Around Me!

A Mini-Project Report
Under
Project Workshop

Submitted by

Rijul Luman (B051)
Apoorva Mehta (B054)

Under The Guidance Of

Prof. Supriya Agarwal

In partial fulfillment for the award of the degree
Of

Bachelor of Technology
IN
Computer Engineering

At
NMIMS' MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT AND ENGINEERING, MUMBAI

April 2014

CERTIFICATE

This is to certify that the project entitled "Around Me!" is the bonafide work carried out by Rijul Luman and Apoorva Mehta of B.Tech (Computer Engineering), MPSTME (NMIMS), Mumbai, during the VI semester of the academic year 2013-14, in partial fulfillment of the requirements for the award of the Degree of Bachelors of Technology as per the norms prescribed by NMIMS. The mini-project work has been assessed and found to be satisfactory.

Prof. Supriya Agarwal

Internal Mentor

Examiner 1

Examiner 2

Dean

Dr. S. Y. Mhaiskar

DECLARATION

We, Rijul Luman and Apoorva Mehta, Roll No. B051 and B054, B.Tech (Computer Engineering), VI semester understand that plagiarism is defined as anyone or combination of the following:

1. Un-credited verbatim copying of individual sentences, paragraphs or illustration (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.

2. Un-credited improper paraphrasing of pages paragraphs (changing a few words phrases, or rearranging the original sentence order)

3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did wrote what. (Source: IEEE, The institute, Dec. 2004)

4. I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

5. I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the seminar/ project report may not be in a position to check for the possibility of such incidences of plagiarism in this body of work.

Signature of the Student

Name: Rijul Luman

Roll No. B051

Place: Mumbai

Date:

Signature of the Student

Name: Apoorva Mehta

Roll No. B054

Place: Mumbai

Date:

Table of contents

CHAPTER NO.	TITLE	PAGE NO.
	List of Figures	i
	Abbreviations	iii
	Abstract	ii
1.	INTRODUCTION	
	1.1 Project Overview	1
	1.2 Hardware Specification	1
	1.3 Software Specification	1
2.	INSTALLATION & SETUP	
	2.1 Installing ADT	3
	2.2 Set Up	6
3.	ANALYSIS & DESIGN	
	3.1 Analysis and design	10
	3.2 Analysis Models	15
4.	PROJECT IN DETAIL	
	4.1 Splash Screen	17
	4.2 Main Screen	18
	4.3 Nearby Places	19
	4.4 Stations	22
	4.5 Emergency	24
5.	CONCLUSION & FUTURE SCOPE	
	5.1 Conclusion	25
	5.2 Future Scope	25

6. REFERENCES 26

List of Figures

CHAPTER NO.	TITLE	PAGE NO.
2.	INSTALLATION & SETUP	
	Fig2.1 ADT bundle	3
	Fig2.2 SDK Manager after loading the required packages	4
	Fig2.3 Path followed to start Android Virtual Device Manager	4
	Fig2.4 Creation of Virtual Device	5
	Fig2.5 Emulator home screen	6
	Fig2.6 Specifying the workspace	6
	Fig2.7 Creating a new android project	7
	Fig2.8 Default code screen for new project	8
	Fig2.9 Default graphical layout of project	8
	Fig2.10 Project running on emulator	9
3.	ANALYSIS AND DESIGN	
	Fig3.1 Use case for the Application	12
	Fig3.2 Sequence diagram for Find Places	15
	Fig3.3 Sequence Diagram for handling the At the station requests	15
	Fig3.4 Sequence diagram for Emergency Calls	16
	Fig3.5 DFD for the system	16
4.	PROJECT IN DETAIL	
	Fig4.1: Splash Screen	17
	Fig4.2: Main screen/Home screen	18
	Fig4.3: Loading screen	19
	Fig4.4: Search screen	19
	Fig4.5 Search result for ATM	20

Fig4.6 Hospitals searched on map 21
Fig4.7 Station List 22
Fig4.8 Selected station on map 23
Fig4.9 Malls displayed near the station 23
Fig4.10 Emergency screen 24
i
Fig4.11 Calling Police 24

Abbreviations

Abbreviation	Description
API	Application Programing Interface
HTTP	Hyper Text Transfer Protocol
GUI	Graphical User Interface
App	Android Application
WiFi	Wireless Fidelity
GPS	Global Positioning System

Abstract

Our project is an Android Application called Around Me! This application is designed for everyday use for everyone. This application will help the user to locate nearby destinations like ATM's, Malls, etc. The user will be able to search places based on key words too. Apart from the received information in list format, the user can also view these destinations on Google maps which have been implemented.

Around Me shall provide user the ability to find ATM's, Malls and restaurants near stations of the Mumbai local. These would be displayed on the map so that the user can estimate the distance. This app will be developed for the convenience of people travelling by the crowded trains so that at a button press the nearby places would be shown.

There is an added functionality of calling emergency numbers at the press of a button. This will enable the user to contact the police or ambulance helpline in case of emergency since while travelling it might be needed.

1. INTRODUCTION

1.1 Project Overview

This software system will be an Android Application for the use of every android user to find the different places around. The application shall provide the user the ability to find places like ATM's, Malls, Hospitals, etc. The app will be set up in a user friendly interface. This app will be developed for everyone as all of us require to locate the presence of places around us for day to day purposes.

This is a relatively new concept of integrating local stations already as well as providing emergency features. There are a few simple applications available on the android market but most of them are integrated with the localized version for Mumbai as well as provisions for emergency dial. Our aim is to create a free and useful app for locating nearby places. It will be available for everyone with Android 2.3. (Gingerbread) or higher; with all basic requirements fulfilled by every mobile phone (RAM, memory, etc.)

This is a very small application that takes less than 4.45 Megabytes when installed on the device. Also it is not a processor intensive application but uses internet and thus can be used by any person with any android device no matter what its hardware specifications as long as it is running on Android 2.3.5 (Honeycomb) or higher with a good speed internet connectivity preferably WiFi.

There will most probably be updates available to the App changing the user interface as well as to support some devices and also to some bugs if they are brought to notice.

1.2 Hardware Requirements

Recommend 2GB memory for IDE and 250MB RAM for the device.

1.3 Software Requirements

Operating Systems

- Windows XP (32-bit), Vista (32- or 64-bit), or Windows 7 (32- or 64-bit)

- Mac OS X 10.5.8 or later (x86 only)
- Linux (Ubuntu Linux, Lucid Lynx)
- GNU C Library (glibc) 2.7 or later is required.
- On Ubuntu Linux, version 8.04 or later is required.
- 64-bit distributions must be capable of running 32-bit applications.

Eclipse IDE

- Eclipse 3.7.2 (Indigo) or greater

Note: Eclipse 3.6 (Helios) is no longer supported with the latest version of ADT.

- Eclipse JDT plugin (included in most Eclipse IDE packages)
- JDK 6 (JRE alone is not sufficient)
- Android Development Tools plugin (recommended)
- Not compatible with GNU Compiler for Java (gcj)

Other development environments

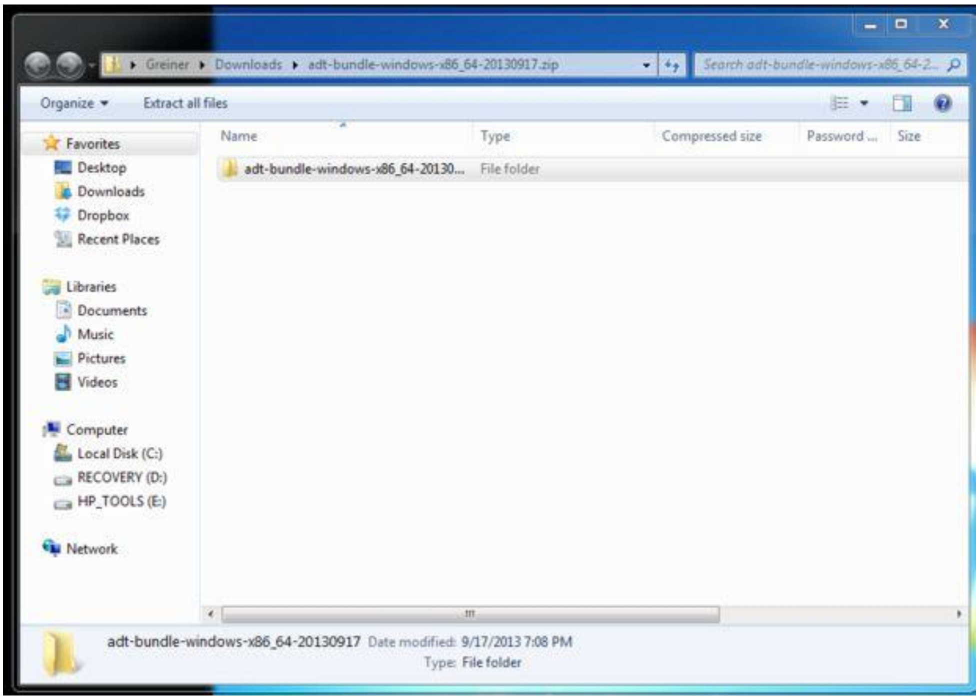
- JDK 6 (JRE alone is not sufficient)
- Apache Ant 1.8 or later
- Not compatible with Gnu Compiler for Java (gcj)

Note: Some Linux distributions may include JDK 1.4 or Gnu Compiler for Java, both of which are not supported for Android development.

