είναι απαραίτητα δύο αρχεία, το greendao.jar και το greendao-generator.jar.



## Download/Source

The preferred approach to include greenDAO is referencing greenDAO using Maven or Gradle. Add dependencies to group ID “de.greenrobot” with the artifact IDs [greendao](#) and [greendao-generator](#) to add greenDAO to your project.

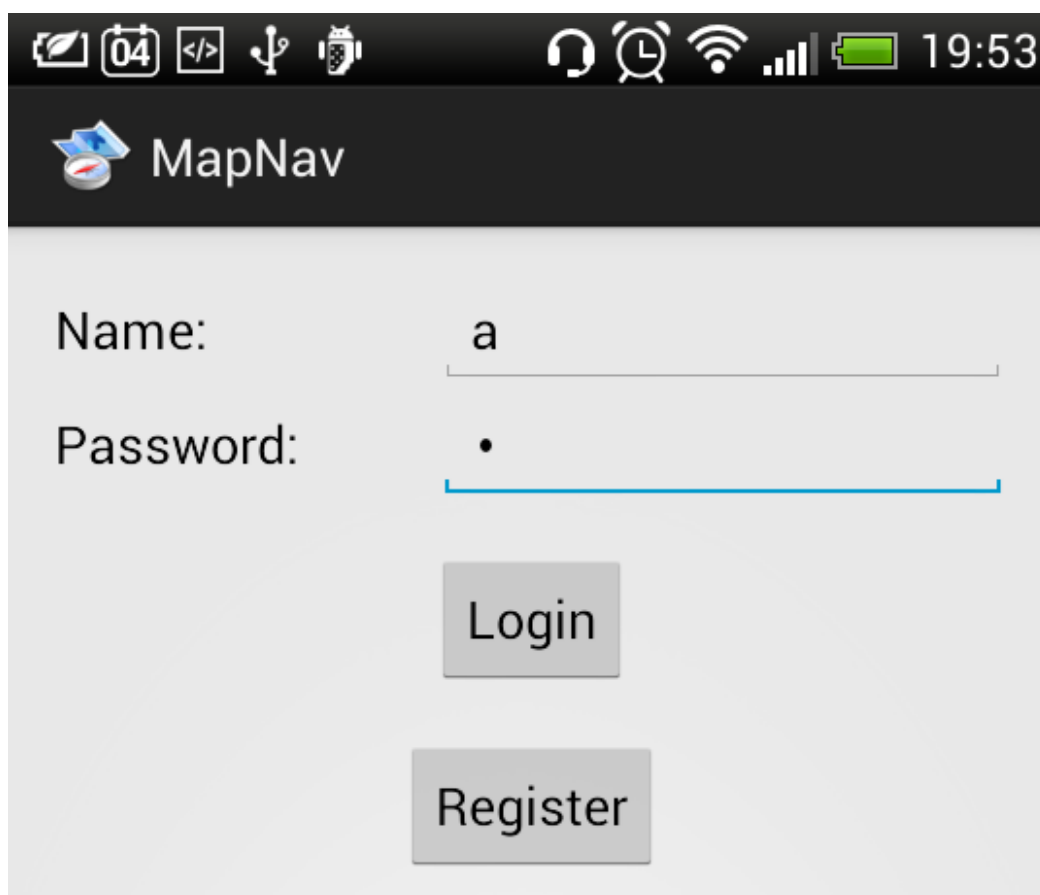
## 5 Οδηγίες χρήσης

### 5.1 Εισαγωγή

Σε αυτή την ενότητα θα δώσουμε γενικές οδηγίες για την χρήση της εφαρμογής μας.

### 5.2 Είσοδος

Με το που ξεκινάει η εφαρμογή ο χρήστης βλέπει την οθόνη εισόδου.



Εδώ μπορεί να εισάγει τον κωδικό του και να μπει στο σύστημα πατώντας. Σε περίπτωση που δεν έχει κωδικό μπορεί να πατήσει Register για να φτιάξει έναν καινούριο λογαριασμό.

### 5.3 Δημιουργία λογαριασμού

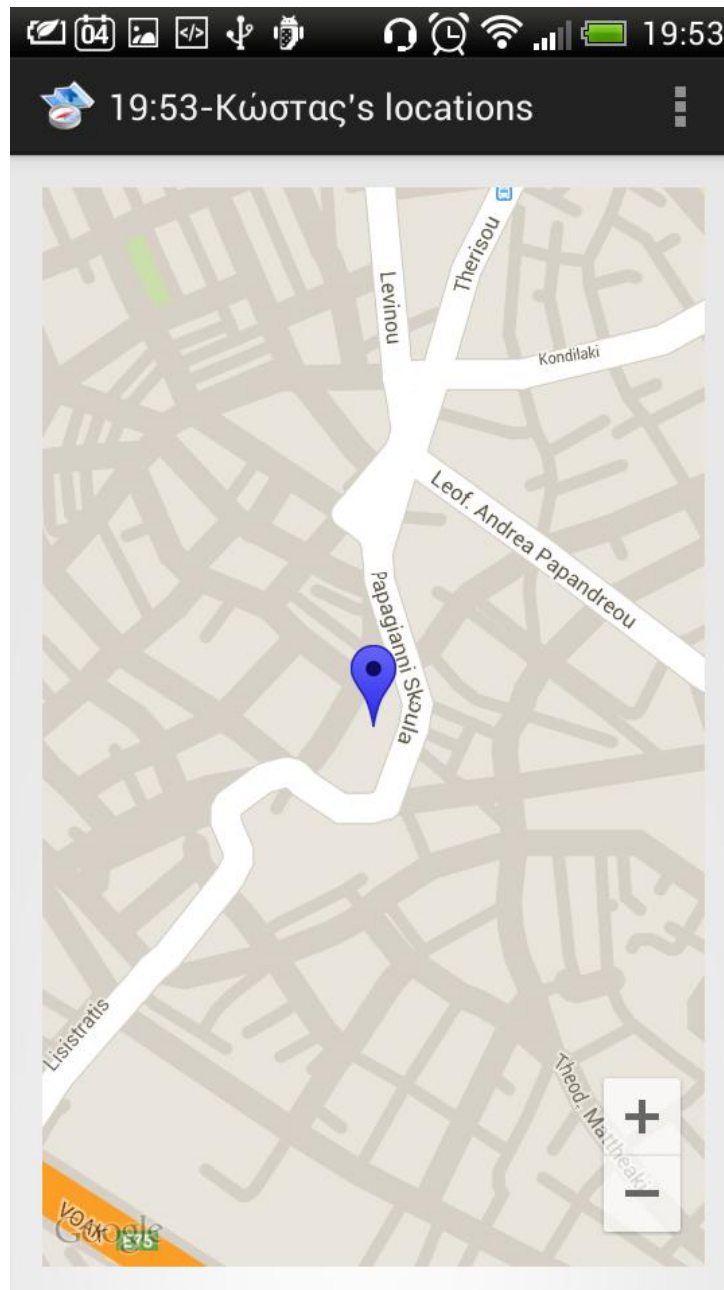
Αν ο χρήστης δεν έχει λογαριασμό μπορεί να πατήσει Register στην αρχική οθόνη για να φτιάξει έναν νέο λογαριασμό.

