



Πανεπιστήμιο Θεσσαλίας

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

Διπλωματική εργασία:

Ανάπτυξη πλατφόρμας για την αναφορά προβλημάτων πόλης.

Development of platform for urban issues reporting.

Μαγιώνας Σωτήρης

Επιβλέπουσα καθηγήτρια : Δασκαλοπούλου Ασπασία

Επίκουρη καθηγήτρια Π.Θ.

Δεύτερο μέλος επιτροπής : Σταμούλης Γεώργιος

Καθηγητής Π.Θ.

Βόλος, 2016

Η σελίδα αυτή είναι σκόπιμα λευκή.

Ευχαριστίες

Με την ολοκλήρωση της παρούσας διπλωματικής εργασίας, θα ήθελα να ευχαριστήσω την κ. Δασκαλοπούλου Ασπασία για την εμπιστοσύνη που μου έδειξε καθώς και για τις πολύτιμες συμβουλές και ιδέες που απλόχερα μου παρείχε.

Ακόμη, θα ήθελα να ευχαριστήσω έναν προς έναν όλους τους καθηγητές με τους οποίους είχα την τιμή να συνεργαστώ αυτά τα χρόνια, για τις γνώσεις που προσπάθησαν να μου μεταλαμπαδεύουν και κυρίως γιατί μου εμφύσησαν μια μέθοδο να αντιλαμβάνομαι, να προσεγγίζω και τελικά να επιλύω προβλήματα σχετιζόμενα τόσο με το αντικείμενο της Επιστήμης των Ηλεκτρονικών Υπολογιστών όσο και με την πεζή, πολλές φορές, καθημερινότητα.

Τέλος, θα ήθελα να ευχαριστήσω τους φίλους μου που στάθηκαν δίπλα μου όλα αυτά τα χρόνια και κυρίως την οικογένεια μου, χωρίς την πολύπλευρη στήριξη της οποίας, τίποτε δεν θα ήταν εφικτό.

Περίληψη

Η παρούσα διπλωματική εργασία με τίτλο «Ανάπτυξη πλατφόρμας για την αναφορά προβλημάτων πόλης» έχει ως στόχο την ανάπτυξη μια πλατφόρμας μέσω της οποίας οι Δημότες ενός Δήμου και ο Δήμος συνεργάζονται ώστε να κάνουν την επίλυση προβλημάτων που εμφανίζονται εντός των γεωγραφικών ορίων του Δήμου ευκολότερη και πιο αποτελεσματική.

Χωρίζεται σε δύο συστατικά στοιχεία. Στην Android εφαρμογή που οι Δημότες μπορούν να κατεβάσουν και να εγκαταστήσουν στην συσκευή τους και στην οποία δόθηκε και το περισσότερο βάρος ,καθώς και σε ένα web-app που είναι εγκατεστημένο στον server του Δήμου.

Στην παρούσα φάση κανένας Δήμος δεν χρησιμοποιεί την εφαρμογή, αλλά για τις ανάγκες της παρουσίασης υποθέτουμε ότι αυτό το πράττει ο Δήμος Βόλου. Η ενσωμάτωση νέων Δήμων στην εφαρμογή είναι αρκετά εύκολη και περιγράφεται αναλυτικότερα σε επόμενο κεφάλαιο.

Φιλοδοξία της εργασίας είναι το «CityFix» να αποτελέσει το «de-facto» εργαλείο που κάθε Δήμος και δημότης θα χρησιμοποιεί για την αναφορά και την επίλυση προβλημάτων που τον απασχολούν.

Abstract

The thesis entitled “Development of platform for urban issues reporting” aims to develop a platform that allows Municipalities and burghers to cooperate, in order to make urban issues reporting and fixing, easier and more efficient than today.

It is the sum of two parts. An Android application that burghers can download and install on their device, and a Web application resting on the Municipality server. The real focus of this thesis however, has been the Android application.

Currently, no Municipality uses the application, but for the purpose of this thesis we will assume that the Municipality of Volos does.

The ultimate goal of this thesis is to make “CityFix” the de-facto standard tool Municipalities and burghers would use to report and fix issues that they find troubling.

Πίνακας περιεχομένων

Εισαγωγή	10
Το πρόβλημα	10
Τι είναι το CityFix	12
Υπόβαθρο	14
Android	14
Εκδόσεις του Android	15
Συνοπτικά για την ανάπτυξη Android εφαρμογών	17
Τεχνολογίες	20
MySQL	20
phpMyAdmin	20
Apache HTTP Server	21
Apache Tomcat	22
PHP	22
Java	23
XML	26
Εργαλεία που χρησιμοποιήθηκαν	27
Android Studio	27
NetBeans	28
Υπόλοιπα	28
Συσκευές που χρησιμοποιήθηκαν για τον έλεγχο της εφαρμογής	30
Παρουσίαση εφαρμογής	32
Κύρια δραστηριότητα	34
Δραστηριότητα τοποθεσίας	35
Σενάριο 1: Επιτυχής εύρεση χρήστη	36
Υποσενάριο 1 : Ο Δήμος υποστηρίζει την εφαρμογή	36
Υποσενάριο 2 : Ο Δήμος δεν υποστηρίζει την εφαρμογή	36
Σενάριο 2 : Ανεπιτυχής εύρεση χρήστη	36

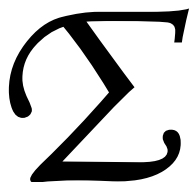
Δραστηριότητα πληροφοριών τοποθεσίας.....	37
Δραστηριότητα κατηγορίας προβλήματος.....	38
Δραστηριότητα πληροφοριών προβλήματος.....	40
Δραστηριότητα πολυμέσων.....	41
Αρχεία κώδικα.....	43
Για την εφαρμογή.....	43
Configuration files.....	43
AndroidManifest.xml.....	43
build.gradle (Module: app).....	44
Main activity.....	45
Activity_main.xml.....	45
MainActivity.java.....	46
Location Activity.....	47
Activity_location.xml.....	47
LocationActivity.java.....	49
LocationDialogFragment_CityNotSupported.java.....	53
LocationDialogFragment_CitySupported.java.....	54
LocationDialogFragment_LocationNull.java.....	55
Location Info Activity.....	56
Activity_location_info.xml.....	56
LocationInfoActivity.java.....	58
Problem Category Activity.....	59
Activity_problem_category.xml.....	59
ProblemCategoryActivity.java.....	61
StableArrayAdapter.java.....	64
Problem Info Activity.....	65
Activity_problem_info.xml.....	65
ProblemInfoActivity.java.....	68

Multimedia Activity.....	70
Activity_multimedia.xml.....	70
MultimediaActivity.java.....	74
Utility.java.....	81
Upload_complete_dialog.java.....	82
Resources.....	83
anim_alpha.xml.....	83
Colors.xml.....	83
Dimens.xml.....	83
Dimens.xml (w820dp).....	84
Styles.xml.....	84
Cities.....	85
Strings.xml.....	86
Strings.xml (el).....	88
Για το WEB-App στον Server.....	90
Στον Apache.....	90
Upload.php.....	90
Στον Tomcat.....	92
Results.jsp.....	92
Index.jsp.....	94
Συμπεράσματα και μελλοντικές επεκτάσεις.....	99
Δύο σημεία με ενδιαφέρον.....	99
Τοποθεσία.....	99
HTTP μηνύματα.....	100
Μελλοντικές επεκτάσεις.....	101
Κατηγορίες προβλημάτων.....	101
Ενσωμάτωση νέων Δήμων.....	102
Υποστήριξη βίντεο.....	103

Ενσωμάτωση ΔΕΚΟ.....	103
Δημιουργία έκδοσης του CityFix για τους εργαζομένους του Δήμου.....	103
Δυνατότητα δια-Δημοτικής επικοινωνίας και επαναπρώθησης αναφοράς.	104
Ασφάλεια.....	105
Βιβλιογραφία.....	106

Εισαγωγή

Το πρόβλημα



ε έναν Δήμο μιας σύγχρονης μεγαλούπολης που μπορεί να αριθμεί δεκάδες ή και εκατοντάδες χιλιάδες Δημότες και να εκτείνεται σε μια περιοχή δεκάδων τετραγωνικών χιλιομέτρων, είναι λογικό καθημερινά να εκδηλώνονται μια σειρά από προβλήματα.

Αυτά μπορεί να είναι πολύ σημαντικά και επικίνδυνα, όπως η απορρύθμιση των φωτεινών σηματοδοτών που ρυθμίζουν την κυκλοφορία στους δρόμους, λιγότερο σοβαρά, όπως ένας χαλασμένος κάδος απορριμμάτων, ή και φαινομενικά ασήμαντα όπως ένα δέντρο που έχει μεγαλώσει πολύ και δυσχεραίνει την διέλευση των πεζών από το πεζοδρόμιο. Ακόμη όμως κι αυτά τα εκ πρώτης όψεως ασήμαντα προβλήματα, αν αφεθούν ανεπίλυτα, μπορούν σταδιακά να οδηγήσουν σε δυσάρεστες καταστάσεις ή ακόμη και σε ατυχήματα, όπως για παράδειγμα στην περίπτωση που ένας πεζός για να αποφύγει το πληθωρικό δέντρο αφήνει το πεζοδρόμιο και εισέρχεται απρόσεκτα στον δρόμο όπου κινούνται αυτοκίνητα με μεγάλες ταχύτητες.

Πολλά από αυτά τα προβλήματα, και κυρίως αυτά που εντάσσονται στις δύο τελευταίες κατηγορίες, έχουν την τάση να μένουν ανεπίλυτα για μεγάλο χρονικό διάστημα. Αυτό οφείλεται σε δύο κυρίως παράγοντες.

Πρώτον, ο εκάστοτε Δήμος συνηθίζει να χρονοπρογραμματίζει κάποιες περιοδικές εργασίες σε συγκεκριμένες χρονικές στιγμές κάθε χρόνο. Έτσι, το κλάδεμα των δέντρων γίνεται μία ή δύο φορές τον χρόνο και κάποιο δέντρο που παραβλέφθηκε ή μεγάλωσε με ταχύτερους ρυθμούς σε σχέση με άλλα, θα ξανακλαδευτεί την επόμενη φορά που τα συνεργεία του Δήμου θα αναλάβουν να καλλωπίσουν το σύνολο των δέντρων.

Ο δεύτερος λόγος είναι πρακτικός και έχει να κάνει με τις περιορισμένες προσλαμβάνουσες του Δήμου. Οι εργαζόμενοι καθώς και οι υπηρεσίες του Δήμου, όσο ζήλο και αίσθηση καθήκοντος κι αν επιδεικνύουν, δεν είναι δυνατόν να είναι ανά πάσα στιγμή ενημερωμένοι για το σύνολο των προβλημάτων

που έχουν παρουσιασθεί. Πολύ απλά, τα προβλήματα εμφανίζονται με πολύ μεγαλύτερο ρυθμό από ότι γίνονται αντιληπτά.

Ίσως σε μια κοινωνία του όχι και τόσο απώτερου μέλλοντος, κάμερες και αισθητήρες τοποθετημένοι σε όλα τα μήκη και πλάτη ενός Δήμου, σε συνεργασία με εκλεπτυσμένο λογισμικό να προχωρούν σε ταυτοποίηση και αναφορά των προβλημάτων σε πραγματικό χρόνο, δίνοντας την δυνατότητα στα συνεργεία να διατηρούν έναν Δήμο ελεύθερο από προβλήματα.

Όμως ,δεν είμαστε ακόμη εκεί...

Τι είναι το CityFix

Μπορεί η καταγραφή των προβλημάτων από αισθητήρες σε πραγματικό χρόνο να φαντάζει σε κάποιους ως σενάριο επιστημονικής φαντασίας, όμως σήμερα, την ώρα που γράφονται αυτές οι γραμμές, μέσα στα γεωγραφικά όρια οποιουδήποτε Δήμου της χώρας μας υπάρχουν δεκάδες ή και εκατοντάδες χιλιάδες αισθητήρες. Αισθητήρες ικανοί να τραβήξουν μια φωτογραφία ή ένα βίντεο κλιπ, εξοπλισμένοι με την δυνατότητα πρόσβασης στο Διαδίκτυο και ελεύθεροι να κινούνται απρόσκοπτα σε κάθε τοποθεσία της αρεσκείας τους. Ο λόγος φυσικά για τις έξυπνες συσκευές ή αλλιώς «Smartphones», «Tablets» και τα λοιπά.

Αν λοιπόν, αφού εντοπίσουμε το πρόβλημα με τις αισθήσεις μας, χρησιμοποιήσουμε τις έξυπνες συσκευές για την αναφορά του στον τοπικό μας Δήμο, φέρνουμε αυτό το σενάριο επιστημονικής φαντασίας ένα βήμα πιο κοντά και σε κάθε περίπτωση αυξάνουμε τρομακτικά τις προσλαμβάνουσες του Δήμου και του δίνουμε την δυνατότητα να επιτελέσει καλύτερα το έργο του προς όφελος όλων μας.

Το **CityFix**, που είναι και το αντικείμενο της παρούσας διπλωματικής εργασίας, φιλοδοξεί να κάνει ακριβώς αυτό. Να παρέχει μια πλατφόρμα επικοινωνίας ανάμεσα στον Δήμο και τους Δημότες του για την αναφορά διαφόρων προβλημάτων δίνοντας έμφαση στην πλευρά του Δημότη ή αλλιώς, στην Android εφαρμογή.

Χωρίζεται σε δύο συστατικά στοιχεία. Στην Android εφαρμογή που οι Δημότες μπορούν να κατεβάσουν και να εγκαταστήσουν στην συσκευή τους και σε ένα web-app που είναι εγκατεστημένο στον server του Δήμου.

Το web-app είναι γραμμένο σε java και php, ακολουθεί την MVC φιλοσοφία, τρέχει στον «apache Tomcat» (αν και εναλλακτικά θα μπορούσε να τρέχει σε μια πληθώρα web servers) και αναπτύχθηκε ως επί των πλείστων στο ολοκληρωμένο προγραμματιστικό περιβάλλον «NetBeans». Είναι υπεύθυνο για την παραλαβή των αναφορών των Δημοτών και την τοποθέτησή τους σε μια βάση δεδομένων αλλά και για την παρουσίασή τους με την μορφή πίνακα σε φυλλομετρητή της αρεσκείας του εργαζόμενου στον Δήμο που είναι επιφορτισμένος με αυτήν την φροντίδα.

Η Android εφαρμογή CitiFix αναπτύχθηκε στο επίσημο εργαλείο ανάπτυξης Android εφαρμογών «Android Studio». Οι Δημότες θα μπορούν δυνητικά να την βρουν και να την κατεβάσουν από το Google Play Store. Ανάλογα με την ρύθμιση γλώσσας της συσκευής τους, μπορούν να χρησιμοποιήσουν την εφαρμογή είτε στην Ελληνική είτε στην Αγγλική γλώσσα. Η εφαρμογή δύναται να τρέξει σε ένα ευρύ φάσμα συσκευών τόσο ως προς το μέγεθος της οθόνης όσο και ως προς την έκδοση του λειτουργικού

συστήματος της συσκευής. Ειδικά όσον αφορά την έκδοση του λειτουργικού, η εφαρμογή μπορεί να εγκατασταθεί και να λειτουργήσει σε συσκευές που «τρέχουν» από την έκδοση 4.1 “Jelly Bean” μέχρι την 6.0.1 «Marshmallow», εύρος που καλύπτει περίπου το 95,2% του συνόλου των χρηστών του λειτουργικού Android σύμφωνα με στοιχεία της Google τον μήνα Αύγουστο του 2016.

Η εφαρμογή, μέσα από ένα απλό, ευνόητο και εύχρηστο user interface καθοδηγεί τον χρήστη στα απαιτούμενα βήματα για την αναφορά ενός προβλήματος. Αυτά συνίστανται στην παροχή της θέσης και μίας φωτογραφίας του προβλήματος, καθώς επίσης στην επιλογή της κατηγορίας στην οποία εντάσσεται το πρόβλημα και στην παροχή επιπλέον πληροφοριών υπό την μορφή ελεύθερου κειμένου εάν ο χρήστης το κρίνει απαραίτητο. Με την ολοκλήρωση της διαδικασίας, η αναφορά αποστέλλεται στο web-app υπό την μορφή HTTP μηνύματος και εάν δεν παρουσιαστεί κάποιο πρόβλημα, η εφαρμογή περιμένει την HTTP απάντηση του server, ενημερώνει τον χρήστη για την επιτυχή έκβαση της διαδικασίας αναφοράς, τον ευχαριστεί και τον προτρέπει να κλείσει την εφαρμογή.

Το CityFix θα μπορούσε να είχε υλοποιηθεί ως εφαρμογή για οποιαδήποτε από τις διαδεδομένες πλατφόρμες πέρα από αυτήν του Android (iOS, Windows phone/mobile). Ο λόγος που επιλέχθηκε να αναπτυχθεί για συσκευές Android είναι η πρωτοκαθεδρία του λειτουργικού στην αγορά έξυπνων συσκευών πράγμα που σημαίνει πως σε κάθε Δήμο οι συσκευές Android είναι πολλαπλάσιες των υπολοίπων. Αφού απώτερος σκοπός είναι η όσο το δυνατόν μεγαλύτερη αύξηση των προσλαμβανουσών του Δήμου το γεγονός αυτό αξιολογήθηκε ως καθοριστικός παράγοντας.

Στις επόμενες ενότητες γίνεται μια παρουσίαση των τεχνολογιών και των εργαλείων που χρησιμοποιήθηκαν για την περάτωση της ανάπτυξης του CityFix.

Υπόβαθρο

Android

ΑΝΔΡΟΙΔ

Το **Android** είναι [λειτουργικό σύστημα](#) για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού [Linux](#). Αρχικά αναπτύχθηκε από την Android, Inc. η οποία εξαγοράστηκε από την [Google](#) το 2005. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού [Java](#), ελέγχοντας την συσκευή μέσω [βιβλιοθηκών](#) λογισμικού ανεπτυγμένων από την Google.

Το Android είναι κατά κύριο λόγο σχεδιασμένο για συσκευές με [οθόνη αφής](#), όπως τα [έξυπνα τηλέφωνα](#) και οι [ταμπλέτες](#), με διαφορετικό περιβάλλον χρήσης για τηλεοράσεις ([Android TV](#)), αυτοκίνητα ([Android Auto](#)) και ρολόγια χειρός ([Android Wear](#)). Παρόλο που έχει αναπτυχθεί για συσκευές με οθόνη αφής, έχει χρησιμοποιηθεί σε κονσόλες παιχνιδιών, ψηφιακές φωτογραφικές μηχανές, συνηθισμένους Η/Υ (π.χ. το HP Slate 21) και σε άλλες ηλεκτρονικές συσκευές.

Το Android είναι το πιο ευρέως διαδεδομένο λογισμικό στον κόσμο. Οι συσκευές με Android έχουν περισσότερες πωλήσεις από όλες τις συσκευές [Windows](#), [iOS](#) και [Mac OS X](#) μαζί.

Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας κοινοπραξίας 48 τηλεπικοινωνιακών εταιριών, εταιριών λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων στις συσκευές κινητής τηλεφωνίας. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της [Apache License](#), μιας ελεύθερης άδειας λογισμικού. Το λογότυπο για το λειτουργικό σύστημα Android είναι ένα ρομπότ σε χρώμα πράσινου μήλου και σχεδιάστηκε από τη γραφίστρια Ιρίνα Μπλόκ.^[1]



Εκδόσεις του Android

Η ιστορία εκδόσεων του Android του λειτουργικού συστήματος των κινητών ξεκίνησε με την κυκλοφορία του Android beta το Νοέμβριο του 2007. Η πρώτη εμπορική έκδοση ήταν το Android 1.0 που κυκλοφόρησε το Σεπτέμβριο του 2008 και έχουν γίνει μια σειρά από ενημερώσεις στην λειτουργία του συστήματος από την αρχική κυκλοφορία του.

