BLIND COMMUNICATOR

A Project Report submitted for the partial fulfilment of the requirement for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

SUBMITTED
BY

K.MAMATHA
Roll. No. 18699F0049

Under the Guidance of
Mr. V. MARUTHI PRASAD
Assistant Professor

DEPARTMENT OF COMPUTER APPLICATIONS
MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE
MADANAPALLE
UGC AUTONOMOUS
(Approved by AICTE, New Delhi & Affiliated to JNTUA, Anathapuramu)
www.mits.ac.in

2019-2020

MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE MADANAPALLE

(UGC AUTONOMOUS)

(Approved by AICTE, New Delhi & Affiliated to JNTUA, Ananthapuramu)

DEPARTMENT OF COMPUTER APPLICATIONS

BONAFIDE CERTIFICATE

This is to certify that the project work entitled "**BLIND COMMUNICATOR**" is a bonafide work carried out by **K.MAMATHA**, **Roll No. 18699F0049,** submitted in the partial fulfilment of the requirements for the award of the degree of Master of Computer Applications in Madanapalle Institute of Technology & Science, Madanapalle, affiliated to Jawaharlal Nehru Technological University Anantapur, Anantapuramu during the academic year 2019 -2020.

PROJECT GUIDE	**HEAD OF THE DEPARTMENT**
Mr. V. MARUTHI PRASAD,MCA.,M.E	**Dr.V.L.Pavani,MCA.,M.Phil, Ph.D**
Assistant Professor	Assistant Professor

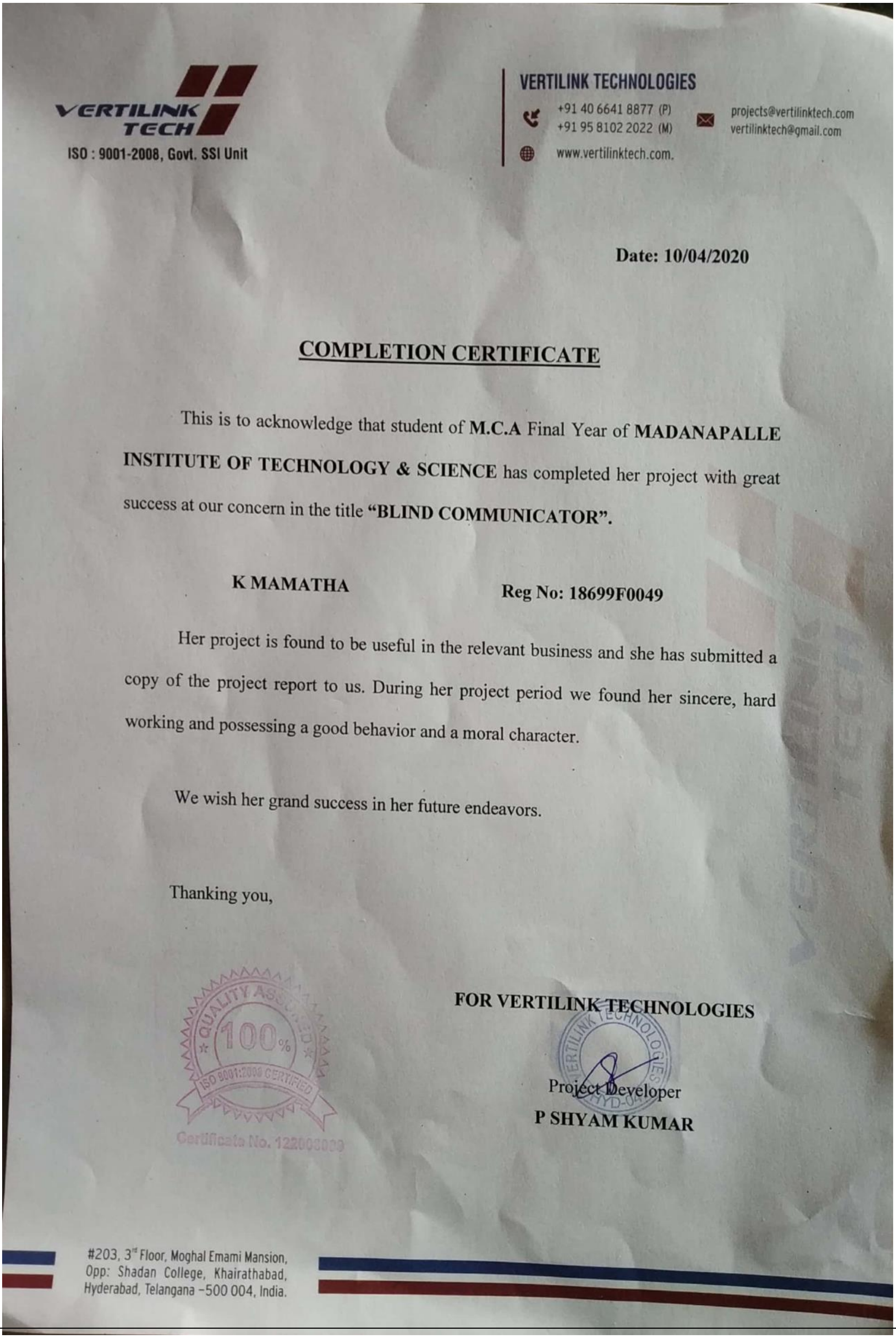
Submitted for viva voce examination held on ______________

Internal Examiner	**External Examiner**
Date:	Date:

DECLARATION

I, **K.MAMATHA (Roll No. 18699F0049)** hereby declare that the project entitle "**BLIND COMMUNICATOR**" is done by me under the guidance of **Mr. V. MARUTHI PRASAD** submitted in partial fulfilment of the requirements for the award of degree of Master of Computer Applications at MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE Madanapalle, affiliated to Jawaharlal Nehru Technological University Anantapur, Anantapuramu during the academic year 2019-2020. This work has not been submitted by anybody towards the award of any degree.

Date:	**Signature of the Student**
Place: Madanapalle	**K. MAMATHA**
	(Roll No. 18699F0049)



VERTILINK TECH
ISO : 9001-2008, Govt. SSI Unit

VERTILINK TECHNOLOGIES
☎ +91 40 6641 8877 (P)
+91 95 8102 2022 (M)
✉ projects@vertilinktech.com
vertilinktech@gmail.com
www.vertilinktech.com.

Date: 10/04/2020

COMPLETION CERTIFICATE

This is to acknowledge that student of **M.C.A** Final Year of **MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE** has completed her project with great success at our concern in the title **"BLIND COMMUNICATOR".**

K MAMATHA **Reg No: 18699F0049**

Her project is found to be useful in the relevant business and she has submitted a copy of the project report to us. During her project period we found her sincere, hard working and possessing a good behavior and a moral character.

We wish her grand success in her future endeavors.

Thanking you,

QUALITY ASSURED
100%
ISO 9001:2008 CERTIFIED
Certificate No. 122008009

FOR VERTILINK TECHNOLOGIES

VERTILINK TECHNOLOGIES
HYD-04

Project Developer
P SHYAM KUMAR

#203, 3rd Floor, Moghal Emami Mansion,
Opp: Shadan College, Khairathabad,
Hyderabad, Telangana –500 004, India.

ACKNOWLEDGEMENT

At the outset, I offer our sincere thanks and deep gratitude to **Mr. V. Maruthi Prasad.** I had the privilege to receive great inspiration and guidance from him, which helped us to set the objectives of this study and complete the report, I fortunate to have constant encouragement and practical guidance in all aspect of study. His experience in this line has been a boon to us in this respect.

I express my deep gratitude and thanks to the project Coordinator., **Dr. S. G. Shrinivas, M.C.A., Ph.D.** Assistant Professor, Department of Computer Applications, for his valuable suggestions during the project work. I am very much indebted to him for being a role model and for providing necessary guidance throughout the project.

I wish to express my grateful thanks to **Dr.V.L. Pavani., M.C.A., M.Phil., Ph.D**., Head of the Department of Computer Applications for hearty advice and guidance without with the entire course work would not be fruitful.

I am thankful to the principal, **Dr. C.Yuvaraj** for is constant support and for providing us the necessary facilities which helped us in completion of the project work and the course as well.

I thank all the non-teaching faculty of the Computer Application department for their support and guidance in acquiring relevant knowledge in the core Computer Applications subjects which helped us in implementation of the project work.

I am very much indebted to institute management for providing us an opportunity with necessary academic, infrastructure facilities and immense support to carry out my graduation program successfully.

Finally, I would like to thank my parents, friends and well-wishers whose constant encouragement and faith in my abilities inspired me to deliver the best.

K. MAMATHA

Roll. No. 18699F0049

ABSTRACT

Communication is a major issue for a person who is blind. According to World Health Organization, about 285 million people are visually impaired world wide, 466 million people have disabling hearing loss and 1 million people are dumb. In this project, we propose a new system prototype that can help people who are suffering from blindness. Especially the tts.speech. Speech To Text and tts.speech.Recognizer Intent has been taken advantage to create custom voice command functionalities. Creating, replying, sending and forwarding messages are among the primary and fundamental features that this study has to offer. The researchers analyzed the results of the test survey and evaluation form and proved that the application is a user friendly, efficient and accurate in delivering messages to the recipient and has the important features that the users expected. We use gestures to communicate with the mobile.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBRIVATIONS

S.NO	**ABBRIVATION**	**EXPANSION**
1	DB	Data base
2	JVM	Java Virtual Machine
3	JSP	Java Server Page
4	CB	Collective Behavior
5	RSSS	Ramp Secret Sharing Scheme
6	JRE	Java Runtime Environment

Chapter-1

INTRODUCTION

1.1 About the project

Blind Communicator is an android type mobile messaging application designed and developed for the thorough communication between two disabled people most especially the hearing-impaired and blind people. With stable and smart Eclipse IDE and the availability of different built-in libraries in java especially the tts.speech.SpeechToText and tts.speech.Recognizer Intent has been taken advantage to create custom voice command functionalities. Creating, replying, sending and forwarding messages are among the primary and fundamental features that this study has to offer. The researchers analyzed the results of the test survey and evaluation form and proved that the application is a user friendly, efficient and accurate in delivering messages to the recipient and has the important features that the users expected.

Communication the ability to convey information to others and the ability to receive and interpret information from others is fundamental to learning. Individuals learn about the world mostly through their senses of vision and hearing. Vision and hearing are the main sensory avenues for accessing and interacting with the world around us and for perceiving events as close or distant. When vision and hearing are reduced, even to a mild level, the losses affect the ability to communicate, develop personal relationships, and acquire concepts.

Chapter-2

SYSTEM ANALYSIS

2.1 problem definition

communication is an important human characteristic. In order to maintain relationships effectively humans must communicate with each other. In everyday life, there are variety of communications including with work colleagues, family, neighbors, and friends, some efficient and some inefficient. The present study evaluates communication skills in the blind. How do blind individuals communicate with each other. How do they verbal and non-verbal communication. How do they cope with everyday life. Are there some aids that can help them to communicate with others. In order answer these questions, biblical verses related to the communication skills are studied.

2.2 EXISTING SYSTEM

Some of the note takers can be used in face-to-face conversation of the blind and hearing-impaired person. The hearing-impaired person types the text in the keyboard and the blind reads the text through the Braille display in the device. The blind can reply through typing in the Braille keyboard the text and read in the screen by the hearing-impaired person.

2.3 Disadvantages of Existing system

- We need a virtual keyboard to send messages.
- There are no gestures.
- Only voice commands present in existing system.
- No reminders of battery low or system changes.

2.4 proposed system

The hearing-impaired and blind users are using the same interface and functionality in the blind Communicator Messaging application. Moreover, the pop up microphone icon for voice command is intended for the blind.

2.5 Advantages of proposed system

- Using gestures user can control the mobile phone.
- The text will be speak out.
- It reads the messages sent by users.
- It cans speak out the contacts names.
- Setting alarms, settings using gestures and voice control.

Chapter-3

SYSTEM REQUIREMENTS

The project provides an effective solution in resolving all the arguments that occurs in organizations by considering all employees opinions. Android voting system using some main Hardware & Software requirements.

3.1.Hardware Requirements

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It should what the system do and not how it should be implemented.

Processor – i3 Processor Based Computer

Hard Disk – 80 GB

Memory – 4GB RAM

Android Device

3.2 Software Requirements

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team's progress throughout the development activity.

Operating System : Windows 10 or higher

Front End : Android

Server side Script : java

3.3 Functional and Non Functional Requirements

3.3.Functional Requirements

- A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behavior, and outputs.

- The system has an admin login that has overall control over it. Admin feed the issues or arguments in the system along with desired options.

- It provides authentication to individuals who wish to get connected. The registration details are stored in the database and whenever the Employees logs in, the employees credentials are retrieved to check whether the employee is an authorized employee or not.

- If the details entered do not match with any of the existing data then the system displays a warning.
- These questions can then be visible to all the employees through android devices. Employees have to first create an account into the system for casting their votes.
- At the end of the voting process the system counts all the votes casted and generates a brief report of the total votes accounted for yes, no and neutral. Eventually, the report is made available to admin and he may view the maximum votes casted for.

