

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
Τ.Ε.Ι. ΗΠΕΙΡΟΥ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη εφαρμογής Android για την παροχή
υπηρεσιών καθοδήγησης στους χρήστες.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ

ΓΚΟΥΡΝΕΛΟΣ ΛΑΜΠΡΟΣ ΑΜ. 9147

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΤΣΟΥΛΟΣ ΙΩΑΝΝΗΣ

Άρτα 2017

Περίληψη

Στην παρούσα πτυχιακή εργασία περιγράφονται τεχνικές ανάπτυξης εφαρμογών για smartphones και αναπτύχθηκε εφαρμογή με χρήση του Android Sdk.

Στην εισαγωγή αναλύεται και παρουσιάζεται η ραγδαία ανάπτυξη του λειτουργικού συστήματος Android και τις προοπτικές που ανοίγονται για τους νέους προγραμματιστές την σύγχρονη εποχή. Γίνεται αναφορά πάνω στις πλέον αναπτυσσόμενες πλατφόρμες και σύγκριση αυτών καθώς και των πλεονεκτημάτων και μειονεκτημάτων που παρουσιάζουν.

Στην συνέχεια περιγράφεται η χρήση και ενσωμάτωση των εργαλείων στο σύστημα ανάπτυξης εφαρμογών του Android, και εμβαθύνουμε πάνω στις πιο χρήσιμες παραμέτρους τους χρησιμοποιώντας την αντικειμενοστραφής γλώσσα προγραμματισμού JAVA. Τέλος, χρησιμοποιώντας τα παραπάνω εργαλεία και τις εμπειρίες που αποκτήθηκαν από την μελέτη αυτών, αναπτύχθηκε μια εφαρμογή διεπαφής χρήστη πάνω στην πλατφόρμα της Google, Android SDK, και περιγράφεται η διαδικασία για την ανάπτυξη αυτής.

Στόχος της εφαρμογής είναι να δίνει στον χρήστη τη δυνατότητα να οργανώσει καλύτερα τον χρόνο του βρίσκοντας τις κοντινότερες παραλίες στη δυτική Ελλάδα, παρέχοντάς του την κοντινότερη διαδρομή σύμφωνα με τους χάρτες της Google. Η εφαρμογή τρέχει σε κινητά με λειτουργικό σύστημα Android και χρησιμοποιεί τους χάρτες που προσφέρονται από την εταιρεία Google.

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή στο Λειτουργικό Σύστημα Android

- 1.1** Τι είναι το Android
- 1.2** Εφαρμογές Android
- 1.3** Εξέλιξη του Android
 - 1.3.1** Android 1.5 Cupcake
 - 1.3.2** Android 1.6 Donut
 - 1.3.3** Android 2.0/2.1 Eclair
 - 1.3.4** Android 2.2 Froyo
 - 1.3.5** Android 2.3 Gingerbread
 - 1.3.6** Android 3.0 Honeycomb
 - 1.3.7** Android 4.0 Ice Cream Sandwich
 - 1.3.8** Android 5.0 "Lollipop"
- 1.4** Αρχιτεκτονική του Android
 - 1.4.1** Πυρήνας (Linux kernel)
 - 1.4.2** Βιβλιοθήκες
 - 1.4.3** Η εικονική μηχανή Dalvik
 - 1.4.4** Χρόνος Εκτέλεσης Εφαρμογής (Android Runtime)
 - 1.4.5** Πλαίσιο Εφαρμογής (Application Framework)
- 1.5** Στο εσωτερικό μιας εφαρμογής του Android
 - 1.5.1** Το αρχείο AndroidManifest.xml
 - 1.5.2** Οι φάκελοι src & res
 - 1.5.3** Οι υπόλοιποι φάκελοι του project
 - 1.5.4** Δομικά Μέρη μιας Εφαρμογής
- 1.6** Ασφάλεια στο Android

ΚΕΦΑΛΑΙΟ 2: Περιβάλλον Ανάπτυξης

- 2.1** Android Studio
- 2.2** Εγκατάσταση
- 2.3** Δημιουργία νέου project
- 2.4** Πλοήγηση στο Project
 - 2.4.1** Το project panel
 - 2.4.2** Δομή του project
- 2.5** Δημιουργία User Interface
 - 2.5.1** Graphical Editor
 - 2.5.2** Text-based editor
 - 2.5.3** Πολλαπλές Οθόνες
 - 2.5.4** Διαχείριση Συμβάντων
- 2.6** Android Virtual Device (AVD) Manager
- 2.7** Εκτέλεση και Debugging
- 2.8** Προετοιμασία της εφαρμογής για κυκλοφορία

ΚΕΦΑΛΑΙΟ 3: Εφαρμογή Ionion

3.1 Αρχείο AndroidManifest

3.2 Περιγραφή της Εφαρμογής

ΚΕΦΑΛΑΙΟ 4: Συμπεράσματα και μελλοντικές επεκτάσεις

4.1 Συμπεράσματα

4.2 Δυνατότητες επέκτασης

Βιβλιογραφία

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή στο Λειτουργικό Σύστημα Android

1.1 Τι είναι το Android;

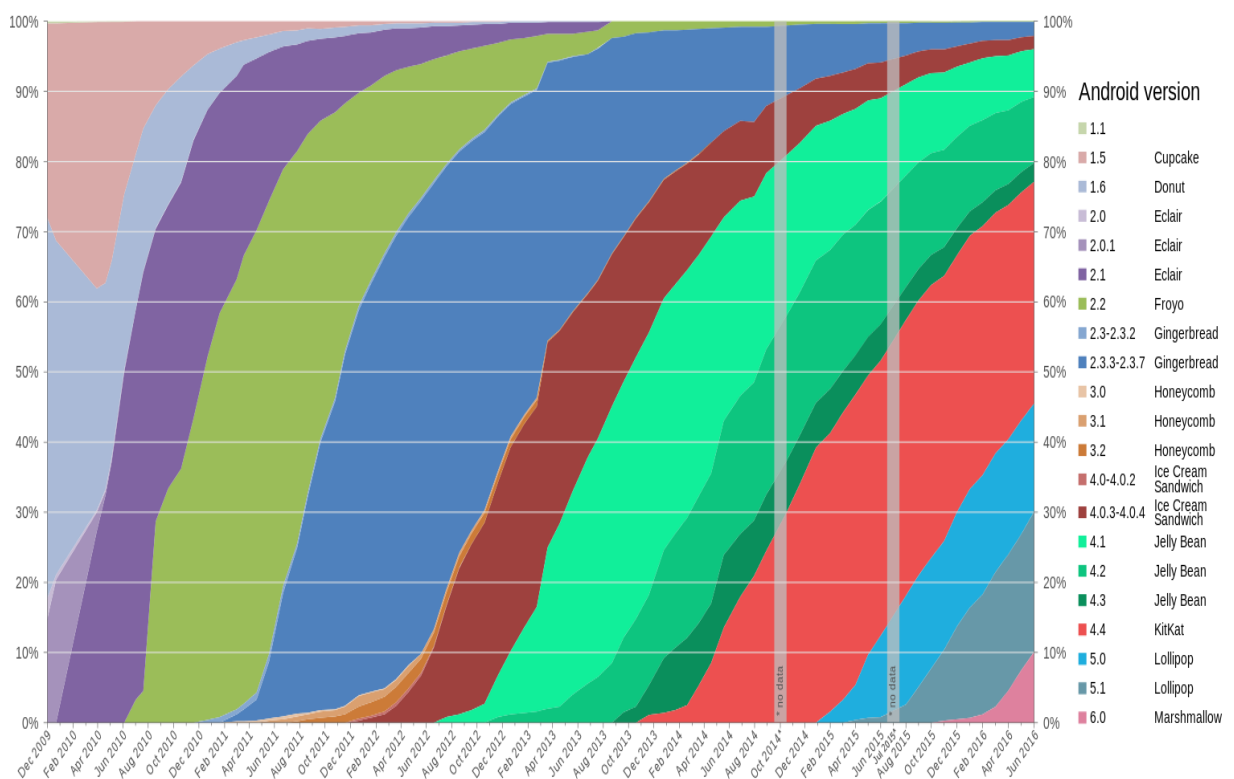
Το Android είναι ένα λειτουργικό σύστημα ανοιχτού κώδικα, βασισμένο στο Linux, για φορητές συσκευές όπως smartphones και tablets. Αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance η οποία είναι μια κοινοπραξία εταιριών λογισμικού, κατασκευής hardware και τηλεπικοινωνιών, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων στις φορητές συσκευές. Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού.

Τον Ιούλιο του 2005, η Google εξαγόρασε την Android Inc, μια μικρή εταιρεία με έδρα το Palo Alto στην California των ΗΠΑ. Εκείνη την εποχή ελάχιστα ήταν γνωστά για τις λειτουργίες της Android Inc, εκτός του ότι ανέπτυσαν λογισμικό για κινητά τηλέφωνα. Αυτή ήταν η αρχή της φημολογίας περί σχεδίων της Google για να διεισδύσει στην αγορά κινητής τηλεφωνίας.

Στην Google, η ομάδα με επικεφαλής τον Andy Rubin ανέπτυξε μια κινητή πλατφόρμα που στηρίζεται στον πυρήνα του Linux, την οποία προώθησαν με την παροχή ενός ευέλικτου, αναβαθμίσιμου συστήματος. Έχει αναφερθεί ότι η Google είχε ήδη συγκεντρώσει μια σειρά από εταίρους hardware και software και επισήμανε στους παρόχους ότι ήταν ανοικτή σε διάφορους βαθμούς συνεργασίας εκ μέρους της. Έντυπα και ηλεκτρονικά μέσα ενημέρωσης σύντομα ανέφεραν φήμες ότι η Google ανέπτυξε μια Google-branded συσκευή. Περισσότερες φήμες ακολούθησαν, αναφέροντας ότι η Google καθόριζε τις τεχνικές προδιαγραφές και έδειχνε πρωτότυπα στους κατασκευαστές κινητών τηλεφώνων και τους φορείς δικτύων. Τελικά η Google παρουσίασε το smartphone της Nexus One που χρησιμοποιεί το open source λειτουργικό σύστημα Android. Η συσκευή κατασκευάστηκε από την HTC , και έγινε διαθέσιμη στις 5 Ιανουαρίου 2010.

1.2 Εφαρμογές Android

Το Android έχει μια μεγάλη κοινότητα προγραμματιστών που γράφουν εφαρμογές, οι οποίες επεκτείνουν τη λειτουργικότητα των συσκευών. Οι εφαρμογές γράφονται σε μια προσαρμοσμένη έκδοση της JAVA και μπορεί κανείς να κατεβάσει από το online κατάστημα Google Play (πρώην Android Market) της Google όπως και από άλλα sites. Μέχρι τον Φεβρουάριο του 2012 περισσότερες από 450000 εφαρμογές ήταν διαθέσιμες για Android ενώ εκτιμάτε ότι ο αριθμός των downloads από το Android Market μέχρι το Δεκέμβριο του 2011 είχε υπερβεί τα 10 δισεκατομμύρια. Το Android είναι η πρώτη σε πωλήσεις παγκοσμίως πλατφόρμα για smartphones καθώς μέχρι το Φεβρουάριο του 2012 μετρούσε περισσότερες από 300 εκατομμύρια συσκευές σε χρήση.



1.3.1 Android 1.5 Cupcake

Η έκδοση “Cupcake”, βασισμένη στο Linux Kernel 2.6.27, παρουσιάστηκε στις 30 Απριλίου του 2009. Υποστηρίζει νέες λειτουργίες για την κάμερα τις συσκευής, όπως η καταγραφή και παρακολούθηση βίντεο από την λειτουργία της κάμερας και η άμεση μεταφόρτωση του βίντεο αλλά και των φωτογραφιών στο Youtube και το Picasa αντίστοιχα απευθείας από το τηλέφωνο. Έχει νέο έξυπνο πληκτρολόγιο με πρόβλεψη κειμένου. Υποστηρίζει πρότυπο Bluetooth A2DP και AVRCP ενώ έχει και την ικανότητα να συνδέεται αυτόματα σε μικροσυσκευές Bluetooth από μια συγκεκριμένη απόσταση. Ακόμα στην έκδοση αυτή έχει νέο γραφικό περιβάλλον με κινούμενες μεταβάσεις οθόνης.

1.3.2 Android 1.6 Donut

Η έκδοση “Donut”, βασισμένη στο Linux Kernel 2.6.29, παρουσιάστηκε στις 15 Σεπτεμβρίου του 2009. Έχει ταχύτερη απόκριση σε σχέση με την προηγούμενη έκδοση. Υποστηρίζεται πλέον η επιλογή πολλαπλών αρχείων ταυτόχρονα, έχει ανανεωμένο γκάλερι και φωτογραφική μηχανή, καθώς και βελτιωμένο Android Market. Έχει ανανεωμένη φωνητική αναζήτηση, με ταχύτερη απόκριση και βαθύτερη ολοκλήρωση με εγγενείς (native) εφαρμογές, συμπεριλαμβανομένης της δυνατότητας κλήσης επαφών. Δυνατότητα αναζήτησης σελιδοδεικτών, ιστορικού, επαφών αλλά και στο διαδίκτυο από την αρχική οθόνη. Υποστήριξη για ανάλυση οθονών WVGA. Ανανεωμένη υποστήριξη τεχνολογιών για CDMA/EVDO, 802.1x, VPNs και με μηχανή μετατροπής κειμένου σε ομιλία (text to speech).

1.3.3 Android 2.0/2.1 Eclair

Η έκδοση “Eclair”, βασισμένη και αυτή στον Linux Kernel 2.6.29, παρουσιάστηκε στις 26 Οκτωβρίου του 2009, ενώ τον Ιανουάριο του 2010 επανεκδόθηκε σε Android 2.1 Eclair (MR1). Σε αυτή την έκδοση υπάρχει ακόμα ταχύτερη απόκριση του υλικού σε σχέση με τις δυο προηγούμενες και πλέον υποστηρίζονται περισσότερες οθόνες και αναλύσεις. Υπάρχει νέος browser ο οποίος υποστηρίζει το πρότυπο HTML5, νέο User Interface, και βελτιωμένοι χάρτες Google (Google Maps 3.1.2). Έχει ενσωματωθεί η υποστήριξη φλας για την κάμερα η οποία έχει πλέον και ψηφιακό zoom. Επίσης έχει βελτιωθεί η κλάση MotionEvent ώστε να υπάρχει η δυνατότητα για γεγονότα πολλαπλής αφής (multitouch events). Υποστηρίζεται Bluetooth 2.1 και έχει βελτιωθεί και το πληκτρολόγιο.