Από τον Απρίλιο του 2009, οι εκδόσεις του Android έχουν θέμα από την ζαχαροπλαστική στην κωδική ονομασία τους, και κυκλοφόρησαν σε αλφαβητική σειρά, εξαιρουμένων των εκδόσεων 1.0 και 1.1, που δεν τέθηκαν υπό συγκεκριμένα κωδικά ονόματα: ^[2]

- Apple Pie (1.0) , September 23, 2008



ANDROID 1.0
APPLE PIE

- Banana Bread (1.1), February 9, 2009



ANDROID 1.1
BANANA BREAD

- Cupcake (1.5) , April 27, 2009



Cupcake
Android 1.5

- Donut (1.6) , September 15, 2009



Donut
Android 1.6

- Eclair (2.0-2.1) , October 26, 2009



Eclair
Android 2.0/2.1

- Froyo (2.2-2.2.3) , May 20, 2010



Froyo
Android 2.2/2.2.3

- Gingerbread (2.3-2.3.7) , December 6, 2010



- Honeycomb (3.0-3.2.6) , February 22, 2011



- Ice Cream Sandwich (4.0-4.0.4) , October 18, 2011



- Jelly Bean (4.1-4.3.1) , July 9, 2012



- KitKat (4.4-4.4.4) , October 31, 2013



- Lollipop (5.0-5.0.2) , November 12, 2014



- Marshmallow (6.0) , October 5, 2015



Την ώρα που γράφονται αυτές οι γραμμές βρισκόμαστε εν αναμονή της νέας μεγάλης αναβάθμισης με κωδική ονομασία [Android 7.0 Nougat](#). Η έκδοση για προγραμματιστές κυκλοφόρησε στις 9 Μαρτίου 2016 ενώ η τελική κυκλοφορία αναμένεται στα τέλη Αυγούστου ή στις αρχές Σεπτεμβρίου 2016.

Συνοπτικά για την ανάπτυξη Android εφαρμογών.

Σε αυτήν την παράγραφο σκιαγραφούνται τα βασικά σημεία της ανάπτυξης εφαρμογών για Android χωρίς να μπαίνουμε σε πολύ μεγάλη λεπτομέρεια, καθώς επίσης επισημαίνονται οι κυριότερες διαφορές ανάμεσα στην ανάπτυξη Android εφαρμογών και στην ανάπτυξη «παραδοσιακών» Java εφαρμογών.

Προγραμματιστές εξοικειωμένοι με την ανάπτυξη εφαρμογών σε Java θα βρουν την μετάβαση στην ανάπτυξη Android εφαρμογών σχετικά εύκολη καθώς η φιλοσοφία είναι αντίστοιχη. Συγκεκριμένα, όπως στην Java τα πηγαία αρχεία κώδικα μαζί με όποια άλλα resources μεταγλωττίζονται όχι σε τελικό εκτελέσιμο αλλά σε έναν ενδιάμεσο κώδικα ο οποίος μετά τρέχει στην εικονική μηχανή της Java προσφέροντας έτσι μεταφερσιμότητα και ασφάλεια, έτσι και στο Android η διαδικασία είναι σχεδόν ταυτόσημη με κάποιες όμως διαφορές.

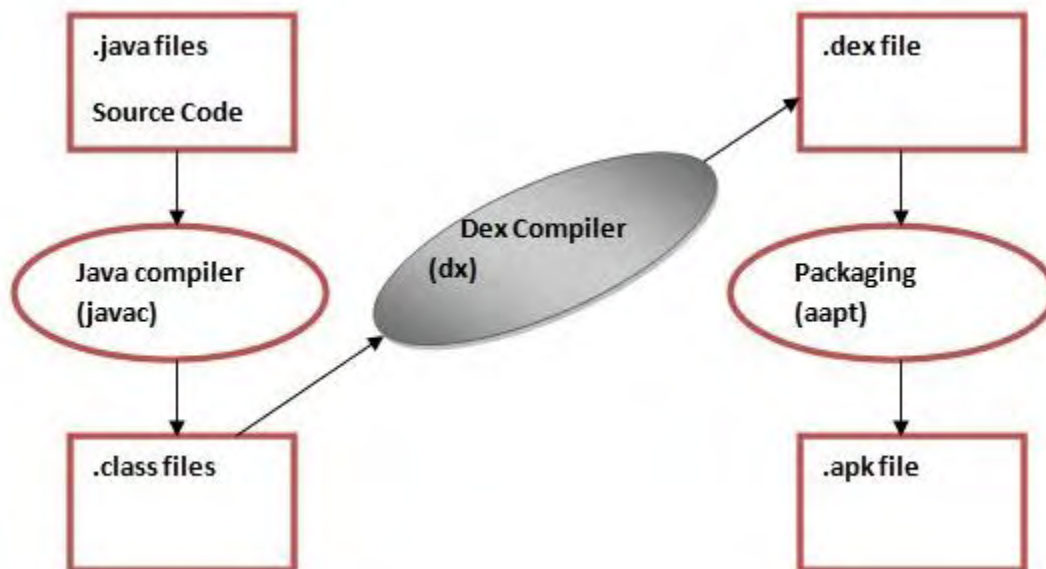
Το Android χρησιμοποιεί μόνο τη σύνταξη και τη σημασιολογία της γλώσσας Java, και δεν παρέχει τις πλήρεις βιβλιοθήκες κλάσεων και API που περιλαμβάνονται στο Java SE ή ME. Κάποιες βιβλιοθήκες έχουν αντικατασταθεί από άλλες, καταλληλότερες για το περιβάλλον του Android, όπως οι [AWT](#) και [Swing](#) για την παραγωγή διεπαφών γραφικού περιβάλλοντος που δεν συμπεριλαμβάνονται στο SDK του Android. Άλλες δεν έχουν συμπεριληφθεί καθόλου γιατί στερούνται νοήματος στην ανάπτυξη Android εφαρμογών όπως για παράδειγμα η «`javax.print`» καθώς η ενσύρματη σύνδεση κινητού και εκτυπωτή είναι εξαιρετικά σπάνιο φαινόμενο. ^[3]

Μια άλλη, σημαντική διαφορά ανάμεσα στα δύο, έχει να κάνει με σχεδιαστικές επιλογές του προγραμματιστή, κατανόηση του πως οι χρήστες χρησιμοποιούν ένα κινητό και πρόβλεψη πιθανών τεχνολογικών ή φυσικών περιορισμών. Για παράδειγμα, ένας προγραμματιστής Android εφαρμογών με συνδεσιμότητα στο Ίντερνετ πρέπει να σκεφτεί τι θα συμβεί στην εφαρμογή του στην περίπτωση που κάποια στιγμή η σύνδεση στο Ίντερνετ είναι κακή ή και εντελώς απύουσα, κάτι που σπάνια χρειάζεται να κάνει ένας προγραμματιστής Java εφαρμογών που τρέχουν σε κάποιον προσωπικό υπολογιστή. Σημαντική είναι ακόμη η απαίτηση οι εφαρμογές να είναι όσο το δυνατόν μικρότερες σε μέγεθος καθώς ο αποθηκευτικός χώρος των κινητών συσκευών είναι ακόμη και σήμερα περιορισμένος.

Η παραδοσιακή java χρησιμοποιεί JVM (Εικονική μηχανή java), ενώ το Android χρησιμοποιεί είτε την εικονική μηχανή [Dalvik Virtual Machine](#) (DVM) είτε το [Android Runtime](#) (ART) το οποίο αντικατέστησε το Dalvik αρχής γενομένης από την έκδοση [Android 5.0 Lollipop](#). ^[4]

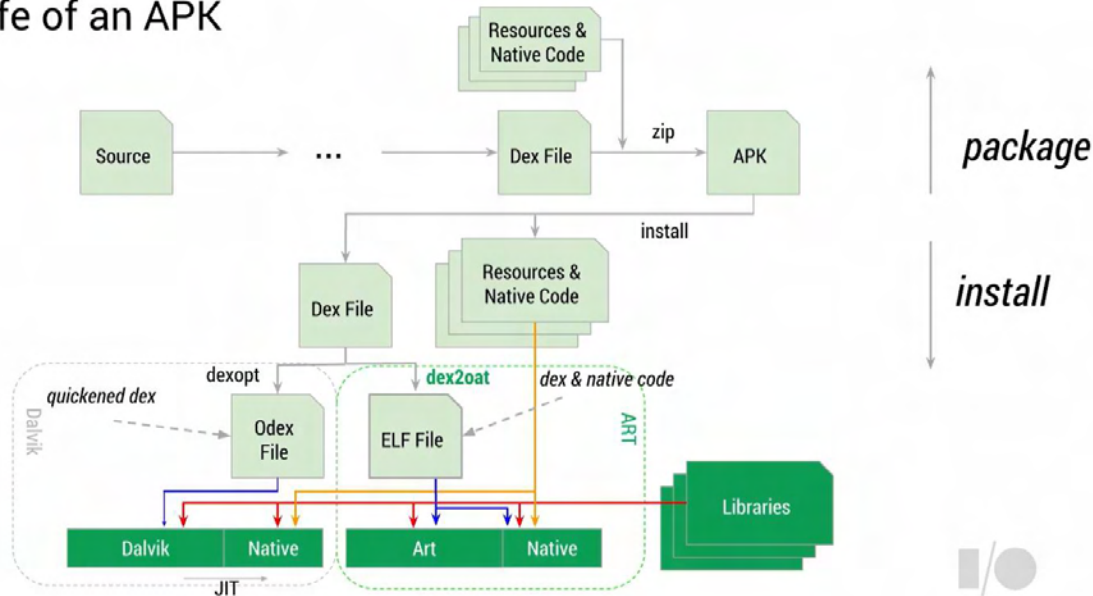
Το Dalvik είναι μία εικονική μηχανή android που έχει βελτιστοποιηθεί για συσκευές κινητής τηλεφωνίας. Ενώ η αρχιτεκτονική της JVM είναι βασισμένη στη στοίβα, αυτή της DVM βασίζεται στους καταχωρητές. Η JVM χρησιμοποιεί java byte code και τρέχει τα .class αρχεία ενώ η DVM χρησιμοποιεί το δικό της byte code της και τρέχει τα αρχεία .Dex. Η DVM έχει σχεδιαστεί ώστε μια συσκευή να μπορεί να χρησιμοποιεί αποτελεσματικά πολλά στιγμιότυπα (instances) της εικονικής μηχανής, ένα για κάθε εφαρμογή που τρέχει έτσι στον προσωπικό της χώρο, και έχει βελτιστοποιηθεί ώστε να χρειάζεται ελάχιστη μνήμη, να έχει καλή απόδοση και να επιμηκύνει τη διάρκεια ζωής της μπαταρίας.^{[5] [6] [7]}

Οι παρακάτω εικόνες δείχνουν την διαδικασία που ακολουθείται από την συγγραφή των αρχείων πηγαίου κώδικα (.java files) μέχρι την παραγωγή του τελικού εκτελέσιμου αρχείου .apk καθώς και τον ρόλο των εικονικών μηχανών που τρέχουν στην συσκευή και χρησιμοποιούνται για να τρέξει το πρόγραμμα.



1. <http://www.javatpoint.com/dalvik-virtual-machine>

The life of an APK



2. <http://www.anandtech.com/show/8231/a-closer-look-at-android-runtime-art-in-android-l>

Περισσότερα για την ανάπτυξη εφαρμογών σε περιβάλλον Android, συμπεριλαμβανομένων παραδειγμάτων κώδικα, documentation, συμβουλών για best-practices, video-tutorials καθώς και ολοκληρωμένων step-by-step training courses μπορεί κανείς να βρει στο επίσημο site της Google για την ανάπτυξη εφαρμογών Android ^[8] καθώς και σε sites όπως το [udacity](http://udacity.com) και το [coursera](http://coursera.com).

Τεχνολογίες

MySQL

Η **MySQL** είναι ένα σύστημα διαχείρισης [σχεσιακών βάσεων δεδομένων](#) που μετρά περισσότερες από 11 εκατομμύρια εγκαταστάσεις. Έλαβε το όνομά της από την ένωση δύο συνθετικών, την κόρη του Μόντυ Βιντένιους, τη Μάι (αγγλ. My) και την SQL^[9]. Το πρόγραμμα τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο [βάσεων δεδομένων](#).

Ο κώδικας του εγχειρήματος είναι διαθέσιμος μέσω της [GNU General Public License](#), καθώς και μέσω ορισμένων ιδιόκτητων συμφωνιών όταν πρόκειται για εμπορική χρήση και σε αυτήν την περίπτωση παρέχονται περισσότερες δυνατότητες. Ανήκει και χρηματοδοτείται από μία και μοναδική κερδοσκοπική εταιρία, τη σουηδική MySQL AB, η οποία σήμερα ανήκει στην [Oracle](#).^[10]



phpMyAdmin

Το **phpMyAdmin** είναι ένα ελεύθερο και ανοιχτού κώδικα εργαλείο σχεδιασμένο ώστε να επιτρέπει την διαχείριση της MySQL (και του MariaDB) από γραφικό περιβάλλον μέσω web-browser. Έτσι, η δημιουργία, επεξεργασία και διαγραφή πινάκων ,κελιών και τα λοιπά, καθώς και η εκτέλεση ερωτημάτων προς την βάση και άλλες διαχειριστικές εργασίες γίνονται πολύ εύκολα.^[11]



Apache HTTP Server

Ο **Apache HTTP** γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού (web). Όποτε ένας χρήστης επισκέπτεται ένα ιστότοπο το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου [HTTP](#), ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης.

Ο Apache είναι ένας από τους δημοφιλέστερους εξυπηρετητές ιστού, εν μέρει γιατί λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το UNIX και το Mac OS X. Κυκλοφόρησε υπό την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation).

Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. Oracle, MySQL.

Η πρώτη του έκδοση, γνωστή ως NCSA HTTPd, δημιουργήθηκε από τον Robert McCool και κυκλοφόρησε το 1993. Θεωρείται ότι έπαιξε σημαντικό ρόλο στην αρχική επέκταση του παγκόσμιου ιστού. Ήταν η πρώτη βιώσιμη εναλλακτική επιλογή που παρουσιάστηκε απέναντι στον εξυπηρετητή http της [εταιρείας Netscape](#) και από τότε έχει εξελιχθεί στο σημείο να ανταγωνίζεται άλλους εξυπηρετητές βασισμένους στο Unix σε λειτουργικότητα και απόδοση. Από το 1996 ήταν από τους πιο δημοφιλείς όμως από τον Μάρτιο του 2006 έχει μειωθεί το ποσοστό της εγκατάστασής του κυρίως από τον [Microsoft Internet Information Services](#) και την πλατφόρμα .NET. Τον Οκτώβριο του 2007 το μερίδιο του ήταν 47.73% από όλους τους ιστοτόπους. ^[12]



Apache Tomcat

Ο **Apache Tomcat** ή κοινώς « **Tomcat** » είναι ένας εξυπηρετητής του παγκόσμιου ιστού απαραίτητος για το τρέξιμο Java εφαρμογών και την παραγωγή και εμφάνιση δυναμικών ιστοσελίδων. Αναπτύσσεται και συντηρείται κι αυτός από το Ίδρυμα Λογισμικού Apache και συνήθως συνυπάρχει με τον Apache HTTP Server.^[13]



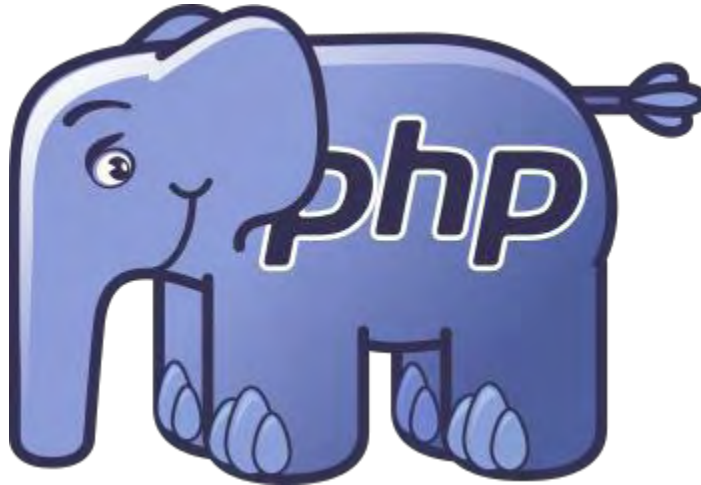
PHP

Η **PHP** (PHP: Hypertext Preprocessor) είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που είτε θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML ή θα επεξεργασθεί τις εισόδους δίχως να προβάλλει την έξοδο στο χρήστη, αλλά θα τις μεταβιβάσει σε κάποιο άλλο PHP script.

Η PHP αποτελεί μια από τις πιο διαδεδομένες τεχνολογίες στο Παγκόσμιο Ιστό, καθώς χρησιμοποιείται από πληθώρα εφαρμογών και ιστοτόπων. Η ευρύτητα στη χρήση της είναι απόρροια της ευκολίας που παρουσιάζει ο προγραμματισμός με αυτή αλλά και στο γεγονός πως είναι μια γλώσσα η οποία βρίσκεται σχεδόν σε κάθε διακομιστή. Διάσημες εφαρμογές που κάνουν εκτενή χρήση της PHP είναι το γνωστό Σύστημα Διαχείρισης Περιεχομένου ([Content Management System](#)) "[WordPress](#)" και το "[Drupal](#)".

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού C ένα απλό script με όνομα rhr.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του

ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0 αριθμώντας περισσότερους από 50.000 ιστότοπους που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP.^[14]



Java

Στις αρχές του 1991, η Sun αναζητούσε το κατάλληλο εργαλείο για να αποτελέσει την πλατφόρμα ανάπτυξης λογισμικού σε μικρο-συσκευές (έξυπνες οικιακές συσκευές έως πολύπλοκα συστήματα παραγωγής γραφικών). Τα εργαλεία της εποχής ήταν γλώσσες όπως η C++ και η C. Μετά από διάφορους πειραματισμούς προέκυψε το συμπέρασμα ότι οι υπάρχουσες γλώσσες δεν μπορούσαν να καλύψουν τις ανάγκες τους. Ο "πατέρας" της Java, James Gosling, που εργαζόταν εκείνη την εποχή για την Sun, έκανε ήδη πειραματισμούς πάνω στη C++ και είχε παρουσιάσει κατά καιρούς κάποιες πειραματικές γλώσσες ως πρότυπα για το νέο εργαλείο που αναζητούσαν στην Sun. Τελικά μετά από λίγο καιρό κατέληξαν με μια πρόταση για το επιτελείο της εταιρίας, η οποία ήταν η γλώσσα Oak. Το όνομά της το πήρε από το ομώνυμο δένδρο (βελανιδιά) το οποίο ο Gosling είχε έξω από το γραφείο του και έβλεπε κάθε μέρα.

Η Oak ήταν μία γλώσσα που διατηρούσε μεγάλη συγγένεια με την C++. Παρόλα αυτά είχε πολύ πιο έντονο αντικειμενοστραφή (*object oriented*) χαρακτήρα σε σχέση με την C++ και χαρακτηριζόταν για την απλότητα της. Σύντομα οι υπεύθυνοι ανάπτυξης της νέας γλώσσας ανακάλυψαν ότι το όνομα Oak ήταν ήδη κατοχυρωμένο οπότε κατά την διάρκεια μιας εκ των πολλών συναντήσεων σε κάποιο τοπικό καφέ αποφάσισαν να μετονομάσουν το νέο τους δημιουργήμα σε Java που εκτός των άλλων ήταν το όνομα της αγαπημένης ποικιλίας καφέ για τους δημιουργούς της. Η επίσημη εμφάνιση της Java αλλά και του HotJava (πλοηγός με υποστήριξη Java) στη βιομηχανία της πληροφορικής έγινε το Μάρτιο του 1995 όταν η Sun την

ανακοίνωσε στο συνέδριο *Sun World* 1995. Ο πρώτος μεταγλωττιστής (*compiler*) της ήταν γραμμένος στη γλώσσα C από τον James Gosling. Το 1994, ο A. Van Hoff ξαναγράφει τον μεταγλωττιστή της γλώσσας σε *Java*, ενώ το Δεκέμβριο του 1995 πρώτες οι IBM, Borland, Mitsubishi Electronics, Sybase και Symantec ανακοινώνουν σχέδια να χρησιμοποιήσουν τη *Java* για την δημιουργία λογισμικού. Από εκεί και πέρα η *Java* ακολουθεί μία ανοδική πορεία και είναι πλέον μία από τις πιο δημοφιλείς γλώσσες στον χώρο της πληροφορικής. Στις 13 Νοεμβρίου του 2006 η *Java* έγινε πλέον μια γλώσσα ανοιχτού κώδικα (GPL) όσον αφορά το μεταγλωττιστή (*javac*) και το πακέτο ανάπτυξης (JDK, *Java Development Kit*).

