**hapA Project Report**

On

**“Women’s Safety App using Android”**

Submitted in partial fulfillment of the requirement of **University of Mumbai**

For the Degree of

**Bachelor of Computer Science**

Submitted By

**Akhilesh Yadav**

Under the Guidance of

**Prof. Ms. Anita Gaikwad**

Final Year Computer Science

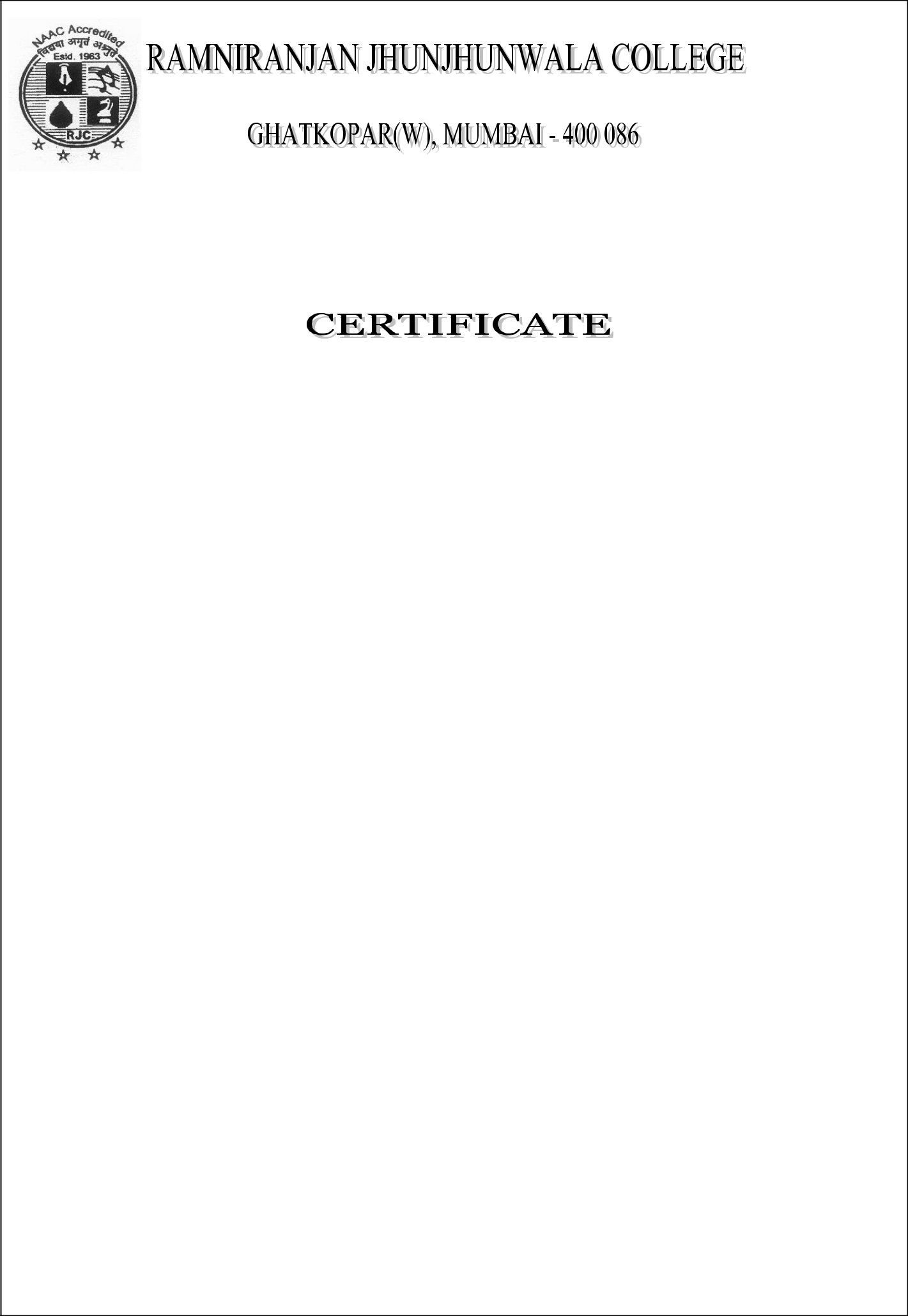


**Department of BSc. Computer Science Ramniranjan Jhunjhunwala College of Arts, Science & Commerce**

Affiliated to University of Mumbai.

Mumbai

(2019-2020)



**This is to certify that *Akhilesh Yadav* has successfully completed the project titled**

***“*Women’s Safety App using Android*”***

**under our guidance towards the partial fulfillment of degree of Bachelors of Science (Computer Science – SEM V) submitted to Ramniranjan Jhunjhunwala College, Ghatkopar (W) during the academic year 2019 – 2020 as specified by**

**UNIVERSITY OF MUMBAI.**

**Prof. Anita Gaikwad**

**Project Guide**

**Dept. Of Computer Science**

**Prof. Anita Gaikwad**

**In-Charge**

**Dept. of Computer Science**

**Declaration**

I declare that this written submission represents my own ideas in my own words and where other’s ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/fact/data/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.



**Akhilesh Yadav**

**Acknowledgement**

I have great pleasure for representing this project report entitled **“Women’s Safety App using Android”** and I grab this opportunity to convey my immense regards towards all the distinguished people who have their valuable contribution in the hour of need.

I take this opportunity to thank **“Dr. Usha Mukundan”**, our **Principal of Ramniranjan Jhunjhunwala college, Ghatkopar (W)** for giving me an opportunity to study in the instituteand the most needed guidance throughout the duration of the course.

I also like to extend my gratitude to **“Mrs. Anita Gaikwad”**, Head of the Department for their timely and prestigious guidance.

I also owe to my fellow friends who have been constant source of help to solve the problems and also helped me during the project development phase.

Thanking You,

**Akhilesh Yadav**

**INDEX**

**TABLE OF CONTENTS**

**Declaration……………………………………………………………………………….... 03**

**Acknowledgement**………………………………………………………………………… 04

1. **Project Overview**

**1 Abstract**

1.1 introduction………………………………………………………………. 09

1.1.1 Motivation……………………………………………………….. 09

1.2 Application Requirements ………………………………………………. 09

1.2.1 Time Schedule …………………………………………………... 09

1.2.2 Tracking ………………………………………………………… 10

1.2.3 Notification ……………………………………………………… 10

1.2.4 Network Provider Support ………………………………………. 10

1.2.5 GPS (Global positioning system) ……………………………….. 10

1.3 Features ………………………………………………………………....... 11

**2 Project Overview**

2.1 The Problems with Current System………………………………………. 12

2.2 Advantages Over Current System………………………………………... 12

2.3 Goals and Objectives……………………………………………………... 13

2.4 Scope and Applications ………………………………………………….. 13

**3 Requirement Analysis**

3.1 Platform Requirement…………………………………………………...... 14

3.2 Supportive Operating Systems…………………………………………… 14

3.3 Software Requirement……………………………………………………. 14

3.3.1 Backend Software Requirement………………………………… 15

3.4 Hardware Requirement…………………………………………………… 15

**4 Software Development Model**

4.1 Requirement Analysis…………………………………………………….. 16

4.2 System Design……………………………………………………………. 17

4.3 Architectural Design……………………………………………………… 17

4.4 Module Design……………………………………………………………. 17

4.5 Coding Phase……………………………………………………………... 18

4.6 Validation Phases…………………………………………………………. 18

4.6.1 Unit Testing……………………………………………………… 18

4.6.2 Integration Testing……………………………………………….. 18

4.6.3 System Testing………………………………………………… 19

4.6.4 Acceptance Testing……………………………………………… 19

**5 GANTT CHART…………………………………………………………………. 20**

5.1 System Design and Architecture…………………………………………. 23

5.1 Design Approach ………………………………………………... 23

5.1.2 System Architecture……………………………………………... 23

5.1.3 Entity-Relationship Diagram…………………………………….. 25

5.1.4 Class Diagram ………………………………………….............. 27

5.1.5 Activity Diagram………………………………………………… 28

5.1.5.1 Flow –Chart……………………………………………... 30

5.1.6 Sequence Diagram……………………………………………….. 30

5.1.7 Use Case Diagram……………………………………………….. 31

5.1.8 Deployment Diagram……………………………………………. 34

5.1.9 Component Diagram……………………………………………... 35

**6 Table Design……………………………………………………………………… 36**

**7 System Coding & Conventions**

**7.1 Program Source Code……..…………………………………37**

**7.1.1 AndroidManifest.xml…………………………………………... 37**

7.1.2 AboutUs.java…………………………………………………….. 40

7.1.3 CallBReciever.java………………………………………………. 42

7.1.4 CorruptionFragment.java………………………………………… 43

7.1.5 DomesticViolenceFragment.java………………………………... 45

7.1.6 EveTeasingFragment.java……………………………………….. 46

7.1.7 ExpandableListAdapter.java……………………………………... 48

7.1.8 FragHome.java…………………………………………………... 51

7.1.9 GalleryView.java………………………………………………… 52

7.2.0 GPSTracker.java…………………………………………………. 53

7.2.1 Help.java…………………………………………………………. 60

7.2.2 **MainActivity.java……………………………………………….** 65

7.2.3 MyBroadcastReceiver.java………………………………………. 76

7.2.4 NavigationAlert.java……………………………………………... 77

7.2.5 NavigationComplaint.java……………………………………….. 80

7.2.6 **People.java………………………………………………………. 85**

7.2.7 PeopleSecond.java……………………………………………….. 98

7.2.8 RecievedCall.java………………………………………………... 102

7.2.9 SessionManager.java…………………………………………….. 104.

7.3.0 Splash.java……………………………………………………….. 108

7.3.1 TabBars.java……………………………………………………... 109

7.3.2 TabFragment.java………………………………………………... 110

7.3.3 TheftAttemptFragment.java……………………………………... 113

7.3.4 ThreatFragment.java……………………………………………... 114

7.3.5 VoiceCallActivity.java…………………………………………... 116

7.3.6 VolumeChangeReceiver.java……………………………………. 118

**7.4 Layout**

7.4.1 about\_us.xml……………………………………………………... 119

7.4.2 activity\_main.xml………………………………………………... 121

7.4.3 activity\_navigation\_alert.xml……………………………………. 127

7.4.4 app\_bar\_navigation\_alert.xml…………………………………… 128

7.4.5 app\_bar\_navigation\_complaint.xml……………………………... 129

7.4.6 content\_navigation\_alert.xml……………………………………. 130 7.4.7 content\_navigation\_complaint.xml……………………………… 131

7.4.8 corruption.xml…………………………………………………… 133

7.4.9 domestic\_violence.xml…………………………………………... 134

7.5.0 eve\_teasing.xml………………………………………………….. 135

7.5.1 frag\_home.xml…………………………………………………… 136

7.5.2 help.xml………………………………………………………….. 137

7.5.3 list\_group.xml…………………………………………… ………. 138

7.5.4 list\_item.xml…………………………………………………… 139

7.5.5 nav\_header\_navigation\_alert.xml………………………………... 140

7.5.6 navigation\_complaint.xml……………………………… ………. 141

7.5.7 people\_second.xml………………………………………………. 142

7.5.8 people.xml……………………………………………………….. 145

7.5.9 recievedcall.xml………………………………………………….. 151

7.6.0 sent\_layout.xml…………………………………………………... 154

7.6.1 splash.xml………………………………………………………... 155

7.6.2 tab\_bars.xml……………………………………………………... 155

7.6.3 tab\_layout.xml…………………………………………………… 156

7.6.4 theft\_attempt.xml………………………………………………… 157

7.6.5 threat.xml………………………………………………………… 158

7.6.6 title.xml…………………………………………………………... 159

7.6.7 toolbar\_complaint.xml…………………………………………… 160

7.6.8 toolbar\_home.xml………………………………………………... 161

7.6.9 toolbar1.xml……………………………………………………… 162

7.7.0 voice\_layout.xml………………………………………………… 164

**8 Results and Analysis……………………………………………………………... 169**

**9 System Testing……………………………………………………………………. 175**

### 9.1 Testing the wholesystem…………………………………… 175

### 9.2 Validation

#### 9.2.1 Test Cases………………………................................................................ 177

#### Test Results and Test Data …………………………………………………………. 178

9.3 Black Box Testing ……………………………………………………….. 179

## 10 Conclusion and Future Enhancement…………………………………………... 180

## 11 References ………………………………………………………………………... 182

#### 

|  |  |  |
| --- | --- | --- |
|  | **Abstract** |  |

Security for women has become a major issue as the number of crimes over women and girls increasing day-by-day. This paper describes about women safety and their security by using electronic device to both detect The problem & alert Authorities This paper suggests a new perspective to use technology to protect women. we use an android based smart phone with an integrated feature that alert and provide location based Information. This Document describe GPS and GSM based “Women Security System “that provides the combination of GPS devices as- well-as provide alerts and message with an emergency button Trigger. Whenever somebody is in Trouble They Only have to press Volume Key Button After that a message alert is sent to Register Contact list and a Voice Call to the Number registered first and give a message “I AM In TROUBLE PLEASE HELP ME” Now a day safety of women is becoming very poor with the help of this Application The project was development in Android Which Graphical User Interface it provide the level of reliability, availability and compatibility .All these make Android an appropriate language for this project because Android language is based on JAVA language.

**Keywords: Women security, Smartphone, Registered contacts, Database, GPS (Global Positioning System), GSM (Global System for Mobile)**

# I.Introduction

This application designed to provide security to women main purpose of this application to provide the awareness on the time of critical situation for women. Generally you can active this service by clicking on ACTIVE SERVICE button. When you clicking on this button service get activate and know whenever you click on VOLUME key it open a new window and send SMS to those contact which you saved at the time of registration the SMS contain your message and your current location. This application is provide deactivate service also when we fill secure this application is mainly required correct information to fill otherwise this app will not work correctly

# II.Application Requirements

To develop the application android based mobile application the station and the environment of the surrounding has to been interviewed then the dangerous and the suspicious activates should be notified to the device by the user .if the device detects the problem then only other process creates the attention for reporting to the authority.

**1) Time Schedule:**

When the apps get installed it traces out the IMEI[International Mobile Equipment identity], SIM, mobile number and stores on the remote server, the daily schedule list gets uploaded by the person from her mobile and get tracked as per the schedule list, if there is any deviation in the schedule they relatives of the person will get an SMS along with the current location and the schedule and Police station members will get a call from the mobile.

**2) Tracking :**

The GPS will regularly transmit the latitude and longitude of the person and via GPRS we will store the entire transmitted cloud server; if the person is lost we can trace the end location with this kind of approach.

**3) Notification :**

When the person is in trouble, when she presses an alarm in the application all her relatives, voluntary organization members, police men will get an SMS with her full contact details along with the current location from where the message has been delivered. Using the IMEI number the policemen can track the location where they are travelling.

**4) Network Provider Support:**

Network provider has got the important role, by sending the commands through the short message type

to the different applications.

5) **GPS (Global positioning system):**

The GPS, elaborated as Global Positioning System, is a satellite based navigation system made up of a network of 24 satellites placed into orbit by the U.S. GPS works in any weather conditions, anywhere in the world, 24 hours a day. There are no subscription fees or setup charges to use GPS . A GPS can help us to determine exactly where we are at any given moment. Not only can a GPS give us the name of the street we might be traveling on, but many GPS systems can also give us the exact latitude and longitude of where you are located. On the other hand, Android mobile platform is becoming more popular to the users for its multidimensional purposes.Tracking System via Android Device” uses GPS and any mobile phones having an Android operating system to track the location of a person whenever necessary.

### Features

* This project presents an alert system for Women safety detection.
* The system provides a realizable and efficient.
* The application is used Advanced Encryption Standard (AES) algorithm.
* The application is easier to use all the woman.
* The application is normal budget.

**Project Overview**

# The Problems with Current System

The Drawbacks of the existing system is when emergency situation occurs the user cannot set the alert function when they are in risk situation and immediately they cannot pass their location to friends and family members to intimate they are in risk. Some of the existing comprises of three colored buttons based on the seriousness of a situation. So, you can tap the one you want depending on your situation. The problem with existing app sends panic alerts to emergency contacts in a problematic situation. It also records voices and also takes photographs during the panic situation.

# Advantages Over Current System

The proposed system has the following advantages compared to other approaches that currently existing system exist for:

a. Unlike other apps work only at danger, our app can be used as precaution measure .

b. Doesn't require Internet connection to send message as it is offline message format.

c. Easy to use and friendly interface.

# Goals and Objectives:

We are trying to contribute little efforts towards women which will ensure the safety and respect for women, so that she can also have rights to grow equally like men. Our project is based on such application which is going to help user by locating the root device through GPS and also be able to scan the QR code and message the content in text format to contacts as per user wish.

# Scope and Applications :

Here we introduce an app which ensures the safety of women. This helps to identify and call on resources to help the one out of dangerous situations. These reduce risk and bring assistance when we need it and help us to identify the location of the one in danger.

This app designed to provide security to women main purpose of this app to provide the awareness on the time of critical situation for women. Generally you can active this service by clicking on ACTIVE SERVICEBUTTON**.** When you clicking on this button service get activate and know whenever you click on VOLUME KEY it open a new window and send SMS to those contact which you saved at the time of registration the SMS contain your message and your current location. This app is provide deactivate service also when we fill secure this app is mainly required correct information to fill otherwise this app will not work correctly .