1.3.4 Android 2.2 Froyo

Η έκδοση “Froyo”, βασισμένη στο Linux Kernel 2.6.32, παρουσιάστηκε στις 20 Μαΐου του 2010. Υπάρχουν βελτιστοποιήσεις στην ταχύτητα γενικά του λειτουργικού συστήματος, στην μνήμη και στην απόδοση. Έχει ενσωματωθεί ο μηχανισμός JavaScript του Chrome V8 στον browser, υπάρχει πλέον Adobe Flash 10.1, ενώ υποστηρίζεται καλύτερα πλέον το Microsoft Exchange. Έχει γίνει ανανέωση του Android Market. Ο χρήστης μπορεί πλέον να ελέγχει αν θα γίνεται ή όχι κίνηση πακέτων δεδομένων από το δίκτυο κινητής τηλεφωνίας. Υπάρχει η δυνατότητα εγκατάστασης εφαρμογών στην κάρτα μνήμης και η μεταφορά τους εκεί από τη μνήμη του τηλεφώνου. Επίσης το τηλέφωνο πλέον μπορεί να μετατραπεί σε WiFi hotspot.

1.3.5 Android 2.3 Gingerbread

Η έκδοση “Gingerbread”, βασισμένη στο Linux Kernel 2.6.35.7, παρουσιάστηκε στις 6 Δεκεμβρίου του 2010, ενώ τον Φεβρουάριο του 2011 επανεκδόθηκε σε Android 2.3.3. Στην έκδοση αυτή υπάρχουν αλλαγές στο User Interface το οποίο έχει γίνει πιο απλό και ταχύ, ενώ υποστηρίζονται πλέον οθόνες μεγάλων μεγεθών και αναλύσεων. Υπάρχει πλέον το πρωτόκολλο SIP για κλήσεις μέσω VoIP, υποστηρίζεται ο τύπος βίντεο WebM/VP8 και ο κωδικοποιητής AAC, έχει βελτιωθεί ο ήχος καθώς και οι λειτουργίες απεικόνισης για την ανάπτυξη παιχνιδιών. Υπάρχει η δυνατότητα για Copy-Paste σε όλο το σύστημα και όχι μόνο στην ίδια εφαρμογή. Υποστηρίζεται το NFC (Near Field Communication) και η ύπαρξη πολλαπλών καμερών. Επίσης, έχει βελτιωθεί η ενεργειακή υποστήριξη και έχει γίνει μετάβαση από το σύστημα αρχείων YAFFS στο ext4 στις νέες συσκευές.

1.3.6 Android 3.0 Honeycomb

Η έκδοση “Honeycomb”, βασισμένη στο Linux Kernel 2.6.36, παρουσιάστηκε στις 9 Μαΐου του 2011, με την ιδιαιτερότητα ότι προοριζόταν αποκλειστικά για tablets. Οι αλλαγές που έγιναν στην έκδοση αυτή έχουν να κάνουν κυρίως με τη βελτίωση της υποστήριξης των tablets. Υπάρχει ένα νέο, εντελώς διαφορετικό, User Interface και υποστηρίζονται διπύρηντοι και τετραπύρηντοι επεξεργαστές. Ακόμα, έχει απλοποιηθεί το multitasking έτσι ώστε ο χρήστης να μπορεί με τη χρήση ενός πλήκτρου (recent apps) να περνάει από μια εφαρμογή σε άλλη. Υπάρχει η δυνατότητα για Video Chat μέσω της εφαρμογής Google Talk καθώς η ανάγνωση βιβλίων μέσω του Google eBooks. Επιπλέον, μπορούν να κρυπτογραφηθούν όλα τα δεδομένα χρήστη.

1.3.7 Android 4.0 Ice Cream Sandwich

Η έκδοση “Ice Cream Sandwich”, βασισμένη στο Linux Kernel 3.0.1, παρουσιάστηκε στις 19 Οκτωβρίου του 2011. Για άλλη μια φορά έχει βελτιωθεί η ταχύτητα και η απόδοση του συστήματος. Πλέον στο User Interface, το οποίο είναι και πάλι διαφορετικό, υπάρχουν εικονικά πλήκτρα τα οποία παίρνουν τη θέση των φυσικών ή αφής που υπήρχαν στις συσκευές. Βελτίωση της ασφάλειας του συστήματος με την προσθήκη αναγνώρισης προσώπου για να ξεκλειδώσει η συσκευή. Ο browser μπορεί να ανοίξει ταυτόχρονα μέχρι και 16 καρτέλες. Υπάρχει η δυνατότητα ο χρήστης να τερματίσει εφαρμογές οι οποίες τρέχουν στο background, ενώ μπορεί να θέσει και όρια στην κίνηση πακέτων δεδομένων. Η εφαρμογή Android Beam αξιοποιεί πλέον το NFC αφού επιτρέπει την αποστολή δεδομένων από τη συσκευή σε όσες βρίσκονται εντός μιας μικρής ακτίνας εμβέλειας. Ακόμα με την ύπαρξη του Wi-Fi Direct συσκευές μπορούν να συνδεθούν μεταξύ τους ασύρματα χωρίς την μεσολάβηση κάποιου access point. Τέλος, υποστηρίζεται η εγγραφή βίντεο σε 1080p.



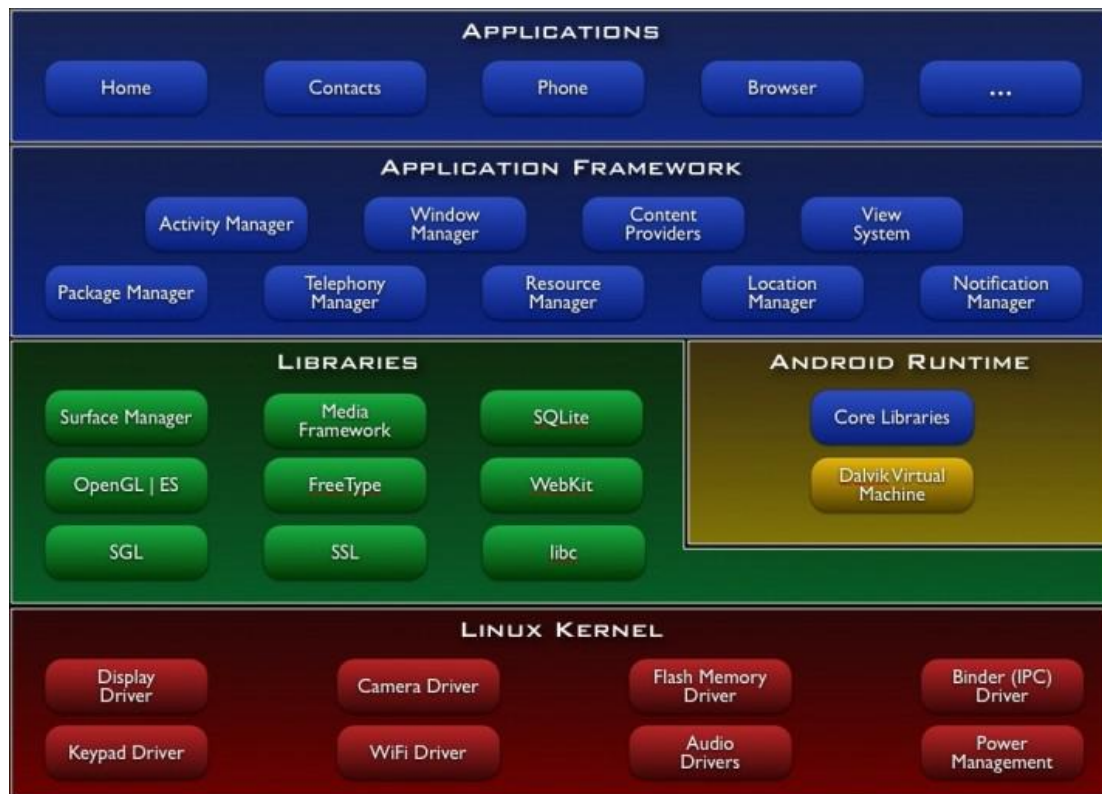
Android 4.0 (Ice cream sandwich)

1.2.8 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.

1.4 Αρχιτεκτονική του Android

Το Android δεν είναι μόνο ένα λειτουργικό σύστημα. Είναι μια στοίβα λογισμικού η οποία αποτελείται από το λειτουργικό σύστημα, τις υπηρεσίες διασύνδεσης με τις εφαρμογές (middleware) και τέλος από τις κύριες (core) εφαρμογές, μεταξύ αυτών, ενός email client, μιας εφαρμογής διαχείρισης SMS, ενός ημερολογίου, ενός browser, εφαρμογή διαχείρισης επαφών, και άλλες οι οποίες έρχονται δεμένες με την υπόλοιπη στοιβάδα λογισμικού του Android. Στο επίσημο σχεδιάγραμμα που ακολουθεί θα δούμε οπτικά την αρχιτεκτονική αυτή.



Η αρχιτεκτονική του λειτουργικού συστήματος αποτελείται από 5 βασικά επίπεδα.

- Τον πυρήνα Linux (Linux Kernel)
- Τις εγγενείς και τις προηγμένες βιβλιοθήκες (Libraries)
- Την εικονική μηχανή Dalvik (Dalvik VM)
- Τον χρόνο εκτέλεσης (Android Runtime)
- Το πλαίσιο εφαρμογής (Application Framework)

1.4.1 Πυρήνας Linux (Linux Kernel)

Η βάση της στοίβας λογισμικού του Android είναι ο πυρήνας Linux. Ο τροποποιημένος πυρήνας του συστήματος βασίζεται στην έκδοση 2.6 (και στην έκδοση 3.0.1 για το Android 4.0) του Linux Kernel, η οποία υποστηρίζει όλες τις κύριες λειτουργίες του λειτουργικού συστήματος. Οι λειτουργίες αυτές αφορούν διαχείριση μνήμης, διαχείριση διεργασιών, λειτουργίες δικτύου, ασφάλεια του λειτουργικού, και ένα σύνολο οδηγών υλικού (hardware drivers). Οι οδηγοί αυτοί είναι υπεύθυνοι για την επικοινωνία του software με το hardware της συσκευής. Ενδεικτικά ο πυρήνας του Android περιέχει:

- Οδηγό προβολής οθόνης
- Οδηγό Wifi και Bluetooth
- Οδηγό κάμερας
- κλπ

Ο πυρήνας του Android μπορεί να βασίζεται στον πυρήνα του Linux, αλλά διαφέρει αρκετά από αυτόν. Ο λόγος είναι οι αλλαγές στην αρχιτεκτονική που έχει κάνει η Google για να είναι ελαφρύτερος και βελτιστοποιημένος για χρήση σε κινητές συσκευές. Αυτό σημαίνει ότι παρότι το Android είναι κατά βάση Linux, επί της ουσίας είναι αρκετά δύσκολο να τρέξουν εφαρμογές ή να χρησιμοποιηθούν βιβλιοθήκες από τη μία πλατφόρμα στην άλλη. Ο Linus Torvalds έχει αναφέρει ότι τελικά στο μέλλον το Android και το Linux θα μοιράζονται έναν κοινό πυρήνα, αλλά αυτό θα αργήσει 4-5 χρόνια ακόμα.

1.4.2 Βιβλιοθήκες

Στο δεύτερο επίπεδο της στοίβας έχουμε τις βιβλιοθήκες του Android. Αυτές ουσιαστικά αποτελούν τα APIs που είναι διαθέσιμα στους προγραμματιστές για την ανάπτυξη των εφαρμογών. Οι βιβλιοθήκες από μόνες τους δεν αποτελούν εφαρμογές αλλά ενσωματώνονται και χρησιμοποιούνται από τις εφαρμογές για τις διάφορες λειτουργίες που παρέχει η καθεμία από αυτές. Ουσιαστικά αποτελούν ένα από τα δομικά υλικά των εφαρμογών, και άρα είναι αναπόσπαστο κομμάτι τους. Οι δυνατότητες των βιβλιοθηκών του Android γίνονται εμφανείς στους προγραμματιστές στην στοίβα του πλαισίου εφαρμογής.

Το σύνολο σχεδόν των βιβλιοθηκών είναι γραμμένο σε C και C++, οι οποίες έχουν μεταγλωττιστεί για τη χρήση τους από το λειτουργικό.

Μερικές από τις κύριες βιβλιοθήκες του Android είναι:

- **System C library** – μια ενσωμάτωση της standard βιβλιοθήκης συστήματος της C (libc) τροποποιημένη για κινητές συσκευές βασισμένες στο Linux.
- **Βιβλιοθήκες Πολυμέσων** – Υποστηρίζει αναπαραγωγή και εγγραφή πολλών δημοφιλών μέσων ήχου και εικόνας, όπως: MPEG4, H.264, MP3, AAC, AMR, JPG, και PNG
- **Surface Manager** – διαχειρίζεται την πρόσβαση στο υποσύστημα προβολής, και συνθέτει απρόσκοπτα δισδιάστατα και τρισδιάστατα επίπεδα γραφικών τα οποία προέρχονται από πολλαπλές εφαρμογές.
- **LibWebCore** – μια μοντέρνα μηχανή υποστήριξης πλοήγησης στο διαδίκτυο (browser engine) η οποία χρησιμοποιείτε και από τον ενσωματωμένο browser του Android αλλά και από τις WebViews που ενσωματώνονται στις εφαρμογές.
- **SGL** – η γνωστή μηχανή δισδιάστατων γραφικών
- **Βιβλιοθήκες 3D** – μια υλοποίηση βασισμένη στα APIs του OpenGL ES 1. Οι βιβλιοθήκες χρησιμοποιούν είτε τρισδιάστατη επιτάχυνση υλικού, όπου αυτή είναι διαθέσιμη, είτε μια υψηλά βελτιωμένη τρισδιάστατη επιτάχυνση λογισμικού σε περίπτωση που η πρώτη δεν είναι διαθέσιμη.
- **SQLite** – μια πανίσχυρη και συνάμα πολύ ελαφριά σχεσιακή βάση δεδομένων

1.4.3 Η εικονική μηχανή Dalvik

Σχεδόν το σύνολο των APIs του Android βασίζονται στη γλώσσα προγραμματισμού Java. Στην Java ως γνωστόν υπάρχει η λεγόμενη Java Virtual Machine στην οποία εκτελείτε ο κώδικας bytecode των εφαρμογών. Στο Android υπάρχει κάτι παρόμοιο και δεν είναι άλλο από την εικονική μηχανή Dalvik.

Η Dalvik λοιπόν είναι η εικονική μηχανή μέσω της οποίας τρέχουν οι εφαρμογές του Android. Η κάθε εφαρμογή τρέχει μέσω τις δικής της εικονικής μηχανής στη δικιά της διεργασία και για αυτό το λόγο καμία εφαρμογή δεν έχει επαφή με την άλλη, ενώ εκτελούνται ταυτόχρονα. Η Dalvik δεν υποστηρίζει τον κώδικα bytecode, αντί αυτού οι κλάσεις της Java γίνονται compile σε αρχεία .dex ώστε να τρέξουν στην VM. Τα αρχεία dex ουσιαστικά αποτελούν συμπιεσμένα δεδομένα για εξοικονόμηση χώρου κατά την εκτέλεση.

Το Android είναι από τη φύση του multitasking λειτουργικό σύστημα και για αυτό επιτρέπει στις εφαρμογές του να τρέχουν σε πολλά νήματα ταυτόχρονα και να απασχολούν πολλές διαδικασίες εάν αυτό είναι αναγκαίο. Για να γίνει αυτό εφικτό η μηχανή Dalvik είναι σχεδιασμένη για να έχει ελάχιστο αντίκτυπο στη χρήση της μνήμης. Χάρη στον λιτό της σχεδιασμό, το σύστημα είναι σε θέση να τρέχει πολλές εικονικές μηχανές ταυτόχρονα.

