ADVANCE CALL MANAGER

Project Report submitted in partial fulfillment of the requirement for the degree of
Bachelor of Technology
In

COMPUTER SCIENCE AND ENGINEERING

By

GAURAV GUPTA (131220)

Under the supervision of

Dr. RAVINDRA BHATT

To

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Waknaghat, Solan-173234,

Himachal Pradesh

Certificate

Candidate's Declaration

I am here to declare that the work presented in this report entitled "**Advance Call Manager**" in partial fulfilment of the requisites for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** submitted in the **department of Computer Science & Engineering and Information Technology**, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from February 2017 to August 2017 under the supervision of Dr. Ravindra Bhatt (Designation, CSE).

I have not submitted this work anywhere and I am solely responsible for the work presented in my B.Tech Indsutrial Project.

(Student Signature)
Gaurav Gupta, 131220

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. Ravindra Bhatt
Assistant Professor
CSE
Dated:

Acknowledgement

I optate to express my profound gratitude and indebtedness to my guide, Dr. Ravindra Bhatt, Department of Computer Science and Engineering, JUIT, Solan for giving us the opportunity to work with him.

I additionally place on record, my sense of gratitude to one and all, who directly or indirectly have lent their availing hand in this venture.

I feel proud and privileged in expressing my deep sense of gratitude to all those who have availed me in presenting this project.

Last but not least, my sincere thanks to all my friends who have patiently elongated all sorts of avail for accomplishing this undertaking.

TABLE OF CONTENT

List of Abbreviations

Sr. No	**Abbreviations**	**Term**
1	Advance Call Manager	ACM
2	Graphic User Interface Module	GUI Module
3	Database Module	DB Module
4	Open Handset Coalition	OHA
5	Java Run time Environment	JRE
6	Application Programming Interface	API
7	Java Virtual Machine	JVM
8	Acorn RISC Machine	ARM
9	Software Development Life Cycle	SDLC
10	Software Development Kit	SDK

List of Figures

List of Tables

Abstract

During my internship at Click Labs, I was trained in Android Development. I learnt about various tools and technologies that are used in IT Industry. I learnt concepts that can be used to solve real time problems. After the successful completion of my training I was a given a project **“Advance Call Manager”** to implement the concepts that I had learnt during my training.

I also revised the concepts of Core Java and also worked on GIT. I learned about View Pagers, Recycler View, Maps, and JSON Parsing.

In my project **“Advance Call Manager”**, I have made use of SQLite Database. The basic idea of my project is to provide functionalities in a single application instead of having separate application for different functionalities. So it saves memory and also the most important feature of this application is to reject fake call and Call Recording. So it’s a multipurpose application. I tried to implement most of the concepts in my project.

CHAPTER-1

INTRODUCTION

1.1 Introduction of Project

In this time of information and technology, we have to face an abundance of quandaries while we are working on our Android predicated Perspicacious phones. We have to utilize separate application to work upon sundry day-to-day call cognate operations like recording a call, viewing the call log, managing the blacklists, transmuting call and in-call setting for our contrivance. We don't have a single application, which may provide us all of these operations within a single package. No matter how advance your phone is, when you receive a call you are just left with two or three options-answer or to hang up.

Advanced Call Manager (ACM) is an android application that gives you the to take your phone calls just the way you like to plus it has additional features that gives the user a technology boost. You get more call-handling options (abnegating a call, fake call, call recording) and can personalize your replication for different groups or contacts.

If we opt to install separate applications to perform all of these operations, then these applications, takes a plethora of space and recollection in our wireless handheld contrivance. Hence, we are going to design "Advance Call Manager" application, which will be able to provide all the above mentioned operations and options within a single application. This application will take less recollection and storage space for its execution. Individual settings are additionally sanctioned- abnegation of an individual call or repudiating every incoming call (for example blocking few contacts out of all the contacts present in your phone). Advance Call Manager's utilizer-centred design makes Advance Call Manager intuitive despite its advanced features so you can expeditiously become a potency utilizer.

Modules of the project

- GUI MODULE
- PROCESS MODULE
- DB MODULE

- **GUI MODULE**

This module is being used to design the GUI Interface for the application. To design the GUI Module of our application, we have utilized sundry aspects of Android containing sundry kind of GUI Controls, Components, Notifications, Widgets, etc. utilizing the XML, utilizer can withal transmute the theme of his application for overall GUI components.

- **PROCESS MODULE**

In our application, we will utilize Java SE 8.0 to perform all the operations in its backend. Here, sundry aspects of Java SE 8.0 will be used like I/O, Multithreading, Serialization, Event handling, Functions, Classes, Package etc. Android Support package will additionally be habituated to provide the android framework and utilities in the java programming.

- **DB MODULE**

Sundry kind of database operations will additionally be sanctioned to the utilizer to maintain and manage his/her blacklists call-logs etc. But as we kenned, wireless smartphones doesn’t have such configuration to provide us better and efficient database systems. Hence, we will utilize SQLite database in our application, as it is very light weight, facile to utilize compact database management system for the android systems.

1.2 Problem Statement

As it has been already described that our application has been designed to provide all the day-to-day call cognate operations like recording a call, viewing the call log, managing thee blacklists, transmuting call and in-call setting for our contrivance and other options within a single application, so that we may not require to install separate application to perform all of these operations. This application will provides all these operations in less storage space and main recollection. optically discerning all the call logs, setting of the subsisting contacts of the phone optically discerning all the call logs, setting of the subsisting contacts of the phone for instance integrating the incipient contacts in subsisting contacts, viewing all subsisting contacts in sorted order (names along with their numbers), effacing the contacts from the subsisting contacts of phone recollection, updating the data of subsisting contacts and making the fictitiously unauthentic calls if need arises. It additionally facilitates the recording of all incoming and outgoing calls. These call logs can be stored in recollection of the phone through the utilization of file handling mechanism however it may require the utilization of root access of the phone.

As it has been already described that our application has been designed to provide all the day-to-day call related operations like –

- recording a call
- viewing the call log
- managing the blacklists
- changing call
- in-call setting

For our contrivance and other options within a single application, so that we may not require to install separate application to perform all of these operations. This application will provides all these operations in less storage space and main recollection.

1.3 Objectives

ACM is an Advance Call Manager mobile predicated application that enable you to handle the call, integrate the number, efface the number, block a call, record the call and call to someone from your contacts within a single app.

Whenever any data is transmitted on the network then any uncertified person can read the data on network which is not good for data transmission. Security is provided by enciphering the data so that an outsider cannot tamper the data and send it to the person intended to and the receiver or the recipient will decrypt the encrypted data (image) and uses it.

For handling call we have to download multiple app like for recording a call, for tracing a call, for blocking a call etc. But ACM provide all these facilities within the single app.

Individual settings are additionally sanctioned- abnegation of an individual call or repudiating every incoming call (for example blocking few contacts out of all the contacts present in your phone). ACM's utilizer-centred design makes ACM intuitive despite its advanced features so you can expeditiously become a potency utilizer.

The main purport of developing ACM is to learn about the facts how system app interact others app. & how a system app is developed.

Advance call manager provide day to day life facility. The main objective of advance call manager is:

- Recording a call
- Viewing the call log
- Managing the blacklists
- Share the phone number
- Call to person
- Make a fictitiously unauthentic call

For our contrivance and other options within a single application, so that we may not require to install separate application to perform all of these operations. This application will provides all these operations in less storage space and main recollection.

1.4 Methodology

We utilize android & SQLite for developed android app. In android java programming language is utilized for handle the events & XML is utilized for provide the layout of app. Android is a platform for making applications that runs on computers, television, tablets, etc. It is basically an operating system for mobile applications. Its operating system is Linux based. It is currently utilized in sundry contrivances such as mobiles, tablets, televisions etc. It was initially developed and later OHA added some features. The code is written in JAVA and the design part is written in XML.

The main objective of android project is to make applications with better user interface and also has some features which make user experience memorable. There are various versions of android such as Kit Kat, Jelly Bean, Lollipop, Frozen di-hydrogen monoxide cream Sandwich, Frodo, Eclair , Donut etc. Now the latest version is Marshmallow. Our Android App is additionally support Marshmallow version.

1.5 Organization

Analysis of Android

After deep analysis of android, the various features of android can be listed as follows:-

1) Android development is open -source

2) Android Platform can be customized by anyone.

3) There are many mobile applications that are consumed by the users.

There are three approach for development of an android app. These are

- **Java Predicated:** Use Java to define Strings, lay out window, engender GUI controls, and assign event handlers. Like Swing programming.
- **XML-predicated:** Use XML files to define Strings, lay out window, engender GUI controls, and assign event handlers. The Java method will read the layout from XML file and pass it to Set Content View.
- **Hybrid:** Utilize an XML file to define Strings, lay out window and create GUI controls. Utilize Java to assign event handlers. For development of Advance Call Manager app we utilize Hybrid approach that contain Java & XML technology.

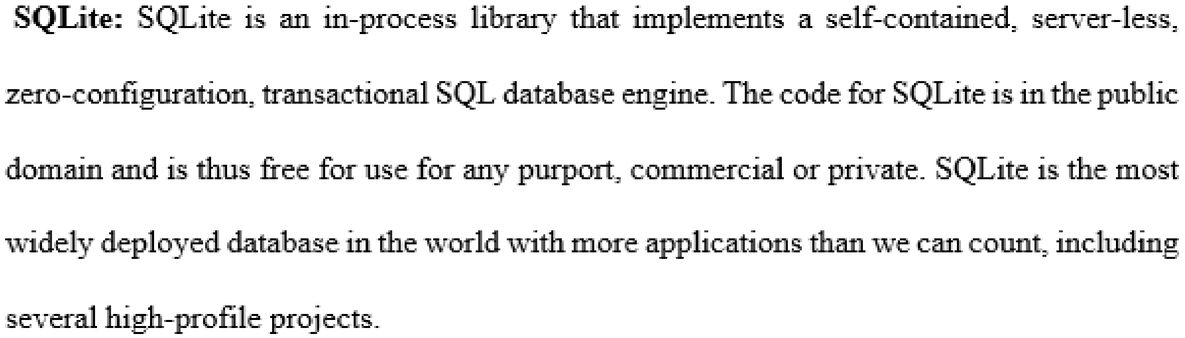
Java: Code for android development is mainly written in java language. Java was first relinquished in the mid-1990s by Sun Microsystems. It was designed to be facile to learn by programmers who already kenned C and C++. During 2006 and 2007 Sun relinquished Java as free and open-source software, under the terms of the GNU General Public License (GPL). Sun was bought by Oracle in 2009/2010, and Oracle remains committed to Java.

To start inditing Java programs you require a way to compile source code and turn it into an executable for the Java runtime. The mundane way to do this is to install the Java Development Kit. At the time of inditing the current version of Java is Java 8, however it is relatively incipient, so Android uses Java 7.

XML: XML full form is extensible mark-up language. It is basically used to share data around the world. Here it will be explained how to parse an xml file and take out important information from it.

There are various kinds of xml parsers which are used to parse an xml file. Android provides three types of XML parsers which are SAX, XMLPullParser and DOM. The best xml parser is XMLPullParser as it is very easy to use.

The first and foremost step is to find the fields in XML we have to deal with. The main motive behind using XML is to design the front end or the user interface of the mobile application or android application.



SQLite: SQLite is an in-process library that implements a self-contained, server-less, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purport, commercial or private. SQLite is the most widely deployed database in the world with more applications than we can count, including several high-profile projects.