Στις 27 Απριλίου 2010 η εταιρία λογισμικού Oracle Corporation ανακοίνωσε ότι μετά από πολύμηνες συζητήσεις ήρθε σε συμφωνία για την εξαγορά της Sun Microsystems και των τεχνολογιών (πνευματικά δικαιώματα/ πατέντες) που η δεύτερη είχε στην κατοχή της ή δημιουργήσει. Η συγκεκριμένη συμφωνία θεωρείται σημαντική για το μέλλον της *Java* και του γενικότερου οικοσυστήματος τεχνολογιών γύρω από αυτή μιας και ο έμμεσος έλεγχος της τεχνολογίας και η εξέλιξη της περνάει σε άλλα χέρια.

Ένα από τα βασικά πλεονεκτήματα της *Java* έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. 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. Για να επιτευχθεί όμως αυτό χρειαζόταν κάποιος τρόπος έτσι ώστε τα προγράμματα γραμμένα σε *Java* να μπορούν να είναι «κατανοητά» από κάθε υπολογιστή ανεξάρτητα του είδους επεξεργαστή (Intel x86, IBM, Sun SPARC, Motorola) αλλά και λειτουργικού συστήματος (Windows, Unix, Linux, BSD, MacOS). Ο λόγος είναι ότι κάθε κεντρική μονάδα επεξεργασίας κατανοεί διαφορετικό κώδικα μηχανής. Ο συμβολικός κώδικας (*assembly*) που μεταφράζεται και εκτελείται σε Windows είναι διαφορετικός από αυτόν που μεταφράζεται και εκτελείται σε έναν υπολογιστή Macintosh. Η λύση δόθηκε με την ανάπτυξη της *Εικονικής Μηχανής (Virtual Machine)*.

Αφού γραφεί 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 μεταγλωττίζεται. Όταν πρόκειται να εκτελεστεί η εφαρμογή σε ένα μηχάνημα, το *Java Virtual Machine* που πρέπει να είναι εγκατεστημένο σε αυτό θα αναλάβει να διαβάσει τα αρχεία *.class*. Στη συνέχεια τα μεταφράζει σε γλώσσα μηχανής που να υποστηρίζεται από το λειτουργικό σύστημα και τον επεξεργαστή, έτσι ώστε να εκτελεστεί (αυτό συμβαίνει με την παραδοσιακή Εικονική Μηχανή (*Virtual Machine*)). Πιο σύγχρονες εφαρμογές της εικονικής Μηχανής μπορούν και μεταγλωττίζουν εκ των προτέρων τμήματα *bytecode* απευθείας σε κώδικα μηχανής (εγγενή κώδικα ή *native code*) με αποτέλεσμα να βελτιώνεται η ταχύτητα). Χωρίς αυτό δε θα ήταν δυνατή η εκτέλεση λογισμικού γραμμένου σε *Java*. 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 και τα λοιπά.

Οτιδήποτε θέλει να κάνει ο προγραμματιστής (ή ο χρήστης) γίνεται μέσω της εικονικής μηχανής. Αυτό βοηθάει στο να υπάρχει μεγαλύτερη ασφάλεια στο σύστημα γιατί η εικονική μηχανή είναι υπεύθυνη για την επικοινωνία χρήστη - υπολογιστή. 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. Από την άλλη μεριά ούτε ο χρήστης μπορεί να κατεβάσει «κακό» κώδικα από το δίκτυο και να τον εκτελέσει. Αυτό είναι ιδιαίτερα χρήσιμο για μεγάλα [κατανεμημένα συστήματα](#) όπου πολλοί χρήστες χρησιμοποιούν το ίδιο πρόγραμμα συγχρόνως.

Ακόμα μία ιδέα που βρίσκεται πίσω από τη *Java* είναι η ύπαρξη του [συλλέκτη απορριμμάτων](#) (*Garbage Collector*). 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. Αυτή η απελευθέρωση μνήμης στη *Java* είναι αυτόματη και γίνεται μέσω του συλλέκτη απορριμμάτων. Υπεύθυνη για αυτό είναι και πάλι η εικονική μηχανή η οποία 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. Έτσι 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 (και όχι μόνο) τα πλεονεκτήματα, η *Java* αρχικά ήταν πιο αργή σε σχέση με άλλες προγραμματιστικές γλώσσες υψηλού επιπέδου (high-level) όπως η *C* και η *C++*. Εμπειρικές μετρήσεις στο παρελθόν είχαν δείξει ότι η *C++* μπορούσε να είναι αρκετές φορές γρηγορότερη από την *Java*. Ωστόσο 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 (Just In Time), οι οποίοι μετατρέπουν τον κώδικα byte απευθείας σε γλώσσα μηχανής, η διαφορά ταχύτητας από τη *C++* έχει μικρύνει κατά πολύ.

Οι τελευταίες εκδόσεις του *javac* με τη χρήση της τεχνολογίας Hot Spot έχουν καταφέρει αξιόλογες επιδόσεις που πλησιάζουν ή και ξεπερνούν σε μερικές περιπτώσεις τον εγγενή κώδικα.^[15]



XML

Η XML (αγγλ. αρκτικόλεξο από το **Extensible Markup Language**) είναι μία [γλώσσα σήμανσης](#), που περιέχει ένα σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση κειμένων. Ορίζεται, κυρίως, στην προδιαγραφή XML 1.0 ([XML 1.0 Specification](#)), που δημιούργησε ο διεθνής οργανισμός προτύπων W3C (World Wide Web Consortium), αλλά και σε διάφορες άλλες σχετικές προδιαγραφές [ανοιχτών προτύπων](#).

Η XML σχεδιάστηκε δίνοντας έμφαση στην απλότητα, τη γενικότητα και τη χρησιμότητα στο Διαδίκτυο. Είναι μία μορφοποίηση δεδομένων κειμένου, με ισχυρή υποστήριξη [Unicode](#) για όλες τις γλώσσες του κόσμου. Αν και η σχεδίαση της XML εστιάζει στα κείμενα, χρησιμοποιείται ευρέως για την αναπαράσταση αυθαίρετων [δομών δεδομένων](#), που προκύπτουν για παράδειγμα στις υπηρεσίες ιστού. Υπάρχει μία ποικιλία [διεπαφών προγραμματισμού εφαρμογών](#), που μπορούν να χρησιμοποιούν οι προγραμματιστές, για να προσπελαίνουν δεδομένα XML, αλλά και διάφορα συστήματα σχημάτων XML, τα οποία είναι σχεδιασμένα για να βοηθούν στον ορισμό γλωσσών, που προκύπτουν από την XML.^[16]



Εργαλεία που χρησιμοποιήθηκαν

Android Studio

Για την ανάπτυξη της Android εφαρμογής CityFix χρησιμοποιήθηκε το ολοκληρωμένο περιβάλλον ανάπτυξης Android Studio.

Το **Android Studio** είναι ένα [ολοκληρωμένο προγραμματιστικό περιβάλλον](#) (IDE) για ανάπτυξη εφαρμογών στην πλατφόρμα Android. Ανακοινώθηκε στις 16 Μαΐου 2013 στο συνέδριο Google I/O από την Google Product Manager, Katherine Chou. Το Android Studio είναι διαθέσιμο ελεύθερα με την άδεια Apache License 2.0.

Το Android Studio ήταν διαθέσιμο σε πρώιμο στάδιο για προεπισκόπηση ξεκινώντας από την έκδοση 0.1 τον Μάιο του 2013, έπειτα ξεκίνησε το δοκιμαστικό στάδιο από την έκδοση 0.8 που βγήκε τον Ιούνιο του 2014. Η πρώτη σταθερή έκδοση βγήκε το Δεκέμβριο του 2014, με την έκδοση 1.0.

Βασισμένο στο λογισμικό της JetBrains' IntelliJ IDEA, το Android Studio σχεδιάστηκε αποκλειστικά για προγραμματισμό Android . Είναι διαθέσιμο για Windows, Mac OS X και Linux , και αντικατέστησε τα Eclipse Android Development Tools (ADT) ως το κύριο IDE της Google για ανάπτυξη εφαρμογών Android.^[17]



Netbeans

Για την ανάπτυξη του web-app στον apache server χρησιμοποιήθηκε το δημοφιλές ολοκληρωμένο περιβάλλον ανάπτυξης **Netbeans**.

Το **Netbeans** ανήκε στην **Sun** και πέρασε και αυτό στην **Oracle** όταν η δεύτερη εξαγόρασε την πρώτη. Χρησιμοποιείται κυρίως για ανάπτυξη Java εφαρμογών αλλά υποστηρίζει κι άλλες γλώσσες όπως PHP, C , C++ και HTML5. Υπάρχουν εκδόσεις για όλα τα δημοφιλή λειτουργικά συστήματα (Microsoft Windows, Mac OS X, Linux, Solaris).^[18]



Υπόλοιπα

Για την συγγραφή της εργασίας προτού αυτή γίνει export σε pdf μορφή, χρησιμοποιήθηκε το [Microsoft Word 2007](#).^[19]



Για την επικοινωνία με τον server, το ανέβασμα ή κατέβασμα αρχείων από και προς τον server χρησιμοποιήθηκαν τα PuTTY και Filezilla .^{[20] [21]}



Συσκευές που χρησιμοποιήθηκαν για τον έλεγχο της εφαρμογής

Κατά την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν τέσσερις (4) συσκευές με διαφορετικές αναλογίες οθόνες και διαφορετικές εκδόσεις του λειτουργικού Android για να διασφαλιστεί η απροβλημάτιστη λειτουργία της εφαρμογής σε μια ευρεία γκάμα συσκευών. Στον παρακάτω πίνακα παρουσιάζονται τα μοντέλα των συσκευών καθώς και η αναλογία της οθόνης και η έκδοση του λειτουργικού για κάθε ένα από αυτά.

Μοντέλο	Μέγεθος οθόνης	Έκδοση λειτουργικού	API Level	Εικόνα
Samsung Galaxy Trend Plus – GT-S7580	4"	4.2.2 Jelly Bean	17	
LG L90 D405	4.7"	4.4.2 KitKat	19	
Sony Ericsson Xperia Arc S	4.2"	4.0.4 Ice Cream Sandwich	15	
ZTE Blade III	4"	4.0 Ice Cream Sandwich	14	

Πέρα από τις παραπάνω συσκευές, χρησιμοποιήθηκε και ο integrated Android simulator του Android Studio.^[21] Μέσω αυτού, η εφαρμογή ελέγχθηκε σε μια πληθώρα emulated συσκευών, διαφορετικών μεγεθών και API Level.

Παρουσίαση εφαρμογής

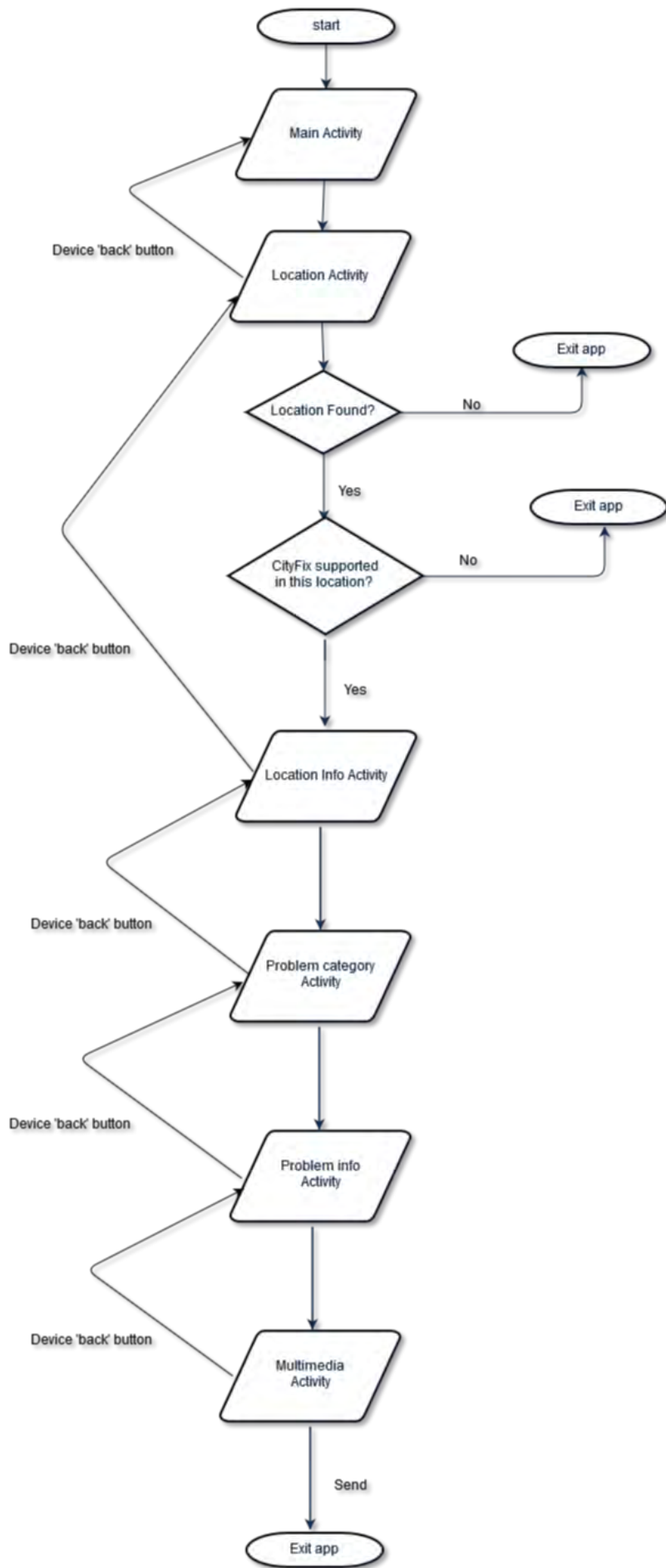
Σε αυτό το κεφάλαιο θα γίνει μια παρουσίαση της android εφαρμογής CityFix και θα συζητηθούν ορισμένα θέματα που σχετίζονται με το navigation αυτής.

Η εφαρμογή CityFix βοηθάει τον χρήστη να υλοποιήσει έναν πολύ συγκεκριμένο σκοπό, να αναφέρει στον τοπικό του Δήμο κάποιο πρόβλημα το οποίο αντιμετωπίζει συχνά ώστε αυτό να επιλυθεί. Έτσι η εφαρμογή θα πρέπει να ικανοποιεί ένα σύνολο απαιτήσεων όπως η απλότητα, η σαφήνεια και η ταχύτητα ολοκλήρωσης. Σε αντίθεση με κάποιο παιχνίδι ή με κάποια εφαρμογή περιεχομένου όπως για παράδειγμα εφαρμογές ειδήσεων ή κοινωνικής δικτύωσης, ο χρήστης δεν επιθυμεί να παραμείνει για πολύ χρόνο στην εφαρμογή αλλά να ολοκληρώσει την διαδικασία γρήγορα και να κλείσει την εφαρμογή.

Λαμβάνοντας αυτά υπόψη, κρίθηκε συνετό η εφαρμογή να έχει ένα απλό και straightforward navigation. Κάθε δραστηριότητα είναι καλά ορισμένη και σε αυτήν ο χρήστης ολοκληρώνει ένα συγκεκριμένο και διακριτό βήμα της διαδικασίας αναφοράς του προβλήματος όπως για παράδειγμα την επιλογή του είδους του προβλήματος ή την προσθήκη μιας φωτογραφίας. Μόλις ολοκληρώσει τις ενέργειες που απαιτούνται από αυτόν σε μια δραστηριότητα, ο χρήστης πιέζοντας το κουμπί «Επόμενο» οδηγείται στην επόμενη δραστηριότητα και ούτω καθεξής. Εάν ο χρήστης θελήσει να διορθώσει κάτι που επέλεξε σε κάποια προηγούμενη δραστηριότητα, μπορεί πατώντας το κουμπί «Πίσω» της συσκευής να διατρέξει τις δραστηριότητες με την αντίθετη φορά μέχρι να φτάσει σε αυτήν που επιθυμεί.

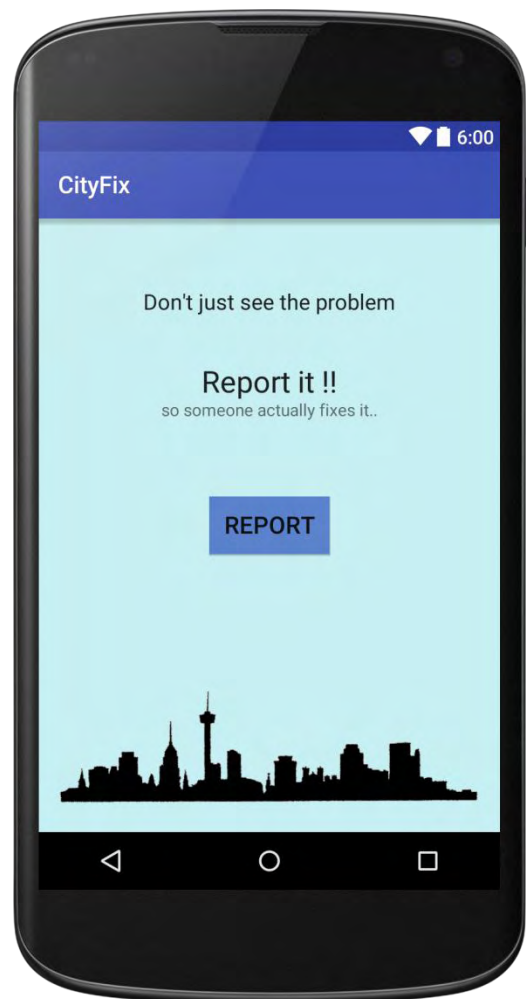
Με αυτό το streamlined navigation, η διαδικασία αναφοράς ενός προβλήματος, αν υποθέσουμε πως ο χρήστης δεν χρειάζεται να πάει προς τα πίσω για να διορθώσει κάποιο προηγούμενο λάθος του, κι αν δεν παρέχει επιπλέον text-input στις δύο δραστηριότητες που τον προτρέπουν να το κάνει, διαρκεί περίπου τριάντα (30) με σαράντα (40) δευτερόλεπτα. Στην περίπτωση που παρέχει text-input ή πηγαίνει προς τα πίσω για να διορθώσει λάθη ο χρόνος αυτός προσαυξάνεται κατά ένα ποσό που είναι user-specific και δεν μπορεί να εκτιμηθεί με ακρίβεια. Είναι λογικό πάντως να υποθέσουμε πως μόνο σε εξαιρετικά σπάνιες περιπτώσεις ο χρόνος παραμονής στην εφαρμογή θα ξεπερνά τα πέντε (5) λεπτά.

Στην συνέχεια παρουσιάζονται ένα απλό flowchart του navigation της εφαρμογής και οι δραστηριότητες μία προς μία μαζί με μια σύντομη περιγραφή της λειτουργίας κάθε μιας.