2. INSTALLATION AND SETUP

2.1 Installing ADT Bundle (Android SDK + Eclipse IDE)

1. First run the SDK Setup File

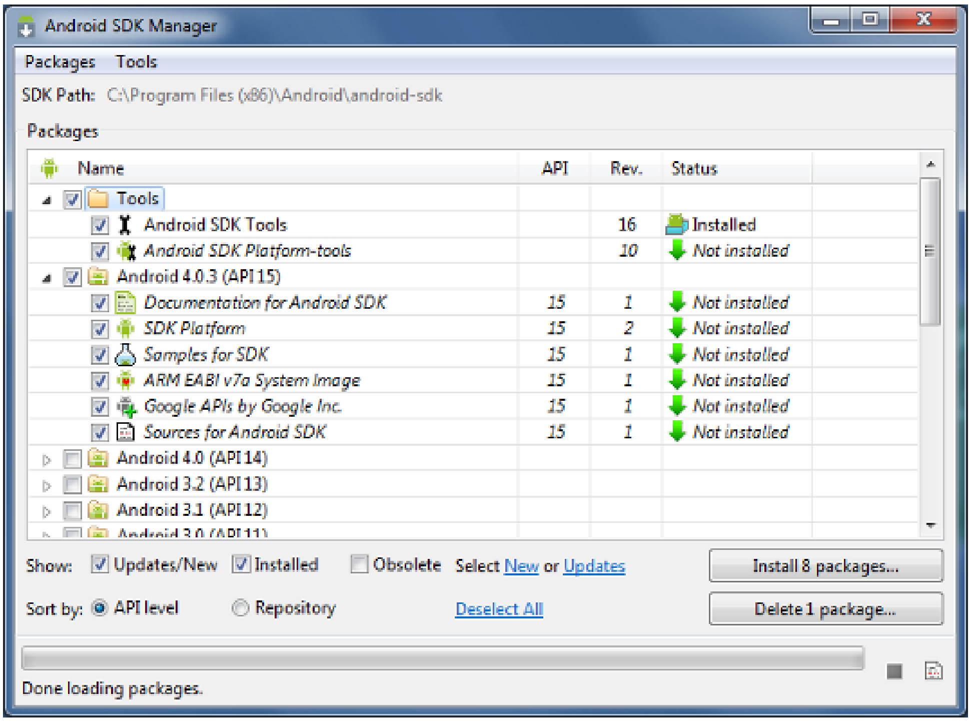


Name	Type	Compressed size	Password ...	Size
adt-bundle-windows-x86_64-20130...	File folder			

adt-bundle-windows-x86_64-20130917 Date modified: 9/17/2013 7:08 PM
Type: File folder

Fig2.1:Adt Bundel

2. Then open the Android SDK Manager and install the packages that you require (like Android Documentation, Samples for various android versions, SDK platform, etc.)



Android SDK Manager

Packages Tools

SDK Path: C:\Program Files (x86)\Android\android-sdk

Packages

Name	API	Rev.	Status
☑ Tools			
☑ Android SDK Tools		16	Installed
☑ *Android SDK Platform-tools*		*10*	*Not installed*
☑ Android 4.0.3 (API 15)			
☑ *Documentation for Android SDK*	*15*	*1*	*Not installed*
☑ *SDK Platform*	*15*	*2*	*Not installed*
☑ *Samples for SDK*	*15*	*1*	*Not installed*
☑ *ARM EABI v7a System Image*	*15*	*1*	*Not installed*
☑ *Google APIs by Google Inc.*	*15*	*1*	*Not installed*
☑ *Sources for Android SDK*	*15*	*1*	*Not installed*
□ Android 4.0 (API 14)			
□ Android 3.2 (API 13)			
□ Android 3.1 (API 12)			
□ Android 3.0 (API 11)			

Show: ☑ Updates/New ☑ Installed □ Obsolete Select New or Updates [Install 8 packages...]

Sort by: ● API level ○ Repository Deselect All [Delete 1 package...]

Done loading packages.

Fig2.2:SDK Manager after loading the required packages

After this open Eclipse (it will have automatically been installed when you unpack the ADT bundle) and it will automatically load in the ADT mode.

3.Setting up the Emulator

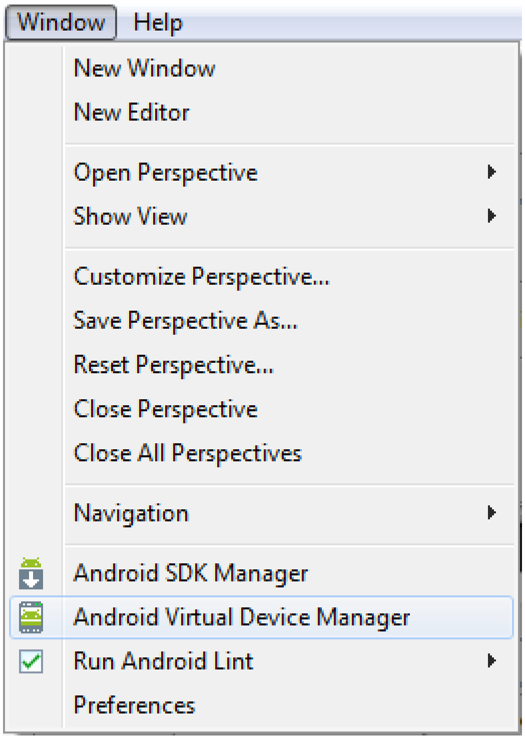
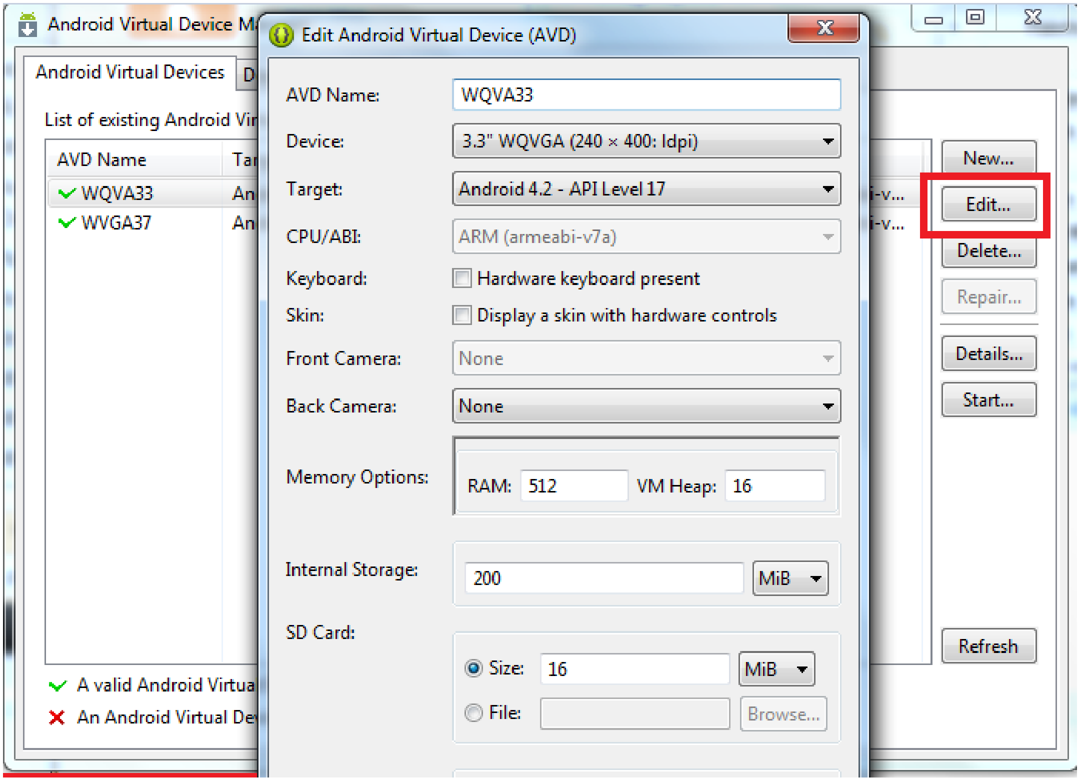


Fig2.3: Path followed to start Android Virtual Device Manager

4. Select the appropriate configurations and run the emulator.



AVD Name: WQVA33

Device: 3.3" WQVGA (240 × 400: ldpi)

Target: Android 4.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: □ Hardware keyboard present

Skin: □ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 512 VM Heap: 16

Internal Storage: 200 MiB

SD Card: ● Size: 16 MiB

○ File: Browse...