We utilize android & SQLite for developed android app. In android java programming language is utilized for handle the events & XML is utilized for provide the layout of app. Android is a platform for making applications that runs on computers, television, tablets, etc. It is basically an operating system for mobile applications. Its operating system is Linux based. It is currently utilized in sundry contrivances such as mobiles, tablets, televisions etc. It was initially developed and later OHA added some features

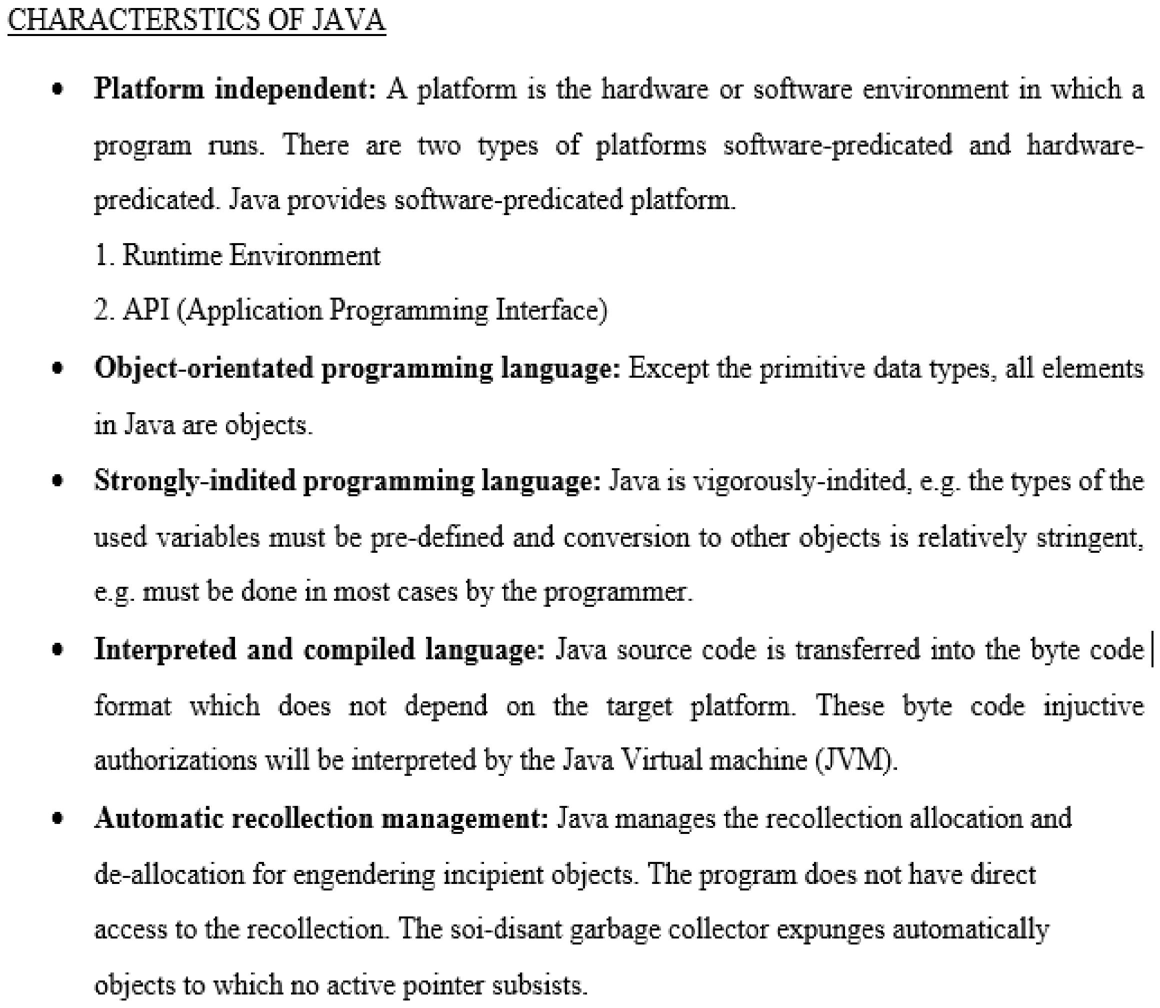
CHAPTER-2

TOOLS AND TECHNOLOGIES

2.1 INTRODUCTION TO PROGRAMMING LANGUAGE

JAVA INTRODUCTION

Java is an object oriented programming language. It is also a platform for writing the android development code. It is secure and high calibre Java is called a platform as it is having its own JRE and Application Programming Interface.



CHARACTERSTICS OF JAVA

- **Platform independent:** A platform is the hardware or software environment in which a program runs. There are two types of platforms software-predicated and hardware-predicated. Java provides software-predicated platform.
  1. Runtime Environment
  2. API (Application Programming Interface)
- **Object-orientated programming language:** Except the primitive data types, all elements in Java are objects.
- **Strongly-indited programming language:** Java is vigorously-indited, e.g. the types of the used variables must be pre-defined and conversion to other objects is relatively stringent, e.g. must be done in most cases by the programmer.
- **Interpreted and compiled language:** Java source code is transferred into the byte code format which does not depend on the target platform. These byte code injuctive authorizations will be interpreted by the Java Virtual machine (JVM).
- **Automatic recollection management:** Java manages the recollection allocation and de-allocation for engendering incipient objects. The program does not have direct access to the recollection. The soi-disant garbage collector expunges automatically objects to which no active pointer subsists.

Java Virtual Machine (JVM)

JVM full form is Java Virtual Machine which is a conceptual machine and the main use of JVM is to provide a runtime environment so that the java byte code can be executed.

In our application, we will utilize Java SE 8.0 to perform all the operations in its backend. Here, sundry aspects of Java SE 8.0 will be used like I/O, Multithreading, Serialization, Event handling, Functions, Classes, Package etc. Android Support package will additionally be habituated to provide the android framework and utilities in the java programming.

But as we kenned, wireless smartphones doesn’t have such configuration to provide us better and efficient database systems. Hence, we will utilize SQLite database in our application, as it is very light weight, facile to utilize compact database management system for the android systems.

2.2 INTRODUCTION TO ANDROID

Android is a software platform and operating system for mobile contrivances, predicated on the Linux kernel, and developed by Google and later the Open Handset Coalition. It sanctions developers to indite managed code in the Java language, controlling the contrivance via Google-developed Java libraries. Applications indited in C and other languages can be compiled to ARM native code and run, but this development path is not officially fortified by Google. The unveiling of the Android platform on 5 November 2007 was promulgated with the founding of the Open Handset Coalition, a consortium of 48 hardware, software, and telecom companies devoted to advancing open standards for mobile contrivances. Google relinquished most of the Android code under the Apache license, a free-software and open source license.

What is the main purpose of Open Handset Alliance?

Open Handset Alliance is basically an association of several companies like Google, Intel, Motorola, etc. It is used Code for android development is mainly written in java language. Java was first relinquished in the mid-1990s by Sun Microsystems. It was designed to be facile to learn by programmers who already kenned C and C++. During 2006 and 2007 Sun relinquished Java as free and open-source software, under the terms of the GNU General Public License (GPL). Sun was bought by Oracle in 2009/2010, and Oracle remains committed to Java. The main aim or objective is to develop applications with lower cost and also to lower cost of distributing the devices and the services given.

2.3 Platform (Hardware Used)

As we know the main purpose behind Android is to develop the applications that run on mobile devices and tablets. So the first and foremost motive is to manage the battery of mobile devices efficiently and to preserve the resources efficiently. There are different layers in Android system stack which are generally five in numbers. The ARM core of Linux provides a platform upon which all the other layers lay upon.

Linux is highly reliable and it is a fact that on Linux high performance is achieved at a very lower power consumption and the libraries helps in a way that the code is reusable. Linux is used for developing software for digital sound and video coding and decoding

The Dalvik run-time byte-code interpreter is a kind of the JAVA language byte-code interpreter, adding and integrating few features that define the security and the power-preservation model of android uniquely and differently. The main use of Android application framework is that it enables us to use the components that we want to use easily and effectively and we can use the components that we think is right for our application. The Java classes are also used to define android APIs

The core applications of Android include the Google Maps, Gmail, the web kit browser, and many more in line. The Android applications are basically written in java and most of the applications that are found in market are basically written in java and you can easily download these from the play-store or other providers on one go.

Android has a feature that it can run on multiple hardware configurations. An application that runs on one device can also run on another device with different hardware configurations. So it is good in that sense that it works on multiple hardware configurations. Android applications have everything like boot loader, etc.

2.4 A brief description of the Operating System

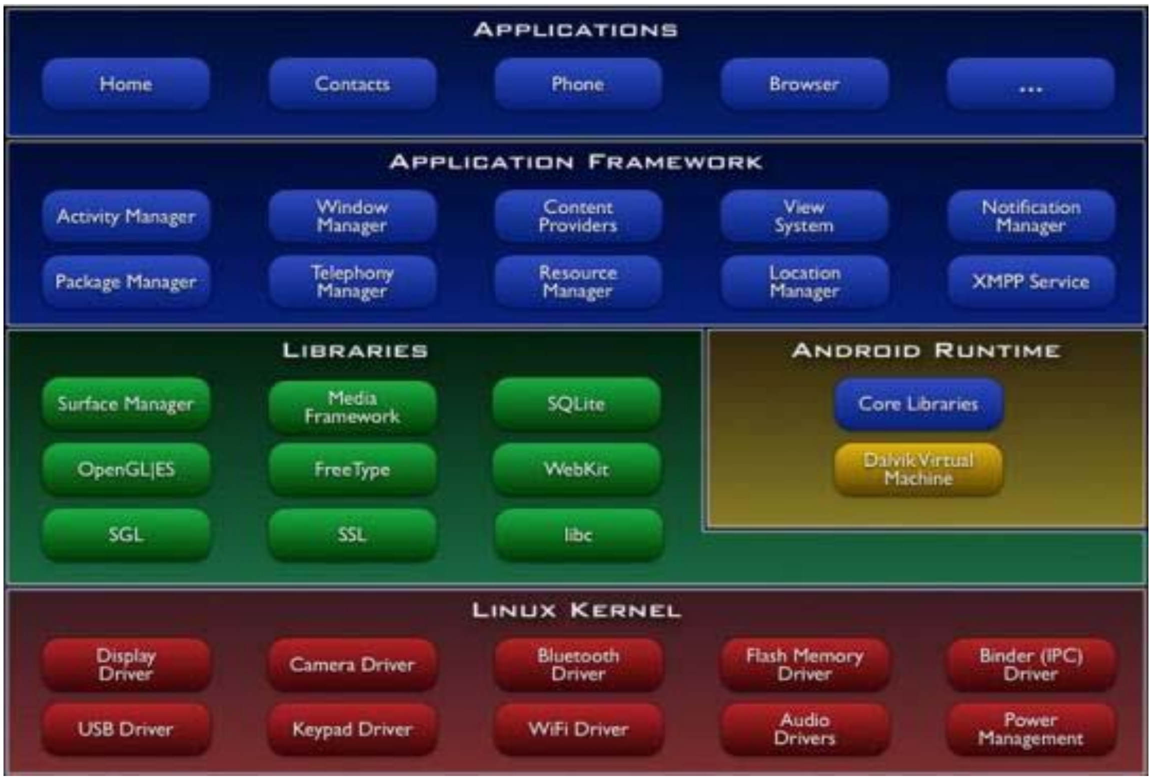


Fig 2.1 Block Diagram of Operating System

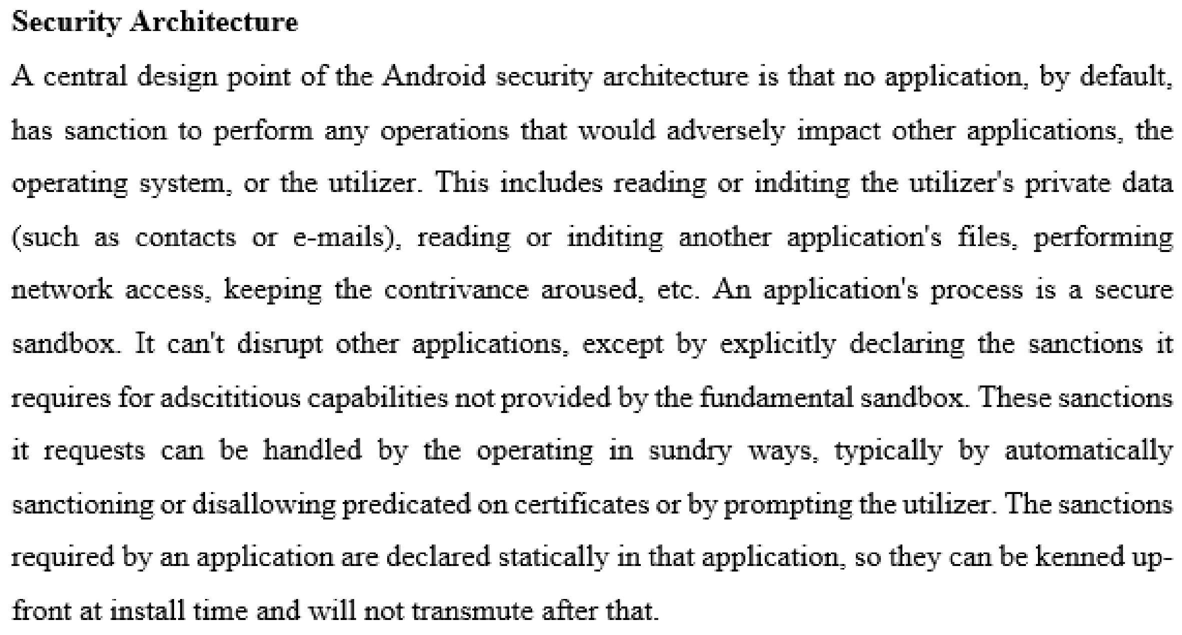
The first layer is Linux kernel which contains mainly the drivers like display driver, camera driver, Bluetooth driver, Wi-Fi driver, Audio drivers etc. These drivers are basically used for networking. This layer is never programmed directly. All these are written internally in C/C++, they are just called trough java interfaces. Next layer includes the libraries like Webkit, SSL and a web browser engine. The next layer includes android runtime including the virtual machines which gets converted at compile time.