1.4.4 Χρόνος Εκτέλεσης Εφαρμογής (Android Runtime)

Ο χρόνος εκτέλεσης των εφαρμογών του Android, βρίσκεται στο ίδιο επίπεδο με τις κύριες βιβλιοθήκες και την μηχανή Dalvik. Εδώ βρίσκουμε το κοινό σημείο επαφής μεταξύ των δυνατοτήτων που παρέχουν οι βιβλιοθήκες και του χρόνου εκτέλεσης της εικονικής μηχανής Dalvik τις λειτουργίες τις οποίας, περιγράψαμε παραπάνω.

1.4.5 Πλαίσιο Εφαρμογής (Application Framework)

Το Android παρέχει στους developers μια ανοιχτού κώδικα πλατφόρμα ανάπτυξης και τη δυνατότητα να αναπτύξουν με αυτή ιδιαίτερα καινοτόμες και πλούσιες σε υλικό, εφαρμογές. Οι developers έχουν στην διάθεση τους τη δυνατότητα ελέγχου του υλικού της συσκευής και μέσω αυτής μπορούν να αποκτήσουν πρόσβαση σε υπηρεσίες εντοπισμού, εκτέλεση διεργασιών παρασκηνίου, και πάρα πολλές ακόμη δυνατότητες οι οποίες βασίζονται στα APIs που είναι διαθέσιμα.

Στο επόμενο επίπεδο της αρχιτεκτονικής του Android λοιπόν, συναντάμε το πλαίσιο των εφαρμογών. Οι developers έχουν πρόσβαση σε όλα τα

APIs μεταξύ αυτών και στα κύρια APIs που χρησιμοποιούν οι ενσωματωμένες εφαρμογές. Η δομή των εφαρμογών είναι τέτοια που ευνοείται η επαναχρησιμοποίηση δομικών συστατικών, και επίσης επιτρέπεται η χρήση των δυνατοτήτων τις μίας εφαρμογής από άλλες εφαρμογές, βέβαια κάτω από τις προδιαγραφές ασφάλειας του Android.

Τα σημαντικότερα δομικά στοιχεία του πλαισίου εφαρμογών είναι:

- **Σύστημα προβολών (View System)** – αποτελεί ένα εκτενές σύνολο από αντικείμενα GUI τα οποία μπορούν να χρησιμοποιηθούν κατά το σχεδιασμό μιας εφαρμογής.

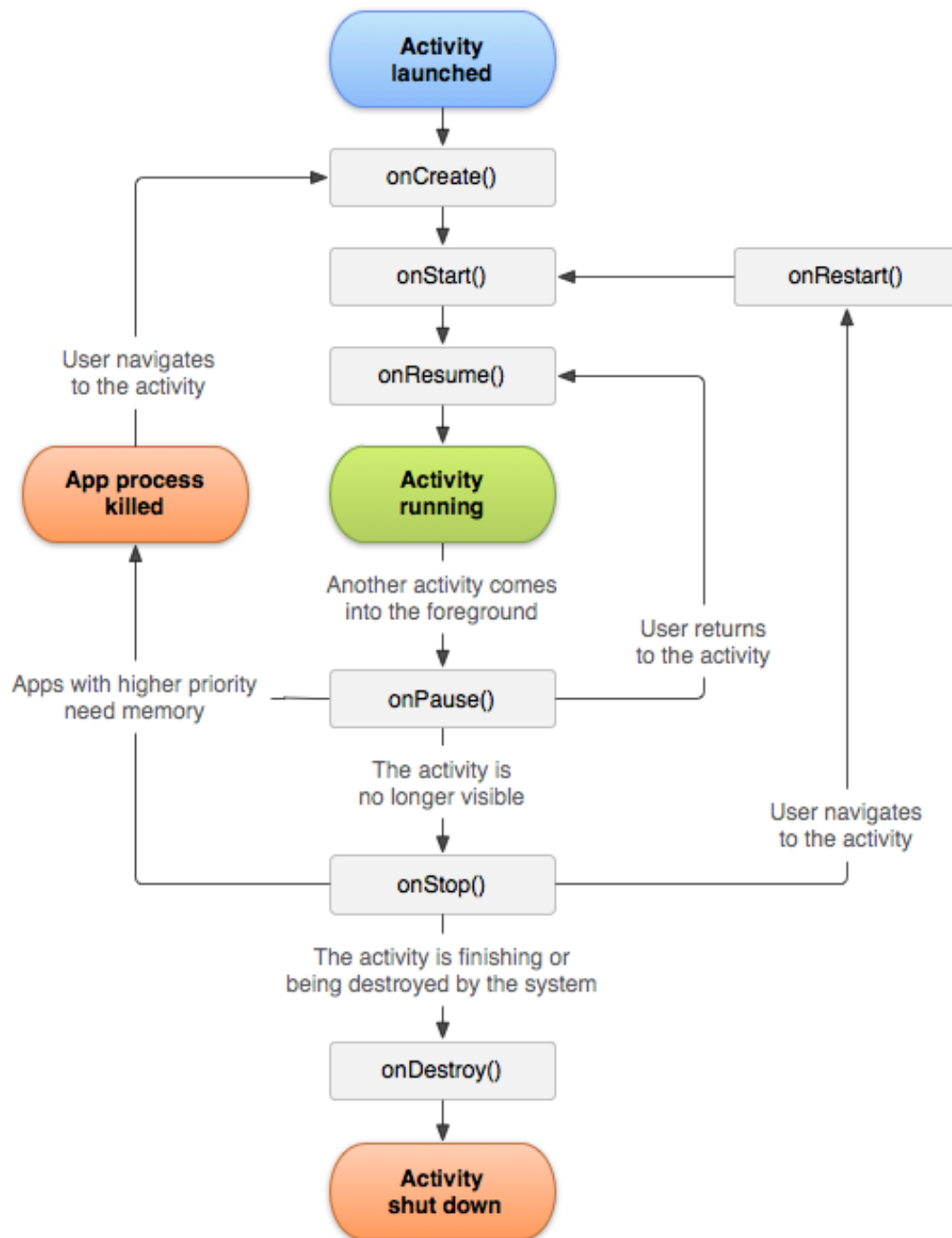
Παραδείγματα προβολών είναι οι λίστες (listView), το πλέγμα (GridView), πεδία εισαγωγής κειμένου, κουμπιά, κλπ

- **Πάροχος Περιεχομένου (Content Provider)** – δίνει τη δυνατότητα στις εφαρμογές να μοιράζονται ή να ανταλλάσσουν δεδομένα μιας συγκεκριμένης μορφής η οποία ορίζεται από τον πάροχο. Παραδείγματα δεδομένων, είναι οι επαφές χρήστη και οι βάσεις δεδομένων των εφαρμογών.

- **Διαχειριστής Πόρων (Resource Manager)** – παρέχει πρόσβαση σε υλικό το οποίο δεν είναι σε μορφή κώδικα όπως πχ, εικόνες, αρχεία xml, πίνακες χαρακτήρων, κλπ

- **Διαχειριστής Ειδοποιήσεων (Notification Manager)** – δίνει στις εφαρμογές πρόσβαση στις υπηρεσίες ειδοποιήσεων χρήστη. Τέτοιες είναι οι ειδοποιήσεις στη notification bar, τα toast μηνύματα στο κάτω μέρος της οθόνης, η δόνηση του κινητού και η ενεργοποίηση της οθόνης, κλπ

- **Διαχειριστής Δραστηριοτήτων (Activity Manager)** – διαχειρίζεται τον κύκλο ζωής των δραστηριοτήτων και παρέχει δυνατότητα πλοήγησης από δραστηριότητα σε δραστηριότητα κρατώντας αποθηκευμένη στη μνήμη τη σειρά εκτέλεσης αυτών. Στο παρακάτω σχεδιάγραμμα φαίνεται λεπτομερώς ο κύκλος ζωής κάθε δραστηριότητας.



1.5 Στο εσωτερικό μιας εφαρμογής του Android

Κάθε εφαρμογή αποτελείται από ένα σύνολο αρχείων και φακέλων δομημένα σε μορφή project, τα οποία αφού γίνουν compiled μέσω του Android SDK μας δίνουν το αρχείο .apk. Το αρχείο αυτό αποτελεί την εφαρμογή και μπορούμε να εγκαταστήσουμε στις συσκευές μας. Ξεκινώντας, η κάθε εφαρμογή αποτελείται όπως είπαμε από πολλά αρχεία δομημένα σε φακέλους. Όλες οι εφαρμογές πρέπει να έχουν ένα μοναδικό όνομα πακέτου (package name) το οποίο χρησιμοποιείτε από το λειτουργικό σύστημα για αναγνώριση της εφαρμογής. Μια εφαρμογή μπορεί να αποτελείται από πολλά υποπακέτα, εφόσον αυτό είναι απαραίτητο λόγω της πολυπλοκότητας της εφαρμογής, αλλά μόνο από ένα κύριο.

1.5.1 Το αρχείο AndroidManifest.xml

Κάθε project εφαρμογής περιέχει ένα αρχείο στο οποίο βρίσκονται καταχωρημένες οι σημαντικότερες πληροφορίες της εφαρμογής, και το αρχείο αυτό ονομάζεται AndroidManifest.xml. Πρόκειται όπως λέει και το όνομα του για ένα αρχείο xml μέσα στο οποίο ο προγραμματιστής καταχωρεί τις σημαντικότερες πληροφορίες της εφαρμογής για χρήση από το λειτουργικό σύστημα. Κάποιες από αυτές τις πληροφορίες είναι:

- Το όνομα του πακέτου της εφαρμογής
- Το κανονικό της όνομα που φαίνεται στον χρήστη
- Η έκδοση των APIs που χρησιμοποιούνται
- Ο αριθμός έκδοσης της εφαρμογής
- Οι άδειες χρήσης που ζητάει η εφαρμογή
- Όλες οι δραστηριότητες, πάροχοι περιεχομένου, υπηρεσίες, κλπ, που περιέχει και χρησιμοποιεί η εφαρμογή.

Όπως αντιλαμβανόμαστε πρόκειται για πολύ σημαντικό αρχείο και αποτελεί κύριο συστατικό κάθε εφαρμογής.

1.5.2 Οι φάκελοι src & res

Στον φάκελο src (εκ του source) περιέχονται τα αρχεία κλάσης τις Java όλων των Activities, Services, Content Providers, βοηθητικά αρχεία, κλπ. Ο φάκελος περιέχει το πακέτο ή τα πακέτα της εφαρμογής τα οποία περιέχουν τα αρχεία Java, και αποτελεί τον μοναδικό φάκελο στο project στον οποίο αποθηκεύονται τα αρχεία του κώδικα μας. Ο φάκελος res (εκ του resources) περιέχει όλα τα αρχεία εικόνας, κειμένου, xml layout, κλπ τα οποία χρησιμοποιούνται από τις Activities που βρίσκονται στον φάκελο src. Φυσικά δεν βρίσκονται όλα τα αρχεία πόρων, σε έναν φάκελο, αλλά είναι χωρισμένα και ταξινομημένα σε υποφακέλους ανάλογα με το είδος τους. Συνηθισμένοι υποφάκελοι του κύριου φακέλου res, είναι ο φάκελος drawable ο οποίος περιέχει τα αρχεία εικόνας (.png, .jpg, .gif) τα οποία χρησιμοποιεί η εφαρμογή μας, ο φάκελος layout ο οποίος περιέχει όλα τα αρχεία xml τα οποία ορίζουν τα διάφορα layouts που υπάρχουν στην εφαρμογή, και τέλος ο φάκελος values στον οποίο αποθηκεύονται όλοι οι πόροι κειμένου που χρησιμοποιούνται στην εφαρμογή.

1.5.3 Οι υπόλοιποι φάκελοι του project

Ένα project αποτελείτε από περισσότερους από τους 3 βασικούς φακέλους, κάποιοι από τους οποίους μπορεί να θεωρηθούν και περιττοί αναλόγως την περίπτωση. Στο project λοιπόν περιλαμβάνονται και ο φάκελος με τα διαθέσιμα APIs αναλόγως την έκδοση που έχουμε επιλέξει να δουλέψουμε, ο φάκελος με τις διαθέσιμες βιβλιοθήκες που έχουμε εισάγει στο build path του project μας, και επίσης περιλαμβάνει και τις διαβαθμίσεις του φακέλου res, όπως είναι οι φάκελοι drawable-hdpi, drawable-mdpi, layout-port, menu, κλπ. Σε αυτούς περιλαμβάνονται τα ειδικά διαμορφωμένα αρχεία πόρων που έχουμε τοποθετήσει ώστε να είναι διαθέσιμα από το λειτουργικό σύστημα, αναλόγως την περίπτωση.

1.5.4 Δομικά Μέρη μιας Εφαρμογής

Παραπάνω αναφέραμε ότι όλα τα δομικά μέρη της εφαρμογής πρέπει να αναφέρονται αναλυτικά στο αρχείο `AndroidManifest.xml`, πια είναι όμως αυτά τα δομικά μέρη και πια η λειτουργία του καθενός;

- **Δραστηριότητες (Activities)**

Πρόκειται ίσως για το κύριο δομικό στοιχείο μιας εφαρμογής. Δραστηριότητα είναι μια οθόνη διεπαφής χρήστη (GUI) και προβολής πληροφοριών. Κάθε εφαρμογή έχει τόσες Activities όσες και οι διαφορετικές οθόνες οι οποίες εμφανίζονται στον χρήστη. Όλες οι δραστηριότητες συνεργάζονται μεταξύ τους για να δώσουν στον χρήστη μια συνολική εμπειρία χρήσης της εφαρμογής.

- **Προθέσεις (Intents)**

Οι δραστηριότητες επικοινωνούν και εναλλάσσουν την λειτουργία τους μέσω των Intents. Ουσιαστικά τα Intents εξασφαλίζουν την μετάβαση από την μία δραστηριότητα σε μια άλλη και επίσης χρησιμοποιούνται για ανταλλαγή δεδομένων. Η ανταλλαγή δεδομένων, μπορεί να γίνει είτε μεταξύ των Activities μιας εφαρμογής, είτε από τη μία εφαρμογή στην άλλη. Παραδείγματος χάρη μπορούμε μέσω ενός Intent να εκκινήσουμε έναν browser ώστε να μας ανοίξει απευθείας ένα url το οποίο έχουμε παρέχει εμείς μέσω ενός Intent.

- **Υπηρεσίες (Services)**

Πρόκειται για λειτουργίες της εφαρμογής οι οποίες είναι σχεδιασμένες να τρέχουν στο παρασκήνιο και να επιστρέφουν αποτελέσματά ακόμη και όταν η εφαρμογή δεν είναι στο προσκήνιο. Πχ μια εφαρμογή media player μπορεί μέσω μιας υπηρεσίας να συνεχίσει να παίζει μουσική ακόμη και όταν το κύριο παράθυρο της εφαρμογής δεν βρίσκεται στο προσκήνιο.