The image shows a screenshot of an Android application interface. At the top, there is a status bar with various icons and the time 19:53. Below the status bar is a dark header with a folder icon and the text 'RegisterActivity'. The main content area is light gray and contains four text input fields labeled 'Username:', 'Password:', 'Password (again):', and 'Name:'. A blue cursor is visible in the 'Username' field. Below the fields is a gray button labeled 'Register'. At the bottom of the screen, a virtual keyboard is visible, showing the letters 'q', 'w', 'e', 'r', 't', 'y', 'u', 'i', 'o', 'p' on the top row, and other keys like 'a', 's', 'd', 'f', 'g', 'h', 'j', 'k', 'l' on the second row.

Εδώ πρέπει να εισάγει τα στοιχεία που επιθυμεί. Αν δεν έχει βάλει σωστά τον κωδικό ή αν έχει βάλει όνομα χρήστη που χρησιμοποιείται ήδη ενημερώνεται για να βάλει τα σωστά στοιχεία. Αν έχει συμπληρώσει σωστά τη φόρμα πηγαίνει πίσω.

#### 5.4 Βασική οθόνη χάρτη

Αφού ο χρήστης μπει στην εφαρμογή βάζοντας τα στοιχεία του βλέπει τον χάρτη.

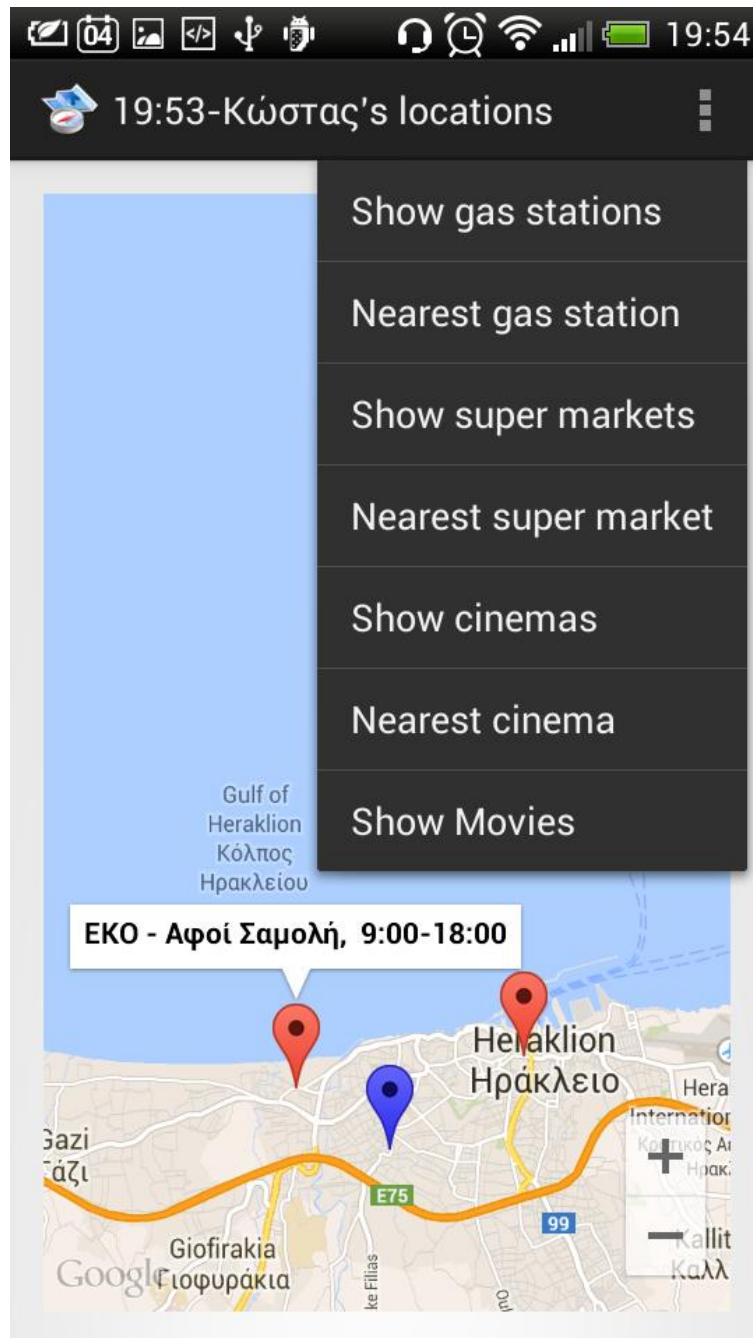


Φαίνεται σημειωμένη η θέση του. Αν επιθυμεί μπορεί να σύρει την μπλε ακίδα (αφού προηγουμένως την πατήσει για λίγη ώρα) ώστε να αλλάξει την θέση του. Η νέα θέση θα αποθηκευτεί αυτόματα.

## 5.5 Επιλογή λειτουργιών

Πατώντας στην μπάρα λειτουργιών της εφαρμογής εμφανίζεται ένα μενού με τις διαθέσιμες ενέργειες.

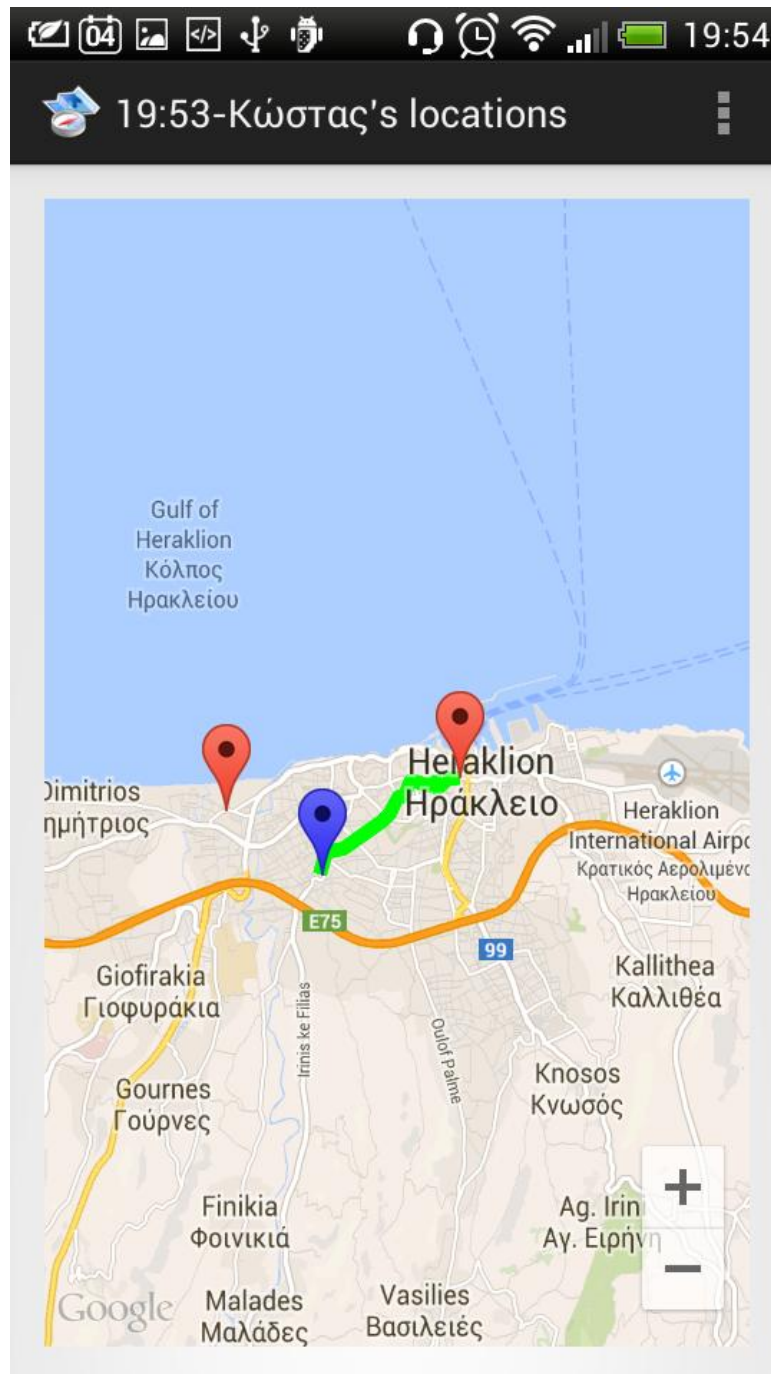




Από εδώ μπορεί να διαλέξει για να δει κάποιες από τις τοποθεσίες που είναι αποθηκευμένες στην εφαρμογή. Πατώντας πάνω στην κάθε τοποθεσία εμφανίζεται το όνομά της.