The java libraries play an important role. These libraries are mainly written in java and everything above these layers is also written in java only. Android gives a major set of Standard Edition packages of Java 5.

Above this layer is the Application framework layer which contains home, contacts, phone, and browser. The main component of this framework is the Activity Manager which is responsible for managing the life cycle of all the applications. Also the major part of coding lies in this part only along with applications that are built in like phone and the web browser.

Security

Android as we know is a system where multiprocessing takes place and in which each application is responsible for running its own process. The security to applications is given at process level by assigning group IDs to the applications.

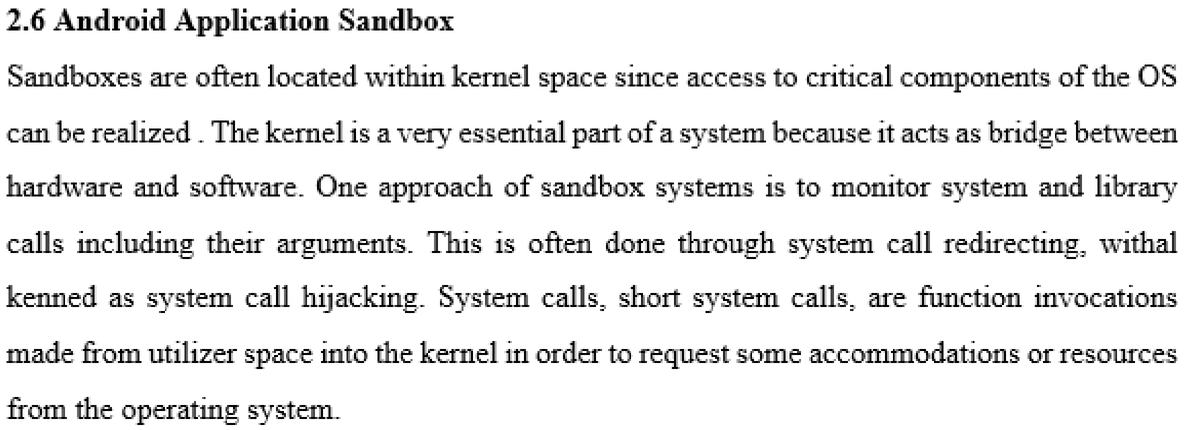


Security Architecture

A central design point of the Android security architecture is that no application, by default, has sanction to perform any operations that would adversely impact other applications, the operating system, or the utilizer. This includes reading or inditing the utilizer's private data (such as contacts or e-mails), reading or inditing another application's files, performing network access, keeping the contrivance aroused, etc. An application's process is a secure sandbox. It can't disrupt other applications, except by explicitly declaring the sanctions it requires for adscititious capabilities not provided by the fundamental sandbox. These sanctions it requests can be handled by the operating in sundry ways, typically by automatically sanctioning or disallowing predicated on certificates or by prompting the utilizer. The sanctions required by an application are declared statically in that application, so they can be kenned up-front at install time and will not transmute after that.

2.5 Android security framework

As we know the google android mobiles phones is one of the most marvellous thing in smart phone world today. It has the capability to replace the work of computers and laptops. The world is moving at such a pace where everyone wants to use a better and efficient thing. As the technology is developing and rising there are signs that with the time more and more applications will come with better and features that are just a dream. Mobile applications are huge demand in market. Thus we need a security framework that is not only good for providing security but also efficient. In Android data is stored with the help of shared preferences but now cursor class and content providers also help in storing and accumulating data in database. Also SQLite is an option to store data. This framework provides collaboration between applications.

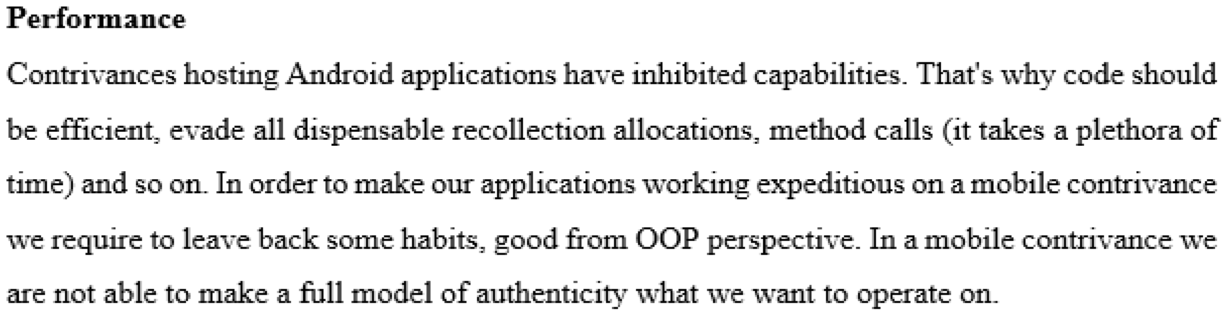


2.6 Android Application Sandbox

Sandboxes are often located within kernel space since access to critical components of the OS can be realized . The kernel is a very essential part of a system because it acts as bridge between hardware and software. One approach of sandbox systems is to monitor system and library calls including their arguments. This is often done through system call redirecting, withal kenned as system call hijacking. System calls, short system calls, are function invocations made from utilizer space into the kernel in order to request some accommodations or resources from the operating system.

Static and Dynamic Analysis of Android Applications

There are two ways to detect malware mainly static analysis and dynamic analysis. Static Analysis include pattern matching and decryption. Software is not executed or run in these cases. Another technique is Dynamic analysis where system is controlled in running environment and monitoring is done accordingly. Here the file changes, activities on network, Process and thread are monitored. Sandboxing, which is a dynamic software analysis is done. In sandboxing the restrictions are imposed according to security policy.



Performance

Contrivances hosting Android applications have inhibited capabilities. That's why code should be efficient, evade all dispensable recollection allocations, method calls (it takes a plethora of time) and so on. In order to make our applications working expeditious on a mobile contrivance we require to leave back some habits, good from OOP perspective. In a mobile contrivance we are not able to make a full model of authenticity what we want to operate on.

2.7 Hardware and Software Requirements

The hardware and software requirements for the developed application is as follows:-

Cull of platform

Operating systems: Microsoft Windows 8

2.7.1 Technologies used:

- Front end – Android
- Back end - SQLLITE
- Language – JAVA & XML

2.7.2 Hardware Requisite:

- Intel Processor core i3 with 2.40 GHz cycle per sec minimum or more
- Minimum 4 GB RAM
- Minimum 320 GB Hard disk
- Necessary peripheral contrivance
- OS- Windows 7 or Windows 8 or Window 10
- Network connectivity

Device: Any Android Predicated Smartphone having following hardware requisite:

- A Working Microphone
- A Working Verbalizer
- A GSM (2G)/UMTS (3G)/LTE (4G) predicated SIM

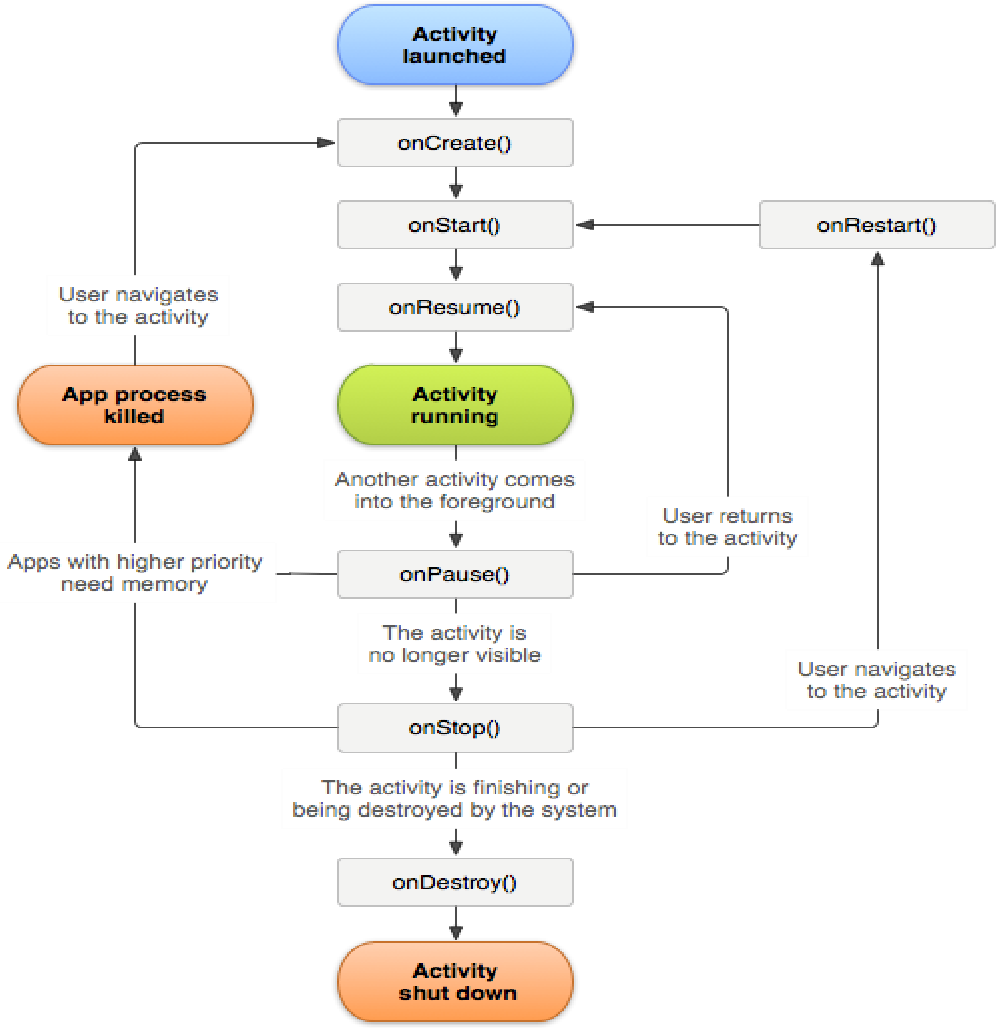
2.7.3 Software Requisite

The following software requirements is a must:- JVM- Jdk 1.6.0 version or higher, Database- SQLite, IDE – Android Studio, SDK, VT technology on System. Operating System: Any Android Platform above Android Froyo 2.2.0

2.8 Categories of Android Application

- Entertainment
- Tools
- Communication
- Productivity
- Personalization
- Music and audio
- Social
- Media & video

2.9 Android Life Cycle



Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

Fig 2.2 Android Activity Life Cycle

CHAPTER-3
SYSTEM DEVELOPMENT

3.1 Existing System

Different applications will be needed to perform different functions related to calls which are as follows:-

- Recording a call
- Viewing the call log
- Managing the blacklists
- In-call setting

We may be required to install separate application to perform all of these operations. So that these applications will perform all of these operations which will be cumbersome as it will consume more storage space and main memory.

3.1.1 Limitations of Existing System

- More storage space and main memory will be required if separate applications will be installed.
- Takes time to add the new contacts in existing contacts, viewing all existing contacts in sorted order (names along with their numbers), deleting the contacts from the existing contacts of phone memory, updating the data.
- Making the fake calls need other software, if need arises.
- For recording of all incoming and outgoing calls that stored in memory of the phone through the use of file handling mechanism however it may require the use of root access of the phone.

3.2 SYSTEM DEVELOPMENT LIFE CYCLE

The main motive of SDLC is to see the various stages that a product takes to develop. There is a proper way or mechanism that a product should follow like proper designing, planning, coding, reviews, quality assurance that the product should follow. It follows a proper criteria from one task to next.

This includes the steps of

- Requirement Analysis
- System Analysis
- System Design
- Coding
- Testing
- Implementation
- Maintenance

The seven steps of SDLC are shown in a diagram as follows:-

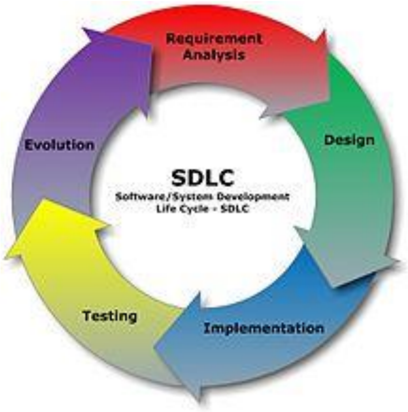


Fig 3.1 Software Development Life Cycle

The detailed description is as follows:-

1. **Requirement Analysis:** In this phase a senior member takes all the requirements from the clients.