- Hence the system helps admin to receive appropriate response from employees for the matters in question.

3.3.2 Non Functional Requirements

- **Portability:** It should run on specified platforms successfully. To achieve this we should test the product on all platforms before launching the product. If our project runs successfully on different platforms then our system is portable in nature.
- **Reliability:** The system should perform its intended functions under specified conditions. If our system satisfies all the specified conditions then it is Reliable in nature.
- **Reusability:** The system should be extremely reusable as a whole or part. Make the system modularize and make sure that modules are loosely coupled. This project is having reusability nature because we can reuse whole or part of this project on other systems.
- **Robustness:** The system on the whole should be robust enough to perform well under different circumstances without any inconsistencies.
- **Testability:** The product of a given development phase should satisfy the conditions imposed at the start of that phase
- **Usability:** It should be perfect and comfortable for users to work.
- **Security:** The system is completely based on the security. This system will provide security base on the password.

Chapter-4

SYSTEM DESIGN

4.1.Module Description

Blind Communicator is an android type mobile messaging application designed and developed for the thorough communication between two disabled people most especially the hearing-impaired and blind people. With stable and smart Eclipse IDE and the availability of different built-in libraries in java especially the tts.speech.SpeechToText and tts.speech.RecognizerIntent has been taken advantage to create custom voice command functionalities.

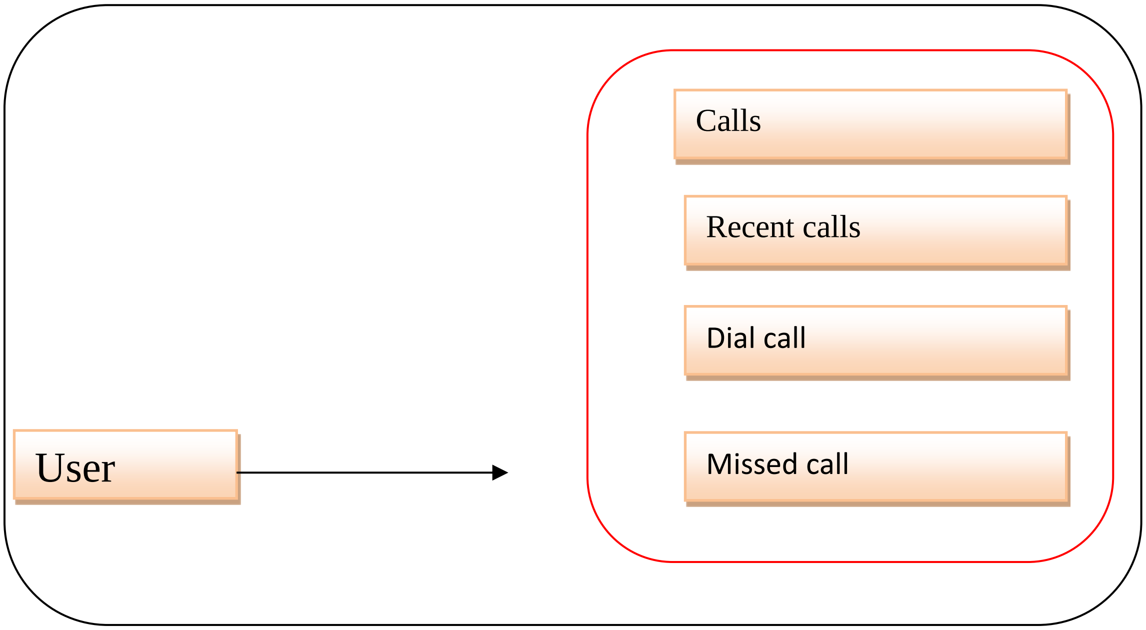
This project having the following modules:

- Calls
- Messages
- Contacts
- Music
- Internet
- Alarm
- Voice recorder
- Applications
- Settings
- Status

Calls:

- In the main window user swipes the right or left to got the calls menu. In the calls menu according to gestures user can listen call logs. He can make calls. He can enter the number to call .
- In the calls menu consists of call logs
  delete logs
  make a call

- **Call logs:** In this menu user swipes the right or left to got the call logs menu.
- **Delete logs:** The user swipes the right or left to delete the call logs.
- **Make a call:** In this menu consists of input phone number and call options.
- **Input phone number:** This is used to type the numbers, letters and symbols.
- Slide up and down to select character.
- Slide right to confirm character or key and move to the next one.
- Slide left to delete the last character.



4.1.1 Call Module

Messages :

- In the main window user swipes the right or left to got the messages menu. In the messages menu according to gestures user can listen the messages. He can send the messages.
- In messages menu consists of Inbox,

  Sent,

  compose.
- **Inbox:** In this menu the user can listen the messages one by one.
- In this menu consists of Compose message,

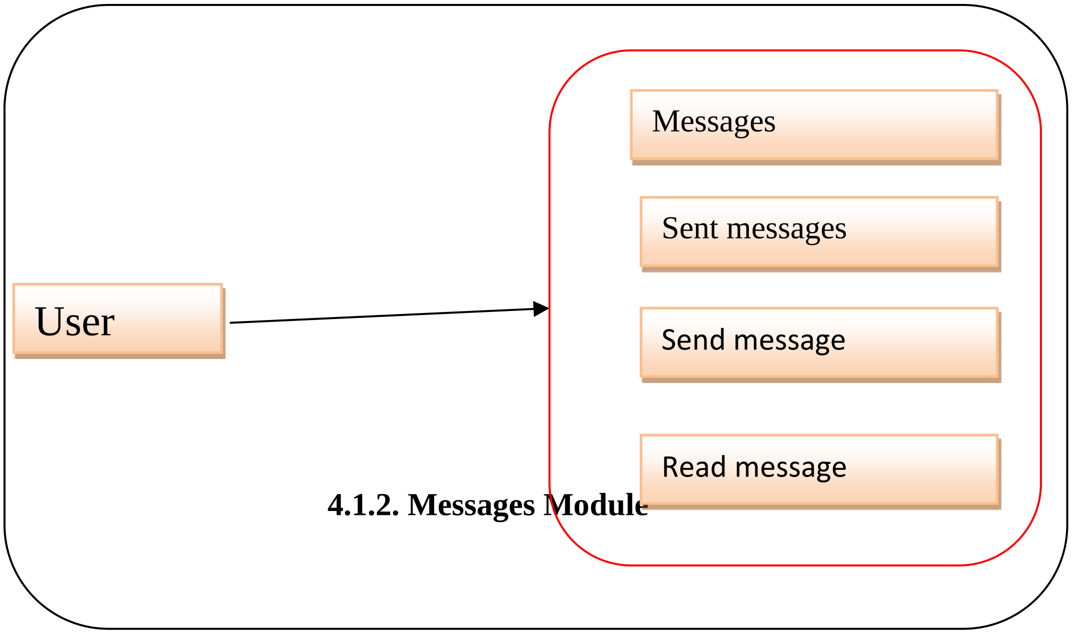
Delete message,

Call to contact,

Add to new contact,

Delete all received messages.

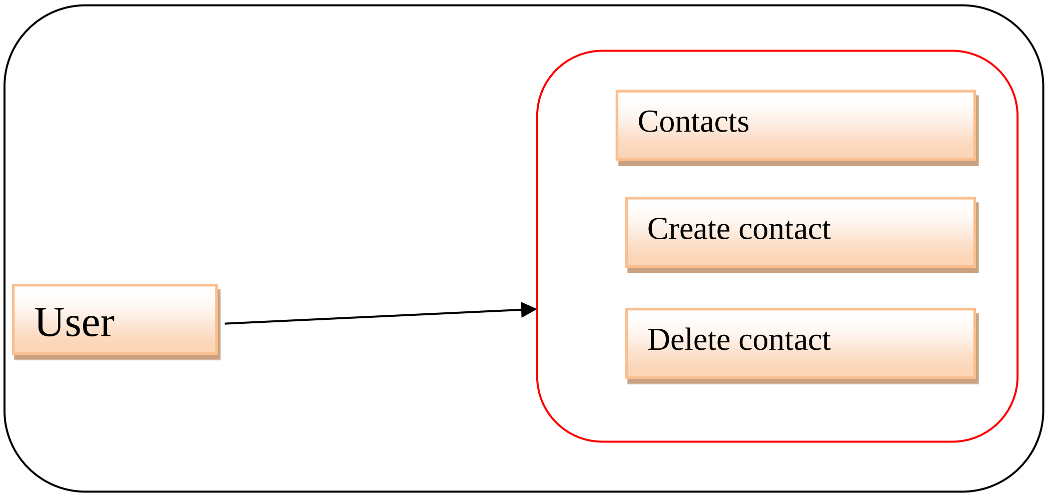
- **Sent:** In the sent messages menu the user swipes to left. User can listen the sent messages and also delete the messages.
- **Compose:** In the compose menu the user select the number in the contact list, type the message, and send message to that number.



4.1.2. Messages Module

Contacts:

- In the main window user swipes the right or left to got the contacts menu. In the contact menu according to gestures user can listen the all contacts in the mobile.
- In this menu consists contacts and create contact.
- **Contacts:** This menu is used to listen the contact list and select the number. The user send the message that number and also call the selected contact.
- **Create contact:** This is used to create the contacts



4.1.3. Contacts Module

Music:

- In the main window user swipes the right or left to got the Music menu. In the music menu according to gestures user can listen music tracks and play the tracks in the music player.
- In this menu consists of Now playing ,

  Play/Stop,

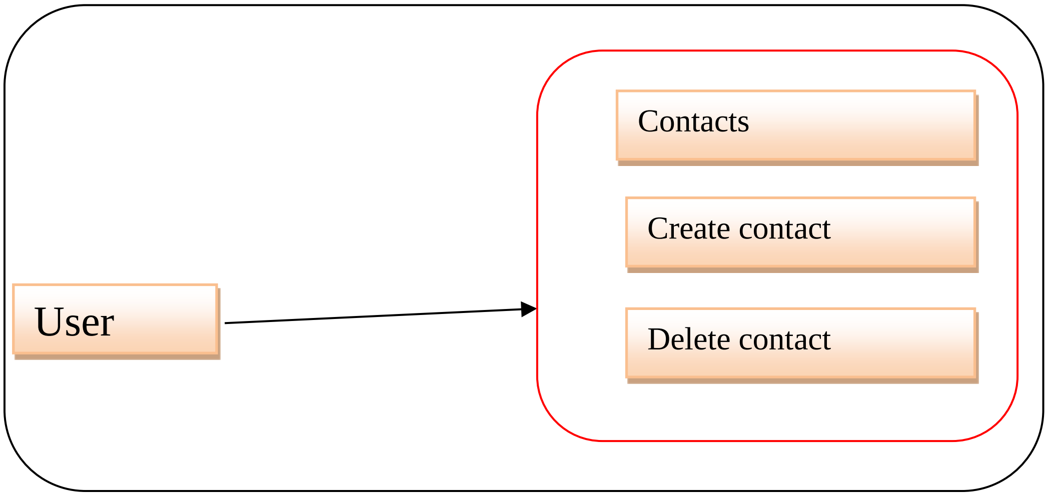
  Previous track,

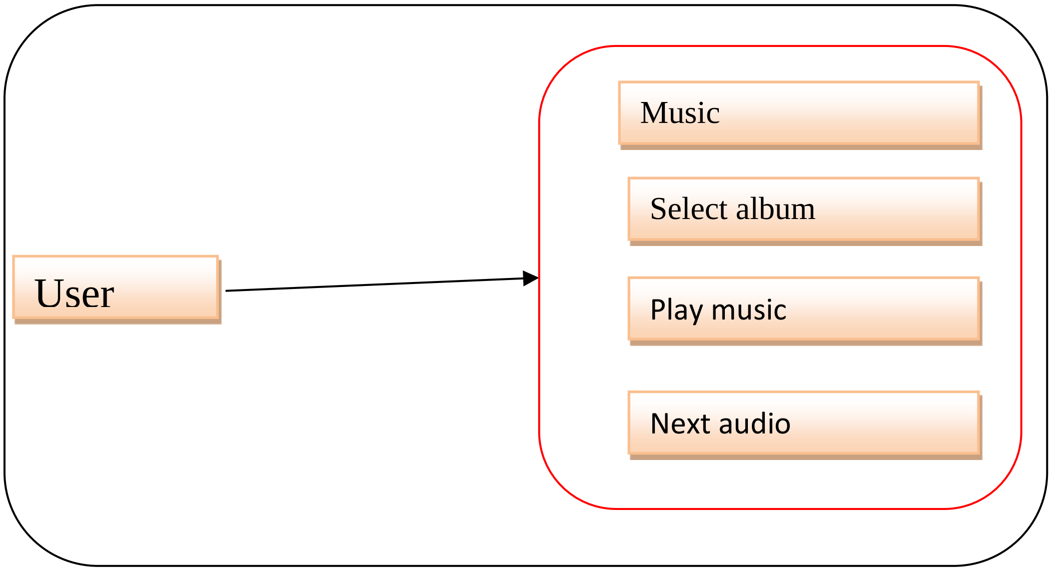
  Next track,

  Lower volume,

  Higher volume,

  Artists.

