**FindIt!!**

Project Report

Submitted in partial fulfillment of the requirements for the award of

the degree of

**Bachelor of Technology**

Computer Science & Engineering/ Information Technology

**BY**

***VIRESH KUMAR, 12002001068***



CSE/IT DEPARTMENT

Bhagwan Mahaveer Institute Of Engineering & Technology

Sonepat,Haryana

May 2016

Acknowledgement

It is a pleasure to acknowledge many people who knowingly and unwittingly helped us, to complete my project. First of all let us thank God for all the blessings, which carried us through all these years.

We extend my utmost gratitude to **Mr.Vikas Kuchchal** our project guide who has always stood by my side and guided, appreciated and encouraged me to get into more and more ventures. Continuing the same, he enlightened me in the various stages during the development of this project and provided me with many insights and useful examples, which proved to be of immense help in successful completion of this project.

Last but not least, I pay my sincere thanks and gratitude to all of the faculty and other Staff Members for their support and for making our training valuable and fruitful.

Viresh Kumar (12002001068)

Certificate

This is to certify that Report entitled **“FindIt !!”** which is submitted by **Viresh Kumar** inpartial fulfillment of the requirement for the award of degree B.TECH CSE, is candidate’s own work carried out by him under our supervision.

Mr. Vikas Kuchchal Mrs. Gurminder

Department of CSE/IT Department of CSE/IT

**Project guide**  **Project Incharge**

Mr. Abhinav Juneja

**HOD CSE/IT**

TABLE OF CONTENTS

ACKNOWLEDGEMENT ii

CERTIFICATE iii

LIST OF FIGURES v

----------------------------------------------------------------------------------------------------------------

1. ABSTRACT 6

2. INTRODUCTION TO PROJECT 7

3. ABOUT THE PROJECT 12

3.1 OBJECTIVES 12

3.2 REQUIREMENT ANALYSIS 12

3.3 FEASIBILITY STUDY 13

3.4 SYSTEM REQUIREMENTS 16

4. SYSTEM DESIGN 17

4.1 SOFTWARE REQUIREMENT SPECIFICATION 17

4.2 DATA FLOW DIAGRAM 18

4.3 SDLC MODEL 19

4.4 USE CASE DIAGRAM 21

4.5 ACTIVITY DIAGRAM 23

4.6 E-R DIAGRAM 25

4.7 CLASS DIAGRAM 27

5. SCRENSHOTS 29

6. CODING 36

7. DATABASE SCHEMA 60

8. TESTING 63

9. RESULT AND DISCUSSION 66

9.1 ADVANTGES 66

9.2 LIMITATIONS 66

10. CONCLUSION AND FUTURE PROSPECTS 67

11. REFERENCES 68

12. GLOSSARY 69

LIST OF FIGURES

1.DIMENSIONS OF FEASIBILITY STUDY 14

2.DATA FLOW DIAGRAM 18

3. SPIRAL MODEL 19

4. USE CASE DIAGRAM 22

5. ACTIVITY DIAGRAM 24

6. E-R DIAGRAM 26

7. CLASS DIAGRAM 28

CHAPTER 1

ABSTRACT

The purpose of developing android application “FindIt !! ” is to help people in ***finding*** the phone who continously or occasionally misses , losts or forgets the place where they kept their phone. If any user loses their phone then they call on their phone with other numbers to locate their phone . But what happens if the lost phone is on SILENT MODE .On receiving a particular message the application will automatically start playing an audio , so that user can easily locate the phone within an small area This is when FindIt !!application works.

CHAPTER 2

INTRODUCTION TO PROJECT

**FindIt !!**is the service application which helps you to find your lost phone within an small area.

In general if any userlooses there phone then they call there phone with other numbers to locate therephone . But what happens if the lost phone is on SILENT MODE . This is when FindIt !!application works. If FindIt!!

Application is running in background, then this application will read each and every message received. If phone is on silent mode then it will shifts to general mode so that if the user calls on the respective phone then he can hear audio / ringtones set by the user.

If the phone is lost in a market , public bus , school , college or any large area where it is impossible to locate phone via audio , Then by that message which is passed by other phone will enable the internet & GPS.

**What is Android ?**

Android is a [mobile operating system](https://en.wikipedia.org/wiki/Mobile_operating_system) (OS) currently developed by [Google](https://en.wikipedia.org/wiki/Google), based on the [Linux kernel](https://en.wikipedia.org/wiki/Linux_kernel) and designed primarily for [touch screen](https://en.wikipedia.org/wiki/Touchscreen) mobile devices such as [smartphones](https://en.wikipedia.org/wiki/Smartphone) and [tablets](https://en.wikipedia.org/wiki/Tablet_computer). Android's [user interface](https://en.wikipedia.org/wiki/User_interface) is mainly based on [direct manipulation](https://en.wikipedia.org/wiki/Direct_manipulation_interface), using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a [virtual keyboard](https://en.wikipedia.org/wiki/Virtual_keyboard) for text input. In addition to touchscreen devices, Google has further developed [Android TV](https://en.wikipedia.org/wiki/Android_TV) for televisions, [Android Auto](https://en.wikipedia.org/wiki/Android_Auto) for cars, and [Android Wear](https://en.wikipedia.org/wiki/Android_Wear) for wrist watches, each with a specialized user interface. Variants of Android are also used on [notebooks](https://en.wikipedia.org/wiki/Laptop), [game consoles](https://en.wikipedia.org/wiki/Video_game_console), [digital cameras](https://en.wikipedia.org/wiki/Digital_camera), and other electronics.

### Android has the largest [installed base](https://en.wikipedia.org/wiki/Installed_base) of all operating systems of any kind. Android has been the best selling OS on tablets since 2013, and on smartphones it is dominant by any metric.

**Features of Android**

**General :**

1. **Messaging**

[SMS](https://en.wikipedia.org/wiki/SMS) and [MMS](https://en.wikipedia.org/wiki/Multimedia_Messaging_Service) are available forms of messaging, including threaded [text messaging](https://en.wikipedia.org/wiki/Text_messaging) and Android Cloud To Device Messaging (C2DM) and now enhanced version of C2DM, Android Google Cloud Messaging (GCM) is also a part of Android Push Messaging services.

1. **Web browser**

The web browser available in Android is based on the open-source [Blink](https://en.wikipedia.org/wiki/Blink_(layout_engine)) (previously [WebKit](https://en.wikipedia.org/wiki/WebKit)) layout engine, coupled with [Chrome](https://en.wikipedia.org/wiki/Google_Chrome)'s [V8 JavaScript engine](https://en.wikipedia.org/wiki/V8_JavaScript_engine). Then the WebKit-using Android Browser scored 100/100 on the [Acid3](https://en.wikipedia.org/wiki/Acid3#Mobile_browsers) test on Android 4.0 ICS; the Blink-based browser currently has better standards support. The browser is variably known as 'Android Browser', '[AOSP](https://en.wikipedia.org/wiki/Android_Open_Source_Project) browser', 'stock browser', 'native browser', and 'default browser'. Starting with Android 4.4 KitKat, Google has mandated that the default browser for Android proper be [Google Chrome](https://en.wikipedia.org/wiki/Google_Chrome). Since Android 5.0 Lollipop, the WebView browser that apps can use to display web content without leaving the app has been separated from the rest of the Android firmware in order to facilitate separate security updates by Google.

1. **Voice-based features**

Google search through voice has been available since initial release. Voice actions for calling, texting, navigation, etc. are supported on Android 2.2 onwards. As of Android 4.1, Google has expanded Voice Actions with ability to talk back and read answers from Google's Knowledge Graph when queried with specific commands. The ability to control hardware has not yet been implemented.

1. **Multi-touch**

Android has native support for [multi-touch](https://en.wikipedia.org/wiki/Multi-touch) which was initially made available in handsets such as the [HTC Hero](https://en.wikipedia.org/wiki/HTC_Hero). The feature was originally disabled at the kernel level (possibly to avoid infringing Apple's patents on touch-screen technology at the time).[]](https://en.wikipedia.org/wiki/List_of_features_in_Android#cite_note-7) Google has since released an update

for the [Nexus One](https://en.wikipedia.org/wiki/Nexus_One) and the [Motorola Droid](https://en.wikipedia.org/wiki/Motorola_Droid) which enables multi-touch natively.

1. **Multitasking**

Multitasking of applications, with unique handling of memory allocation, is available.

1. **Screen capture**

Android supports capturing a [screenshot](https://en.wikipedia.org/wiki/Screenshot) by pressing the power and home-screen buttons at the same time. Prior to Android 4.0, the only methods of capturing a screenshot were through manufacturer and third-party customizations (apps), or otherwise by using a PC connection (DDMS developer's tool). These alternative methods are still available with the latest Android.

1. **TV recoding**

Android TV supports capturing [video](https://en.wikipedia.org/wiki/Video) and replay it.

1. **Video calling**

Android does not support native video calling, but some handsets have a customized version of the operating system that supports it, either via the [UMTS](https://en.wikipedia.org/wiki/UMTS) network (like the[Samsung Galaxy S](https://en.wikipedia.org/wiki/Samsung_Galaxy_S)) or over IP. Video calling through Google Talk is available in Android 2.3.4 (Gingerbread) and later. Gingerbread allows [Nexus S](https://en.wikipedia.org/wiki/Nexus_S) to place Internet calls with a SIP account. This allows for enhanced VoIP dialing to other SIP accounts and even phone numbers. Skype 2.1 offers video calling in Android 2.3, including front camera support. Users with the [Google+ Android app](https://play.google.com/store/apps/details?id=com.google.android.apps.plus&hl=en) can video chat with other Google+ users through [Hangouts](http://www.google.com/+/learnmore/hangouts/).

1. **Multiple language support**

Android supports multiple languages.

1. **Accessibility**

Built-in text-to-speech is provided by *TalkBack* for people with low or no vision. Enhancements for people with hearing difficulties are available, as are other aids.

**Connectivity :**

1. **Connectivity**

Android supports connectivity technologies including [GSM](https://en.wikipedia.org/wiki/GSM)/[EDGE](https://en.wikipedia.org/wiki/Enhanced_Data_Rates_for_GSM_Evolution), [Bluetooth](https://en.wikipedia.org/wiki/Bluetooth), [LTE](https://en.wikipedia.org/wiki/LTE_Advanced), [CDMA](https://en.wikipedia.org/wiki/Code_division_multiple_access), [EV-DO](https://en.wikipedia.org/wiki/Evolution-Data_Optimized), [UMTS](https://en.wikipedia.org/wiki/Universal_Mobile_Telecommunications_System), [NFC](https://en.wikipedia.org/wiki/Near_field_communication), [IDEN](https://en.wikipedia.org/wiki/Integrated_Digital_Enhanced_Network) and [WiMAX](https://en.wikipedia.org/wiki/WiMAX)

.

1. **Bluetooth**

Supports voice dialing and sending contacts between phones, playing music, sending files ([OPP](https://en.wikipedia.org/wiki/Object_Push_Profile)), accessing the phone book ([PBAP](https://en.wikipedia.org/wiki/Bluetooth_profile#Phone_Book_Access_Profile_.28PBAP.2C_PBA.29)), [A2DP](https://en.wikipedia.org/wiki/A2DP) and [AVRCP](https://en.wikipedia.org/wiki/AVRCP). Keyboard, mouse and joystick ([HID](https://en.wikipedia.org/wiki/Bluetooth_profile#Human_Interface_Device_Profile_.28HID.29)) support is available in Android 3.1+, and in earlier versions through manufacturer customizations and third-party applications.

1. **Tethering**

Android supports [tethering](https://en.wikipedia.org/wiki/Tethering), which allows a phone to be used as a wireless/wired [Wi-Fi hotspot](https://en.wikipedia.org/wiki/Wi-Fi_hotspot). Before Android 2.2 this was supported by third-party applications or manufacturer customizations.

**Media :**

1. **Streaming media support**

RTP/RTSP streaming ([3GPP PSS](https://en.wikipedia.org/w/index.php?title=3GPP_PSS&action=edit&redlink=1), [ISMA](https://en.wikipedia.org/wiki/Internet_Streaming_Media_Alliance)), HTML progressive download ([HTML5 <video> tag](https://en.wikipedia.org/wiki/HTML5_video)). Adobe Flash Streaming (RTMP) and HTTP Dynamic Streaming are supported by the [Flash plugin](https://en.wikipedia.org/wiki/Adobe_Flash_Player#Mobile_platforms).[[15]](https://en.wikipedia.org/wiki/List_of_features_in_Android#cite_note-15) Apple HTTP Live Streaming is supported by [RealPlayer for Android](https://en.wikipedia.org/wiki/RealPlayer_for_Android),[[16]](https://en.wikipedia.org/wiki/List_of_features_in_Android" \l "cite_note-16) and by the operating system since Android 3.0 (Honeycomb).