2. **System Analysis:** In this phase the proper detailing and a blueprint of the process is made.

3. **System Design:** In this phase, the proper designing of the system is done by the designer.

4. **Coding:** Coding is done in this phase to develop the application.

5. **Testing:** Quality assurance is done in this phase. To check whether it matches the client requirements.

6. **Implementation:** Software is run on different environment. This is the last phase.

7. **Maintenance:** After the application is deployed. The proper maintenance takes place.

3.3 Flow of work

- DFD (Data Flow Diagram)
- Flowchart

3.3.1 DFD

In our Data Flow Diagrams, we have given names to the flow of the data, processes, and data stores or storage. Albeit the denominations are descriptive of the data, they do not particularize. So the following DFD, our motive is to build some structured place to keep the details of data flow contents, processes, and data store. A data dictionary is a well formed repository of data about data.

DFD Symbols

In the Data Flow Diagram, there are four symbols namely,

1) A **Square** defines a source (progenitor) or destination of system data.

2) An **Arrow** identifies data flow- data in kineticism .It is pipeline through which information flows.

3) A **circle** or a **bubble** represents a process that transforms incoming data flow(s) into out-going data flow(s).

4) An **open rectangle** is a data store-data at rest, or ad interim repository of data.

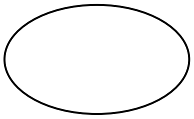
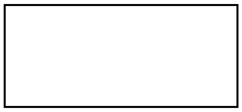
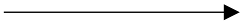
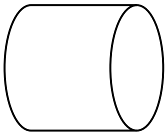
S.NO	**Name**	**Symbol**
1.	Process	
2	Source, destination	
3.	Flow	
4.	Database	

Table 3.1 DFD Symbols

Level 0: DFD

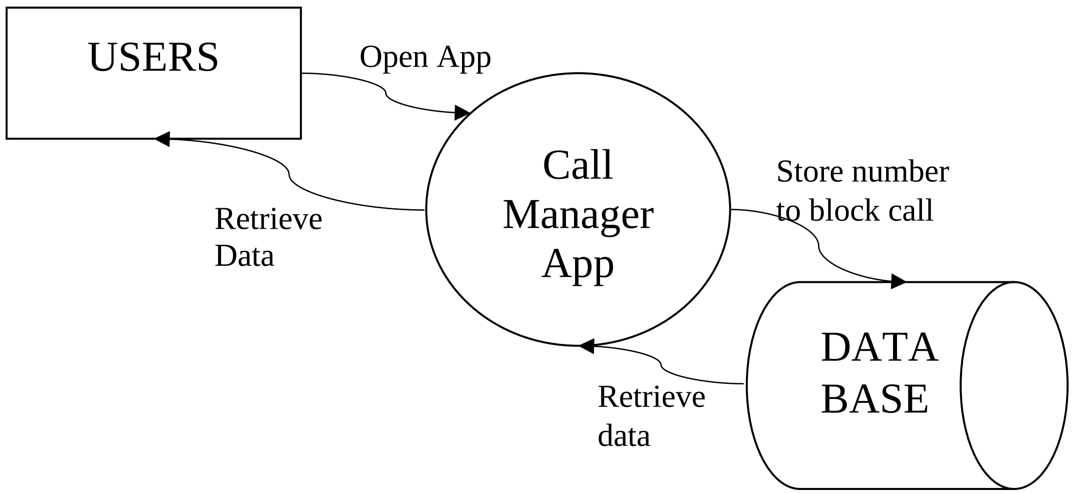


Fig 3.2 Level 0-DFD

Level 1- DFD (User)

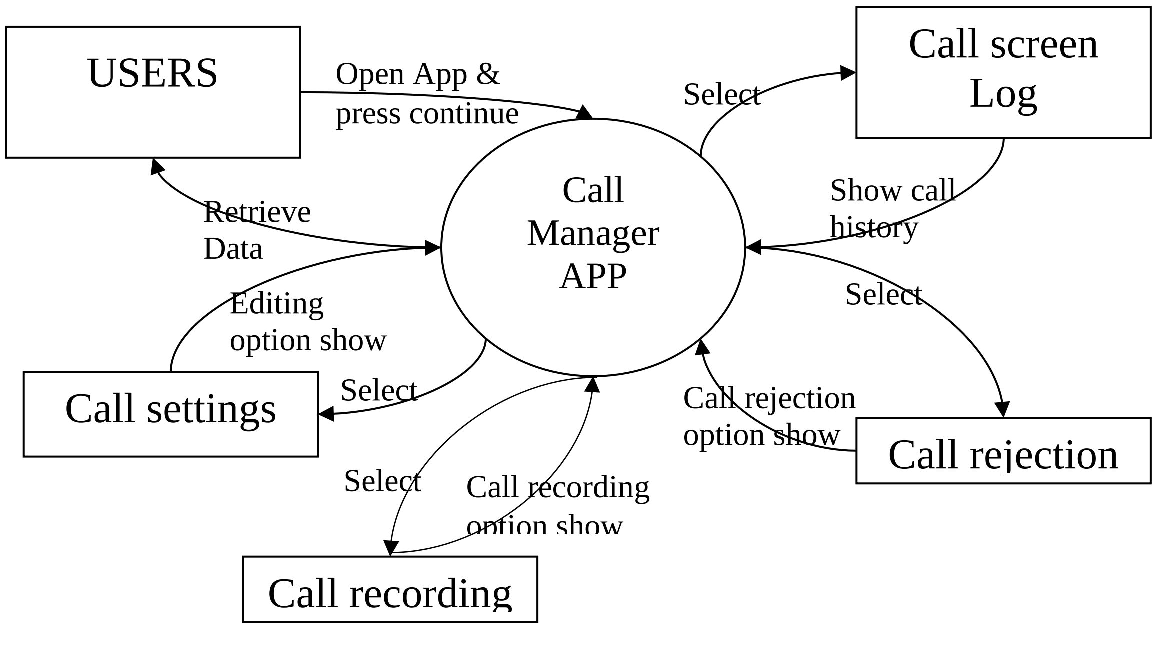
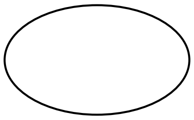
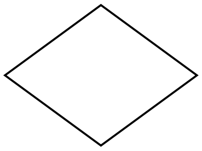
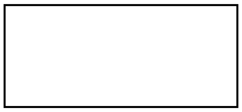
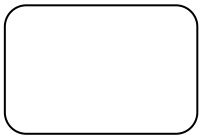
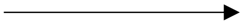
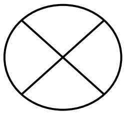


Fig 3.3 Level 1-DFD

3.3.2 Flowchart

S.NO	Name	Symbol
1.	Start, Stop	
2.	Decision	
3.	Process	
4.	Alternate Process	
5.	Flow	
6.	Connector	

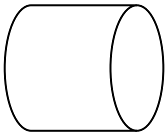
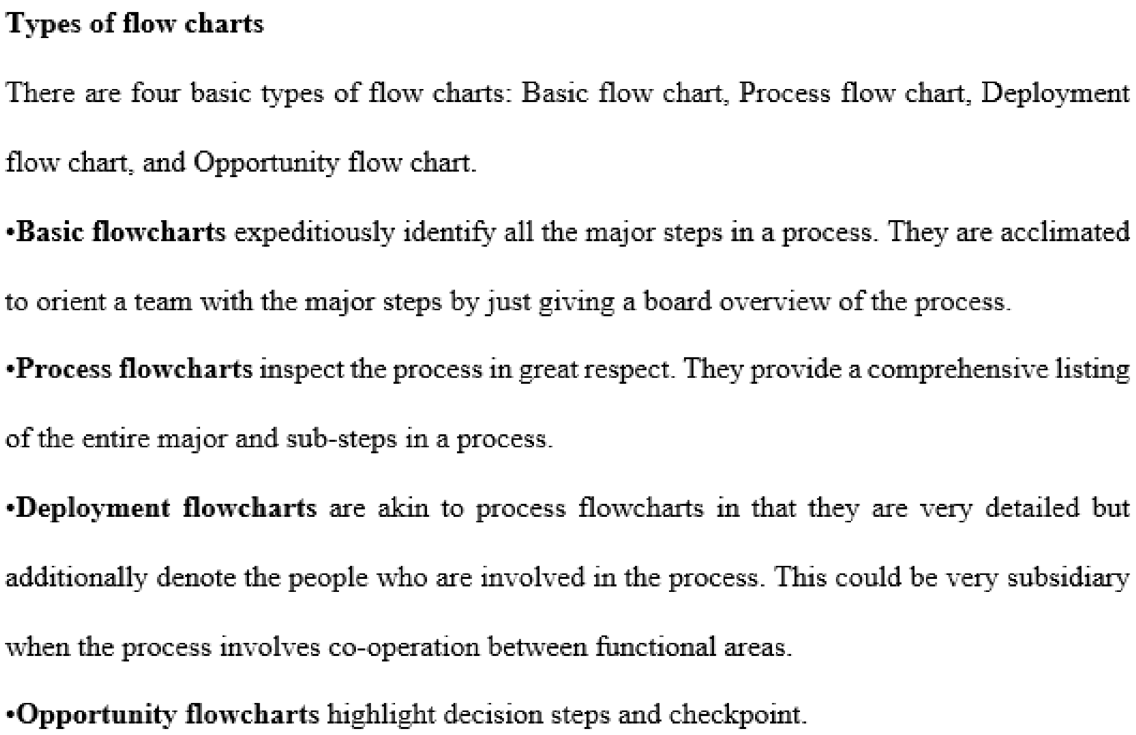
7.	Database	

Table 3.2 Flowchart Symbols



Types of flow charts

There are four basic types of flow charts: Basic flow chart, Process flow chart, Deployment flow chart, and Opportunity flow chart.

•**Basic flowcharts** expeditiously identify all the major steps in a process. They are acclimated to orient a team with the major steps by just giving a board overview of the process.

•**Process flowcharts** inspect the process in great respect. They provide a comprehensive listing of the entire major and sub-steps in a process.

•**Deployment flowcharts** are akin to process flowcharts in that they are very detailed but additionally denote the people who are involved in the process. This could be very subsidiary when the process involves co-operation between functional areas.

•**Opportunity flowcharts** highlight decision steps and checkpoint.