Fig2.4: Creation of Virtual Device

5.After successful setting up the emulator runs as an active android platform.

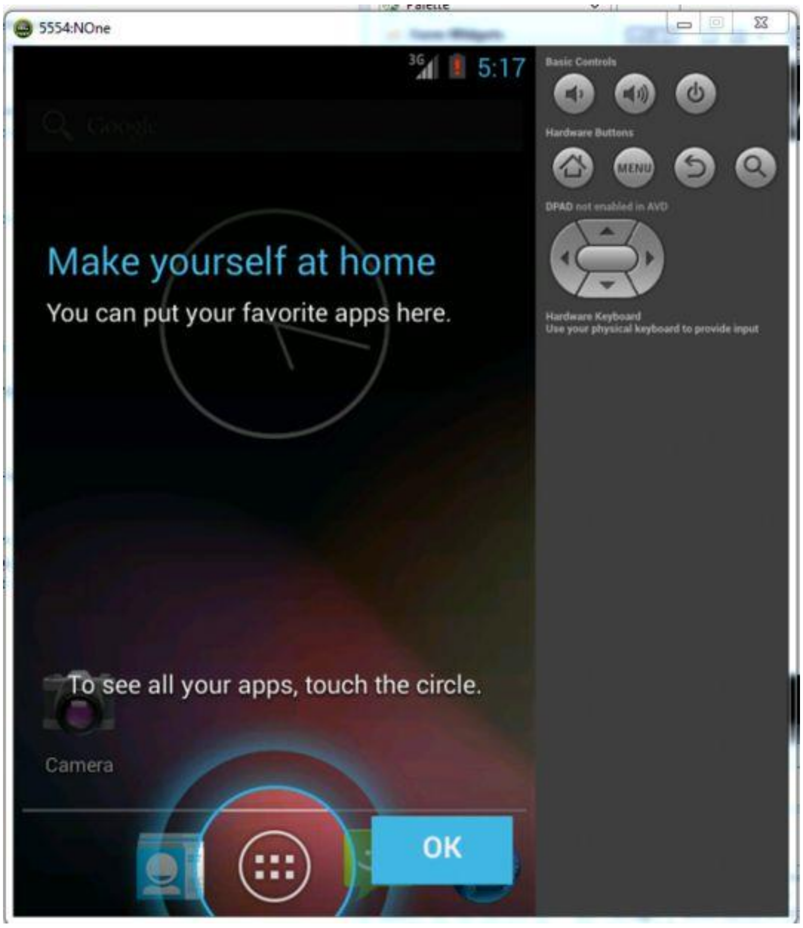
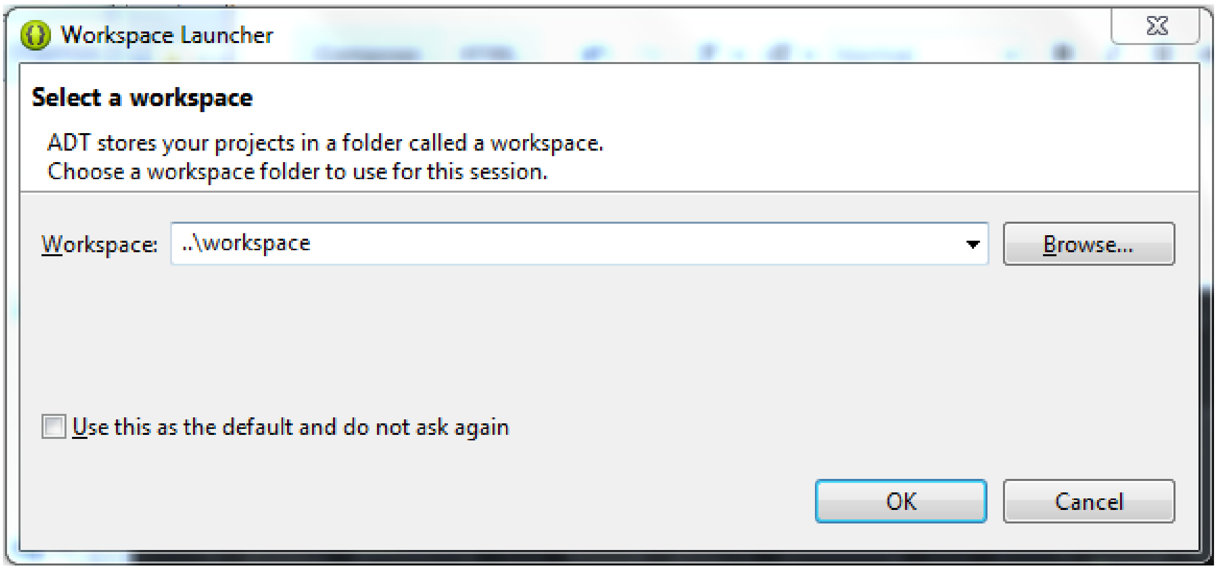

5554:NOne
5:17
Make yourself at home
You can put your favorite apps here.
To see all your apps, touch the circle.
Camera
OK
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input

Fig2.5: Emulator home screen

2.2 Set-up (Running Test App)

Steps of creating applications

1.Start Eclipse and select your workspace


Workspace Launcher

Select a workspace

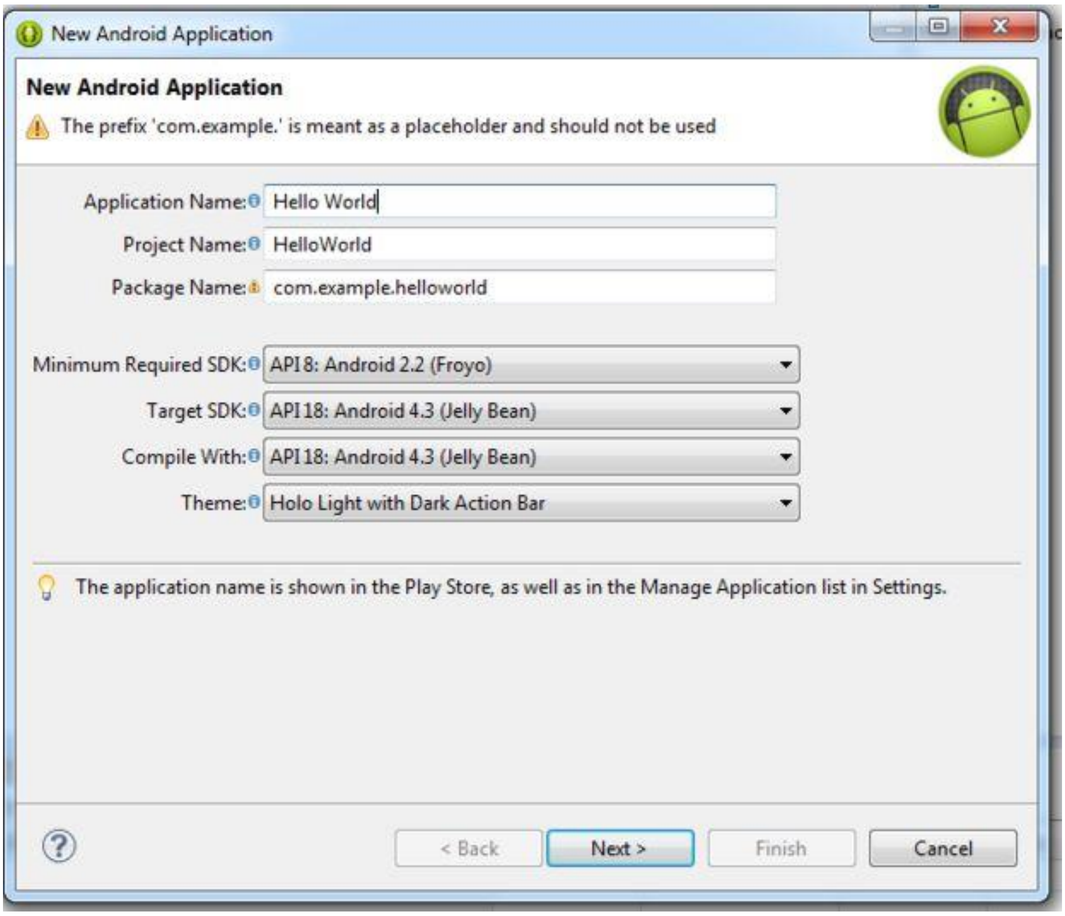
ADT stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: ..\workspace