1. **Media support**

Android supports the following audio/video/still media formats: [WebM](https://en.wikipedia.org/wiki/WebM), [H.263](https://en.wikipedia.org/wiki/H.263), [H.264](https://en.wikipedia.org/wiki/H.264), [AAC](https://en.wikipedia.org/wiki/Advanced_Audio_Coding), [HE-AAC](https://en.wikipedia.org/wiki/HE-AAC) (in [3GP](https://en.wikipedia.org/wiki/3GP) or [MP4](https://en.wikipedia.org/wiki/MP4) [container](https://en.wikipedia.org/wiki/Container_format_(digital))), [MPEG-4 SP](https://en.wikipedia.org/wiki/MPEG-4_Part_2), [AMR](https://en.wikipedia.org/wiki/Adaptive_multi-rate_compression), [AMR-WB](https://en.wikipedia.org/wiki/AMR-WB) (in 3GP container), [MP3](https://en.wikipedia.org/wiki/MP3), [MIDI](https://en.wikipedia.org/wiki/Musical_Instrument_Digital_Interface), [Ogg Vorbis](https://en.wikipedia.org/wiki/Vorbis), [FLAC](https://en.wikipedia.org/wiki/Free_Lossless_Audio_Codec), [WAV](https://en.wikipedia.org/wiki/WAV), [JPEG](https://en.wikipedia.org/wiki/JPEG), [PNG](https://en.wikipedia.org/wiki/Portable_Network_Graphics), [GIF](https://en.wikipedia.org/wiki/Graphics_Interchange_Format), [BMP](https://en.wikipedia.org/wiki/BMP_file_format), and [WebP](https://en.wikipedia.org/wiki/WebP).

1. **External storage**

Most Android devices include microSD card slots and can read microSD cards formatted with the [FAT32](https://en.wikipedia.org/wiki/FAT32), [Ext3](https://en.wikipedia.org/wiki/Ext3) or [Ext4](https://en.wikipedia.org/wiki/Ext4) file systems. To allow use of external storage media such as [USB flash drives](https://en.wikipedia.org/wiki/USB_flash_drive) and [USB HDDs](https://en.wikipedia.org/wiki/USB_HDD), some Android devices are packaged with [USB-OTG](https://en.wikipedia.org/wiki/USB-OTG) cables. Storage formatted with [FAT32](https://en.wikipedia.org/wiki/FAT32) is handled by the [Linux Kernel](https://en.wikipedia.org/wiki/Linux_Kernel) vFAT driver, while 3rd party solutions are required to handle some other file systems such as [NTFS](https://en.wikipedia.org/wiki/NTFS), [HFS Plus](https://en.wikipedia.org/wiki/HFS%2B) and [exFAT](https://en.wikipedia.org/wiki/ExFAT).

**Hardware Support :**

Android devices can include still/video cameras,  [touchscreens](https://en.wikipedia.org/wiki/Touchscreen), [GPS](https://en.wikipedia.org/wiki/Global_Positioning_System), [accelerometers](https://en.wikipedia.org/wiki/Accelerometer) , [gyroscopes](https://en.wikipedia.org/wiki/Gyroscope), [barometers](https://en.wikipedia.org/wiki/Barometer), [magnetometers](https://en.wikipedia.org/wiki/Magnetometer), dedicated gaming controls , [proximity](https://en.wikipedia.org/wiki/Proximity_sensor) and[pressure sensors](https://en.wikipedia.org/wiki/Pressure_sensor), [thermometers](https://en.wikipedia.org/wiki/Thermometer), accelerated 2D [bit blits](https://en.wikipedia.org/wiki/Bit_blit) (with hardware orientation, scaling, pixel format conversion) and accelerated 3D graphics.

**Other :**

1. **Java support**

While most Android applications are written in [Java](https://en.wikipedia.org/wiki/Java_(programming_language)), there is no [Java Virtual Machine](https://en.wikipedia.org/wiki/Java_Virtual_Machine) in the platform and Java byte code is not executed. Java classes are compiled into Dalvik executables and run on using [Android Runtime](https://en.wikipedia.org/wiki/Android_Runtime) or in [Dalvik](https://en.wikipedia.org/wiki/Dalvik_virtual_machine) in older versions, a specialized virtual machine designed specifically for Android and optimized for battery-powered mobile devices with limited memory and CPU. [J2ME](https://en.wikipedia.org/wiki/J2ME) support can be provided via third-party applications.

1. **Handset layouts**

The platform works for various screen sizes from smartphone sizes and to tablet size, and can potentially connect to an external screen, e.g. through [HDMI](https://en.wikipedia.org/wiki/HDMI), or wirelessly with[Miracast](https://en.wikipedia.org/wiki/Miracast). Portrait and landscape orientations are supported and usually switching between by turning. A [2D graphics](https://en.wikipedia.org/wiki/2D_computer_graphics) library, [3D graphics](https://en.wikipedia.org/wiki/3D_computer_graphics) library based on [OpenGL ES](https://en.wikipedia.org/wiki/OpenGL_ES) 2.0 specifications is used.

1. **Storage**

[SQLite](https://en.wikipedia.org/wiki/SQLite), a lightweight [relational database](https://en.wikipedia.org/wiki/Relational_database), is used for [data](https://en.wikipedia.org/wiki/Data) storage purposes.

1. **Native Apps**

Android apps are also written in HTML.

CHAPTER 3

ABOUT THE PROJECT

3.1 OBJECTIVES

**FindIt !!**is the service application which helps you to find your lost phone within an small area.

3.2 REQUIREMENT ANALYSIS

The purpose of developing android application “FindIt !! ” is to help people in ***finding*** the phone who continuously or occasionally misses , losts or forgets the place where they kept there phone .

In general if any userlooses there phone then they call there phone with other numbers to locate therephone . But what happens if the lost phone is on SILENT MODE . This is when FindIt !!application works.

**The facilities provided by this application are following**

1. Phone Lock via password protection.
2. Location Tracker
   1. Enabling GPS.
   2. Enabling Internet Access.
   3. Saving co-ordinates.

3. Shifts to General Mode.

4. Plays an Audio

5 Lists of audio available .

**Phone Lock Via Password :**

When a user sends an message to the lost phone with a password , then the lost phone automatically locks the phone i.e. screen lock . This lock is enabled only via special password set by the user.

**Location Tracker :**

If the phone is lost in a market , public bus , school , college or any large area where it is impossible to locate phone via audio , Then by that message which is passed by other phone will enable the internet & GPS (also enables if these services are disabled ). These services cannot be disabled physically until the phone is unlocked via password.

**Enabling GPS :** When GPS is enabled , it will find the location by using google services. If GPS is not getting locations , then the locations can be retrieved via mobile towers .

**Enabling Internet Access :**By using this functionality locations can be directly saved on google cloud account , can be emailed or can be messaged via watsapp or SMS .

**Saving Co-ordinates :**Bysaving the co-ordinates we can keep a track of the path where the phone was commuted .The locations will be retrieved after consecutive 5 minutes of intervals .

**Shifts to general mode :**

If FindIt!! Application is running in background, then this application will read each and every message received. If phone is on silent mode then it will shifts to general mode so that if the user calls on the respective phone then he can hear audio / ringtones set by the user.

**Plays an audio:**

On receiving a particular message the application will automatically start playing an audio , so that user can easily locate the phone within an small area .

3.3 Feasibility Study

The objective of a feasibility study is to find if out an information system project can be done, and if so, how.

A feasibility study should tell us :

1. Whether the project can be done;
2. Whether the final product will benefit its intended users;
3. What are alternative solutions?
4. What are the criteria for choosing among them?
5. Is there a preferred alternative?

Preliminary investigation examines project feasibility, the likelihood of the application will be useful to the android users. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All systems are feasible if they are given unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

* Technical Feasibility
* Operation Feasibility
* Economical Feasibility

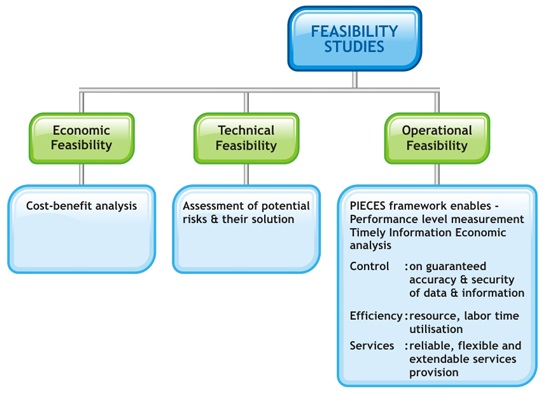


Figure 1 : Dimensions Of Feasibility Study

3.3.1 TECHNICAL FEASIBILITY

The technical issue usually raised during the feasibility stage of the investigation includes the following:

* Does the necessary technology exist to do what is suggested?
* Will the proposed system provide adequate response to services, regardless of the number or locations of users?
* Can the system be upgraded if developed?

Are there technical guarantees of accuracy, reliability, ease of access and data security?

3.3.2 OPERATIONAL FEASIBILITY

1. **User-friendly**

It has simple GUI for easy operational functionality.

1. **Reliability**
2. **Security**
3. **Portability**

The application will be developed using standard open source software (Except Oracle) like Android Java, these software will work on android phones. Hence portability problems will not arise.

1. **Availability**

This software will be available on android market.

**6. Maintainability**

The system called the ewheelz uses the 2-tier architecture. The 1st tier is the GUI, which is said to be front-end and the 2nd tier is the database, which uses My-Sql, which is the back-end.

The front-end can be run on different systems (clients). The database will be running at the server. Users access these forms by using the user-ids and the passwords.

3.3.3 ECONOMIC FEASIBILITY

Economic feasibility attempts 2 weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system.

A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case. In addition, this proves to be a

useful point of reference to compare actual costs as the project progresses. There could be various types of intangible benefits on account of automation. These could include increased customer satisfaction, improvement in product quality better decision making timeliness of information, expediting activities, improved accuracy of operations, better documentation and record keeping, faster retrieval of information, better employee morale.

3.4 SYSTEM REQUIREMNETS

**3.4.1 SOFTWARE REQUIREMENTS:**

Operating System : Android Version Ginger Bread(minimum).

Technology : Android Java.

Software’s : JDK1.8, Android Studio.

**3.4.2 HARDWARE REQUIREMENTS:**

Hardware : Android Platform based phone.

RAM : 1GB (minimum)

**3.4.3 Additional Tools:**

GUI Designing : Adobe Photoshop

Development Tool kit : Android sdk

CHAPTER 4

SYSTEM DESIGN

4.1 SOFTWARE REQUIREMENT SPECIFICATION

4.4.1 PURPOSE**:** The purpose of developing android application “FindIt !! ” is to help people in ***finding*** the phone who continuously or occasionally misses , losts or forgets the place where they kept there phone .

4.4.2 SCOPE**:** In general if any userlooses there phone then they call there phone with other numbers to locate therephone . But what happens if the lost phone is on SILENT MODE . This is when FindIt !!application works.