## Requirement Analysis

### Platform Requirement :

### Supportive Operating Systems:

The supported Operating Systems for client include:

1. Windows
2. Linux

Windows and Linux are two of the operating systems that will support comparative applications. Win- dows is a metafamily of graphical operating systems developed consists of several families of operating systems. The project is developed on Windows but also compatible with Linux too.

### Software Requirement :

**Front End Software Requirement .**

1.Android studio

In this project we used Android Studio offers even more features that enhance your productivity when building Android apps, such as

a. A flexible gradle based system.

b. Build variants and multiple apk file generation.

c.Code templates to help you build common app features.

d. A rich layout editor with support for drag and drop theme editing.

e. Lint tools to catch performance, usability, version compatibility, and other problems.

2. JDK(Java Development Kit)

3. JRE(Java Runtime Environment)

4. Windows or Linux Operating system

Back End Software Requirement

1. PHP
2. SQLite

Hardware Requirement :

### Hardware Required for Implementation .

1. Device with GPS support

2. Device with Text messaging support

**Hardware Requirement for Deployment .**

RAM: 4GB minimum or more

Processor: Intel i3 minimum or more

Disk Space: 500 MB for AndroidStudio, atleast 1GB for Android SDK.

## Software Development Model

**V-Model**

The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as **Verification and Validation model**.

The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. This is a highly-disciplined model and the next phase starts only after completion of the previous phase.

1. **Requirement Analysis**

This is the first phase in the development cycle where the product requirements are understood from the customer’s perspective. This phase involves detailed communication with the customer to understand his expectations and exact requirement. This is a very important activity and needs to be managed well, as most of the customers are not sure about what exactly they need. The **acceptance test design planning** is done at this stage as business requirements can be used as an input for acceptance testing.

1. **System Design**

Once you have the clear and detailed product requirements, it is time to design the complete system. The system design will have the understanding and detailing the complete hardware and communication setup for the product under development. The system test plan is developed based on the system design. Doing this at an earlier stage leaves more time for the actual test execution later.

1. **Architectural Design**

Architectural specifications are understood and designed in this phase. Usually more than one technical approach is proposed and based on the technical and financial feasibility the final decision is taken. The system design is broken down further into modules taking up different functionality. This is also referred to as **High Level Design (HLD)**.

The data transfer and communication between the internal modules and with the outside world (other systems) is clearly understood and defined in this stage. With this information, integration tests can be designed and documented during this stage.

1. **Module Design**

In this phase, the detailed internal design for all the system modules is specified, referred to as **Low Level Design (LLD)**. It is important that the design is compatible with the other modules in the system architecture and the other external systems. The unit tests are an essential part of any development process and helps eliminate the maximum faults and errors at a very early stage. These unit tests can be designed at this stage based on the internal module designs.

1. **Coding Phase**

The actual coding of the system modules designed in the design phase is taken up in the Coding phase. The best suitable programming language is decided based on the system and architectural requirements.

The coding is performed based on the coding guidelines and standards. The code goes through numerous code reviews and is optimized for best performance before the final build is checked into the repository.

1. **Validation Phases**

The different Validation Phases in a V-Model are explained in detail below.

1. **Unit Testing**

Unit tests designed in the module design phase are executed on the code during this validation phase. Unit testing is the testing at code level and helps eliminate bugs at an early stage, though all defects cannot be uncovered by unit testing.

1. **Integration Testing**

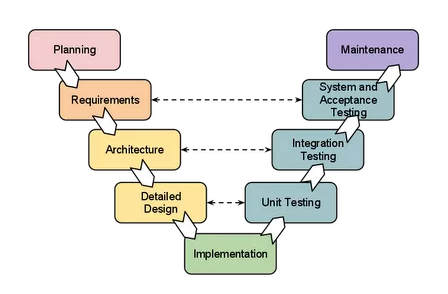
Integration testing is associated with the architectural design phase. Integration tests are performed to test the coexistence and communication of the internal modules within the system.

1. **System Testing**

System testing is directly associated with the system design phase. System tests check the entire system functionality and the communication of the system under development with external systems. Most of the software and hardware compatibility issues can be uncovered during this system test execution.

1. **Acceptance Testing**

Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment. Acceptance tests uncover the compatibility issues with the other systems available in the user environment. It also discovers the non-functional issues such as load and performance defects in the actual user environment.

****

**GANTT CHART**

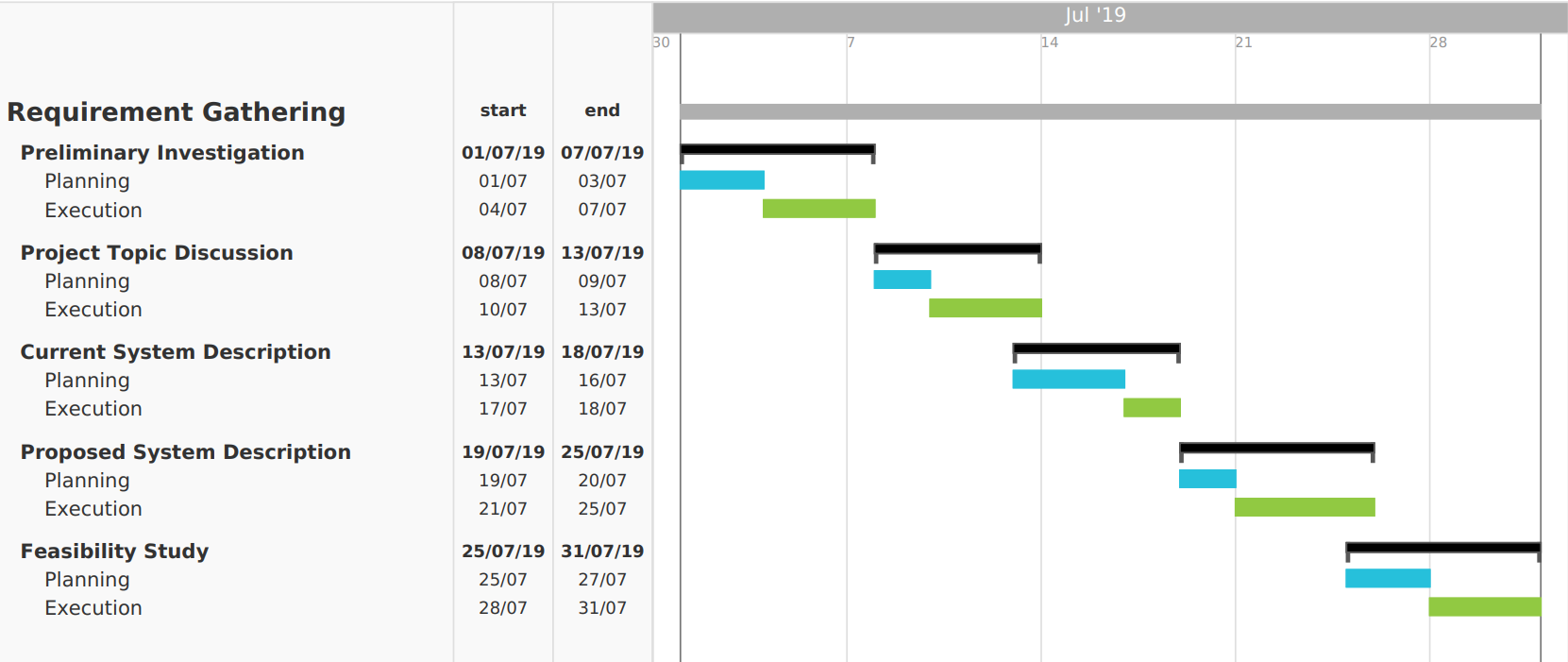
A Gantt chart is a type of bar chart that illustrates a project schedule. Gantt chart illustrates the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project. Gantt Charts are useful tools for analyzing and planning more complex projects.

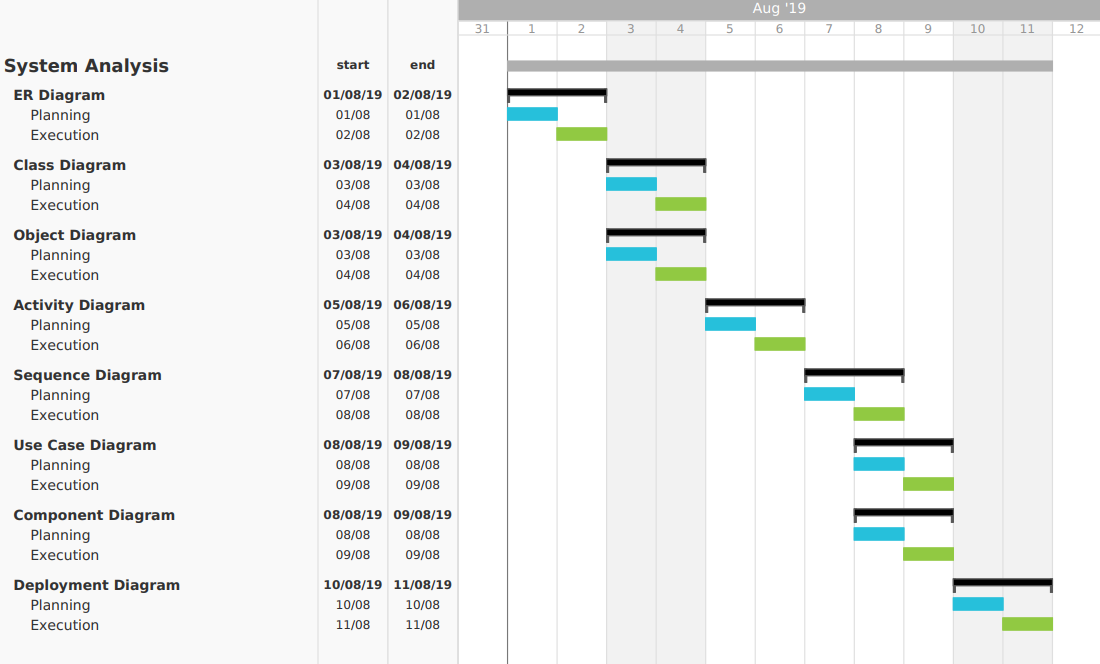
They:

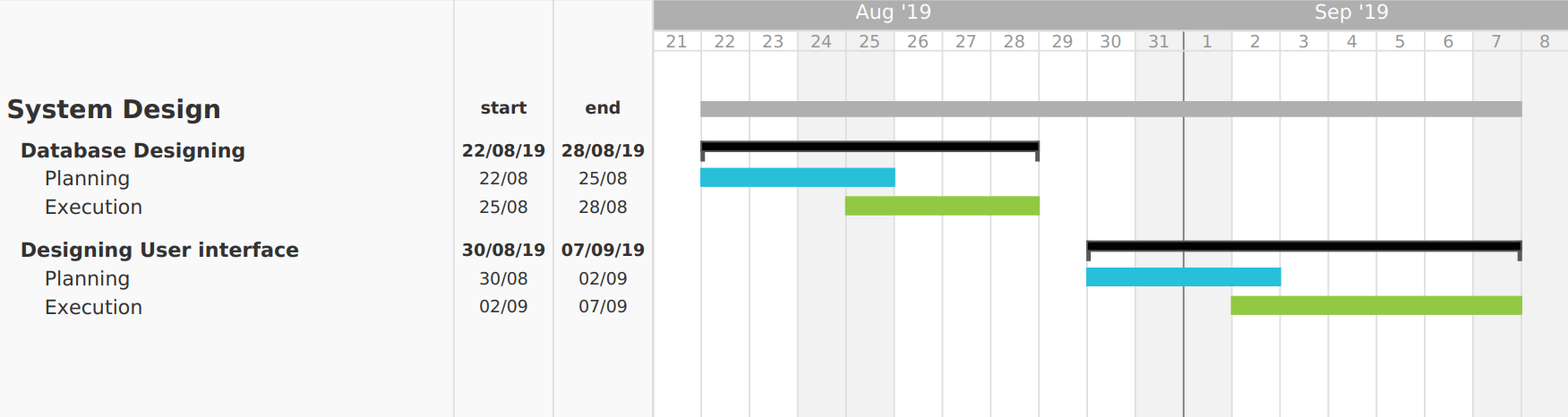
* Help you to plan out the tasks that need to be completed.
* Give you a basis for scheduling when these tasks will be carried out.
* Help you to work out critical path for a project where you must be complete it by a particular date.

When a project is under way, Gantt chart helps you to monitor whether the project is on schedule. If it is not, it allows you to pinpoint the remedial action necessary to put it back on schedule. This chart is also used in Information Technology to represent data that has been collected.

|  |  |  |  |
| --- | --- | --- | --- |
| **Title** | **Start date** | **End date** | **Duration** |
| **Requirement Analysis** | | |  |
| Preliminary Investigation | 01/07/19 | 07/07/19 | 6 |
| Project Topic Discussion | 08/07/19 | 13/07/19 | 5 |
| Current System Description | 13/07/19 | 18/07/19 | 5 |
| Proposed System Description | 19/07/19 | 25/07/19 | 6 |
| Feasibility Study | 25/07/19 | 31/07/19 | 6 |
| **System Analysis** | | |  |
| ER Diagram | 01/08/19 | 02/08/19 | 1 |
| Class Diagram | 03/08/19 | 04/08/19 | 1 |
| Object Diagram | 03/08/19 | 04/08/19 | 1 |
| Activity Diagram | 05/08/19 | 06/08/19 | 3 |
| Sequence Diagram | 07/08/19 | 08/08/19 | 1 |
| Use Case Diagram | 08/08/19 | 09/08/19 | 1 |
| Component Diagram | 08/08/19 | 09/08/19 | 1 |
| Deployment Diagram | 10/08/19 | 11/08/19 | 1 |
| System Design | 22/08/19 | 07/09/19 | 15 |
| **System Coding**  21-10-2019 19-12-2019 | | |  |
| **System Testing**  02-01-2020 14-02-2020 | | |  |
| **Implementation**  15-02-2020 25-02-2020 | | |  |







## System Design and Architecture

### Design Approach

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization. Once the soft- ware requirements have been analyzed and specified the software design involves the technical activities design, coding, implementation and testing that are required to build and verify the software. The design activities are of main importance in this phase, because in this activity decision ultimately affecting the success of the software implementation and its ease of maintenance are made. These decision have the final bearing upon realibility and maintainability of the system. design is the only way to accurately transalte the customer requirements into finished software or a system. Design is the place where qualityis fostered in development. software design is a process through which requirements are translated into a representation of software.

### System Architecture

Every software has a model. Before the implementation of the software, architecture is drawn in the form of any model or any diagram. So it become quite easier to understand the flow of the software and it also help in the easier implementation of the software.

The client interacts with the developer team so that client can make them understand that what kind of software he wants. According to his idea the developer team makes diagram so that it can become easy for the both of the parties. If any changes are to be made, then the client could also ask the developer to make the required changes.

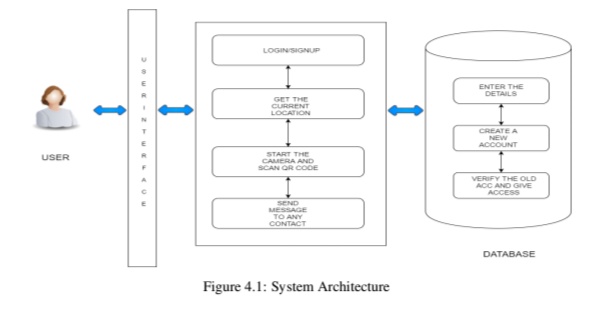
The explanation of the above architecture is as follows:

System architecture of proposed method consists of mainly two tasks. In below architecture, we have a user with his/her smartphone by which user set GPS on, then the current location of user fetches.

These all processing does not required any Internet connection. After this, now user will send an offline message to any of the contact in his/her list.