□ Use this as the default and do not ask again

Fig2.6: Specifying the workspace

2. Then, go to File -> New -> Android Application Project. Name your Application and select the version like this:



New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: Hello World

Project Name: HelloWorld

Package Name: com.example.helloworld

Minimum Required SDK: API 8: Android 2.2 (Froyo)

Target SDK: API 18: Android 4.3 (Jelly Bean)

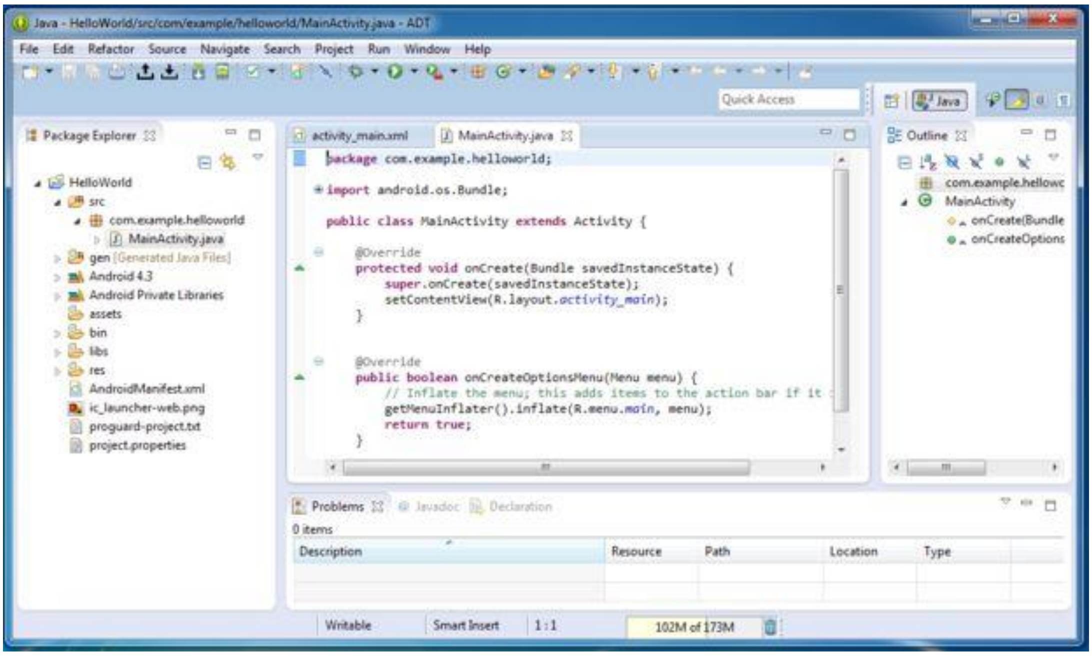
Compile With: API 18: Android 4.3 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

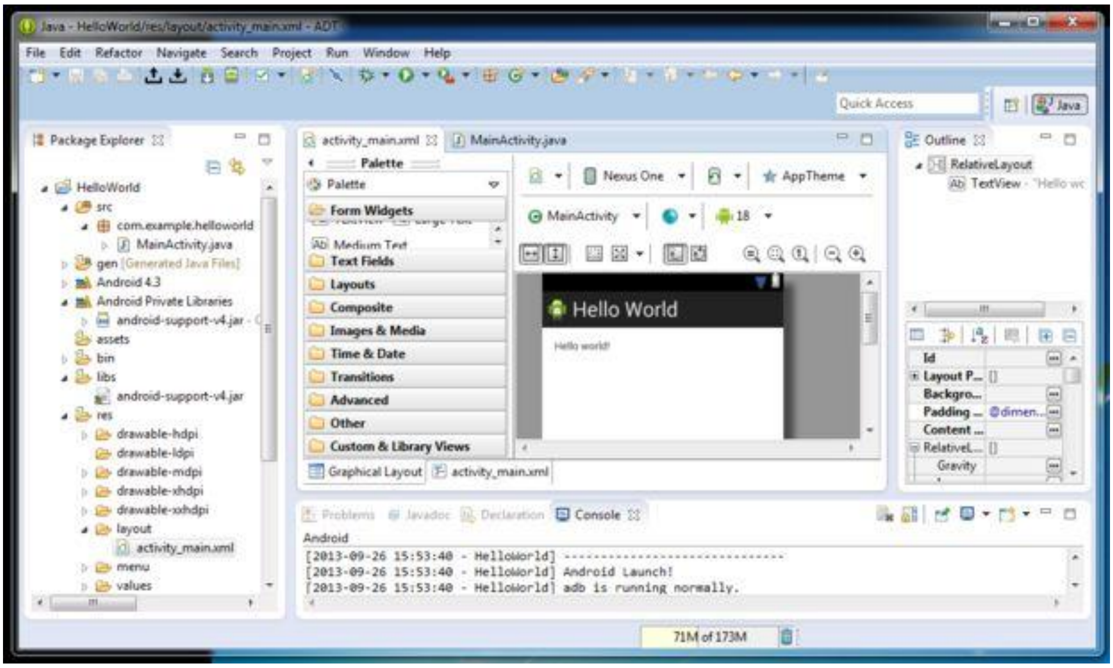
Fig2.7: Creating a new android project

3. By default there will be a program to print hello world. We can run it on either a USB connected Android Device or the AVM emulator.



```
package com.example.helloworld;

import android.os.Bundle;

public class MainActivity extends Activity {

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


    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }
```

Fig2.8: Default code screen for new project

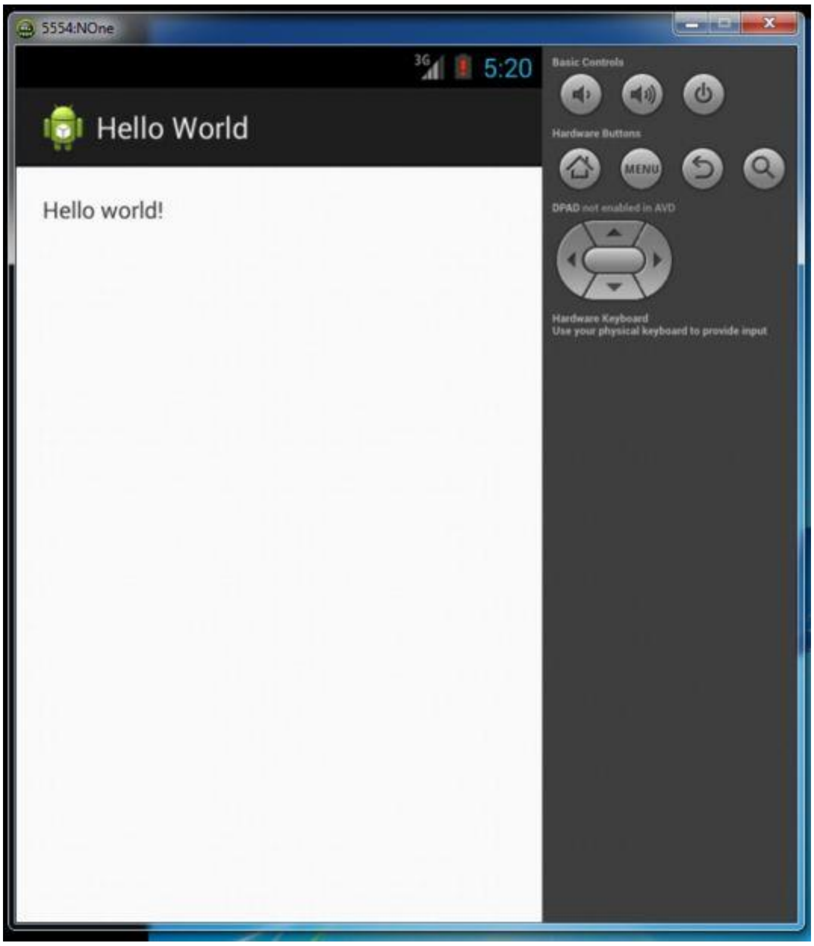


```
[2013-09-26 15:53:40 - HelloWorld] ------------------------------
[2013-09-26 15:53:40 - HelloWorld] Android Launch!
[2013-09-26 15:53:40 - HelloWorld] adb is running normally.
```

Fig2.9: Default graphical layout of project

4. The Emulator has already been loaded in Fig2.5

5. When we run the Hello World program on it, it looks like:



```
Hello world!
```

Fig2.10: Project running on emulator

3. ANALYSIS & DESIGN

3.1 ANALYSIS & DESIGN

3.1.1 External Interface Requirements

Main:

User should be able to access and change the choice with a touch on the predefined button.