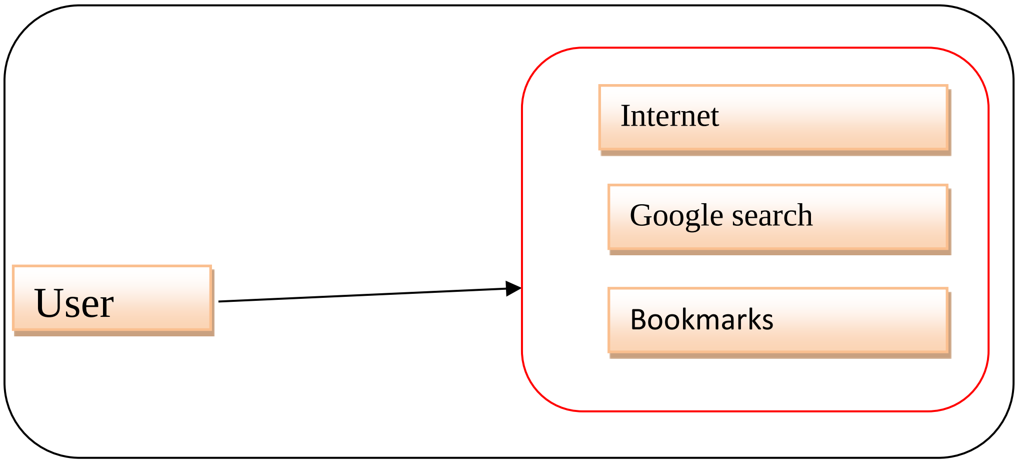




4.1.4.Music Module

Internet:

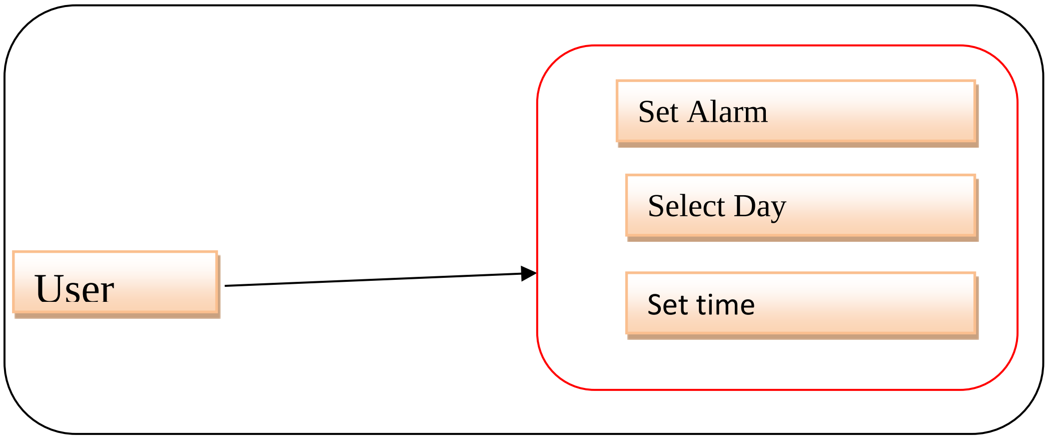
- In the main window user swipes the right or left to go t the Internet menu. In the Internet menu according to gestures user can search in Google with the help of input text mode.



4.1.5. Internet Module

Alarm:

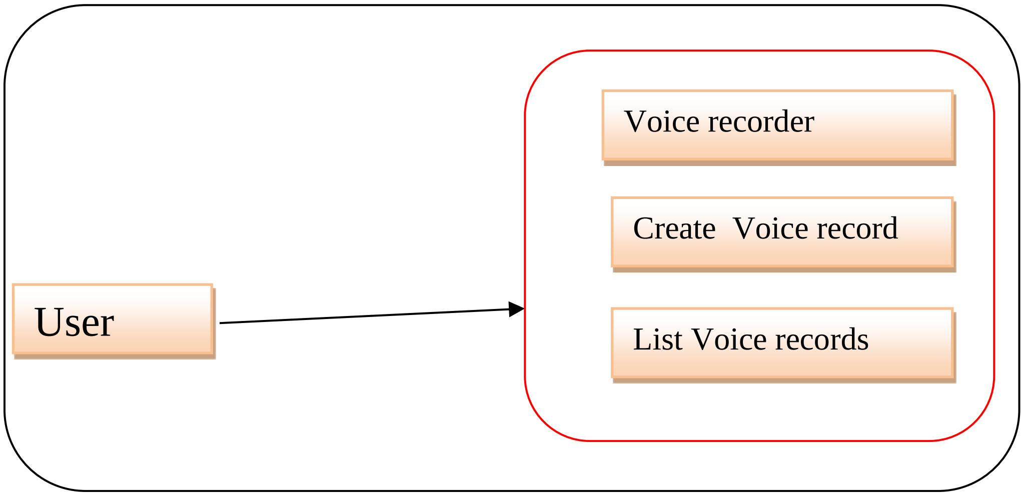
- In the main window user swipes the right or left to got the Alarm menu. In the Alarm menu according to gestures user can set alarm. He can view alarm. Delete an alarm.
- In this menu consists of Alarm list,
  Create alarm.
- **Alarm list:** In the alarm list menu, the user can listen the alarm list and he can delete the selected alarm and also delete the all alarms.
- **Create alarm**: This is used to create the alarms.



4.1.6. Alarm Module

Voice recorder:

- In the main window user swipes the right or left to got the voice recorder menu. In this menu according to gestures user can record the calls and normal voice records. User can also delete the records and play the voice records.
- **Create voice record:** This is used to create the voice records

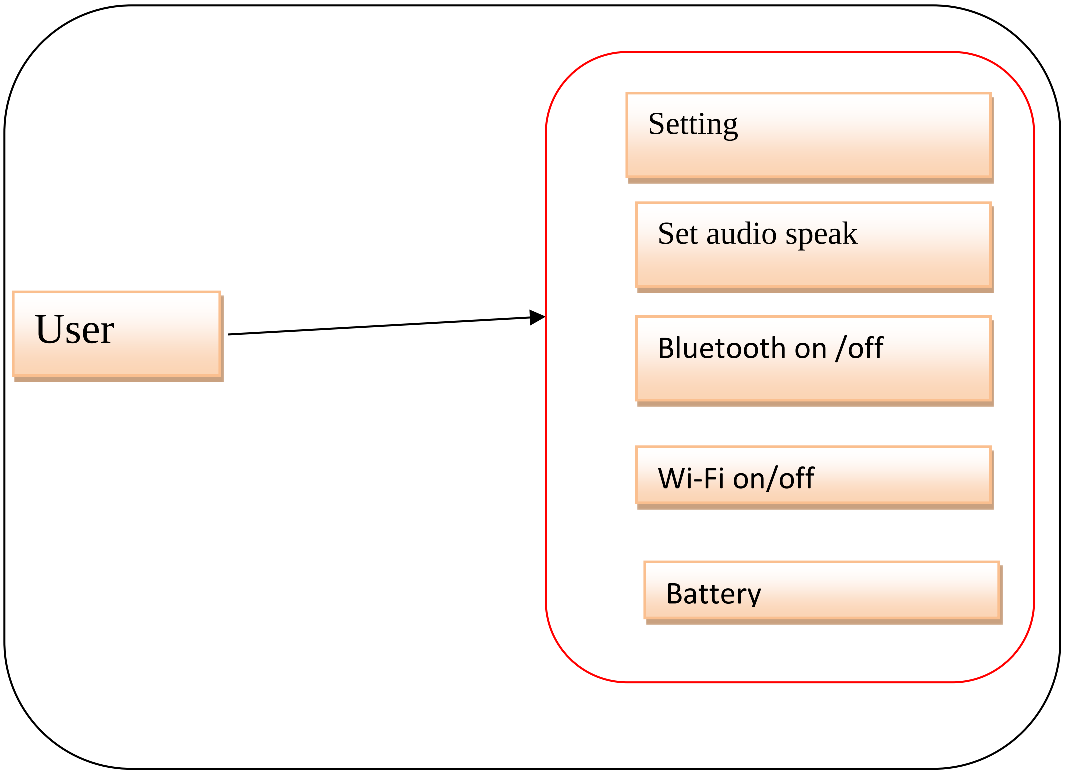


4.1.7. Voice record Module

Settings:

- In the main window user swipes the right or left to got the Setting menu. In the settings menu according to gestures user can set settings and remove the settings.
- In this menu the user can set the changes in the given list.
- Reading speed.
- Screen timeout (seconds).
- Alarm tone.
- Profile.
- Wi-Fi.
- Bluetooth.
- Input mode: keyboard.
- System settings.

- Privacy policy.