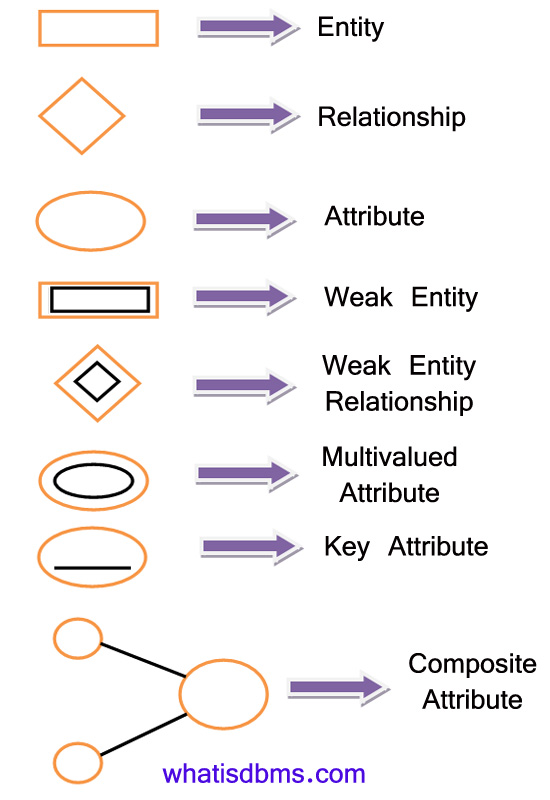
## 

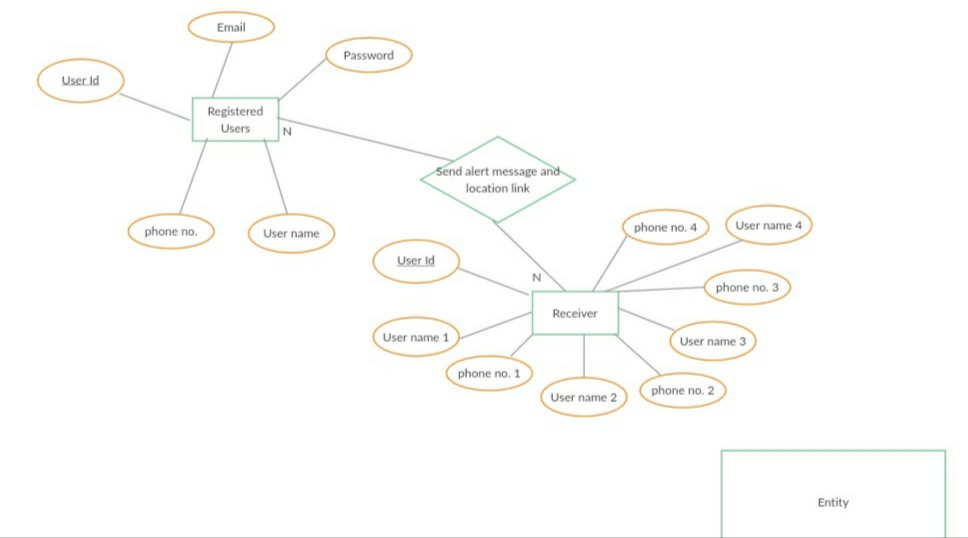
\_



**Entity-Relationship Diagram**

* This document is an entity-relationship diagram, or “ERD”, for a system to manage Inventory Management System.
* An ERD is a model that identifies the concepts or entities that exists in a system and the relationship between those entities.
* An ERD is often used as a way to visualize a relational database: each entity represents a database table, and the relationship lines represent the keys in one table that point to specific records in related tables.
* ERD may also be more abstract, not necessarily capturing every table needed within a database, but serving to diagram the major concepts and relationships.
* This ERD is of the latter type, intended to present an abstract, theoretical view of the major entities and relationships needed for management of electronic resources.
* It may assist the database design process for an e-resource management system, but does not identify every table that would be necessary for an electronic resource management database.





**Class Diagram**

It is a model which is used to show the classes constituting a system and their interrelationship. It is based on UML. Only the important attributes and methods are shown in Class diagrams. In the initial period of analysis, the important attributes of the classes, which must be captured and the functionalities provided by the class may not be very clear. As the analysis progresses, the attributes and methods may be added. If more focus is on interrelationships of classes, then the attributes and methods may not be shown in the class diagram.

The class diagram is used to identify and classify the objects which constitute a system. It also includes the important attributes of the objects which must be captured.

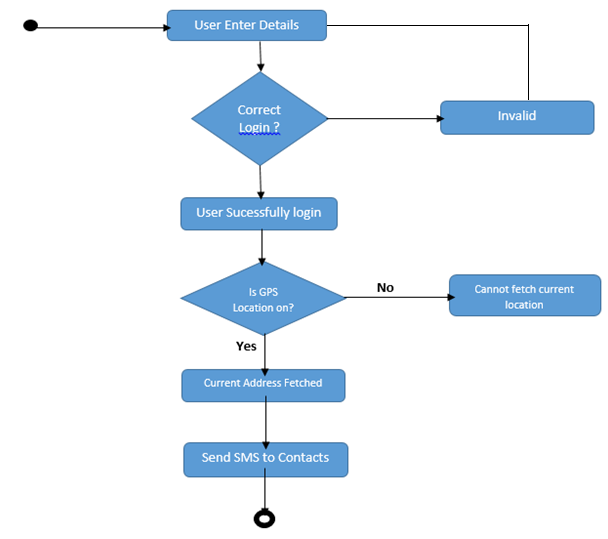
### IMG_20190814_215437.jpgClass Diagram :

**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. An activity diagram shows the overall flow of control. Activity diagrams are constructed from a limited repertoire of shapes, connected with arrows. Activity diagrams are constructed from a limited repertoire of shapes, connected with arrows. The most important shape types:

* Rounded rectangle represent activities.
* Diamonds represent decisions.
* Bars represent the start (split) or end (join) of concurrent activities.
* A black circle represents the start (initial state) of the workflow.
* An encircled black circle represents the end (final state).
* Arrows run from the start towards the end and represent the order in which activities happen.

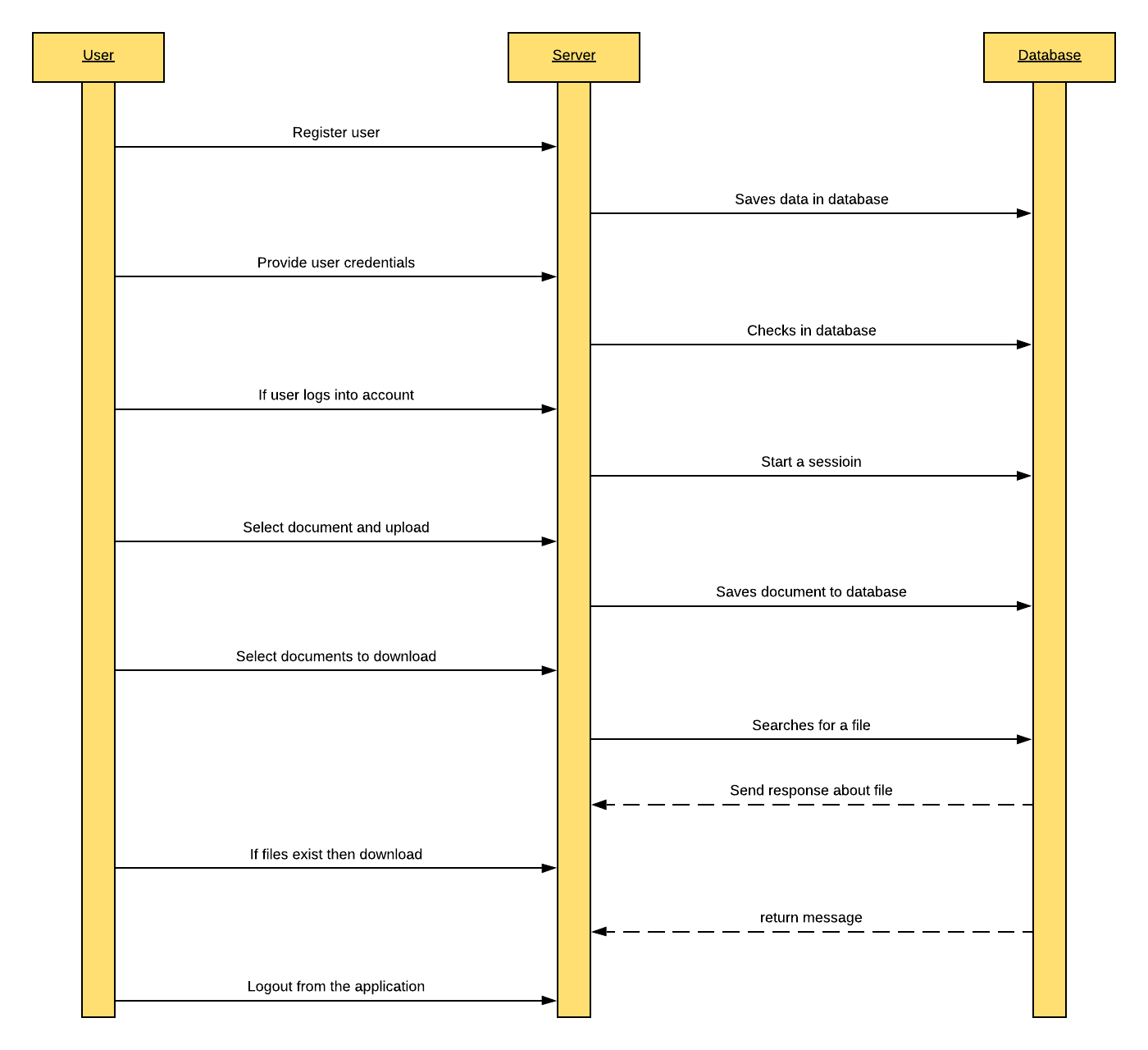
**Flow –Chart:**



**Sequence Diagram**

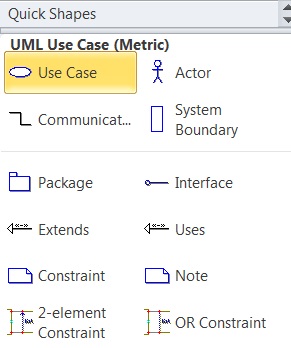
A sequence diagram describes interactions among classes in terms of an “Exchange of message over time”. Sequence diagrams are used to depict the time sequence of message exchanged between objects. Message can correspond to operation on class or an event trigger. Notations of a Sequence Diagram include:

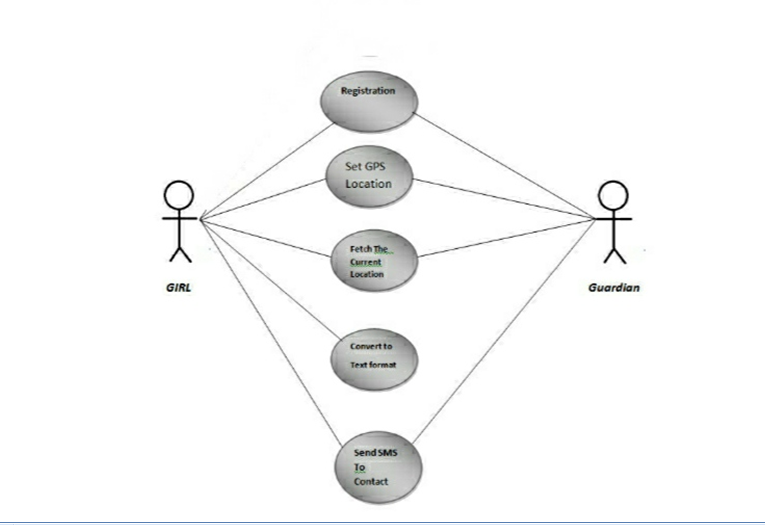
* Lifeline: It is a vertical dashed line that represents the “lifetime” of an object.
* Arrows: They indicate flow of message between objects.
* Activation: It is a thin rectangle showing period of time, during which an object is performing an action.



**Use Case Diagram**

A use case diagram in the unified modeling language is a type of behavioral diagram defined by and created from a Use-Case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.



****

**Deployment Diagram**

Deployment diagram is a structure diagram which shows architecture of the system as deployment (distribution) of software artifacts to deployment targets.

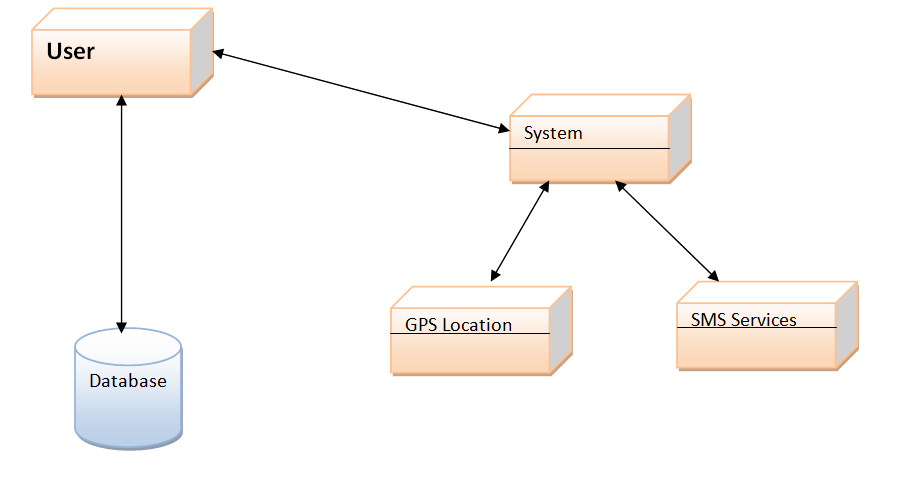
Artifacts represent concrete elements in the physical world that are the result of a development process. Deployment Diagram is usually represented by a node which is either hardware device or some software execution environment. Nodes could be connected through communication paths to create networked systems of arbitrary complexity.

Note, that component was directly deployed to nodes in UML 1.x deployment diagrams. In UML 2.x artifacts are deployed to nodes, and artifacts could manifest (implement) components. Components are deployed to nodes indirectly through artifacts.

Deployment diagrams could describe architecture at specification level (also called type level) or at instance level (similar to class diagrams and object diagrams).

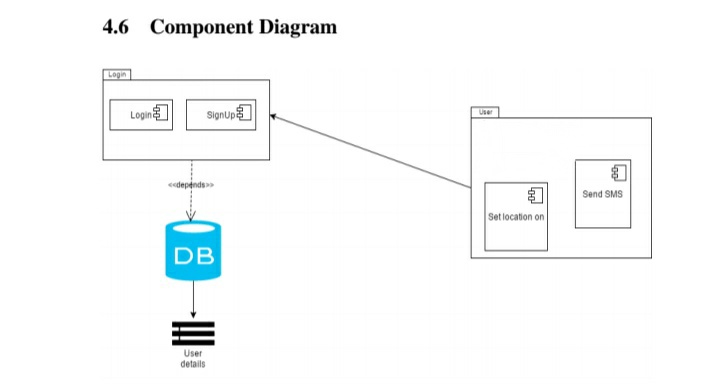
**Specification level** deployment diagram shows some overview of deployment of artifacts to deployment targets, without referencing specific instances of artifacts or nodes.

**Instance level** deployment diagram shows deployment of instances of artifacts to specific instances of deployment targets. It could be used for example to show differences in deployments to development, staging or production .



*Figure : Deployement Diagram*

Component Diagram**:**

****

**Table Design for local storage.**

* **Table for storing Profile of user :**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr.No | Column Name | Datatype | Description |
| 1 | First Name | String | It stores First Name of user |
| 2 | Last Name | String | It stores Last Name of user |
| 3 | Email | String | It stores Email of user |
| 4 | Phone Number | String | It stores Phone Number of user |

* **Table for storing predefind raw data :**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr.No | Column Name | extension | Description |
| 1 | outgoingtone | Support .mp3 | It stores outgoingtone of user device |
| 2 | Policesiren | Support .mp3 | It stores Policesiren for user |

**System Coding & Conventions**

**Program Source Code**

**AndroidManifest.xml**

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

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

package="com.example.akhilesh.oneplussafety">

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

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

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

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

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

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

<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE" />

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

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

<uses-permission android:name="android.permission.CALL\_PHONE" />

<uses-feature

android:name="android.hardware.microphone"

android:required="false" />

<uses-feature

android:name="android.hardware.telephony"

android:required="false" />

<uses-feature android:name="android.hardware.camera" />

<uses-feature

android:name="android.hardware.location"

android:required="true" />

<uses-feature

android:name="android.hardware.location.gps"

android:required="false" />

<application

android:allowBackup="true"

android:icon="@mipmap/one\_plus"

android:label="One Plus Safety"

android:theme="@style/AppTheme">

<activity

android:name="Splash"

android:screenOrientation="portrait">

<intent-filter>

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

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

</intent-filter>

</activity>

<activity

android:name="MainActivity"

android:screenOrientation="portrait" />

<activity

android:name="PeopleSecond"

android:screenOrientation="portrait" />

<receiver android:name="VolumeChangeReceiver">

<intent-filter>

<action android:name="android.media.VOLUME\_CHANGED\_ACTION" />

</intent-filter>

</receiver>

<activity

android:name="People"

android:screenOrientation="portrait" />

<activity

android:name="Help"

android:screenOrientation="portrait"></activity>

<activity

android:name=".NavigationAlert"

android:label="@string/title\_activity\_navigation\_alert"

android:screenOrientation="portrait"

android:theme="@style/AppTheme.NoActionBar" />

<meta-data

android:name="com.google.android.geo.API\_KEY"

android:value="@string/google\_maps\_key" />

<activity

android:name="NavigationComplaint"

android:screenOrientation="portrait"></activity>

<activity

android:name="TabBars"

android:screenOrientation="portrait"></activity>

<activity

android:name="AboutUs"

android:screenOrientation="portrait"></activity>

<activity

android:name="RecievedCall"

android:screenOrientation="portrait"></activity>

<activity

android:name="VoiceCallActivity"

android:screenOrientation="portrait"></activity>

<receiver android:name="MyBroadcastReceiver"></receiver>

<receiver android:name="CallBReciever">

<intent-filter>

<action android:name="call"></action>

</intent-filter>

</receiver>

</application>

</manifest>

**AboutUs.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.ImageButton;

import android.widget.TextView;

import com.example.akhilesh.oneplussafety.R;

public class AboutUs extends AppCompatActivity {

ImageButton backButton;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.about\_us);