Find Places:

The user should be able to find nearby places with respect to the geographical location without any problem. The map loaded will show the listed places on the map fragment.

Stations:

The user must be able to view the list of stations. This will give the user a listview of the stations. On selecting a particular station a Google map should appear with options to find the nearby ATM's, Malls and restaurants which would be located on the map loaded.

Emergency:

The user must be able to call the emergency number defined. The user can call the helpline police or ambulance in case of emergencies.

3.1.2 Hardware Interfaces

Since the app is an Android app, Android hardware will be a requirement to run the app.

For example any Android device like Nexus 5, Note 2, etc. The hardware should also be touch enabled as well as should support Google maps.

3.1.3 Software Interfaces

Around Me! runs on every android phone available so it does not need any specific software from the user's side. However, it depends on some libraries and tools for its function. These packages are already pre-included in the .apk file downloaded.

3.1.4 Communications Interface

The network and internet is absolutely necessary as it is based on the data acquired from the server. A working internet connection, preferably WiFi, to get the information of the nearby places and locate it on the map. The network is also necessary to make the emergency call when required.

3.1.5 Functional Requirements

Nearby Places:

Trigger - Touching the "Find Places" Button.

Function - Will take the user to the page with all the options and a search bar to search nearby places.

Error Handling – User will be prompted that there was some error that occurred with the option to send the log file to the developer and will restart the app.

Places on the map:

Trigger - Touching the "Show all on map" Button.

Function - Will take the user to the page where the user can view all the search places on the map.

Error Handling – User will be prompted that there was some error that occurred with the option to send the log file to the developer and will restart the app.

Stations:

Trigger - Touching the "Stations" Button.

Function - Will take the user to the page where the user can view the different stations and select the station from the list displayed.

Further the user can find nearby ATM's, Restaurants and malls and get their location on the map.

Error Handling – User will be prompted that there was some error that occurred with the option to send the log file to the developer and will restart the app.

Emergency call:

Trigger - Touching the "Emergency" Button.

Function - Will take the user to the page with the police and ambulance number predefined.

Further the user can choose on which number to call.

Error Handling – User will be prompted that there was some error that occurred with the option to send the log file to the developer and will restart the app.

3.1.6 Use Cases

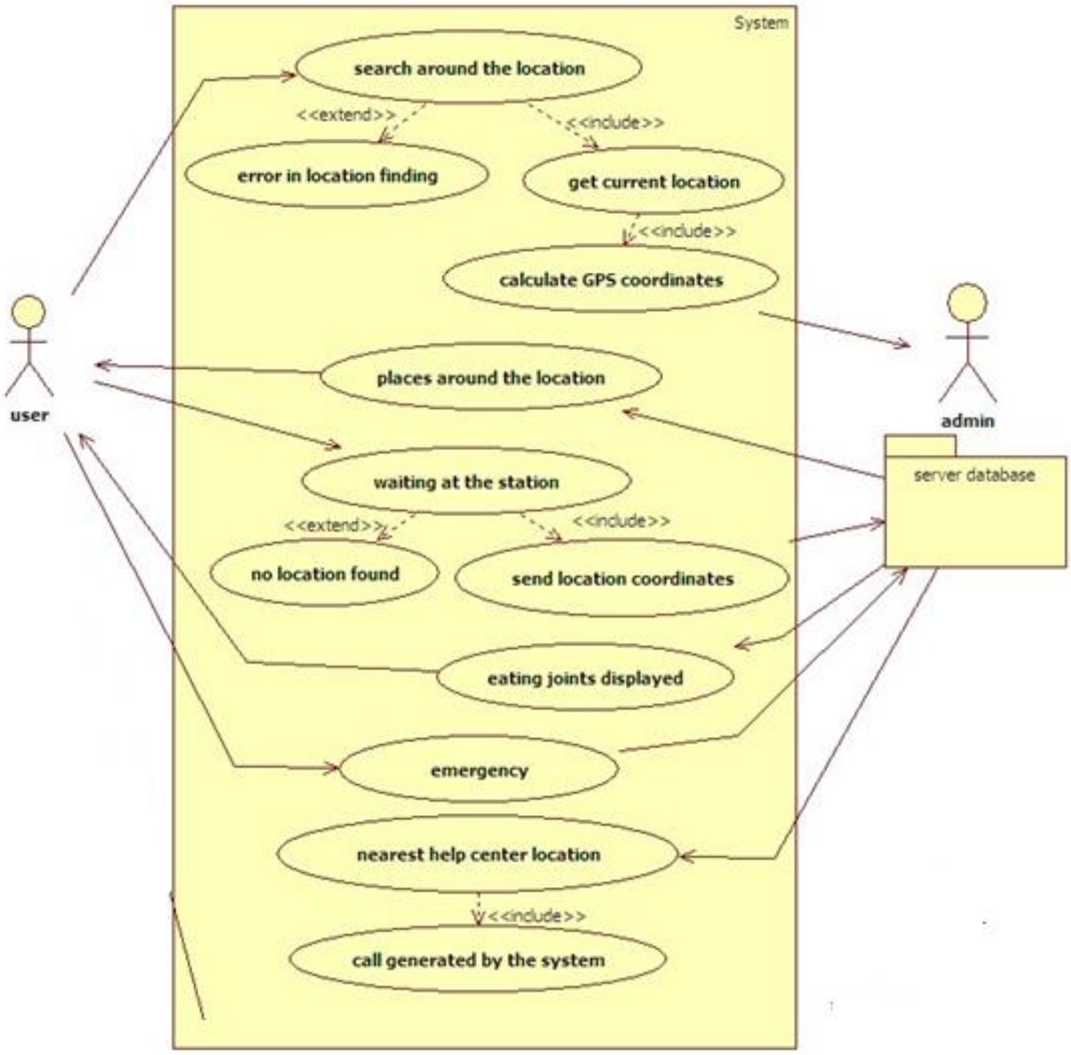


Fig3.1: Use case for the Application

The system use-case:

Purpose: To describe the use-case for the system and its functionalities.

Steps:

- The user opens the app.
- The user gets nearby places.
- The user requests Station.
- The user calls emergency.
- The user might not get the desired outcomes if the requirements are not met.

3.1.7 Non-Functional Requirements

Performance

The application may have subjective performance on high end as well as low end android smartphones because the app is somewhat a resource intensive application due to the implementation of Google maps.

Reliability

The application has the functionality to provide secure and reliable data from time to time and hence increasing the reliability of the information provided.

Availability

The application will be available in the form of ".apk" initially and once approved will be available on the Google Play Store once approved by Google.

Security

Security will not be an issue because of the no use of personal information of the user is stored.

Maintainability

Continuous application updates and maintenance releases will be offered to keep the application running smoothly over time.

Portability

The application will of course be portable because it is a mobile application.

3.1.8 Design Constraints

- The Android design guide was followed to make use of the recommended design language.
- Software interfaces involved in designing the application was constrained to Android SDK.
- The program size can’t be very large as it is a mobile application which have a limited amount of storage space.
- The hardware has to be an android phone with touch screen capabilities.
- The portrait and landscape views are designed separately to enhance the user experience and avoid any distortion of the GUI.

3.1.9 Other Requirements

- The user does not need any special training in using the application as it’s a very simple and intuitive application.
- When it comes to reusability the source code will be open source so anyone can work on it and improve it.
- Google maps support should be available on the device for the proper working of the application.