## 5.6 Εμφάνιση καλύτερης διαδρομής

Αν ο χρήστης πατήσει το κατάλληλο κουμπί από το μενού ενεργειών θα εμφανιστεί η κοντινότερη τοποθεσία αυτού του είδους.



Η απόσταση υπολογίζεται από το Google Directions και λαμβάνει αυτόματα υπόψη τις τις οδηγικές συνθήκες.

## 6 Συμπεράσματα

### 6.1 Εφαρμογή και περιβάλλον Android

Η ανάπτυξη μία εφαρμογής Android ήταν μια διαδικασία που μας άφησε ανάμεικτες εντυπώσεις.

Από την μία η ανάπτυξη μίας βασικής εφαρμογής είναι σχετικά απλή, με την εταιρεία Google να προσφέρει το απαραίτητο documentation που καθοδηγεί τον προγραμματιστή ώστε να ξεκινήσει το συντομότερο να γράφει κώδικα.

Όμως η διαδικασία δεν είναι τόσο απλή όσο φαίνεται αρχικά. Αρχικά, το documentation δεν καλύπτει αναλυτικά διάφορα θέματα, κάτι το οποίο γίνεται εμφανές όταν προσπαθήσουμε να μπούμε σε λεπτομέρειες και να υλοποιήσουμε λειτουργίες που δεν εμπίπτουν στα παραδείγματα που δίνονται.

Επιπλέον η ενσωμάτωση των χαρτών ήταν μία επίπονη διαδικασία. Καθώς οι χάρτες ανήκουν στην ομάδα υπηρεσιών του Google Play χρειάζεται επιπλέον κόπος για τη σωστή χρήση τους. Ένα αρνητικό στοιχείο που συναντήσαμε ήταν ότι κατά τη διάρκεια της ανάπτυξης η Google άλλαξε το API επικοινωνίας με το Google Maps. Το παλιό API έδινε λιγότερες δυνατότητες αλλά ήταν μέρος των βασικών λειτουργιών του συστήματος. Το νέο API δίνει περισσότερες δυνατότητες αλλά είναι μέρος του Google Play Services.

Κατά την πορεία της μετάβασης άλλαξε και το αντίστοιχο τμήμα των σελίδων, κάνοντας ιδιαίτερα δύσκολη την εύρεση υλικού για το παλιό API, και πρακτικά μας ανάγκασε να αλλάξουμε την υλοποίησή μας για να χρησιμοποιεί την έκδοση 2.

Ένα άλλο εμπόδιο το οποίο υπάρχει στην ανάπτυξη των εφαρμογών ήταν η συμβατότητα με τις παλιότερες εκδόσεις του λειτουργικού. Αν και η τρέχουσα έκδοση είναι η 4.4, με την 5.0 να έχει εμφανιστεί σε ελάχιστες συσκευές, η πλειοψηφία των συσκευών χρησιμοποιεί εκδόσεις πολύ παλιότερες. Για την εφαρμογή που αναπτύσσαμε η συμβατότητα με παλιότερο hardware δεν ήταν απαραίτητη, οπότε αν και αρχικά προσπαθήσαμε να διατηρήσουμε τη συμβατότητα με μεγάλο πλήθος παλιότερων συσκευών, σύντομα αναγκαστήκαμε να ορίσουμε σαν ελάχιστη υποστηριζόμενη έκδοση την 4, καθώς είδαμε πως ο όγκος δουλειάς για υποστήριξη παλιότερων εκδόσεων ήταν τόσο μεγάλος που ήταν εκτός των ορίων της εργασίας.

Τέλος ένα απρόσμενο εμπόδιο που συναντήσαμε ήταν η μετάβαση των εργαλείων ανάπτυξης του Google από το περιβάλλον eclipse στο Android Studio. Κατά τη διάρκεια της υλοποίησής μας έπαψε η εξέλιξη των εργαλείων του eclipse ενώ το Android Studio δεν είχε ολοκληρωθεί. Αυτό δημιούργησε θέματα με την νέα έκδοση του Android API.

### 6.2 Μελλοντική επέκταση

Κατά την ανάπτυξη της εφαρμογής είδαμε αρκετά σημεία που μπορούν να επεκταθούν παραπέρα.

Ένας αντικείμενο που αγνοήσαμε εσκεμμένα ήταν η συμβατότητα με παλιότερες εκδόσεις. Ο όγκος των παραλλαγών που υπάρχουν είναι αρκετά μεγάλος, όμως αυτό είναι ένα αντικείμενο που δεν σχετίζεται ιδιαίτερα με τη συγκεκριμένη εφαρμογή και θα μπορούσε να αποτελέσει ένα ανεξάρτητο θέμα από μόνο του.

Η εφαρμογή μας έχει καλύψει τον τομέα του Android σχετικά με αυτά τα αντικείμενα. Ένας ενδιαφέρων τρόπος σύγκρισης θα ήταν η υλοποίηση της ίδιας εφαρμογής στις άλλες δύο δημοφιλείς πλατφόρμες κινητών τηλεφώνων, στο iOS και στο Windows Phone. Με αυτόν τον τρόπο θα μπορούσε να γίνει σύγκριση των δυνατοτήτων των τριών λειτουργικών. Καθώς κάθε ένα από αυτά χρησιμοποιεί διαφορετική (βασική) υπηρεσία για την παροχή των χαρτών θα γίνει κα μία σύγκριση ανάμεσα στις τρεις βασικές υπηρεσίες, του Google Maps, του Apple Maps και του Bing Maps.

Μία άλλη ιδέα για την επέκταση της εφαρμογής είναι η ενσωμάτωση υπηρεσιών κοινωνικής δικτύωσης, για παράδειγμα ο χρήστης να μπορεί να δει τις θέσεις των φίλων του στο Facebook ή στο Google Plus. Μία τέτοια εφαρμογή όμως παρουσιάζει διάφορα θέματα προστασίας προσωπικών υπηρεσιών και χρειάζεται περισσότερη μελέτη και προετοιμασία.

## 7 Βιβλιογραφία

Android Architecture – The Key Concepts of Android OS. *www.android-app-market.com*. [Ηλεκτρονικό] <http://www.android-app-market.com/android-architecture.html>.

Android Camera API - Tutorial. [Ηλεκτρονικό] <http://www.vogella.com/tutorials/AndroidCamera/article.html>.