TextView tv = (TextView) findViewById(R.id.toolbar\_title);

tv.setText(R.string.about);

}

public void back(View v) {

Intent i = new Intent(getApplicationContext(), NavigationAlert.class);

startActivity(i);

}

public void call(View v) {

Intent intent = new Intent(Intent.ACTION\_DIAL, Uri.parse("tel:" + "9136197019"));

startActivity(intent);

}

}

**CallBReciever.java**

package com.example.akhilesh.oneplussafety;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import com.example.akhilesh.oneplussafety.RecievedCall;

public class CallBReciever extends BroadcastReceiver {

@Override

public void onReceive(Context context, Intent intent) {

Intent i = new Intent(context, RecievedCall.class);

context.startActivity(i);

}

}

**CorruptionFragment.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v4.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import com.example.akhilesh.oneplussafety.R;

public class CorruptionFragment extends Fragment {

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

View V = inflater.inflate(R.layout.corruption, container, false);

Button button = (Button) V.findViewById(R.id.Button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent = null, chooser = null;

intent = new Intent(Intent.ACTION\_SEND);

intent.setData(Uri.parse("mailto:"));

String[] to = {"ay925386@gmail.com"};

intent.putExtra(Intent.EXTRA\_EMAIL, to);

intent.putExtra(Intent.EXTRA\_SUBJECT, "Regarding Corruption");

intent.setType("message/rfc822");

chooser = Intent.createChooser(intent, "Send Email");

startActivity(chooser);

}

});

return V;

}

}

**DomesticViolenceFragment.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v4.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import com.example.akhilesh.oneplussafety.R;

public class DomesticViolenceFragment extends Fragment {

Button button;

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

View V= inflater.inflate(R.layout.domestic\_violence,container,false);

button= (Button) V.findViewById(R.id.Button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent=null, chooser=null;

intent=new Intent(Intent.ACTION\_SEND);

intent.setData(Uri.parse("mailto:"));

String[] to={"ay925386@gmail.com"};

intent.putExtra(Intent.EXTRA\_EMAIL,to);

intent.putExtra(Intent.EXTRA\_SUBJECT,"Regarding Domestic Violence");

intent.setType("message/rfc822");

chooser=Intent.createChooser(intent,"Send Email");

startActivity(chooser);

}

});

return V;

}

}

**EveTeasingFragment.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v4.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import com.example.akhilesh.oneplussafety.R;

public class EveTeasingFragment extends Fragment {

Button button;

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

View V= inflater.inflate(R.layout.eve\_teasing,container,false);

button= (Button) V.findViewById(R.id.Button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent=null, chooser=null;

intent=new Intent(Intent.ACTION\_SEND);

intent.setData(Uri.parse("mailto:"));

String[] to={"ay925386@gmail.com"};

intent.putExtra(Intent.EXTRA\_EMAIL,to);

intent.putExtra(Intent.EXTRA\_SUBJECT,"Regarding Eve Teasing");

intent.setType("message/rfc822");

chooser=Intent.createChooser(intent,"Send Email");

startActivity(chooser);

}

});

return V;

}

}

**ExpandableListAdapter.java**

package com.example.akhilesh.oneplussafety;

import android.content.Context;

import android.graphics.Typeface;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseExpandableListAdapter;

import android.widget.TextView;

import com.example.akhilesh.oneplussafety.R;

import java.util.HashMap;

import java.util.List;

public class ExpandableListAdapter extends BaseExpandableListAdapter {

private Context \_context;

private List<String> \_listDataHeader; // header titles

// child data in format of header title, child title

private HashMap<String, List<String>> \_listDataChild;

public ExpandableListAdapter(Context context, List<String> listDataHeader,

HashMap<String, List<String>> listChildData) {

this.\_context = context;

this.\_listDataHeader = listDataHeader;

this.\_listDataChild = listChildData;

}

@Override

public Object getChild(int groupPosition, int childPosititon) {

return this.\_listDataChild.get(this.\_listDataHeader.get(groupPosition))

.get(childPosititon);

}

@Override

public long getChildId(int groupPosition, int childPosition) {

return childPosition;

}

@Override

public View getChildView(int groupPosition, final int childPosition,

boolean isLastChild, View convertView, ViewGroup parent) {

final String childText = (String) getChild(groupPosition, childPosition);

if (convertView == null) {

LayoutInflater infalInflater = (LayoutInflater) this.\_context

.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);

convertView = infalInflater.inflate(R.layout.list\_item, null);

}

TextView txtListChild = (TextView) convertView

.findViewById(R.id.lblListItem);

txtListChild.setText(childText);

return convertView;

}

@Override

public int getChildrenCount(int groupPosition) {

return this.\_listDataChild.get(this.\_listDataHeader.get(groupPosition))

.size();

}

@Override

public Object getGroup(int groupPosition) {

return this.\_listDataHeader.get(groupPosition);

}

@Override

public int getGroupCount() {

return this.\_listDataHeader.size();

}

@Override

public long getGroupId(int groupPosition) {

return groupPosition;

}

@Override

public View getGroupView(int groupPosition, boolean isExpanded,

View convertView, ViewGroup parent) {

String headerTitle = (String) getGroup(groupPosition);

if (convertView == null) {

LayoutInflater infalInflater = (LayoutInflater) this.\_context

.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);

convertView = infalInflater.inflate(R.layout.list\_group, null);

}

TextView lblListHeader = (TextView) convertView

.findViewById(R.id.lblListHeader);

lblListHeader.setTypeface(null, Typeface.BOLD);

lblListHeader.setText(headerTitle);

return convertView;

}

@Override

public boolean hasStableIds() {

return false;

}

@Override

public boolean isChildSelectable(int groupPosition, int childPosition) {

return true;

}

}

**FragHome.java**

package com.example.akhilesh.oneplussafety;

import android.app.Fragment;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import com.example.akhilesh.oneplussafety.R;

public class FragHome extends Fragment {

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

return inflater.inflate(R.layout.frag\_home,container,false);

}

}

**GalleryView.java**

package com.example.akhilesh.oneplussafety;

import android.app.Fragment;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import com.example.akhilesh.oneplussafety.R;

public class GalleryView extends Fragment {

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

return inflater.inflate(R.layout.content\_navigation\_complaint,null);

}

}

**GPSTracker.java**

package com.example.akhilesh.oneplussafety;

import android.Manifest;

import android.app.AlertDialog;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.os.Bundle;

import android.provider.Settings;

import android.support.v4.app.ActivityCompat;

import android.util.Log;

public final class GPSTracker implements LocationListener {

private final Context mContext;

// flag for GPS status

public boolean isGPSEnabled = false;

// flag for network status

public boolean isNetworkEnabled = false;

// flag for GPS status

public boolean canGetLocation = false;

Location location; // location

double latitude; // latitude

double longitude; // longitude

// The minimum distance to change Updates in meters

private static final long MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES = 1; // 10 meters

// The minimum time between updates in milliseconds

private static final long MIN\_TIME\_BW\_UPDATES = 1; // 1 minute

// Declaring a Location Manager

protected LocationManager locationManager;

public GPSTracker(Context context) {

this.mContext = context;

getLocation();

}

/\*\*

\* Function to get the user's current location

\*

\* @return

\*/

public Location getLocation() {

try {

locationManager = (LocationManager) mContext

.getSystemService(Context.LOCATION\_SERVICE);

// getting GPS status

isGPSEnabled = locationManager

.isProviderEnabled(LocationManager.GPS\_PROVIDER);

Log.v("isGPSEnabled", "=" + isGPSEnabled);

// getting network status

isNetworkEnabled = locationManager

.isProviderEnabled(LocationManager.NETWORK\_PROVIDER);

Log.v("isNetworkEnabled", "=" + isNetworkEnabled);

if (isGPSEnabled == false && isNetworkEnabled == false) {

// no network provider is enabled

} else {

this.canGetLocation = true;

if (isNetworkEnabled) {

location = null;

if (ActivityCompat.checkSelfPermission(mContext, Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(mContext, Manifest.permission.ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return getLocation();

}

locationManager.requestLocationUpdates(

LocationManager.NETWORK\_PROVIDER,

MIN\_TIME\_BW\_UPDATES,

MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES, this);

Log.d("Network", "Network");

if (locationManager != null) {

location = locationManager

.getLastKnownLocation(LocationManager.NETWORK\_PROVIDER);

if (location != null) {

latitude = location.getLatitude();

longitude = location.getLongitude();

}

}

}

// if GPS Enabled get lat/long using GPS Services

if (isGPSEnabled) {

location = null;

if (location == null) {

locationManager.requestLocationUpdates(

LocationManager.GPS\_PROVIDER,

MIN\_TIME\_BW\_UPDATES,

MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES, this);

Log.d("GPS Enabled", "GPS Enabled");

if (locationManager != null) {

location = locationManager

.getLastKnownLocation(LocationManager.GPS\_PROVIDER);

if (location != null) {

latitude = location.getLatitude();

longitude = location.getLongitude();

}

}

}

}

}

} catch (Exception e) {

e.printStackTrace();

}

return location;

}

/\*\*

\* Stop using GPS listener Calling this function will stop using GPS in your

\* app

\* \*/

public void stopUsingGPS() {

if (locationManager != null) {

if ((ActivityCompat.checkSelfPermission(mContext, Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) && (ActivityCompat.checkSelfPermission(mContext, Manifest.permission.ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED)) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

locationManager.removeUpdates(GPSTracker.this);

}

}

/\*\*

\* Function to get latitude

\* \*/

public double getLatitude() {

if (location != null) {

latitude = location.getLatitude();

}

// return latitude

return latitude;

}

/\*\*

\* Function to get longitude

\* \*/

public double getLongitude() {

if (location != null) {

longitude = location.getLongitude();

}

// return longitude

return longitude;

}

/\*\*

\* Function to check GPS/wifi enabled

\*

\* @return boolean

\* \*/

public boolean canGetLocation() {

return this.canGetLocation;

}

/\*\*

\* Function to show settings alert dialog On pressing Settings button will

\* lauch Settings Options

\* \*/

public void showSettingsAlert() {

AlertDialog.Builder alertDialog = new AlertDialog.Builder(mContext);

// Setting Dialog Title

alertDialog.setTitle("GPS is settings");

// Setting Dialog Message

alertDialog

.setMessage("GPS is not enabled. Do you want to go to settings menu?");

// On pressing Settings button

alertDialog.setPositiveButton("Settings",

new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

Intent intent = new Intent(

Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS);

mContext.startActivity(intent);

}

});

// on pressing cancel button

alertDialog.setNegativeButton("Cancel",

new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

dialog.cancel();

}

});

// Showing Alert Message

alertDialog.show();

}

@Override

public void onLocationChanged(Location location) {

}

@Override

public void onProviderDisabled(String provider) {

}

@Override

public void onProviderEnabled(String provider) {

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

}

}

**Help.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.os.Bundle;

import android.support.v7.app.ActionBar;

import android.support.v7.app.AppCompatActivity;

import android.support.v7.widget.Toolbar;

import android.view.View;

import android.widget.ExpandableListView;

import android.widget.ExpandableListView.OnGroupClickListener;

import android.widget.ImageButton;

import android.widget.TextView;

import com.example.akhilesh.oneplussafety.R;

import com.example.akhilesh.oneplussafety.ExpandableListAdapter;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

public class Help extends AppCompatActivity {

ExpandableListAdapter listAdapter;

ExpandableListView expListView;

List<String> listDataHeader;

HashMap<String, List<String>> listDataChild;

ImageButton backButton;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.help);

TextView tv= (TextView) findViewById(R.id.toolbar\_title);

tv.setText(R.string.help);

// get the listview

expListView = (ExpandableListView) findViewById(R.id.lvExp);

// preparing list data

prepareListData();

listAdapter = new ExpandableListAdapter(this, listDataHeader, listDataChild);

// setting list adapter

expListView.setAdapter(listAdapter);

// Listview Group click listener

expListView.setOnGroupClickListener(new OnGroupClickListener() {

@Override

public boolean onGroupClick(ExpandableListView parent, View v,

int groupPosition, long id) {

// Toast.makeText(getApplicationContext(),

// "Group Clicked " + listDataHeader.get(groupPosition),

// Toast.LENGTH\_SHORT).show();

return false;

}

});

// Listview Group expanded listener

// expListView.setOnGroupExpandListener(new OnGroupExpandListener() {

//

// @Override

// public void onGroupExpand(int groupPosition) {

// Toast.makeText(getApplicationContext(),

// listDataHeader.get(groupPosition) + " Expanded",

// Toast.LENGTH\_SHORT).show();

// }

// });

// Listview Group collasped listener

// expListView.setOnGroupCollapseListener(new OnGroupCollapseListener() {

//

// @Override

// public void onGroupCollapse(int groupPosition) {

// Toast.makeText(getApplicationContext(),

// listDataHeader.get(groupPosition) + " Collapsed",

// Toast.LENGTH\_SHORT).show();

//

// }

// });

// Listview on child click listener

// expListView.setOnChildClickListener(new OnChildClickListener() {

//

// @Override

// public boolean onChildClick(ExpandableListView parent, View v,

// int groupPosition, int childPosition, long id) {

// // TODO Auto-generated method stub

// Toast.makeText(

// getApplicationContext(),

// listDataHeader.get(groupPosition)

// + " : "

// + listDataChild.get(

// listDataHeader.get(groupPosition)).get(

// childPosition), Toast.LENGTH\_SHORT)

// .show();

// return false;

// }

// });

}

/\*

\* Preparing the list data

\*/

private void prepareListData() {

listDataHeader = new ArrayList<String>();

listDataChild = new HashMap<String, List<String>>();

// Adding child data

listDataHeader.add("Ask for Help");

listDataHeader.add("Privacy and Safety Center");

listDataHeader.add("Information we collect");

listDataHeader.add("Cookies and other technologies");

listDataHeader.add("How to contact us");

// Adding child data

List<String> ask = new ArrayList<String>();

ask.add("When you are feeling unsafe,quickly press the power button thrice and inform your Alert Contacts with information like current location and predefined messages about your situation.");

// top250.add("The Godfather");

// top250.add("The Godfather: Part II");

// top250.add("Pulp Fiction");

// top250.add("The Good, the Bad and the Ugly");

// top250.add("The Dark Knight");

// top250.add("12 Angry Men");

List<String> privacy = new ArrayList<String>();

privacy.add("By using our Service you understand and agree that we are providing you a platform to reach out to police or your well-wishers when in danger with just 3 clicks.\n" +

" Your current location would be sent with the message.\n" +

" This means that the emergency contacts would be able to know you present location.\n" +

" Our policy applies to all visitors,users and others who access the Service");

// nowShowing.add("Despicable Me 2");

// nowShowing.add("Turbo");

// nowShowing.add("Grown Ups 2");

// nowShowing.add("Red 2");

// nowShowing.add("The Wolverine");

List<String> info = new ArrayList<String>();

info.add("Information you provide us:\n" +

" -> Your location\n" +

" -> Emergency Contacts\n" +

" -> Messages sent ");

// comingSoon.add("The Smurfs 2");

// comingSoon.add("The Spectacular Now");

// comingSoon.add("The Canyons");

// comingSoon.add("Europa Report");

List<String> cookie = new ArrayList<String>();

cookie.add(" -> When you visit the Service, we may use cookies and similar technologies like pixels,web becons and local storage to collect information.\n" +

" -> We may ask advertisers or other partners to serve ads or services to your devices,which may use cookies and similar technologies. ");

List<String> contact = new ArrayList<String>();

contact.add(" If you have any questions about Privacy Policy or the Service, please find appropriate support channel in Help Center at which to contact us.");

listDataChild.put(listDataHeader.get(0), ask); // Header, Child data

listDataChild.put(listDataHeader.get(1), privacy);

listDataChild.put(listDataHeader.get(2), info);

listDataChild.put(listDataHeader.get(3), cookie);

listDataChild.put(listDataHeader.get(4), contact);

}

public void back(View v) {

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

startActivity(i);

}

}

**MainActivity.java**

package com.example.akhilesh.oneplussafety;

import android.annotation.TargetApi;

import android.app.AlarmManager;

import android.app.PendingIntent;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.location.Location;

import android.os.Build;

import android.os.Bundle;

import android.support.annotation.NonNull;

import android.support.annotation.Nullable;

import android.support.v4.app.ActivityCompat;

import android.support.v7.app.AlertDialog;

import android.support.v7.app.AppCompatActivity;

import android.telephony.SmsManager;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.LinearLayout;

import android.widget.TextView;

import android.widget.Toast;

import com.example.akhilesh.oneplussafety.R;

import com.example.akhilesh.oneplussafety.MyBroadcastReceiver;

import com.example.akhilesh.oneplussafety.GPSTracker;

import com.example.akhilesh.oneplussafety.SessionManager;

import com.google.android.gms.common.ConnectionResult;

import com.google.android.gms.common.GoogleApiAvailability;

import com.google.android.gms.common.api.GoogleApiClient;

import com.google.android.gms.location.LocationListener;

import com.google.android.gms.location.LocationRequest;

import com.google.android.gms.location.LocationServices;

import java.util.ArrayList;

import java.util.HashMap;

import static android.Manifest.permission.ACCESS\_COARSE\_LOCATION;

import static android.Manifest.permission.ACCESS\_FINE\_LOCATION;

public class MainActivity extends AppCompatActivity implements GoogleApiClient.ConnectionCallbacks,