3.2Analysis Models

3.2.1. Sequence Diagrams:

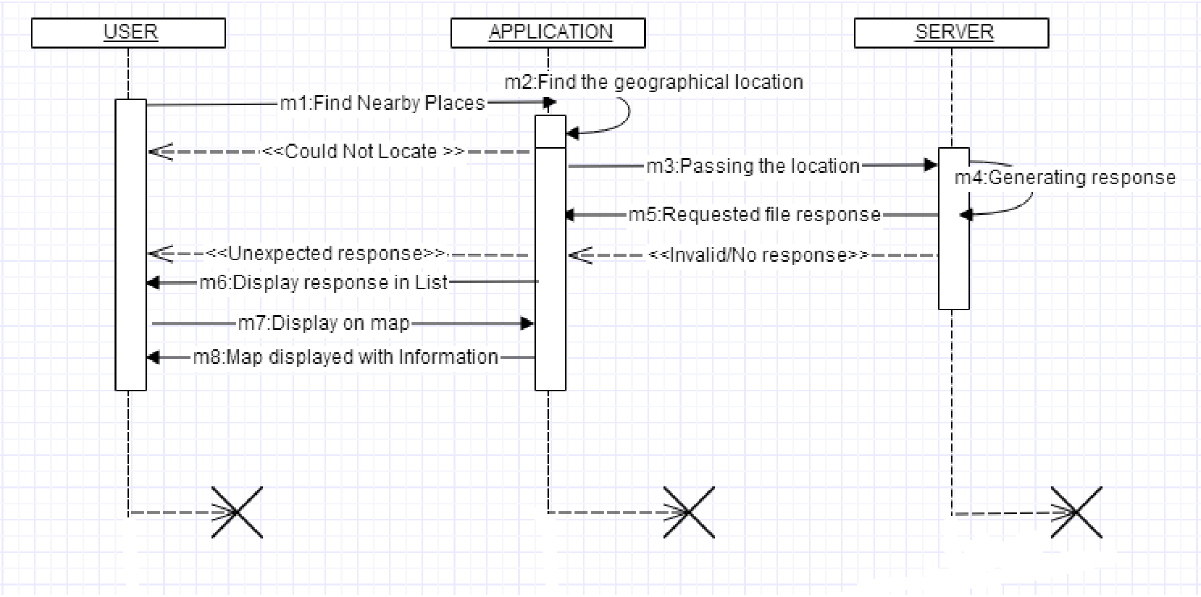


Fig3.2: Sequence diagram for Find Places

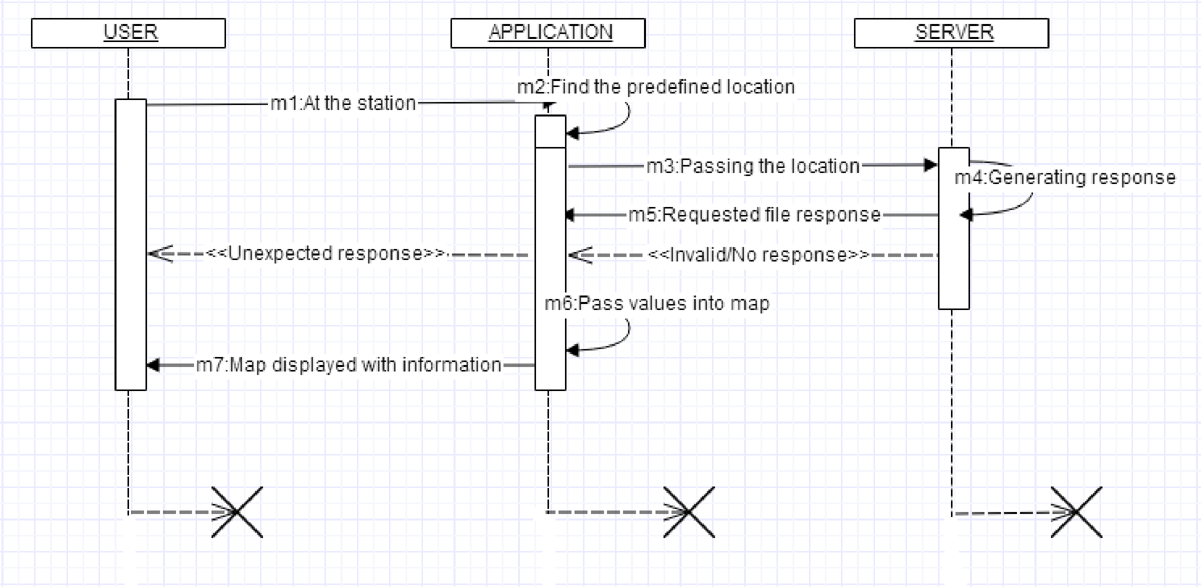


Fig3.3: Sequence Diagram for handling the At the station requests

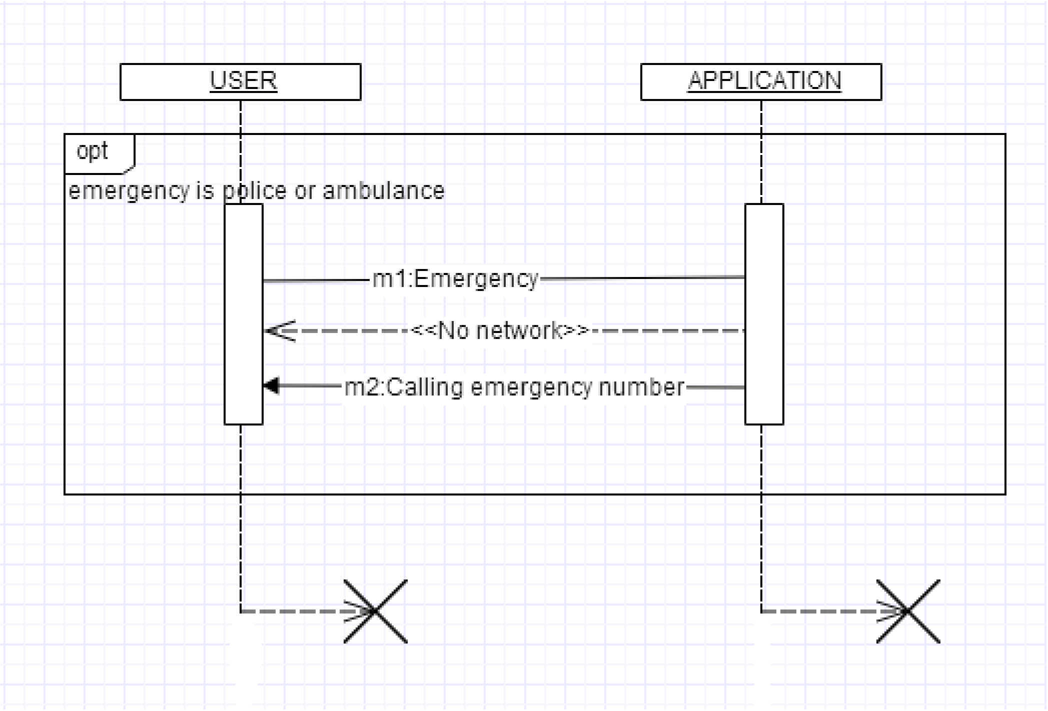


Fig3.4: Sequence diagram for Emergency Calls

3.2.2. Data Flow Diagram (DFD)

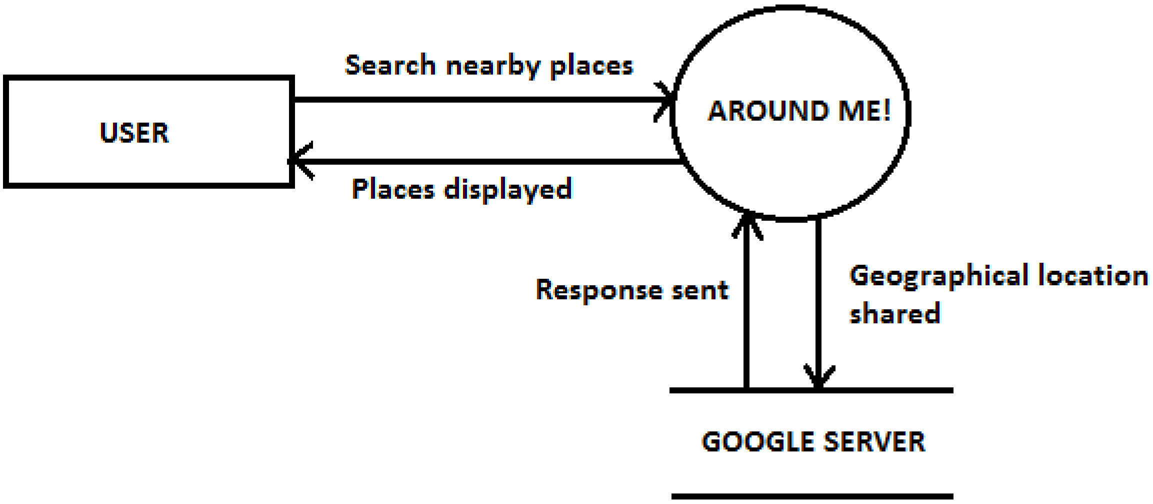


Fig3.5:DFD for the system

4. PROJECT IN DETAIL

Here we will describe the working of the project including the working of the code given above in detail

4.1 Splash Screen

A splash screen is an image that appears while a game or program is loading. The term may also be used to describe an introduction page on a website. Splash screens cover the entire screen or simply a rectangle near the center of the screen. Our slash screen will be displayed for 3 seconds. For the app to display the splash screen, we declare the SplashScreen activity with intent.action.MAIN to that it is opened first in AndroidManifest.xml.