Android Developers. *android.com*. [Ηλεκτρονικό] <http://developer.android.com/index.html>.

com.google.android.gms.maps. [Ηλεκτρονικό] <http://developer.android.com/reference/com/google/android/gms/maps/package-summary.html>.

Definitions of Smartphone. *Pc Magazine Encyclopedia*. [Ηλεκτρονικό] [http://www.pcmag.com/encyclopedia\\_term/0,2542,t=Smartphone&i=51537,00.asp#fbid=Ps mZ](http://www.pcmag.com/encyclopedia_term/0,2542,t=Smartphone&i=51537,00.asp#fbid=Ps mZ).

Developer Tools. *developer.android.com*. [Ηλεκτρονικό] <http://developer.android.com/tools/index.html>.

Get driving directions using Google Maps API v2. [Ηλεκτρονικό] <http://stackoverflow.com/questions/14495030/get-driving-directions-using-google-maps-api-v2>.

Get the Android SDK. *developer.android.com*. [Ηλεκτρονικό] <http://developer.android.com/sdk/index.html>.

Getting distance from Google Direction API in Android. [Ηλεκτρονικό] <http://stackoverflow.com/questions/20979853/getting-distance-from-google-direction-api-in-android>.

Google Maps Android API v2. [Ηλεκτρονικό] <https://developers.google.com/maps/documentation/android/>.

Google-Directions-Android. [Ηλεκτρονικό] <https://github.com/jd-alexander/Google-Directions-Android>.

greenDAO – Android ORM for SQLite. [Ηλεκτρονικό] <http://greendao-orm.com/>.

greenrobot/greenDAO. [Ηλεκτρονικό] <https://github.com/greenrobot/greenDAO>.

Guide: Google Maps V2 for Android: Draw Driving Direction on Map. [Ηλεκτρονικό] <http://blog-emildesign.rhcloud.com/?p=822>.

iPhone. *Βικιπαίδεια*. [Ηλεκτρονικό] <http://el.wikipedia.org/wiki/IPhone>.

Modelling entities. [Ηλεκτρονικό] <http://greendao-orm.com/documentation/modelling-entities/>.

ORM performance: is greenDAO faster than ORMLite? [Ηλεκτρονικό]  
<http://stackoverflow.com/questions/20079568/orm-performance-is-greendao-faster-than-ormlite>.

Why retrieving Google Directions for Android using KML data is not working anymore? [Ηλεκτρονικό] <http://stackoverflow.com/questions/11745314/why-retrieving-google-directions-for-android-using-kml-data-is-not-working-anymo/11745316#11745316>.

Έξυπνο τηλέφωνο. Βικιπαίδεια. [Ηλεκτρονικό]  
[http://el.wikipedia.org/wiki/%CE%88%CE%BE%CF%85%CF%80%CE%BD%CE%BF\\_%CF%84%CE%B7%CE%BB%CE%AD%CF%86%CF%89%CE%BD%CE%BF](http://el.wikipedia.org/wiki/%CE%88%CE%BE%CF%85%CF%80%CE%BD%CE%BF_%CF%84%CE%B7%CE%BB%CE%AD%CF%86%CF%89%CE%BD%CE%BF).

Ορολογία του Android. *myphone.gr*. [Ηλεκτρονικό]  
<http://www.myphone.gr/forum/showthread.php?t=306146>.

## 8 Παράρτημα Α – Κώδικας εφαρμογής

### 8.1 AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="gr.epp.mapnav"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="14"
        android:targetSdkVersion="21" />

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

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission
android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission
android:name="com.google.android.providers.gsf.permission.READ_GSERV
ICES" />
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION" />

    <permission
        android:name="gr.epp.mapnav.permission.MAPS_RECEIVE"
        android:protectionLevel="signature" />

    <uses-permission
android:name="gr.epp.mapnav.permission.MAPS_RECEIVE" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".LoginActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

```
        </intent-filter>
    </activity>
    <activity
        android:name=".RegisterActivity"
        android:label="@string/title_activity_register" >
    </activity>

    <meta-data
        android:name="com.google.android.gms.version"
        android:value="@integer/google_play_services_version" />
    <meta-data
        android:name="com.google.android.maps.v2.API_KEY"
        android:value="AIzaSyAWWoVSlhyS4did6I7nC95DoibwL8rKZG0"
    />

    <activity
        android:name=".DefaultMapActivity"
        android:label="@string/title_activity_default_map" >
    </activity>
    <activity
        android:name=".MovieActivity"
        android:label="@string/title_activity_movie" >
    </activity>
</application>

</manifest>
```

## 8.2 Utils.java

```
package gr.epp.mapnav.util;

import android.annotation.SuppressLint;
import android.content.Context;
import android.widget.Toast;

public class Utils {

    public static boolean isEmpty(String... str) {
        if (str == null) {
            return true;
        }

        for (String string : str) {
            if(string == null || string.length() == 0) {
                return true;
            }
        }

        return false;
    }

    public static Toast toast(Context context, String message) {
        return Toast.makeText(context, message,
            Toast.LENGTH_SHORT);
    }
}
```



```
    }

    @SuppressWarnings("DefaultLocale")
    public static String formatTime(int timeSec) {
        String time;
        int minutes = (timeSec%3600)/60;

        if (minutes < 10) {
            time = String.format("%2d:0%d", timeSec/3600,
minutes);
        } else {
            time = String.format("%2d:%d", timeSec/3600,
minutes);
        }

        return time;
    }
}
```

### 8.3 AbstractMapNavActivity.java

```
package gr.epp.mapnav;

import gr.epp.mapnav.dao.User;
import gr.epp.mapnav.data.GlobalData;
import android.app.Activity;
import android.view.Menu;
import android.view.MenuItem;

public abstract class AbstractMapNavActivity extends Activity {

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
//        getMenuInflater().inflate(R.menu.register, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    public User user() {
        return GlobalData.currentUser();
    }
}
```

## 8.4 LoginActivity.java

```
package gr.epp.mapnav;

import gr.epp.mapnav.dao.User;
import gr.epp.mapnav.data.ContentManager;
import gr.epp.mapnav.data.GlobalData;
import gr.epp.mapnav.util.Utils;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;

public class LoginActivity extends AbstractMapNavActivity {
    protected EditText userNameField, passwordField;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        ContentManager.init(getApplicationContext());

        setContentView(R.layout.activity_login);

        userNameField = (EditText)
findViewById(R.id.loginUsernameField);
        passwordField = (EditText)
findViewById(R.id.loginPasswordField);
    }

    protected void initUserFields() {
        User user = GlobalData.currentUser();

        if (user != null) {
            userNameField.setText(user.getUsername());
            passwordField.setText(user.getPassword());
        }
    }

    @Override
    protected void onStart() {
        super.onStart();
        initUserFields();
    }

    public void loginClick(View view) {
        String username = userNameField.getText().toString();
        String password = passwordField.getText().toString();

        if(Utils.isEmpty(username, password)) {
            Utils.toast(this, "Please fill in every
field").show();
            return;
        }
    }
}
```

```
        User user = ContentManager.instance().findUser(username,
password);

        if (user == null) {
            Utils.toast(this, "Login failed").show();
        } else {
            Utils.toast(this, "Login successful!").show();
            GlobalData.setCurrentUser(user);

            Intent intent = new
Intent(getApplicationContext(), DefaultMapActivity.class);
            startActivity(intent);
        }
    }

    public void registerClick(View view) {
        Intent intent = new Intent(getApplicationContext(),
RegisterActivity.class);
        startActivity(intent);
    }
}
```