GoogleApiClient.OnConnectionFailedListener, LocationListener {

Location mLocation;

GoogleApiClient mGoogleApiClient;

private static final int PLAY\_SERVICES\_RESOLUTION\_REQUEST = 9000;

private LocationRequest mLocationRequest;

private long UPDATE\_INTERVAL = 15000; /\* 15 secs \*/

private long FASTEST\_INTERVAL = 5000; /\* 5 secs \*/

private ArrayList<String> permissionsToRequest;

private ArrayList<String> permissionsRejected = new ArrayList<>();

private ArrayList<String> permissions = new ArrayList<>();

private final static int ALL\_PERMISSIONS\_RESULT = 101;

SessionManager session;

private LinearLayout alarm, profile, fake\_call;

private SmsManager sm;

Button b1;

String location = "";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

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

alarm = (LinearLayout) findViewById(R.id.alarm);

profile = (LinearLayout) findViewById(R.id.profile);

fake\_call = (LinearLayout) findViewById(R.id.fake\_call);

alarm.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent = new Intent(MainActivity.this, MyBroadcastReceiver.class);

PendingIntent pendingIntent = PendingIntent.getBroadcast(getApplicationContext(), 232, intent, 0);

AlarmManager alarmManager = (AlarmManager) getSystemService(ALARM\_SERVICE);

alarmManager.set(AlarmManager.RTC\_WAKEUP, System.currentTimeMillis() + (5000), pendingIntent);

Toast.makeText(MainActivity.this, "Alarm will start with in five seconds", Toast.LENGTH\_SHORT).show();

}

});

fake\_call.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent = new Intent(MainActivity.this, CallBReciever.class);

intent.setAction("call");

sendBroadcast(intent);

}

});

profile.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent i = new Intent(MainActivity.this, NavigationAlert.class);

startActivity(i);

}

});

permissions.add(ACCESS\_FINE\_LOCATION);

permissions.add(ACCESS\_COARSE\_LOCATION);

permissionsToRequest = findUnAskedPermissions(permissions);

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.M) {

if (permissionsToRequest.size() > 0)

requestPermissions(permissionsToRequest.toArray(new String[permissionsToRequest.size()]), ALL\_PERMISSIONS\_RESULT);

}

mGoogleApiClient = new GoogleApiClient.Builder(this)

.addApi(LocationServices.API)

.addConnectionCallbacks(this)

.addOnConnectionFailedListener(this)

.build();

session = new SessionManager(getApplicationContext());

}

public void Complaint(View v) {

Intent i = new Intent(this, NavigationComplaint.class);

startActivity(i);

}

public void Emer(View v) {

HashMap<String, String> user = session.getUserDetails();

String number1 = user.get(SessionManager.CONTACT1);

// name

String number2 = user.get(SessionManager.CONTACT2);

// email

String number3 = user.get(SessionManager.CONTACT3);

sm = SmsManager.getDefault();

if (number1 !=null) {

sm.sendTextMessage(number1, null, "I'm in danger..My current location is http://maps.google.com/?q=" + location, null, null);

sm.sendTextMessage(number2, null, "I'm in danger..My current location is http://maps.google.com/?q=" + location, null, null);

sm.sendTextMessage(number3, null, "I'm in danger..My current location is http://maps.google.com/?q=" + location, null, null);

Toast.makeText(getApplicationContext(),"message sent",Toast.LENGTH\_SHORT).show();

} else {

Intent i = new Intent(this, People.class);

startActivity(i);

}

}

private ArrayList<String> findUnAskedPermissions(ArrayList<String> wanted) {

ArrayList<String> result = new ArrayList<String>();

for (String perm : wanted) {

if (!hasPermission(perm)) {

result.add(perm);

}

}

return result;

}

@Override

protected void onStart() {

super.onStart();

if (mGoogleApiClient != null) {

mGoogleApiClient.connect();

}

}

@Override

protected void onResume() {

super.onResume();

if (!checkPlayServices()) {

location ="Please install Google Play services.";}

}

@Override

public void onConnected(@Nullable Bundle bundle) {

if (ActivityCompat.checkSelfPermission(this, ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(this, ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

mLocation = LocationServices.FusedLocationApi.getLastLocation(mGoogleApiClient);

if(mLocation!=null)

{

location= mLocation.getLatitude()+","+mLocation.getLongitude();

}

startLocationUpdates();

}

@Override

public void onConnectionSuspended(int i) {

}

@Override

public void onConnectionFailed(@NonNull ConnectionResult connectionResult) {

}

@Override

public void onLocationChanged(Location location1) {

if(location1!=null)

location= mLocation.getLatitude()+","+mLocation.getLongitude();

}

private boolean checkPlayServices() {

GoogleApiAvailability apiAvailability = GoogleApiAvailability.getInstance();

int resultCode = apiAvailability.isGooglePlayServicesAvailable(this);

if (resultCode != ConnectionResult.SUCCESS) {

if (apiAvailability.isUserResolvableError(resultCode)) {

apiAvailability.getErrorDialog(this, resultCode, PLAY\_SERVICES\_RESOLUTION\_REQUEST)

.show();

} else

finish();

return false;

}

return true;

}

protected void startLocationUpdates() {

mLocationRequest = new LocationRequest();

mLocationRequest.setPriority(LocationRequest.PRIORITY\_HIGH\_ACCURACY);

mLocationRequest.setInterval(UPDATE\_INTERVAL);

mLocationRequest.setFastestInterval(FASTEST\_INTERVAL);

if (ActivityCompat.checkSelfPermission(this, ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(this, ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

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

}

LocationServices.FusedLocationApi.requestLocationUpdates(

mGoogleApiClient, mLocationRequest, this);

}

private boolean hasPermission(String permission) {

if (canMakeSmores()) {

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.M) {

return (checkSelfPermission(permission) == PackageManager.PERMISSION\_GRANTED);

}

}

return true;

}

private boolean canMakeSmores() {

return (Build.VERSION.SDK\_INT > Build.VERSION\_CODES.LOLLIPOP\_MR1);

}

@TargetApi(Build.VERSION\_CODES.M)

@Override

public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {

switch (requestCode) {

case ALL\_PERMISSIONS\_RESULT:

for (String perms : permissionsToRequest) {

if (!hasPermission(perms)) {

permissionsRejected.add(perms);

}

}

if (permissionsRejected.size() > 0) {

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.M) {

if (shouldShowRequestPermissionRationale(permissionsRejected.get(0))) {

showMessageOKCancel("These permissions are mandatory for the application. Please allow access.",

new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.M) {

requestPermissions(permissionsRejected.toArray(new String[permissionsRejected.size()]), ALL\_PERMISSIONS\_RESULT);

}

}

});

return;

}

}

}

break;

}

}

private void showMessageOKCancel(String message, DialogInterface.OnClickListener okListener) {

new AlertDialog.Builder(MainActivity.this)

.setMessage(message)

.setPositiveButton("OK", okListener)

.setNegativeButton("Cancel", null)

.create()

.show();

}

@Override

protected void onDestroy() {

super.onDestroy();

stopLocationUpdates();

}

public void stopLocationUpdates()

{

if (mGoogleApiClient.isConnected()) {

LocationServices.FusedLocationApi

.removeLocationUpdates(mGoogleApiClient, this);

mGoogleApiClient.disconnect();

}

}

}

**MyBroadcastReceiver.java**

package com.example.akhilesh.oneplussafety;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.media.MediaPlayer;

import android.widget.Toast;

import com.example.akhilesh.oneplussafety.R;

public class MyBroadcastReceiver extends BroadcastReceiver {

MediaPlayer mediaPlayer;

@Override

public void onReceive(Context context, Intent intent) {

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

mediaPlayer.start();

Toast.makeText(context,"Alarm..",Toast.LENGTH\_SHORT).show();

}

}

**NavigationAlert.java**

package com.example.akhilesh.oneplussafety;

import android.app.Fragment;

import android.app.FragmentTransaction;

import android.content.Intent;

import android.os.Bundle;

import android.support.design.widget.NavigationView;

import android.support.v4.view.GravityCompat;

import android.support.v4.widget.DrawerLayout;

import android.support.v7.app.ActionBarDrawerToggle;

import android.support.v7.app.AppCompatActivity;

import android.support.v7.widget.Toolbar;

import android.view.MenuItem;

public class NavigationAlert extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener {

SessionManager session;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_navigation\_alert);

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);

setSupportActionBar(toolbar);

session = new SessionManager(getApplicationContext());

DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer\_layout);

ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(

this, drawer, toolbar, R.string.navigation\_drawer\_open, R.string.navigation\_drawer\_close);

drawer.setDrawerListener(toggle);

toggle.syncState();

NavigationView navigationView = (NavigationView) findViewById(R.id.nav\_view);

navigationView.setNavigationItemSelectedListener(this);

}

@Override

public void onBackPressed() {

DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer\_layout);

if (drawer.isDrawerOpen(GravityCompat.START)) {

drawer.closeDrawer(GravityCompat.START);

} else {

super.onBackPressed();

}

}

@SuppressWarnings("StatementWithEmptyBody")

@Override

public boolean onNavigationItemSelected(MenuItem item) {

// Handle navigation view item clicks here.

int id = item.getItemId();

Fragment fragment = null;

if (id == R.id.Home) {

Intent i = new Intent(this, MainActivity.class);

startActivity(i);

// Handle the camera action

} else if (id == R.id.People) {

// session.checkLogin();

if (session.isLoggedIn()) {

Intent i = new Intent(this, PeopleSecond.class);

startActivity(i);

} else {

Intent i = new Intent(this, People.class);

startActivity(i);

}

} else if (id == R.id.AboutUs) {

Intent i = new Intent(this, AboutUs.class);

startActivity(i);

} else if (id == R.id.Help) {

Intent i = new Intent(this, Help.class);

startActivity(i);

}

if (fragment != null) {

android.app.FragmentManager fragmentManager = getFragmentManager();

FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction().replace(R.id.content\_frame, fragment);

fragmentTransaction.addToBackStack("MainActivity");

fragmentTransaction.commit();

}

DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer\_layout);

drawer.closeDrawer(GravityCompat.START);

return true;

}

}

**NavigationComplaint.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.os.Handler;

import android.support.design.widget.NavigationView;

import android.support.v4.widget.DrawerLayout;

import android.support.v7.app.ActionBarDrawerToggle;

import android.support.v7.app.AppCompatActivity;

import android.view.MenuItem;

import android.view.animation.Animation;

import android.view.animation.AnimationUtils;

import android.widget.ImageView;

import com.example.akhilesh.oneplussafety.R;

import java.util.Timer;

import java.util.TimerTask;

public class NavigationComplaint extends AppCompatActivity{

DrawerLayout mDrawerLayout;

NavigationView mNavigationView;

//FragmentManager mFragmentManager;

//FragmentTransaction mFragmentTransaction;

public int currentimageindex=0;

// Timer timer;

// TimerTask task;

ImageView slidingimage;

private int[] IMAGE\_IDS = {

R.drawable.domesticviolence1, R.drawable.corruption1, R.drawable.theft1,

R.drawable.threat1, R.drawable.eveteasing1

};

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.navigation\_complaint);

final Handler mHandler = new Handler();

// Create runnable for posting

final Runnable mUpdateResults = new Runnable() {

public void run() {

AnimateandSlideShow();

}

};

int delay = 1000; // delay for 1 sec.

int period = 5000; // repeat every 4 sec.

Timer timer = new Timer();

timer.scheduleAtFixedRate(new TimerTask() {

public void run() {

mHandler.post(mUpdateResults);

}

}, delay, period);

/\*\*

\*Setup the DrawerLayout and NavigationView

\*/

mDrawerLayout = (DrawerLayout) findViewById(R.id.drawerLayout);

mNavigationView = (NavigationView) findViewById(R.id.shitstuff) ;

/\*\*

\* Lets inflate the very first fragment

\* Here , we are inflating the TabFragment as the first Fragment

\*/

// mFragmentManager = getSupportFragmentManager();

// mFragmentTransaction = mFragmentManager.beginTransaction();

// mFragmentTransaction.replace(R.id.content\_frame,new TabFragment()).commit();

// /\*\*

// \* Setup click events on the Navigation View Items.

// \*/

mNavigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener() {

@Override

public boolean onNavigationItemSelected(MenuItem menuItem) {

mDrawerLayout.closeDrawers();

if (menuItem.getItemId() == R.id.Home) {

Intent intent=new Intent(NavigationComplaint.this,MainActivity.class);

startActivity(intent);

}

if (menuItem.getItemId() == R.id.mail) {

// FragmentTransaction xfragmentTransaction = mFragmentManager.beginTransaction();

// xfragmentTransaction.replace(R.id.containerView,new TabFragment()).commit();

//

Intent intent=null, chooser=null;

intent=new Intent(Intent.ACTION\_SEND);

intent.setData(Uri.parse("mailto:"));

String[] to={"ishumehta99@gmail.com"};

intent.putExtra(Intent.EXTRA\_EMAIL,to);

intent.setType("message/rfc822");

chooser=Intent.createChooser(intent,"Send Email");

startActivity(chooser);

}

if (menuItem.getItemId()== R.id.complaints){

Intent i=new Intent(NavigationComplaint.this,TabBars.class);

startActivity(i);

}

if (menuItem.getItemId()== R.id.Help){

Intent i=new Intent(NavigationComplaint.this,Help.class);

startActivity(i);

}

if (menuItem.getItemId()== R.id.AboutUs){

Intent i=new Intent(NavigationComplaint.this,AboutUs.class);

startActivity(i);

}

return false;

}

});

/\*\*

\* Setup Drawer Toggle of the Toolbar

\*/

android.support.v7.widget.Toolbar toolbar = (android.support.v7.widget.Toolbar) findViewById(R.id.toolbar);

ActionBarDrawerToggle mDrawerToggle = new ActionBarDrawerToggle(this,mDrawerLayout, toolbar,R.string.app\_name,

R.string.app\_name);

mDrawerLayout.setDrawerListener(mDrawerToggle);

mDrawerToggle.syncState();

}

/\*\*

\* Helper method to start the animation on the splash screen

\*/

private void AnimateandSlideShow() {

slidingimage = (ImageView)findViewById(R.id.ImageView3\_Left);

slidingimage.setImageResource(IMAGE\_IDS[currentimageindex%IMAGE\_IDS.length]);

currentimageindex++;

Animation rotateimage = AnimationUtils.loadAnimation(this, android.R.anim.fade\_in);

slidingimage.startAnimation(rotateimage);

}

}

**People.java**

package com.example.akhilesh.oneplussafety;

import android.Manifest;

import android.content.Context;

import android.content.Intent;

import android.content.SharedPreferences;

import android.content.pm.PackageManager;

import android.database.Cursor;

import android.net.Uri;

import android.os.Build;

import android.os.Bundle;

import android.provider.ContactsContract;

import android.support.annotation.NonNull;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ImageButton;

import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

import com.example.akhilesh.oneplussafety.R;

import com.example.akhilesh.oneplussafety.SessionManager;

public class People extends AppCompatActivity {

EditText editText1, editText2, editText3;

ImageView phoneBook1, phoneBook2, phoneBook3;

ImageButton backButton;

int flow = 0;

public static int getContact = 2;

int globalContact = 0;

boolean saveContactd = false;

// Session Manager Class

SessionManager session;

// Button Logout

Button btnNxt;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.people);

backButton = (ImageButton) findViewById(R.id.back);

if (Build.VERSION.SDK\_INT < 23) {

} else

// for alove L version we need to get permission at the door to get throught that location ...

if (checkSelfPermission(Manifest.permission.READ\_CONTACTS) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.SEND\_SMS) != PackageManager.PERMISSION\_GRANTED) { // checkSelfPermission is a method avail in 23 api ..without if condition of (SDK\_INT < 23 ) you cant implement it..

requestPermissions(new String[]{Manifest.permission.READ\_CONTACTS, Manifest.permission.ACCESS\_FINE\_LOCATION, Manifest.permission.SEND\_SMS}, getContact);

} else {

}

TextView tv = (TextView) findViewById(R.id.toolbar\_title);

tv.setText(R.string.contacts);

//session manager

// Session class instance

session = new SessionManager(getApplicationContext());

editText1 = (EditText) findViewById(R.id.editText);

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

editText2 = (EditText) findViewById(R.id.editText3);

//=======================================

//=======================================

phoneBook1 = (ImageView) findViewById(R.id.iv\_phonebook1);

phoneBook2 = (ImageView) findViewById(R.id.iv\_phonebook2);

phoneBook3 = (ImageView) findViewById(R.id.iv\_phonebook3);

phoneBook1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// openContact1(101);

globalContact = 101;

flow = 1;

if (Build.VERSION.SDK\_INT < 23) {

openContact1();

} else

// for alove L version we need to get permission at the door to get throught that location ...

if (checkSelfPermission(Manifest.permission.READ\_CONTACTS) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.SEND\_SMS) != PackageManager.PERMISSION\_GRANTED) { // checkSelfPermission is a method avail in 23 api ..without if condition of (SDK\_INT < 23 ) you cant implement it..