Once the timer expires we will be redirected to the Main Activity

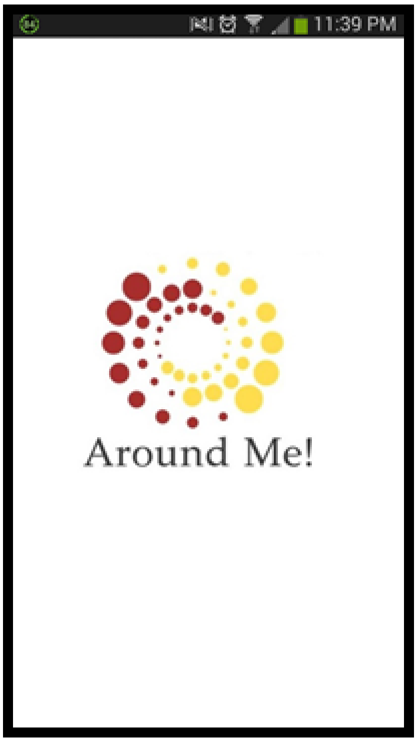


Fig4.1: Splash Screen

4.2 Main Screen

Here there will be choice for the user to access the functionalities available for searching nearby places, search near a local station, or call on the emergency helpline.

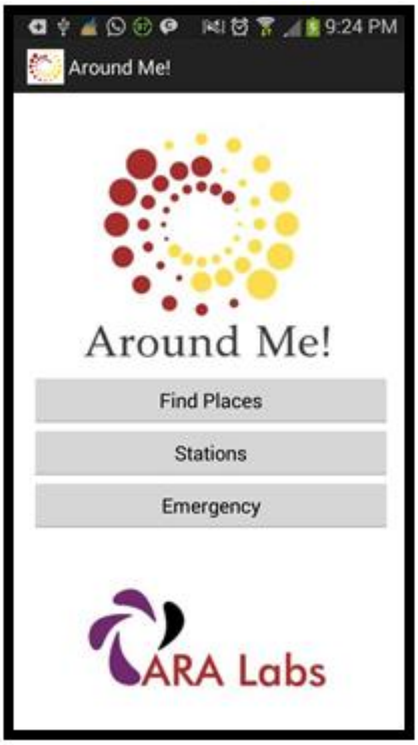
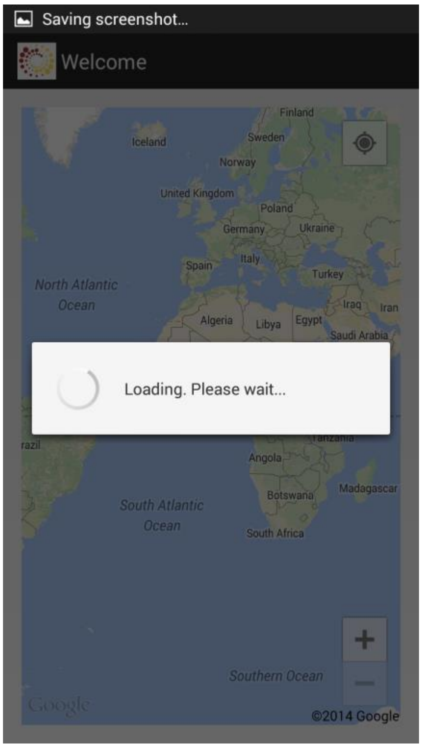


Fig4.2: Main screen/Home screen

4.3 Nearby Places

On selecting Find Places we get a loading screen which acquires our current location. As the location is taken it is taken to the actual search activity that contains all the options and search bar to search key words.



Loading. Please wait...

Fig4.3: Loading screen

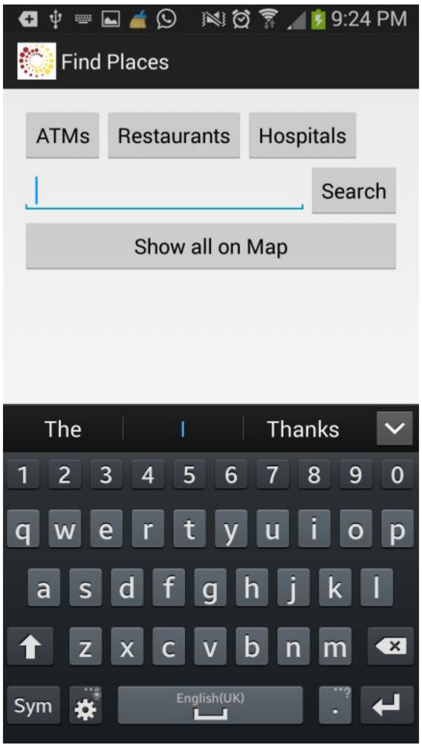
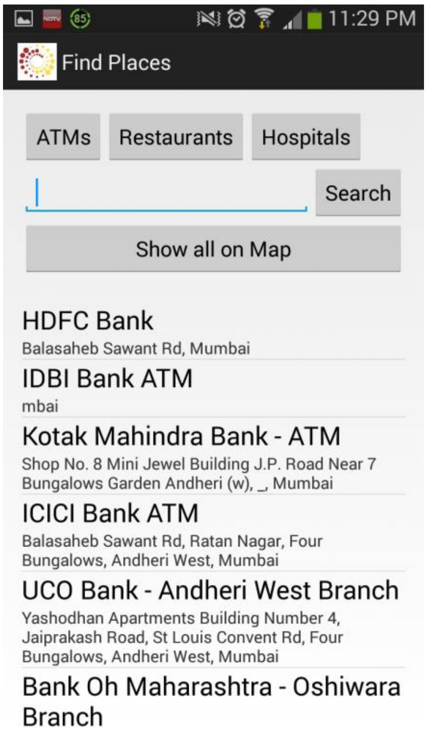


Fig4.4: Search screen

At this stage there will be a choice on the user to search or to find places defined on the buttons. The output will be displayed with the help of listView which will be populated dynamically. As the places are many and would be unnecessary, we have limited ourselves to a radius of 1300 meters and 20 nearest places.



HDFC Bank
Balasaheb Sawant Rd, Mumbai

IDBI Bank ATM
mbai

Kotak Mahindra Bank - ATM
Shop No. 8 Mini Jewel Building J.P. Road Near 7 Bungalows Garden Andheri (w), _, Mumbai

ICICI Bank ATM
Balasaheb Sawant Rd, Ratan Nagar, Four Bungalows, Andheri West, Mumbai

UCO Bank - Andheri West Branch
Yashodhan Apartments Building Number 4, Jaiprakash Road, St Louis Convent Rd, Four Bungalows, Andheri West, Mumbai

Bank Oh Maharashtra - Oshiwara Branch

Fig4.5 Search result for ATM

The user can see the obtained result on the map by pressing the ”Show all on map” button.

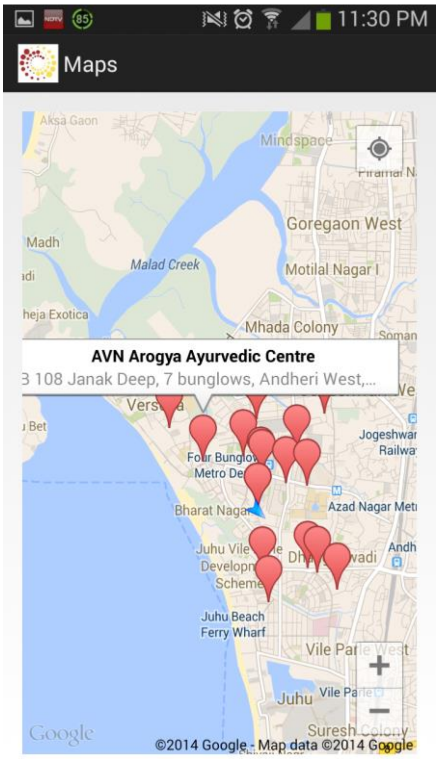


Fig4.6 Hospitals searched on map

The rest of the markers depict the other hospitals that are found.

The other places can also be found and will be displayed in a similar layout with the same options.