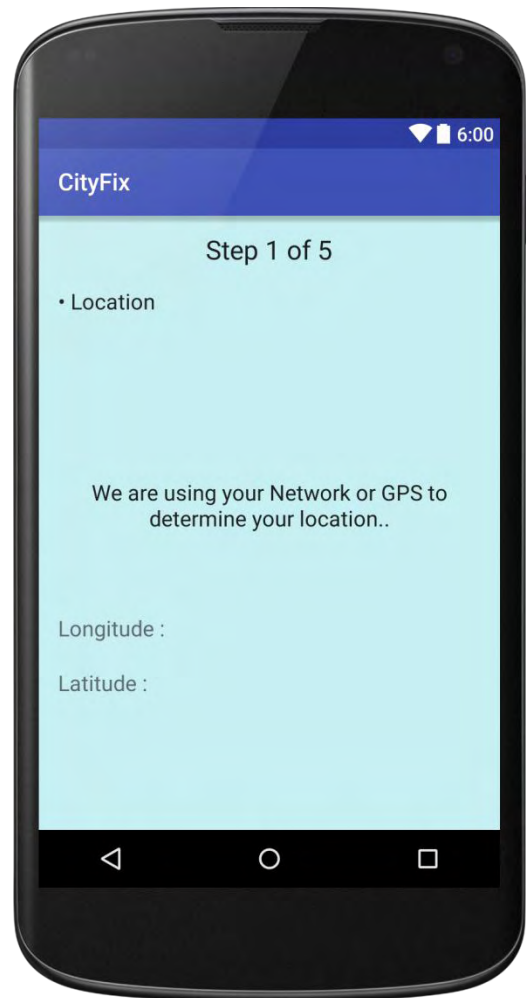
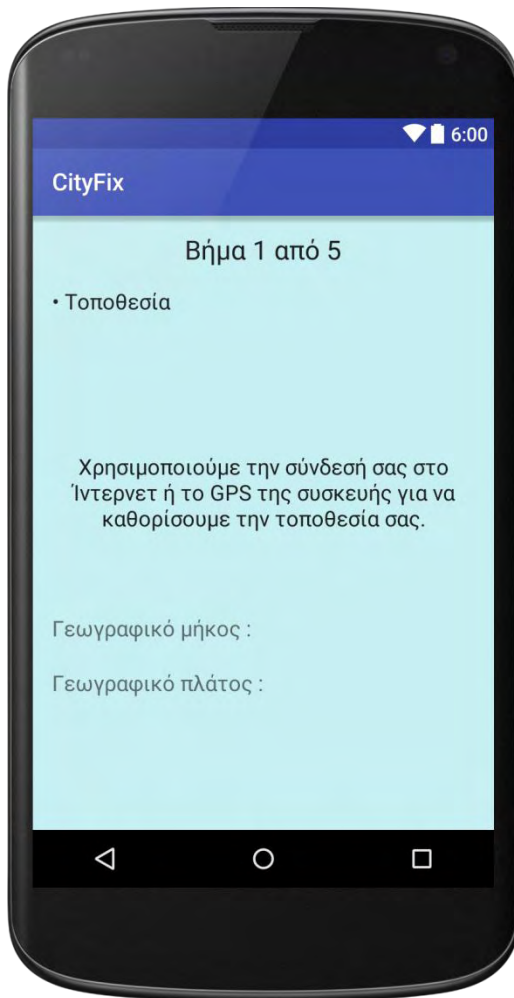
Κύρια δραστηριότητα

Ο χρήστης ανοίγοντας την εφαρμογή βλέπει την Κύρια δραστηριότητα, όπου ένα μήνυμα τον προτρέπει να αναφέρει το πρόβλημα που έχει παρατηρήσει. Πατώντας το κουμπί «Αναφορά» μεταφέρεται στην επόμενη δραστηριότητα.



Δραστηριότητα τοποθεσίας

Σε αυτήν την δραστηριότητα επιχειρείται να βρεθεί η τοποθεσία του χρήστη χρησιμοποιώντας τον GPS provider ή εάν αυτός δεν είναι διαθέσιμος, τον Network provider (Wi-Fi , Cellular data). Σε αυτό το σημείο υπάρχουν δύο (2) σενάρια:



Σενάριο 1 : Επιτυχής εύρεση χρήστη

Σε αυτό το σενάριο η τοποθεσία του χρήστη βρέθηκε επιτυχώς. Τα πεδία «γεωγραφικό μήκος» και «γεωγραφικό πλάτος» παίρνουν τις αντίστοιχες τιμές. Πραγματοποιείται έλεγχος για να διαπιστωθεί εάν ο Δήμος στον οποίο βρίσκεται ο χρήστης υποστηρίζει την εφαρμογή και ανάλογα με το αποτέλεσμα αυτού του ελέγχου υπάρχουν δύο υποσενάρια.

Υποσενάριο 1 : Ο Δήμος υποστηρίζει την εφαρμογή

Σε αυτήν την περίπτωση εμφανίζεται ένα “dialog pop-up” που ενημερώνει τον χρήστη πως η θέση του εντοπίστηκε με επιτυχία, εμφανίζεται το όνομα του Δήμου στον οποίον θα αποσταλεί η αναφορά και καθορίζεται (χωρίς να ενημερώνεται ο χρήστης για αυτό) η IP του Server στον οποίον θα σταλεί η αναφορά. Ο χρήστης πατώντας το κουμπί «Εντάξει» κάνει “dismiss” το pop-up και πατώντας το κουμπί «Επόμενο» στο κάτω δεξί μέρος της εφαρμογής, το οποίο προηγουμένως δεν ήταν ορατό, περνάει στην επόμενη δραστηριότητα.

Υποσενάριο 2 : Ο Δήμος δεν υποστηρίζει την εφαρμογή

Σε αυτήν την περίπτωση πάλι εμφανίζεται ένα “dialog pop-up” που αυτήν την φορά ενημερώνει τον χρήστη πως ο Δήμος του δεν υποστηρίζει την εφαρμογή και ένα κουμπί με τίτλο «Κλείσιμο εφαρμογής». Πατώντας το κουμπί ο χρήστης μπορεί να κλείσει την εφαρμογή.

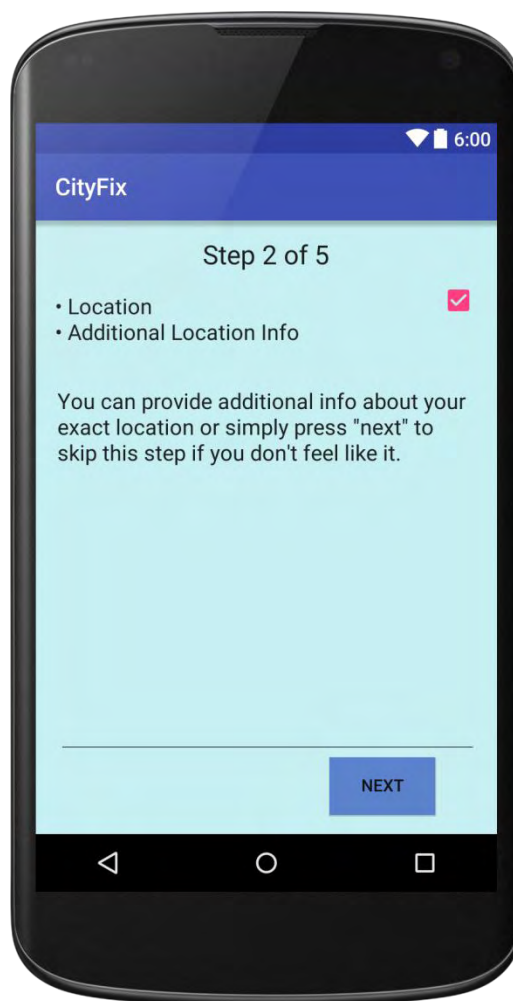
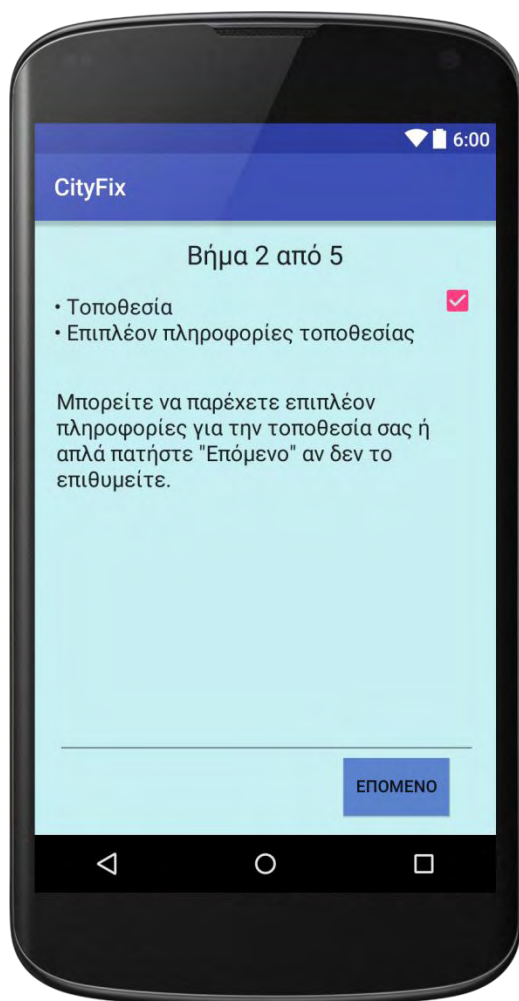
Σενάριο 2 : Ανεπιτυχής εύρεση χρήστη

Σε αυτό το σενάριο η τοποθεσία του χρήστη δεν κατέστη δυνατό να βρεθεί επιτυχώς. Μια σειρά από λόγους μπορεί να ευθύνονται για αυτό με κυριότερους όμως την μη διαθεσιμότητα των providers που αναφέρθηκαν προηγουμένως. Πιο συγκεκριμένα, είτε το GPS της συσκευής δεν κατάφερε να επικοινωνήσει επιτυχώς με τους δορυφόρους είτε το κινητό του χρήστη έχει κακή ή καθόλου σύνδεση στο Ίντερνετ.

Σε αυτήν την περίπτωση εμφανίζεται ένα “dialog pop-up” που ενημερώνει τον χρήστη πως η θέση του δεν κατέστη δυνατόν να βρεθεί επιτυχώς και ένα κουμπί με τίτλο «Κλείσιμο εφαρμογής». Πατώντας το κουμπί ο χρήστης κλείνει την εφαρμογή.

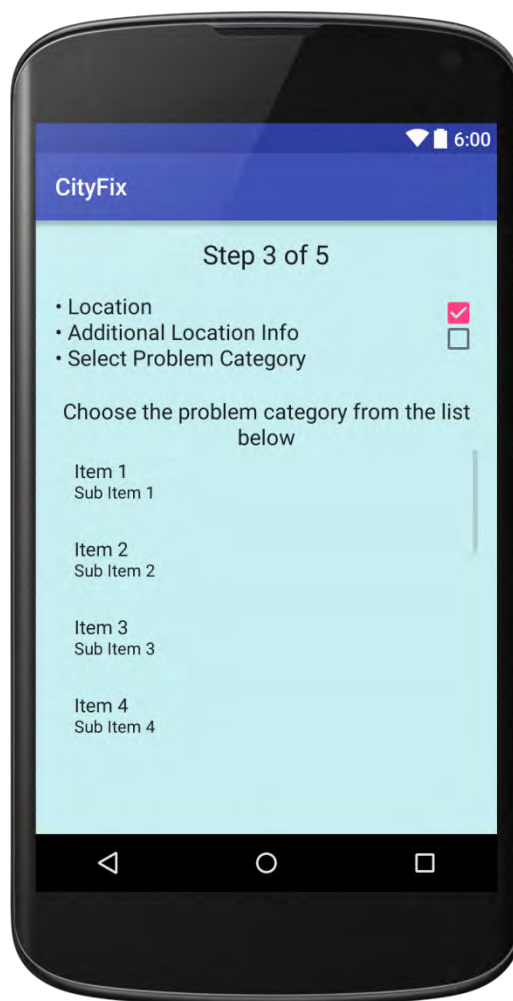
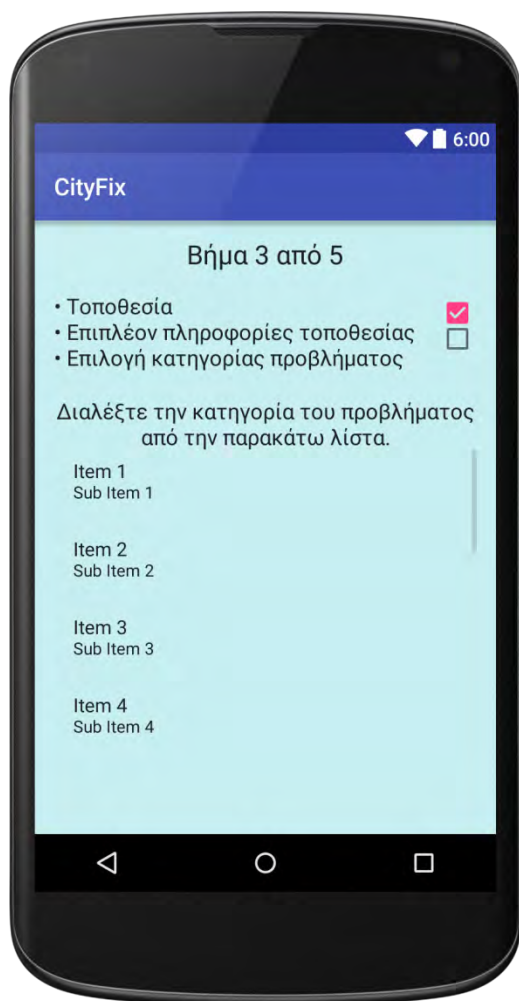
Δραστηριότητα πληροφοριών τοποθεσίας

Φτάνοντας σε αυτήν την δραστηριότητα, η εφαρμογή έχει ήδη εντοπίσει την θέση του χρήστη με την ακρίβεια που παρέχουν οι προαναφερθέντες providers. Ο χρήστης έχει την δυνατότητα να παρέχει επιπλέον πληροφορίες για την ακριβή τοποθεσία του προβλήματος με την μορφή κειμένου (π.χ. “Ακριβώς μπροστά από το Δημοτικό σχολείο” , “Ανάμεσα στο φαρμακείο και την καφετέρια” κτλ) ή αν θεωρεί πως δεν είναι απαραίτητο, μπορεί να πατήσει το κουμπί «Επόμενο» χωρίς να έχει εισάγει κάποιο κείμενο και να περάσει έτσι στην επόμενη δραστηριότητα.



Δραστηριότητα κατηγορίας προβλήματος

Σε αυτήν την δραστηριότητα ο χρήστης καλείται να επιλέξει την κατηγορία στην οποία εντάσσεται το πρόβλημα. Αυτό μπορεί να φανεί εξαιρετικά χρήσιμο στον υπάλληλο του Δήμου που δέχεται τις αναφορές καθώς μπορεί να προχωρήσει σε ταξινομημένη αναζήτηση ως προς το είδος του προβλήματος αντί να τα διαπερνά όλες τις αναφορές σειριακά. Ακόμη, εάν στον Δήμο υπάρχουν διάφορα τμήματα καθένα επιφορτισμένο με την διόρθωση συγκεκριμένου είδους προβλήματος, ο υπάλληλος που δέχεται τις αναφορές μπορεί εύκολα να επαναπροωθήσει κάθε πρόβλημα στο αντίστοιχο τμήμα.



Την ώρα που γράφονται αυτές οι γραμμές το CityFix υποστηρίζει τις εξής κατηγορίες προβλημάτων :

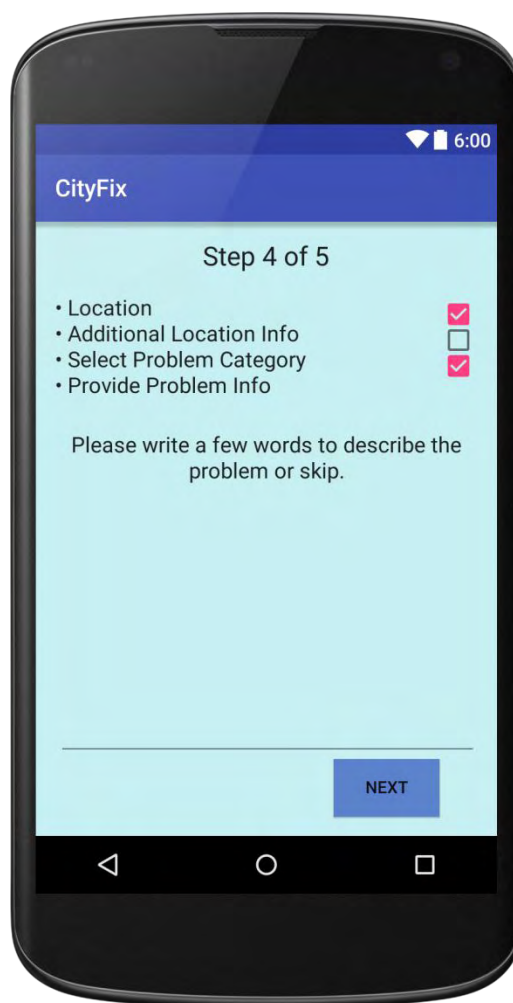
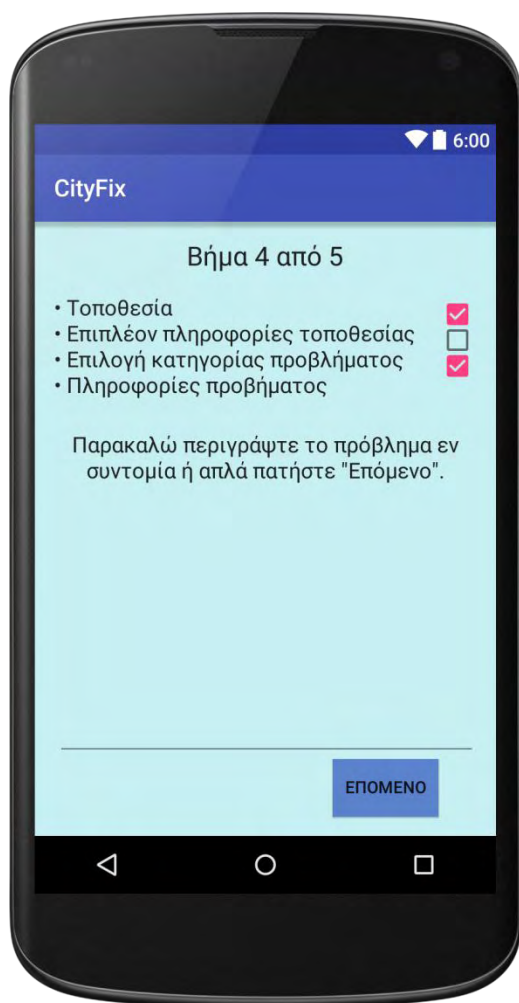
- Φωτισμός
- Φανάρια και σήματα οδικής κυκλοφορίας
- Δρόμοι και ποδηλατοδρόμοι
- Πεζόδρομοι, πεζοδρόμια και δημόσιοι χώροι
- Πάρκα και αθλητικοί χώροι
- Δέντρα και διακόσμηση
- Μέσα μαζικής κυκλοφορίας
- Δημόσια κτίρια
- Απορρίμματα και αποχετεύσεις
- Εγκαταλελειμμένα οχήματα
- Άλλα

Οι παραπάνω κατηγορίες προέκυψαν έπειτα από σχετική ερώτηση που υπέβαλα σε υπάλληλο του Δήμου Κατερίνης και της γραπτής του ανταπόκρισης. Είναι πιθανό μεγαλύτεροι Δήμοι να έχουν ακόμη περισσότερες αρμοδιότητες.

Αφού ο χρήστης επιλέξει την κατηγορία του προβλήματος, μπορεί να πατήσει το κουμπί «Επόμενο» στο κάτω δεξί μέρος της εφαρμογής, το οποίο προηγουμένως δεν ήταν ορατό, κι έτσι να περάσει στην επόμενη δραστηριότητα.