requestPermissions(new String[]{Manifest.permission.READ\_CONTACTS, Manifest.permission.ACCESS\_FINE\_LOCATION, Manifest.permission.SEND\_SMS}, getContact);

} else {

openContact1();

}

}

});

phoneBook2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// openContact1(102);

globalContact = 102;

flow = 1;

if (Build.VERSION.SDK\_INT < 23) {

openContact1();

} else

// for alove L version we need to get permission at the door to get throught that location ...

if (checkSelfPermission(Manifest.permission.READ\_CONTACTS) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.SEND\_SMS) != PackageManager.PERMISSION\_GRANTED) { // checkSelfPermission is a method avail in 23 api ..without if condition of (SDK\_INT < 23 ) you cant implement it..

requestPermissions(new String[]{Manifest.permission.READ\_CONTACTS, Manifest.permission.ACCESS\_FINE\_LOCATION, Manifest.permission.SEND\_SMS}, getContact);

} else {

openContact1();

}

}

});

phoneBook3.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// openContact1(103);

globalContact = 103;

flow = 1;

if (Build.VERSION.SDK\_INT < 23) {

openContact1();

} else

// for alove L version we need to get permission at the door to get throught that location ...

if (checkSelfPermission(Manifest.permission.READ\_CONTACTS) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.SEND\_SMS) != PackageManager.PERMISSION\_GRANTED) { // checkSelfPermission is a method avail in 23 api ..without if condition of (SDK\_INT < 23 ) you cant implement it..

requestPermissions(new String[]{Manifest.permission.READ\_CONTACTS, Manifest.permission.ACCESS\_FINE\_LOCATION, Manifest.permission.SEND\_SMS}, getContact);

} else {

openContact1();

}

}

});

// location and message

editText1 = (EditText) findViewById(R.id.editText);

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

editText3 = (EditText) findViewById(R.id.editText3);

}

@Override

public void onActivityResult(int requestCode, int resultCode, Intent data) {

super.onActivityResult(requestCode, resultCode, data);

if (requestCode == (101) && resultCode == RESULT\_OK && null != data) {

Uri contactData = data.getData();

//String[] projection = { Phone.NUMBER, Phone.DISPLAY\_NAME };

Cursor c = managedQuery(contactData, null, null, null, null);

if (c.moveToFirst()) {

String id =

c.getString(c.getColumnIndexOrThrow(ContactsContract.Contacts.\_ID));

String hasPhone =

c.getString(c.getColumnIndex(ContactsContract.Contacts.HAS\_PHONE\_NUMBER));

if (hasPhone.equalsIgnoreCase("1")) {

Cursor phones = getContentResolver().query(

ContactsContract.CommonDataKinds.Phone.CONTENT\_URI, null,

ContactsContract.CommonDataKinds.Phone.CONTACT\_ID + " = " + id,

null, null);

phones.moveToFirst();

String phn\_no1 = phones.getString(phones.getColumnIndex("data1"));

//String phn\_no2 = phones.getString(phones.getColumnIndex("data2"));

//String phn\_no3 = phones.getString(phones.getColumnIndex("data3"));

//String name = c.getString(c.getColumnIndex(ContactsContract.CommonDataKinds.StructuredPostal.DISPLAY\_NAME));

//Toast.makeText(this, "contact info : " + phn\_no + "\n" + name, Toast.LENGTH\_LONG).show();

editText1.setText(phn\_no1);

//editText2.setText(phn\_no2);

//editText3.setText(phn\_no3);

}

}

} else if (requestCode == (102) && resultCode == RESULT\_OK && null != data) {

Uri contactData = data.getData();

//String[] projection = { Phone.NUMBER, Phone.DISPLAY\_NAME };

Cursor c = managedQuery(contactData, null, null, null, null);

if (c.moveToFirst()) {

String id =

c.getString(c.getColumnIndexOrThrow(ContactsContract.Contacts.\_ID));

String hasPhone =

c.getString(c.getColumnIndex(ContactsContract.Contacts.HAS\_PHONE\_NUMBER));

if (hasPhone.equalsIgnoreCase("1")) {

Cursor phones = getContentResolver().query(

ContactsContract.CommonDataKinds.Phone.CONTENT\_URI, null,

ContactsContract.CommonDataKinds.Phone.CONTACT\_ID + " = " + id,

null, null);

phones.moveToFirst();

//String phn\_no1 = phones.getString(phones.getColumnIndex("data1"));

String phn\_no2 = phones.getString(phones.getColumnIndex("data1"));

//String phn\_no3 = phones.getString(phones.getColumnIndex("data1"));

//String name = c.getString(c.getColumnIndex(ContactsContract.CommonDataKinds.StructuredPostal.DISPLAY\_NAME));

//Toast.makeText(this, "contact info : " + phn\_no + "\n" + name, Toast.LENGTH\_LONG).show();

//editText1.setText(phn\_no1);

editText2.setText(phn\_no2);

//editText3.setText(phn\_no3);

}

}

} else //(requestCode == (101) && resultCode == RESULT\_OK && null != data)

{

try {

Uri contactData = data.getData();

//String[] projection = { Phone.NUMBER, Phone.DISPLAY\_NAME };

Cursor c = managedQuery(contactData, null, null, null, null);

if (c.moveToFirst()) {

String id =

c.getString(c.getColumnIndexOrThrow(ContactsContract.Contacts.\_ID));

String hasPhone =

c.getString(c.getColumnIndex(ContactsContract.Contacts.HAS\_PHONE\_NUMBER));

if (hasPhone.equalsIgnoreCase("1")) {

Cursor phones = getContentResolver().query(

ContactsContract.CommonDataKinds.Phone.CONTENT\_URI, null,

ContactsContract.CommonDataKinds.Phone.CONTACT\_ID + " = " + id,

null, null);

phones.moveToFirst();

//String phn\_no1 = phones.getString(phones.getColumnIndex("data1"));

//String phn\_no2 = phones.getString(phones.getColumnIndex("data1"));

String phn\_no3 = phones.getString(phones.getColumnIndex("data1"));

//String name = c.getString(c.getColumnIndex(ContactsContract.CommonDataKinds.StructuredPostal.DISPLAY\_NAME));

//Toast.makeText(this, "contact info : " + phn\_no + "\n" + name, Toast.LENGTH\_LONG).show();

//editText1.setText(phn\_no1);

//editText2.setText(phn\_no2);

editText3.setText(phn\_no3);

}

}

} catch (Exception e) {

}

} //end of else

}

public void save(View v) {

saveContactd = true;

flow = 2;

if (Build.VERSION.SDK\_INT < 23) {

saveContactData();

} else

// for alove L version we need to get permission at the door to get throught that location ...

if (checkSelfPermission(Manifest.permission.READ\_CONTACTS) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED || checkSelfPermission(Manifest.permission.SEND\_SMS) != PackageManager.PERMISSION\_GRANTED) { // checkSelfPermission is a method avail in 23 api ..without if condition of (SDK\_INT < 23 ) you cant implement it..

requestPermissions(new String[]{Manifest.permission.READ\_CONTACTS, Manifest.permission.ACCESS\_FINE\_LOCATION, Manifest.permission.SEND\_SMS}, getContact);

} else {

saveContactData();

}

}

// public void next(View view){

// Intent i=new Intent (this,Second.class);

//

// startActivity(i);

//

// }

//

// @Override

// public boolean onCreateOptionsMenu(Menu menu) {

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

// getMenuInflater().inflate(R.menu.main, menu);

// return true;

// }

public void back(View v) {

Intent i = new Intent(getApplicationContext(), NavigationAlert.class);

startActivity(i);

}

//===================open contact

public void openContact1() {

Intent intent = new Intent(Intent.ACTION\_PICK);

intent.setType(ContactsContract.Contacts.CONTENT\_TYPE);

if (intent.resolveActivity(getPackageManager()) != null) {

startActivityForResult(intent, globalContact);

}

}

// permission ===================================================

@Override

public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {

super.onRequestPermissionsResult(requestCode, permissions, grantResults);

if (requestCode == getContact) {

if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

if (flow == 0) {

} else if (flow == 1) {

openContact1();

} else if (flow == 2) {

saveContactData();

}

}

}

}

public void saveContactData() {

String cn1 = editText1.getText().toString();

String cn2 = editText2.getText().toString();

String cn3 = editText3.getText().toString();

// Check if username, password is filled

if (cn1.trim().length() > 0 && cn2.trim().length() > 0 && cn3.trim().length() > 0) {

// session.createLoginSession();

session.createContactSession(cn1, cn2, cn3);

Intent i = new Intent(getApplicationContext(), PeopleSecond.class);

startActivity(i);

Toast.makeText(this, "Contacts Saved Successfully", Toast.LENGTH\_LONG).show();

finish();

} else {

// username / password doesn't match

Toast.makeText(getApplicationContext(), "Please Add all 3 Contacts", Toast.LENGTH\_LONG).show();

}

}

}

**PeopleSecond.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.os.Bundle;

import android.support.v7.app.ActionBar;

import android.support.v7.app.AppCompatActivity;

import android.support.v7.widget.Toolbar;

import android.telephony.SmsManager;

import android.view.View;

import android.widget.ImageButton;

import android.widget.TextView;

import android.widget.Toast;

import com.example.akhilesh.oneplussafety.R;

import com.example.akhilesh.oneplussafety.NavigationAlert;

import com.example.akhilesh.oneplussafety.GPSTracker;

import com.example.akhilesh.oneplussafety.SessionManager;

import java.util.HashMap;

public class PeopleSecond extends AppCompatActivity {

ImageButton backButton;

SmsManager sm;

TextView tv1,tv2,tv3,tv4;

ImageButton edit;

// Session Manager Class

SessionManager session;

public static final String DEFAULT="N/A";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.people\_second

Toolbar tb = (Toolbar) findViewById(R.id.toolbar1);

backButton = (ImageButton) findViewById(R.id.back);

// title = (TextView) findViewById(R.id.toolbar\_title);

// title.setText(R.string.welcome);

backButton.setVisibility(View.VISIBLE);

backButton.setImageResource(R.drawable.ic\_keyboard\_arrow\_left\_24dp);

// notificationIcon.setImageResource(R.drawable.notification);

// notificationIcon.setVisibility(View.VISIBLE);

setSupportActionBar(tb);

getSupportActionBar().setDisplayShowTitleEnabled(true);

getSupportActionBar().setDisplayOptions(ActionBar.DISPLAY\_SHOW\_CUSTOM);

getSupportActionBar().setCustomView(R.layout.title);

// getSupportActionBar().setTitle("Help");

TextView tv= (TextView) findViewById(R.id.ab);

tv.setText(R.string.savedcontacts);

// Session Manager

session = new SessionManager(getApplicationContext());

// Email, Password input text

tv1 = (TextView) findViewById(R.id.textview1);

tv2 = (TextView) findViewById(R.id.textview2);

tv3 = (TextView) findViewById(R.id.textview3);

//Toast.makeText(getApplicationContext(), "User Login Status: " + session.isLoggedIn(), Toast.LENGTH\_LONG).show();

// Login button

edit = (ImageButton) findViewById(R.id.edit);

session.checkStatusOfContactsAdd();

HashMap<String, String> user = session.getUserDetails();

String number1 = user.get(SessionManager.CONTACT1);

// name

String number2 = user.get(SessionManager.CONTACT2);

// email

String number3 = user.get(SessionManager.CONTACT3);

// displaying user data

tv1.setText(number1);

tv2.setText(number2);

tv3.setText(number3);

// Login button click event

edit.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View arg0) {

session.logoutUser();

}

});

//Location

GPSTracker mGPS = new GPSTracker(this);

tv4 = (TextView) findViewById(R.id.texts);

if(mGPS.canGetLocation ){

mGPS.getLocation();

tv4.setText( mGPS.getLatitude() + "," + mGPS.getLongitude());

}else{

tv4.setText("Unabletofind");

// System.out.println("Unable");

tv1= (TextView) findViewById(R.id.textview1);

tv2= (TextView) findViewById(R.id.textview2);

tv3= (TextView) findViewById(R.id.textview3);

sm=SmsManager.getDefault();

}

public void send(View view){

sm.sendTextMessage(tv1.getText().toString(),null,"I'm in danger..My current location is http://maps.google.com/?q="+tv4.getText().toString(),null,null);

sm.sendTextMessage(tv2.getText().toString(),null,"I'm in danger..My current location is http://maps.google.com/?q="+tv4.getText().toString(),null,null);

sm.sendTextMessage(tv3.getText().toString(),null,"I'm in danger..My current location is http://maps.google.com/?q="+tv4.getText().toString(),null,null);

Toast.makeText(this,"message sent",Toast.LENGTH\_LONG).show();

}

public void back(View v) {

Intent i = new Intent(getApplicationContext(), NavigationAlert.class);

startActivity(i);

}

}

**RecievedCall.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.media.Ringtone;

import android.media.RingtoneManager;

import android.net.Uri;

import android.os.Bundle;

import android.os.Vibrator;

import android.support.annotation.Nullable;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.ImageView;

import com.example.akhilesh.oneplussafety.R;

public class RecievedCall extends AppCompatActivity implements View.OnClickListener {

ImageView attendCall, endCall;

private Ringtone r;

private Vibrator myVib;

@Override

protected void onCreate(@Nullable Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.recievedcall);

playRingtone();

attendCall = (ImageView) findViewById(R.id.attend\_call);

endCall = (ImageView) findViewById(R.id.end\_call);

attendCall.setOnClickListener(this);

endCall.setOnClickListener(this);

}

@Override

public void onClick(View v) {

switch (v.getId()) {

case R.id.attend\_call:

if (r.isPlaying()) {

r.stop();

myVib.cancel();

}

Intent intent1 = new Intent(this, VoiceCallActivity.class);

startActivity(intent1);

finish();

break;

case R.id.end\_call:

if (r.isPlaying()) {

r.stop();

myVib.cancel();

}

finish();

break;

}

}

private Ringtone playRingtone() {

Uri notification = RingtoneManager.getDefaultUri(RingtoneManager.TYPE\_RINGTONE);

r = RingtoneManager.getRingtone(getApplicationContext(), notification);

myVib = (Vibrator) this.getSystemService(VIBRATOR\_SERVICE);

myVib.vibrate(40000);

r.play();

return r;

}

}

**SessionManager.java**

package com.example.akhilesh.oneplussafety;

import android.content.Context;

import android.content.Intent;

import android.content.SharedPreferences;

import android.content.SharedPreferences.Editor;

import com.example.akhilesh.oneplussafety.People;

import java.util.HashMap;

public class SessionManager {

// Shared Preferences

private SharedPreferences sharedPreferences;

// Editor for Shared preferences

private Editor editor;

// Context

private Context \_context;

// Shared pref mode

private int PRIVATE\_MODE = 0;

// Sharedpref file name

// private static final String PREF\_NAME = "AndroidHivePref";

// All Shared Preferences Keys

private static final String IS\_CADDED = "Is\_C\_Added";

// User name (make variable public to access from outside)

public static final String CONTACT1 = "number1";

// Email address (make variable public to access from outside)

public static final String CONTACT2 = "number2";

public static final String CONTACT3 = "number4";

// Constructor

public SessionManager(Context context) {

this.\_context = context;

sharedPreferences = \_context.getSharedPreferences("IU", PRIVATE\_MODE);

editor = sharedPreferences.edit();

editor.apply();

}

/\*\*

\* Create login session

\*/

public void createContactSession(String number1, String number2, String number3) {

// Storing login value as TRUE

editor.putBoolean(IS\_CADDED, true);

// Storing name in pref

editor.putString(CONTACT1, number1);

// Storing email in pref

editor.putString(CONTACT2, number2);

editor.putString(CONTACT3, number3);

// commit changes

editor.commit();

}

/\*\*

\* Check login method wil check user login status

\* If false it will redirect user to login page

\* Else won't do anything

\*/

public void checkStatusOfContactsAdd() {

// Check login status

if (!this.isLoggedIn()) {

// user is not logged in redirect him to Login Activity

Intent i = new Intent(\_context, People.class);

// Closing all the Activities

i.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP);

// Add new Flag to start new Activity

i.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

// Staring Login Activity

\_context.startActivity(i);

}

}

/\*\*

\* Get stored session data

\*/

public HashMap<String, String> getUserDetails() {

HashMap<String, String> user = new HashMap<String, String>();

// user name

user.put(CONTACT1, sharedPreferences.getString(CONTACT1, null));

// user email id

user.put(CONTACT2, sharedPreferences.getString(CONTACT2, null));

user.put(CONTACT3, sharedPreferences.getString(CONTACT3, null));

// return user

return user;

}

/\* \* Clear session details

\* \*/

public void logoutUser() {

// Clearing all data from Shared Preferences

editor.clear();

editor.commit();

// After logout redirect user to Loing Activity

Intent i = new Intent(\_context, People.class);

// Closing all the Activities

i.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP);

// Add new Flag to start new Activity

i.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

// Staring Login Activity

\_context.startActivity(i);

}

/\*\*

\* Quick check for login

\*\*/

// Get Login State

public boolean isLoggedIn() {

return sharedPreferences.getBoolean(IS\_CADDED, false);

}

}

**Splash.java**

package com.example.akhilesh.oneplussafety;

import android.app.Activity;

import android.content.Context;

import android.content.Intent;

import android.os.Bundle;

import com.example.akhilesh.oneplussafety.R;

public class Splash extends Activity {

Context context;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.splash);

new Thread() {

public void run() {

try {

Thread.sleep(3000);

} catch (Exception e) {

}

startActivity( new Intent(Splash.this, MainActivity.class));

}

}.start();

}

@Override

protected void onPause() {

super.onPause();

finish();

}

}

**TabBars.java**

package com.example.akhilesh.oneplussafety;

import android.os.Bundle;

import android.support.v4.app.FragmentManager;

import android.support.v4.app.FragmentTransaction;

import android.support.v7.app.AppCompatActivity;

import com.example.akhilesh.oneplussafety.R;

import com.example.akhilesh.oneplussafety.TabFragment;

public class TabBars extends AppCompatActivity {

FragmentManager mFragmentManager;

FragmentTransaction mFragmentTransaction;

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.tab\_bars);

mFragmentManager = getSupportFragmentManager();

mFragmentTransaction = mFragmentManager.beginTransaction();

mFragmentTransaction.replace(R.id.containerView,new TabFragment()).commit();

}

}

**TabFragment.java**

package com.example.akhilesh.oneplussafety;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.design.widget.TabLayout;

import android.support.v4.app.Fragment;

import android.support.v4.app.FragmentManager;

import android.support.v4.app.FragmentPagerAdapter;

import android.support.v4.view.ViewPager;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import com.example.akhilesh.oneplussafety.R;

public class TabFragment extends Fragment {

public static TabLayout tabLayout;

public static ViewPager viewPager;

public static int int\_items = 5 ;

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

/\*\*

\*Inflate tab\_layout and setup Views.