* 1. DATA FLOW DIAGRAM

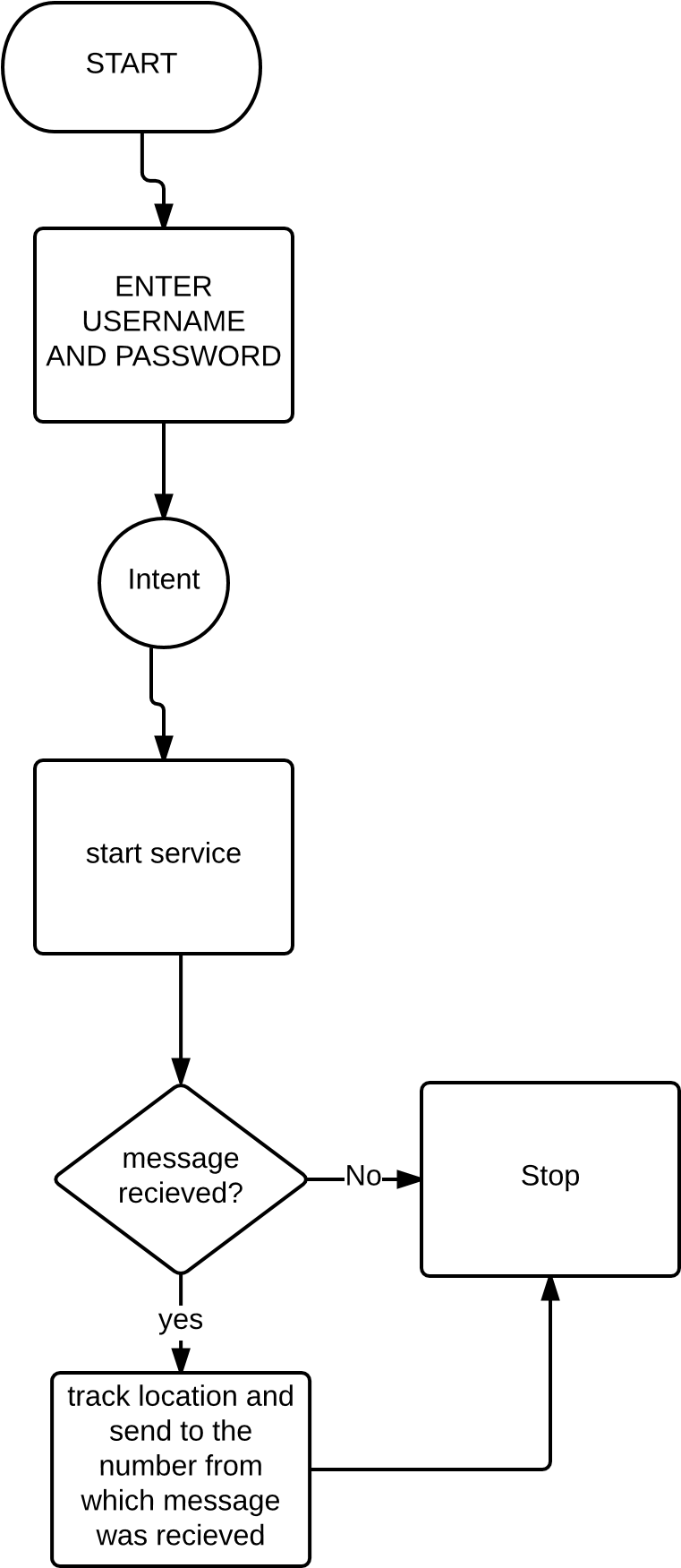
Data Flow Diagrams are used to represent the flow of data from one part of the system to another part of the system. Mainly these DFD’s are used to represent the existing system .

Figure 2

4.3 SDLC MODEL

4.3.1 SPIRAL MODEL

Spiral model is a combination of iterative development process model and sequential linear development model i.e. waterfall model with very high emphasis on risk analysis.It allows for incremental releases of the product, or incremental refinement through each iteration around the spiral.

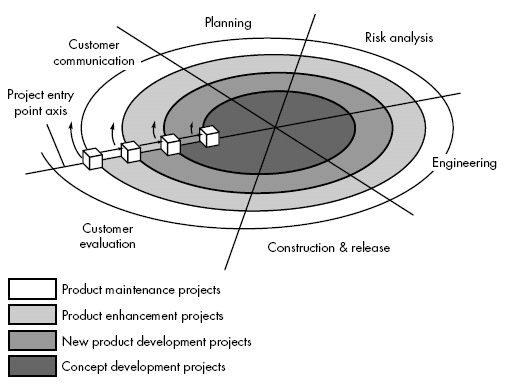
****

Figure 3

***Concept Development Project:***

Start at the core and continues for multiple iterations until it is complete.If concept is developed into an actual product, the process proceeds outward on the spiral.

***New Product Development Project:***

New product will evolve through a number of iterations around the spiral.Later, a circuit around spiral might be used to represent a “Product Enhancement Project”

***Product Enhancement Project:***

There are times when process is dormant or software team not developing new things but change is initiated, process start at appropriate entry point.It allows for elements of the product to be added in when they become available or known. This assures that there is no conflict with previous requirements and design. BUT.. it takes very strict management to complete such products and there is a risk of running the spiral in indefinite loop.

4.3.2 SPIRAL MODEL APPLICATION

Spiral Model is very widely used in the software industry as it is in synch with the natural development process of any product i.e. learning with maturity and also involves minimum risk for the customer as well as the development firms. Following are the typical uses of Spiral model:

* When costs there is a budget constraint and risk evaluation is important.
* For medium to high-risk projects.
* Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
* Customer is not sure of their requirements which is usually the case.
* Requirements are complex and need evaluation to get clarity.
* New product line which should be released in phases to get enough customer feedback.
* Significant changes are expected in the product during the development cycle.

4.4 USE CASE DIAGRAM

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different [use cases](https://en.wikipedia.org/wiki/Use_case) in which the user is involved.

The purposes of use case diagrams can be as follows:

* Used to gather requirements of a system.
* Used to get an outside view of a system.
* Identify external and internal factors influencing the system.
* Show the interacting among the requirements are actors.

Notations used in use case diagram are:

**1.Actor**- Actors are the entities that interact with a system. Although in most cases, actors are used to represent the users of system, actors can actually be anything that needs to exchange information with the system. So, an actor may be people, computer hardware, other systems, etc.

**2. Use Case**- A use case represents a user goal that can be achieved by accessing the system or software application. In Visual Paradigm, you can make use of the sub-diagram feature to describe the interaction between user and system within a use case by creating a sub-sequence diagram under a use case. You can also describe the use case scenario using the Flow of Events editor.

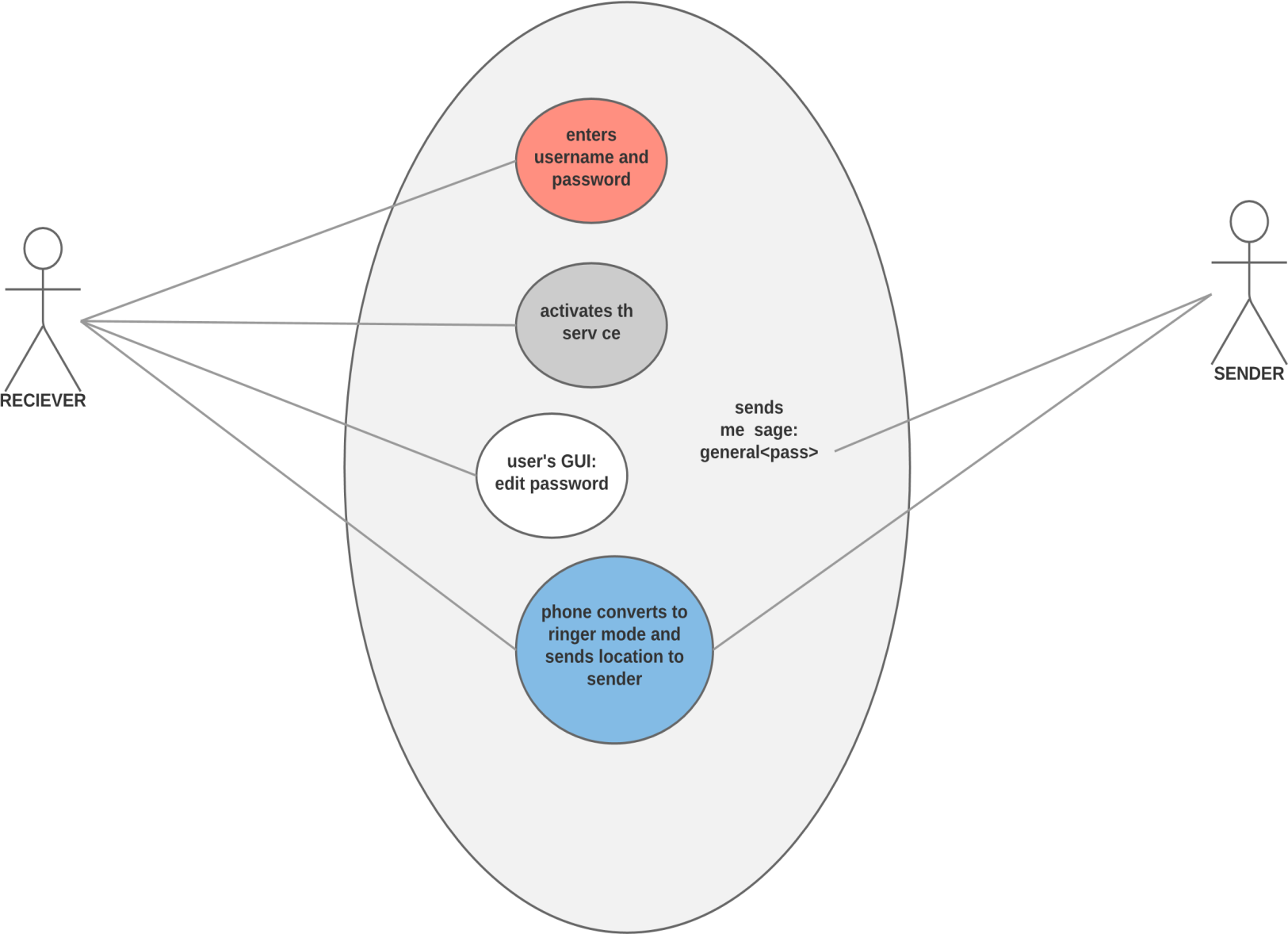


Figure 4

4.5 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actionswith support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams show the overall flow of control.

The most important notations used are:

1. Rounded rectangles represent actions;

1. Diamonds represent decisions;
2. Bars represent the start (split) or end (join) of concurrent activities;
3. A black circle represents the start (initial state) of the workflow;
4. An encircled black circle represents the end (final state).
5. Arrows run from the start towards the end and represent the order in which activities happen.

The purposes can be described as:

1. Draw the activity flow of a system.

2. Describe the sequence from one activity to another.

3. Describe the parallel, branched and concurrent flow of the system.

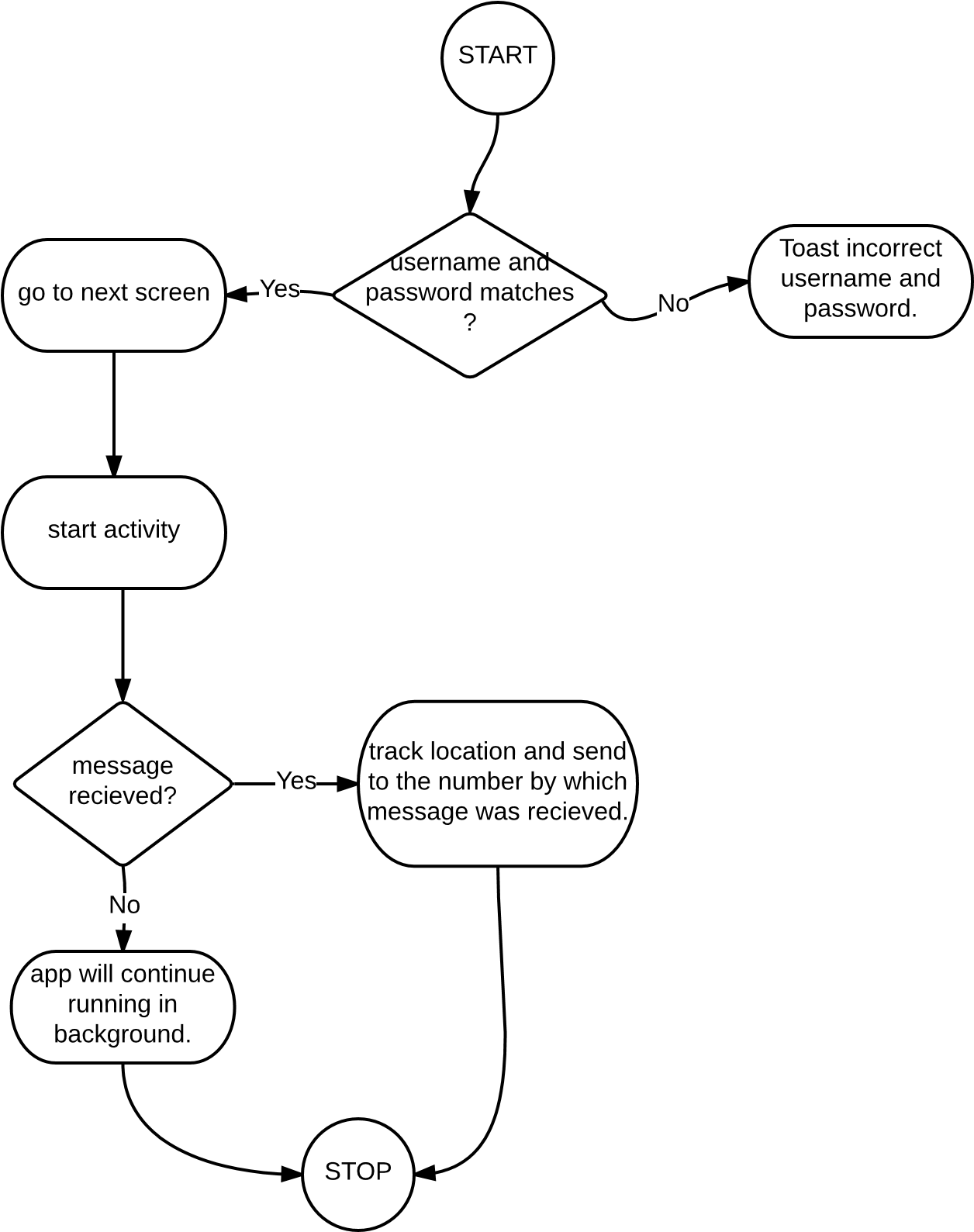


Figure 5

4.6 E-R DIAGRAM

The ER model defines the conceptual view of a database. It works around real-world entities and the associations among them. At view level, the ER model is considered a good option for designing databases.