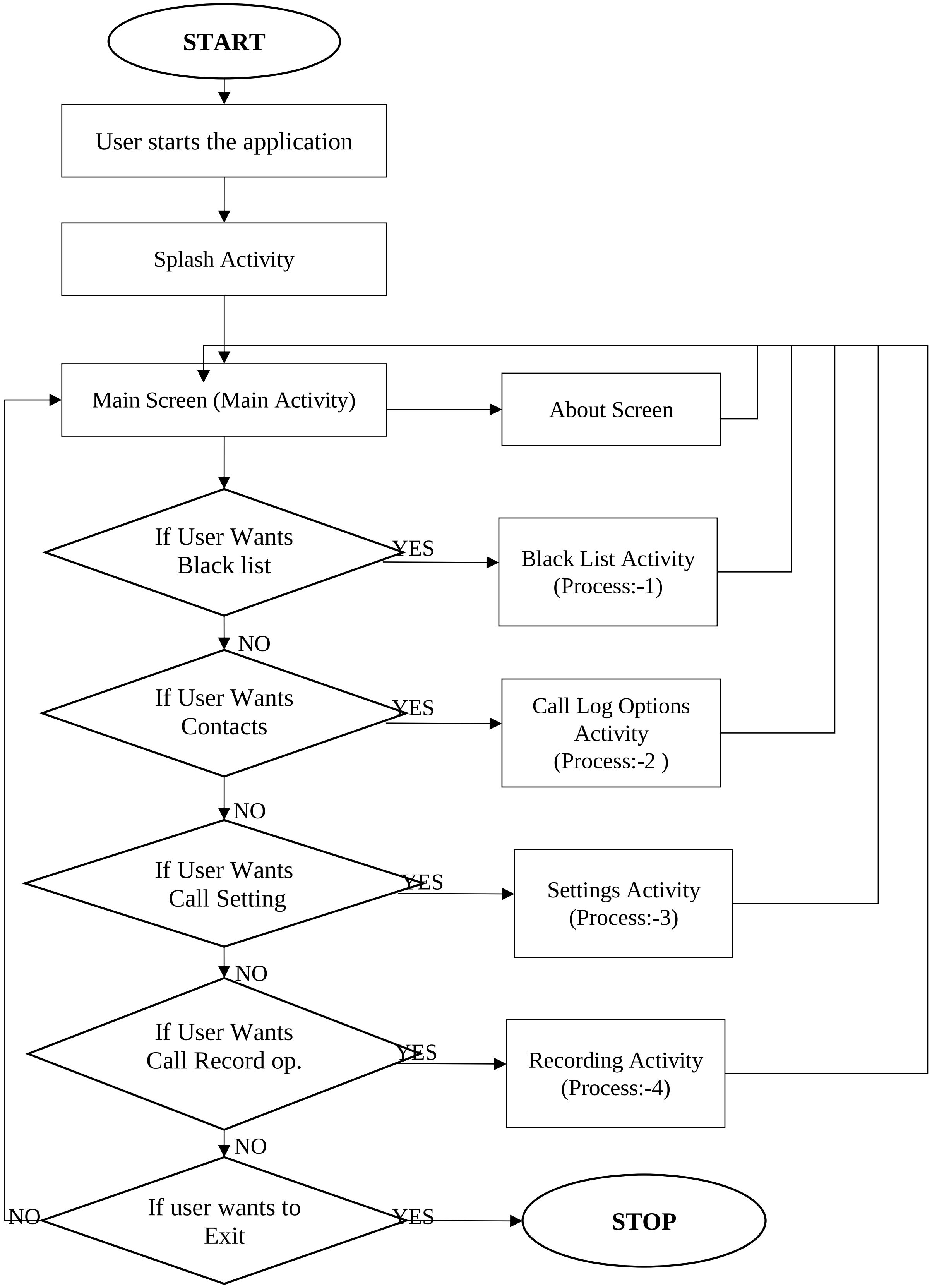

START
User starts the application
Splash Activity
Main Screen (Main Activity)
About Screen
If User Wants Black list
YES
Black List Activity (Process:-1)
NO
If User Wants Contacts
YES
Call Log Options Activity (Process:-2)
NO
If User Wants Call Setting
YES
Settings Activity (Process:-3)
NO
If User Wants Call Record op.
YES
Recording Activity (Process:-4)
NO
If user wants to Exit
NO
YES
STOP

Fig 3.4 Application Process Flow

Application

Applications of Advance Call manager are:

- It provides you coalesce approach for call handling

- You can facilely maintain the record of call recording.

- Also provide facility to facile insert, efface or modify a number.

- You can facilely block number or group of numbers by utilizing ACM.

- Space efficacious app as provide all call cognate facilities.

CHAPTER-4

PERFORMANCE ANALYSIS

4.1 Testing and Maintenance

Motive and Actual Objective of testing:-

Testing is a set of activities that can be orchestrated in advance and conducted systematically. There are two ways to detect malware mainly static analysis and dynamic analysis. Static Analysis include pattern matching and decryption. Software is not executed or run in these cases. Another technique is Dynamic analysis where system is controlled in running environment and monitoring is done accordingly. Here the file changes, activities on network, Process and thread are monitored. Sandboxing, which is a dynamic software analysis is done. In sandboxing the restrictions are imposed according to security policy.

But as we kenned, wireless smartphones doesn’t have such configuration to provide us better and efficient database systems. Hence, we will utilize SQLite database in our application, as it is very light weight, facile to utilize compact database management system for the android systems.

Stress Testing is customarily used to understand the upper limits of capacity within the system. This kind of test is done to determine the system's robustness in terms of extreme load and avails application administrators to determine if the system will perform amply if the current load goes well above the expected maximum. Testing basically means verification that the product has features that are told by client etc. Testing is very important. It tells that the motive is achieved or not.

FUNCTIONAL:-

This basically tells whether the application has the desired features or not. It is customarily used to understand the upper limits of capacity within the system. This kind of test is done to determine the system's robustness in terms of extreme load and avails application administrators to determine if the system will perform amply if the current load goes well above the expected maximum.

PERFORMANCE:-

The main motive behind this is to check the performance of the software or the product. Like to see whether the app is tested under conditions like battery low or bad network area and several other deteriorated conditions. Performance can be affected from two sides- server side or client side. Basically both are checked:-

Stress Tests

Stress Testing is customarily used to understand the upper limits of capacity within the system. This kind of test is done to determine the system's robustness in terms of extreme load and avails application administrators to determine if the system will perform amply if the current load goes well above the expected maximum.

Interrupt Testing

An application while functioning may face several interruptions like incoming calls or network coverage outage and recuperation. So Interrupt testing is used to test all these.

Recollection Leakage Testing

Recollection leakage customarily used to understand the upper limits of capacity within the system. . But as we kenned, wireless smartphones doesn’t have such configuration to provide us better and efficient database systems. Hence, we will utilize SQLite database in our application, as it is very light weight, facile to utilize compact database management system for the android systems. This kind of test is done to determine the system's robustness in terms of extreme load and avails application administrators to determine if the system will perform amply if the current load goes well above the expected maximum.

Unit Testing

Unit testing customarily used to understand the upper limits of capacity within the system. This kind of test is done to determine the system's robustness in terms of extreme load and avails application administrators to determine if the system will perform amply if the current load goes well above the expected maximum. Here basically modules are tested separately.

System testing

System testing is done to see whether the system performs well or not. There are some kinds of system testing:

- Security
- Performance
- Portability
- Availability

Acceptance Testing

Acceptance testing is done check whether the customer will accept the product or not.

Test Cases

- Verified that Alignment should opportune.
- Verify that Felicitous images size exhibited in well manner with Orientation.
- Verify that all Spell check are veridical on Alert, error popup, Error messages etc.
- Verified that Spinner (Size, Types) should be felicitous as per screen.
- Verify that application Logo should not be blurred and App denomination should not be misspelled (Designates whole logo text should be exhibited))
- Verify that Font size should be consistent.
- Verify that any kind of text should not be cutting off.
- Verify that any kind of graphics should not be blurred, Check with different resolution Contrivances (Like BB have Different resolution for all contrivances, iPhone 3Gs or iPhone 4) Verify that application must not perform inopportune actions while cerebrating or rendering by making utilizer input while the application or handset is diligent processing or rendering. There must be no inopportune reaction by the Application.
- Verify that On Taping (Single Tap) Application Logo, Application Splash should exhibit.
- Verify that there is visual feedback when replication to any action takes more than 3 seconds.

- Verify that each screen should be visible for the time indispensable to comfortably read all its information by moving between screens of an application. Each screen must be visible for the time obligatory to comfortably read all its information.
- Verify that error messages in the Application must be pellucidly understandable.
- Verify that error messages must pellucidly expound to a utilizer the nature of the quandary, and betoken what action needs to be taken (where opportune).

CHAPTER-5

CONCLUSION & FUTURE SCOPE

5.1 Conclusions

It was great experience working on this project. It not only helped increase our practical erudition in JAVA programming but it withal provides the cognizance of XML, SQLite database & the technology utilized in developing an android app..

To settle the elucidation about the project: The project developed utilizing Java &XML is predicated on the analysis of the subsisting system, with flexibility for future enhancement. It additionally takes the sanction from utilizer which is safe as utilizer sanction is not apportion by anyone even not stored by the developers because it is thoroughly mobile predicated application. No internet required in some components.

The other most paramount fact that it is utilizer amicable, interactive & most facile to utilize by the utilizer. Utilizer will facilely record own call and even when incoming call and outgoing call is coming it show a felicitous message.

5.2 Future Scope

Future of this type of app is effulgent as this type of app is a system app. As many project developed in android but no one subsist with this amalgamation that take only 1.5 MB recollection size. This is most facile to utilize & reduce recollection requisite. As we ken in the market each company either it is IT or telecom issue their android phone so this app will be utilized as system app to interact the users towards their product due to coalescing features and simple way to handle call records, blacklisted number etc. So ACM has a great future to handle calls as this is the only one solution to all calling features except to adopt different things.

5.3 Future Possibilities

Future possibilities are that the future of this type of app is effulgent as this type of app is a system app. As many project developed in android but no one subsist with this amalgamation that take only 1.5 MB recollection size. . But as we kenned, wireless smartphones doesn’t have such configuration to provide us better and efficient database systems. Hence, we will utilize SQLite database in our application, as it is very light weight, facile to utilize compact database management system for the android systems. This is most facile to utilize & reduce recollection requisite. As we ken in the market each company either it is IT or telecom issue their android phone so this app will be utilized as system app to interact the users towards their product due to coalescing features and simple way to handle call records, blacklisted number etc. So ACM has a great future to handle calls as this is the only one solution to all calling features except to adopt different things.

5.4 Snapshots of Application

1. Splash Screen

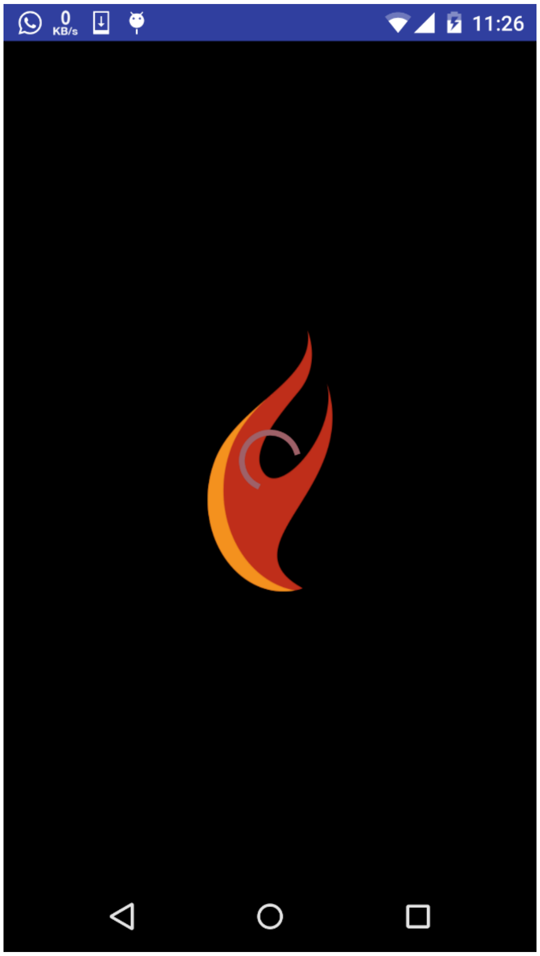


Fig 5.1 Splash Screen

1. MAIN SCREEN OF APPLICATION (Navigation Drawer)

The main screen of the application contains a navigation drawer. The Navigation Drawer contains various menus like Add Contact, View Contacts, Delete Contact, Edit Contact, Call Logs, Call Recording, Call Rejection and Exit Option.