\*/

View x = inflater.inflate(R.layout.tab\_layout,null);

tabLayout = (TabLayout) x.findViewById(R.id.tabs);

viewPager = (ViewPager) x.findViewById(R.id.viewpager);

/\*\*

\*Set an Apater for the View Pager

\*/

viewPager.setAdapter(new MyAdapter(getChildFragmentManager()));

/\*\*

\* Now , this is a workaround ,

\* The setupWithViewPager dose't works without the runnable .

\* Maybe a Support Library Bug .

\*/

tabLayout.post(new Runnable() {

@Override

public void run() {

tabLayout.setupWithViewPager(viewPager);

}

});

return x;

}

class MyAdapter extends FragmentPagerAdapter {

public MyAdapter(FragmentManager fm) {

super(fm);

}

/\*\*

\* Return fragment with respect to Position .

\*/

@Override

public Fragment getItem(int position)

{

switch (position){

case 0 : return new DomesticViolenceFragment();

case 1 : return new ThreatFragment();

case 2 : return new CorruptionFragment();

case 3 : return new EveTeasingFragment();

case 4 : return new TheftAttemptFragment();

}

return null;

}

@Override

public int getCount() {

return int\_items;

}

/\*\*

\* This method returns the title of the tab according to the position.

\*/

@Override

public CharSequence getPageTitle(int position) {

switch (position){

case 0 :

return "Domestic Violence";

case 1 :

return "Threat";

case 2 :

return "Corruption";

case 3 :

return "Eve Teasing";

case 4 :

return "Theft Attempt";

}

return null;

}

}

}

**TheftAttemptFragment.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v4.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import com.example.akhilesh.oneplussafety.R;

public class TheftAttemptFragment extends Fragment {

Button button;

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

View V= inflater.inflate(R.layout.theft\_attempt,container,false);

button= (Button) V.findViewById(R.id.Button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent=null, chooser=null;

intent=new Intent(Intent.ACTION\_SEND);

intent.setData(Uri.parse("mailto:"));

String[] to={"ishumehta99@gmail.com"};

intent.putExtra(Intent.EXTRA\_EMAIL,to);

intent.putExtra(Intent.EXTRA\_SUBJECT,"Regarding Theft Attempt");

intent.setType("message/rfc822");

chooser=Intent.createChooser(intent,"Send Email");

startActivity(chooser);

}

});

return V;

}

}

**ThreatFragment.java**

package com.example.akhilesh.oneplussafety;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v4.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import com.example.akhilesh.oneplussafety.R;

public class ThreatFragment extends Fragment {

Button button;

@Nullable

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

View V= inflater.inflate(R.layout.threat,container,false);

button= (Button) V.findViewById(R.id.Button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent=null, chooser=null;

intent=new Intent(Intent.ACTION\_SEND);

intent.setData(Uri.parse("mailto:"));

String[] to={"ay19jsn@gmail.com"};

intent.putExtra(Intent.EXTRA\_EMAIL,to);

intent.putExtra(Intent.EXTRA\_SUBJECT,"Regarding Threat");

intent.setType("message/rfc822");

chooser=Intent.createChooser(intent,"Send Email");

startActivity(chooser);

}

});

return V;

}

}

**VoiceCallActivity.java**

package com.example.akhilesh.oneplussafety;

import android.os.Bundle;

import android.os.Handler;

import android.os.SystemClock;

import android.support.annotation.Nullable;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.LinearLayout;

import android.widget.TextView;

import com.example.akhilesh.oneplussafety.R;

import java.text.SimpleDateFormat;

import java.util.TimeZone;

public class VoiceCallActivity extends AppCompatActivity {

LinearLayout lv\_endcall;

TextView tv\_time;

long startTime, timeInMilliseconds = 0;

Handler customHandler = new Handler();

@Override

protected void onCreate(@Nullable Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.voice\_layout);

lv\_endcall = (LinearLayout) findViewById(R.id.lv\_endcall);

tv\_time = (TextView) findViewById(R.id.time);

lv\_endcall.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

customHandler.removeCallbacks(updateTimerThread);

finish();

}

});

startTime = SystemClock.uptimeMillis();

customHandler.postDelayed(updateTimerThread, 0);

}

private Runnable updateTimerThread = new Runnable() {

public void run() {

timeInMilliseconds = SystemClock.uptimeMillis() - startTime;

tv\_time.setText(getDateFromMillis(timeInMilliseconds));

customHandler.postDelayed(this, 1000);

}

};

public static String getDateFromMillis(long d) {

SimpleDateFormat df = new SimpleDateFormat("HH:mm:ss");

df.setTimeZone(TimeZone.getTimeZone("GMT"));

return df.format(d);

}

}

/\* private void timer(int timeInMillis) {

new CountDownTimer(timeInMillis, 1000) {

@Override

public void onTick(long leftTimeInMilliseconds) {

long seconds = leftTimeInMilliseconds / 1000;

minute = (seconds % 3600) / 60;

sec = (seconds % 60);

timer.setText(String.format("%02d:%02d",

(seconds % 3600) / 60, (seconds % 60)));

}

@Override

public void onFinish() {

endCall();

// timer.setText("Time up!");

timer.setVisibility(View.VISIBLE);

}

}.start();

}

\*/

**VolumeChangeReceiver.java**

package com.example.akhilesh.oneplussafety;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import com.example.akhilesh.oneplussafety.PeopleSecond;

public class VolumeChangeReceiver extends BroadcastReceiver {

@Override

public void onReceive(Context context, Intent intent) {

if (intent.getAction().equals("android.media.VOLUME\_CHANGED\_ACTION")) {

int newVolume = intent.getIntExtra("android.media.EXTRA\_VOLUME\_STREAM\_VALUE",0);

int oldVolume = intent.getIntExtra("android.media.EXTRA\_PREV\_VOLUME\_STREAM\_VALUE", 0);

if (newVolume != oldVolume) {

Intent i = new Intent();

i.setClass(context, PeopleSecond.class);

i.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

context.startActivity(i);

}

}

}

}

**Layout**

**about\_us.xml**

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

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

android:layout\_width="match\_parent"

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

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<include

android:id="@+id/toolbar"

layout="@layout/toolbar" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="310dp"

android:background="@drawable/background"

android:orientation="vertical"

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

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

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

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

<ImageView

android:layout\_width="150dp"

android:layout\_height="50dp"

android:src="@drawable/aboutus" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:text="CAREVAULT"

android:textSize="20sp"

android:textStyle="bold" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/aboutus" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="0dp"

android:layout\_weight=".4">

<Button

style="?android:attr/borderlessButtonStyle"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:onClick="call"

android:text="Contact Us"

android:textColor="#B36887"

android:textSize="15sp" />/>

</LinearLayout>

</LinearLayout>

</ScrollView>

**activity\_main.xml**

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

<LinearLayout 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:background="@drawable/intro\_without\_logo"

android:orientation="vertical"

android:weightSum="1"

tools:context="com.example.ishu.carevault.activites.MainActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="vertical">

<LinearLayout

android:id="@+id/lv\_sos\_emergency"

android:layout\_width="100dp"

android:layout\_height="100dp"

android:layout\_gravity="center"

android:layout\_marginTop="10dp"

android:background="@drawable/red\_background"

android:gravity="center">

<TextView

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_gravity="center"

android:gravity="center"

android:onClick="Emer"

android:text="SOS/\nEmergency"

android:textColor="@color/white"

android:textSize="15sp"

android:textStyle="bold"

tools:ignore="OnClick" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"

android:gravity="center"

android:orientation="horizontal"

android:weightSum="1">

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".50"

android:gravity="center">

<LinearLayout

android:id="@+id/profile"

android:layout\_width="100dp"

android:layout\_height="120dp"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="vertical"

>

<ImageView

android:layout\_width="match\_parent"

android:layout\_height="80dp"

android:src="@drawable/user"

/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="Profile"

android:textColor="@android:color/holo\_red\_dark"

android:textSize="16sp"

android:textStyle="bold"

/>

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".50"

android:gravity="center">

<LinearLayout

android:layout\_width="100dp"

android:layout\_height="120dp"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="vertical"

android:onClick="Complaint"

>

<ImageView

android:layout\_width="match\_parent"

android:layout\_height="80dp"

android:src="@drawable/complaint"

/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="Complaint"

android:textColor="@android:color/holo\_red\_dark"

android:textSize="16sp"

android:textStyle="bold"

/>

</LinearLayout>

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:gravity="center"

android:orientation="horizontal"

android:weightSum="1">

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".50"

android:gravity="center">

<LinearLayout

android:layout\_width="100dp"

android:layout\_height="100dp"

android:gravity="center"

android:visibility="gone"

>

<Button

android:layout\_width="70dp"

android:layout\_height="70dp"

android:layout\_gravity="center"

android:background="@drawable/alert"

android:onClick="Alert" />

</LinearLayout>

<LinearLayout

android:id="@+id/alarm"

android:layout\_width="100dp"

android:layout\_height="120dp"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="vertical"

>

<ImageView

android:layout\_width="match\_parent"

android:layout\_height="80dp"

android:src="@drawable/alarm\_clock"

/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="Alarm"

android:textColor="@android:color/holo\_red\_dark"

android:textSize="16sp"

android:textStyle="bold"

/>

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".50"

android:gravity="center">

<LinearLayout

android:id="@+id/fake\_call"

android:layout\_width="100dp"

android:layout\_height="120dp"

android:layout\_gravity="center"

android:gravity="center"

android:orientation="vertical"

>

<ImageView

android:layout\_width="match\_parent"

android:layout\_height="80dp"

android:src="@drawable/support"

/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="Fake call"

android:textColor="@android:color/holo\_red\_dark"

android:textSize="16sp"

android:textStyle="bold"

/>

</LinearLayout>

</LinearLayout>

</LinearLayout>

</LinearLayout>

</LinearLayout>

**activity\_navigation\_alert.xml**

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

<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/drawer\_layout"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:openDrawer="start">

<include

layout="@layout/app\_bar\_navigation\_alert"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

<android.support.design.widget.NavigationView

android:id="@+id/nav\_view"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:layout\_gravity="start"

android:fitsSystemWindows="true"

app:headerLayout="@layout/nav\_header\_navigation\_alert"

app:menu="@menu/activity\_navigation\_alert\_drawer" />

</android.support.v4.widget.DrawerLayout>

**app\_bar\_navigation\_alert.xml**

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

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:context="com.example.ishu.carevault.activites.NavigationAlert">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.Toolbar

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="?attr/colorPrimary"

app:popupTheme="@style/AppTheme.PopupOverlay" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content\_navigation\_alert" />

</android.support.design.widget.CoordinatorLayout>

**app\_bar\_navigation\_complaint.xml**

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

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:context="com.example.ishu.carevault.activites.NavigationComplaint">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.Toolbar

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

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="@color/colorPrimary"

android:id="@+id/toolbar"

android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content\_navigation\_complaint" />

</android.support.design.widget.CoordinatorLayout>

**content\_navigation\_alert.xml**

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

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

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

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

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

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

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

android:id="@+id/content\_frame"

app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"

tools:context="com.example.ishu.carevault.activites.NavigationAlert"

tools:showIn="@layout/app\_bar\_navigation\_alert"

android:background="@drawable/background">

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/linearLayout"

android:gravity="center"

>

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:src="@drawable/alertlogo"/>

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/linearLayout">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/alertinfo"

android:textSize="15dp"

android:textColor="#06A7BF"/>

</LinearLayout>

</RelativeLayout>

**content\_navigation\_complaint.xml**

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

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

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

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

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

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

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

android:background="@drawable/background"

android:id="@+id/content\_frame"

app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"

tools:context="com.example.ishu.carevault.activites.NavigationComplaint"

tools:showIn="@layout/app\_bar\_navigation\_complaint">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight=".5">

<ImageView

android:id="@+id/ImageView3\_Left"

android:layout\_width="350dp"

android:layout\_height="200dp"

android:layout\_gravity="center\_horizontal"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight=".5">

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/TextViewTopTitle"

android:text="@string/complainthome"

android:textColor="#abc"

android:layout\_gravity="top"

android:gravity="top|center" />

</LinearLayout>

</LinearLayout>

</RelativeLayout>

**corruption.xml**

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

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

android:orientation="vertical" android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/background">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="30sp"

android:layout\_marginTop="40dp"

android:gravity="center"

android:id="@+id/textView"

android:layout\_gravity="center\_horizontal"

android:layout\_centerHorizontal="true"

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

android:text="Corruption"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autoLink="web"

android:textSize="15sp"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"

android:text="@string/corruption"

android:textColor="#000"

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

android:textStyle="italic"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_horizontal"

android:layout\_marginTop="45dp"

android:id="@+id/Button"

android:text="Send Mail!"/>

</LinearLayout>

**domestic\_violence.xml**

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

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

android:orientation="vertical" android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/background">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textSize="30sp"

android:layout\_gravity="center"

android:id="@+id/textView"

android:layout\_centerHorizontal="true"

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

android:text="Domestic Violence"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autoLink="web"

android:textSize="15sp"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"

android:text="@string/dmv"

android:textColor="#000"

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

android:textStyle="italic"

android:id="@+id/textView4" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/Button"

android:text="Send Mail!"

android:layout\_gravity="center"

android:layout\_below="@+id/textView4"

android:layout\_marginTop="30dp" />

</LinearLayout>

**eve\_teasing.xml**

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

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

android:orientation="vertical" android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/background">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:textSize="30sp"

android:gravity="center"

android:id="@+id/textView"

android:layout\_gravity="center\_horizontal"

android:textColor="@android:color/holo\_green\_dark"

android:text="Eve Teasing"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autoLink="web"

android:textSize="15sp"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"

android:text="@string/eve"

android:textColor="#000"

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

android:textStyle="italic"

android:id="@+id/textView3" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/Button"

android:layout\_gravity="center\_horizontal"

android:text="Send Mail!"

android:layout\_marginTop="30dp"

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

</LinearLayout>

**frag\_home.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="15dp"

android:background="#000000">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/introhome"

android:textColor="#fff"

/>

</LinearLayout>

**help.xml**

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

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

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:background="#64e6f5ef"

android:orientation="vertical">

<include

android:id="@+id/toolbar"

layout="@layout/toolbar" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"></LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="20dp"

>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:layout\_marginTop="10dp"

android:text="@string/introhelp"

android:textColor="@color/colorPrimary"

android:textSize="15sp"

android:textStyle="bold" />

</LinearLayout>

<ExpandableListView

android:id="@+id/lvExp"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

</LinearLayout>

**list\_group.xml**

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

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

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:padding="8dp"

android:background="#4F4C4B">

<TextView

android:id="@+id/lblListHeader"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:paddingLeft="?android:attr/expandableListPreferredItemPaddingLeft"

android:textSize="15sp"

android:textColor="#178178" />

</LinearLayout>

**list\_item.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="55dip"

android:orientation="vertical" >

<TextView

android:id="@+id/lblListItem"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:textSize="15sp"

android:paddingTop="5dp"

android:textColor="@color/buttoncolor"

android:paddingBottom="5dp"

android:paddingLeft="5dp"/>

</LinearLayout>

**nav\_header\_navigation\_alert.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="@dimen/nav\_header\_height"

android:background="@drawable/side\_nav\_bar"

android:gravity="bottom"

android:orientation="vertical"

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

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

android:theme="@style/ThemeOverlay.AppCompat.Dark">

<ImageView

android:id="@+id/imageView"

android:layout\_width="80dp"

android:layout\_height="80dp"

android:paddingTop="@dimen/nav\_header\_vertical\_spacing"

android:src="@mipmap/carevault" />

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:paddingTop="@dimen/nav\_header\_vertical\_spacing"