- **Πάροχος Περιεχόμενου (Content Providers)**

Η ανταλλαγή δεδομένων από μια εφαρμογή στην άλλη όπως είπαμε παραπάνω μπορεί να γίνει μέσω ενός Intent, ένας πάροχος περιεχομένου όμως έχει πιο σύνθετη λειτουργία. Οι content providers μιας εφαρμογής διαχειρίζονται συγκεκριμένα δεδομένα της εφαρμογής τα οποία έχει ορίσει ο προγραμματιστής κατά την κατασκευή του. Συνηθισμένα δεδομένα τα οποία μοιράζονται μέσω Content Providers, είναι οι βάσεις δεδομένων SQLite μιας εφαρμογής, και οι επαφές του χρήστη.

· Δέκτες Μετάδοσης (Broadcast Receivers)

Πρόκειται για ένα είδους υπηρεσία η οποία αντιλαμβάνεται κάποια γεγονότα του συστήματος και αναλαμβάνει να ενημερώσει το σύστημα ή τις υπόλοιπες εφαρμογές. Ο σκοπός τους είναι διπλός καθότι μπορούν και να ενημερωθούν για κάποιο συμβάν από άλλες εφαρμογές, αλλά και να ειδοποιήσουν τις υπόλοιπες εφαρμογές και το σύστημα για κάποιο συμβάν που τις ενεργοποίησε. Δεν έχουν γραφικό περιβάλλον αλλά μπορούν να προβάλλουν ειδοποίηση στον χρήστη μέσω της μπάρας ειδοποιήσεων. Συνήθως χρησιμοποιούνται ως διαμεσολαβητές μεταξύ των Activities και των Services μιας εφαρμογής.

1.6 Ασφάλεια στο Android

Τη στιγμή που μια εφαρμογή εγκαθίσταται στη συσκευή, λειτουργεί αποκλειστικά στο δικό της εικονική μηχανή η οποία αποτελεί και το πλαίσιο ασφαλείας (sandbox) της εφαρμογής. Το Android είναι ένα λειτουργικό σύστημα πολλών χρηστών στο οποίο:

- Η κάθε εφαρμογή αντιμετωπίζεται σαν διαφορετικός χρήστης
- Από προεπιλογή το σύστημα δίνει έναν μοναδικό αριθμό ID ο οποίος είναι άγνωστος στην εφαρμογή. Το σύστημα αναθέτει συγκεκριμένες άδειες χρήσης στα αρχεία της εφαρμογής, και μόνο η εφαρμογή με το σωστό ID μπορεί να έχει πρόσβαση σε αυτά.
- Κάθε εφαρμογή τρέχει στην δική της εικονική μηχανή (VM) απομονωμένη από τις υπόλοιπες εφαρμογές. Η κάθε VM εκκινεί μόλις ζητηθεί από το σύστημα και κλείνει είτε επειδή δεν χρησιμοποιείτε πλέον, είτε επειδή το σύστημα θέλει να ελευθερώσει τους πόρους της μνήμης για χρήση από άλλη εφαρμογή. Με αυτό τον τρόπο το Android χρησιμοποιεί την αρχή των ελαχίστων δικαιωμάτων. Η κάθε εφαρμογή έχει πρόσβαση μέσω του AndroidManifest μόνο σε όσους πόρους συστήματος χρειάζεται και κανέναν περισσότερο. Οι πόροι και τα δικαιώματα που απαιτούνται από μία εφαρμογή γίνονται γνωστά στον χρήστη τη στιγμή της εγκατάστασης της, και ο χρήστης μπορεί να επιλέξει να μην εγκαταστήσει μια εφαρμογή εφόσον δεν συμφωνεί να τις παρέχει πρόσβαση στους πόρους που ζητάει.

ΚΕΦΑΛΑΙΟ 2

Περιβάλλον Ανάπτυξης

2.1 Android Studio

Το android studio είναι το επίσημο περιβάλλον ανάπτυξης (ide - integrated development environment) για τις εφαρμογές android. Ανακοινώθηκε το 2013 σε συνέδριο της google, χρησιμοποιείτε αποκλειστικά για εφαρμογές android και η τελευταία έκδοσή του είναι η 2.2.1.

2.2 Εγκατάσταση

Για να ελέγξουμε αν έχουμε εγκαταστήσει τη σωστή έκδοση του JDK (6 ή ανώτερη) πληκτρολογούμε *javac -version* σε ένα terminal Εκτελούμε το αρχείο **.exe** και ακολουθούμε το setup wizard για την εγκατάσταση του Android Studio και των αναγκαίων εργαλείων SDK. Σε περίπτωση που **δεν** είναι δυνατόν να εντοπισθεί η εγκατάσταση της Java πρέπει να υποδειχθεί η σωστή τοποθεσία:

Start menu > Computer > System Properties > Advanced System Properties.

Στη συνέχεια: **Advanced tab > Environment Variables.** Προσθέτουμε νέα μεταβλητή συστήματος: **JAVA_HOME** που «δείχνει» στον κατάλογο JDK, π.χ. C:\Program Files\Java\jdk1.7.0_21. Τα επιμέρους εργαλεία και άλλα SDK packages βρίσκονται εκτός του Android Studio application directory. Για άμεση πρόσβαση, μπορούμε μέσω ενός terminal να μεταφερθούμε στην αντίστοιχη τοποθεσία. Π.χ. \Users\<user>\sdk\

Μεταβλητές χρήστη Gebruiker

Μεταβλητή	Τιμή
path	C:\Program Files\Java\jdk1.8.0_65\bin
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Δημιουργία...

Επεξεργασία...

Διαγραφή

Μεταβλητές συστήματος

Μεταβλητή	Τιμή
ComSpec	C:\WINDOWS\system32\cmd.exe
FP_NO_HOST_CHECK	NO
JAVA_HOME	C:\Program Files\Java\jdk1.8.0_92
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	C:\ProgramData\Oracle\Java\javapath;C:\Program Files (x86)\...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64

Δημιουργία...

Επεξεργασία...

Διαγραφή

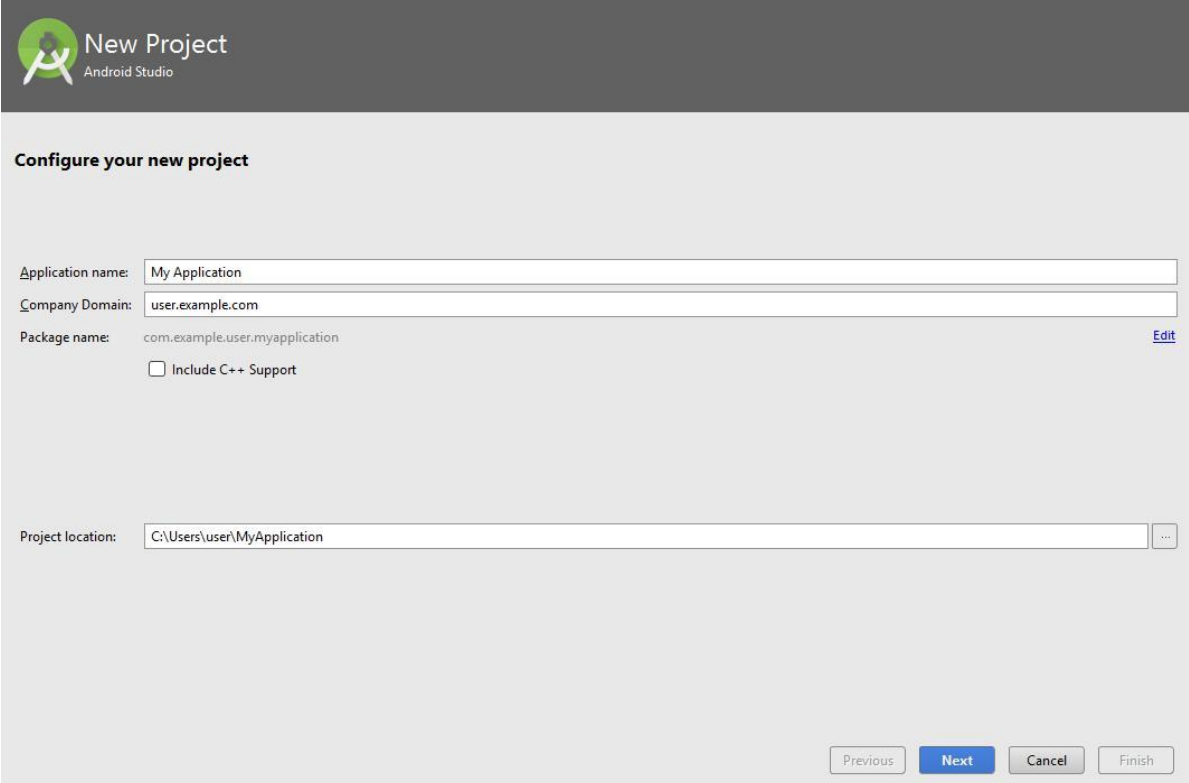
OK

Άκυρο

2.3 Δημιουργία νέου project

Στην οθόνη καλωσορίσματος: Επιλογή **New Project**

Εκτός οθόνης καλωσορίσματος: **File | New Project** για έναρξη του *New project wizard*



New Project
Android Studio

Configure your new project

Application name: My Application

Company Domain: user.example.com

Package name: com.example.user.myapplication [Edit](#)

☐ Include C++ Support

Project location: C:\Users\user\MyApplication

Previous Next Cancel Finish

Πεδία:

- **Application name:** Όνομα που φαίνεται στο Google Play και βλέπει ο χρήστης.
- **Company Domain.** Μοναδικό αναγνωριστικό της εφαρμογής. Χρήσιμο όταν αποφασίσουμε να ανεβάσουμε την εφαρμογή στο Google Play.
- **Package name:** Μοναδικό αναγνωριστικό εφαρμογής συνήθως της μορφής *com.company_name.app_name* ή *reverse_company_domain.app_name*
- **Project location:** Κατάλογος του project στο σύστημά μας.
- **Minimum SDK:** Το ελάχιστο SDK που υποστηρίζεται από την εφαρμογή. Εάν η εφαρμογή δεν απαιτεί συγκεκριμένα χαρακτηριστικά που περιέχονται στα νεότερα SDKs, τότε επιλέγουμε μία παλαιότερη API (Application Programming Interface).



Select the form factors your app will run on

Different platforms may require separate SDKs

☒ Phone and Tablet

Minimum SDK

Lower API levels target more devices, but have fewer features available.

By targeting API 15 and later, your app will run on approximately **97.4%** of the devices that are active on the Google Play Store.

[Help me choose](#)

☐ Wear

Minimum SDK

☐ TV

Minimum SDK

☐ Android Auto

☐ Glass

Minimum SDK

Previous

Next

Cancel

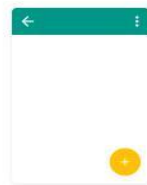
Finish



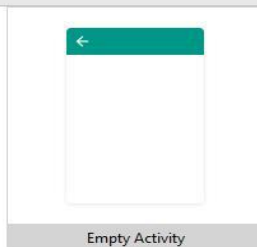
Add an Activity to Mobile



Add No Activity



Basic Activity



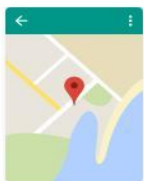
Empty Activity



Fullscreen Activity



Google AdMob Ads Activity



Google Maps Activity



Login Activity



Master/Detail Flow



Navigation Drawer Activity



Scrolling Activity

Previous

Next

Cancel

Finish

Basic Activity: Δημιουργεί μία *basic activity* με μία *app bar*.

Empty Activity: Δημιουργεί μία κενή δραστηριότητα

Fullscreen Activity. Η *full-screen mode* εναλλάσσεται με την ορατότητα της διεπαφής χρήστη όποτε ο χρήστης αγγίζει την οθόνη

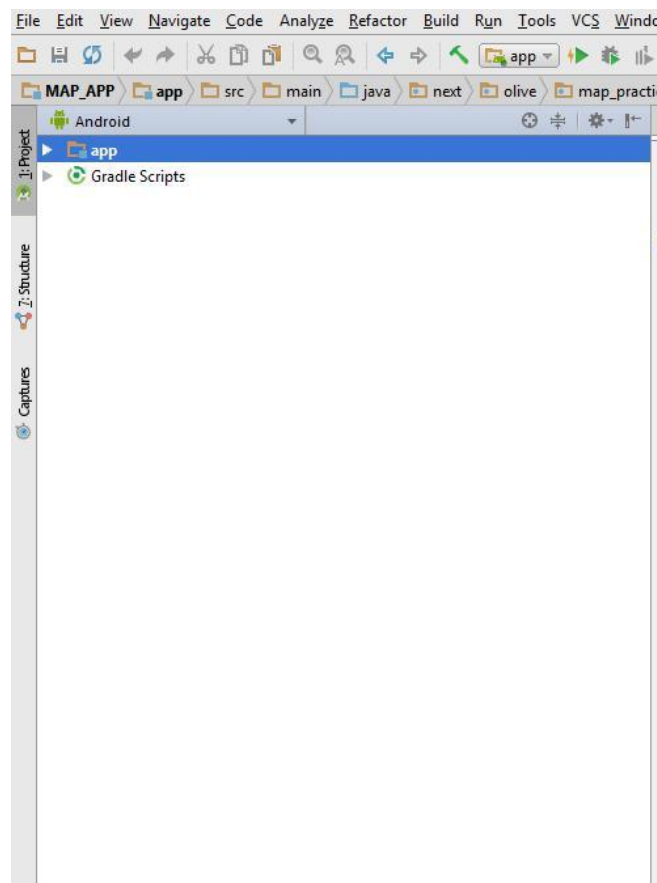
Login activity: Επιτρέπει στους χρήστες να κάνουν login ή εγγραφή με ένα e-mail και ένα κωδικό πρόσβασης.

Master/Detail Flow: Χωρίζει την οθόνη σε 2 περιοχές: Στην αριστερή ένα μενού και οι λεπτομέρειες του επιλεγέντος αντικειμένου στη δεξιά.