## 8.5 RegisterActivity.java

```
package gr.epp.mapnav;

import gr.epp.mapnav.dao.User;
import gr.epp.mapnav.data.ContentManager;
import gr.epp.mapnav.data.GlobalData;
import gr.epp.mapnav.util.Utils;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;

public class RegisterActivity extends AbstractMapNavActivity {
    protected EditText usernameField, passwordField,
passwordConfirmField, nameField;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);

        usernameField = (EditText)
findViewById(R.id.registerUsernameField);
        passwordField = (EditText)
findViewById(R.id.registerPasswordField);
        passwordConfirmField = (EditText)
findViewById(R.id.registerPasswordConfirmField);
        nameField = (EditText)
findViewById(R.id.registerNameField);
    }
}
```

```
public void registerClick(View view) {
    String username = usernameField.getText().toString();
    String name = nameField.getText().toString();
    String password = passwordField.getText().toString();
    String passwordConfirm =
passwordConfirmField.getText().toString();

    if(Utils.isEmpty(username, password, passwordConfirm,
name)) {
        Utils.toast(this, "Please fill in every
field").show();
        return;
    }

    if(!password.equals(passwordConfirm)) {
        Utils.toast(this, "Make sure you have entered the
correct password").show();
        return;
    }

    if(ContentManager.instance().findUser(username) != null)
{
        Utils.toast(this, "This username is already
taken").show();
        return;
    }

    User user =
ContentManager.instance().createUser(username, passwordConfirm,
name);
    GlobalData.setCurrentUser(user);

    Utils.toast(this, "Success!").show();
    finish();
}
}
```

## 8.6 GlobalData.java

```
package gr.epp.mapnav.data;

import gr.epp.mapnav.dao.Location;
import gr.epp.mapnav.dao.User;

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

public class GlobalData {
    protected static User currentUser;
    public static final LatLng HERAKLION_LL = new LatLng(35.33963,
25.13077);
}
```

```
// public static final Location HERAKLION = new Location(0L, "",
35.33963, 25.13077);
    public static final float DEFAULT_ZOOM = 16;

    public static final int SUPER_MARKET_DELAY_MIN = 45;

    public static User currentUser() {
        return currentUser;
    }

    public static void setCurrentUser(User currentUser) {
        GlobalData.currentUser = currentUser;
    }

    public static Location defaultLocation(User user) {
        Location location = new Location();
        location.setName(user.getName() + "'s default
location");
        location.setType(LType.USER);

        location.setLat(HERAKLION_LL.latitude);
        location.setLng(HERAKLION_LL.longitude);

        location.setOpeningTime(-1);
        location.setClosingTime(-1);

        return location;
    }

    public static LatLng coords(Location location) {
        LatLng latLng = new LatLng(location.getLat(),
location.getLng());

        return latLng;
    }
}
```