Δραστηριότητα πληροφοριών προβλήματος

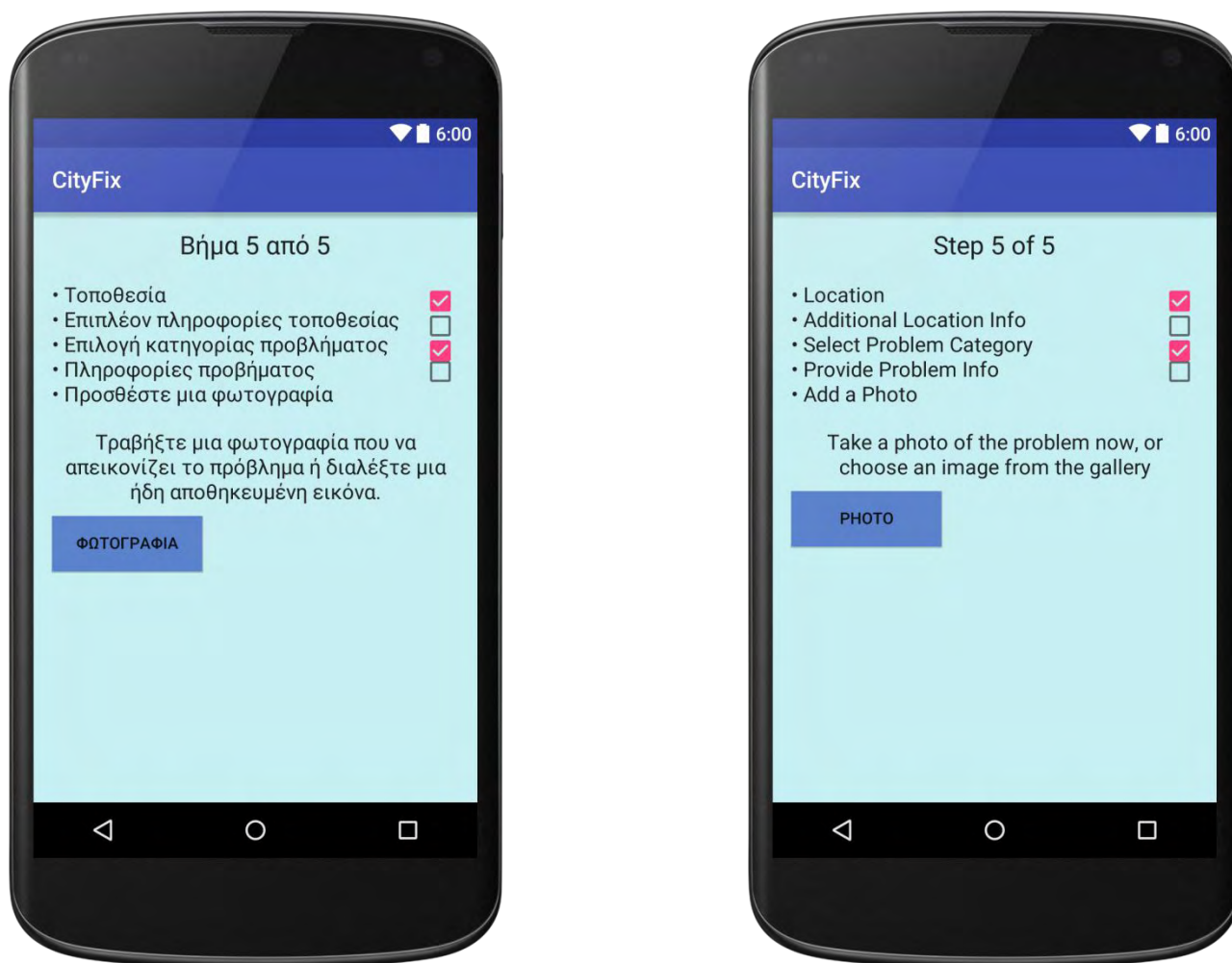
Στην συγκεκριμένη δραστηριότητα ο χρήστης καλείται να γράψει μερικές γραμμές που να περιγράφουν το πρόβλημα. Έχει φυσικά την δυνατότητα να περάσει στην επόμενη δραστηριότητα αφήνοντας το πεδίο κειμένου κενό, όμως συνίσταται να το συμπληρώσει. Το κείμενο καλό θα ήταν να είναι σύντομο και περιεκτικό. Για παράδειγμα, “Το δέντρο έχει φουντώσει δυσχεραίνοντας την διέλευση πεζών από το πεζοδρόμιο” ή “Η λάμπα ορισμένες βραδιές τρεμοπαίζει και άλλες δεν ανάβει καθόλου” κτλ.



Αφού συμπληρώσει το κείμενο της επιλογής του, ο χρήστης πιέζοντας το κουμπί «Επόμενο» οδηγείται στην τελευταία δραστηριότητα της εφαρμογής.

Δραστηριότητα πολυμέσων

Η δραστηριότητα πολυμέσων μαζί με την δραστηριότητα τοποθεσίας αποτελούν τις δύο πιο σημαντικές ίσως δραστηριότητες της εφαρμογής. Σε αυτήν ο χρήστης καλείται να τραβήξει μια φωτογραφία που να αποθανατίζει το πρόβλημα. Πιέζοντας το κουμπί «Φωτογραφία» εμφανίζεται ένα “dialog pop-up” από το οποίο ο χρήστης μπορεί να επιλέξει να τραβήξει μια φωτογραφία ή εάν έχει ήδη φωτογραφίσει το πρόβλημα, να διαλέξει την αποθηκευμένη φωτογραφία όπου κι αν αυτή βρίσκεται (phone memory, sd card, Google drive κτλ).



Αφότου ο χρήστης έχει τραβήξει ή έχει επιλέξει φωτογραφία μπορεί να πατήσει το κουμπί «Αποστολή» το οποίο βρίσκεται δίπλα στο κουμπί «Φωτογραφία» και δεν ήταν ορατό προηγουμένως, για να αποστείλει την αναφορά του στον Δήμο της περιοχής του όπως αυτός ορίστηκε στην δραστηριότητα τοποθεσίας.

Μόλις η εφαρμογή λάβει την απάντηση του Server για την επιτυχή αποστολή της αναφοράς, ειδοποιεί τον χρήστη για την επιτυχή ολοκλήρωση της διαδικασίας μέσω ενός “toast-message” καθώς κι ενός “dialog pop-up” στο οποίο υπάρχει κουμπί με τον τίτλο «Κλείσιμο εφαρμογής» το οποίο μπορεί να πατήσει ο χρήστης για να εξέλθει. Σε περίπτωση που ο Server αναφέρει κάποιο πρόβλημα στη αποστολή της αναφοράς, το toast-message εμφανίζει τον αντίστοιχο κωδικό σφάλματος.

ΓΙΑ ΤΗΝ ΕΦΑΡΜΟΓΗ.

Configuration Files

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.sotiris.cityfix">

    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.MANAGE_DOCUMENTS"/>
    <uses-feature android:name="android.hardware.screen.portrait" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name="com.example.sotiris.cityfix.MainActivity"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name="com.example.sotiris.cityfix.LocationActivity"
            android:screenOrientation="portrait"/>
        <activity android:name="com.example.sotiris.cityfix.LocationInfoActivity"
            android:screenOrientation="portrait" />
        <activity android:name="com.example.sotiris.cityfix.ProblemCategoryActivity"
            android:screenOrientation="portrait" />
        <activity android:name="com.example.sotiris.cityfix.ProblemInfoActivity"
            android:screenOrientation="portrait" />
        <activity android:name="com.example.sotiris.cityfix.MultimediaActivity"
            android:screenOrientation="portrait"></activity>
    </application>
</manifest>
```

build.gradle (Module: app)

```
apply plugin: 'com.android.application'

android {

    compileSdkVersion 23
    buildToolsVersion "24.0.0"

    defaultConfig {
        applicationId "com.example.sotiris.cityfix"
        minSdkVersion 16
        targetSdkVersion 19
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-rules.pro')
        }
    }
    useLibrary 'org.apache.http.legacy'
    productFlavors {
    }
}

dependencies {
    compile fileTree(include: ['*.jar'], dir: 'libs')
    testCompile 'junit:junit:4.12'
    compile 'com.android.support:appcompat-v7:23.4.0'
    compile 'com.android.support:design:23.4.0'
    compile
'org.jbundle.util.osgi.wrapped:org.jbundle.util.osgi.wrapped.org.apache.http.client:4.1.2'
    compile 'com.mcxiaoke.volley:library-aar:1.0.0'
}
```

Main activity

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<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="com.example.sotiris.cityfix.MainActivity"
    android:background="#c7eef0">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity1_message1"
        android:id="@+id/textView"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="43dp"
        android:textSize="18dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity1_message2"
        android:id="@+id/textView2"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="41dp"
        android:textSize="25dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="@string/activity1_message3"
        android:id="@+id/textView3"
        android:layout_below="@+id/textView2"
        android:layout_centerHorizontal="true" />

    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="@string/activity1_report"
        android:id="@+id/button"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        android:clickable="true"
        android:textSize="20dp"
        android:onClick="onClick"
        android:background="#5b82cb" />

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:src="@mipmap/skyline" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.sotiris.cityfix;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.content.Intent;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;

public class MainActivity extends AppCompatActivity {

    public Animation animAlpha;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        System.out.println("*****App has started*****");

        animAlpha = AnimationUtils.loadAnimation(this, R.anim.anime_alpha);
    }

    public void onClick(View view){

        view.startAnimation(animAlpha);
        Intent Step1 = new Intent(this, LocationActivity.class);
        startActivity(Step1);
    }
}
```

Location Activity

Activity_location.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<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="com.example.sotiris.cityfix.LocationActivity"
    android:background="#c7eef0">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/step_1"
        android:id="@+id/textView5"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity2_location"
        android:id="@+id/textView4"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="45dp"
        android:textSize="18dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity2_message"
        android:id="@+id/textView6"
        android:textAlignment="center"
        android:textSize="18dp"
        android:layout_marginBottom="70dp"
        android:layout_above="@+id/textView7"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:text="@string/activity2_longitude"
        android:id="@+id/textView7"
        android:textSize="18dp"
        android:clickable="false"
        android:layout_marginBottom="24dp"
        android:layout_above="@+id/textView8"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceMedium"
```

```

android:text="@string/activity2_latitude"
    android:id="@+id/textView8"
    android:textSize="18dp"
    android:layout_marginBottom="47dp"
    android:layout_above="@+id/button4"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:id="@+id/textView9"
    android:layout_toRightOf="@+id/textView7"
    android:layout_marginLeft="30dp"
    android:layout_alignTop="@+id/textView7"
    />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:id="@+id/textView10"
    android:layout_alignTop="@+id/textView8"
    android:layout_alignLeft="@+id/textView9"
    android:layout_alignStart="@+id/textView9" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/activity2_next"
    android:id="@+id/button4"
    android:layout_alignParentBottom="true"
    android:layout_toRightOf="@+id/textView5"
    android:layout_toEndOf="@+id/textView5"
    android:clickable="true"
    android:onClick="onClickNext"
    android:visibility="invisible"
    android:background="#5b82cb" />

</RelativeLayout>

```


LocationActivity.java

```
package com.example.sotiris.cityfix;

import android.location.Address;
import android.location.Geocoder;
import android.provider.Settings;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.content.Intent;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.TextView;
import android.text.Editable;
import android.text.TextWatcher;
import android.widget.Button;
import android.content.Context;
import android.location.Location;
import android.location.LocationManager;
import android.widget.Toast;
import android.location.LocationListener;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.List;
import java.util.Locale;

public class LocationActivity extends AppCompatActivity implements LocationListener {

    private static TextView Field1;
    private TextView latitudeField;
    private TextView longitudeField;
    public Button NextButton;
    private LocationManager locationManager;
    private String provider;
    public static String City;
    private String registry;
    private String CheckCitySupportResult;
    public float lat;
    public float lng;
    public Animation animAlpha;
    public Location location;

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

        latitudeField = (TextView) findViewById(R.id.textView9);
        longitudeField = (TextView) findViewById(R.id.textView10);
        NextButton = (Button) findViewById(R.id.button4);

        Field1 = (TextView) findViewById(R.id.textView9);
        Field1.addTextChangedListener(new TextWatcher() {
            @Override
            public void onTextChanged(CharSequence s, int start, int before, int count) {
                // Fires right as the text is being changed (even supplies the range of text)
            }

            @Override
            public void beforeTextChanged(CharSequence s, int start, int count, int after) {
                // Fires right before text is changing
            }
        });
    }
}
```

```

    }

    @Override
    public void afterTextChanged(Editable s) {
        // Fires right after the text has changed
        //NextButton.setVisibility(View.VISIBLE);
    }
});

// Check if GPS is ON and if not prompt the user to open it

LocationManager service = (LocationManager) getSystemService(LOCATION_SERVICE);
boolean enabled = service.isProviderEnabled(LocationManager.GPS_PROVIDER);

// check if enabled and if not send user to the GSP settings
if (!enabled) {
    Intent intent = new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
    startActivity(intent);
}

// Get the location manager
locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);

// Choose the provider
provider = locationManager.NETWORK_PROVIDER;

location = locationManager.getLastKnownLocation(provider);

// Initialize the location fields
if (location != null) {
    System.out.println("Provider " + provider + " has been selected.");
    onLocationChanged(location);
} else {
    latitudeField.setText("No GPS or Internet connection");
    longitudeField.setText("No GPS or Internet connection");

    DialogFragment newFragment = new LocationDialogFragment_LocationNull(); // Show dialog
failure in detecting the City name
    newFragment.show(getSupportFragmentManager(), "Location Null");
}

animAlpha = AnimationUtils.loadAnimation(this, R.anim.anime_alpha);
}

```

```

/* Request updates at startup */
@Override
protected void onResume() {
    super.onResume();

    locationManager.requestLocationUpdates(provider, 60000, 5, this); //Update every 60
seconds or every time user moves more than 5 meters

    Geocoder gcd = new Geocoder(getBaseContext(), Locale.getDefault()); // Find City name
    List<Address> addresses = null;
    try {
        addresses = gcd.getFromLocation(lat, lng, 1);
    } catch (IOException e) {
        e.printStackTrace();
    }
    if (addresses.size() > 0)
        System.out.println(addresses.get(0).getLocality());

    City = addresses.get(0).getLocality().toString();

    CheckCitySupportResult = CheckCitySupport(City); // Check if the city
is supported

    if (CheckCitySupportResult.equals("City not supported")) { // Show dialog for
not supported city
        DialogFragment newFragment = new LocationDialogFragment_CityNotSupported();
        newFragment.show(getSupportFragmentManager(), "Sorry");

    } else { // Show dialog for
supported city
        NextButton.setVisibility(View.VISIBLE);
        DialogFragment newFragment = new LocationDialogFragment_CitySupported();
        newFragment.show(getSupportFragmentManager(), "Success");

    }

}

public String CheckCitySupport(String city) { // Check CityFix
availability in current city

    try {
        BufferedReader reader = new BufferedReader(new
InputStreamReader(getAssets().open("Cities")));

        registry = reader.readLine();
        while (!registry.equals(city) && !registry.equals("EOF")) {
            registry = reader.readLine();
        }
        registry = reader.readLine();
    } catch (IOException e) {
        e.printStackTrace();
    }

    return registry;
}

```

```

/* Remove the locationlistener updates when Activity is paused */
@Override
protected void onPause() {
    super.onPause();

    locationManager.removeUpdates(this);
}

@Override
public void onLocationChanged(Location location) {
    lat = (float) (location.getLatitude());
    lng = (float) (location.getLongitude());
    latitudeField.setText(String.valueOf(lat));
    longitudeField.setText(String.valueOf(lng));
}

@Override
public void onStatusChanged(String provider, int status, Bundle extras) {
    // TODO Auto-generated method stub
}

@Override
public void onProviderEnabled(String provider) {
    Toast.makeText(this, "Enabled new provider " + provider,
        Toast.LENGTH_SHORT).show();
}

@Override
public void onProviderDisabled(String provider) {
    Toast.makeText(this, "Disabled provider " + provider,
        Toast.LENGTH_SHORT).show();
}

public void onClickNext(View view){

    view.startAnimation(animAlpha);

    Intent Step2 = new Intent(this, LocationInfoActivity.class);

    String lng = longitudeField.getText().toString();
    String lat = latitudeField.getText().toString();
    Step2.putExtra("LongitudeValue", lng);
    Step2.putExtra("LatitudeValue", lat);
    Step2.putExtra("Server_ip", registry);

    startActivity(Step2);
}
}

```

LocationDialogFragment_CityNotSupported.java

```
package com.example.sotiris.cityfix;

import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AlertDialog;

public class LocationDialogFragment_CityNotSupported extends DialogFragment {
    @Override

    public Dialog onCreateDialog(Bundle savedInstanceState) {

        AlertDialog.Builder builder = new AlertDialog.Builder(getActivity());
        builder.setTitle(R.string.activity2_not_supported_dialog_title);
        builder.setMessage(R.string.activity2_not_supported_dialog_message);

        builder.setNegativeButton(R.string.activity2_not_supported_dialog_cancel, new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int id) { //
Exit app.
                Intent homeIntent = new Intent(Intent.ACTION_MAIN);
                homeIntent.addCategory(Intent.CATEGORY_HOME);
                homeIntent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                startActivity(homeIntent);
            }
        });

        return builder.create();
    }
}
```

LocationDialogFragment_CitySupported.java

```
package com.example.sotiris.cityfix;

import android.app.Dialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AlertDialog;

public class LocationDialogFragment_CitySupported extends DialogFragment {
    @Override

    public Dialog onCreateDialog(Bundle savedInstanceState) {

        AlertDialog.Builder builder = new AlertDialog.Builder(getActivity());
        builder.setTitle(R.string.activity2_supported_dialog_title);
        String message = getString(R.string.activity2_supported_dialog_message) + "
'" + LocationActivity.City + "'"+ ".";
        builder.setMessage(message);
        builder.setPositiveButton(R.string.activity2_supported_dialog_ok, new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int id) { //
Dismiss dialog.

            }
        });

        return builder.create();
    }
}
```

LocationDialogFragment_LocationNull.java

```
package com.example.sotiris.cityfix;

import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AlertDialog;

public class LocationDialogFragment_LocationNull extends DialogFragment {
    @Override

    public Dialog onCreateDialog(Bundle savedInstanceState) {

        AlertDialog.Builder builder = new AlertDialog.Builder(getActivity());
        builder.setTitle(R.string.activity2_location_null_dialog_title);
        builder.setMessage(R.string.activity2_location_null_dialog_message);

        builder.setNegativeButton(R.string.activity2_location_null_dialog_cancel, new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int id) { //
Exit app.
                Intent homeIntent = new Intent(Intent.ACTION_MAIN);
                homeIntent.addCategory( Intent.CATEGORY_HOME );
                homeIntent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                startActivity(homeIntent);
            }
        });

        return builder.create();
    }
}
```

Location Info Activity

Activity_location_info.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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="com.example.sotiris.cityfix.LocationInfoActivity"
    android:background="#c7eef0">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/step_2"
        android:id="@+id/textView11"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity3_location"
        android:id="@+id/textView12"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="45dp"
        android:textSize="18dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity3_additional_location_info"
        android:id="@+id/textView13"
        android:layout_below="@+id/textView12"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:textSize="18dp" />

```



```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/activity3_next"
    android:id="@+id/button6"
    android:clickable="true"
    android:onClick="onClickNext"
    android:layout_alignParentBottom="true"
    android:layout_toRightOf="@+id/textView11"
    android:layout_toEndOf="@+id/textView11"
    android:background="#5b82cb" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity3_message"
    android:id="@+id/textView14"
    android:layout_marginTop="36dp"
    android:singleLine="false"
    android:textSize="18dp"
    android:layout_below="@+id/textView13"
    android:layout_alignLeft="@+id/LocationInput"
    android:layout_alignStart="@+id/LocationInput" />
```

```
<EditText
    android:layout_width="350dp"
    android:layout_height="wrap_content"
    android:inputType="textMultiLine"
    android:ems="10"
    android:id="@+id/LocationInput"
    android:focusable="true"
    android:layout_above="@+id/button6"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:layout_below="@+id/textView14" />
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox2"
    android:checked="true"
    android:layout_alignBottom="@+id/textView12"
    android:layout_alignRight="@+id/textView14"
    android:layout_alignEnd="@+id/textView14"
    android:clickable="false" />
```

```
</RelativeLayout>
```

LocationInfoActivity.java

```
package com.example.sotiris.cityfix;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.content.Intent;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.EditText;

public class LocationInfoActivity extends AppCompatActivity {

    EditText LocationInput;
    public String lng;
    public String lat;
    public String Server_ip;
    public String AdditionalInfo;
    public Animation animAlpha;

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

        Bundle Step1Info = getIntent().getExtras();
        lng = Step1Info.getString("LongitudeValue");
        lat = Step1Info.getString("LatitudeValue");
        Server_ip = Step1Info.getString("Server_ip");

        LocationInput = (EditText) findViewById(R.id.LocationInput);

        animAlpha = AnimationUtils.loadAnimation(this, R.anim.anime_alpha);
    }

    public void onClickNext(View view){

        view.startAnimation(animAlpha);

        AdditionalInfo = LocationInput.getText().toString();

        Intent step3 = new Intent(this, ProblemCategoryActivity.class);

        step3.putExtra("LongitudeValue", lng);
        step3.putExtra("LatitudeValue", lat);
        step3.putExtra("LocationInformation", AdditionalInfo);
        step3.putExtra("Server_ip", Server_ip);

        startActivity(step3);
    }
}
```