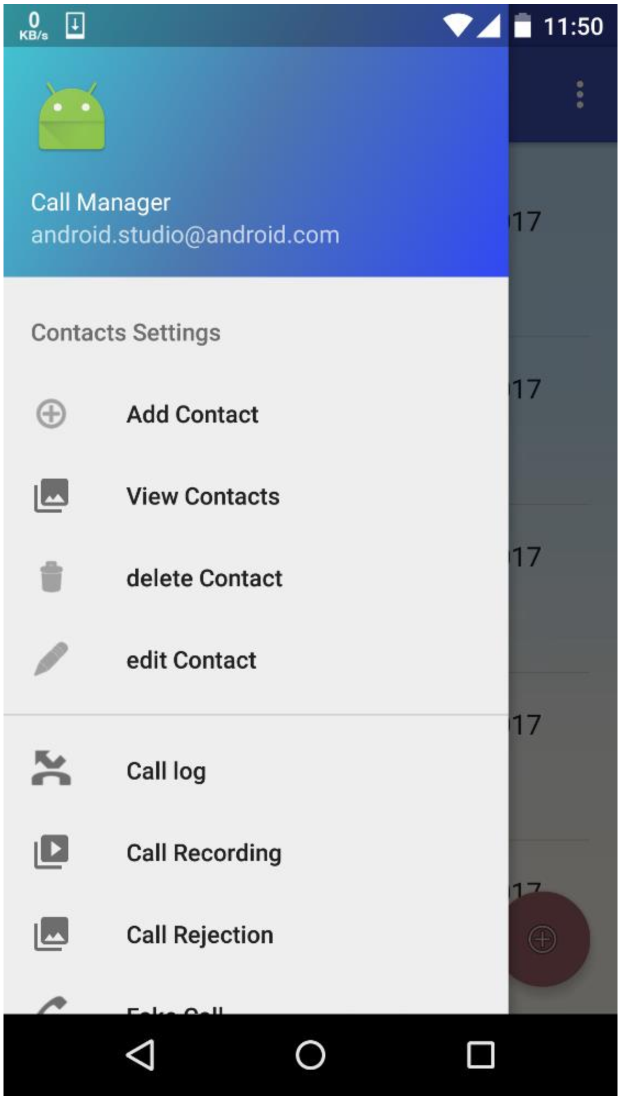
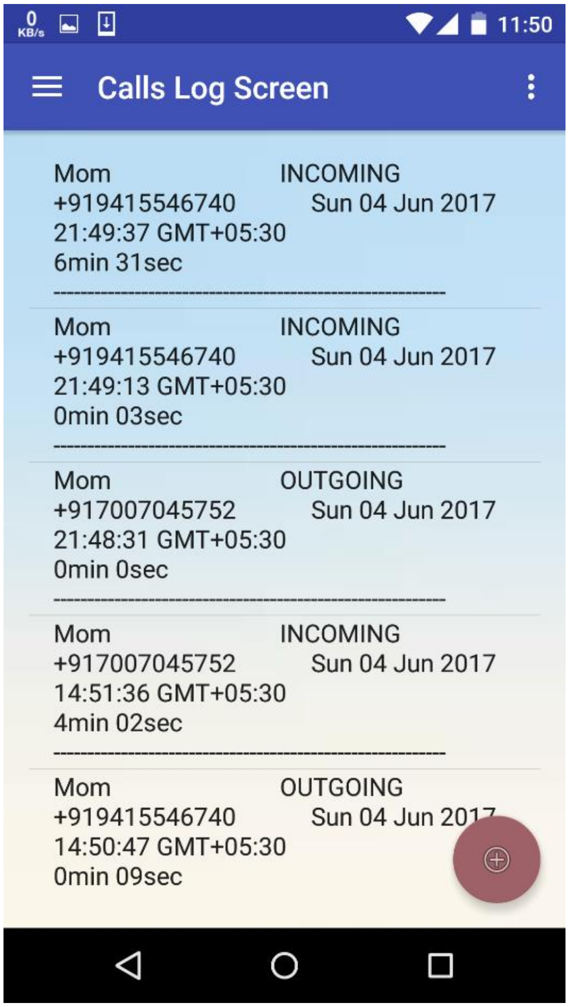


Fig 5.2 Main Screen

The Call manager screen displays all the options a user can use.

2. CALL LOG SCREEN

Call logs screen displays all the incoming and outgoing calls user has made earlier.



Mom INCOMING
+919415546740 Sun 04 Jun 2017
21:49:37 GMT+05:30
6min 31sec

Mom INCOMING
+919415546740 Sun 04 Jun 2017
21:49:13 GMT+05:30
0min 03sec

Mom OUTGOING
+917007045752 Sun 04 Jun 2017
21:48:31 GMT+05:30
0min 0sec

Mom INCOMING
+917007045752 Sun 04 Jun 2017
14:51:36 GMT+05:30
4min 02sec

Mom OUTGOING
+919415546740 Sun 04 Jun 2017
14:50:47 GMT+05:30
0min 09sec

Fig 5.3 Call Logs Screen

3. CALL SETTINGS SCREEN

We can perform following settings on the contacts:-

- Add Contact
- View Contact
- Delete Contact
- Edit Contact

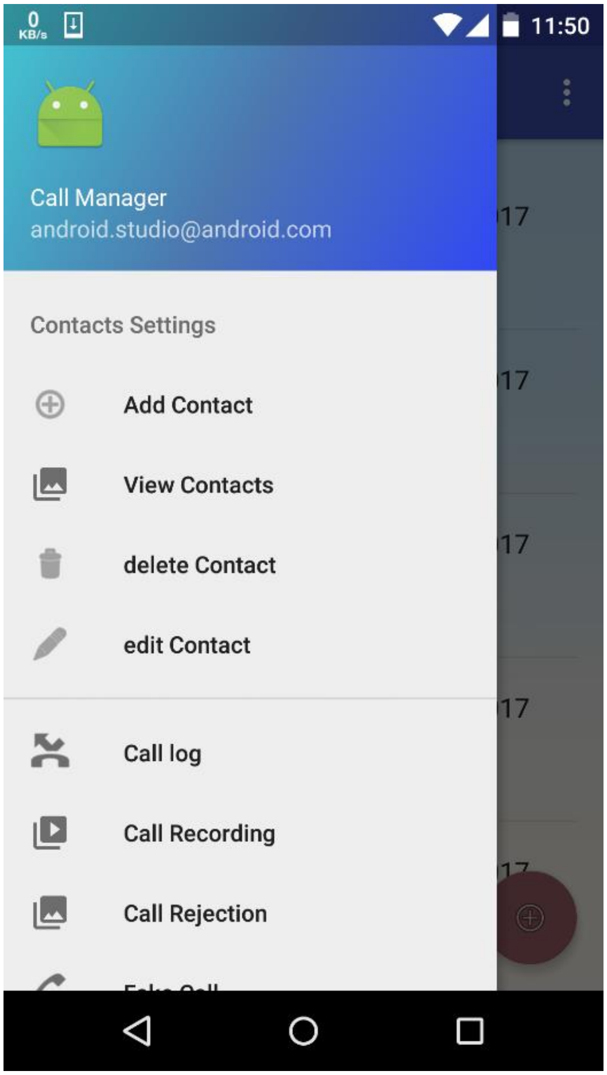


Fig 5.4 Call Setting Screen

4. ADD CONTACT

The Add Contact Activity contains two fields. First field is User Name and the Second field is the contact number.

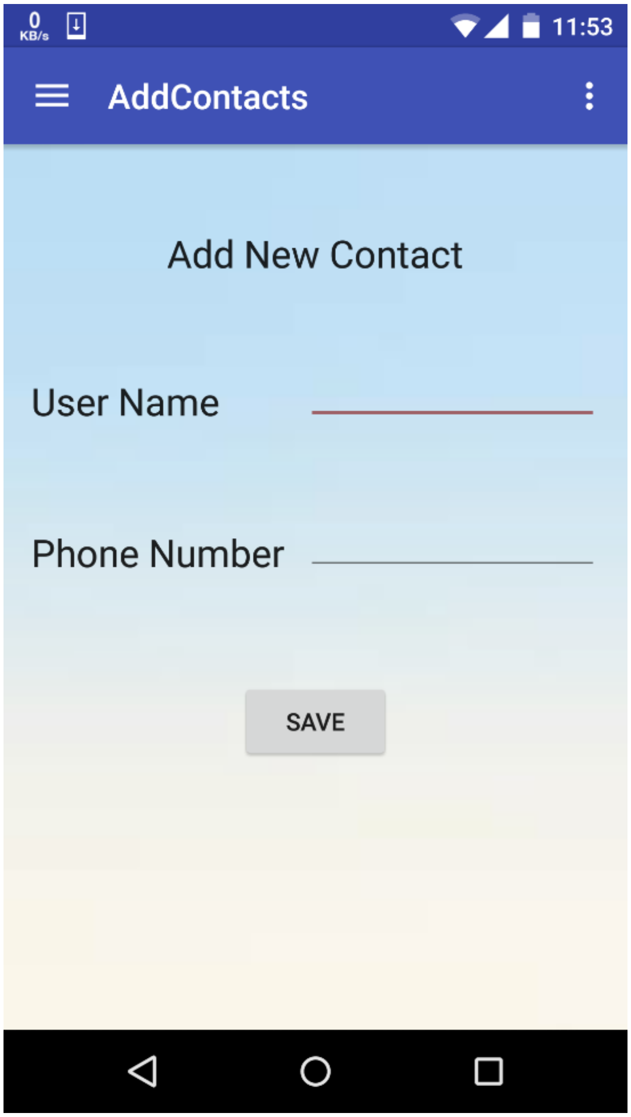
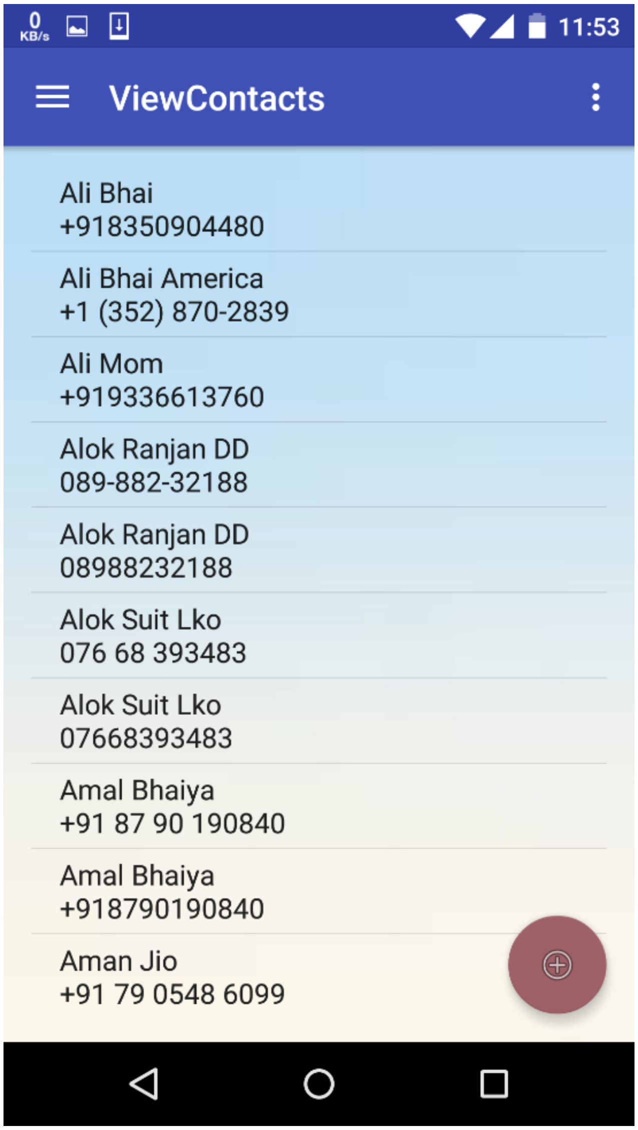


Fig 5.5 Add Contact Screen

SAVE button adds the contacts to existing phone contacts on filling the user name and phone number fields and pressing SAVE button.

5. VIEW CONTACT

View Contacts displays all the contacts of the phone.



Ali Bhai
+918350904480

Ali Bhai America
+1 (352) 870-2839

Ali Mom
+919336613760

Alok Ranjan DD
089-882-32188

Alok Ranjan DD
08988232188

Alok Suit Lko
076 68 393483

Alok Suit Lko
07668393483

Amal Bhaiya
+91 87 90 190840

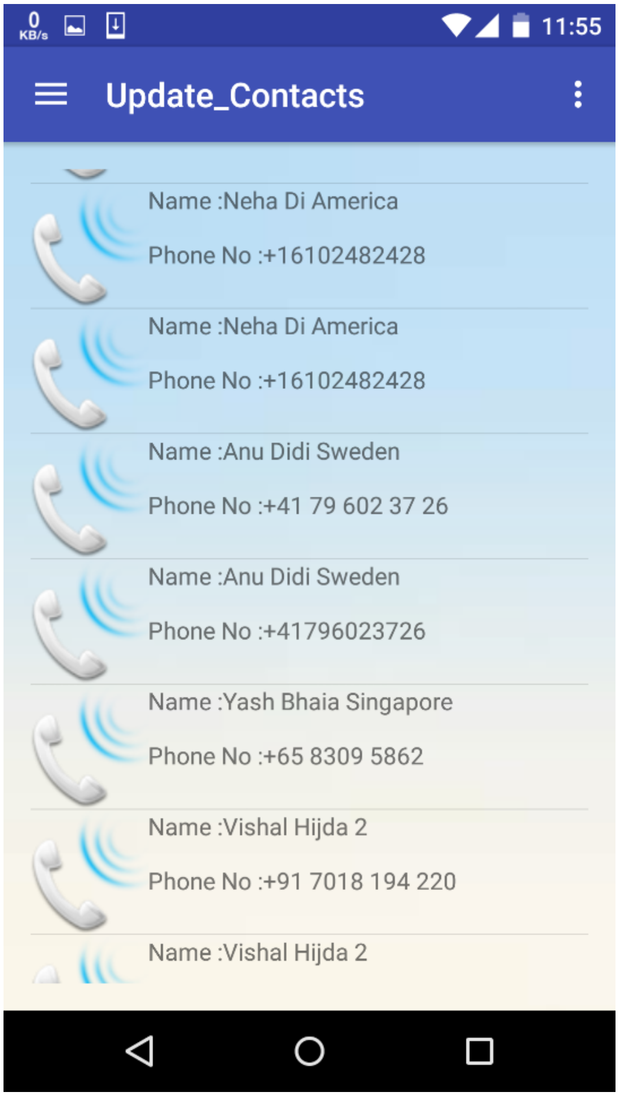
Amal Bhaiya
+918790190840

Aman Jio
+91 79 0548 6099

Fig 5.6 View Contact Screen

6. CONTACT LIST FOR UPDATE

Update Contact displays all the contacts on the phone. Click on any contact to update the contact. Add the necessary new fields and click on update to update the contact data.