**Entity**

An entity can be a real-world object, either animate or inanimate, that can be easily identifiable. For example, in a school database, students, teachers, classes, and courses offered can be considered as entities. All these entities have some attributes or properties that give them their identity.

An entity set is a collection of similar types of entities. An entity set may contain entities with attribute sharing similar values.

**Relationship**

The association among entities is called a relationship. For example, an employee works\_at a department, a student enrolls in a course. Here, Works\_at and Enrolls are called relationships.

**Relationship Set**

A set of relationships of similar type is called a relationship set. Like entities, a relationship too can have attributes. These attributes are called descriptive attributes.

**Degree of Relationship**

The number of participating entities in a relationship defines the degree of the relationship.

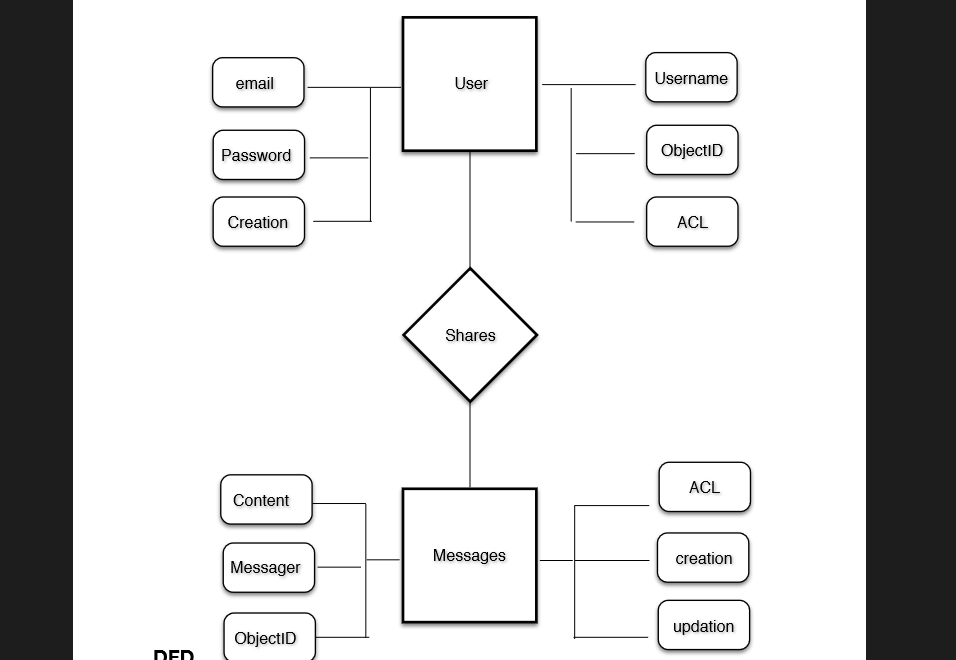
****

Figure 6

4.7 CLASS DIAGRAM

The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application.

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages.

The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a structural diagram.

**Purpose:**

The purpose of the class diagram is to model the static view of an application. The class diagrams are the only diagrams which can be directly mapped with object oriented languages and thus widely used at the time of construction.

The UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application but class diagram is a bit different. So it is the most popular UML diagram in the coder community.

So the purpose of the class diagram can be summarized as:

1.Analysis and design of the static view of an application.

2.Describe responsibilities of a system.

3.Base for component and deployment diagrams.

4.Forward and reverse engineering.

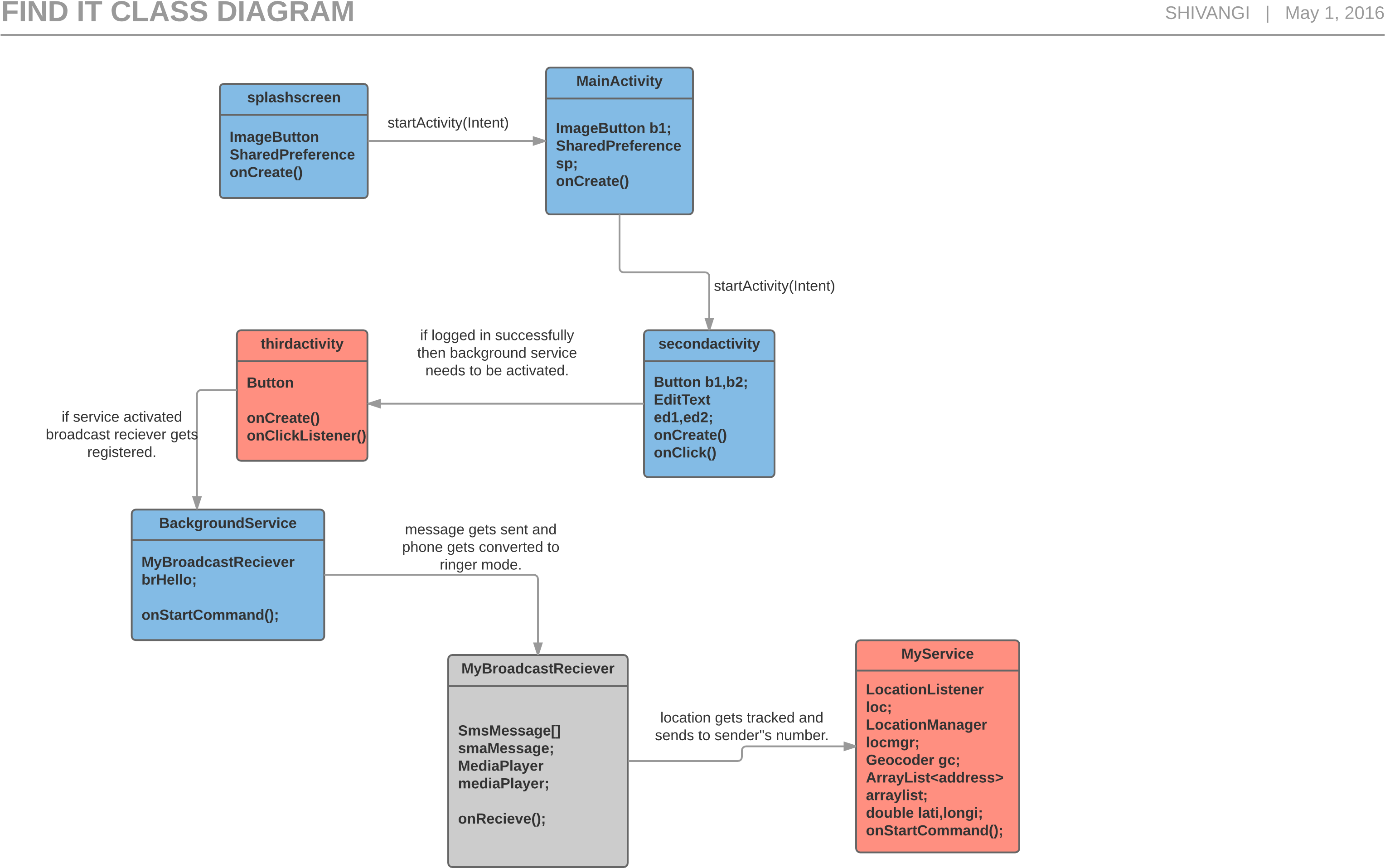


Figure 7

CHAPTER 5

SCREENSHOTS

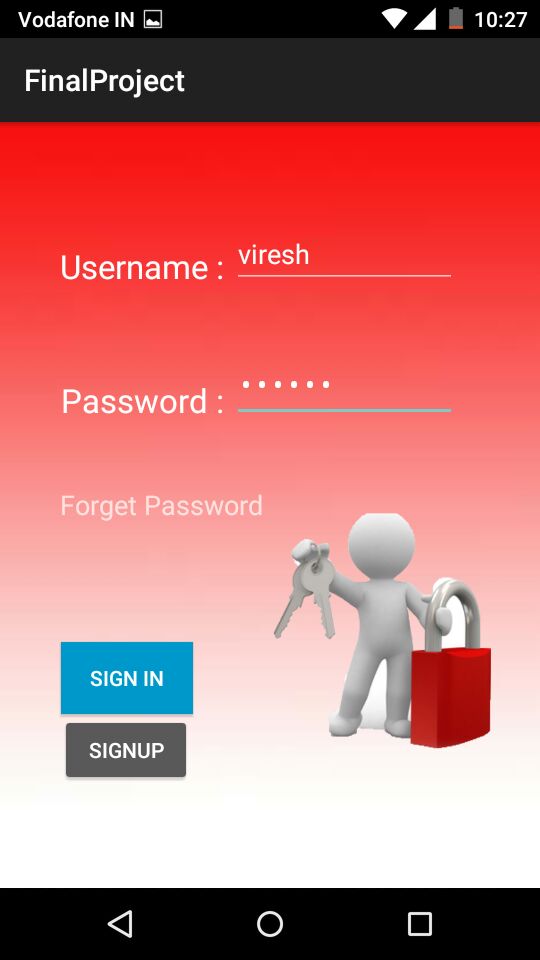
* 1. First Screen (Splash Screen)

****

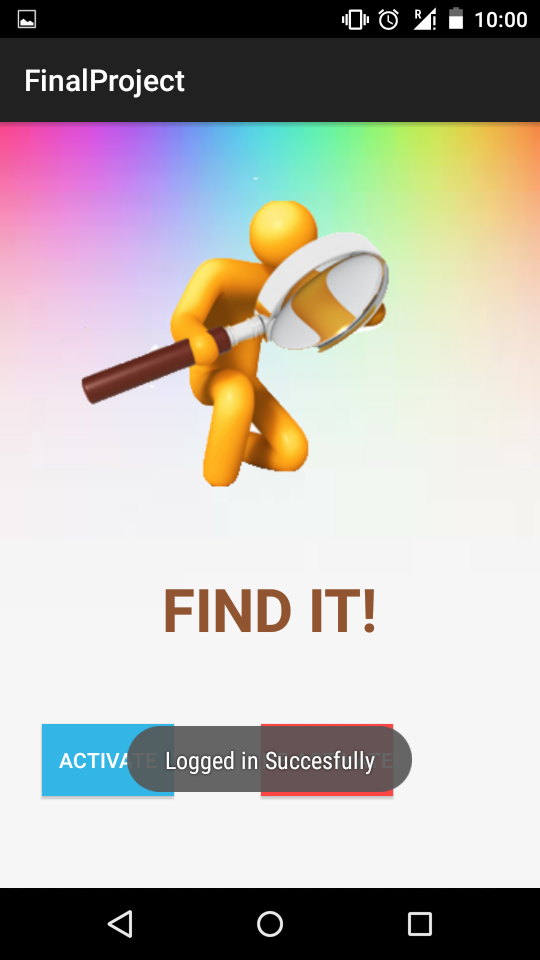
* 1. Second Screen (Introduction )