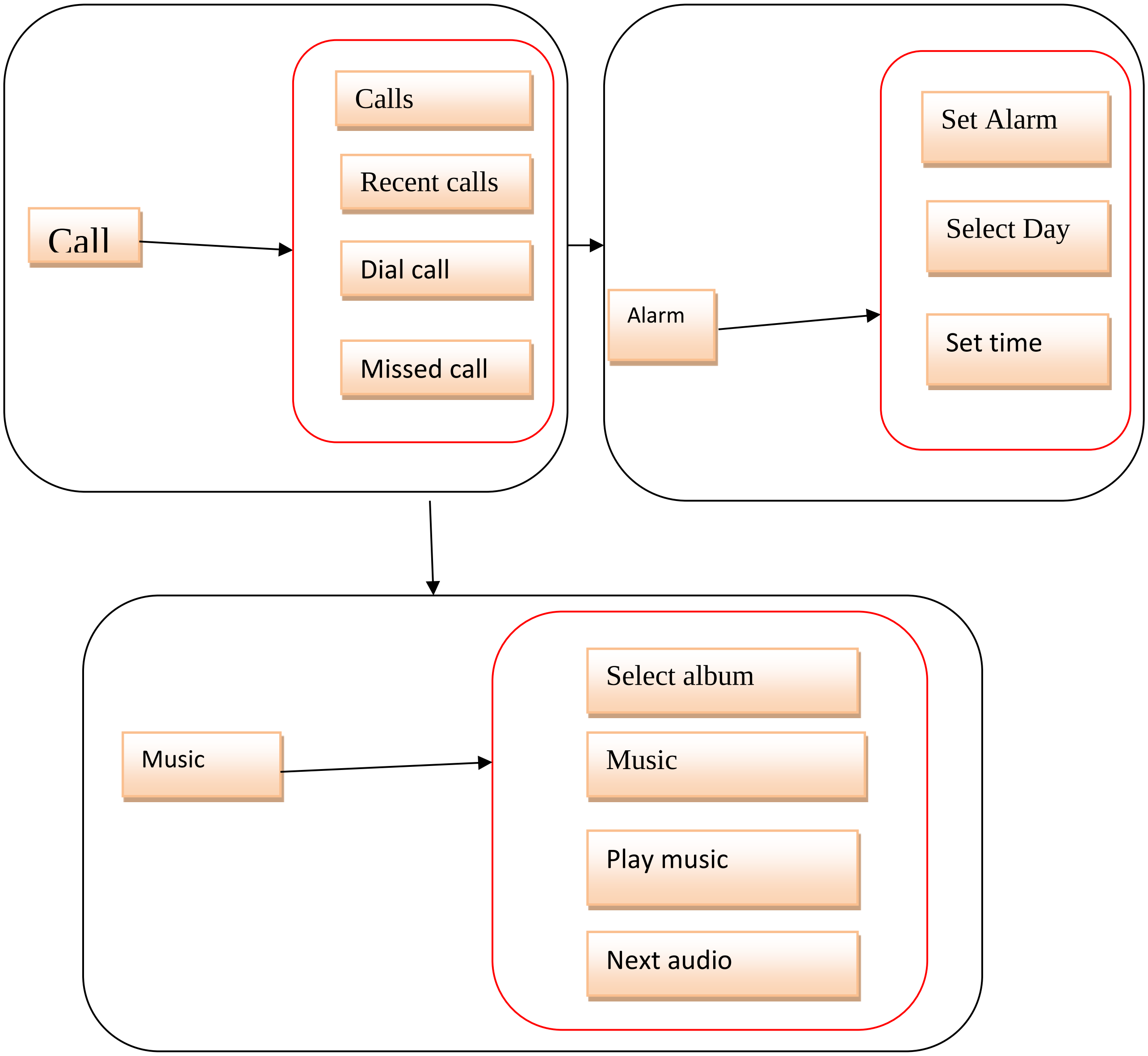
4.1.8. Setting Module

4.2.Data Flow Diagram

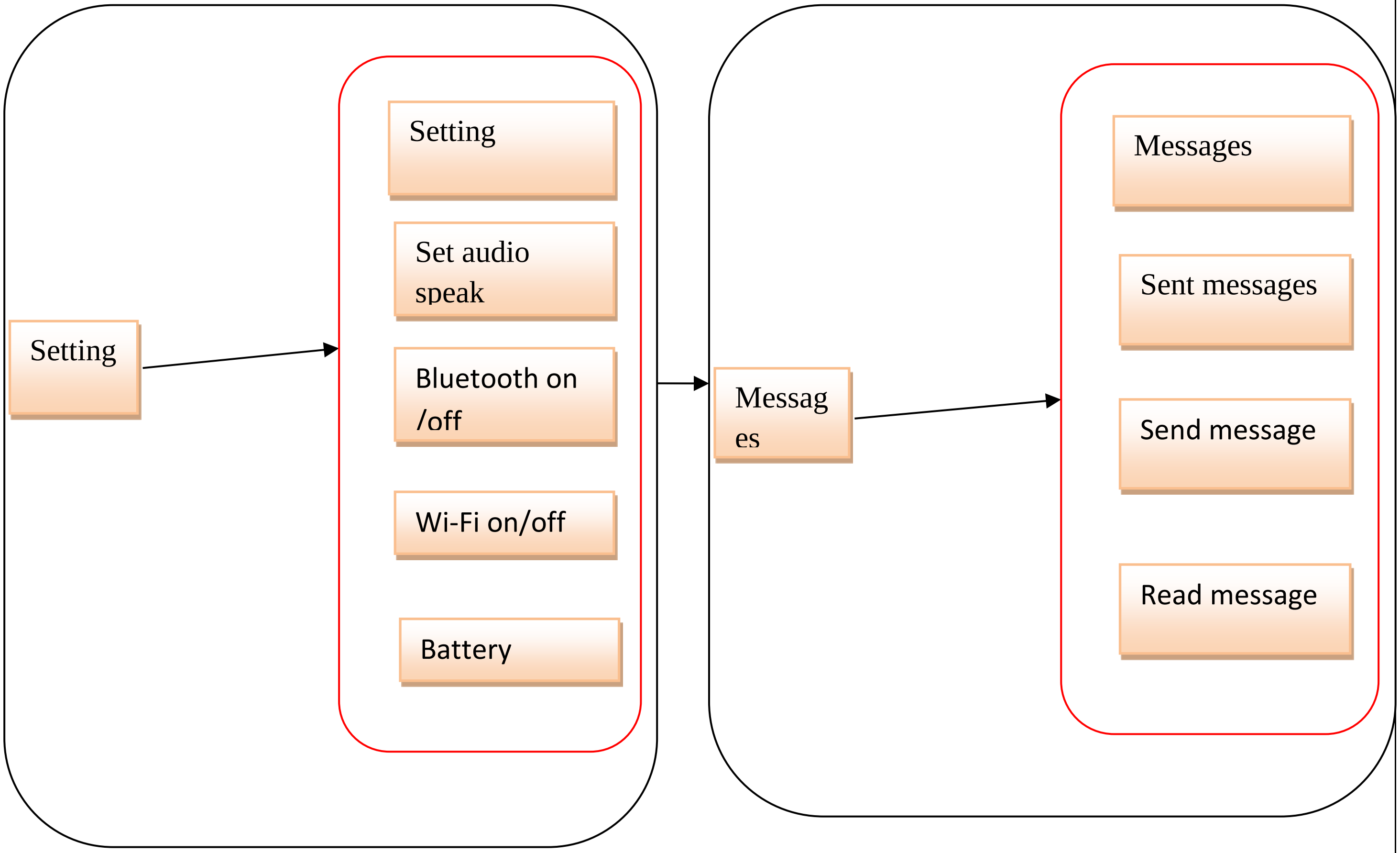
A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of data will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel.

Dataflow Diagram level:1



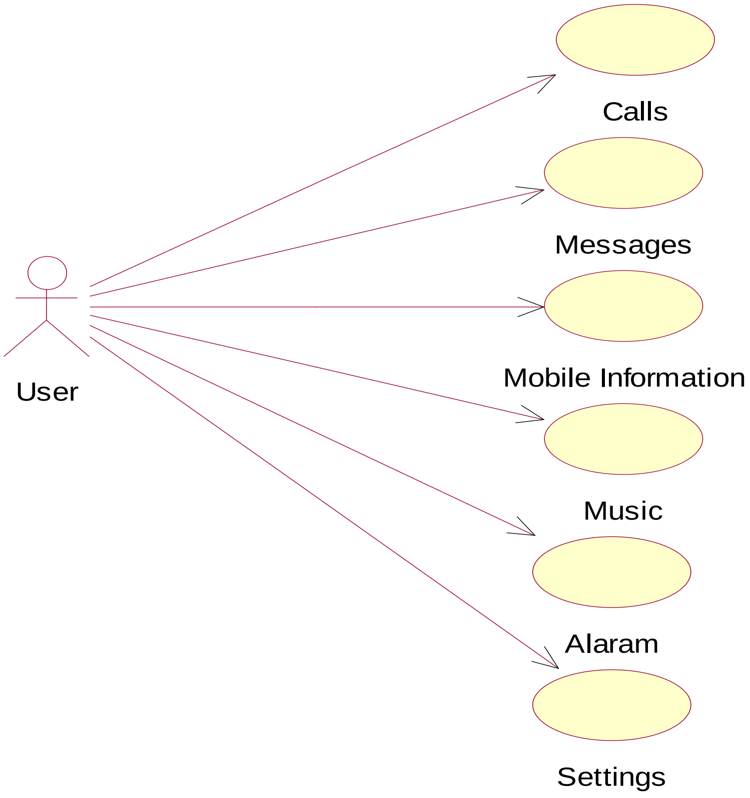
Dataflow Diagram level: 2



4.3 UML Diagrams

4.3.1 Use Case Diagram

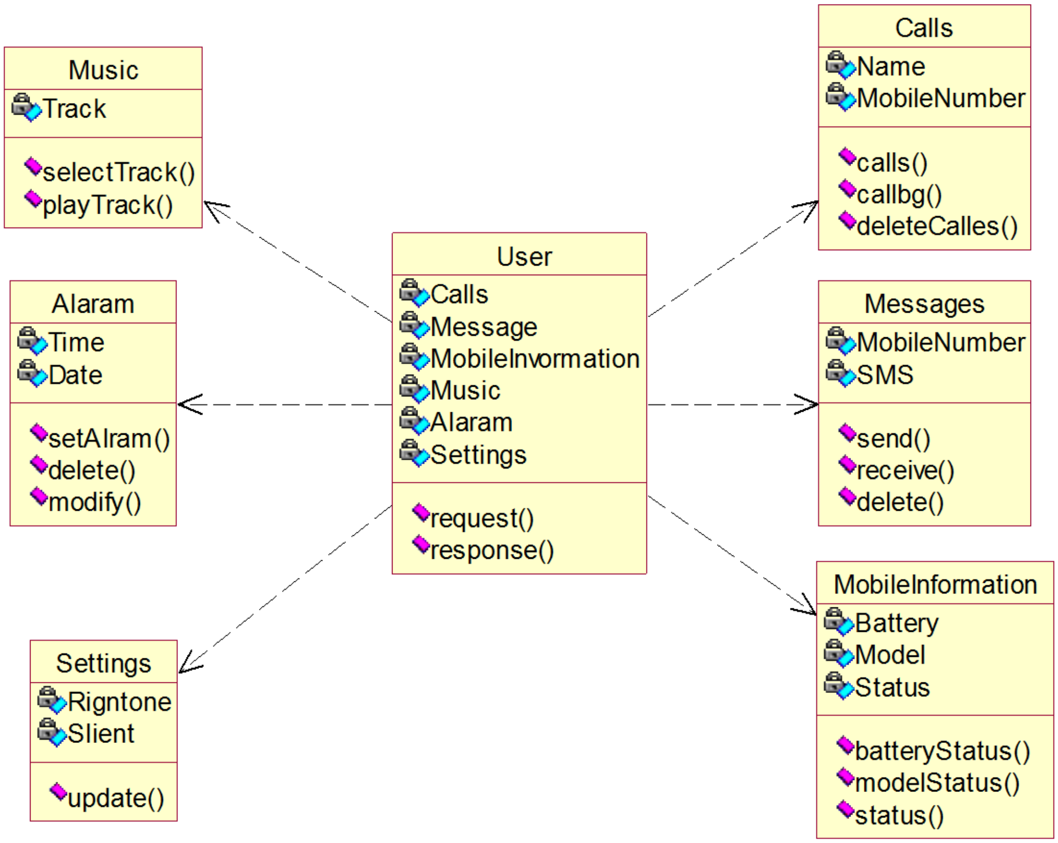
The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted. The given diagram consists of user as actor. Each will play a certain role to achieve the concept.



4.3.1 Use case Diagram

4.3.2.Class diagram

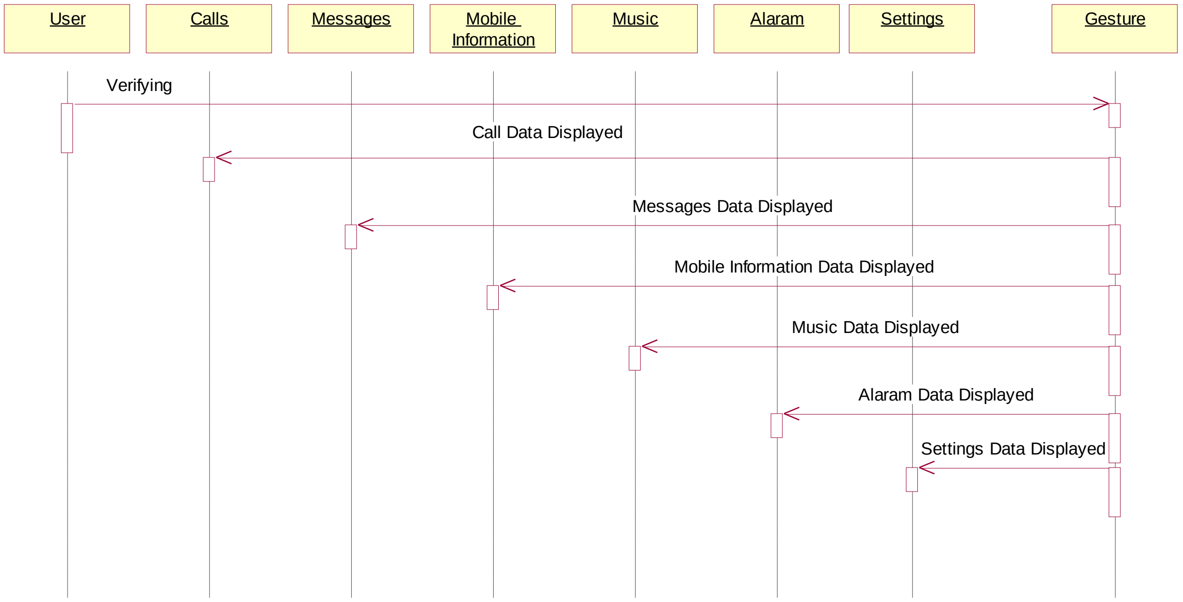
In this class diagram represents how the classes with attributes and methods are linked together to perform the verification with security. From the above diagram shown the various classes involved in our project.



4.3.2. Class Diagram

4.3.3.Sequence diagram

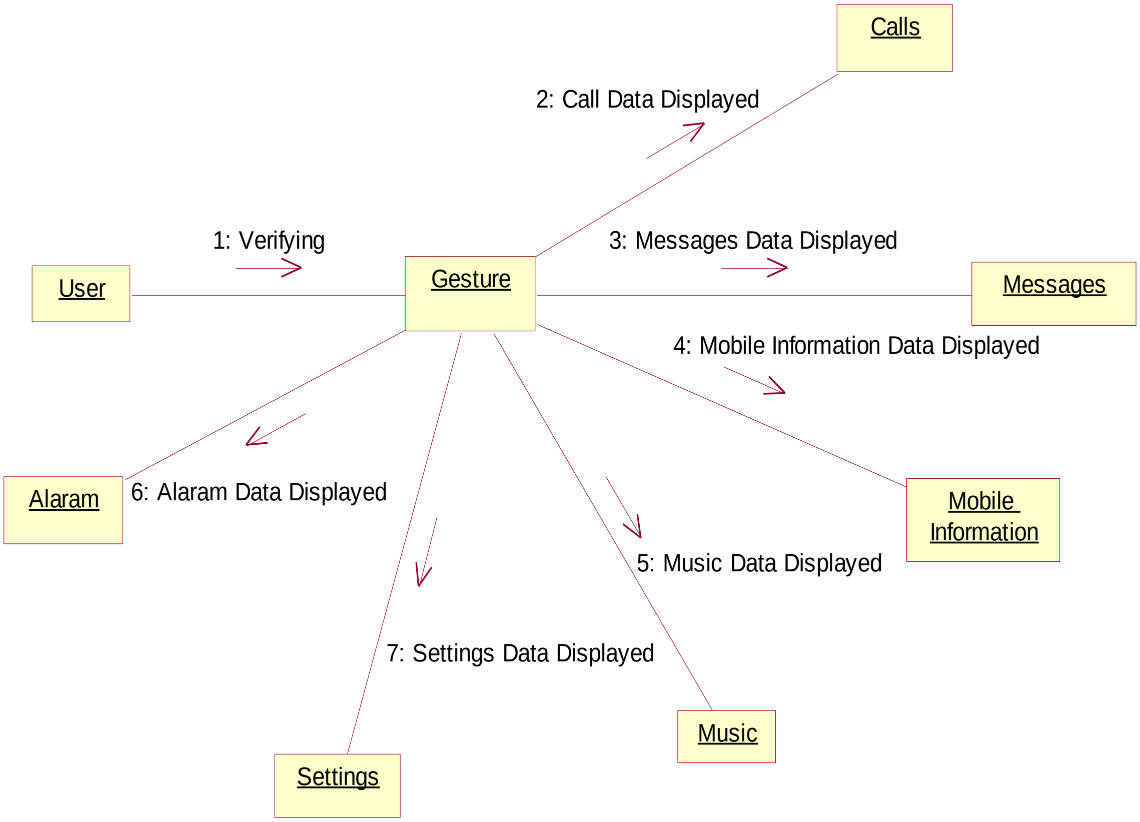
A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.