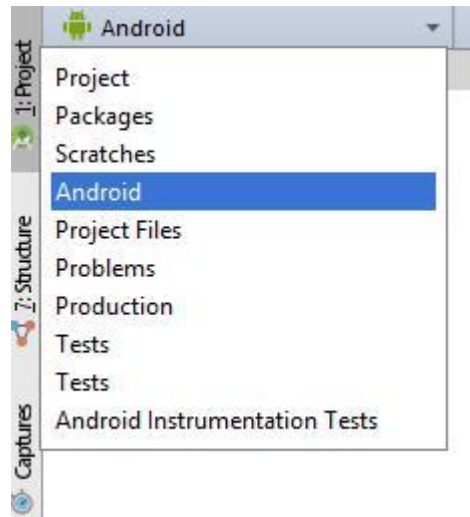
2.4 Πλοήγηση στο Project

2.4.1 To project panel

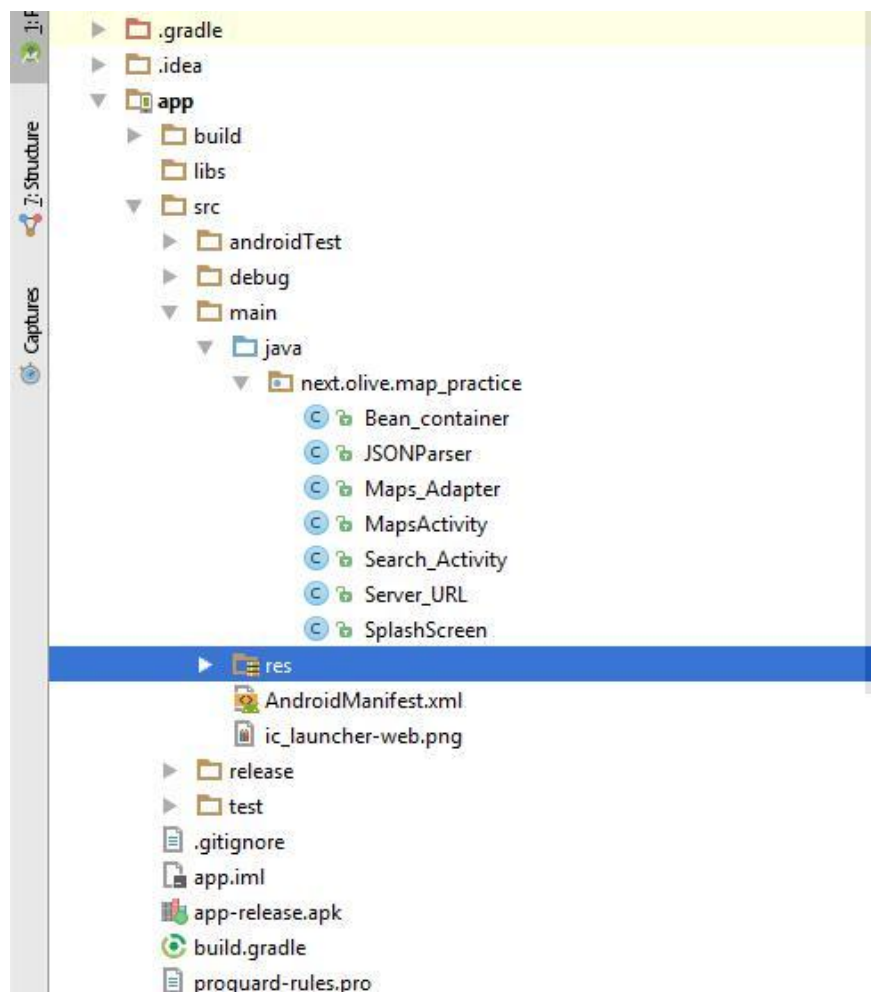
Πιέζουμε *Alt+1* για να ανοίξουμε την project view (λίστα των ανοιχτών projects). Στην άνω αριστερή γωνία μπορούμε να εναλλάσσουμε **project** και **package views**



Στην άνω δεξιά γωνία υπάρχουν ορισμένες *ενέργειες (actions)* και ένα *drop-down menu* που διαμορφώνουν την *project view*.



2.4.2 Δομή του project



Στο **project navigation pane**, βλέπουμε τη δομή του project. Μέσα στη δομή αυτή βρίσκεται ένας φάκελος με το όνομα της εφαρμογής μας, που περιέχει τη δομή της εφαρμογής και τα αρχεία. Το σπουδαιότερα στοιχεία στη δομή αυτή είναι:

build/: Περιέχει τους μεταγλωττισμένους πόρους (resources) μετά το χτίσιμο της εφαρμογής και τις κλάσεις που παράγονται από τα εργαλεία Android όπως το αρχείο *R.java*, που περιέχει τις αναφορές στους πόρους της εφαρμογής.

libs/: Περιέχει τις βιβλιοθήκες στις οποίες αναφέρεται ο κώδικάς μας.

src/main/: Περιέχει τις πηγές (sources) της εφαρμογής. Όλα τα αρχεία με τα οποία συνήθως δουλεύουμε βρίσκονται σ' αυτόν τα φάκελο.

Περιέχει τους υποφακέλους:

java/: Περιέχει τις κλάσεις της Java οργανωμένες ως *packages*.

Κάθε κλάση που δημιουργούμε θα βρίσκεται στον χώρο ονομάτων του project package (**com.example.myapplication**). Όταν δημιουργήσαμε το project, δημιουργήσαμε και την κύρια δραστηριότητα, άρα η κλάση *activity* πρέπει να βρίσκεται σ' αυτό το πακέτο.

res/: Περιέχει πόρους του project όπως τα αρχεία XML που προσδιορίζουν τα layouts και menus ή τα αρχεία εικόνων.

drawable/: Περιέχει αρχεία bitmap (PNG, JPEG, ή GIF), αρχεία 9-Patch image, και αρχεία XML που περιγράφουν Drawable σχήματα ή Drawable αντικείμενα που περιέχουν πολλαπλές καταστάσεις (normal, pressed, ή focused).

mipmap/: Περιέχει τα εικονίδια εκκίνησης της εφαρμογής. Το σύστημα διατηρεί τους πόρους στον φάκελο αυτό (όπως και τους φακέλους συγκεκριμένης πυκνότητας του τύπου *mipmap-xxxhdpi*) ανεξάρτητα από την ανάλυση οθόνης της συσκευής που είναι εγκατεστημένη η εφαρμογή μας. Αυτό επιτρέπει στις εφαρμογές εκκίνησης να επιλέγουν την βέλτιστη ανάλυση εικόνας για να εμφανίσουν στην home screen

layout/: Περιέχει τους XML ορισμούς των όψεων (views) και των στοιχείων τους.

menu/: Περιέχει τους XML ορισμούς των μενού της εφαρμογής.

values/: Περιέχει τα XML αρχεία που ορίζουν σύνολα ζευγών 'ονομα-τιμή' (name-value). Οι τιμές αυτές μπορεί να είναι χρώματα, αλφαριθμητικά ή στυλ. Υπάρχουν διαφορετικοί φάκελοι values που κατηγοριοποιούνται σύμφωνα με τις διαφορετικές screens options για να προσαρμόσουν την διεπαφή σ' αυτές. Για παράδειγμα, να διευρύνουν τα components ή τα fonts όταν η εφαρμογή τρέχει σε τάμπλετ.

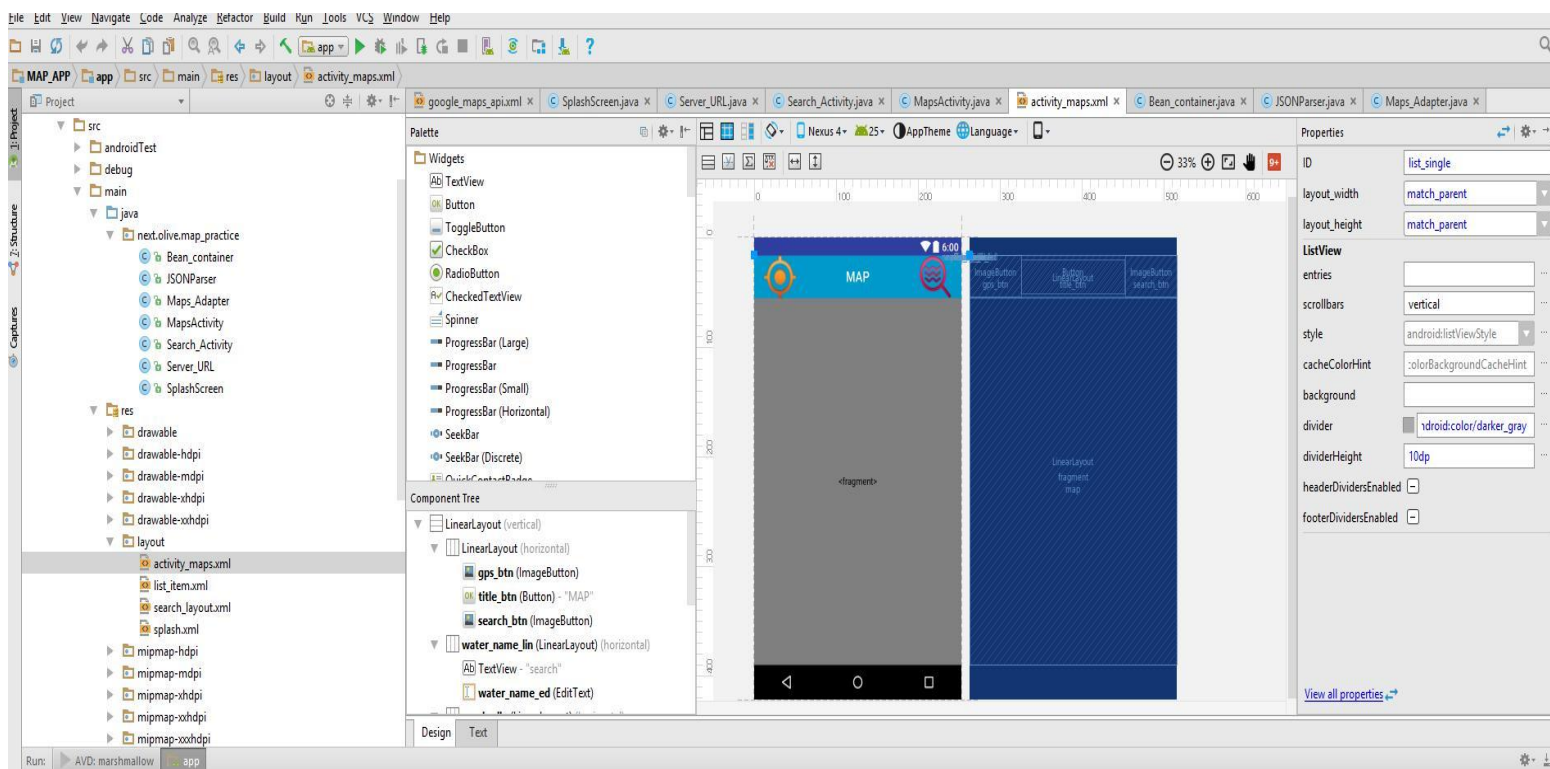
AndroidManifest.xml: Το αρχείο αυτό παράγεται αυτόματα όταν δημιουργούμε το project. Δηλώνει τις βασικές πληροφορίες που χρειάζεται το σύστημα Android για να τρέξει την εφαρμογή: όνομα

package, έκδοση, δραστηριότητες, άδειες, προθέσεις, ή απαιτούμενο υλικό.

build.gradle: Το αρχείο αυτό είναι το script που χρησιμοποιούμε για να χτίσουμε την εφαρμογή μας.

2.5.1 Graphical Editor

Ανοίγουμε το **main layout** που βρίσκεται στο **/src/main/res/layout/activity_main.xml** του project. Ο graphical editor θα ανοίξει εξ ορισμού. Αρχικά, αυτό το main layout περιέχει μόνο μία text view με ένα μήνυμα “*Hello world!* “. Για εναλλαγή μεταξύ του graphical και του text editor, κάνουμε κλικ στα bottom tabs, **Design** και **Text**.



Η **toolbar** περιέχει ορισμένες επιλογές αλλαγής του στυλ του layout και preview.

Το **components tree** εμφανίζει τα συστατικά (components) της διάταξης (layout) ως ιεραρχία.

Ο **properties inspector** δείχνει τις ιδιότητες του επιλεγμένου συστατικού από το layout και επιτρέπει να τις αλλάξουμε.

Η **palette** παρέχει τα υπάρχοντα συστατικά της διεπαφής χρήστη (User Interface –UI components) που μπορούν να τοποθετηθούν στο layout. Τα components είναι οργανωμένα σε διαφορετικές κατηγορίες :

Layouts: Ένα layout είναι ένα αντικείμενο container για την κατανομή των συστατικών στην οθόνη. Το αρχικό στοιχείο (root element) μίας διεπαφής χρήστη είναι ένα layout object, αλλά τα layouts μπορούν επίσης να περιέχουν περισσότερα layouts, δημιουργώντας έτσι μία ιεραρχία συστατικών δομημένη σε layouts. Η ιεραρχία των layouts πρέπει να παραμείνει όσο γίνεται απλή. Το main layout έχει ως βασικό στοιχείο (root element) το Relative Layout

Widgets: Η κατηγορία αυτή περιέχει τα πιο κοινά συστατικά των περισσότερων layouts: Buttons, checkboxes, text views, switches, image views, progress bars, spinners, web views.

Text Fields: Πεδία όπου ο χρήστης μπορεί να πληκτρολογήσει κείμενο. Η διαφορά μεταξύ τους είναι ο τύπος κειμένου που ο χρήστης μπορεί να εισάγει.

Containers: Τα containers ομαδοποιούν συστατικά με κοινή συμπεριφορά: Radio groups, list views, scroll views, tab hosts κ.α.

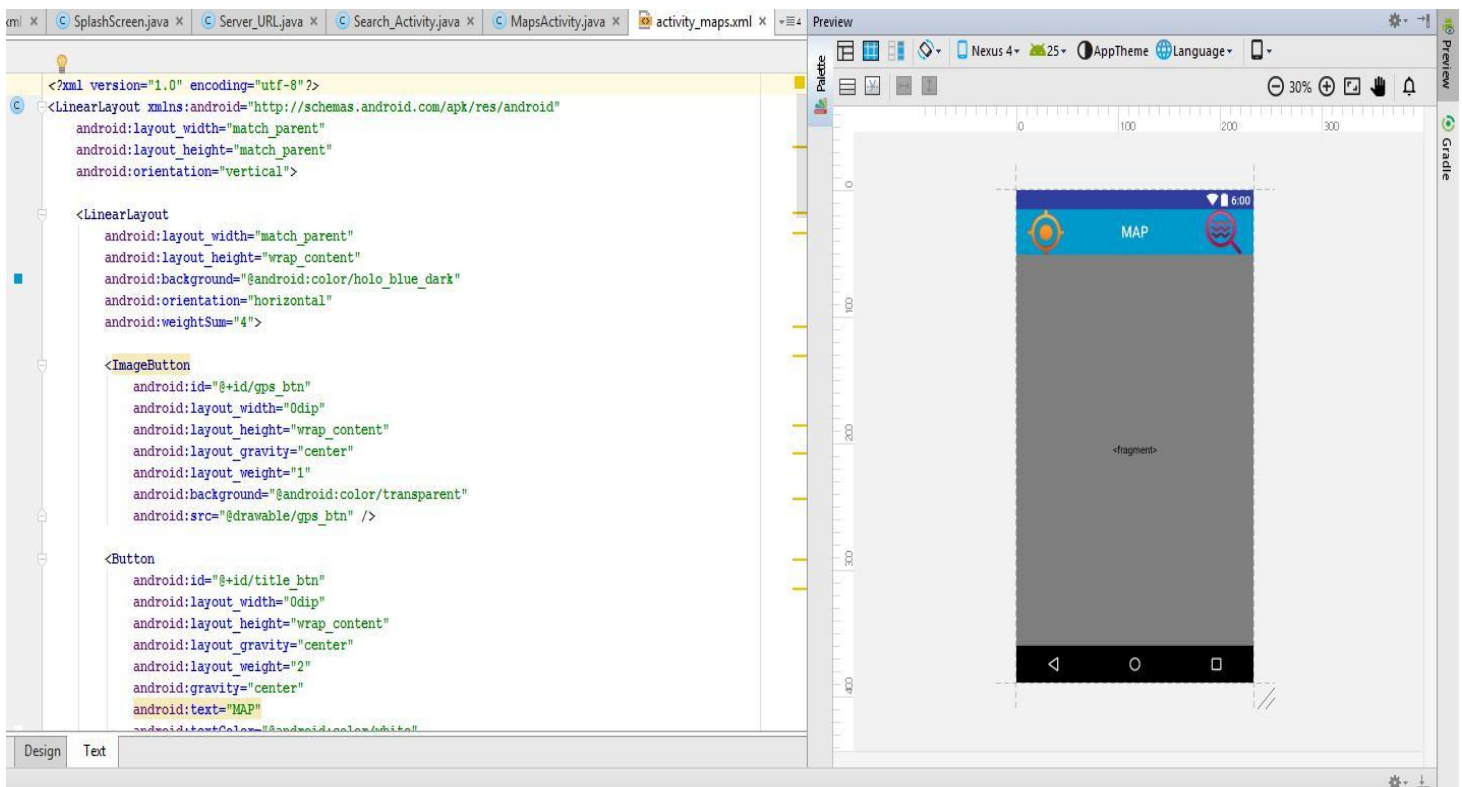
Date & Time: Συστατικά που αφορούν ημερομηνία και ώρα όπως: ημερολόγια ή ρολόγια

Expert: Συστατικά λιγότερο κοινά από αυτά που περιέχονται στην κατηγορία widgets.