****

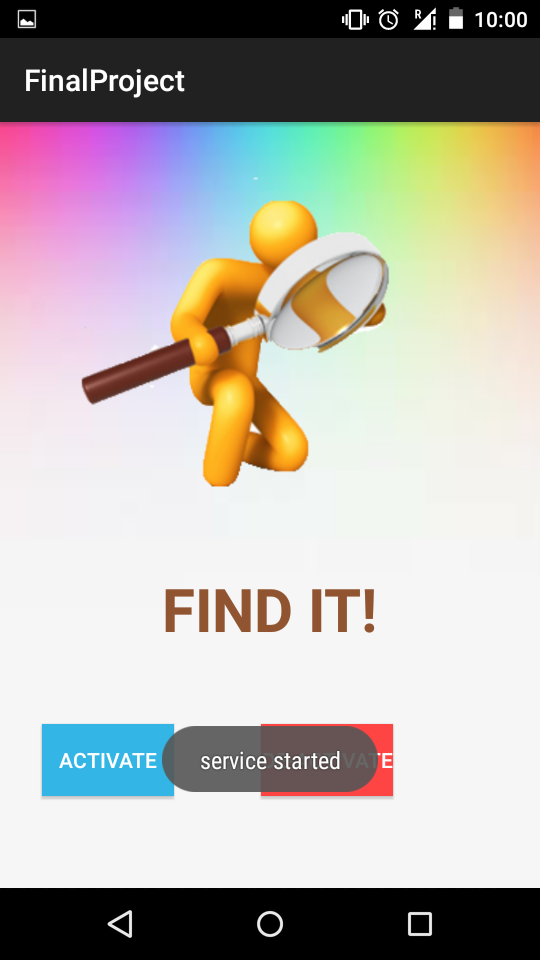
5.3 Third Screen (Login Screen)



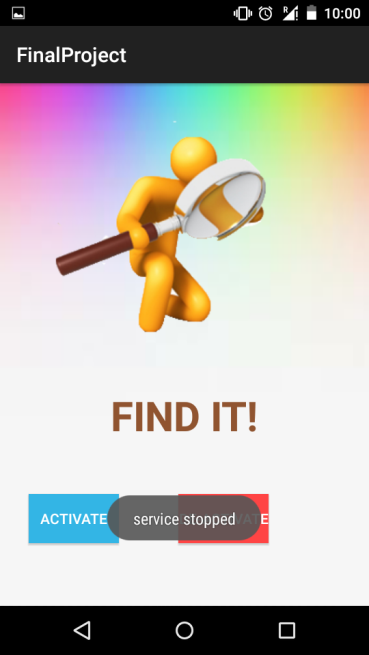
* 1. Fourth Screen(Main Screen)

****

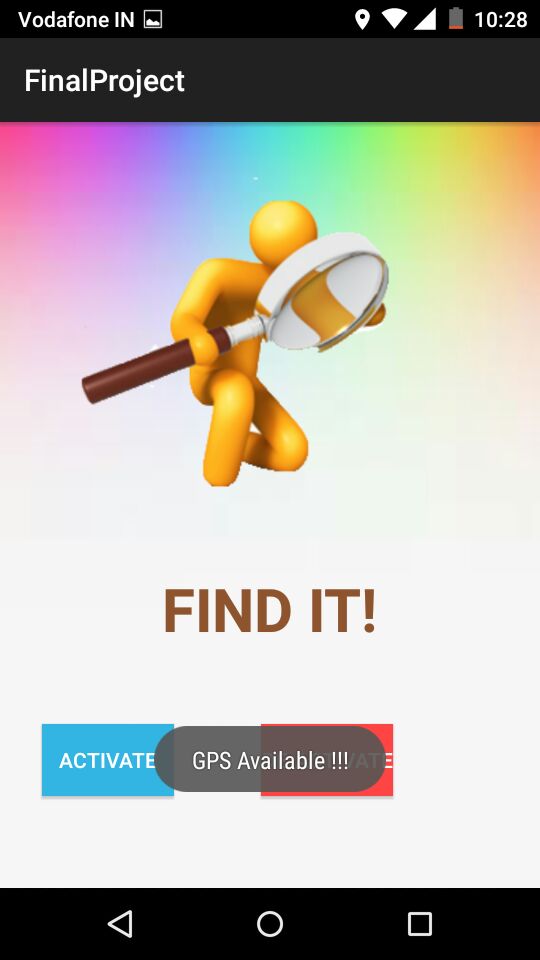
5.5 Fifth Screen(Background Service Started)

****

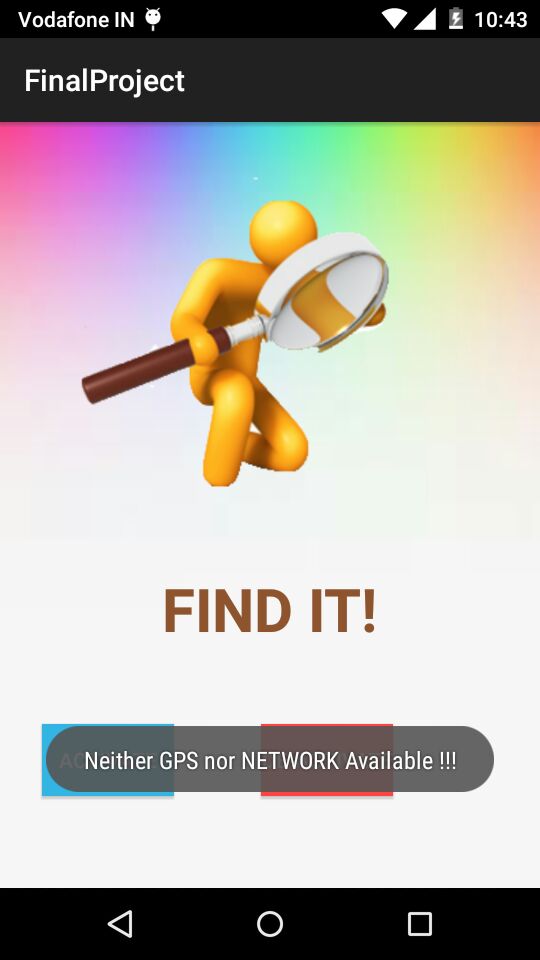
5.6 Sixth Screen (Background Service Stooped)

****

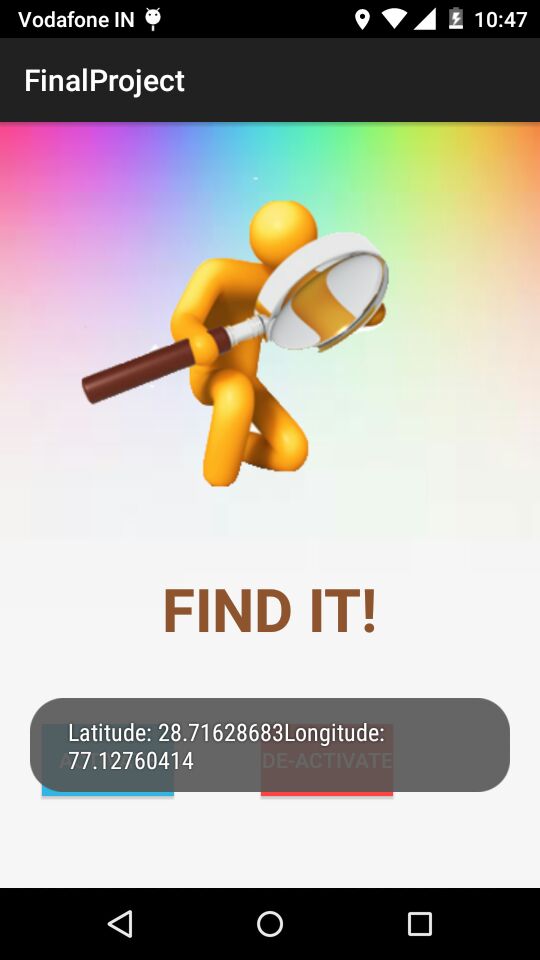
* 1. IF GPS IS AVAILABLE



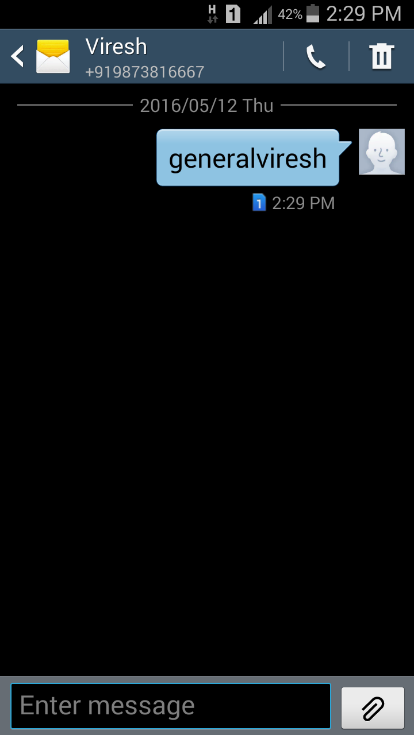
* 1. IF NEITHER GPS NOR NETWORK IS AVAILABLE



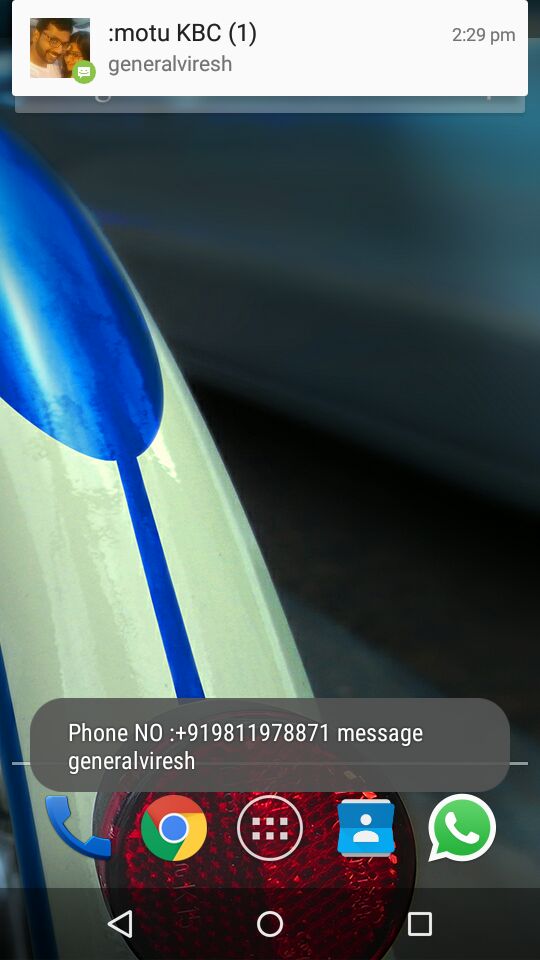
* 1. LOCATION DETECTED



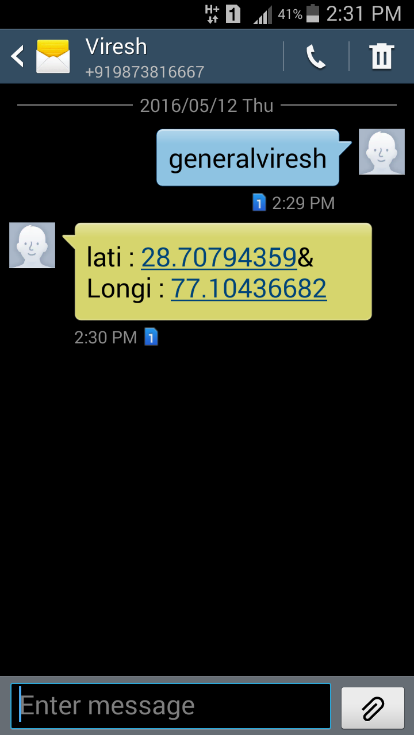
* 1. MESSAGE SENT ON LOST PHONE



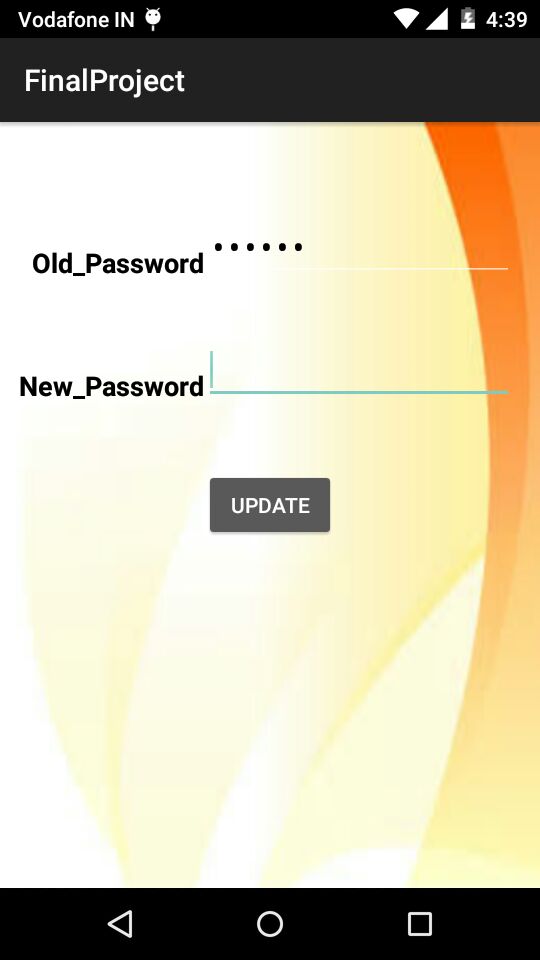
* 1. TOAST ON LOST PHONE(After receiving message)