4.3.3. Sequence Diagram

4.3.4. Collaboration diagram

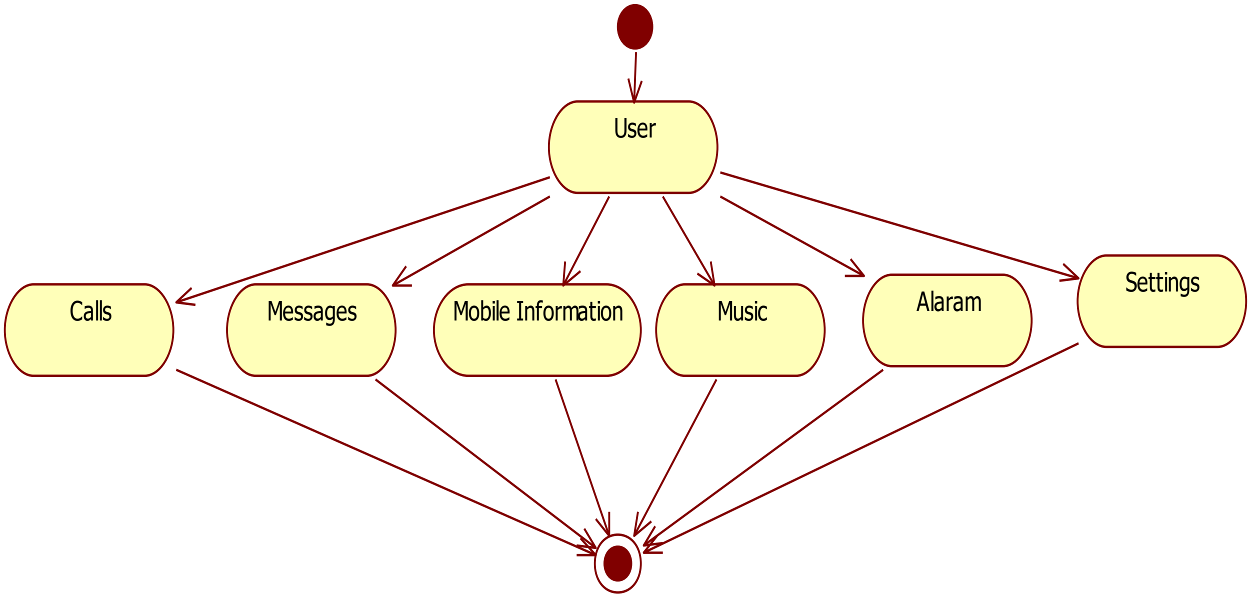
A collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). The concept is more than a decade old although it has been refined as modeling paradigms have evolved.



4.3.4. Collaboration Diagram

4.3.5. Activity diagram

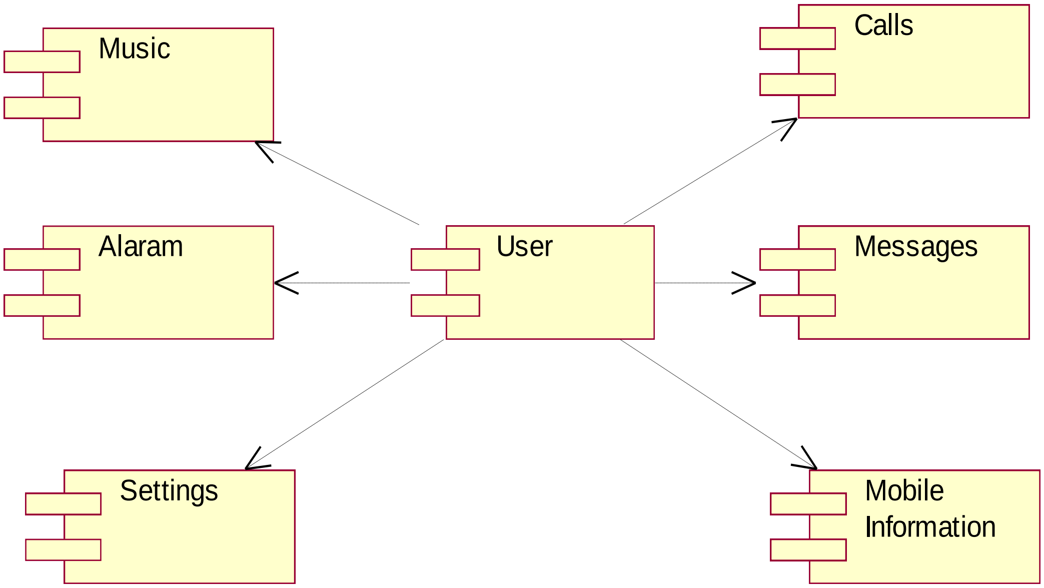
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



4.3.5. Activity Diagram

4.3.6.Component diagram

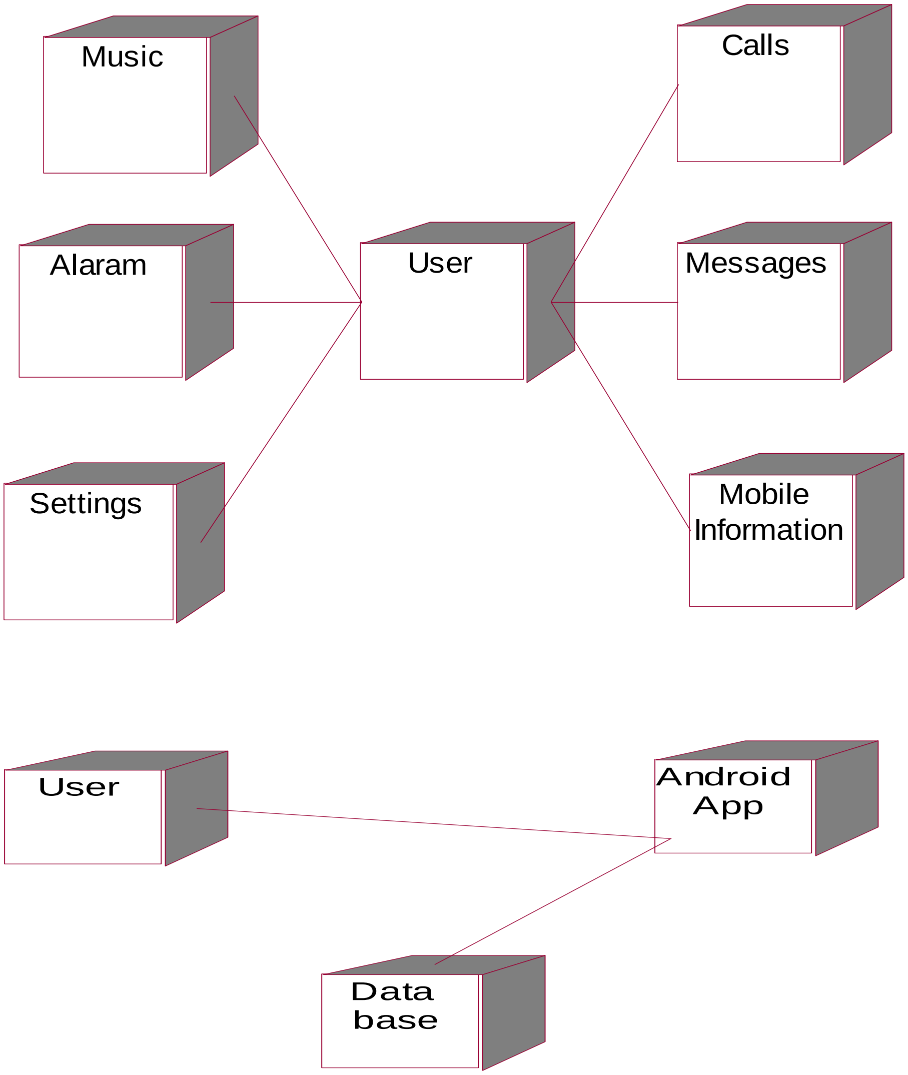
In the Unified Modeling Language, a component diagram depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems. User gives main query and it converted into sub queries and sends through data dissemination to data aggregators. Results are to be showed to user by data aggregators. All boxes are components and arrow indicates dependencies.



4.3.6. Component Diagram

4.3.7 Deployment Diagram:

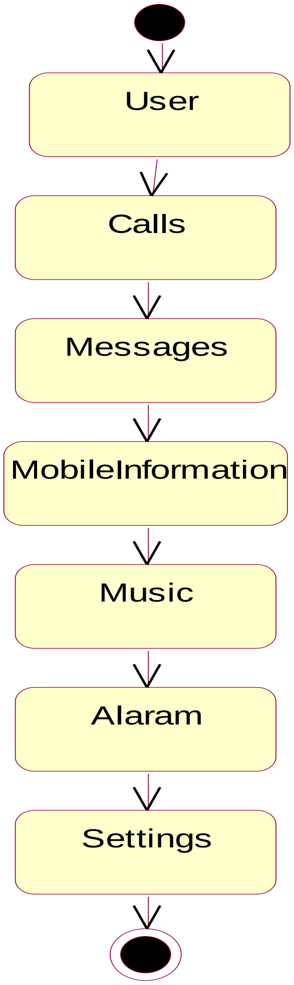
In the Unified Modeling Language, a deployment diagram depicts how deploys are wired together to form larger deployment and or software systems. They are used to illustrate the structure of arbitrarily complex systems. User gives main query and it converted into sub queries and sends through data dissemination to data aggregators. Results are to be showed to user by data aggregators. All boxes are components and arrow indicates dependencies



4.3.7. Deployment Diagram

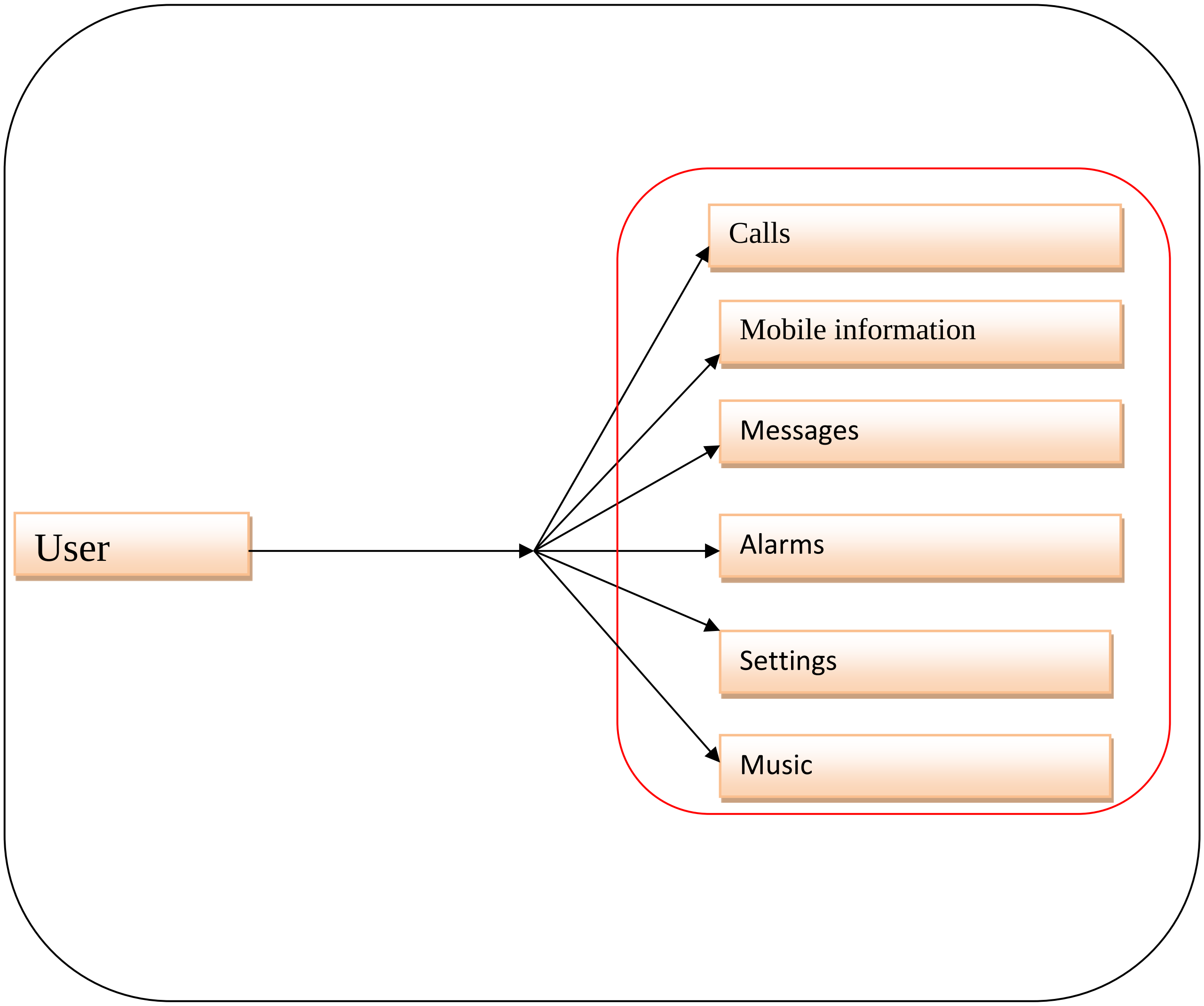
4.3.8. State chart diagram

State diagram are a loosely defined diagram to show workflows of stepwise activities and actions, with support for choice, iteration and concurrency. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics.