Custom: Συστατικά που μας επιτρέπουν την συμπερίληψη των δικών μας προσαρμοσμένων συστατικών (custom components), που είναι συνήθως άλλα layouts από το project.

2.5.2 Text-based editor

Εναλλάσσεται με τον graphical editor κάνοντας κλικ στο Text tab. Η toolbar είναι ίδια με τον graphical editor. Η preview εμφανίζει το layout αλλά δεν μπορεί να το αλλάξει, θα πρέπει να χρησιμοποιήσουμε το design tab αν θέλουμε αλλαγές. Τα components προστίθενται στο layout χρησιμοποιώντας τις δηλώσεις XML. Οι ιδιότητες (properties) διαμορφώνονται επίσης με χρήση των δηλώσεων XML. Όπως και ο graphical editor, ο text editor εμφανίζει μόνο το στοιχείο text view μέσα στο root layout.



2.5.3 Πολλαπλές οθόνες

Δυνατότητα αλλαγής την προεπισκόπηση της διάταξης (layout preview) όταν είμαστε στο design mode.

Πρέπει να ελέγχουμε όλες τις διαμορφώσεις οθόνης που υποστηρίζει η εφαρμογή μας για να βεβαιωθούμε ότι εμφανίζονται σωστά.

Οι κατηγορίες πυκνότητας οθόνης στο Android είναι:

ldpi (low-density dots per inch): Περίπου 120 dpi

mdpi (medium-density dots per inch): Περίπου 160 dpi

hdpi (high-density dots per inch): Περίπου 240 dpi

xhdpi (extra-high-density dots per inch): Περίπου 320 dpi

xxhdpi (extra-extra-high-density dots per inch): Περίπου 480 dpi

Δυνατότητα αλλαγής του προσανατολισμού της συσκευής μέσω του layout state option στη γραμμή εργαλείων . Αν δεν εμφανίζεται σωστά, δημιουργούμε παραλλαγή :

Κλικ στο πρώτο εικονίδιο του *toolbar configuration option* και επιλέγουμε **Create Landscape Variation**: ανοίγει νέο layout στο editor/src/main/res/layout-land/activity_main.xml και χρησιμοποιεί το ίδιο όνομα με το portrait layout

Κάνουμε τις αλλαγές σύμφωνα με τον προσανατολισμό landscape Παρομοίως μπορούμε να δημιουργήσουμε παραλλαγή του layout για υπερμεγέθεις οθόνες Επιλέγουμε **Create layout-xlarge Variation**

2.5.4 Διαχείριση Συμβάντων

Τα συμβάντα (events) στο Android παράγονται όταν ο χρήστης αλληλεπιδρά με την εφαρμογή. Όλα τα widgets της διεπαφής (UI widgets) είναι απόγονοι της κλάσης *View* και μοιράζονται τα ίδια συμβάντα που διαχειρίζονται οι παρακάτω ακροατές (listeners):

OnClickListener: Αντιστοιχίζει το συμβάν όταν ο χρήστης κάνει κλικ στο view element

OnCreateContextMenu: Αντιστοιχίζει το συμβάν όταν ο χρήστης πατάει παρατεταμένα το view element και θέλουμε να ανοίξουμε ένα context menu

OnDragListener: Αντιστοιχίζει το συμβάν όταν ο χρήστης μεταφέρει και αποθέτει το στοιχείο συμβάντος

OnFocusChange: Αντιστοιχίζει το συμβάν όταν ο χρήστης κάνει πλοηγείται από ένα στοιχείο σε στοιχείο στην ίδια view

OnKeyListener: Αντιστοιχίζει το συμβάν όταν ο χρήστης πιέζει οποιοδήποτε πλήκτρο ενώ η εστίαση είναι στο view element

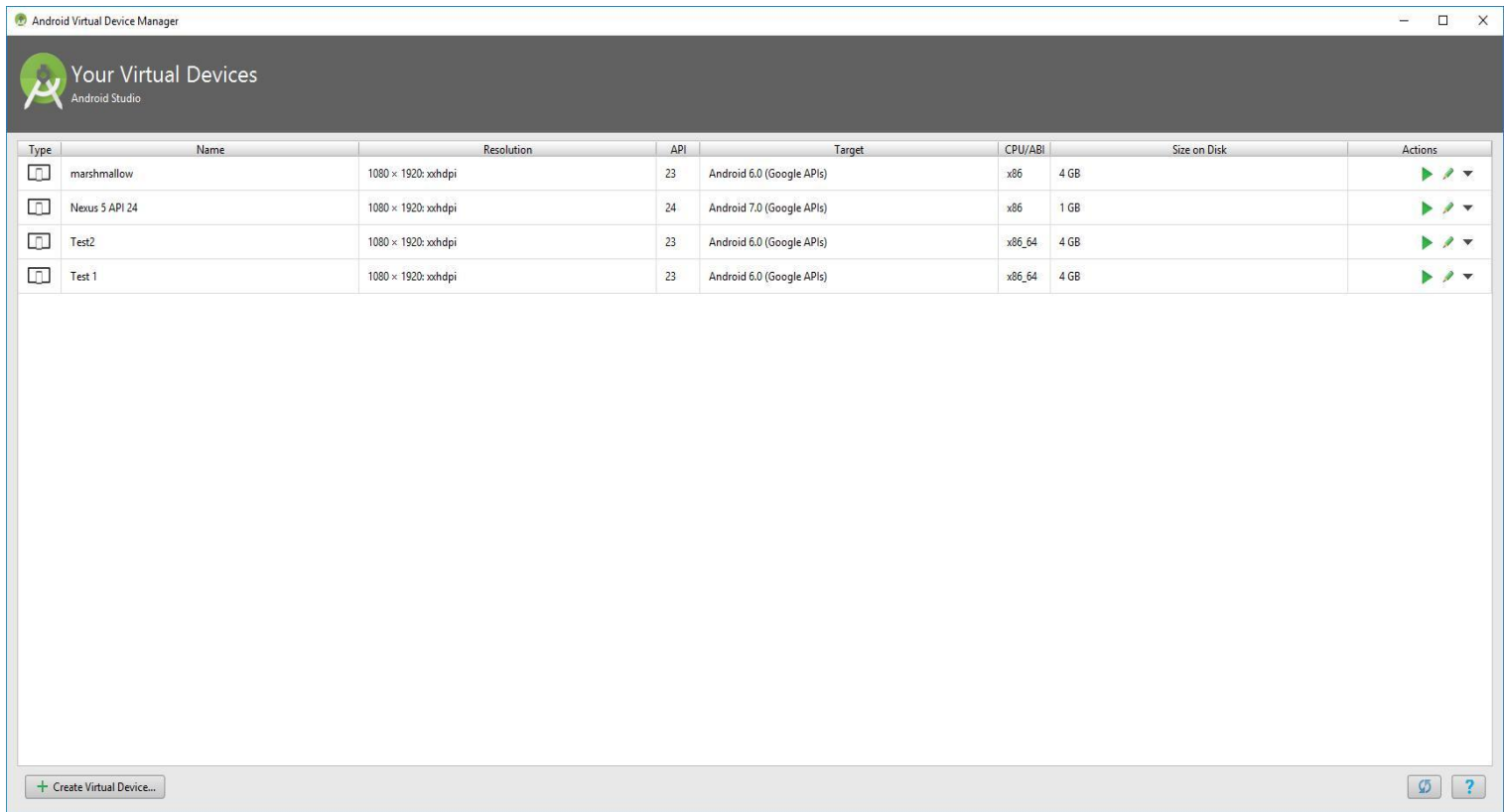
OnLongClickListener: Αντιστοιχίζει το συμβάν όταν ο χρήστης αγγίζει το view element και το κρατάει πατημένο

OnTouchListener: Αντιστοιχίζει το συμβάν όταν ο χρήστης αγγίζει το view element

2.6 Android Virtual Device (AVD) Manager

Ο Android Virtual Device Manager (AVD Manager) είναι ένα εργαλείο Android ενσωματωμένο στο Android Studio για την διαχείριση των εικονικών συσκευών Android που θα εκτελούνται στον Android emulator.

Ανοίγουμε τον AVD Manager από τον μενού **Tools | Android | AVD Manager**. Επίσης υπάρχει αντίστοιχο shortcut στη γραμμή εργαλείων. Ο AVD Manager εμφανίζει την λίστα των διαμορφωμένων εικονικών συσκευών.



2.7 Εκτέλεση και Debugging

Για απευθείας εκτέλεση μίας εφαρμογής επιλέγουμε **Run | Run 'MyApplication'**. Επίσης μπορούμε να χρησιμοποιήσουμε το *play icon button* στη γραμμή εργαλείων.

Για απασφαλμάτωση μίας εφαρμογής, επιλέγουμε **Run | Debug 'MyApplication'** ή κάνουμε κλικ στο *bug icon* στη γραμμή εργαλείων. Ανοίγει παράθυρο διαλόγου για επιλογή της συσκευής:

(α) Επιλέγουμε μία συσκευή από τον κατάλογο πραγματικών ή εικονικών συσκευών.

(β) Ξεκινούμε ένα νέο στιγμιότυπο του emulator, επιλέγουμε την επιθυμητή εικονική συσκευή και πατάμε OK. Την επόμενη φορά είτε κατά την εκτέλεση είτε κατά την απασφαλμάτωση, ο emulator εκτελείται ήδη, οπότε επιλέγουμε το (α).

Κατά την απασφαλμάτωση, στο κάτω μέρος της οθόνης υπάρχουν 3 tabs: **Console**: εμφανίζει τα events που λαμβάνουν χώρα όταν ξεκινάει ο emulator.

Waiting for device: Σημείο εκκίνησης του emulator.

Uploading file: Η εφαρμογή «πακετάρεται» και αποθηκεύεται στη συσκευή

Installing: Η εφαρμογή εγκαθίσταται στη συσκευή.

Launching application: Εκκίνηση της εφαρμογής.

Waiting for process: Η εφαρμογή εκτελείται και το σύστημα απασφαλμάτωσης επιχειρεί να συνδεθεί με την διαδικασία στη συσκευή.

Debug: Διαχειρίζεται τα breakpoints, ελέγχει την εκτέλεση του κώδικα και δείχνει πληροφορίες σχετικά με τις μεταβλητές. Για να προσθέσουμε ένα breakpoint στον κώδικά μας, κάνουμε κλικ στο αριστερό άκρο της γραμμής. Εμφανίζεται μία κόκκινη κουκίδα. Ξεκινάμε τον debugger ξανά

LogCat: Εμφανίζει όλα τα log μηνύματα που παράγονται από το σύστημα στη συσκευή εκτέλεσης. Τα μηνύματα αυτά έχουν διάφορα επίπεδα σημασίας που λειτουργούν ως φίλτρα και ενεργοποιούνται από το LogCat tab. Για να εκτυπώσουμε τα μηνύματα από τον κώδικά μας πρέπει να εισάγουμε την Log class. Η κλάση αυτή έχει μία μέθοδο για κάθε επίπεδο: v (verbose), d (debug), i (information), w (warning), e (error). Οι μέθοδοι αυτές δέχονται δύο παραμέτρους string. Η πρώτη ταυτοποιεί την source κλάση του μηνύματος και η δεύτερη το ίδιο το μήνυμα.

2.8 Προετοιμασία της εφαρμογής για κυκλοφορία

Οι εφαρμογές Android «πακετάρονται» σε ένα αρχείο με επέκταση .APK, που είναι μία παραλλαγή Java JAR (Java Archive) αρχείου. Αυτά τα αρχεία είναι συμπίεσμένα ZIP αρχεία. Ένα APK περιέχει συνήθως:

- assets/:** Κατάλογος που περιέχει τα *assets* αρχεία της εφαρμογής (κατάλογος assets του project).

- META-INF/:** Κατάλογος που περιέχει τα *certificates*.

- lib/:** Κατάλογος που περιέχει μεταγλωττισμένο κώδικα να χρειάζεται για κάποιον επεξεργαστή.

- res/:** Κατάλογος που περιέχει του πόρους της.

AndroidManifest.xml: Το αρχείο manifest της εφαρμογής.

classes.dex: Αρχείο που περιέχει τον μεταγλωττισμένο κώδικα της εφαρμογής.

resources.arsc: Αρχείο που περιέχει ορισμένους προ-μεταγλωττισμένους πόρους.

Με το αρχείο .APK, η εφαρμογή μπορεί να διανεμηθεί και να εγκατασταθεί.

ΚΕΦΑΛΑΙΟ 3

Εφαρμογή Beachapp

Η εφαρμογή αυτή δημιουργήθηκε με σκοπό να παρέχει στο χρήστη της τη δυνατότητα να μπορεί να αναζητήσει παραλίες με γαλάζια σημαία στο ιόνιο πέλαγος και με τη χρήση του gps να οδηγηθεί και να προσανατολισθεί. Μπορεί να επιλέξει μέσα από μια λίστα παραλιών ή απλά να ορίσει μια ακτίνα χιλιομέτρων και η εφαρμογή να εντοπίσει και να εμφανίσει όλες παραλίες είναι στο εύρος που έχει ορίσει. Σε αυτό το κεφάλαιο θα παρουσιαστεί η εφαρμογή που υλοποιήσαμε και όλες οι λειτουργίες της.