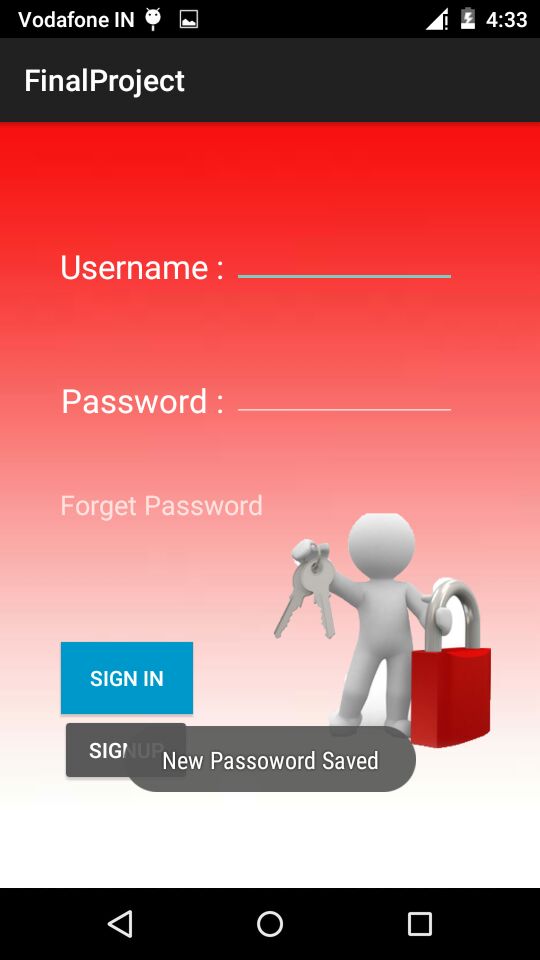
* 1. LOCATION MESSAGE RECEIVED ON SENDER’S NO.



5.13 Fourth Screen(Reset password)



* 1. PASSWORD CHANGED SUCCESSFULLY



CHAPTER 6

CODING

1. **MainActivity.java**

package com.example.vrayz.finalproject;

//prefs , activation , song->shared prefs files

import android.app.Activity;

import android.content.Intent;

import android.content.SharedPreferences;

import android.support.v7.app.ActionBarActivity;

import android.os.Bundle;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.widget.Button;

import android.widget.ImageButton;

import android.widget.Toast;

public class

MainActivity extends Activity {

//Button b1;

ImageButton b1;

Boolean act;

@Override

protected void onCreate(Bundle savedInstanceState) {

SharedPreferences sp=getSharedPreferences("prefs",MODE\_PRIVATE);

SharedPreferences.Editor speditor=sp.edit();

Boolean earlier\_activation =sp.getBoolean("activate",false);

// Toast.makeText(getApplicationContext(),"Earlier Activation was : " +earlier\_activation+ act, Toast.LENGTH\_SHORT).show();

super.onCreate(savedInstanceState);

setContentView(R.layout.firstscreen);

b1=(ImageButton)findViewById(R.id.imageButton);

b1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

startActivity(new Intent(getApplicationContext(), secondactivity.class));

}

});

//speditor.putBoolean("activate", true);

// speditor.commit();

act=sp.getBoolean("activate",false);

//Toast.makeText(getApplicationContext(), "Key saved " + act, Toast.LENGTH\_SHORT).show();

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.menu\_main, menu);

return true;

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

int id = item.getItemId();

//noinspection SimplifiableIfStatement

if (id == R.id.action\_settings) {

return true;

}

return super.onOptionsItemSelected(item);

}

}

1. **Secondactivity.java**

package com.example.vrayz.finalproject;

import android.app.Activity;

import android.content.Intent;

import android.content.SharedPreferences;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 22-07-2015.

\*/

public class secondactivity extends Activity {

boolean act;

Button b1,b2;

EditText ed1,ed2;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.secondscreen);

ed1=(EditText)findViewById(R.id.editText);

ed2=(EditText)findViewById(R.id.editText2);

TextView tv12=(TextView)findViewById(R.id.textView12);

//b1=login

b1= (Button) findViewById(R.id.signin);

b1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String n=ed1.getText().toString();

String p=ed2.getText().toString();

//Toast.makeText(getApplicationContext(), "EditText -> " + n + " " + p, Toast.LENGTH\_SHORT).show();

SharedPreferences spp=getSharedPreferences("users", MODE\_PRIVATE);

String name=spp.getString("username","");

String password=spp.getString("password","");

//Toast.makeText(getApplicationContext(),"sp.getString-> "+name+" "+password,Toast.LENGTH\_SHORT).show();

if(n!=null && p!=null && name.equals(n) && password.equals(p))

{

Toast.makeText(getApplicationContext(),"Logged in Succesfully",Toast.LENGTH\_SHORT).show();

startActivity(new Intent(getApplicationContext(), thirdactivity.class));

}

else

{

Toast.makeText(getApplicationContext(),"Wrong Username & Password ",Toast.LENGTH\_SHORT).show();

}

}

});

tv12.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

setContentView(R.layout.forgetpas);

}

});

//b2=signup

b2= (Button) findViewById(R.id.signup);

b2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String n=ed1.getText().toString();

String p=ed2.getText().toString();

SharedPreferences sp=getSharedPreferences("users", MODE\_PRIVATE);

SharedPreferences.Editor speditor=sp.edit();

speditor.putString("username", n);

speditor.putString("password", p);

speditor.commit();

// Toast.makeText(getApplicationContext(),"Values Entered ",Toast.LENGTH\_SHORT).show();

}

});

SharedPreferences sp=getSharedPreferences("prefs",MODE\_PRIVATE);

act=sp.getBoolean("activate",false);

//Toast.makeText(getApplicationContext(),""+act,Toast.LENGTH\_SHORT).show();

}

}

1. **ThirdActivity.java**

package com.example.vrayz.finalproject;

import android.app.Activity;

import android.content.Intent;

import android.content.SharedPreferences;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.support.v7.app.ActionBarActivity;

import android.support.v7.internal.widget.AdapterViewCompat;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Spinner;

import android.widget.TextView;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 22-07-2015.

\*/

public class thirdactivity extends Activity {

Button b1,b2;

// TextView pc;

//TextView tv7,tv8;

Spinner spinner;

MediaPlayer mediaPlayer;

@Override

protected void onCreate(Bundle savedInstanceState) {

setContentView(R.layout.thirdscreen);

super.onCreate(savedInstanceState);

b1=(Button)findViewById(R.id.button2);

b2=(Button)findViewById(R.id.button3);

b1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Toast.makeText(getApplicationContext(), "service started", Toast.LENGTH\_SHORT).show();

startService(new Intent(getApplicationContext(), BackgroundService.class));//error

startService(new Intent(getApplicationContext(),Myservice.class));//gps location

BackgroundService bs=new BackgroundService();

}

});

b2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Toast.makeText(getApplicationContext(),"service stopped" , Toast.LENGTH\_SHORT).show();

stopService(new Intent(getApplicationContext(), BackgroundService.class));

stopService(new Intent(getApplicationContext(), Myservice.class));

}

});

}}

1. **Fourthactivity.java**

package com.example.vrayz.finalproject;

import android.app.Activity;

import android.content.Intent;

import android.content.SharedPreferences;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 28-03-2016.

\*/

public class fourthactivity extends Activity {

EditText ed1,ed2;

Button up;

@Override

protected void onCreate(Bundle savedInstanceState) {

setContentView(R.layout.fourthscreen);

SharedPreferences spp=getSharedPreferences("users", MODE\_PRIVATE);

final String oldpassword=spp.getString("password", "");

final SharedPreferences.Editor speditor=spp.edit();

ed1=(EditText)findViewById(R.id.opass);

ed2=(EditText)findViewById(R.id.npass);

up=(Button)findViewById(R.id.update);

up.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String op=ed1.getText().toString();

String np=ed2.getText().toString();

if(oldpassword!=null && op!=null && oldpassword.equals(op))

{

speditor.putString("password", np);

speditor.apply();

Toast.makeText(getApplicationContext(),"New Passoword Saved",Toast.LENGTH\_LONG).show();

//setContentView(R.layout.seco);

startActivity(new Intent(getApplicationContext(), secondactivity.class));

}

else

Toast.makeText(getApplicationContext(),"gadbad h ",Toast.LENGTH\_LONG).show();

}

});

super.onCreate(savedInstanceState);

}

}

1. **BackgroudService.java**

package com.example.vrayz.finalproject;

import android.app.Service;

import android.content.Intent;

import android.content.IntentFilter;

import android.os.IBinder;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 22-07-2015.

\*/

public class BackgroundService extends Service {

MyBroadcastReciever brHello;

// public boolean isServicenotrunning=true;

@Override

public IBinder onBind(Intent intent) {

return null;

}

@Override

public void onCreate() {

//Toast.makeText(getApplicationContext(), "onCreate of service", Toast.LENGTH\_SHORT).show();

super.onCreate();

}

@Override

public int onStartCommand(Intent intent, int flags, int startId) {

brHello = new MyBroadcastReciever();

// Toast.makeText(getApplicationContext(), "onStart of service", Toast.LENGTH\_SHORT).show();

registerReceiver(brHello, new IntentFilter("android.provider.Telephony.SMS\_RECEIVED"));

return super.onStartCommand(intent, flags, startId);

}

@Override

public void onDestroy() {

super.onDestroy();

unregisterReceiver(brHello);

}

}

1. **MyBroadcastReciever.java**

package com.example.vrayz.finalproject;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.content.SharedPreferences;

import android.media.AudioManager;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.telephony.SmsManager;

import android.telephony.SmsMessage;

import android.view.View;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 22-07-2015.

\*/

public class MyBroadcastReciever extends BroadcastReceiver {

SmsMessage[] smsMessage;

String modeChange;

String pwd,phnno,senderNum,message;

MediaPlayer mediaPlayer;

String msg;

@Override

public void onReceive(Context context, Intent intent) {

//Toast.makeText(context, "onReceive of BroadcastReceiver", Toast.LENGTH\_SHORT).show();

Bundle b = intent.getExtras();

SharedPreferences sp=context.getSharedPreferences("users",Context.MODE\_PRIVATE);

pwd=sp.getString("password","1234");

modeChange="general"+pwd;

//Toast.makeText(context,modeChange+" ",Toast.LENGTH\_SHORT).show();

if(b!=null)

{

//Toast.makeText(context,"Enter In If Loop",Toast.LENGTH\_SHORT).show();

Object[] pdus=(Object[])b.get("pdus");

smsMessage = new SmsMessage[pdus.length];

for(int j=0;j<pdus.length; j++) {

SmsMessage smsMessage = SmsMessage.createFromPdu((byte[]) pdus[j]);

phnno = smsMessage.getDisplayOriginatingAddress();

senderNum =phnno;

message=smsMessage.getDisplayMessageBody();

Toast.makeText(context,"Phone NO :"+senderNum+" message "+ message,Toast.LENGTH\_SHORT).show();//Bilkul thik run hora h

if(message.equals(modeChange))

{

SharedPreferences sharedpref=context.getSharedPreferences("location", Context.MODE\_PRIVATE);

// SharedPreferences.Editor editor=sharedpref.edit();

String loc=sharedpref.getString("location","");

Toast.makeText(context,loc,Toast.LENGTH\_LONG).show();

SmsManager smsManager=SmsManager.getDefault();

//smsManager.sendTextMessage("9811978871" , null , "heyy" , null , null);

//Toast.makeText(context,"Boadcast wala"+coord,Toast.LENGTH\_LONG).show();//location not detected

AudioManager am=(AudioManager)context.getSystemService(Context.AUDIO\_SERVICE);

am.setRingerMode(AudioManager.RINGER\_MODE\_NORMAL);

mediaPlayer = MediaPlayer.create(context,R.raw.song1);

mediaPlayer.start();

mediaPlayer.setVolume(1, 1);

//Toast.makeText(context,"RINGER MODE NORMAL", Toast.LENGTH\_SHORT).show();

}

}

}

}

}

1. **Splashscreen.java**

package com.example.vrayz.finalproject;

import android.app.Activity;

import android.content.Intent;

import android.content.SharedPreferences;

import android.os.Bundle;

import android.widget.Button;

import android.widget.ImageButton;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 25-07-2015.