4.3.8. State chart diagram

4.4. System architecture



CHAPTER V

SYSTEM IMPLEMENTATION

5.1 Language Selection

5.1.1 About android

Android is a complete set of software for mobile devices such as tablet computers, notebooks, smart phones, electronic book readers, set-top boxes. It contains a linux-based Operating System, middleware and key mobile applications. It can be thought of as a mobile operating system. But it is not limited to mobile only. It is currently used in various devices such as mobiles, tablets, televisions etc

5.1.2 What is android?

Android is a software package and Linux based operating system for mobile devices such as tablet computers and smart phones. It is developed by Google and later the OHA (Open Handset Alliance). Java language is mainly used to write the android code even though other languages can be used. The goal of android project is to create a successful real-world product that improves the mobile experience for end users. There are many code names of android such as Lollipop, KitKat, Jelly Bean, Ice cream Sandwich, Froyo, Eclair, Donut etc. which is covered in next page

5.1.3. About Open Handset Alliance (OHA)

It's a consortium of 84 companies such as Google, Samsung, AKM, synaptics, KDDI, Garmin, Teleca, Ebay, Intel etc. It was established on 5th November, 2007, led by Google. It is committed to advance open standards, provide services and deploy handsets using the Android Platform.

5.1.4. Features of Android

1) It is open-source.

2) Anyone can customize the Android Platform.

3) There are a lot of mobile applications that can be chosen by the consumer.

4) It provides many interesting features like weather details, opening screen, live RSS (Really Simple Syndication) feeds etc.

It provides support for messaging services(SMS and MMS), web browser, storage (SQLite), connectivity (GSM, CDMA, Blue Tooth, Wi-Fi etc.), media, handset layout etc.

5.1.5 Categories of Android applications

- Entertainment
- Tools
- Communication
- Productivity
- Personalization
- Music and Audio
- Social
- Media and Video
- Travel and Local etc.

5.1.6 History of Android

- Initially, **Andy Rubin** founded Android Incorporation in Palo Alto, California, United
- States in October, 2003.
- In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.
- The key employees of Android Incorporation are **Andy Rubin**, **Rich Miner**, **Chris White** and **Nick Sears**.
- Originally intended for camera but shifted to smart phones later because of low market for camera only.
- Android is the nick name of Andy Rubin given by coworkers because of his love to robots.
- In 2007, Google announces the development of android OS. In 2008, HTC launched the first android mobile.

5.1.7 Android Architecture

- **Linux kernel:** It is the heart of android architecture that exists at the root of android architecture. Linux kernel is responsible for device drivers, power management, memory management, device management and resource access.

- **Native libraries (middleware):** On the top of Linux kernel, there are **Native libraries** such as Web Kit, OpenGL, Free Type, SQLite, Media, C runtime library (libc) etc. The Web Kit library is responsible for browser support, SQLite is for database, Free Type for font support, Media for playing and recording audio and video formats.

- **Android Runtime:** In android runtime, there are core libraries and DVM (Dalvik Virtual Machine) which is responsible to run android application. DVM is like JVM but it is optimized for mobile devices. It consumes less memory and provides fast performance.
- **Application Framework:** On the top of Native libraries and android runtime, there is android framework. Android framework includes **Android API's** such as UI (User Interface), telephony, resources, locations, Content Providers (data) and package managers. It provides a lot of classes and interfaces for android application development.
- **Applications:** On the top of android framework, there are applications. All applications such as home, contact, settings, games, browsers are using android framework that uses android runtime and libraries. Android runtime and native libraries are using Linux kernel.

5.1.8 Android Core Building Blocks

An android component is simply a piece of code that has a well-defined life cycle e.g. Activity, Receiver, Service etc.

The core building blocks or fundamental components of android are activities, views, intents, services, content providers, fragments and AndroidManifest.xml.

Activity

An activity is a class that represents a single screen. It is like a Frame in AWT.

View

A view is the UI element such as button, label, text field etc. Anything that you see is a view.

Intent

Intent is used to invoke components. It is mainly used to:

- Start the service
- Launch an activity
- Display a web page
- Display a list of contacts
- Broadcast a message
- Dial a phone call etc.
- For example, you may write the following code to view the webpage.

Example :

```
Intent intent=new Intent(Intent.ACTION_VIEW);

intent.setData(Uri.parse("http://www.gurunanak.com"));

startActivity(intent);
```

Service

Service is a background process that can run for a long time. There are two types of services local and remote. Local service is accessed from within the application whereas remote service is accessed remotely from other applications running on the same device.

Content Provider

Content Providers are used to share data between the applications.

Fragment

Fragments are like parts of activity. An activity can display one or more fragments on the screen at the same time.

AndroidManifest.xml

It contains information about activities, content providers, permissions etc. It is like the web.xml file in Java EE.

Android Virtual Device (AVD)

It is used to test the android application without the need for mobile or tablet etc. It can be created in different configurations to emulate different types of real devices.

APK File

An apk file is created by the framework automatically. If you want to run the android application on the mobile, transfer and install it.

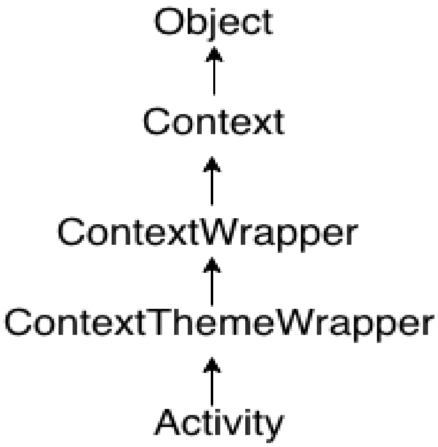
Resources

It contains resource files including activity main, strings, styles etc.

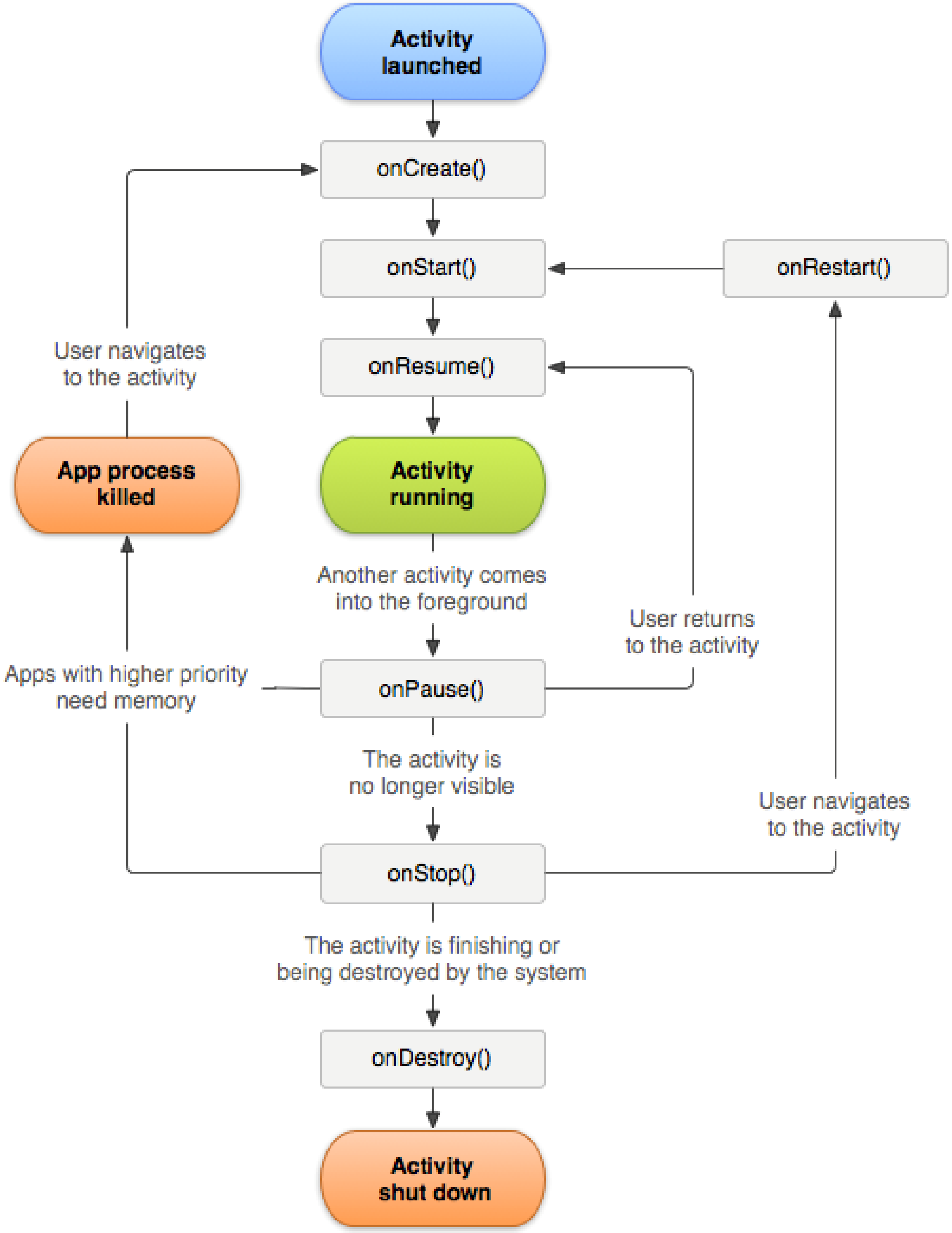
Manifest file

It contains information about package including components such as activities, services, content providers etc.

5.1.9 Android Activity Lifecycle



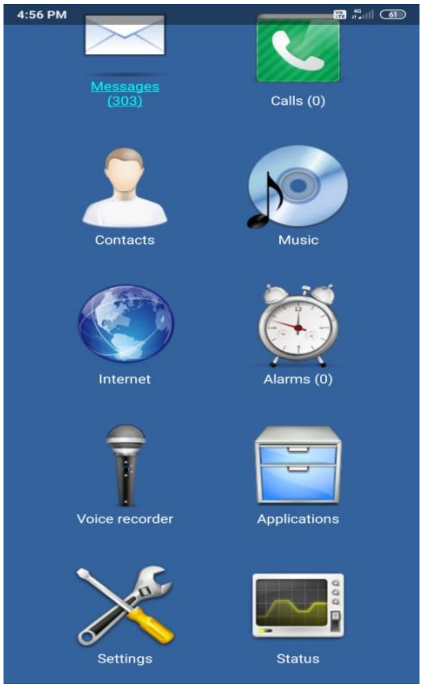
Android Activity Lifecycle is controlled by 7 methods of android.app.Activity class. The android Activity is the subclass of ContextThemeWrapper class. An activity is the single screen in android. It is like window or frame of Java. By the help of activity, you can place all your UI components or widgets in a single screen. The 7 lifecycle method of Activity describes how activity will behave at different states.


Activity launched
onCreate()
onStart()
onRestart()
onResume()
User navigates to the activity
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

5.2 Screenshots

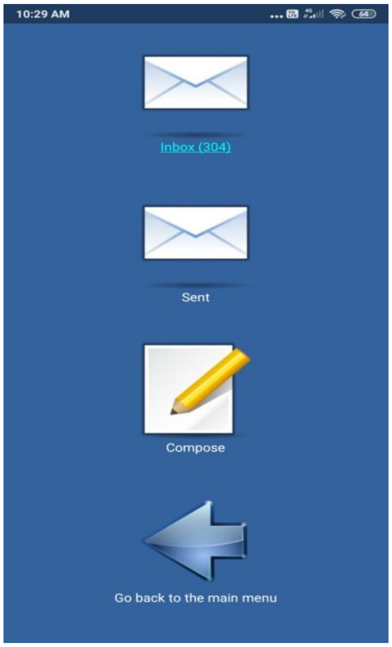
Screen 1:

Description: This screen shows the home page.