## 8.7 DefaultMapActivity.java

```
package gr.epp.mapnav;

import gr.epp.mapnav.dao.Location;
import gr.epp.mapnav.dao.Movie;
import gr.epp.mapnav.data.ContentManager;
import gr.epp.mapnav.data.GlobalData;
import gr.epp.mapnav.data.LType;
import gr.epp.mapnav.util.Utills;

import java.util.ArrayList;
import java.util.Calendar;
import java.util.Collections;
```

```
import java.util.Comparator;

import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;

import com.directions.route.Route;
import com.directions.route.Routing;
import com.directions.route.RoutingListener;
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.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.maps.model.PolylineOptions;

public class DefaultMapActivity extends AbstractMapNavActivity {
    public static final int REQUEST_CINEMA = 1;

    protected GoogleMap map;
    protected Marker userMarker;
    protected ArrayList<Location> currentLocations = new
ArrayList<Location>();
    protected ArrayList<RouteInfo> routeInfos = new
ArrayList<RouteInfo>();

    protected int routesRemaining = 0;
    protected int currentRouteType = -1;
    protected int currentTimeSec = -1;

    GoogleMap.OnMarkerDragListener userMarkerDragListener = new
GoogleMap.OnMarkerDragListener() {

        @Override
        public void onMarkerDragStart(Marker marker) {
            // Empty
        }

        @Override
        public void onMarkerDragEnd(Marker marker) {
            if(marker.equals(userMarker)) {

                ContentManager.instance().updateLocation(user().getLocation(),
marker.getPosition());
            }
        }

        @Override
```

```
        public void onMarkerDrag(Marker marker) {
            // Empty
        }
    };

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_default_map);

        currentTimeSec =
        Calendar.getInstance().get(Calendar.HOUR_OF_DAY) * 3600 +
        Calendar.getInstance().get(Calendar.MINUTE) * 60;

        updateTitle();

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

        initMapMarkers();

        // map.setOnMapLongClickListener(new
        GoogleMap.OnMapLongClickListener() {
        //
        //         @Override
        //         public void onMapLongClick(LatLng position) {
        //             setMapMarker(position);
        //         }
        //     });

        map.setOnMarkerDragListener(userMarkerDragListener);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.default_map, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.action_list_super_markets) {
            showMarkers(LType.SUPER_MARKET);

            return true;
        }

        if (id == R.id.action_list_cinemas) {
            showMarkers(LType.CINEMA);

            return true;
        }
    }
}
```

```
        if (id == R.id.action_list_gas_stations) {
            showMarkers(LType.GAS_STATION);

            return true;
        }

        if (id == R.id.action_nearest_gas_station) {
            findBestRoute(LType.GAS_STATION);

            return true;
        }

        if (id == R.id.action_nearest_super_market) {
            findBestRoute(LType.SUPER_MARKET);

            return true;
        }

        if (id == R.id.action_show_movies) {
            Intent intent = new
Intent(getApplicationContext(), MovieActivity.class);
            startActivityForResult(intent, REQUEST_CINEMA);

            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onActivityResult(int requestCode, int resultCode,
Intent data) {
        switch(requestCode) {
            case (REQUEST_CINEMA) : {
                if (resultCode == RESULT_OK) {
                    Long movieId = data.getLongExtra("movieId", -1);

                    Log.i("ROUT", "forResult: " + movieId);

                    findBestRoute(ContentManager.instance().findMovie(movieId));

                    return;
                }
                break;
            }
        }
        super.onActivityResult(requestCode, resultCode, data);
    }

    public void updateTitle() {
        setTitle( Utils.formatTime(currentTimeSec) + "-" +
user().getName() + "'s locations");
    }
}
```



```
    }

    public void initMapMarkers() {
        map.clear();
        showUserMarker();

        map.moveCamera(CameraUpdateFactory.newLatLngZoom(userMarker.ge
tPosition(), GlobalData.DEFAULT_ZOOM));
    }

    public void showUserMarker() {
        userMarker = map.addMarker(
            new MarkerOptions()

                .position(GlobalData.coords(GlobalData.currentUser().getLocati
on()))

                .draggable(true)

                .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFa
ctory.HUE_BLUE))

                .title(GlobalData.currentUser().getName() +
"'s location")
        );
    }

    public void showMarkers(int type) {
        currentLocations.clear();

        currentLocations.addAll(ContentManager.instance().findLocation
s(type));

        map.clear();
        showUserMarker();
        for(Location loc: currentLocations) {
            String text;

            if(type == LType.CINEMA) {
                text = loc.getName();
            } else {
                text = loc.getName()
                    + ", " +
Utils.formatTime(loc.getOpeningTime())
                    + "-" +
Utils.formatTime(loc.getClosingTime());
            }

            map.addMarker(
                new MarkerOptions()
                    .position(GlobalData.coords(loc))
                    .title(text)
            );
        }
    }
}
```

```
        map.animateCamera(CameraUpdateFactory.newLatLngZoom(GlobalData
        .HERAKLION_LL, GlobalData.DEFAULT_ZOOM - 4));
    }

    public void showMarkers(Movie movie) {
        currentLocations.clear();

        currentLocations.addAll(ContentManager.instance().findCinemas(
        movie.getName()));

        map.clear();
        showUserMarker();
        for(Location loc: currentLocations) {
            String text;

            text = movie.getName()
                + ", " +
        Utils.formatTime(loc.getOpeningTime())
                + "-" +
        Utils.formatTime(loc.getClosingTime());

            map.addMarker(
                new MarkerOptions()
                .position(GlobalData.coords(loc))
                .title(text)
            );
        }

        map.animateCamera(CameraUpdateFactory.newLatLngZoom(GlobalData
        .HERAKLION_LL, GlobalData.DEFAULT_ZOOM - 4));
    }

    public void findBestRoute (int type) {
        showMarkers(type);
        currentRouteType = type;
        LatLng start = GlobalData.coords(user().getLocation());

        routeInfos.clear();
        routesRemaining = currentLocations.size();

        for(Location loc: currentLocations) {
            Log.i("ROUT", loc.getName() + ", " + loc.getLat()
            + ", " + loc.getLng());
            LatLng end = GlobalData.coords(loc);

            Routing routing = new
        Routing(Routing.TravelMode.WALKING);
            routing.registerListener(new RouteInfo(loc));
            routing.execute(start, end);
        }
    }
}
```

```
    }

    public void findBestRoute (Movie movie) {
        showMarkers(movie);
        currentRouteType = LType.MOVIE;
        LatLng start = GlobalData.coords(user().getLocation());

        routeInfos.clear();
        routesRemaining = currentLocations.size();

        for(Location loc: currentLocations) {
            Log.i("ROUT", loc.getName() + ", " + loc.getLat()
+ ", " + loc.getLng());
            LatLng end = GlobalData.coords(loc);

            Routing routing = new
Routing(Routing.TravelMode.DRIVING);
            routing.registerListener(new RouteInfo(loc, movie));
            routing.execute(start, end);
        }
    }

    protected void showBestRoute() {
        RouteInfo bestRoute = null;

        Collections.sort(routeInfos, new Comparator<RouteInfo>()
{

            @Override
            public int compare(RouteInfo a, RouteInfo b) {
                return a.route.getDuration() -
b.route.getDuration();
            }
        });

        for(RouteInfo routeInfo: routeInfos) {

            int duration = routeInfo.route.getDuration();
            Location dest = routeInfo.location;
            int openingTime = dest.getOpeningTime();
            int closingTime = dest.getClosingTime();

            int arrivalTime = currentTimeSec + duration;

            Log.i("ROUT", "Validate: " + duration + "-" +
openingTime + "-" + closingTime + "-" + arrivalTime);

            if(currentRouteType == LType.GAS_STATION) {
                if(arrivalTime > openingTime && arrivalTime
< closingTime) {
                    bestRoute = routeInfo;
                    break;
                }
            }
        }
    }
}
```

```
        }

        if(currentRouteType == LType.SUPER_MARKET) {
            if(arrivalTime > openingTime && arrivalTime
+ GlobalData.SUPER_MARKET_DELAY_MIN * 60 < closingTime) {
                bestRoute = routeInfo;
                break;
            }
        }

        if (currentRouteType == LType.MOVIE) {
            openingTime =
routeInfo.movie.getOpeningTime();
            closingTime =
routeInfo.movie.getClosingTime();

            Log.i("ROUT-MOV", "Validate: " + duration +
"- " + openingTime + "- " + closingTime + "- " + arrivalTime);

            if (arrivalTime < openingTime) {
                bestRoute = routeInfo;
                break;
            }
        }
    }

    if(bestRoute != null) {
        drawRoute(bestRoute.route);
        Utils.toast(this, bestRoute.location.getName() +
"!");

        Log.i("ROUT", "bestRoute found");
    } else {
        Utils.toast(this, "No route found");
        Log.i("ROUT", "No bestRoute found");
    }
}

protected void drawRoute(Route route) {
    PolylineOptions polyoptions = new PolylineOptions();
    polyoptions.color(Color.GREEN);
    polyoptions.width(10);
    polyoptions.addAll(route.getPolylineOptions().getPoints());
    map.addPolyline(polyoptions);
}

class RouteInfo implements RoutingListener {
    protected Location location;
    protected Movie movie;
    protected Route route;

    public RouteInfo(Location location) {
        this.location = location;
    }
}
```

```
    }

    public RouteInfo(Location location, Movie movie) {
        this.location = location;
        this.movie = movie;
    }

    @Override
    public void onRoutingSuccess(Route route) {
        Log.i("ROUT", "Success: "+ route.getDuration());

        this.route = route;

        routeInfos.add(this);
        routesRemaining--;

        if(routesRemaining == 0) {
            showBestRoute();
        }
    }

    @Override
    public void onRoutingStart() {
    }

    @Override
    public void onRoutingFailure() {
        Utils.toast(DefaultMapActivity.this, "Failed to
contact Directions server!");
        routesRemaining--;
    }
}
}
```

## 8.8 MovieActivity.java

```
package gr.epp.mapnav;

import gr.epp.mapnav.dao.Movie;
import gr.epp.mapnav.data.ContentManager;
import gr.epp.mapnav.util.Utils;

import java.util.ArrayList;
import java.util.List;

import android.content.Context;
import android.content.Intent;
```

```
import android.os.Bundle;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;

public class MovieActivity extends AbstractMapNavActivity {
    protected List<Movie> movies;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_movie);

        ListView movieListView = (ListView)
        findViewById(R.id.movieListView);

        movies = ContentManager.instance().findMovies();

        List<String> movieTitles = new ArrayList<String>();

        for(Movie movie: movies) {
            int opening = movie.getOpeningTime();
            int closing = movie.getClosingTime();
            String title = movie.getName() + " (" +
movie.getLocation().getName() + ", " + Utils.formatTime(opening) +
"- " + Utils.formatTime(closing) + ")";
            movieTitles.add(title);
        }

        ArrayAdapter<String> adapter = new
ArrayAdapter<String>(this,
android.R.layout.simple_expandable_list_item_1, movieTitles);
//        MovieAdapter movieAdapter = new MovieAdapter(this);
        movieListView.setAdapter(adapter);
//        movieListView.setOnItemClickListener(movieAdapter);

        movieListView.setOnItemClickListener(new
AdapterView.OnItemClickListener() {

            @Override
            public void onItemClick(AdapterView<?> parent,
View view, int position, long id) {
                Intent resultIntent = new Intent();
                resultIntent.putExtra("movieId",
movies.get(position).getId().longValue());
                setResult(RESULT_OK, resultIntent);

                finish();
            }
        });
    }
}
```

```
        });
    }

    setTitle("Movies");
}

}

class MovieAdapter extends ArrayAdapter<Movie> implements
AdapterView.OnItemClickListener {
    protected List<Movie> movies;
    protected Context context;

    public MovieAdapter(Context context) {
        super(context, R.layout.adapter_row_movie);

        this.context = context;
        movies = ContentManager.instance().findMovies();

        Log.i("ROUT-Movies", "Start");
        for(Movie movie: movies) {
            Log.i("ROUT", movie.getName());
        }
        Log.i("ROUT-Movies", "End");
    }

    @Override
    public View getView(int position, View rowView, ViewGroup
parent) {
        LayoutInflater inflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);

        if (rowView == null) {
            rowView =
inflater.inflate(R.layout.adapter_row_movie, parent, false);
        }
        TextView titleView = (TextView)
rowView.findViewById(R.id.movieTitleText);
        TextView timeView = (TextView)
rowView.findViewById(R.id.movieTimeText);
        TextView cinemaView = (TextView)
rowView.findViewById(R.id.movieCinemaText);

        titleView.setText(movies.get(position).getName());

        cinemaView.setText(movies.get(position).getLocation().getName(
));
        int opening = movies.get(position).getOpeningTime();
        int closing = movies.get(position).getClosingTime();
        timeView.setText(Utils.formatTime(opening) + "-" +
Utils.formatTime(closing));
    }
}
```

```
        return rowView;
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        Log.i("ROUT", movies.get(position).getName());
    }
}
```