\*/

public class splashscreen extends Activity {

ImageButton imageButton;

Boolean earlier\_activation;

SharedPreferences sp;

@Override

protected void onCreate(Bundle savedInstanceState) {

setContentView(R.layout.splashwindow);

sp=getSharedPreferences("activate",MODE\_PRIVATE);

SharedPreferences.Editor speditor=sp.edit();

earlier\_activation =sp.getBoolean("activate",false);

// Toast.makeText(this, "Activate -> "+earlier\_activation, Toast.LENGTH\_SHORT).show();

super.onCreate(savedInstanceState);

String act="" , deact;

//final Intent i=new Intent(getApplicationContext(),MainActivity.class);

new Thread(new Runnable(){

@Override

public void run()

{

try {

Thread.sleep(3000);

} catch (InterruptedException e) {

e.printStackTrace();

}

startActivity(new Intent(getApplicationContext(),MainActivity.class));

/\*earlier\_activation =sp.getBoolean("activate",false);

Toast.makeText(getApplicationContext(),earlier\_activation+"",Toast.LENGTH\_SHORT).show();\*/

/\* if(earlier\_activation=true)

startActivity(new Intent(getApplicationContext(),thirdactivity.class));

else

startActivity(new Intent(getApplicationContext(),MainActivity.class));\*/

}

}).start();

}

}

1. **MyService.java**

package com.example.vrayz.finalproject;

import android.app.Service;

import java.io.IOException;

import java.util.ArrayList;

import android.content.Context;

import android.content.Intent;

import android.content.SharedPreferences;

import android.location.Address;

import android.location.Geocoder;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.location.LocationProvider;

import android.os.Bundle;

import android.os.IBinder;

import android.provider.Settings;

import android.telephony.SmsManager;

import android.widget.Toast;

/\*\*

\* Created by Vrayz on 24-04-2016.

\*/

public class Myservice extends Service{

LocationListener loc;

LocationManager locMgr;

Geocoder gc;

ArrayList<Address> addressList;

public static String coord;

Location location;

public static String addressString;

public static double lati,longi;

Intent intent;

boolean isGPS,isNetwork;

public static String provider;

@Override

public IBinder onBind(Intent arg0) {

// TODO Auto-generated method stub

return null;

}

@Override

public void onCreate() {

// TODO Auto-generated method stub

coord="Latitude: "+lati+" Longitude: "+longi;

locMgr=(LocationManager)getSystemService(LOCATION\_SERVICE);

gc=new Geocoder(getApplicationContext());

addressList=new ArrayList<Address>();

SharedPreferences sharedpref=getSharedPreferences("location", Context.MODE\_PRIVATE);

SharedPreferences.Editor editor=sharedpref.edit();

editor.putString("location",String.valueOf("latitude : "+lati+" & Longitude : "+longi));

Toast.makeText(getApplicationContext(),"saved",Toast.LENGTH\_LONG).show();

String loc=sharedpref.getString("location","");

// Toast.makeText(getApplicationContext(),loc,Toast.LENGTH\_LONG).show();

super.onCreate();

}

@Override

public int onStartCommand(Intent intent, int flags, int startId) {

// TODO Auto-generated method stub

isGPS=locMgr.isProviderEnabled(LocationManager.GPS\_PROVIDER);

isNetwork=locMgr.isProviderEnabled(LocationManager.NETWORK\_PROVIDER);

loc=new LocationListener() {

@Override

public void onStatusChanged(String arg0, int arg1, Bundle arg2) {

// TODO Auto-generated method stub

}

@Override

public void onProviderEnabled(String agr0) {

//

}

@Override

public void onProviderDisabled(String arg0) {

//commented by srishti //Intent intent = new Intent(Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS);

// commented by srishti// startActivity(intent);

}

@Override

public void onLocationChanged(Location location) {

if(location != null)

{

lati=location.getLatitude();

longi=location.getLongitude();

Toast.makeText(getApplicationContext(),"Latitude: "+lati+"Longitude: " +longi,Toast.LENGTH\_LONG ).show();

}

//SmsManager smsManager=SmsManager.getDefault();

//smsManager.sendTextMessage("9873816667", null , String.valueOf("latitude : "+lati+" & Longitude : "+longi), null , null);

}

};

if(isGPS)

{

locMgr.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 10000, 0, loc);

Toast.makeText(getApplicationContext(),"GPS Available !!!",Toast.LENGTH\_LONG).show();

}

else if(isNetwork)

{

Toast.makeText(getApplicationContext(),"NETWORK Available !!!",Toast.LENGTH\_LONG).show();

locMgr.requestLocationUpdates(LocationManager.NETWORK\_PROVIDER,1000,0,loc);

}

else

{

Toast.makeText(getApplicationContext(),"Neither GPS nor NETWORK Available !!!",Toast.LENGTH\_LONG).show();

// Intent intent1 = new Intent(Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS);

// startActivity(intent1);

}

try {

addressList=(ArrayList<Address>) gc.getFromLocation(lati, longi,1);

if(addressList.size()>0)

{

Address a=addressList.get(0);

for(int i=0;i<a.getMaxAddressLineIndex();i++)

{

addressString+=a.getAddressLine(i);

Toast.makeText(getApplicationContext(),"Location"+ addressString, Toast.LENGTH\_LONG).show();

}

}

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return super.onStartCommand(intent, flags, startId);

}

@Override

public void onDestroy() {

// TODO Auto-generated method stub

super.onDestroy();

locMgr.removeUpdates(loc);

}

}

1. **Splashscreen.xml**

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:background="@drawable/j">

<ImageView

android:layout\_width="250dp"

android:layout\_height="250dp"

android:background="@android:color/transparent"

android:id="@+id/imageView"

android:scaleType="centerCrop"

android:layout\_gravity="center\_vertical"

android:src="@drawable/icon3"

android:layout\_alignParentBottom="true"

android:layout\_alignLeft="@+id/textView9"

android:layout\_alignStart="@+id/textView9"

android:layout\_marginBottom="51dp" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="FIND IT!"

android:textColor="#905430"

android:textSize="75dp"

android:id="@+id/textView9"

android:textStyle="bold"

android:textColorHighlight="@android:color/black"

android:layout\_above="@+id/imageView"

android:layout\_centerHorizontal="true" />

</RelativeLayout>

1. **Firstscreen.xml**

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"

android:layout\_height="match\_parent" android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

android:paddingBottom="@dimen/activity\_vertical\_margin" tools:context=".MainActivity"

android:background="@drawable/orange1">

<ImageButton

android:layout\_width="90dp"

android:layout\_height="75dp"

android:scaleType="center"

android:background="@android:color/transparent"

android:id="@+id/imageButton"

android:src="@drawable/next11"

android:fadingEdge="horizontal"

android:layout\_alignParentBottom="true"

android:layout\_toRightOf="@+id/textView10"

android:layout\_toEndOf="@+id/textView10" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceLarge"

android:text="Tutorials"

android:textStyle="bold"

android:textColor="@android:color/black"

android:id="@+id/textView3"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="1. Sign-Up with a unique username and password."

android:id="@+id/textView6"

android:layout\_below="@+id/textView3"

android:layout\_alignParentLeft="true"

android:textColor="@android:color/black"

android:layout\_alignParentStart="true"

android:layout\_marginTop="42dp" />

<TextView

android:textColor="@android:color/black"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="2.Activate the Application to get started"

android:id="@+id/textView7"

android:layout\_below="@+id/textView6"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true"

android:layout\_marginTop="31dp" />

<ImageView

android:layout\_width="wrap\_content"

android:background="@android:color/transparent"

android:layout\_height="wrap\_content"

android:id="@+id/imageView5"

android:src="@drawable/pleasenote"

android:layout\_above="@+id/imageButton"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true"

android:layout\_marginBottom="57dp" />

<TextView

android:layout\_width="wrap\_content"

android:textColor="@android:color/black"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="\* To change the mode , send message as &quot;general&lt;password>&quot;"

android:id="@+id/textView8"

android:layout\_alignBottom="@+id/imageView5"

android:layout\_toRightOf="@+id/imageView5"

android:layout\_toEndOf="@+id/imageView5" />

<TextView

android:layout\_width="wrap\_content"

android:textColor="@android:color/black"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="\*For eg :- &quot;general1234&quot;"

android:id="@+id/textView10"

android:textStyle="italic"

android:layout\_above="@+id/imageButton"

android:layout\_alignLeft="@+id/textView8"

android:layout\_alignStart="@+id/textView8" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textColor="@android:color/black"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="3. PASSWORD once set cannot be changed"

android:id="@+id/textView11"

android:layout\_below="@+id/textView7"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true"

android:layout\_marginTop="31dp" />

</RelativeLayout>

1. **Secondscreen.xml**

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:background="@drawable/red8">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceLarge"

android:text="Username : "

android:id="@+id/textView"

android:layout\_marginTop="81dp"

android:layout\_marginLeft="40dp"

android:layout\_marginStart="40dp"

android:layout\_alignParentTop="true"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceLarge"

android:text="Password : "

android:id="@+id/textView2"

android:layout\_below="@+id/textView"

android:layout\_toLeftOf="@+id/editText2"

android:layout\_toStartOf="@+id/editText2"

android:layout\_marginTop="60dp" />

<EditText

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:id="@+id/editText"

android:layout\_alignBottom="@+id/textView"

android:layout\_toRightOf="@+id/textView"

android:layout\_toEndOf="@+id/textView" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Sign In"

android:background="@android:color/holo\_blue\_dark"

android:id="@+id/signin"

android:textColor="@android:color/white"

android:layout\_below="@+id/textView12"

android:layout\_alignLeft="@+id/textView2"

android:layout\_alignStart="@+id/textView2"

android:layout\_marginTop="79dp" />

<!--

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textColor="@android:color/white"

android:background="@android:color/holo\_blue\_dark"

android:text="Sign Up"

android:id="@+id/signup"

android:layout\_centerVertical="true"

android:layout\_toRightOf="@+id/imageView2"

android:layout\_toEndOf="@+id/imageView2" />

-->

<EditText

android:layout\_width="150dp"

android:layout\_height="50dp"

android:inputType="textPassword"

android:ems="10"

android:id="@+id/editText2"

android:layout\_alignBottom="@+id/textView2"

android:layout\_toRightOf="@+id/textView"

android:layout\_toEndOf="@+id/textView" />

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/imageView2"

android:layout\_alignTop="@+id/signup"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true"

android:layout\_marginLeft="154dp"

android:layout\_marginStart="154dp" />

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/imageView3"

android:src="@drawable/login3"

android:layout\_alignTop="@+id/textView2"

android:layout\_alignLeft="@+id/imageView2"

android:layout\_alignStart="@+id/imageView2" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="Forget Password"

android:id="@+id/textView12"

android:layout\_centerVertical="true"

android:layout\_alignLeft="@+id/textView"

android:layout\_alignStart="@+id/textView" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="signup"

android:id="@+id/signup"

android:layout\_below="@+id/signin"

android:layout\_alignLeft="@+id/textView12"

android:layout\_alignStart="@+id/textView12" />

</RelativeLayout>

1. **Thirdscreen.xml**

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:background="@drawable/white4">