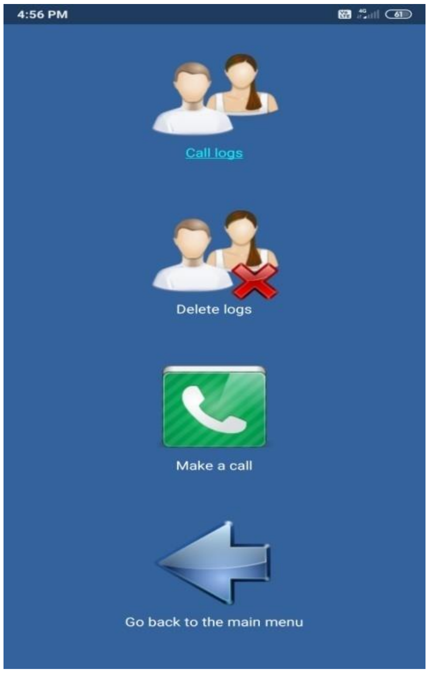
Screen 2:

Description: This page shows the message screen.



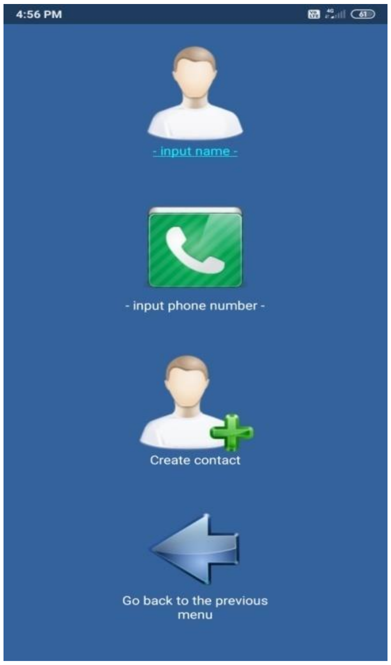
Screen 3:

Description: This page shows the call logs.



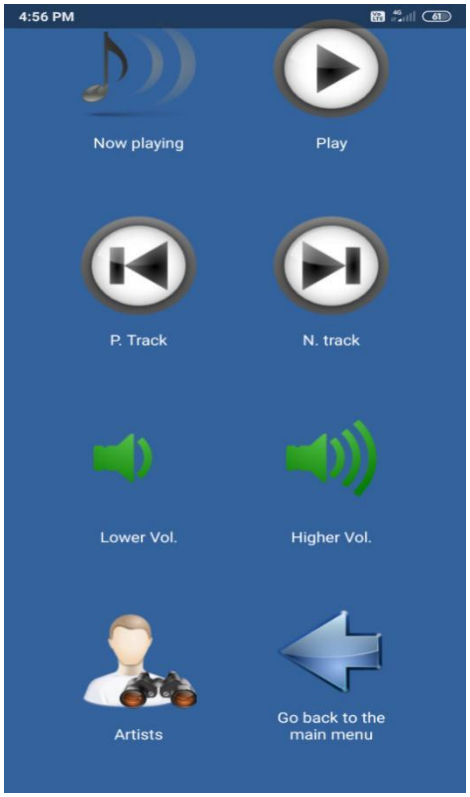
Screen 4:

Description: This page shows the create the contacts.



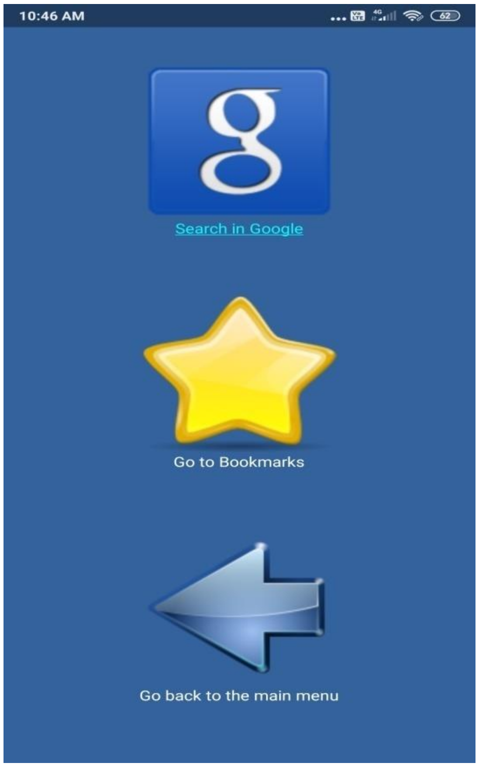
screen:5

Description: This page shows the music menu.



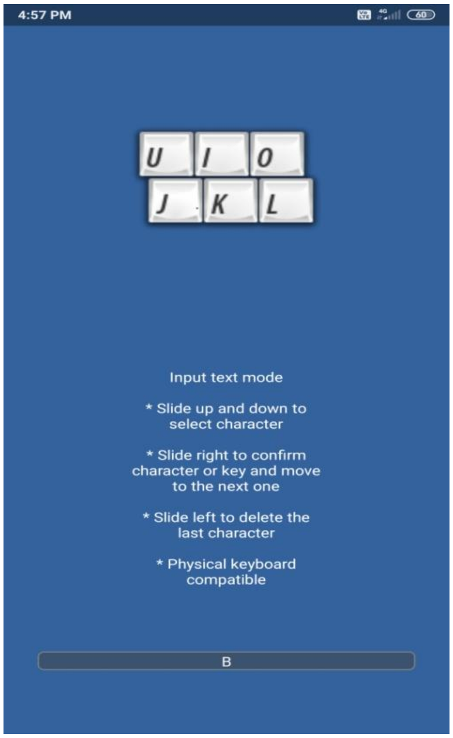
screen:6

Description: This page shows the internet menu.



screen:7

Description: This page shows the input text mode.



Input text mode

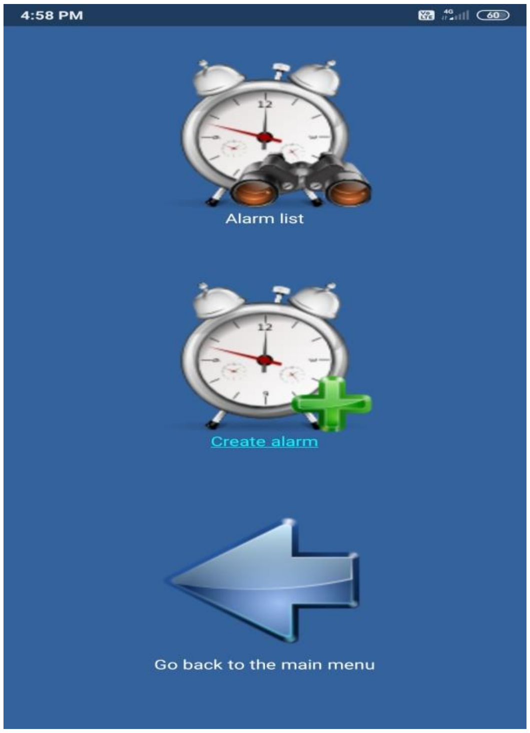
* Slide up and down to select character

* Slide right to confirm character or key and move to the next one

* Slide left to delete the last character

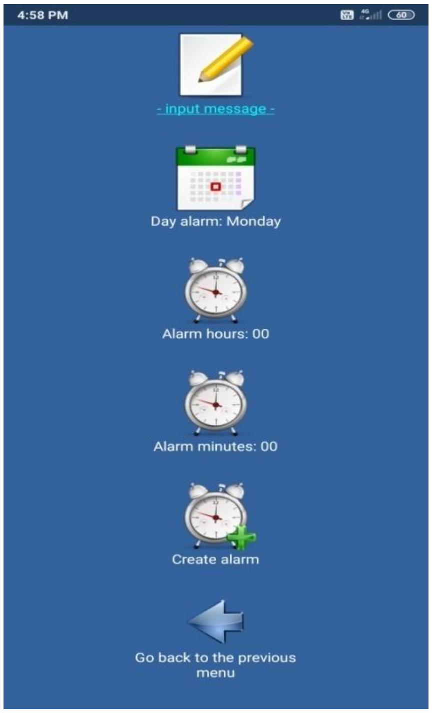
* Physical keyboard compatible

Screen 8:
Description: This page shows the alarm menu.



screen:9

Description: This page shows the create the alarm.



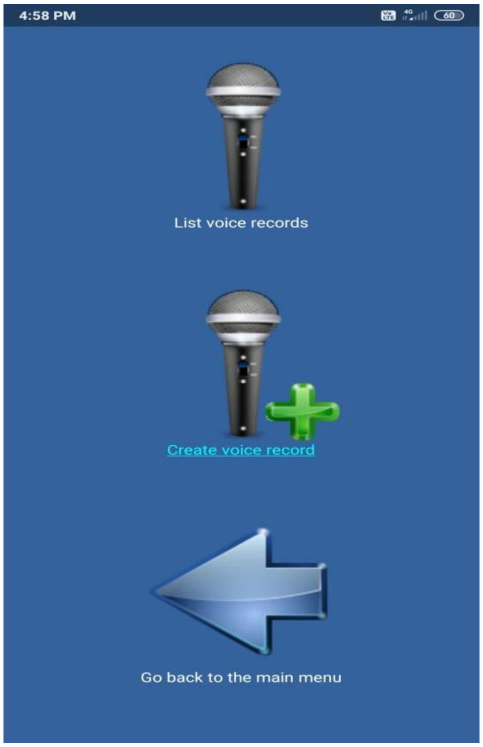
Day alarm: Monday

Alarm hours: 00

Alarm minutes: 00

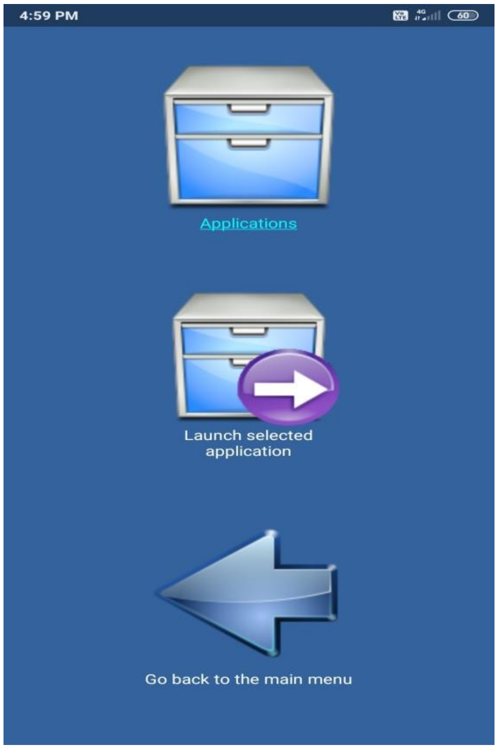
screen: 10

Description: This page shows the voice record menu.



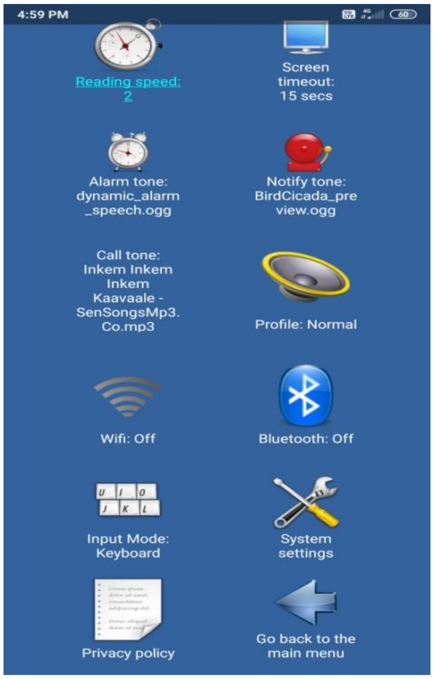
screen: 11

Description: This page shows the applications menu.



screen: 12

Description: This page shows the settings menu.