Problem Category Activity

activity_problem_category.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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="com.example.sotiris.cityfix.ProblemCategoryActivity"
    android:background="#c7eef0">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/step_3"
        android:id="@+id/textView15"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity4_location"
        android:id="@+id/textView16"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="45dp"
        android:textSize="18dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity4_additional_location_info"
        android:id="@+id/textView17"
        android:layout_below="@+id/textView16"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:textSize="18dp" />
```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity4_select_problem_category"
    android:id="@+id/textView18"
    android:layout_below="@+id/textView17"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity4_message"
    android:id="@+id/textView19"
    android:layout_marginTop="23dp"
    android:textAlignment="center"
    android:textSize="18dp"
    android:layout_below="@+id/textView18"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/activity4_next"
    android:id="@+id/button7"
    android:layout_alignParentBottom="true"
    android:layout_toRightOf="@+id/textView15"
    android:layout_toEndOf="@+id/textView15"
    android:clickable="true"
    android:onClick="onClickNext"
    android:background="#5b82cb"
    android:visibility="invisible" />

<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkbox"
    android:checked="true"
    android:layout_alignTop="@+id/textView16"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:clickable="false" />

<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkbox3"
    android:checked="false"
    android:layout_alignTop="@+id/textView17"
    android:layout_alignLeft="@+id/checkbox"
    android:layout_alignStart="@+id/checkbox"
    android:clickable="false"
    />

<ListView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/listview"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_above="@+id/button7"
    android:layout_below="@+id/textView19" />
</RelativeLayout>

```

ProblemCategoryActivity.java

```
package com.example.sotiris.cityfix;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.content.Intent;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.ListView;
import java.util.ArrayList;
import java.util.HashMap;
import android.content.Context;

public class ProblemCategoryActivity extends AppCompatActivity {

    public CheckBox LocationInfoBox;
    public String lng;
    public String lat;
    public String ProblemCategory;
    public String input;
    public String Server_ip;
    public String LocationInfoBoxChecked = "false";
    public Animation animAlpha;

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

        LocationInfoBox = (CheckBox) findViewById(R.id.checkBox3);

        Bundle Step2Info = getIntent().getExtras();

        input = Step2Info.getString("LocationInformation");
        lng = Step2Info.getString("LongitudeValue");
        lat = Step2Info.getString("LatitudeValue");
        Server_ip = Step2Info.getString("Server_ip");

        final Button NextButton = (Button) findViewById(R.id.button7);

        if(input.length() != 0) {
            LocationInfoBox.setChecked(true);
            LocationInfoBoxChecked = "true";
        }
    }
}
```

```

// Listview code
final ListView listview = (ListView) findViewById(R.id.listview);
final String[] values = new String[] {
    getString(R.string.activity5_problem_list_item1),
    getString(R.string.activity5_problem_list_item2),
    getString(R.string.activity5_problem_list_item3),
    getString(R.string.activity5_problem_list_item4),
    getString(R.string.activity5_problem_list_item5),
    getString(R.string.activity5_problem_list_item6),
    getString(R.string.activity5_problem_list_item7),
    getString(R.string.activity5_problem_list_item8),
    getString(R.string.activity5_problem_list_item9),
    getString(R.string.activity5_problem_list_item10),
    getString(R.string.activity5_problem_list_item11) };

final ArrayList<String> list = new ArrayList<String>();
for (int i = 0; i < values.length; ++i) {
    list.add(values[i]);
}
final StableArrayAdapter adapter = new StableArrayAdapter(this,
    android.R.layout.simple_list_item_1, list);
listview.setAdapter(adapter);

listview.setOnItemClickListener(new AdapterView.OnItemClickListener() {

    @Override
    public void onItemClick(AdapterView<?> parent, final View view,
        int position, long id) {
        final String item = (String) parent.getItemAtPosition(position);
        view.animate().setDuration(1000).alpha(0)
            .withEndAction(new Runnable() {
                @Override
                public void run() {
                    for (int i=0; i<values.length; i++){
                        if(values[i]!=item){

                            list.remove(values[i]);
                            adapter.notifyDataSetChanged();
                            view.setAlpha(1);

                        }

                    }

                }
            });

        ProblemCategory = item;
        NextButton.setVisibility(View.VISIBLE);
    }

});

animAlpha = AnimationUtils.loadAnimation(this, R.anim.anime_alpha);
}

```

```
public void onClickNext(View view){

    view.startAnimation(animAlpha);

    Intent step4 = new Intent(this, ProblemInfoActivity.class);

    step4.putExtra("LongitudeValue", lng);
    step4.putExtra("LatitudeValue", lat);
    step4.putExtra("LocationInformation", input);
    step4.putExtra("LocationInfoBoxChecked", LocationInfoBoxChecked);
    step4.putExtra("ProblemCategory", ProblemCategory);
    step4.putExtra("Server_ip", Server_ip);

    startActivity(step4);

}
}
```

StableArrayAdapter.java

```
package com.example.sotiris.cityfix;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.content.Intent;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.ListView;
import java.util.ArrayList;
import java.util.HashMap;
import android.content.Context;

class StableArrayAdapter extends ArrayAdapter<String> {

    HashMap<String, Integer> mIdMap = new HashMap<String, Integer>();

    public StableArrayAdapter(Context context, int textViewResourceId,
        ArrayList<String> objects) {
        super(context, textViewResourceId, objects);
        for (int i = 0; i < objects.size(); ++i) {
            mIdMap.put(objects.get(i), i);
        }
    }

    @Override
    public long getItemId(int position) {
        String item = getItem(position);
        return mIdMap.get(item);
    }

    @Override
    public boolean hasStableIds() {
        return true;
    }
}
```


Problem Info Activity

Activity_problem_info.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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="com.example.sotiris.cityfix.ProblemInfoActivity"
    android:background="#c7eef0">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/step_4"
        android:id="@+id/textView20"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity5_location"
        android:id="@+id/textView21"
        android:layout_below="@+id/textView20"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="45dp"
        android:textSize="18dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity5_additional_location_info"
        android:id="@+id/textView22"
        android:layout_below="@+id/textView21"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:textSize="18dp" />
</RelativeLayout>
```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity5_select_problem_category"
    android:id="@+id/textView23"
    android:layout_below="@+id/textView22"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp"
    android:textIsSelectable="false" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity5_provide_problem_info"
    android:id="@+id/textView24"
    android:layout_below="@+id/textView23"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity5_message"
    android:id="@+id/textView25"
    android:layout_marginTop="28dp"
    android:textAlignment="center"
    android:textSize="18dp"
    android:layout_below="@+id/textView24"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/activity5_next"
    android:id="@+id/button9"
    android:clickable="true"
    android:onClick="onClickNext"
    android:background="#5b82cb"
    android:layout_alignParentBottom="true"
    android:layout_toLeftOf="@+id/checkbox6"
    android:layout_toStartOf="@+id/checkbox6" />

```

```
<EditText
    android:layout_width="350dp"
    android:layout_height="wrap_content"
    android:inputType="textMultiLine"
    android:ems="10"
    android:id="@+id/editText2"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:layout_above="@+id/button9"
    android:layout_below="@+id/textView25" />
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox4"
    android:checked="true"
    android:layout_alignTop="@+id/textView21"
    android:layout_alignRight="@+id/textView25"
    android:layout_alignEnd="@+id/textView25"
    android:clickable="false"/>
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox5"
    android:checked="false"
    android:layout_alignTop="@+id/textView22"
    android:layout_alignLeft="@+id/checkBox4"
    android:layout_alignStart="@+id/checkBox4"
    android:clickable="false"/>
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox6"
    android:checked="true"
    android:layout_below="@+id/textView22"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:clickable="false"/>
```

```
</RelativeLayout>
```

ProblemInfoActivity.java

```
package com.example.sotiris.cityfix;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.content.Intent;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.CheckBox;
import android.widget.EditText;

public class ProblemInfoActivity extends AppCompatActivity {

    EditText ProblemInput;
    public CheckBox LocationInfoBox;
    public String lng;
    public String lat;
    public String ProblemCategory;
    public String input;
    public String Server_ip;
    public String AdditionalProblemInfo;
    public String LocationInfoBoxChecked;
    public Animation animAlpha;

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

        LocationInfoBox = (CheckBox) findViewById(R.id.checkBox5);

        Bundle Step3Info = getIntent().getExtras();

        LocationInfoBoxChecked = Step3Info.getString("LocationInfoBoxChecked");
        input = Step3Info.getString("LocationInformation");
        lng = Step3Info.getString("LongitudeValue");
        lat = Step3Info.getString("LatitudeValue");
        ProblemCategory = Step3Info.getString("ProblemCategory");
        Server_ip = Step3Info.getString("Server_ip");

        ProblemInput = (EditText) findViewById(R.id.editText2);

        if(LocationInfoBoxChecked.equals("true")) {
            LocationInfoBox.setChecked(true);
        }

        animAlpha = AnimationUtils.loadAnimation(this, R.anim.anime_alpha);
    }
}
```

```
public void onClickNext(View view){

    view.startAnimation(animAlpha);

    AdditionalProblemInfo = ProblemInput.getText().toString();

    Intent step5 = new Intent(this, MultimediaActivity.class);

    step5.putExtra("LongitudeValue", lng);
    step5.putExtra("LatitudeValue", lat);
    step5.putExtra("LocationInformation", input);
    step5.putExtra("LocationInfoBoxChecked", LocationInfoBoxChecked);
    step5.putExtra("ProblemCategory", ProblemCategory);
    step5.putExtra("AdditionalProblemInfo", AdditionalProblemInfo);
    step5.putExtra("Server_ip", Server_ip);

    startActivity(step5);

}

}
```

Multimedia Activity

Activity_multimedia.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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="com.example.sotiris.cityfix.MultimediaActivity"
    android:background="#c7eef0">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/step_5"
        android:id="@+id/textView26"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity6_location"
        android:id="@+id/textView27"
        android:layout_below="@+id/textView26"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="45dp"
        android:textSize="18dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/activity6_additional_location_info"
        android:id="@+id/textView28"
        android:layout_below="@+id/textView27"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:textSize="18dp" />
```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity6_select_problem_category"
    android:id="@+id/textView29"
    android:layout_below="@+id/textView28"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity6_provide_problem_info"
    android:id="@+id/textView30"
    android:layout_below="@+id/textView29"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity6_add_a_photo"
    android:id="@+id/textView31"
    android:layout_below="@+id/textView30"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="@string/activity6_message"
    android:id="@+id/textView32"
    android:layout_marginTop="20dp"
    android:textAlignment="center"
    android:layout_below="@+id/textView31"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="18dp" />

```

```
<Button
    android:layout_width="130dp"
    android:layout_height="wrap_content"
    android:text="@string/activity6_photo"
    android:id="@+id/button10"
    android:layout_marginTop="10dp"
    android:background="#5b82cb"
    android:layout_below="@+id/textView32"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox7"
    android:checked="true"
    android:layout_alignTop="@+id/textView27"
    android:layout_alignRight="@+id/textView32"
    android:layout_alignEnd="@+id/textView32"
    android:clickable="false" />
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox8"
    android:checked="false"
    android:layout_below="@+id/textView27"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:clickable="false" />
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox9"
    android:checked="true"
    android:layout_alignTop="@+id/textView29"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:clickable="false" />
```



```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox10"
    android:checked="false"
    android:layout_alignBaseline="@+id/textView30"
    android:layout_alignBottom="@+id/textView30"
    android:layout_alignLeft="@+id/checkBox9"
    android:layout_alignStart="@+id/checkBox9"
    android:clickable="false"/>
```

```
<Button
    android:layout_width="130dp"
    android:layout_height="wrap_content"
    android:text="@string/activity6_send"
    android:id="@+id/button2"
    android:background="#5b82cb"
    android:visibility="invisible"
    android:onClick="onClickSend"
    android:layout_alignTop="@+id/button10"
    android:layout_toLeftOf="@+id/checkBox11"
    android:layout_toStartOf="@+id/checkBox11"
    android:layout_toRightOf="@+id/button10"
    android:layout_marginLeft="30dp"/>
```

```
<CheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/checkBox11"
    android:checked="true"
    android:clickable="false"
    android:layout_below="@+id/checkBox10"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:visibility="invisible" />
```

```
</RelativeLayout>
```

MultimediaActivity.java

```
package com.example.sotiris.cityfix;

import android.app.ProgressDialog;
import android.database.Cursor;
import android.graphics.BitmapFactory;
import android.net.Uri;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Base64;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.CheckBox;

import java.io.IOException;
import java.io.ByteArrayOutputStream;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.provider.MediaStore;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import java.util.Hashtable;
import java.util.Map;
import android.widget.Toast;
import com.android.volley.AuthFailureError;
import com.android.volley.DefaultRetryPolicy;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

public class MultimediaActivity extends AppCompatActivity {

    public CheckBox LocationInfoBox;
    public CheckBox ProblemInfoBox;
    public CheckBox MultimediaBox;
    private Button btnSelect;
    public Button SendButton;
    public Bitmap bm;
    public String lng;
    public String lat;
    public String ProblemCategory;
    public String input;
    public String AdditionalProblemInfo;
    public String LocationInfoBoxChecked;
    public String ProblemInfoBoxChecked = "false";
    private String userChosenTask;
    private String UPLOAD_URL;
    private int REQUEST_CAMERA = 0, SELECT_FILE = 1;
    public Animation animAlpha;
```

```

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

    animAlpha = AnimationUtils.loadAnimation(this, R.anim.anime_alpha);

    LocationInfoBox = (CheckBox) findViewById(R.id.checkBox8);
    ProblemInfoBox = (CheckBox) findViewById(R.id.checkBox10);
    MultimediaBox = (CheckBox) findViewById(R.id.checkBox11);

    Bundle Step4Info = getIntent().getExtras();

    LocationInfoBoxChecked = Step4Info.getString("LocationInfoBoxChecked");

    input = Step4Info.getString("LocationInformation");
    lng = Step4Info.getString("LongitudeValue");
    lat = Step4Info.getString("LatitudeValue");
    ProblemCategory = Step4Info.getString("ProblemCategory");
    AdditionalProblemInfo = Step4Info.getString("AdditionalProblemInfo");
    UPLOAD_URL = Step4Info.getString("Server_ip");

    if(LocationInfoBoxChecked.equals("true")) {
        LocationInfoBox.setChecked(true);
    }

    if(AdditionalProblemInfo.length() != 0) {
        ProblemInfoBox.setChecked(true);
        ProblemInfoBoxChecked = "true";
    }

    btnSelect = (Button) findViewById(R.id.button10);
    btnSelect.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View v) {

            v.startAnimation(animAlpha);
            selectImage();
        }
    });

    SendButton= (Button) findViewById(R.id.button2);
}

```

```

@Override
public void onRequestPermissionsResult(int requestCode, String[] permissions,
int[] grantResults) {
    switch (requestCode) {
        case Utility.MY_PERMISSIONS_REQUEST_READ_EXTERNAL_STORAGE:
            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                if(userChosenTask.equals("Take Photo"))
                    cameraIntent();
                else if(userChosenTask.equals("Choose from Library"))
                    galleryIntent();
            } else {
                //code for deny
            }
            break;
    }
}

private void selectImage() {
    final CharSequence[] items = { getString(R.string.activity6_item1)
,getString(R.string.activity6_item2),getString(R.string.activity6_item3) };

    AlertDialog.Builder builder = new
AlertDialog.Builder(MultimediaActivity.this);
    builder.setTitle(R.string.activity6_dialog_title);
    builder.setItems(items, new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int item) {
            boolean result=Utility.checkPermission(MultimediaActivity.this);

            if (items[item].equals(getString(R.string.activity6_item1))) {
                userChosenTask ="Take Photo";
                if(result)
                    cameraIntent();

            } else if (items[item].equals(getString(R.string.activity6_item2))) {
                userChosenTask ="Choose from Library";
                if(result)
                    galleryIntent();

            } else if (items[item].equals(getString(R.string.activity6_item3))) {
                dialog.dismiss();
            }
        }
    });
    builder.show();
}

private void galleryIntent() {
    Intent intent = new Intent();
    intent.setType("image/*");
    intent.setAction(Intent.ACTION_GET_CONTENT);
    startActivityForResult(Intent.createChooser(intent, "Select
File"),SELECT_FILE);
}

```

```

private void cameraIntent()
{
    Intent intent = new Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
    startActivityForResult(intent, REQUEST_CAMERA);
}

```

```

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (resultCode == Activity.RESULT_OK) {
        if (requestCode == SELECT_FILE)
            onSelectFromGalleryResult(data);
        else if (requestCode == REQUEST_CAMERA)
            onCaptureImageResult(data);
    }
}

```

```

public void onCaptureImageResult(Intent data) {

    Uri selectedImageUri = data.getData();
    System.out.println(selectedImageUri);

    if(selectedImageUri != null) { // Works for most devices

        String selectedImagePath = getRealPathFromURI(selectedImageUri);

        bm = null;
        bm = BitmapFactory.decodeFile(selectedImagePath);
    }else { // Works for some devices

        Bundle extras = data.getExtras();
        bm = (Bitmap) extras.get("data");

    }

    SendButton.setVisibility(View.VISIBLE);
    MultimediaBox.setVisibility(View.VISIBLE);

}

```

```

}

```

```

@SuppressWarnings("deprecation")
public String getRealPathFromURI(Uri contentUri)
{
    try
    {
        String[] proj = {MediaStore.Images.Media.DATA};
        Cursor cursor = managedQuery(contentUri, proj, null, null, null);
        int column_index =
cursor.getColumnIndexOrThrow(MediaStore.Images.Media.DATA);
        cursor.moveToFirst();
        return cursor.getString(column_index);
    }
    catch (Exception e)
    {
        return contentUri.getPath();
    }
}

@SuppressWarnings("deprecation")
public void onSelectFromGalleryResult(Intent data) {

    bm = null;
    if (data != null) {
        try {

            bm =
MediaStore.Images.Media.getBitmap(getApplicationContext().getContentResolver(),
data.getData());

        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    SendButton.setVisibility(View.VISIBLE);
    MultimediaBox.setVisibility(View.VISIBLE);
}

public String getStringImage(Bitmap bmp){
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    bmp.compress(Bitmap.CompressFormat.JPEG, 30, baos);

    byte[] imageBytes = baos.toByteArray();
    String encodedImage = Base64.encodeToString(imageBytes, Base64.DEFAULT);
    return encodedImage;
}

```

```

public void upload(){

    //Showing the progress dialog
    final ProgressDialog loading = ProgressDialog.show(this, "Uploading...", "Please
wait...", false, false);
    final StringRequest stringRequest = new StringRequest(Request.Method.POST,
UPLOAD_URL,
        new Response.Listener<String>() {
            @Override
            public void onResponse(String s) {
                //Dismissing the progress dialog
                loading.dismiss();
                //Showing toast message of the response
                Toast.makeText(MultimediaActivity.this, s ,
Toast.LENGTH_LONG).show();
                Upload_complete_dialog();
            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError volleyError) {
                //Dismissing the progress dialog
                loading.dismiss();

                //Showing toast
                Toast.makeText(MultimediaActivity.this,
volleyError.getMessage().toString(), Toast.LENGTH_LONG).show();
            }
        }) {
            @Override
            protected Map<String, String> getParams() throws AuthFailureError {

                //Converting Bitmap to String
                String image = getStringImage(bm);

                //Creating parameters
                Map<String, String> params = new Hashtable<String, String>();

                // Adding parameters
                params.put("lng", lng);
                params.put("lat", lat);
                params.put("input", input);
                params.put("ProblemCategory", ProblemCategory);
                params.put("AdditionalProblemInfo", AdditionalProblemInfo);
                params.put("Media", image);
            }
        }
}