## 8.9 LType.java

```
package gr.epp.mapnav.data;

public class LType {
    public final static int USER = 1;
    public final static int SUPER_MARKET = 2;
    public final static int GAS_STATION = 3;
    public final static int CINEMA = 4;
    public final static int MOVIE = 4;
}
```

## 8.10 GlobalData.java

```
package gr.epp.mapnav.data;

import gr.epp.mapnav.dao.DaoMaster;
import gr.epp.mapnav.dao.DaoMaster.DevOpenHelper;
import gr.epp.mapnav.dao.DaoSession;
import gr.epp.mapnav.dao.Location;
import gr.epp.mapnav.dao.LocationDao;
import gr.epp.mapnav.dao.Movie;
import gr.epp.mapnav.dao.MovieDao;
import gr.epp.mapnav.dao.User;
import gr.epp.mapnav.dao.UserDao;

import java.util.ArrayList;
import java.util.List;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;

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

public class ContentManager {
    private static ContentManager instance;
```



```
private SQLiteDatabase db;
private DaoMaster daoMaster;
private DaoSession daoSession;
private UserDao userDao;
private LocationDao locationDao;
private MovieDao movieDao;

private final Location LOCATIONS[] = new Location[] {
    new Location(0L, "ΕΚΟ - Αφοί Σαμολή", "Μουστακλή
Σπύρου 26, Ηράκλειο\n+302810111111",
                LType.GAS_STATION, 9*3600, 18*3600,
35.33458, 25.10358),
    new Location(0L, "Shell - Μυρτέα Α.Ε.", "Λεωφόρος
Ικάρου 42, Ηράκλειο\n+302810222222",
                LType.GAS_STATION, 12*3600, 21*3600,
35.33888, 25.14084),
    new Location(0L, "Χαλκιάδακης", "Κορωναίου 7,
Ηράκλειο\n+302810333333",
                LType.SUPER_MARKET, 9*3600, 18*3600,
35.34066, 25.13330),
    new Location(0L, "ΕΛΙΤ", "Λεωφόρος Κνωσού 250,
Ηράκλειο\n+302810444444",
                LType.SUPER_MARKET, 12*3600, 21*3600,
35.31197, 25.14933),
    new Location(0L, "Τεχνόπολις", "Λεωφόρος Ανδρέα
Παπανδρέου 52, Ηράκλειο\n+302810555555",
                LType.CINEMA, 0, 0, 35.33592,
25.07605),
    new Location(0L, "Βιτσέντζος Κορνάρος", "Μαλικούτη
18, Ηράκλειο\n+302810666666",
                LType.CINEMA, 0, 0, 35.34146,
25.13539),
};

public ContentManager(Context applicationContext) {
    DevOpenHelper helper = new
DaoMaster.DevOpenHelper(applicationContext, "mapnav_v1", null);
    db = helper.getWritableDatabase();
    daoMaster = new DaoMaster(db);
    daoSession = daoMaster.newSession();

    userDao = daoSession.getUserDao();
    locationDao = daoSession.getLocationDao();
    movieDao = daoSession.getMovieDao();

    initData();
}

public static void init(Context applicationContext) {
    if (instance == null) {
        instance = new ContentManager(applicationContext);
    }
}
```

```
        public static ContentManager instance() {
            if (instance == null) {
                throw new IllegalStateException("Content manager
not initialized!");
            }

            return instance;
        }

        public User findUser(String username, String password) {
            User user = userDao.queryBuilder()

                .where(UserDao.Properties.Username.eq(username),
UserDao.Properties.Password.eq(password))
                    .limit(1)
                    .unique();

            if (user != null) {
                return user;
            } else {
                return null;
            }
        }

        public User findUser(String username) {
            User user = userDao.queryBuilder()

                .where(UserDao.Properties.Username.eq(username))
                    .limit(1)
                    .unique();

            return user;
        }

        public User createUser(String username, String password,
String name) {

            User user = userDao.queryBuilder()

                .where(UserDao.Properties.Username.eq(username))
                    .limit(1)
                    .unique();

            if(user != null) {
                return null;
            }

            user = new User();

            user.setUsername(username);
            user.setPassword(password);
            user.setName(name);

            Location location = GlobalData.defaultLocation(user);
```

```
        locationDao.insert(location);

        user.setLocation(location);

        userDao.insert(user);

        return user;
    }

    public void updateLocation(Location location, LatLng latLng) {
        location.setLat(latLng.latitude);
        location.setLng(latLng.longitude);

        locationDao.update(location);
    }

    public Location findLocation(String name, int type) {
        Location location = locationDao.queryBuilder()
            .where(LocationDao.Properties.Name.eq(name),
LocationDao.Properties.Type.eq(type))
            .limit(1)
            .unique();

        return location;
    }

    public List<Location> findLocations(int type) {
        return locationDao.queryBuilder()
            .where(LocationDao.Properties.Type.eq(type))
            .list();
    }

    public Location createLocation(
        String name,
        String desc,
        Integer type,
        Integer openingTime,
        Integer closingTime,
        Double lat,
        Double lng) {
        Location location = new Location();
        location.setName(name);
        location.setDesc(desc);
        location.setType(type);
        location.setOpeningTime(openingTime);
        location.setClosingTime(closingTime);
        location.setLat(lat);
        location.setLng(lng);

        locationDao.insert(location);

        return location;
    }
}
```

```
    public Movie findMovie(String name, Integer openingTime,
Integer closingTime, Long locationId) {
        Movie movie = movieDao.queryBuilder()
            .where(

MovieDao.Properties.Name.eq(name),

MovieDao.Properties.OpeningTime.eq(openingTime),

MovieDao.Properties.ClosingTime.eq(closingTime),

MovieDao.Properties.LocationId.eq(locationId)
            )
            .limit(1)
            .unique();

        return movie;
    }

    public List<Movie> findMovies() {
        return movieDao.queryBuilder().list();
    }

    public List<Movie> findMovies(String name) {
        return movieDao.queryBuilder()
            .where(MovieDao.Properties.Name.eq(name))
            .list();
    }

    public Movie findMovie(Long id) {
        Movie movie = movieDao.load(id);

        return movie;
    }

    public List<Location> findCinemas(String movieName) {
        List<Movie> movies = findMovies(movieName);

        List<Location> cinemas = new ArrayList<Location>();

        for(Movie movie: movies) {
            cinemas.add(movie.getLocation());
        }

        return cinemas;
    }

    public Movie createMovie(String name, Integer openingTime,
Integer closingTime, Long locationId) {
        Movie movie = new Movie();

        movie.setName(name);
        movie.setOpeningTime(openingTime);
        movie.setClosingTime(closingTime);
    }
}
```

```
        movie.setLocationId(locationId);

        movieDao.insert(movie);

        return movie;
    }

    protected void initData() {
        createUser("a", "a", "Κώστας");

        for(Location loc: LOCATIONS) {
            Location temp = findLocation(loc.getName(),
loc.getType());
            if(temp == null) {
                createLocation(loc.getName(), loc.getDesc(),
loc.getType(), loc.getOpeningTime(), loc.getClosingTime(),
loc.getLat(), loc.getLng());
            }
        }

        List<Location> cinemaList = locationDao.queryBuilder()

        .where(LocationDao.Properties.Type.eq(LType.CINEMA))
            .list();

        if (findMovie("Lord of the Rings", 21*3600, 24*3600,
cinemaList.get(0).getId()) == null) {
            createMovie("Lord of the Rings", 21*3600, 24*3600,
cinemaList.get(0).getId());
        }

        if (findMovie("Avengers", 20*3600, 22*3600,
cinemaList.get(0).getId()) == null) {
            createMovie("Avengers", 20*3600, 22*3600,
cinemaList.get(0).getId());
        }

        if (findMovie("Avengers", 20*3600, 22*3600,
cinemaList.get(1).getId()) == null) {
            createMovie("Avengers", 20*3600, 22*3600,
cinemaList.get(1).getId());
        }

        if (findMovie("Avengers", 22*3600, 24*3600,
cinemaList.get(1).getId()) == null) {
            createMovie("Avengers", 22*3600, 24*3600,
cinemaList.get(1).getId());
        }

    }
}
```