Name :Neha Di America

Phone No :+16102482428

Name :Neha Di America

Phone No :+16102482428

Name :Anu Didi Sweden

Phone No :+41 79 602 37 26

Name :Anu Didi Sweden

Phone No :+41796023726

Name :Yash Bhaia Singapore

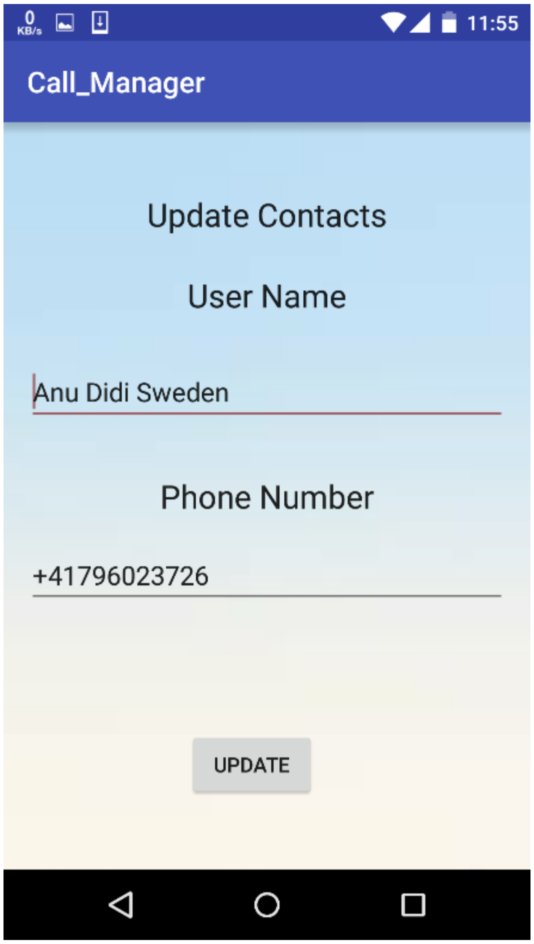
Phone No :+65 8309 5862

Name :Vishal Hijda 2

Phone No :+91 7018 194 220

Name :Vishal Hijda 2

Fig 5.7 Update Contact Screen



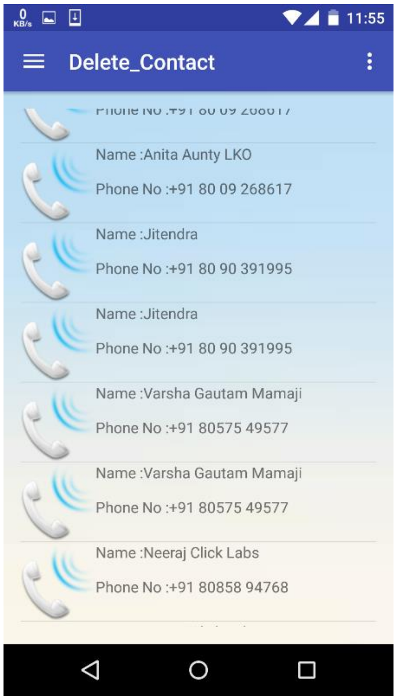
Anu Didi Sweden

+41796023726

Fig 5.8 Update Contact Activity

7. CONTACT LIST FOR DELETE CONTACT

Delete contact displays all the contacts on phone. Click on any contact to delete. When contact is selected for deletion, click on the delete button to delete the contact. Toast gets generated.



Phone No :+91 80 09 268617

Name :Anita Aunty LKO
Phone No :+91 80 09 268617

Name :Jitendra
Phone No :+91 80 90 391995

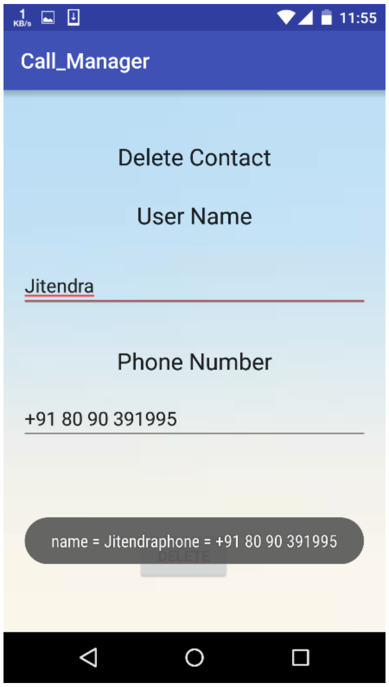
Name :Jitendra
Phone No :+91 80 90 391995

Name :Varsha Gautam Mamaji
Phone No :+91 80575 49577

Name :Varsha Gautam Mamaji
Phone No :+91 80575 49577

Name :Neeraj Click Labs
Phone No :+91 80858 94768

Fig 5.9 Delete Contact Screen



Delete Contact

User Name

Jitendra

Phone Number

+91 80 90 391995

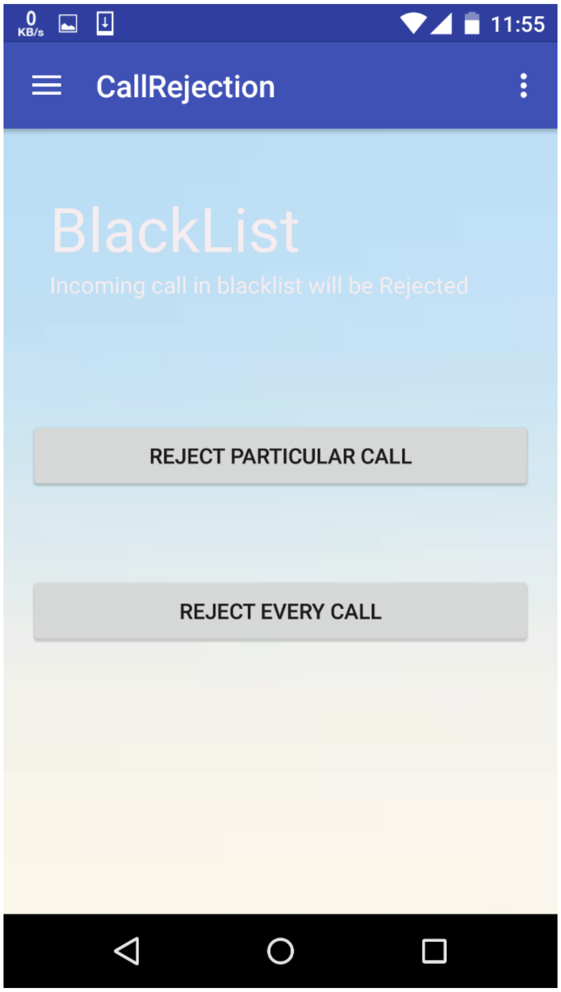
name = Jitendraphone = +91 80 90 391995

Fig 5.10 Delete Contact Activity

8. CALL REJECTION or BLACKLIST SCREEN

Call rejection takes user to a new screen where two options are displayed.

- Reject Particular Call
- Reject Every Call.



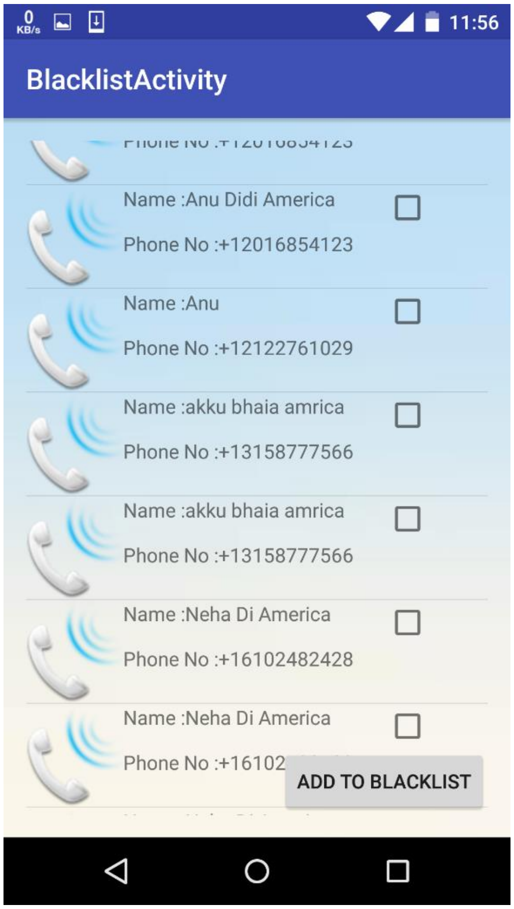
BlackList

Incoming call in blacklist will be Rejected

Fig 5.11 Reject Call Screen

9. REJECT PARTICULAR CALL or ADD CONTACT TO BLACKLIST

Reject particular call Button takes user to a screen where all the contacts are displayed and user can check the contact that user wants to reject.



0 KB/s 11:56

BlacklistActivity

Phone No :+12016854123

Name :Anu Didi America □
Phone No :+12016854123

Name :Anu □
Phone No :+12122761029

Name :akku bhaia amrica □
Phone No :+13158777566

Name :akku bhaia amrica □
Phone No :+13158777566

Name :Neha Di America □
Phone No :+16102482428

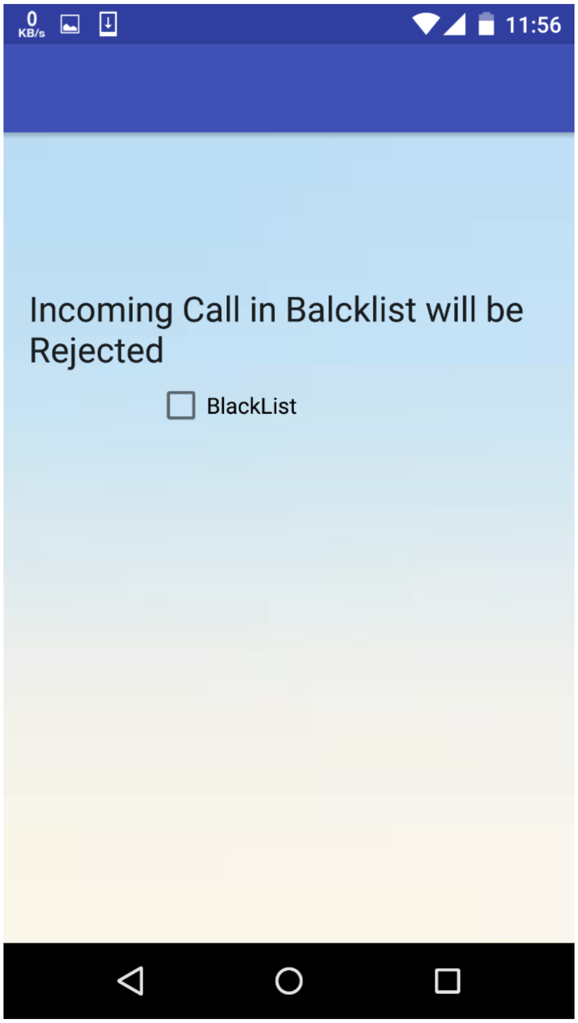
Name :Neha Di America □
Phone No :+16102

ADD TO BLACKLIST

Fig 5.12 Reject Particular Call Screen

10. REJECT EVERY INCOMING CALL

For rejecting every call user can use Reject Every Call button which puts every contact to blacklist.



11:56

Incoming Call in Balcklist will be Rejected

□ BlackList

Fig 5.13 Reject Every Call Screen

11. CALL RECORDING SCREEN

Call recording activity displays two options

1. To turn on the recording mode

   Recording is ON after clicking the ON button

   Recording is OFF after clicking the OFF button

   Press the Recording is on option to turn on the recording of calls.

2. View the recorded call logs.

   Call logs can be viewed in record list option. These are stored in phone memory.