3.1 Αρχείο AndroidManifest

Όλες οι εφαρμογές Android πρέπει να έχουν ένα αρχείο androidmanifest (androidmanifest.xml). Το συγκεκριμένο αρχείο παρέχει, στο σύστημα android που θα τρέξει η εφαρμογή, σημαντικές πληροφορίες. Περιλαμβάνει τα ονόματα των java πακέτων, περιγράφει τα activities, services και άλλα χρήσιμα εργαλεία, δηλώνει τις άδειες της εφαρμογής, τις βιβλιοθήκες και το ελάχιστο επίπεδο api που θα τρέχει η εφαρμογή. Στη συγκεκριμένη εργασία έχουμε δηλώσει ως όνομα της εφαρμογής το Beach app, και έχουμε πάρει άδειες όπως είναι η πρόσβαση στο διαδίκτυο και η δυνατότητα να ελέγχουμε αν είμαστε συνδεδεμένοι στο διαδίκτυο ή όχι (android.permission.INTERNET, android.permission.ACCESS_NETWORK_STATE, android.permission.CHANGE_NETWORK_STATE), άδεια για να μπορούμε να χρησιμοποιήσουμε το δέκτη GPS της συσκευής αλλά και για να γνωρίζουμε άμα είναι ενεργοποιημένο το GPS (android.permission.ACCESS_COARSE_LOCATION, android.permission.ACCESS_FINE_LOCATION). Επίσης δηλώθηκε και το κατάλληλο Api key για την χρησιμοποίηση των google maps (android:name=" com.google.android.geo.API_KEY, android:value=" AIzaSyBJLEOuGrC8AgBkxjUfLocx0vDudeXULzU).

3.2 Περιγραφή της Εφαρμογής

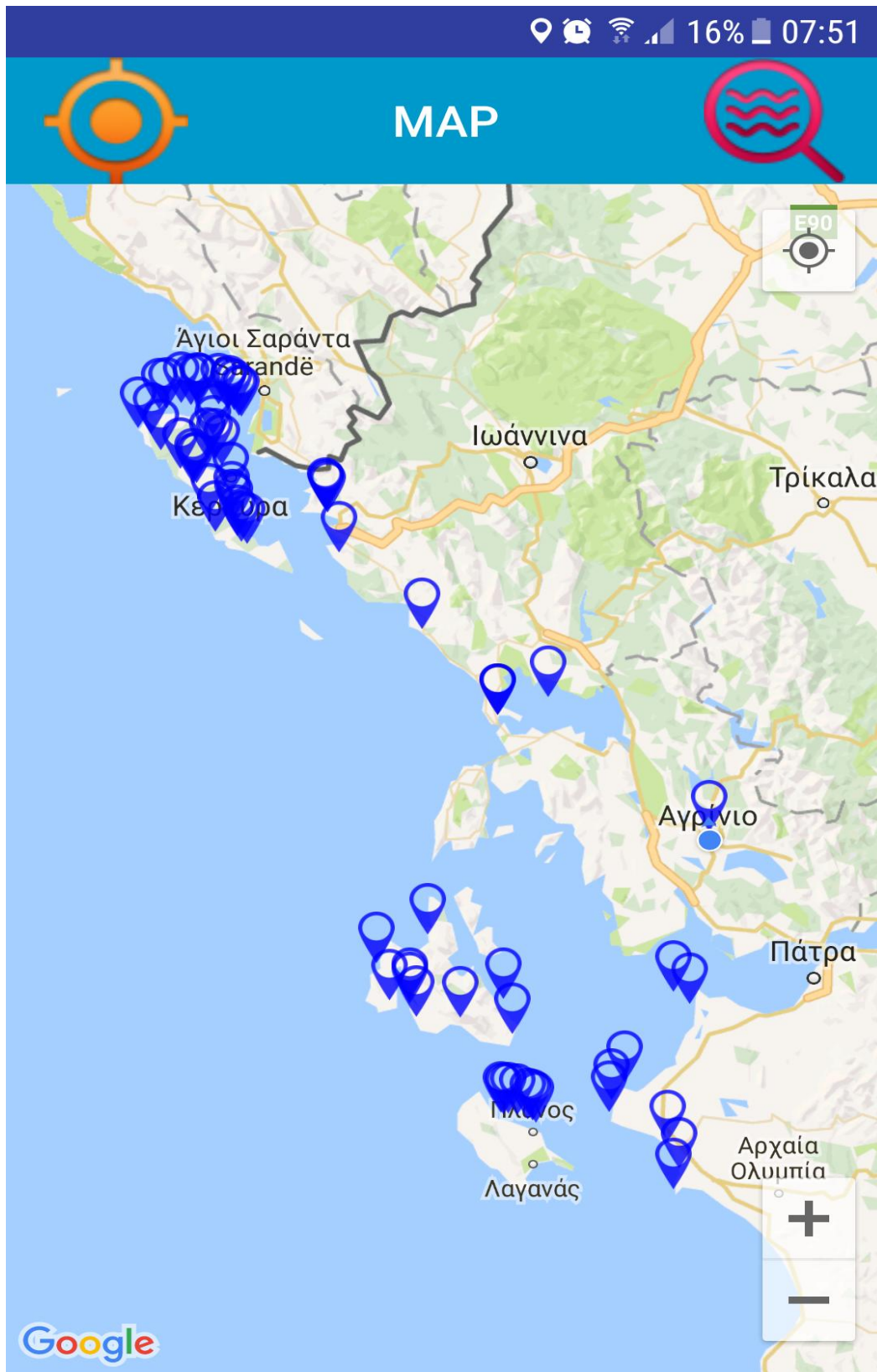
Η εφαρμογή όπως είπαμε ονομάζεται beachapp και παρακάτω φαίνεται το εικονίδιό της



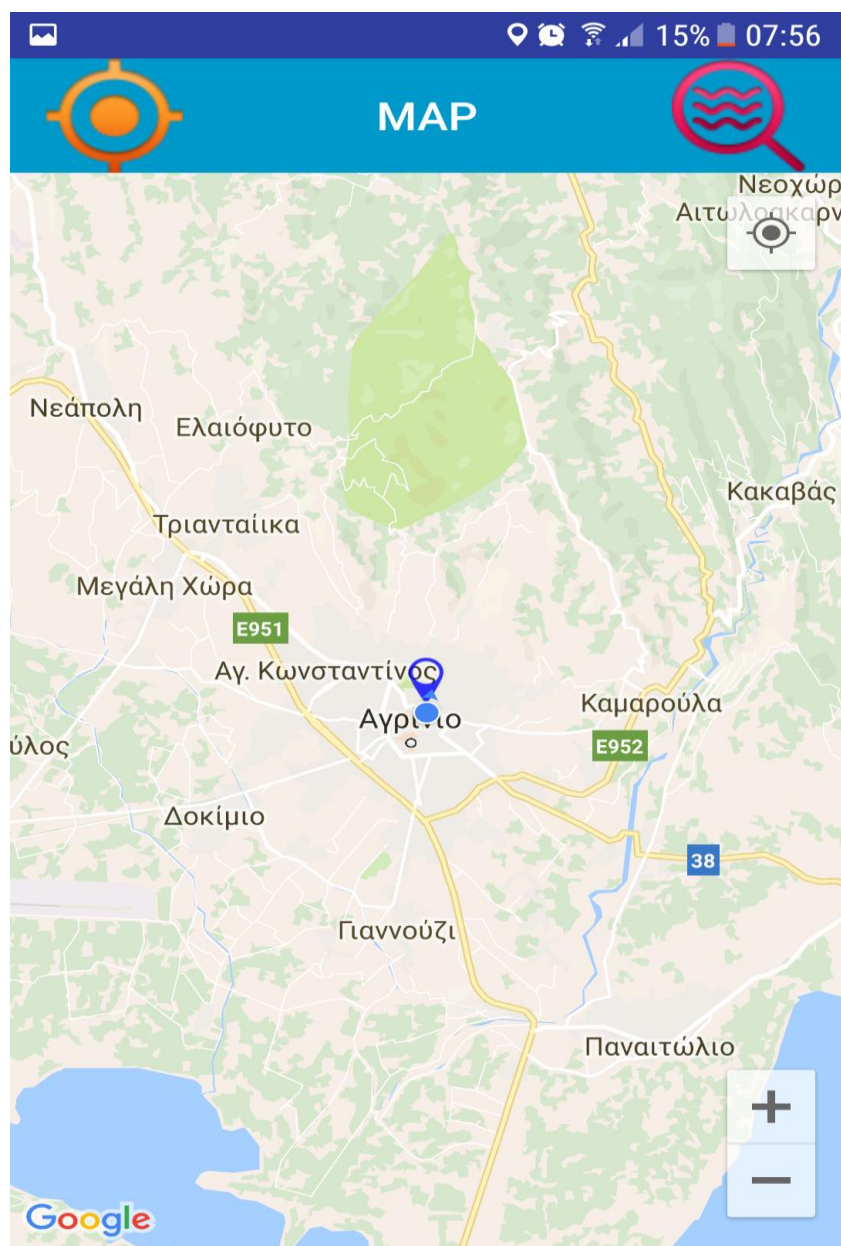
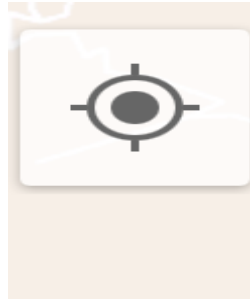
Ανοίγοντας την εφαρμογή, πριν μπούμε στο αρχικό μενού, εμφανίζεται για λίγα δευτερόλεπτα μια εικόνα με το ονοματεπώνυμό μου και τον αριθμό μητρώου μου.



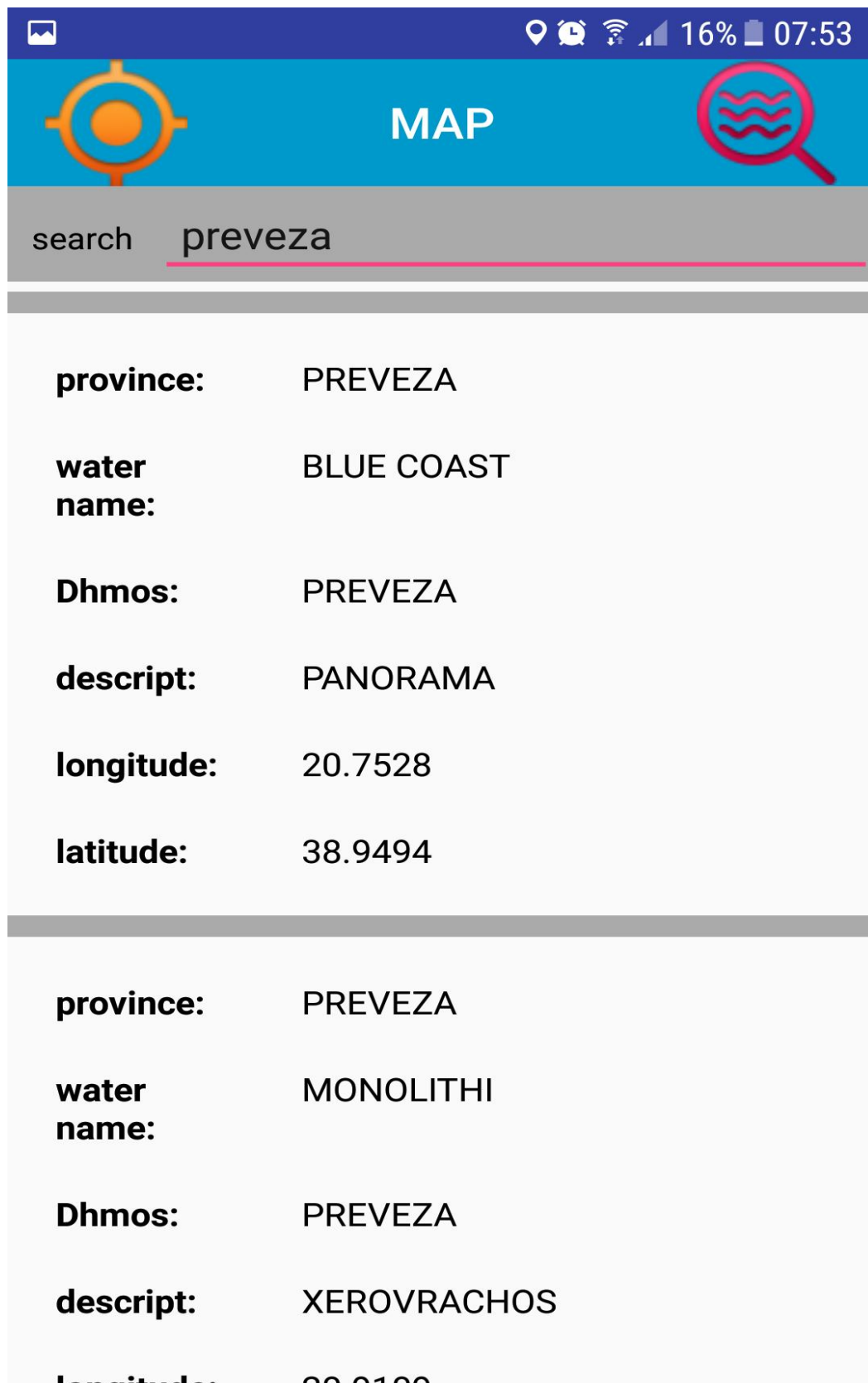
Έπειτα από την αρχική οθόνη ακολουθεί η κεντρική οθόνη που δείχνει τις διαθέσιμες παραλίες και περιέχει το Gps button, τη λίστα με τις παραλίες και το κουμπί ορισμού απόστασης.