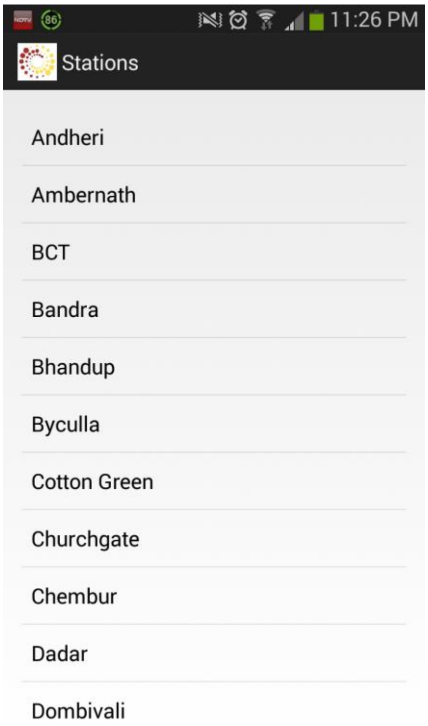
This is done by sending an HTTP request to the Google places API which responds with an XML file which is processed and the details are populated into the dynamic list created by the adapter. The geographic location is also taken and stored in the dataset but not displayed. This information is used when the identity is displayed on the map. The marker indicates the location and the title gives the name of the location, the snippet is used for the address of the location.

4.4 Stations

We have provided the user with a list of all the stations of the Mumbai local. The user can choose the station as per the requirement. As the station is selected it activates the OnItemClickListener, which passes the latitude and longitude coordinates to the next activity (Fig 4.8).

We have a three buttons available for displaying the nearby ATM’s, Restaurants and malls as by public research we got to know that these are the most searched places nearby the stations.

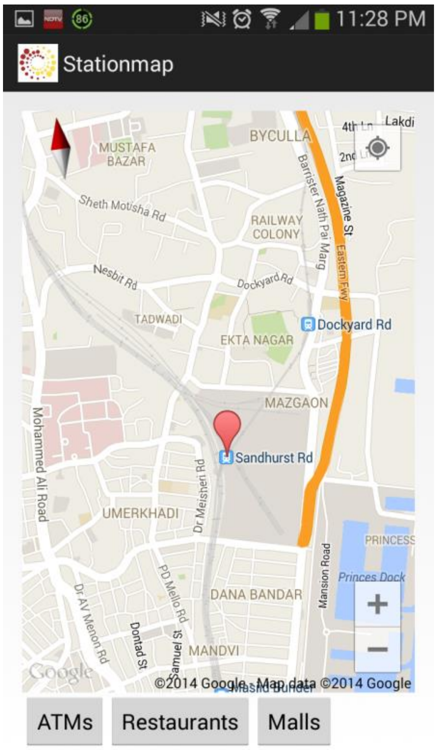
Whenever the user clicks on one of the buttons, its respective destinations will be displayed. When the user clicks on one of the markers, the details are given by the title and snippet. The data is received with the same mechanism as that of the previous module.



Stations

- Andheri
- Ambernath
- BCT
- Bandra
- Bhandup
- Byculla
- Cotton Green
- Churchgate
- Chembur
- Dadar
- Dombivali

Fig4.7 Station List



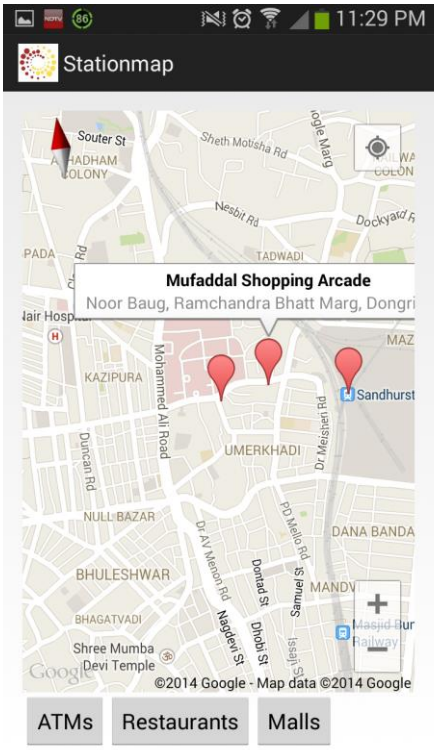


Fig4.8 Selected station on map Fig4.9 Malls displayed near the station

Similar outputs are received for ATM's and Restaurants also provided we have good internet connectivity as well as device support.

4.5 Emergency

When the emergency button is pressed, it navigates the user to the emergency screen with two buttons that have the police and ambulance helpline numbers predefined.

On clicking the button the number is called. This is specially done so there is no time wastage at the time of emergency.

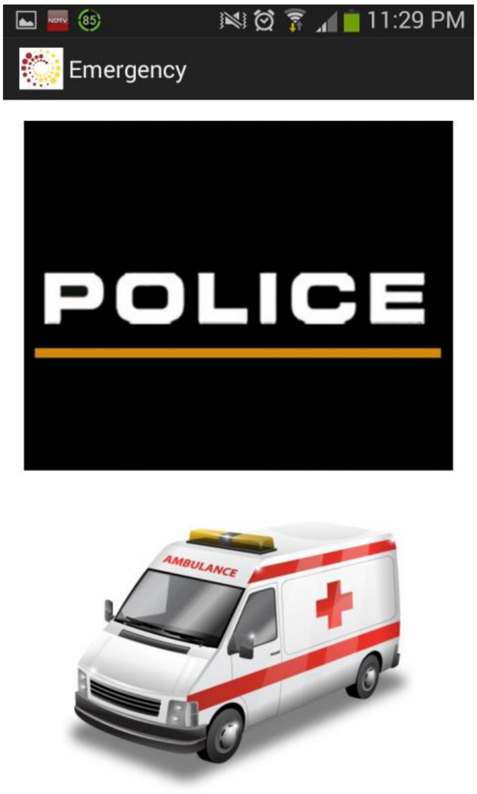


Fig4.10 Emergency screen

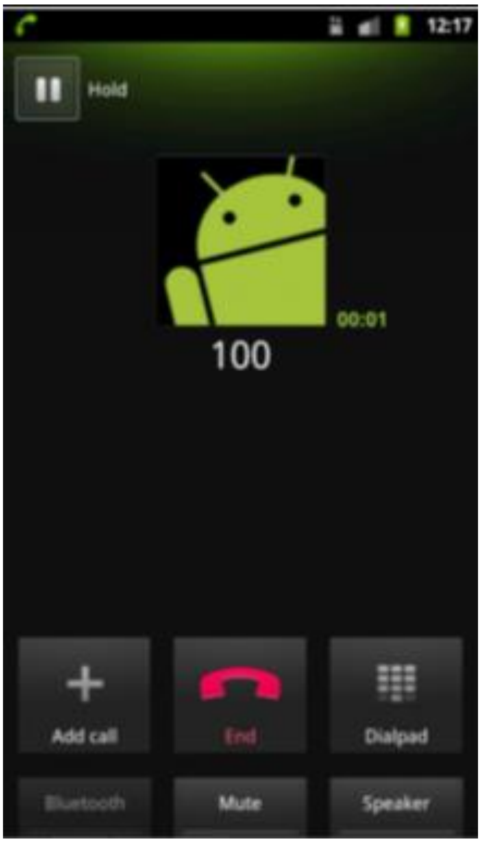


Fig4.11 Calling Police

While calling the ambulance a call to 101 is made automatically by the system

5. CONCLUSION & FUTURE SCOPE

5.1 CONCLUSION

Thus we have made an Android Application, a location based application for everyone using an android device to get information about the nearby places, called Around Me! The application's primary purpose is to help the user **locate different places around** the current location of the user.

- This application will show locations as per the users own specifications and needs.
- To provide an emergency dial to the user in need without delay.
- To provide destinations popularly searched around the local station.
- It reduces the amount of time and energy required to locate a place.
- It enhances the reliability of information.

This can be used to retrieve information around the user's current location.

The project is concluded with a clear purpose as to why the application is being developed. The major requirements of the user are understood and the functionalities are designed to achieve the objective-to satisfy the user.

5.2 FUTURE SCOPE

This is not the final polished Application. There are a number of things that can be added to better the experience of using this App. Some of them are:

- The ability to add favorites i.e. to add information about the places most visited by the user by adding a database.
- The ability to add local as well as national stations so that the app can be used in cities with metro's or some local transport system like Delhi, Chennai, Calcutta, etc.
- The ability to notify the user if he is near his favorite place.

Other amazing features as and when suggested will also be incorporated to the best of our abilities in the Application.

REFERENCES

For installation and basics:

[1] http://developer.android.com/training/basics/firstapp/running-app.html

For information and services:

[2] https://code.google.com/apis/console/?pli=1

[3].http://stackoverflow.com/questions/21950617/there-seems-to-be-an-error-in-sending-request?noredirect=1#comment33353182_21950719

[4]. http://www.androidhive.info/2013/08/android-working-with-google-maps-v2/

[5]. http://developer.android.com/training/index.html

[6]. http://stackoverflow.com/questions/15152099/updating-adt-plugin-in-eclipse

[7]. http://www.w3schools.com/googleapi/default.asp

[8].https://developers.google.com/apis-explorer/#p/