4:59 PM

Reading speed: 2

Screen timeout: 15 secs

Alarm tone: dynamic_alarm_speech.ogg

Notify tone: BirdCicada_preview.ogg

Call tone: Inkem Inkem Inkem Kaavaale - SenSongsMp3.Co.mp3

Profile: Normal

Wifi: Off

Bluetooth: Off

Input Mode: Keyboard

System settings

Privacy policy

Go back to the main menu

5.3 Sample code:

```
buildscript {
   repositories {
      google()
      jcenter()

   }
   dependencies {
      classpath 'com.android.tools.build:gradle:3.3.1'

}
}

allprojects {
   repositories {
      google()
      jcenter()

   }
}

task clean(type: Delete) {
   delete rootProject.buildDir
}
```

Module code:

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

android {
   compileSdkVersion 26
defaultConfig {
      applicationId "ar.com.lrusso.blindcommunicator.app"
minSdkVersion 14
targetSdkVersion 26
versionCode 72
versionName "1.9.8.9.4"
testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
}
   buildTypes {
      release {
         minifyEnabled false
proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-
rules.pro'
}
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    testImplementation 'junit:junit:4.12'
androidTestImplementation 'com.android.support.test:runner:1.0.2'
androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
}
```

Style XML:

```
<style name="AppBaseTheme" parent="android:Theme.Light">
</style>
<style name="AppTheme" parent="AppBaseTheme">
</style>

</resources>
```

Values/strings:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

<string name="app_name">Voice Based Communication</string>
<string name="title_activity_main">Voice Based</string>
<string name="menu_settings">Settings</string>

<string name="mainWelcome">Welcome to the voice based Communicator system. Slide up
and down to browse between options and slide right to choose the selected item. If your device
has a physical keyboard, you can use the directional keys. If you are using TalkBack, slide and
select menus using two fingers.</string>
<string name="mainMessages">Messages</string>
<string name="mainMessagesNew">new messages</string>
<string name="mainMessagesNoNew">Messages. No new messages.</string>
<string name="mainMessagesOneNew">Messages. One new message.</string>
<string name="mainMessagesNewFromContact">New message from&#160;</string>
<string name="mainMessagesNewFromUnknown">New message from unknown
number</string>
<string name="mainMessagesNotAvailable">Messages. Not available in this
device.</string>
<string name="mainCalls">Calls</string>
<string name="mainCallsMissed">missed calls</string>
<string name="mainCallsNoMissed">Calls. No missed calls.</string>
```

```
<string name="mainCallsOneMissed">Calls. One missed call.</string>
<string name="mainCallsNotAvailable">Calls. Not available in this device.</string>
<string name="mainCallsOneMissed">Calls. One missed call.</string>
<string name="mainCallsNotAvailable">Calls. Not available in this device.</string>
<string name="mainNotAvailable">This option is not available in this device.</string>
<string name="mainContacts">Contacts</string>
<string name="mainMusicPlayer">Music</string>
<string name="mainBrowser">Internet</string>
<string name="mainAlarms">Alarms</string>
<string name="mainAlarmsNoAlarms">Alarms. No alarms for today.</string>
<string name="mainAlarmsOneAlarm">Alarms. One alarm for today.</string>
<string name="mainAlarmsForToday">&#160;alarms for today.</string>
<string name="mainVoiceRecorder">Voice recorder</string>
<string name="mainApplications">Applications</string>
<string name="mainSettings">Settings</string>
<string name="mainStatus">Device status</string>
<string name="mainCarrierIs">, the carrier is&#160;</string>
<string name="mainProfileIsNormal">, the profile is normal&#160;</string>
<string name="mainProfileIsSilent">, the profile is silent&#160;</string>
<string name="mainProfileIsVibrate">, the profile is vibrate&#160;</string>
<string name="mainNoSignal">, there is no phone signal&#160;</string>
<string name="mainBatteryChargedAt">The battery is charged at&#160;</string>
<string name="mainPercentAndTime">&#160;percent, the time is&#160;</string>
<string name="mainHours">&#160;hours&#160;</string>
<string name="mainMinutesAndDate">&#160;minutes and the date is&#160;</string>
<string name="mainOf">&#160;of&#160;</string>
<string name="mainCarrierNotAvailable">Unknown</string>
<string name="mainWifiOnWithNetwork">and the wifi is on and connected to a wifi network named&#160;</string>
<string name="mainWifiOnWithoutNetwork">and the wifi is on and not connected to any wifi network.</string>
<string name="mainWifiOff">and the wifi is off.</string>
<string name="mainMusicPlayerPleaseTryAgain">Please try again in a few moments when the music player has processed all your music files.</string>
<string name="mainNoTTSInstalledTitle">Error</string>
<string name="mainNoTTSInstalledMessage">TTS software not found.\nPlease download Google TTS App (free App) in order to this App to work.</string>
<string name="mainNoTTSInstalledButton">Download Google TTS</string>
<string name="mainPressBack">Press Back again to close the App.</string>
<string name="monday">Monday</string>
<string name="tuesday">Tuesday</string>
<string name="wednesday">Wednesday</string>
<string name="thursday">Thursday</string>
<string name="friday">Friday</string>
<string name="saturday">Saturday</string>
<string name="sunday">Sunday</string>
```

```
<string name="january">January</string>
<string name="february">February</string>
<string name="march">March</string>
<string name="april">April</string>
<string name="may">May</string>
<string name="june">June</string>
<string name="july">July</string>
<string name="august">August</string>
<string name="september">September</string>
<string name="october">October</string>
<string name="november">November</string>
<string name="december">December</string>
<string name="volumeMaximum">Maximum volume</string>
<string name="volumeHigher">Higher volume</string>
<string name="volumeLower">Lower volume</string>
<string name="deviceCharged">The device is now fully charged</string>
<string name="deviceChargedStatus">, the device is fully charged,&#160;</string>
<string name="deviceCharging">The device is now charging</string>
<string name="deviceChargingStatus">, the device is charging,&#160;</string>
<string name="deviceUnplugged">The device is now unplugged</string>
<string name="screenOn">The screen is now on</string>
<string name="screenOff">The screen is now off</string>
<string name="screenPortrait">The screen orientation is now in portrait mode</string>
<string name="screenLandscape">The screen orientation is now in landscape mode</string>
<string name="wifiConnectedToNetwork">Connected to a wifi network named&#160;</string>
<string name="wifiDisconnectedFromNetwork">Disconnected from the wifi network</string>
<string name="layoutMainMessages">Messages</string>
<string name="layoutMainCalls">Calls</string>
<string name="layoutMainContacts">Contacts</string>
<string name="layoutMainMusic">Music</string>
<string name="layoutMainBrowser">Internet</string>
<string name="layoutMainAlarms">Alarms</string>
<string name="layoutMainSettings">Settings</string>
<string name="layoutMainStatus">Status</string>
<string name="layoutMainOnResume">Now you are at the main menu</string>
<string name="layoutAlarmsNotifierShutdown">Slide up or down to shutdown alarm</string>
<string name="layoutAlarmsList">Alarm list</string>
<string name="layoutAlarmsCreate">Create alarm</string>
<string name="layoutAlarmsAlarmCreated">The alarm was created and now you are at the alarms menu</string>
<string name="layoutAlarmsOnResume">Now you are at the alarms menu</string>
```

```xml
<string name="layoutAlarmsCreateMessage">- input message -</string>
<string name="layoutAlarmsCreateMessage2">Alarm message. Slide right to input.</string>
<string name="layoutAlarmsCreateMessage3">Alarm message:\n</string>
<string name="layoutAlarmsCreateMessage4">Alarm message.&#160;</string>
<string name="layoutAlarmsCreateCreate">Create alarm</string>
<string name="layoutAlarmsCreateDay">Day alarm:&#160;</string>
<string name="layoutAlarmsCreateDay2">Day alarm.&#160;</string>
<string name="layoutAlarmsCreateTimeHour">Alarm hours:&#160;</string>
<string name="layoutAlarmsCreateTimeHour2">Alarm hours.&#160;</string>
<string name="layoutAlarmsCreateTimeMinute">Alarm minutes:&#160;</string>
<string name="layoutAlarmsCreateTimeMinute2">Alarm minutes.&#160;</string>
<string name="layoutAlarmsCreateHours">&#160;hours</string>
<string name="layoutAlarmsCreateMinutes">&#160;minutes</string>
<string name="layoutAlarmsCreateErrorNoMessage">Error. You must input a message.</string>
<string name="layoutAlarmsCreateErrorAlarmConflict">Error. Another alarm has been set for that day and time.</string>
<string name="layoutAlarmsCreateOnResume">Now you are at the create alarm menu</string>
<string name="layoutAlarmsListNoAlarms">There are no alarms</string>
<string name="layoutAlarmsListList">Alarms</string>
<string name="layoutAlarmsListAlarmAlarmsCount">&#160;alarms has been set</string>
<string name="layoutAlarmsListAt">&#160;at&#160;</string>
<string name="layoutAlarmsListForToday">&#160;for today. Slide right or left to hear each one.</string>
<string name="layoutAlarmsListDelete">Delete selected alarm</string>
<string name="layoutAlarmsListDeleted">The alarm was deleted and now you are at the alarms list menu</string>
<string name="layoutAlarmsListDeleteSelectError">First you must select an alarm</string>
<string name="layoutAlarmsListOnResume">Now you are at the alarm list menu</string>
<string name="callFrom">Call from&#160;</string>
<string name="callFromUnknown">Call from unknown number</string>
<string name="callingTo">Calling to&#160;</string>
<string name="callingToUnknown">Calling to an unregistered number</string>
<string name="callEnded">The call has ended</string>
<string name="backToPreviousMenu">Go back to the previous menu</string>
<string name="backToMainMenu">Go back to the main menu</string>
<string-array name="keysAll">
<item>A</item>
<item>B</item>
<item>C</item>
<item>D</item>
<item>E</item>
<item>F</item>
<item>G</item>
```

```
<item>H</item>
<item>I</item>
<item>J</item>
<item>K</item>
<item>L</item>
<item>M</item>
<item>N</item>
<item>O</item>
<item>P</item>
<item>Q</item>
<item>R</item>
<item>S</item>
<item>T</item>
<item>U</item>
<item>V</item>
<item>W</item>
<item>X</item>
<item>Y</item>
<item>Z</item>
<item>0</item>
<item>1</item>
<item>2</item>
<item>3</item>
<item>4</item>
<item>5</item>
<item>6</item>
<item>7</item>
<item>8</item>
<item>9</item>
<string-array name="timeMinutesValues">
<item>00</item>
<item>05</item>
<item>10</item>
<item>15</item>
<item>20</item>
<item>25</item>
<item>30</item>
<item>35</item>
<item>40</item>
<item>45</item>
<item>50</item>
<item>55</item>
</string-array>

</resources>
```

CHAPTER VI

SYSTEM TESTING

6.1 Testing Description

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.1.1 Developing Methodology

The test process is initiated by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework from developing the testing methodologies.

6.1.2Types of Tests

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

Performance Test

The Performance test ensures that the output be produced within the time limits, and the time taken by the system for compiling, giving response to the users and request being send to the system for to retrieve the results.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Acceptance testing for Data Synchronization:

- The Acknowledgements will be received by the Sender Node after the Packets are received by the Destination Node.
- The Route add operation is done only when there is a Route request in need
- The Status of Nodes information is done automatically in the Cache Updation process

Build the test plan

Any project can be divided into units that can be further performed for detailed processing. Then a testing strategy for each of this unit is carried out. Unit testing helps to identity the possible bugs in the individual component, so the component that has bugs can be identified and can be rectified from errors.

6.2 Test Case

Home page:

Test Case	Expected Output	Actual output
Swipe down	Move to next element	Move to next element
Swipe up	Move to previous element	Move to previous element
Swipe right	Select the element	Select the element

Test Case	Expected Output	Actual output
Select Contacts	Display contacts	Display contacts
Select messages	Display messages	Display messages
Select alarm	Display alarm	Display alarm
Select setting	Display settings	Display settings

Enter message page:

Test Case	Expected Output	Actual output
Add message	Open keyboard	Open keyboard
Enter message	Swipe up and down to select	Swipe and down to select
Swipe right	Move the cursor to right to select element	Enter the element
Swipe left	Move the cursor to left to delete element	Delete the element

CHAPTER VII

CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

This application, “Displaying Smart Phone Data” is based on sharing information such as battery status; miscall info and message from android device to web application, i.e., whenever the battery drains or any miscalls or when a message comes all this information will be updated on the server system and the user can view that information using the web application from anywhere. In future, this project is useful to any Android mobile user where the user can get his mobile information data using Android Broadcasters. Before using the application, the user should register. This application detects the mobile information like SMS, call and battery information. So that the user can view the mobile data in Flex Application.

7.2 Future Enhancement

The purpose of this android application is to provide educational based Chabot for visually impaired people. It will give an answer to the educational based queries asked by the visually impaired people. They can easily launch the application with the help of Google voice search. Once the application is open, it will give a voice instruction to use an application. Output will be provided in voice form as well as in text form. So normal people can also use this application. The output can be converted to according to the user selected language

REFERENCE

[1] Hypponen M. 2006. Malware goes mobile. [Internet] Sci Am: 70−77. Available from: http://www.cs.virginia.edu/~robins/Malware_Goes_Mobile.pdf

[2] Burns J. 2009. Developing secure mobile applications for Android. Black Hat. [Internet]. Available from: https://www.isecpartners.com/files/ iSEC_Securing_Android_Apps.pdf

[3] Hoffman D. 2007. Blackjacking security threats to BlackBerry devices, PDAs and cell phones in the enterprise. Indianapolis (IN): Wiley.

[4] Android Developers. 2007. [Internet]. Available from: http://developer.android.com/ guide/topics/fundamentals.html

[5] Android Developers. 2007. [Internet]. Available from: http://developer.android.com/ guide/basics/what-is-android.html, 2007.

[6] Trend Micro. 2005. Security for mobile devices: Protecting and preserving productivity. [Internet]. Available from: http://us.trendmicro.com/imperia/md/content/us/pdf/ products/homeandhomeoffice/mobilesecurity/wp01tmms0020060104us.pdf

[7] Garfinkel S. 2010. How Android security stacks up. [Internet]. Available from: http://www.technologyreview.com/communications/24944/page2/

[8] Burns R. Symbian [Internet]. Symbian Signed. Available from: http://developer.symbian.org/wiki/index.php/Category:Symbian_Signed