```

```

        //returning parameters
        return params;
    }
};

    stringRequest.setRetryPolicy(new DefaultRetryPolicy( // Don't
retry uploading. Send only once.
    0,
    DefaultRetryPolicy.DEFAULT_MAX_RETRIES,
    DefaultRetryPolicy.DEFAULT_BACKOFF_MULT));

    //Creating a Request Queue
    RequestQueue requestQueue = Volley.newRequestQueue(this);

    //Adding request to the queue
    requestQueue.add(stringRequest);

}

public void Upload_complete_dialog(){

    DialogFragment newFragment = new Upload_complete_dialog(); //
Show dialog for successful upload
    newFragment.show(getSupportFragmentManager(), "Thanks for using");

}

public void onClickSend(View view){

    view.startAnimation(animAlpha);

    upload();

}

```


Utility.java

```
package com.example.sotiris.cityfix;

import android.Manifest;
import android.annotation.TargetApi;
import android.app.Activity;
import android.content.Context;
import android.content.DialogInterface;
import android.content.pm.PackageManager;
import android.os.Build;
import android.support.v4.app.ActivityCompat;
import android.support.v4.content.ContextCompat;
import android.support.v7.app.AlertDialog;

public class Utility {
    public static final int MY_PERMISSIONS_REQUEST_READ_EXTERNAL_STORAGE = 123;

    @TargetApi (Build.VERSION_CODES.JELLY_BEAN)
    public static boolean checkPermission (final Context context)
    {
        int currentAPIVersion = Build.VERSION.SDK_INT;
        if (currentAPIVersion >= android.os.Build.VERSION_CODES.M)
        {
            if (ContextCompat.checkSelfPermission (context,
Manifest.permission.READ_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED) {
                if (ActivityCompat.shouldShowRequestPermissionRationale ((Activity) context,
Manifest.permission.READ_EXTERNAL_STORAGE)) {
                    AlertDialog.Builder alertBuilder = new AlertDialog.Builder (context);
                    alertBuilder.setCancelable (true);
                    alertBuilder.setTitle ("Permission necessary");
                    alertBuilder.setMessage ("External storage permission is necessary");
                    alertBuilder.setPositiveButton (android.R.string.yes, new
DialogInterface.OnClickListener () {
                        @TargetApi (Build.VERSION_CODES.JELLY_BEAN)
                        public void onClick (DialogInterface dialog, int which) {
                            ActivityCompat.requestPermissions ((Activity) context, new
String [] {Manifest.permission.READ_EXTERNAL_STORAGE},
MY_PERMISSIONS_REQUEST_READ_EXTERNAL_STORAGE);
                        }
                    });
                    AlertDialog alert = alertBuilder.create ();
                    alert.show ();
                } else {
                    ActivityCompat.requestPermissions ((Activity) context, new
String [] {Manifest.permission.READ_EXTERNAL_STORAGE},
MY_PERMISSIONS_REQUEST_READ_EXTERNAL_STORAGE);
                }
                return false;
            } else {
                return true;
            }
        } else {
            return true;
        }
    }
}
```

Upload_complete_dialog.java

```
package com.example.sotiris.cityfix;

import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AlertDialog;

public class Upload_complete_dialog extends DialogFragment {
    @Override

    public Dialog onCreateDialog(Bundle savedInstanceState) {

        AlertDialog.Builder builder = new AlertDialog.Builder(getActivity());
        builder.setTitle(R.string.activity6_upload_dialog_title);
        builder.setMessage(R.string.activity6_upload_dialog_message);

        builder.setNegativeButton(R.string.activity6_upload_dialog_close, new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int id) { //
Exit app.
                Intent homeIntent = new Intent(Intent.ACTION_MAIN);
                homeIntent.addCategory( Intent.CATEGORY_HOME );
                homeIntent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                startActivity(homeIntent);
            }
        });

        return builder.create();
    }
}
```

Resources

anim_alpha.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:interpolator="@android:anim/linear_interpolator" >

    <alpha
        android:duration="200"
        android:fromAlpha="1.0"
        android:repeatCount="1"
        android:repeatMode="reverse"
        android:toAlpha="0.1" />

</set>
```

Colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
</resources>
```

Dimens.xml

```
<resources>
    <!-- Default screen margins, per the Android Design guidelines. -->
    <dimen name="activity_horizontal_margin">16dp</dimen>
    <dimen name="activity_vertical_margin">16dp</dimen>
</resources>
```

Dimens.xml (w820dp)

```
<resources>
  <!-- Example customization of dimensions originally defined in
res/values/dimens.xml
      (such as screen margins) for screens with more than 820dp of available width.
This
      would include 7" and 10" devices in landscape (~960dp and ~1280dp
respectively). -->
  <dimen name="activity_horizontal_margin">64dp</dimen>
</resources>
```

Styles.xml

```
<resources>

  <!-- Base application theme. -->
  <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
    <!-- Customize your theme here. -->
    <item name="colorPrimary">@color/colorPrimary</item>
    <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
    <item name="colorAccent">@color/colorAccent</item>
  </style>

</resources>
```

Cities

--- List of supported cities and the corresponding server ip for each city ---

---- Greek -----

Θεσσαλία

Λάρισα
http://lorem ipsumlorem ipsumlorem ipsum
Τρίκαλα
http://lorem ipsumlorem ipsumlorem ipsum
Καρδίτσα
http://lorem ipsumlorem ipsumlorem ipsum
Βόλος
http://83.212.117.155/CityFix/upload.php

Μακεδονία

Κατερίνη
http://lorem ipsumlorem ipsumlorem ipsum
Θεσσαλονίκη
http://lorem ipsumlorem ipsumlorem ipsum

----- English -----

Thessaly

Larissa
http://lorem ipsumlorem ipsumlorem ipsum
Trikala
http://lorem ipsumlorem ipsumlorem ipsum
Karditsa
http://lorem ipsumlorem ipsumlorem ipsum
Volos
http://83.212.117.155/CityFix/upload.php

Macedonia

Katerini
http://lorem ipsumlorem ipsumlorem ipsum
Thessalonikh
http://lorem ipsumlorem ipsumlorem ipsum
EOF
City not supported

Strings.xml

```
<resources>
  <string name="app_name">CityFix</string>

  <string name="activity1_message1">Don't just see the problem</string>
  <string name="activity1_message2">Report it !!</string>
  <string name="activity1_message3">so someone actually fixes it..</string>
  <string name="activity1_report">REPORT</string>

  <string name="step_1">Step 1 of 5</string>
  <string name="activity2_location">• Location</string>
  <string name="activity2_message">We are using your Network or GPS to determine your
location..</string>
  <string name="activity2_longitude">Longitude :</string>
  <string name="activity2_latitude">Latitude :</string>

  <string name="activity2_not_supported_dialog_title">We are sorry..</string>
  <string name="activity2_not_supported_dialog_message">CityFix is not currently supported in
your city.</string>
  <string name="activity2_not_supported_dialog_cancel">Close app</string>

  <string name="activity2_supported_dialog_title">Location found</string>
  <string name="activity2_supported_dialog_message">Your problem will be reported to the City
Hall of</string>
  <string name="activity2_supported_dialog_ok">Ok</string>

  <string name="activity2_location_null_dialog_title">We are sorry..</string>
  <string name="activity2_location_null_dialog_message">We could not pinpoint your location.Check
your GPS/Internet settings or try again later.</string>
  <string name="activity2_location_null_dialog_cancel">Close app</string>

  <string name="activity2_next">Next</string>

  <string name="step_2">Step 2 of 5</string>
  <string name="activity3_location">• Location</string>
  <string name="activity3_additional_location_info">• Additional Location Info</string>
  <string name="activity3_message">You can provide additional info about your exact location or
simply press \"next\" to skip this step if you don't feel like it.</string>
  <string name="activity3_next">Next</string>

  <string name="step_3">Step 3 of 5</string>
  <string name="activity4_location">• Location</string>
  <string name="activity4_additional_location_info">• Additional Location Info</string>
  <string name="activity4_select_problem_category">• Select Problem Category</string>
  <string name="activity4_message">Choose the problem category from the list below</string>
  <string name="activity4_next">Next</string>
```

```

<string name="step_4">Step 4 of 5</string>
<string name="activity5_location">• Location</string>
<string name="activity5_additional_location_info">• Additional Location Info</string>
<string name="activity5_select_problem_category">• Select Problem Category</string>
<string name="activity5_provide_problem_info">• Provide Problem Info</string>
<string name="activity5_message">Please write a few words to describe the problem or
skip.</string>
<string name="activity5_problem_list_item1">Lighting</string>
<string name="activity5_problem_list_item2">Traffic lights and Road signs</string>
<string name="activity5_problem_list_item3">Streets and Bikeways</string>
<string name="activity5_problem_list_item4">Pavements and Public places</string>
<string name="activity5_problem_list_item5">Parks and Sports grounds</string>
<string name="activity5_problem_list_item6">Trees and Decoration</string>
<string name="activity5_problem_list_item7">Public Transport</string>
<string name="activity5_problem_list_item8">Public Buildings (Schools, City Hall, Fire Department
etc)</string>
<string name="activity5_problem_list_item9">Garbage and Drains</string>
<string name="activity5_problem_list_item10">Abandoned Vehicles</string>
<string name="activity5_problem_list_item11">Other</string>
<string name="activity5_next">Next</string>

<string name="step_5">Step 5 of 5</string>
<string name="activity6_location">• Location</string>
<string name="activity6_additional_location_info">• Additional Location Info</string>
<string name="activity6_select_problem_category">• Select Problem Category</string>
<string name="activity6_provide_problem_info">• Provide Problem Info</string>
<string name="activity6_add_a_photo">• Add a Photo</string>
<string name="activity6_message">Take a photo of the problem now, or choose an image from the
gallery</string>
<string name="activity6_photo">Photo</string>
<string name="activity6_dialog_title">Add Photo!</string>
<string name="activity6_item1">Take Photo</string>
<string name="activity6_item2">Choose from Library</string>
<string name="activity6_item3">Cancel</string>

<string name="activity6_send">Send</string>
<string name="activity6_upload_dialog_title">Sent.</string>
<string name="activity6_upload_dialog_message">Your problem has been successfully uploaded.
Thanks for using CityFix.</string>
<string name="activity6_upload_dialog_close">Close app</string>

</resources>

```

Strings.xml (el)

```
<?xml version="1.0" encoding="utf-8" ?>
<resources>
  <string name="app_name">CityFix</string>
  <string name="activity1_message2">Ανάφερε το !!</string>
  <string name="activity1_message1">Μην κοιτάς απλά το πρόβλημα</string>
  <string name="activity1_message3">έτσι ώστε κάποιος να το φτιάξει..</string>
  <string name="activity1_report">ΑΝΑΦΟΡΑ</string>
  <string name="activity2_latitude">Γεωγραφικό πλάτος :</string>
  <string name="activity2_longitude">Γεωγραφικό μήκος :</string>
  <string name="activity2_location">• Τοποθεσία</string>
  <string name="activity2_message">Χρησιμοποιούμε την σύνδεσή σας στο Ίντερνετ ή το
GPS της συσκευής για να καθορίσουμε την τοποθεσία σας.</string>
  <string name="activity2_next">Επομενο</string>
  <string name="activity3_additional_location_info">• Επιπλέον πληροφορίες
τοποθεσίας</string>
  <string name="activity3_location">• Τοποθεσία</string>
  <string name="activity5_location">• Τοποθεσία</string>
  <string name="activity6_location">• Τοποθεσία</string>
  <string name="activity4_additional_location_info">• Επιπλέον πληροφορίες
τοποθεσίας</string>
  <string name="activity5_additional_location_info">• Επιπλέον πληροφορίες
τοποθεσίας</string>
  <string name="activity6_additional_location_info">• Επιπλέον πληροφορίες
τοποθεσίας</string>
  <string name="activity4_location">• Τοποθεσία</string>
  <string name="activity3_next">Επομενο</string>
  <string name="activity4_next">Επομενο</string>
  <string name="activity5_next">Επομενο</string>
  <string name="activity6_send">Αποστολη</string>
  <string name="activity6_photo">Φωτογραφια</string>
  <string name="activity4_select_problem_category">• Επιλογή κατηγορίας
προβλήματος</string>
  <string name="activity5_select_problem_category">• Επιλογή κατηγορίας
προβλήματος</string>
  <string name="activity6_select_problem_category">• Επιλογή κατηγορίας
προβλήματος</string>
  <string name="activity6_provide_problem_info">• Πληροφορίες προβήματος</string>
  <string name="step_1">Βήμα 1 από 5</string>
  <string name="step_2">Βήμα 2 από 5</string>
  <string name="step_3">Βήμα 3 από 5</string>
  <string name="step_4">Βήμα 4 από 5</string>
  <string name="step_5">Βήμα 5 από 5</string>
  <string name="activity6_message">Τραβήξτε μια φωτογραφία που να απεικονίζει το
πρόβλημα ή διαλέξτε μια ήδη αποθηκευμένη εικόνα.</string>
  <string name="activity6_add_a_photo">• Προσθέστε μια φωτογραφία</string>
  <string name="activity5_provide_problem_info">• Πληροφορίες προβήματος</string>
  <string name="activity5_message">Παρακαλώ περιγράψτε το πρόβλημα εν συντομία ή
απλά πατήστε \ "Επόμενο\ "</string>
```



```

<string name="activity4_message">Διαλέξτε την κατηγορία του προβλήματος από την
παρακάτω λίστα.</string>
<string name="activity3_message">Μπορείτε να παρέχετε επιπλέον πληροφορίες για την
τοποθεσία σας ή απλά πατήστε \ "Επόμενο\ " αν δεν το επιθυμείτε.</string>
<string name="activity2_not_supported_dialog_cancel">Κλείσιμο εφαρμογής</string>
<string name="activity2_not_supported_dialog_title">Λυπούμαστε..</string>
<string name="activity2_not_supported_dialog_message">Ο Δήμος σας δεν υποστηρίζει
το CityFix</string>
<string name="activity2_supported_dialog_ok">ΕΝΤΑΞΕΙ</string>
<string name="activity2_supported_dialog_message">Το πρόβλημά σας θα αναφερθεί στο
Δημαρχείο της πόλης</string>
<string name="activity6_dialog_title">Προσθέστε φωτογραφία</string>
<string name="activity6_item1">Φωτογραφία</string>
<string name="activity6_item2">Αποθηκευμένες εικόνες</string>
<string name="activity6_item3">Ακύρωση</string>
<string name="activity6_upload_dialog_close">Κλείσιμο εφαρμογής</string>
<string name="activity6_upload_dialog_title">Εστάλη.</string>
<string name="activity6_upload_dialog_message">Το πρόβλημά σας εστάλη επιτυχώς.
Ευχαριστούμε που χρησιμοποιήσατε το CityFix.</string>
<string name="activity2_location_null_dialog_cancel">Κλείσιμο εφαρμογής</string>
<string name="activity2_location_null_dialog_title">Λυπούμαστε..</string>
<string name="activity2_location_null_dialog_message">Δεν μπορούσαμε να εντοπίσουμε
την τοποθεσία σας. Ελέγξτε τις ρυθμίσεις σας για το GPS/Ιντερνετ ή δοκιμάστε
αργότερα.</string>
<string name="activity5_problem_list_item1">Φωτισμός</string>
<string name="activity5_problem_list_item1">Άλλο </string>
<string name="activity5_problem_list_item2">Φανάρια και σήματα οδικής
κυκλοφορίας</string>
<string name="activity5_problem_list_item3">Δρόμοι και ποδηλατοδρόμοι</string>
<string name="activity5_problem_list_item4">Πεζόδρομοι, πεζοδρόμια και δημόσιοι
χώροι</string>
<string name="activity5_problem_list_item5">Πάρκα και αθλητικοί χώροι</string>
<string name="activity5_problem_list_item6">Δέντρα και διακόσμηση</string>
<string name="activity5_problem_list_item7">Μέσα μαζικής κυκλοφορίας</string>
<string name="activity5_problem_list_item8">Δημόσια κτίρια</string>
<string name="activity5_problem_list_item9">Απορρίμματα και αποχετεύσεις</string>
<string name="activity5_problem_list_item10">Εγκαταλελειμμένα οχήματα</string>
<string name="activity2_supported_dialog_title">Η τοποθεσία εντοπίστηκε</string>
</resources>

```

ΓΙΑ ΤΟ WEB-APP ΣΤΟΝ SERVER.

Στον Apache:

Upload.php

```
<?php
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
define('HOST','localhost');
define('USER','root');
define('PASS','admin');
define('DB','CityFix');

    $long = $_POST['lng'];
    $lati = $_POST['lat'];
    $input = $_POST['input'];
    $ProblemCategory = $_POST['ProblemCategory'];
    $AdditionalProblemInfo = $_POST['AdditionalProblemInfo'];
    $image = $_POST['Media'];

$con = mysqli_connect(HOST,USER,PASS,DB) or die('unable to connect to db');
mysqli_set_charset($con, "utf8");
    $sql = "SELECT id FROM CityFixInformation ORDER BY id ASC";
    $res = mysqli_query($con, $sql);
    $id = 0;

    while ($row = mysqli_fetch_array($res)) {
        $id = $row['id'];
    }
}
```

```
$id=$id+1;
```

```
    $path = "uploads/$id.png";
```

```
    $actualpath = "http://83.212.117.155/CityFix/$path";
```

```
    $sql = "INSERT INTO CityFixInformation (id, longitude, latitude, location_info, problem_category,  
problem_info, Media, Time) VALUES (NULL,' $long ',' $lati ',' $input ',' $ProblemCategory ' ,'  
$AdditionalProblemInfo ',' $actualpath ',CURRENT_TIMESTAMP)";
```

```
    if( mysqli_query($con, $sql)){
```

```
        file_put_contents($path,base64_decode($image));
```

```
        echo "Successfully Uploaded";
```

```
    }
```

```
    mysqli_close($con);
```

```
}
```

```
    else {
```

```
        echo "Error";
```

```
    }
```

```
?>
```

Στον Tomcat:

Results.jsp

<%--

Document : results

Created on : Oct 1, 2016, 4:59:50 PM

Author : sotiris

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="java.sql.ResultSet"%>

<%@ page import="java.io.*,java.util.*,java.sql.*"%>

<%@ page import="javax.servlet.http.*,javax.servlet.*" %>

<%@ page import ="java.sql.*" %>

<%@ page import ="javax.sql.*" %>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>

<!DOCTYPE html>

<html>

<head>

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

<title>JSP Page</title>

</head>

<body>

<h1>Hello World!</h1>

```

<div>
  <center><h2>Get Parameters  </h2></center>
  <p>
    ${requestScope.lng}
    ${requestScope.lat}
    ${requestScope.input}
    ${requestScope.ProblemCategory}
    ${requestScope.AdditionalProblemInfo}

  </p>

  <%
//      String lng = request.getAttribute("lng").toString();
//      String lat = request.getAttribute("lat").toString();
//      String input = request.getAttribute("input").toString();
//      String ProblemCategory = request.getAttribute("ProblemCategory").toString();
//      String AdditionalProblemInfo = request.getAttribute("AdditionalProblemInfo").toString();
//
//      out.println("<p>");
//      out.println(lng + " " + lat + "\n" + input + "\n" + ProblemCategory + "\n" +
AdditionalProblemInfo);
//      out.println("</p>");
  %>
</div>
</body>
</html>

```

Index.jsp

<%--

Document : index

Created on : Oct 1, 2016, 4:12:58 PM

Author : sotiris

--%>

<%@page import="java.sql.ResultSet"%>

<%@ page import="java.io.*,java.util.*,java.sql.*"%>

<%@ page import="javax.servlet.http.*,javax.servlet.*" %>

<%@ page import ="java.sql.*" %>

<%@ page import ="javax.sql.*" %>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>

<jsp:include page="/do.DBconnect" />

<%--<jsp:include page="/do.Fetch" />--%>

<% ResultSet rs = (ResultSet) request.getAttribute("rs");%>

<!--αρκεί να αλλάξει κανείς το url της βάσης, το username και το password και τρέχει κανονικά σε κάθε βάση-->

<sql:setDataSource var="snapshot" driver="com.mysql.jdbc.Driver"

url="jdbc:mysql://localhost:3306/CityFix"

user="root" password="admin"/>

```
<sql:query var="info" dataSource="{snapshot}">
```

```
SELECT * FROM CityFixInformation;
```

```
</sql:query>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