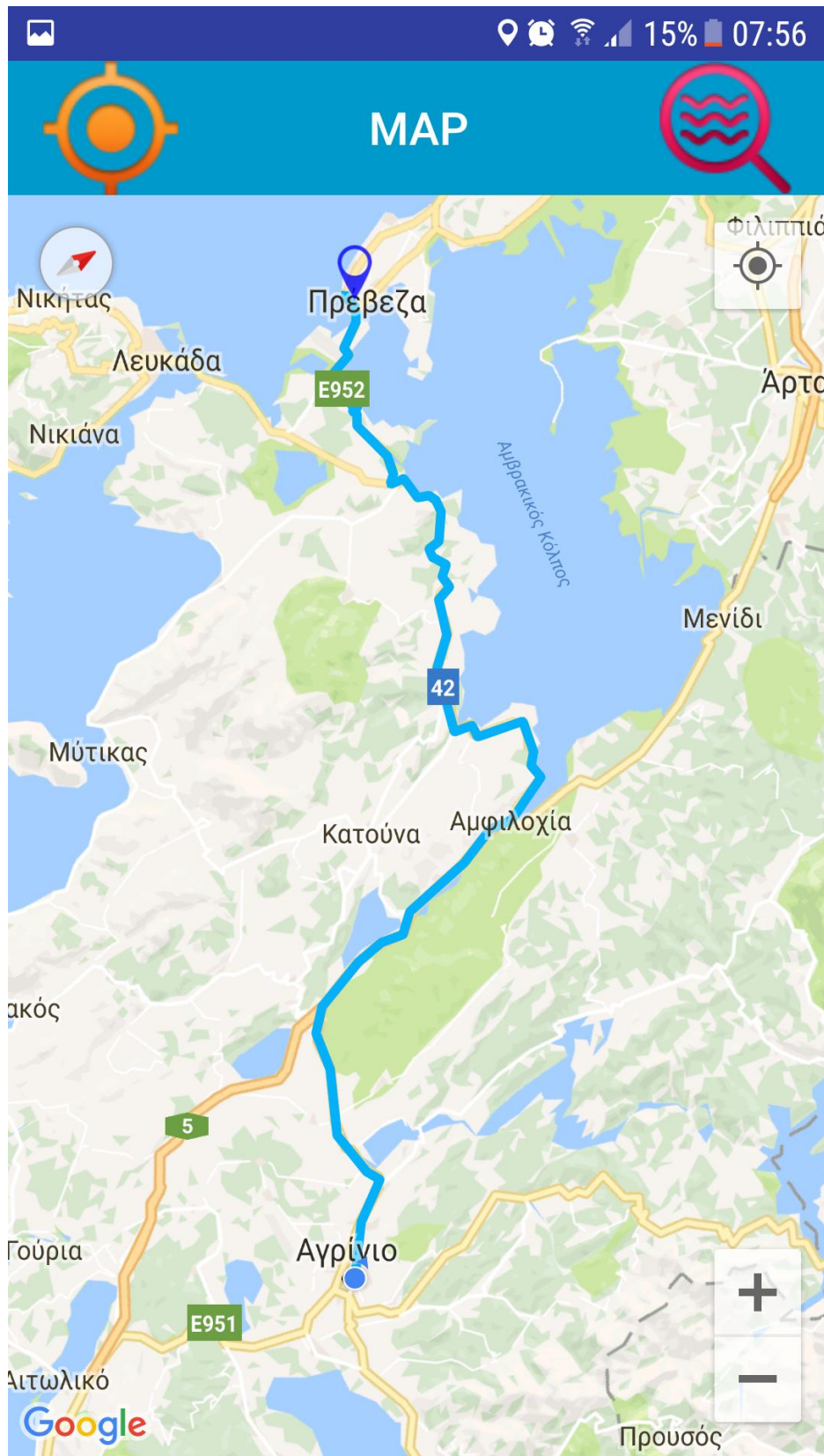
Μόλις πατήσουμε το gps button θα δώσει η εφαρμογή εντολή στη συσκευή να εντοπίσει τη θέση που βρισκόμαστε. Για να γίνει σωστός εντοπισμός πρέπει να είμαστε συνδεδεμένοι με το διαδίκτυο. Στη συγκεκριμένη περίπτωση η συσκευή μας εντοπίζει στην πόλη του Αγρινίου.



Αφού έχουμε εντοπίσει την τοποθεσία μας, μπορούμε να επιλέξουμε μια παραλία από τη λίστα (πάνω δεξιά κόκκινο κουμπί).



Επιλέγοντας την παραλία της Πρεβέζης, σχηματίζεται η βέλτιστη διαδρομή που πρέπει να ακολουθήσουμε, σύμφωνα με τους χάρτες της Google.



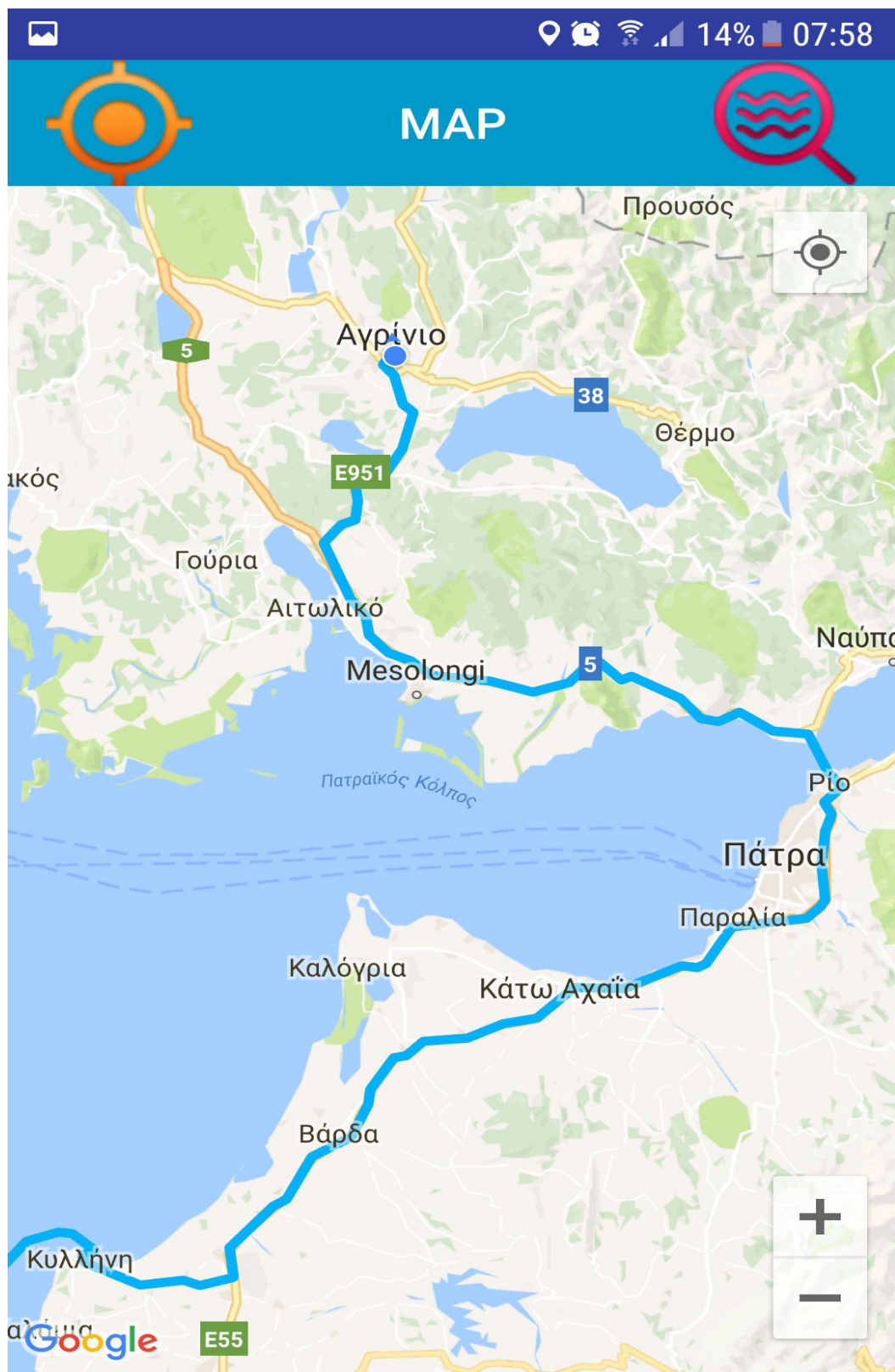
Εναλλακτικά μπορούμε να επιλέξουμε να ορίσουμε μια απόσταση έτσι ώστε να βρούμε τις πιο κοντινές παραλίες. Αυτό μπορεί να γίνει επιλέγοντας το πορτοκαλί gpsbutton(πάνω αριστερά).

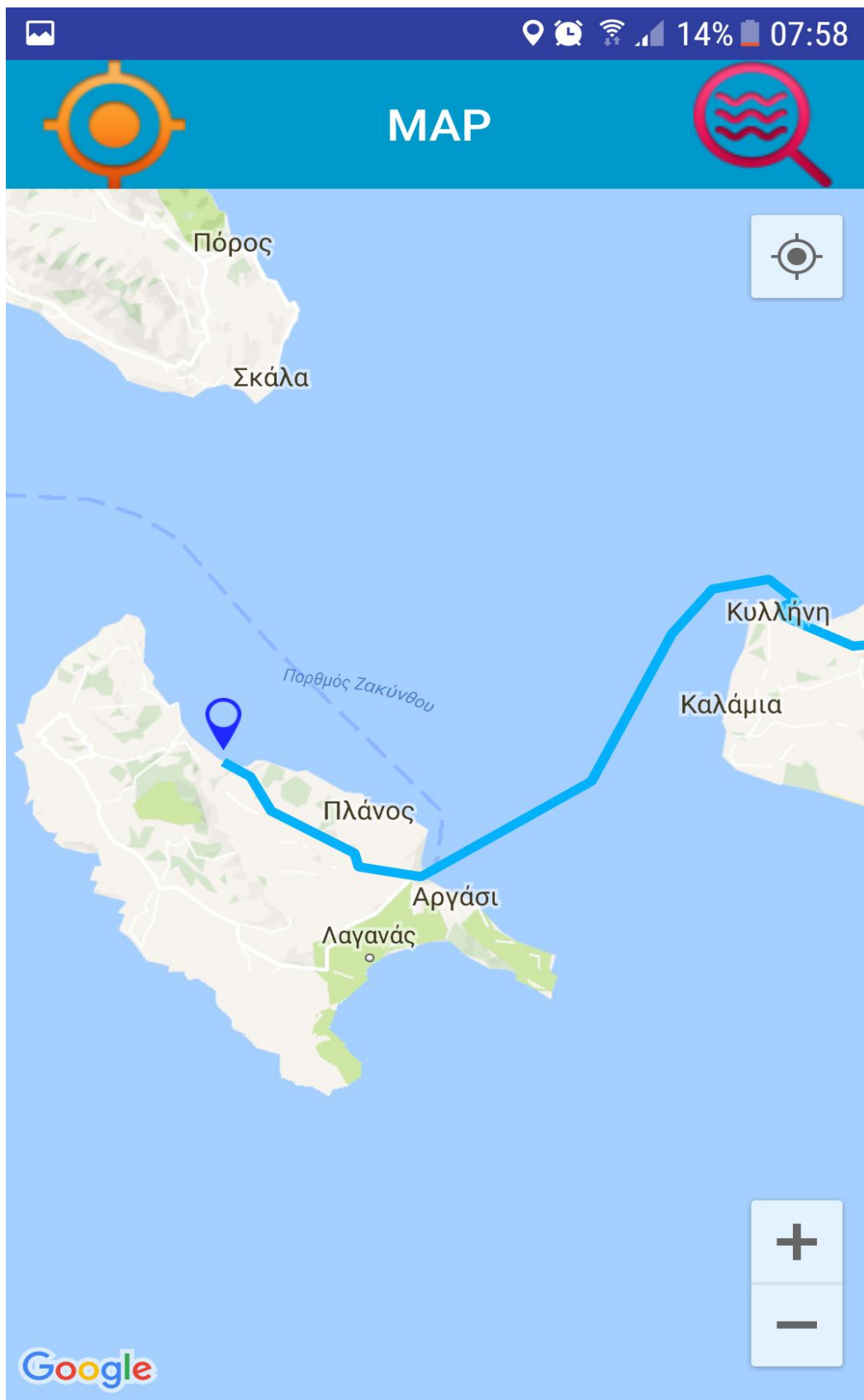


province:	KEFALLINIA
water name:	I wonder
Dhmos:	KEFALLINIA-ELIOS
descript:	EXTREME LEFT COAST
longitude:	20.7717
latitude:	38.1633

province:	ZAKYNTHOS
water name:	Alikes
Dhmos:	Zakynthos-ALYKES
descript:	LEFT HOTEL MONTREAL

Όπως και προηγουμένως αν επιλέξουμε μια από τις παραλίες που έχουν εμφανισθεί, θα σχηματιστεί η διαδρομή. Επιλέγουμε την παραλία της Ζακύνθου.





Κεφάλαιο 4

Συμπεράσματα και μελλοντικές επεκτάσεις

4.1 Συμπεράσματα

Η τεχνολογία καθημερινά εξελίσσεται και δίνει τη δυνατότητα να υλοποιούνται εφαρμογές ώστε να εξυπηρετούν τις αυξημένες ανάγκες που δημιουργούνται. Τα smartphones έχουν κατακλίσει την καθημερινότητα μας και είναι πλέον όλο και πιο απαραίτητα σε κάθε τομέα. Μας προσφέρουν τόσες πολλές υπηρεσίες και μας διευκολύνουν τόσο που δε θα μπορούσε να το διανοηθεί κάποιος στο παρελθόν. Στην εργασία που υλοποιήσαμε κατασκευάσαμε μια εφαρμογή Android για smartphone, η οποία θα εκμεταλλεύεται το gps και τη δυνατότητα σύνδεσης στο internet και με τη βοήθεια της εφαρμογής χάρτες της google θα μπορεί να κατευθύνει τον χρήστη στις περισσότερες παραλίες του ιονίου, σχεδιάζοντάς του τη γρηγορότερη διαδρομή. Η εφαρμογή δίνει στο χρήστη την επιλογή να διαλέξει μόνος του μια από τις παραλίες ή να δώσει μια ακτίνα χιλιομέτρων ώστε να του προτείνει η εφαρμογή τις κοντινότερες παραλίες για να επιλέξει.

Για τη χρήση της εφαρμογής είναι απαραίτητο ένα smartphone με λογισμικό android 3.x.x ή νεότερο, να έχει ενσωματωμένο gps και πρόσβαση στο internet.

4.2 Δυνατότητες επέκτασης

Εκτός από την απόκτηση των γνώσεων για τη δημιουργία παρόμοιων εφαρμογών, η δική μας εφαρμογή έχει τη δυνατότητα να επεκταθεί και να παρέχει και άλλες δυνατότητες στους χρήστες της.

Η συγκεκριμένη εφαρμογή έχει υλοποιηθεί για λειτουργικό android. Θα μπορούσε να υλοποιηθεί και για άλλα λειτουργικά, όπως είναι το ios της Apple ή και για windows phones.

Επίσης θα μπορούσαμε να παρέχουμε στο χρήστη τη δυνατότητα αποθήκευσης ιστορικού ώστε να μπορεί να «κρατάει» τις πιο ενδιαφέρουσες παραλίες.

Ακόμη θα μπορούσε να επεκταθεί πέρα από το ιόνιο και ίσως στο μέλλον να ενσωματώσει όλες τις παραλίες στην Ελλάδα.

Κλείνοντας θα μπορούσαμε να προσθέσουμε ακόμη κάποιες λειτουργίες στην εφαρμογή μας όπως είναι η λειτουργία βαθμολόγησης ή ακόμη και η προσθήκη εικόνων, ώστε να γίνετε πιο εύκολα η επιλογή ανάλογα με τις επιθυμίες του χρήστη.

Βιβλιογραφία

- [1] **Wikipedia Smartphones:**
[Http://en.wikipedia.org/wiki/Smartphone](http://en.wikipedia.org/wiki/Smartphone)
- [2] **Wikipedia Geolocation**
<http://en.wikipedia.org/wiki/Geolocation>
- [3] **Wikipedia Android Studio**
http://en.wikipedia.org/wiki/Android_Studio
- [4] **Generally Android**
<http://www.android.com>
- [5] **Lauren Darcey, Shane Conder, <<Developing apps with android,2nd editon>>**
Εκδόσεις Γκιούρδας, (2011)
- [6] **Walter Savitch <<Absolute Java>> Εκδόσεις Ίων, (2008)**
- [7] **Android Tutorials:**
<https://www.youtube.com/user/thenewboston>
- [8] **Android Architecture – The Key Concepts of Android**
<http://www.android-app-market.com/android-architecture>
- [9] **Get driving directions using Google Maps API**
<http://stackoverflow.com/questions/14495030/get-driving-directions-using-google-maps-api-v2>