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Activate"

android:id="@+id/button2"

android:background="@android:color/holo\_blue\_light"

android:layout\_gravity="center\_vertical"

android:textColor="@android:color/white"

android:layout\_marginRight="58dp"

android:layout\_marginEnd="58dp"

android:layout\_marginBottom="61dp"

android:layout\_alignParentBottom="true"

android:layout\_toLeftOf="@+id/button3"

android:layout\_toStartOf="@+id/button3" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="De-Activate"

android:background="@android:color/holo\_red\_light"

android:textColor="@android:color/white"

android:id="@+id/button3"

android:layout\_gravity="center\_vertical"

android:layout\_marginRight="98dp"

android:layout\_marginEnd="98dp"

android:layout\_alignTop="@+id/button2"

android:layout\_alignParentRight="true"

android:layout\_alignParentEnd="true" />

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/imageView4"

android:src="@drawable/icon3"

android:layout\_alignParentTop="true"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="FIND IT!"

android:textColor="#905430"

android:textSize="40dp"

android:id="@+id/textView9"

android:textStyle="bold"

android:textColorHighlight="@android:color/black"

android:layout\_below="@+id/imageView4"

android:layout\_centerHorizontal="true" />

</RelativeLayout>

1. **fourthscreen.xml**

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:background="@drawable/white3">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="Old\_Password"

android:id="@+id/textView13"

android:layout\_marginTop="81dp"

android:layout\_alignParentTop="true"

android:layout\_toLeftOf="@+id/opass"

android:layout\_toStartOf="@+id/opass" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="New\_Password"

android:id="@+id/textView14"

android:layout\_marginTop="57dp"

android:layout\_below="@+id/textView13"

android:layout\_toLeftOf="@+id/update"

android:layout\_toStartOf="@+id/update" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Update"

android:id="@+id/update"

android:layout\_centerVertical="true"

android:layout\_centerHorizontal="true" />

<EditText

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:inputType="textPassword"

android:ems="10"

android:id="@+id/opass"

android:layout\_above="@+id/textView14"

android:layout\_toRightOf="@+id/textView14"

android:layout\_toEndOf="@+id/textView14" />

<EditText

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:inputType="textPassword"

android:ems="10"

android:id="@+id/npass"

android:layout\_alignBottom="@+id/textView14"

android:layout\_toRightOf="@+id/textView14"

android:layout\_toEndOf="@+id/textView14" />

</RelativeLayout>

1. **Manifest.xml**

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

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.example.vrayz.finalproject" >

<uses-permission android:name="android.permission.RECEIVE\_SMS"/>

<uses-permission android:name="android.permission.READ\_SMS"/>

<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>

<uses-permission android:name="android.permission.SEND\_SMS"/>

<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>

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

<application

android:allowBackup="true"

android:icon="@drawable/icon3"

android:label="@string/app\_name"

android:theme="@android:style/Theme.DeviceDefault" >

<activity

android:name=".MainActivity"

android:label="@string/app\_name" >

</activity>

<activity

android:name=".secondactivity"

android:label="@string/app\_name">

<!-- android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen">

<!-->

</activity>

<activity

android:name=".thirdactivity"

android:label="@string/app\_name"/>

<activity

android:name=".fourthactivity"

android:label="@string/app\_name"/>

<activity

android:name=".Myservice"

android:label="FinalProject"/>

<activity

android:name=".splashscreen"

android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

<service android:name=".BackgroundService"/>

<service android:name=".Myservice"/>

</application>

</manifest>

CHAPTER 7

DATABASE SCHEMA

Shared preference has been used in our project. Android provides many ways of storing data of an application. One of this way is called Shared Preferences. Shared Preferences allow you to save and retrieve data in the form of key,value pair.

In order to use shared preferences , you have to call a method getSharedPreferences() that returns a SharedPreference instance pointing to the file that contains the values of preferences.

SharedPreferencessharedpreferences=getSharedPreferences(MyPREFERENCES,Context.MODE\_PRIVATE);

The first parameter is the key and the second parameter is the MODE.

The first parameter is the key and the second parameter is the MODE. Apart

from private there are other modes available that are listed below:

|  |  |
| --- | --- |
| **Sr.No** | **Mode and description** |
| 1 | **MODE\_APPEND**  This will append the new preferences with the already existing preferences |
| 2 | **MODE\_ENABLE\_WRITE\_AHEAD\_LOGGING**  Database open flag. When it is set , it would enable write ahead logging by default |
| 3 | **MODE\_MULTI\_PROCESS**  This method will check for modification of preferences even if the sharedpreference instance has already been loaded |
| 4 | **MODE\_PRIVATE**  By setting this mode , the file can only be accessed using calling application |
| 5 | **MODE\_WORLD\_READABLE**  This mode allow other application to read the preferences |
| 6 | **MODE\_WORLD\_WRITEABLE**  This mode allow other application to write the preferences |

You can save something in the sharedpreferences by using SharedPreferences.Editor class. You will call the edit method of SharedPreference instance and will receive it in an editor object. Its syntax is –

Editor editor = sharedpreferences.edit();

editor.putString("key", "value");

editor.commit();

Apart from the putString method , there are methods available in the editor class that allows manipulation of data inside shared preferences. They are listed as follows:

|  |  |
| --- | --- |
| **Sr. NO** | **Mode and description** |
| 1 | **apply()**  It is an abstract method. It will commit your changes back from editor to the sharedPreference object you are calling |
| 2 | **clear()**  It will remove all values from the editor |
| 3 | **remove(String key)**  It will remove the value whose key has been passed as a parameter |
| 4 | **putLong(String key, long value)**  It will save a long value in a preference editor |
| 5 | **putInt(String key, int value)**  It will save a integer value in a preference editor |
| 6 | **putFloat(String key, float value)**  It will save a float value in a preference editor |

CHAPTER 8

TESTING

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

Psychology of Testing:

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works, but the intent should be to show that a program doesn’t work. Testing is the process of executing a program with the intent of finding errors.

8.1 TESTING OBJECTIVES

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Stating formally, we can say,

* 1. Testing is a process of executing a program with the intent of finding an error.
  2. A successful test is one that uncovers an as yet undiscovered error.
  3. A good test case is one that has a high probability of finding error, if it exists.
  4. The tests are inadequate to detect possibly present errors.
  5. The software more or less confirms to the quality and reliable standards.

8.2 LEVELS OF TESING

In order to uncover the errors present in different phases we have the concept of levels of testing. The basic levels of testing are as shown below…

Acceptance Testing

System Testing

Integration Testing

Unit Testing

Client Needs

Requirements

Design

Code

8.2.1 Unit testing

*Unit testing focuses verification effort on the smallest unit of software i.e. the module. Using the detailed design and the process specifications, testing is done to uncover errors within the boundary of the module. All modules must be successful in the unit test before the start of the integration testing begins.* In our project we have done unit testing on different modules such as Tracking Location, Sending message, Changing to ringer mode, Saving values in shared preference.

8.2.2 Integration Testing

After unit testing, we have to perform integration testing. The goal here is to see if modules can be integrated properly, the emphasis being on testing interfaces between modules. This testing activity can be considered as testing the design and hence the emphasis on testing module interactions. In our project we did debugging testing by integrating modules such as when we are sending message, at that moment our phone’s mode is changing successfully to ringer mode and then after conversion is it tracking location and sending to sender’s phone.

8.2.3 System Testing

*Here the entire software system is tested. The reference document for this process is the requirements document, and the goal is to see if software meets its requirements.* After integrating all modules we tested our application as whole, is it meeting all the requirements for which it was made.Is the user able to track its lost phone.

8.2.4 Acceptance Testing

Acceptance Testing is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here is focused on external behavior of the system; the internal logic of program is not emphasized.Test cases should be selected so that the largest number of attributes of an equivalence class is exercised at once. The testing phase is an important part of software development. It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

8.2.5 White Box Testing

After integrating all modules we tested our application as whole…is it meeting all the requirements for which it was made.Is the user able to track its lost phone.

This is a unit testing method, where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors.White-box test focuses on the program control structure. Test cases are derived to ensure that all statement in the program control structure. Test cases are derived to ensure that all statement in the program has been executed at least once during testing and that all logical conditions have been exercised. Basis path testing, a white box technique, makes use of program graphs (or graph matrices) to derive the set of linearly independent test that will ensure coverage. Condition and data flow testing further exercising degrees of complexity.

8.2.6 Black Box Testing

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details at statement level. Here the module will be treated as a block that will take some input and generate output. Output for a given set of input combinations are forwarded to other modules.Black-box test are designed to uncover errors functional requirement without regard to the internal workings of a program. Black-box testing techniques focus on the information domain of the software, deriving test cases by partitioning the input and output domain of a program in manner that provides through test

coverage. The black-box test is used to demonstrate that software functions are operational, that input is properly produced, and that the integrity of external information are maintained. A black-box test examines some fundamental aspect of a system with little or no regard for the integral logical structure of the software.

CHAPTER 9

Result And Discussion

9.1 ADVANTAGES

1. An application useful for Finding,Tracing in case of PHONE-LOST.
2. We get to know the exact location of the Phone and it automatically.
3. changes the mode from silent to general and starts playing a song of user

choice.

9.2 LIMITATIONS

1. Can not work if there is no network.
2. Can not work if phone is switched off.
3. Requires another phone to send message.
4. Can not work if there is no balance in the phone to send message.

CHAPTER 10

Conclusion And Future Prospects

10.1 CONCLUSION

The main objective of the project was to create an application that can find a phone that is lost and is in silent mode. The application created is efficient in its terms, although it has some limitations but it had helped us a lot to find the lost phone and to know its exact location.

10.2 FUTURE PROSPECTS

1. Works if no network is there.
2. Works if phone is switched off.
3. Will be able to send message if no balance is there.

CHAPTER 11

REFERENCES

References for the Project Development Were Taken From the following Books and Web Sites.

1. CMC Institute.

2. Android books

CHAPTER 12

GLOSSARY

A

Activity: An activity represents a single screen with a user interface just like window or frame of Java.

Android OS: Android is a [mobile operating system](https://en.wikipedia.org/wiki/Mobile_operating_system) (OS) currently developed by [Google](https://en.wikipedia.org/wiki/Google), based on the [Linux kernel](https://en.wikipedia.org/wiki/Linux_kernel) and designed primarily for [touchscreen](https://en.wikipedia.org/wiki/Touchscreen) mobile devices such as [smartphones](https://en.wikipedia.org/wiki/Smartphone) and [tablets](https://en.wikipedia.org/wiki/Tablet_computer).

Audio Manager: You can easily control your ringer volume and ringer profile i-e:(silent,vibrate,loud e.t.c) in android. Android provides AudioManager class that provides access to these controls.

B

Backgroundservice: The [IntentService](http://developer.android.com/reference/android/app/IntentService.html) class provides a straightforward structure for running an operation on a single background thread.

Binder: Binder is a system for [inter-process communication](https://en.wikipedia.org/wiki/Inter-process_communication).

Broadcast Receivers:  Simply respond to broadcast messages from other applications or from the system itself. These messages are sometime called events or intents.

Bundle:  Bundle is used to share data between activities , and to save state of app in oncreate() method so that app will come to know where it was stopped.

G

Geocoder: A class for handling geocoding and reverse geocoding. Geocoding is the process of transforming a street address or other description of a location into a (latitude, longitude) coordinate.

GPS: The Global Positioning System (GPS) is a space-based [navigation](https://en.wikipedia.org/wiki/Satellite_navigation) system that provides location and time information in all weather conditions, anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites.

I

Intent: Intents are objects of the android.content.Intent type. Your code can send them to the Android system defining the components you are targeting.

J

Java: **Java** is a set of [computer software](https://en.wikipedia.org/wiki/Computer_software) and specifications developed by [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems), which was later acquired by the [Oracle Corporation](https://en.wikipedia.org/wiki/Oracle_Corporation), that provides a system for developing [application software](https://en.wikipedia.org/wiki/Application_software) and deploying it in a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) computing environment.

L

Location listener: Used for receiving notifications from the LocationManager when the location has changed.

Location Manager: This class provides access to the system location services. These services allow applications to obtain periodic updates of the device's geographical location, or to fire an application-specified [Intent](http://developer.android.com/reference/android/content/Intent.html) when the device enters the proximity of a given geographical location.

S

Service: A service is a component that runs in the background to perform long-running operations without needing to interact with the user and it works even if application is destroyed.

Shared Preference: Shared Preferences allow you to save and retrieve data in the form of key,value pair.

Splash screen: Splash screen is an activity that will show for set time when your app is starting and after set time period redirect to application main screen.

T

Testing: Android provides an integrated testing framework that helps you test all aspects of your app.