## 8.11 activity\_login.xml

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:paddingBottom="@dimen/activity_vertical_margin"
android:paddingLeft="@dimen/activity_horizontal_margin"
android:paddingRight="@dimen/activity_horizontal_margin"
android:paddingTop="@dimen/activity_vertical_margin"
tools:context="gr.epp.mapnav.LoginActivity" >

<LinearLayout
    android:id="@+id/loginUsernameLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight=".4"
        android:text="Name:"

        android:textAppearance="?android:attr/textAppearanceMedium" />

    <EditText
        android:id="@+id/loginUsernameField"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight=".6"
        android:ems="10" >

        <requestFocus />
    </EditText>
</LinearLayout>

<LinearLayout
    android:id="@+id/loginPasswordLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/loginUsernameLayout"
    android:layout_centerHorizontal="true"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight=".4"
```

```
        android:text="Password:"

android:textAppearance="?android:attr/textAppearanceMedium" />

    <EditText
        android:id="@+id/loginPasswordField"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight=".6"
        android:ems="10"
        android:inputType="textPassword" />
</LinearLayout>

<Button
    android:id="@+id/loginSubmitButton"
    android:layout_below="@id/loginPasswordLayout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="16dp"
    android:text="Login"
    android:onClick="loginClick" />

<Button
    android:id="@+id/loginRegisterButton"
    android:layout_below="@id/loginSubmitButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="16dp"
    android:text="Register"
    android:onClick="registerClick" />

</RelativeLayout>
```

## 8.12 activity\_register.xml

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="gr.epp.mapnav.RegisterActivity" >

    <LinearLayout
        android:id="@+id/registerUsernameLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".4"
            android:text="Username:"

android:textAppearance="?android:attr/textAppearanceMedium" />

        <EditText
            android:id="@+id/registerUsernameField"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".6"
            android:ems="10" >

            <requestFocus />
        </EditText>
    </LinearLayout>

    <LinearLayout
        android:id="@+id/registerPasswordLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/registerUsernameLayout"
        android:layout_centerHorizontal="true"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".4"
            android:text="Password:"

android:textAppearance="?android:attr/textAppearanceMedium" />

        <EditText
            android:id="@+id/registerPasswordField"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".6"
            android:ems="10"
            android:inputType="textPassword" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/registerPasswordConfirmLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/registerPasswordLayout"
        android:layout_centerHorizontal="true"
```



```
        android:orientation="horizontal" >

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".4"
            android:text="Password (again):"

android:textAppearance="?android:attr/textAppearanceMedium" />

        <EditText
            android:id="@+id/registerPasswordConfirmField"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".6"
            android:ems="10"
            android:inputType="textPassword" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/registerNameLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/registerPasswordConfirmLayout"
        android:layout_centerHorizontal="true"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".4"
            android:text="Name:"

android:textAppearance="?android:attr/textAppearanceMedium" />

        <EditText
            android:id="@+id/registerNameField"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight=".6"
            android:ems="10" />
    </LinearLayout>

    <Button
        android:id="@+id/registerSubmitButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/registerNameLayout"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:text="Register"
        android:onClick="registerClick" />

</RelativeLayout>
```

## 8.13 default\_map.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:tools="http://schemas.android.com/tools"
      tools:context="gr.epp.mapnav.DefaultMapActivity" >

    <item
        android:id="@+id/action_list_gas_stations"
        android:orderInCategory="100"
        android:showAsAction="never|withText"
        android:title="Show gas stations"/>

    <item
        android:id="@+id/action_nearest_gas_station"
        android:orderInCategory="110"
        android:showAsAction="never|withText"
        android:title="Nearest gas station"/>

    <item
        android:id="@+id/action_list_super_markets"
        android:orderInCategory="200"
        android:showAsAction="never|withText"
        android:title="Show super markets"/>

    <item
        android:id="@+id/action_nearest_super_market"
        android:orderInCategory="210"
        android:showAsAction="never|withText"
        android:title="Nearest super market"/>

    <item
        android:id="@+id/action_list_cinemas"
        android:orderInCategory="300"
        android:showAsAction="never|withText"
        android:title="Show cinemas"/>

    <item
        android:id="@+id/action_nearest_cinema"
        android:orderInCategory="310"
        android:showAsAction="never|withText"
        android:title="Nearest cinema"/>

    <item
        android:id="@+id/action_show_movies"
        android:orderInCategory="400"
        android:showAsAction="never|withText"
        android:title="Show Movies"/>

</menu>
```

## 8.14 activity\_default\_map.xml

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:paddingBottom="@dimen/activity_vertical_margin"
  android:paddingLeft="@dimen/activity_horizontal_margin"
  android:paddingRight="@dimen/activity_horizontal_margin"
  android:paddingTop="@dimen/activity_vertical_margin"
  tools:context="gr.epp.mapnav.DefaultMapActivity" >

  <fragment
    android:id="@+id/map"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:name="com.google.android.gms.maps.MapFragment"/>

</RelativeLayout>
```

## 8.15 activity\_movie.xml

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:paddingBottom="@dimen/activity_vertical_margin"
  android:paddingLeft="@dimen/activity_horizontal_margin"
  android:paddingRight="@dimen/activity_horizontal_margin"
  android:paddingTop="@dimen/activity_vertical_margin"
  tools:context="gr.epp.mapnav.MovieActivity" >

  <ListView
    android:id="@+id/movieListView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true" >
  </ListView>

</RelativeLayout>
```