```
<link rel="icon"
```

```
type="image/png"
```

```
href="/uth_logo.png ">
```

```
<title>City Fix - Web Interface</title>
```

```
<link href="css/hover.css" rel="stylesheet" media="all">
```

```
<link rel='icon' href='images/marker.png' type='image/x-icon' />
```

```
<link rel="stylesheet" type="text/css" href="css/main.css"/>
```

```
<link href="css/bootstrap-theme.css" rel="stylesheet" type="text/css"/>
```

```
<link href="css/bootstrap.css" rel="stylesheet" type="text/css"/>
```

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.css">
```

```
<script src="js/jquery-1.11.3.min.js" type="text/javascript"></script>
```

```
<script src="js/bootstrap.js" type="text/javascript" ></script>
```

```
</head>
```

```

<body>

<center><h1>City Fix - Volos</h1></center>

<center></center>

<div class="container">

  <center><h2>Issues Table</h2></center>

<div class="container">

  <table class="table table-striped table-bordered table-hover ">

    <thead>

      <tr>

        <th>id</th>

        <th>Long</th>

        <th>Lat</th>

        <th>Location Info</th>

        <th>Problem Category</th>

        <th>Problem Info</th>

        <th>Media</th>

        <th>TimeStamp</th>

      </tr>

    </thead>

    <tbody>

      <%

        while (rs.next()) {

          out.println("<tr>");

          //Retrieve by column name

          int id = rs.getInt("id");

          float latitude = rs.getFloat("latitude");

```



```

float longitude = rs.getFloat("longitude");

String location_info = rs.getString("location_info");

String problem_category = rs.getString("problem_category");

String problem_info = rs.getString("problem_info");

String media = rs.getString("Media");

String media_string = media.substring(media.lastIndexOf("/") + 1);

String time = rs.getString("Time");

//Display values

out.println("<td>" + id + "</td>");

out.println("<td>" + latitude + "</td>");

out.println("<td>" + longitude + "</td>");

out.println("<td>" + location_info + "</td>");

out.println("<td>" + problem_category + "</td>");

out.println("<td>" + problem_info + "</td>");

//
out.println("<td>" + media + "</td>");

out.println("<td><a href=" + media + "target='_blank'" + media_string + "</a> </td>");

out.println("<td>" + time + "</td>");

out.println("</tr>");

}

%>

</tbody>

</table>

</div>

</body>

</html>

```

Σημείωση: Το web-app στον Tomcat είναι ένα Java-web-app. Τυπικά θα έπρεπε να παρουσιάσω ένα προς ένα τα περιεχόμενα του .war αρχείου, αλλά καθώς τα περισσότερα από αυτά είναι auto-generated και αφορούν κατά κύριο λόγο configurations , επέλεξα να παρουσιάσω μόνο τα δύο (2) παραπάνω αρχεία τα οποία ενθυλακώνουν όλη την απαιτούμενη λειτουργικότητα για την ανάκτηση των δεδομένων από την βάση και την παρουσίαση τους.

Συμπεράσματα και μελλοντικές επεκτάσεις

Δύο σημεία με ενδιαφέρον

Τοποθεσία

Για την εύρεση της τοποθεσίας του χρήστη μπορούμε να χρησιμοποιήσουμε είτε την στρατηγική του GPS Positioning είτε την στρατηγική του Cell Tower/ WiFi Positioning. Κάθε μία έχει τα πλεονεκτήματα και τα μειονεκτήματά της.

- Το GPS positioning χρησιμοποιεί δορυφόρους για να καθορίσει την θέση του χρήστη. Τυπικά πρέπει να επικοινωνήσει με τρεις (3) δορυφόρους και κατόπιν σχετικών υπολογισμών προκύπτει η θέση του χρήστη στον χάρτη. Χαρακτηρίζεται από μεγάλη ακρίβεια εντοπισμού θέσης, όμως είναι η πιο αργή στρατηγική. Ακόμη, επειδή απαιτεί «line of sight» με τους δορυφόρους, δεν μπορεί να χρησιμοποιηθεί σε κλειστούς χώρους ή ακόμη και σε εξωτερικούς χώρους όταν ψηλά κτίρια ή άλλα αντικείμενα μπλοκάρουν την επαφή της συσκευής με τους δορυφόρους. Προϋπόθεση για την χρησιμοποίηση αυτής της στρατηγικής είναι φυσικά η συσκευή να διαθέτει τεχνολογία GPS. Κάθε εφαρμογή που θέλει να κάνει χρήση αυτής της στρατηγικής πρέπει να ζητά την ακόλουθη άδεια από τον χρήστη μέσα στο manifest.xml:

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

- Το WiFi positioning και το Cell Tower positioning είναι δύο αυτοτελείς στρατηγικές εντοπισμού θέσης αλλά καθώς μπορούν να χρησιμοποιηθούν συνεργατικά, αντιμετωπίζονται εδώ ως ένα. Αντίστοιχα με το GPS positioning προσπαθούν να καθορίσουν την θέση του χρήστη στον χάρτη μέσω επικοινωνίας με cell towers και δίκτυα WiFi των οποίων η γεωγραφική θέση είναι καθορισμένη και σταθερή. Η επιστροφή της θέσης του χρήστη γίνεται πολύ πιο γρήγορα αλλά σε μερικές περιπτώσεις δεν έχει την ίδια ακρίβεια σε σχέση με το GPS positioning. Μπορεί να χρησιμοποιηθεί τόσο σε κλειστούς όσο και σε ανοιχτούς χώρους και αρκεί να υπάρχει σύνδεση στο ίντερνετ ώστε να μπορεί κανείς να την χρησιμοποιήσει. Κάθε εφαρμογή που θέλει να κάνει χρήση αυτής της στρατηγικής πρέπει να ζητά την ακόλουθη άδεια από τον χρήστη μέσα στο manifest.xml:

```
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
```

Έχοντας λάβει αυτά υπόψη, η εφαρμογή CityFix προσπαθεί αρχικά να χρησιμοποιήσει Wifi και Cell Tower positioning για τον καθορισμό της θέσης του χρήστη λόγω της ταχύτητας της εν λόγω στρατηγικής, και μόνο σε περίπτωση που αυτό αποτύχει, δοκιμάζει να κάνει GPS Positioning εάν η συσκευή του χρήστη διαθέτει τεχνολογία GPS .

HTTP μηνύματα

Για την διαχείριση των HTTP μηνυμάτων από και προς τον server χρησιμοποιήθηκε το «Android Volley»^[22]. Το Android Volley είναι μια HTTP βιβλιοθήκη που αναπτύχθηκε από την Google το 2013 για να αντικαταστήσει τα «`java.net.HttpURLConnection`» και «`org.apache.http.client`» που χρησιμοποιούνταν ως τότε κατά κόρον. Αφαιρεί μεγάλο μέρος της πολυπλοκότητας γύρω από την σύνταξη, την διαχείριση και την προτεραιότητα των HTTP αιτημάτων επιτρέποντας στον προγραμματιστή να επικεντρωθεί στην λειτουργικότητα της εφαρμογής του. Ακόμη, είναι εξαιρετικά γρήγορο και καθώς κάνει «Cache» τα αιτήματα και τις απαντήσεις του server δεν προκαλεί μεγάλο φόρτο στο δίκτυο.^[23]

Μελλοντικές επεκτάσεις

Στην παράγραφο αυτή παρουσιάζονται λειτουργικότητες που αυτήν την στιγμή δεν υποστηρίζονται από την εφαρμογή αλλά θα μπορούσαν μελλοντικά να συμπεριληφθούν καθώς και λειτουργικότητες που ήδη υποστηρίζονται αλλά θα μπορούσαν να επεκταθούν ακόμη περισσότερο.

Κατηγορίες προβλημάτων

Όπως αναφέρθηκε προηγουμένως, το CityFix υποστηρίζει τις εξής κατηγορίες προβλημάτων όπως αυτές προέκυψαν κατόπιν επικοινωνίας με τον Δήμο Κατερίνης :

- Φωτισμός
- Φανάρια και σήματα οδικής κυκλοφορίας
- Δρόμοι και ποδηλατοδρόμοι
- Πεζόδρομοι, πεζοδρόμια και δημόσιοι χώροι
- Πάρκα και αθλητικοί χώροι
- Δέντρα και διακόσμηση
- Μέσα μαζικής κυκλοφορίας
- Δημόσια κτίρια
- Απορρίμματα και αποχετεύσεις
- Εγκαταλελειμμένα οχήματα
- Άλλα

Κρίνεται σημαντικό οι κατηγορίες αυτές να επαυξηθούν με ακόμη περισσότερες. Οι κυρίαρχοι τρόποι με τους οποίους μπορούν να αναζητηθούν νέες κατηγορίες προβλημάτων είναι :

- Research
- Επικοινωνία με περισσότερους Δήμους
- Feedback από τους χρήστες

- Ανάλυση των αναφορών που ως κατηγορία προβλήματος έχουν επιλέξει «Άλλο» για τον εντοπισμό επαναλαμβανόμενων μοτίβων που μπορούν να οδηγήσουν στην δημιουργία νέων κατηγοριών

Κατά αυτόν τον τρόπο η κατηγορία «Άλλο» μπορεί να αναλυθεί σε μια σειρά νέων κατηγοριών αρκεί αυτές να είναι καλά ορισμένες και μη-επικαλυπτόμενες. Κάτι τέτοιο θα ήταν εξαιρετικά χρήσιμο στους εργαζόμενους των Δήμων για τους λόγους που αναφέρθηκαν κατά την παρουσίαση της δραστηριότητας επιλογής προβλήματος.

Ενσωμάτωση νέων Δήμων

Η ενσωμάτωση ενός νέου Δήμου στο CityFix γίνεται μέσα από δύο (2) απλά βήματα. Πρώτα, πρέπει ο ενδιαφερόμενος Δήμος να εγκαταστήσει το web-app της εφαρμογής σε κάποιον ιδιόκτητο server και κατόπιν να παρέχει την ip-διεύθυνση αυτού του server στον διαχειριστή/ιδιοκτήτη της εφαρμογής CityFix ώστε να ενσωματωθεί στην λίστα των υποστηριζόμενων Δήμων. Σε αυτό το σημείο οι δημότες του εν λόγω Δήμου μπορούν να κατεβάσουν και να ξεκινήσουν να χρησιμοποιούν την εφαρμογή ή εάν την είχαν ήδη κατεβασμένη, θα πρέπει να κατεβάσουν εκ νέου την νέα έκδοση που συμπεριλαμβάνει τον Δήμο τους. Ο λόγος για τον οποίον πρέπει να ξανακατεβάσουν την εφαρμογή έχει να κάνει με τρόπο που η εφαρμογή ενημερώνεται για τους νεοεισελθόντες Δήμους.

Η λίστα των υποστηριζόμενων Δήμων βρίσκεται σε ένα .txt αρχείο που είναι μέρος της android εφαρμογής CityFix. Αυτό σημαίνει πως όποτε προστίθεται ένας νέος Δήμος, στην ουσία δημιουργείται καινούριο update της εφαρμογής. Αυτή η σχεδιαστική επιλογή έχει κάποια πλεονεκτήματα καθώς και κάποια μειονεκτήματα.

Βασικό πλεονέκτημα αυτής της επιλογής είναι πως δεν χρειάζεται να υπάρχει κάποιος κεντρικός server για την εφαρμογή παρά μόνο οι επιμέρους server του κάθε Δήμου.

Το κύριο μειονέκτημα είναι πως κάθε φορά που προστίθεται ένας νέος Δήμος, οι χρήστες της εφαρμογής που κατοικούν σε άλλους Δήμους βλέπουν πως υπάρχει νέο update για την εφαρμογή ακόμη κι εάν αυτό το update δεν έχει τίποτε το καινούριο να προσφέρει σε αυτούς. Μπορούν να αγνοήσουν το update και να συνεχίσουν να χρησιμοποιούν κανονικά την εφαρμογή εντός των ορίων του Δήμου τους. Εναλλακτικά, μπορούν να κατεβάσουν την ανανεωμένη έκδοση για να έχουν την δυνατότητα να αναφέρουν προβλήματα του νεοεισελθόντα Δήμου όποτε βρίσκονται εντός των ορίων αυτού.

Υποστήριξη βίντεο

Όπως αναφέρθηκε στην παρουσίαση της δραστηριότητας πολυμέσων, οι χρήστες σε αυτήν την δραστηριότητα καλούνται να επιλέξουν μια φωτογραφία που να καταδεικνύει το πρόβλημα. Μελλοντικά θα μπορούσε να υποστηρίζεται και η επιλογή ενός βίντεο στην θέση της φωτογραφίας για περιπτώσεις όπου η φύση του προβλήματος είναι τέτοια που να μην μπορεί να αποτυπωθεί σε μια φωτογραφία. Ένα ενδεικτικό παράδειγμα ενός τέτοιου προβλήματος θα μπορούσε να είναι μια χαλασμένη λάμπα φωτισμού που «πετάει σπίθες». Το να αποτυπωθεί σε αυτή η συμπεριφορά σε μια φωτογραφία είναι από απίθανο έως αδύνατο, ενώ με χρήση βίντεο είναι αρκετά εύκολο.

Ενσωμάτωση ΔΕΚΟ

Όλες οι κατηγορίες προβλημάτων που υποστηρίζει αυτήν την στιγμή το CityFix αφορούν αρμοδιότητες ενός Δήμου. Μελλοντικά θα μπορούσε να προστεθεί και υποστήριξη για προβλήματα που είναι αρμοδιότητα διάφορων δημόσιων επιχειρήσεων και επιχειρήσεων όπως για παράδειγμα η ΔΕΗ ή η επιχείρηση ύδρευσης.

Δημιουργία έκδοσης του CityFix για τους εργαζομένους του Δήμου

Θα μπορούσε να δημιουργηθεί ξεχωριστή έκδοση του CityFix για τους εργαζομένους του Δήμου. Έτσι ο υπάλληλος που διαχειρίζεται το web-app στον server του Δήμου και δέχεται τις αναφορές των χρηστών θα μπορούσε να αναθέσει τις επιδιορθώσεις των προβλημάτων στους διάφορους εργαζομένους οι οποίοι θα βλέπουν στην συσκευή τους τα προβλήματα που πρέπει οι ίδιοι να φροντίσουν να επιδιορθώσουν. Θα ήταν πολύ εύκολο, αφού υπάρχουν οι συντεταγμένες του προβλήματος, η εφαρμογή να συνεργάζεται με τους χάρτες της Google ώστε ο εργαζόμενος να μπορεί να οργανώσει την σειρά με την οποία θα επιδιορθώσει τα προβλήματα ώστε να είναι αποδοτικότερος καθώς και να οδηγηθεί εύκολα στην ζητούμενη τοποθεσία με το navigator.

Μόλις το εκάστοτε πρόβλημα έχει επιδιορθωθεί, ο εργαζόμενος μπορεί να το ανακοινώνει στον κεντρικό server του Δήμου για να σημειωθεί πλέον ως επιδιορθωμένο ή ακόμη και να αφαιρείται από την βάση.

Δυνατότητα δια-Δημοτικής επικοινωνίας και επαναπρόσβασης αναφοράς.

Σε σπάνιες περιπτώσεις είναι θεωρητικά δυνατό ένας Δήμος να λάβει μια αναφορά για κάποιο πρόβλημα που βρίσκεται εντός των ορίων άλλου, γειτονικού Δήμου. Για να γίνει αυτό πιο κατανοητό ας πάρουμε για παράδειγμα μια από τις δύο μεγαλουπόλεις της χώρας μας, την Αθήνα ή την Θεσσαλονίκη. Σε τέτοιες πόλεις υπάρχουν πολλοί Δήμοι ο ένας πλάι στον άλλο που σπανίως χωρίζονται από κάποιο φυσικό σύνορο ή από μεγάλη απόσταση. Πλησίον λοιπόν των «συνόρων» των Δήμων αυτών μπορεί να παρατηρηθεί το φαινόμενο ο χρήστης να στέκεται εντός των ορίων του Δήμου Α όμως η συσκευή λανθασμένα να αποφανθεί πως βρίσκεται στον Δήμο Β και να στείλει εκεί την αναφορά. Αυτό οφείλεται στην περιορισμένη ακρίβεια των τεχνικών καθορισμού θέσης είτε πρόκειται για GPS, είτε για cell tower triangulation είτε τέλος για Wi-Fi positioning.

Σε αυτήν την περίπτωση θα ήταν χρήσιμο ο εργαζόμενος που συνειδητοποιεί πως το πρόβλημα ανήκει στα όρια άλλου Δήμου να ενημερώνει τον κεντρικό server του Δήμου του και αυτός με την σειρά του να προωθεί την αναφορά στον κατάλληλο Δήμο.

Ασφάλεια

Ζητήματα ασφάλειας ήταν πέρα και έξω από το score της παρούσας διπλωματικής εργασίας. Η εφαρμογή CityFix δεν απαιτεί εγγραφή και δεν ζητά από τους χρήστες προσωπικά δεδομένα όπως ηλεκτρονική ή φυσική διεύθυνση, όνομα και άλλα. Δεν υπάρχει σύστημα χρηστών και στα μηνύματα ανάμεσα στην συσκευή και τον Server δεν υπάρχουν ευαίσθητα προσωπικά δεδομένα. Ακόμη κι αν κάποιος έπαιρνε στην κατοχή του όλα τα περιεχόμενα της βάσης δεδομένων του server, δεν θα έβλεπε τίποτε περισσότερο από εγγραφές αποτελούμενες από ένα αναγνωριστικό, τις συντεταγμένες του προβλήματος, την κατηγορία και μια φωτογραφία του και ίσως κάποιο κείμενο γραμμένο από τον χρήστη.

Παρόλα αυτά, μελλοντικά θα μπορούσαν να γίνουν κάποια βήματα σε αυτόν τον τομέα έστω κι αν αυτό είναι μια απλή κρυπτογράφηση (md5 κτλ) των HTTP μηνυμάτων ανάμεσα στην εφαρμογή και τον server.

Βιβλιογραφία

- [1]. Android (operating system)
[https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))
- [2]. Android version history
https://en.wikipedia.org/wiki/Android_version_history#Android_7.0_Nougat_.28API_level_24.29
- [3]. Java vs. Android APIs
<http://www.zdnet.com/article/java-vs-android-apis/>
- [4]. Comparison of Java and Android API
https://en.wikipedia.org/wiki/Comparison_of_Java_and_Android_API
- [5]. Dalvik (software)
[https://en.wikipedia.org/wiki/Dalvik_\(software\)](https://en.wikipedia.org/wiki/Dalvik_(software))
- [6]. Android Runtime
https://en.wikipedia.org/wiki/Android_Runtime
- [7]. ART and Dalvik
<https://source.android.com/devices/tech/dalvik/>
- [8]. Android Developers
<https://developer.android.com/index.html>
- [9]. SQL
<https://en.wikipedia.org/wiki/SQL>
- [10]. MySQL
<https://en.wikipedia.org/wiki/MySQL>
- [11]. phpMyAdmin
<https://en.wikipedia.org/wiki/PhpMyAdmin>
- [12]. Apache HTTP Server
https://en.wikipedia.org/wiki/Apache_HTTP_Server
- [13]. Apache Tomcat
https://en.wikipedia.org/wiki/Apache_Tomcat

- [14]. PHP
<https://en.wikipedia.org/wiki/PHP>
- [15]. Java
[https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))
- [16]. XML
<https://en.wikipedia.org/wiki/XML>
- [17]. Android Studio
https://en.wikipedia.org/wiki/Android_Studio
- [18]. NetBeans
<https://en.wikipedia.org/wiki/NetBeans>
- [19]. PuTTY
<https://en.wikipedia.org/wiki/PuTTY>
- [20]. Filezilla
<https://en.wikipedia.org/wiki/FileZilla>
- [21]. Android Emulator
<https://developer.android.com/studio/run/emulator.html>
- [22]. Transmitting Network Data Using Volley
<https://developer.android.com/training/volley/index.html>
- [23]. An Introduction to Volley
<https://code.tutsplus.com/tutorials/an-introduction-to-volley--cms-23800>