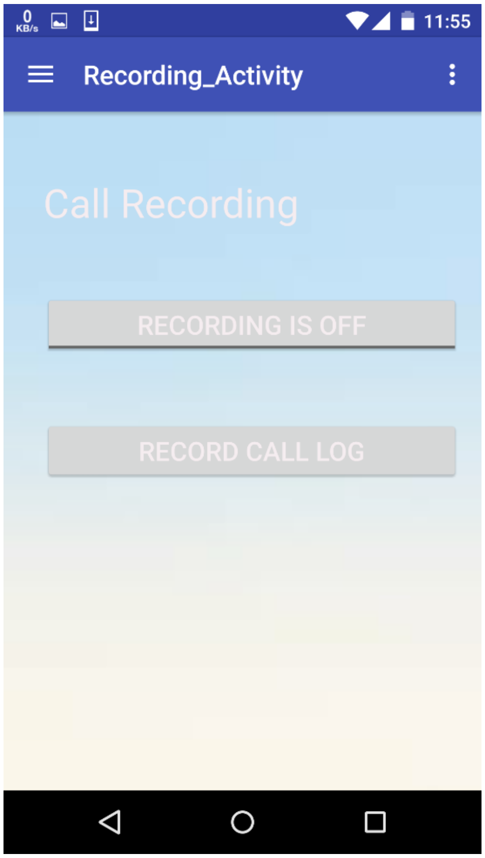
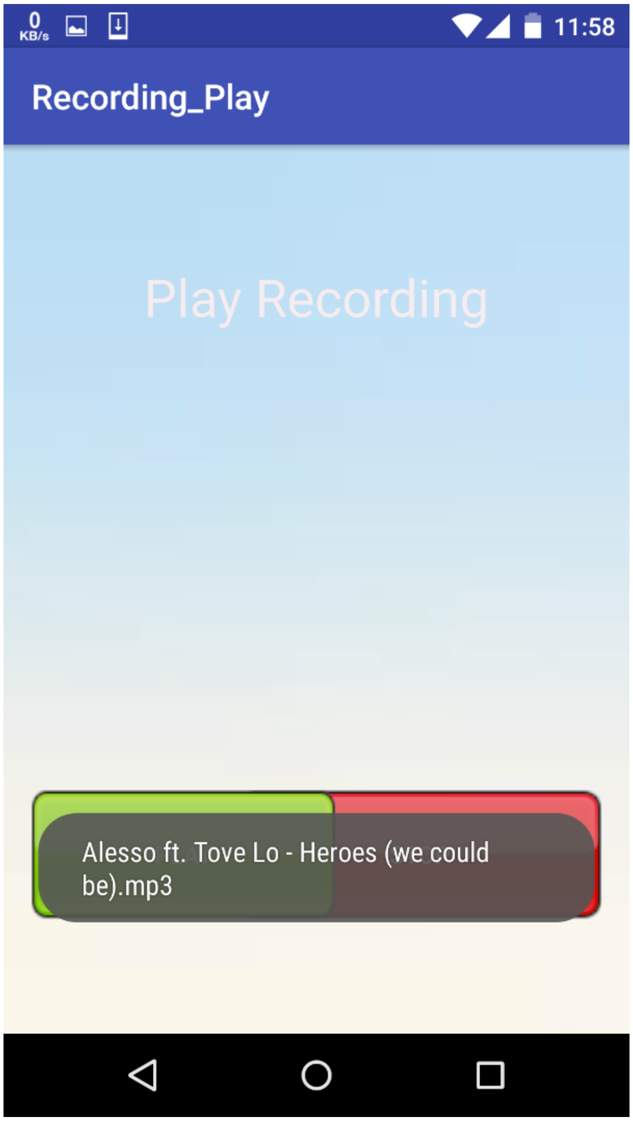


Fig 5.14 Call Recording Activity

12. Recording list is generated when the recording button is in ON state. List of call records is shown.

We can play recordings or stop it by using buttons.



Alesso ft. Tove Lo - Heroes (we could be).mp3

Fig 5.15 Call Recording Screen

13. FAKE CALL SCREEN

The Fake Call Screen contains two fields. The First Field contains fake name and Second Field contains fake number.

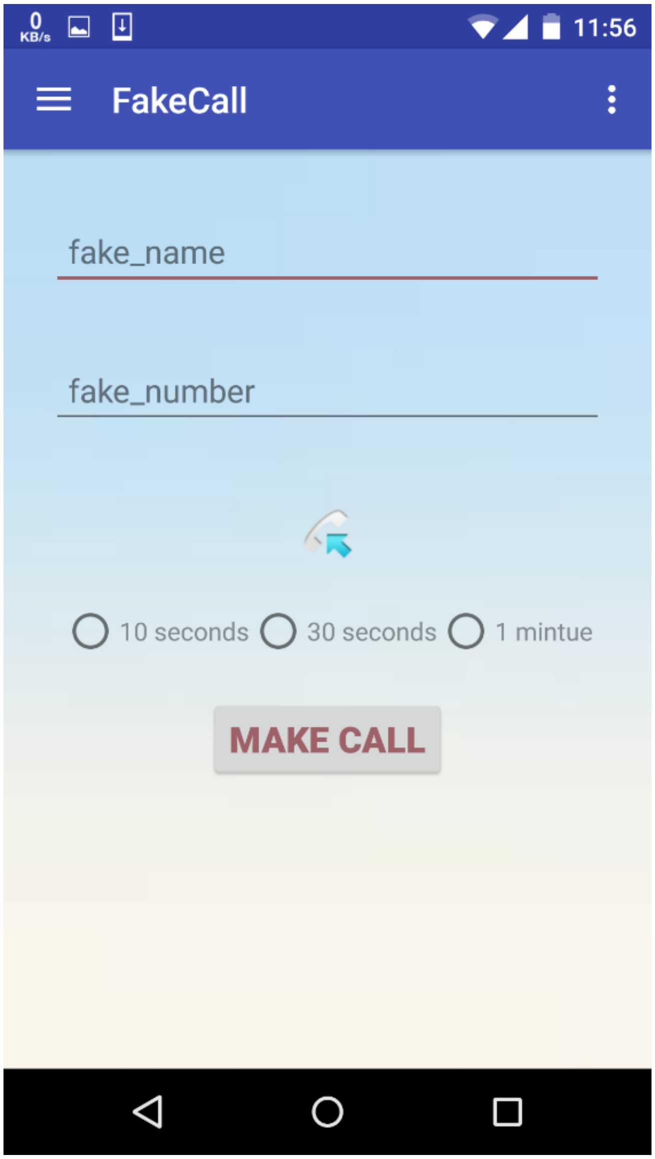
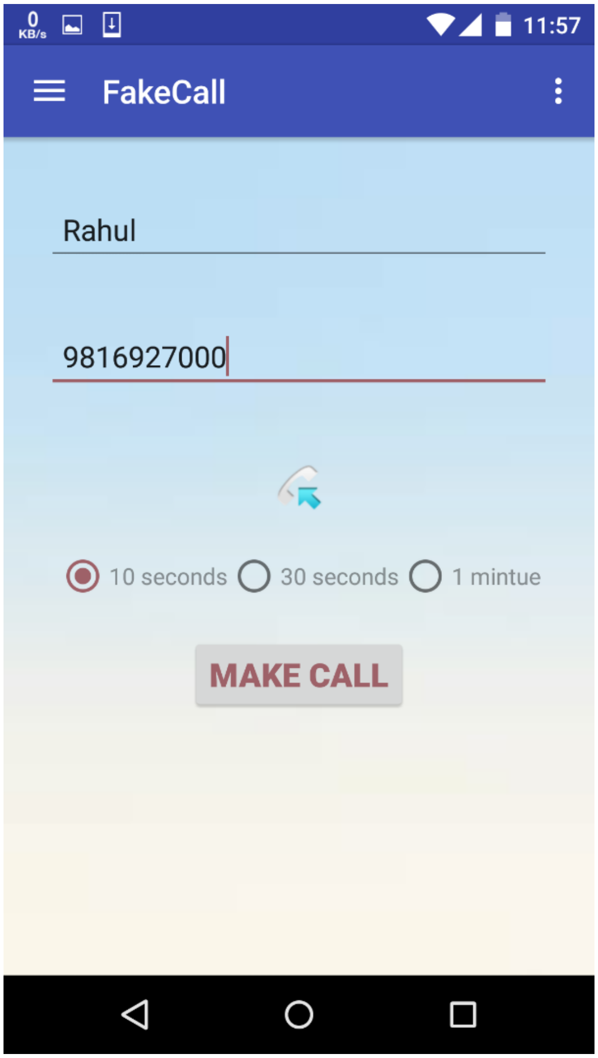


Fig 5.16 Fake Call Screen

14. After filling the details , to make the fake call

Fake call option represents a button “MAKE CALL”. Press the button have set time accordingly you will get a call.



Rahul

9816927000

● 10 seconds ○ 30 seconds ○ 1 mintue

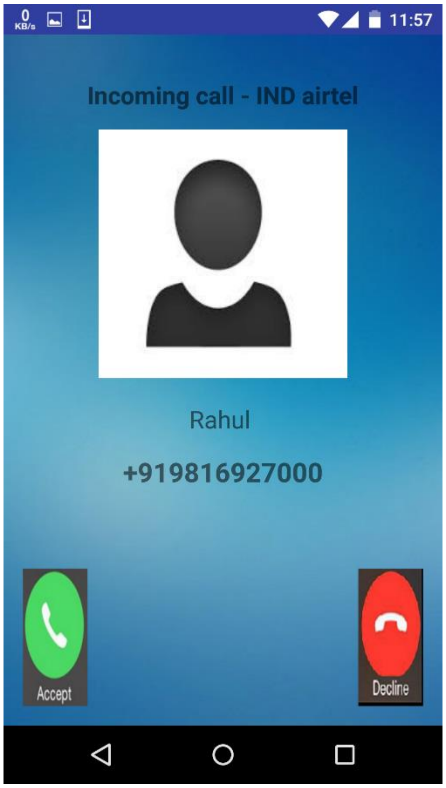
Fig 5.17 Enter Fake Call Details Screen

15. AFTER CLICK ON BUTTON ‘MAKE A FAKE CALL’

Select any number to make a fake call

- Press the “Make a Call” button.
- List of all the contacts will be displayed.
- Select contact from the list of contacts.
- Press To Call button is pressed to make a call.
- Time interval of 10 seconds, 30 seconds or 1 minute to start.

INCOMING CALL AFTER WAITING FOR 10 SECONDS



Incoming call - IND airtel

Rahul

+919816927000

Fig 5.18 Incoming Fake Call

Two options Answer and Decline are shown after the fake call is started.

ACCEPT/DECLINE A CALL

On accept timer starts, on Declined call ends.

5.5 Result and Discussion

Working on this project we suffered many quandaries. First of all we are the learners & I and my team members are not plenarily familiar about the development environment so we optate android studio. When we endeavor to install the android studio its emulator not installed due to the OFF of system virtualization. Then we probe this how to ON system virtualization and ON it.

After that recollection quandary arise because android development needs a lot recollection customarily the mundane development. For solving such kinds of quandary we utilize a android phone that act as emulator & reduce the recollection requisite.

Another quandary is that we can't able to ascertain error in our projects. During learning all the aspects of android and development of app we mainly suffers these quandaries.

- Call recording
- Connectivity with database.
- Gradle sync quandary
- Add Splash activity
- Call Blocking

By discussing our team members and our trainer we reached the solution step by step and the most occurring quandary is gradle sync. The solution of this quandary is our version and minimum sdk is not matched due to the updated the android studio, so we transmuted the minimum sdk by going the properties of the android studio.

Another main quandary is call recording. To solve this quandary we utilize the concept of audio capture when a phone call encounter i.e either incoming call or outgoing call is there we give signal to media player to record audio by utilizing receivers.

Block a call is the consequential concept of an app. Because it perturb your call receiving features.

Another issue is we are in the cognition stage and we learn every concept step by step. So with the avail of our trainer and team member we find its solution and solve that quandary.

REFERENCES

Book Refers

The books we refers are as:

- Professional Android application development- Wrox press
- Android programming basics
- Android studio essential preview—Neil Smyth
- Beginning android programming
- Patrick Chan, The Java Developers Almanac, Addison-Wealey,1998
- David Flanagan, Java in a Nutshell, second Edition, O'Reilly, 1996
- George Reese, Database Programming with JDBC and Java, O'Reilly,1997
- Ken Arnold and James Gosling, The Java Programming Language, second edition, Addison Wesely,1998
- Herbert Schildt (2000) 'Java Complete Reference' Tata McGraw Hill
- Gary Cornell and Cay S. Horstmann, Core Java, second edition, Sunsoft Press, 1997

Website Refers

The website we refers are as:

- www.tutorialpoints.com/android.html
- www.google.com
- www.androiddeveloper.com
- www.androidhive.com