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

android:text="ALert"

android:textAppearance="@style/TextAppearance.AppCompat.Body1" />

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

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

android:text="Stay Safe" /></LinearLayout>

**navigation\_complaint.xml**

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

<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/drawerLayout"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:openDrawer="start">

<include

layout="@layout/app\_bar\_navigation\_complaint"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

<android.support.design.widget.NavigationView

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

xmlns:app="http://schemas.android.com/apk/res-auto"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="start"

android:id="@+id/shitstuff"

app:itemTextColor="@color/black"

app:menu="@menu/drawermenu"

android:layout\_marginTop="24dp" />

</android.support.v4.widget.DrawerLayout>

**people\_second.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/background"

android:orientation="vertical">

<include

android:id="@+id/toolbar1"

layout="@layout/toolbar1"></include>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center">

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="My Emergency Contacts"

android:textAlignment="center"

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

android:textStyle="bold" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="50dp">

<TextView

android:id="@+id/contact1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight=".5"

android:text="Contact 1"

android:textSize="20sp" />

<TextView

android:id="@+id/textview1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight=".5"

android:text=""

android:textSize="20sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="50dp">

<TextView

android:id="@+id/contact2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight=".5"

android:text="Contact 2"

android:textSize="20sp" />

<TextView

android:id="@+id/textview2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight=".5"

android:text=""

android:textSize="20sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="50dp">

<TextView

android:id="@+id/contact3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight=".5"

android:text="Contact 3"

android:textSize="20sp" />

<TextView

android:id="@+id/textview3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight=".5"

android:text=""

android:textSize="20sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center">

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:onClick="send"

android:text="Send Message" />

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/texts"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text=""

android:visibility="invisible" />

</LinearLayout>

</LinearLayout>

**people.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/background"

android:orientation="vertical">

<include

android:id="@+id/toolbar"

layout="@layout/toolbar"></include>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="5dp">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight=".2"

android:gravity="center">

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Add Emergency Contacts"

android:textAlignment="center"

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

android:textColor="@color/colorPrimaryDark"

android:textStyle="bold" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight=".4"

android:gravity="center"

android:orientation="vertical">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="40dp"

android:gravity="center">

<TextView

android:id="@+id/contact1"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".40"

android:text="Emergency Contact 1"

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

android:textSize="14sp"

android:textStyle="bold"

/>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="5dp"

android:layout\_weight=".50"

android:orientation="vertical">

<EditText

android:id="@+id/editText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_horizontal"

android:background="@null"

android:ems="10"

android:inputType="phone" />

<View

android:layout\_width="match\_parent"

android:layout\_height="1dp"

android:background="@color/colorPrimary" />

</LinearLayout>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:layout\_weight=".10"

android:gravity="center">

<ImageView

android:id="@+id/iv\_phonebook1"

android:layout\_width="25dp"

android:layout\_height="25dp"

android:src="@drawable/phonebook" />

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="40dp"

android:gravity="center"

android:orientation="horizontal">

<TextView

android:id="@+id/contact2"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".40"

android:text="Emergency Contact 2"

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

android:textSize="14sp"

android:textStyle="bold"

/>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="5dp"

android:layout\_weight=".50"

android:orientation="vertical">

<EditText

android:id="@+id/editText2"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_horizontal"

android:background="@null"

android:ems="10"

android:inputType="phone" />

<View

android:layout\_width="match\_parent"

android:layout\_height="1dp"

android:layout\_marginLeft="5dp"

android:background="@color/colorPrimary" />

</LinearLayout>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:layout\_weight=".10"

android:gravity="center">

<ImageView

android:id="@+id/iv\_phonebook2"

android:layout\_width="25dp"

android:layout\_height="25dp"

android:src="@drawable/phonebook" />

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="40dp"

android:gravity="center">

<TextView

android:id="@+id/contact3"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".40"

android:text="Emergency Contact 3"

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

android:textSize="14sp"

android:textStyle="bold"

/>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight=".50"

android:orientation="vertical">

<EditText

android:id="@+id/editText3"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_horizontal"

android:layout\_marginLeft="5dp"

android:background="@null"

android:ems="10"

android:inputType="phone" />

<View

android:layout\_width="match\_parent"

android:layout\_height="1dp"

android:background="@color/colorPrimary" />

</LinearLayout>

<LinearLayout

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:layout\_weight=".10"

android:gravity="center">

<ImageView

android:id="@+id/iv\_phonebook3"

android:layout\_width="25dp"

android:layout\_height="25dp"

android:src="@drawable/phonebook" />

</LinearLayout>

</LinearLayout>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight=".4"

android:gravity="top|center">

<Button

android:id="@+id/button"

android:layout\_width="200dp"

android:layout\_height="35dp"

android:background="@color/colorPrimary"

android:onClick="save"

android:text="Save"

android:textColor="@color/white"

android:textSize="16sp" />

</LinearLayout>

</LinearLayout>

</LinearLayout>

**recievedcall.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/background\_login"

android:orientation="vertical"

android:padding="10dp">

<ImageView

android:id="@+id/instaLogo"

android:layout\_width="200dp"

android:layout\_height="150dp"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="10dp"

android:contentDescription=""

android:src="@mipmap/carevault" />

<TextView

android:id="@+id/call\_from\_helpdesk"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

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

android:text="Incoming Call"

android:layout\_marginTop="30dp"

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

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

<ImageView

android:id="@+id/end\_call"

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_above="@+id/reject\_call"

android:layout\_marginEnd="50dp"

android:layout\_marginBottom="10dp"

android:layout\_marginRight="50dp"

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

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

android:background="@drawable/backgroun\_red"

android:contentDescription=""

android:padding="20dp"

android:src="@drawable/end\_call" />

<ImageView

android:id="@+id/attend\_call"

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_above="@+id/accept\_call"

android:layout\_marginLeft="50dp"

android:layout\_marginStart="50dp"

android:layout\_marginBottom="10dp"

android:layout\_toEndOf="@+id/center"

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

android:background="@drawable/backgroundgreen"

android:contentDescription=""

android:gravity="center"

android:padding="20dp"

android:src="@drawable/voice" />

<View

android:id="@+id/center"

android:layout\_width="2dp"

android:layout\_height="80dp"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true" />

<TextView

android:id="@+id/accept\_call"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/attend\_call"

android:layout\_alignParentBottom="true"

android:layout\_alignStart="@+id/attend\_call"

android:layout\_marginBottom="25dp"

android:text="Accept Call"

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

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

<TextView

android:id="@+id/reject\_call"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignEnd="@+id/end\_call"

android:layout\_alignRight="@+id/end\_call"

android:layout\_alignTop="@+id/accept\_call"

android:text="Reject Call"

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

android:layout\_marginBottom="25dp"

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

</RelativeLayout>

**sent\_layout.xml**

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

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

android:orientation="vertical" android:layout\_width="match\_parent"

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

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="30sp"

android:layout\_marginTop="55dp"

android:gravity="center"

android:id="@+id/textView"

android:layout\_centerHorizontal="true"

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

android:text="Sent\nFragment"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autoLink="web"

android:textSize="15sp"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"

android:text="androidbelieve.com"

android:textColor="#000"

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

android:textStyle="italic"/>

</RelativeLayout>

**splash.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/intro\_2">

</LinearLayout>

**tab\_bars.xml**

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

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

android:layout\_width="match\_parent"

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

<FrameLayout

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:id="@+id/containerView">

</FrameLayout>

</LinearLayout>

**tab\_layout.xml**

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

<LinearLayout

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

xmlns:app="http://schemas.android.com/apk/res-auto"

android:layout\_width="match\_parent"

android:orientation="vertical"

android:layout\_height="wrap\_content">

<android.support.design.widget.TabLayout

android:id="@+id/tabs"

app:tabGravity="fill"

app:tabMode="scrollable"

android:background="@color/material\_blue\_grey\_800"

app:tabIndicatorColor="@color/colorPrimary"

app:tabSelectedTextColor="@color/colorPrimary"

app:tabTextColor="@color/white"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

</android.support.design.widget.TabLayout>

<android.support.v4.view.ViewPager

android:id="@+id/viewpager"

android:layout\_width="match\_parent"

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

</android.support.v4.view.ViewPager>

</LinearLayout>

**theft\_attempt.xml**

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

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

android:orientation="vertical" android:layout\_width="match\_parent"

android:background="@drawable/background"

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

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="40dp"

android:textSize="30sp"

android:gravity="center"

android:layout\_gravity="center\_horizontal"

android:id="@+id/textView"

android:layout\_centerHorizontal="true"

android:textColor="@android:color/holo\_green\_dark"

android:text="Theft Attempt"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autoLink="web"

android:textSize="15sp"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"

android:text="@string/theft"

android:textColor="#000"

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

android:textStyle="italic"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/Button"

android:layout\_marginTop="45dp"

android:layout\_gravity="center\_horizontal"

android:text="Send Mail!" />

</LinearLayout>

**threat.xml**

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

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

android:orientation="vertical" android:layout\_width="match\_parent"

android:background="@drawable/background"

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

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="40dp"

android:textSize="30sp"

android:gravity="center"

android:id="@+id/textView"

android:layout\_gravity="center\_horizontal"

android:layout\_centerHorizontal="true"

android:textColor="@android:color/holo\_green\_dark"

android:text="Threat"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:autoLink="web"

android:textSize="15sp"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"

android:text="@string/threat"

android:textColor="#000"

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

android:textStyle="italic"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="45dp"

android:layout\_gravity="center\_horizontal"

android:id="@+id/Button"

android:text="Send Mail!" />

</LinearLayout>

**title.xml**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:gravity="center">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:textSize="18sp"

android:id="@+id/ab"

android:textStyle="bold"

android:gravity="center"

/>

</LinearLayout>

**toolbar\_complaint.xml**

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

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:fitsSystemWindows="true"

tools:context="com.example.ishu.carevault.activites.NavigationComplaint">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.Toolbar

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

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="@color/colorPrimary"

android:id="@+id/toolbar"

android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar" />

</android.support.design.widget.AppBarLayout>

</android.support.design.widget.CoordinatorLayout>

**toolbar\_home.xml**

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

<android.support.v7.widget.Toolbar

android:id="@+id/toolbar2"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:background="#136775"

android:minHeight="?attr/actionBarSize"

android:titleTextColor="#ffffff"

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

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:textSize="30sp"

android:text="CareVault"

android:id="@+id/toolbar\_title" />

</android.support.v7.widget.Toolbar>

**toolbar.xml**

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

<RelativeLayout

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

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="40dp"

android:background="#178178"

android:minHeight="?attr/actionBarSize"

android:titleTextColor="#ffffff">

<ImageButton

android:id="@+id/back"

android:layout\_width="40dp"

android:layout\_height="match\_parent"

android:layout\_gravity="center"

android:background="@drawable/ic\_keyboard\_arrow\_left\_24dp"

android:onClick="back"

/>

<TextView

android:id="@+id/toolbar\_title"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_centerInParent="true"

android:layout\_gravity="center"

android:gravity="center"

android:textColor="#FFFFFF"

android:textSize="18sp"

android:textStyle="bold" />

</RelativeLayout>

**toolbar1.xml**

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

<android.support.v7.widget.Toolbar

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

android:id="@+id/toolbar1"

android:layout\_width="match\_parent"

android:layout\_height="40dp"

android:background="#178178"

android:minHeight="?attr/actionBarSize"

android:titleTextColor="#ffffff">

<ImageButton

android:id="@+id/back"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@drawable/ic\_keyboard\_arrow\_left\_24dp"

android:onClick="back" />

<ImageButton

android:id="@+id/edit"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="right"

android:layout\_marginRight="10dp"

android:background="@drawable/ic\_mode\_edit\_24dp"

android:onClick="edit" />

</android.support.v7.widget.Toolbar>

**voice\_layout.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="#000"

android:orientation="vertical">

<TextView

android:id="@+id/name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="30dp"

android:gravity="center"

android:text="@string/help\_desk"

android:textColor="#fff"

android:textSize="25sp" />

<TextView

android:id="@+id/time"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

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

android:layout\_centerHorizontal="true"

android:layout\_marginTop="30dp"

android:gravity="center"

android:text="00:00"

android:textColor="@color/buttoncolor"

android:textSize="17dp" />

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

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

android:layout\_marginTop="40dp">

<LinearLayout

android:id="@+id/item\_1"

android:layout\_width="70dp"

android:layout\_height="70dp"

android:layout\_alignParentLeft="true"

android:layout\_marginLeft="60dp"

android:background="@drawable/circlebackground"

android:gravity="center">

<ToggleButton

android:id="@+id/mute\_audio"

style="@style/toggleButton"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:layout\_toEndOf="@+id/swap\_camera"

android:layout\_toRightOf="@+id/swap\_camera"

android:background="@drawable/audio\_toggle\_bg"

android:padding="-20dp" />

</LinearLayout>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/item\_1"

android:layout\_marginLeft="70dp"

android:layout\_marginTop="10dp"

android:gravity="center"

android:text="@string/mute"

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

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

<!--<ProgressBar-->

<!--android:id="@+id/mProgress"-->

<!--style="?android:attr/progressBarStyleSmall"-->

<!--android:layout\_width="40dp"-->

<!--android:layout\_height="40dp"-->

<!--android:layout\_centerHorizontal="true"-->

<!--android:layout\_gravity="center"-->

<!--android:foregroundGravity="right" />-->

<LinearLayout

android:layout\_width="70dp"

android:layout\_height="70dp"

android:layout\_alignParentRight="true"

android:layout\_marginRight="60dp"

android:background="@drawable/circlebackground"

android:gravity="center">

<ToggleButton

android:id="@+id/loud\_speaker"

style="@style/toggleButton"

android:layout\_width="35dp"

android:layout\_height="35dp"

android:background="@drawable/speaker\_toggle\_bg"

android:padding="-20dp" />

</LinearLayout>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:layout\_below="@+id/item\_1"

android:layout\_marginRight="65dp"

android:layout\_marginTop="10dp"

android:gravity="center"

android:text="@string/speaker"

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

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

<LinearLayout

android:id="@+id/item\_3"

android:layout\_width="70dp"

android:layout\_height="70dp"

android:layout\_below="@+id/item\_1"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="80dp"

android:visibility="invisible"

android:background="@drawable/circlebackground"

android:gravity="center">

<ImageView

android:id="@+id/patient\_attachment"

android:layout\_width="30dp"

android:layout\_height="30dp"

android:layout\_toRightOf="@+id/chat\_message"

android:background="@drawable/attach"

android:onClick="toAttachment"

android:padding="-20dp" />

</LinearLayout>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/item\_3"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="10dp"

android:gravity="center"

android:visibility="invisible"

android:text="@string/attach"

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

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

<LinearLayout

android:id="@+id/lv\_endcall"

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_below="@+id/item\_3"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="@dimen/voice\_vertical\_margin"

android:background="@drawable/circlebackgroundred"

android:gravity="center">

<ImageView

android:id="@+id/endCall"

android:layout\_width="40dp"

android:layout\_height="40dp"

android:src="@drawable/end\_call"

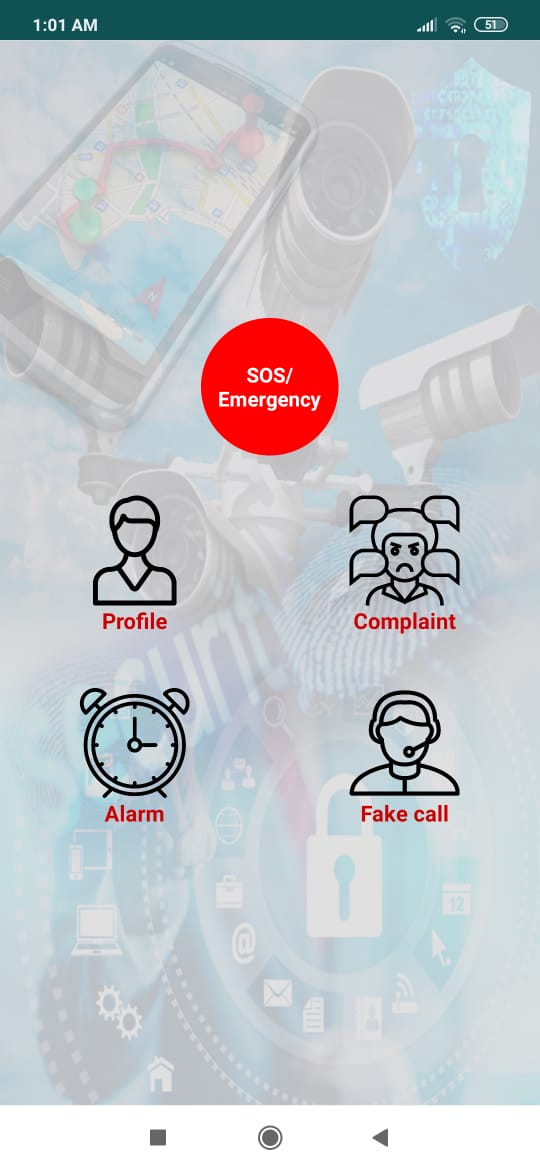
/>

</LinearLayout>

</RelativeLayout>

</RelativeLayout>

**Results and Analysis**

****

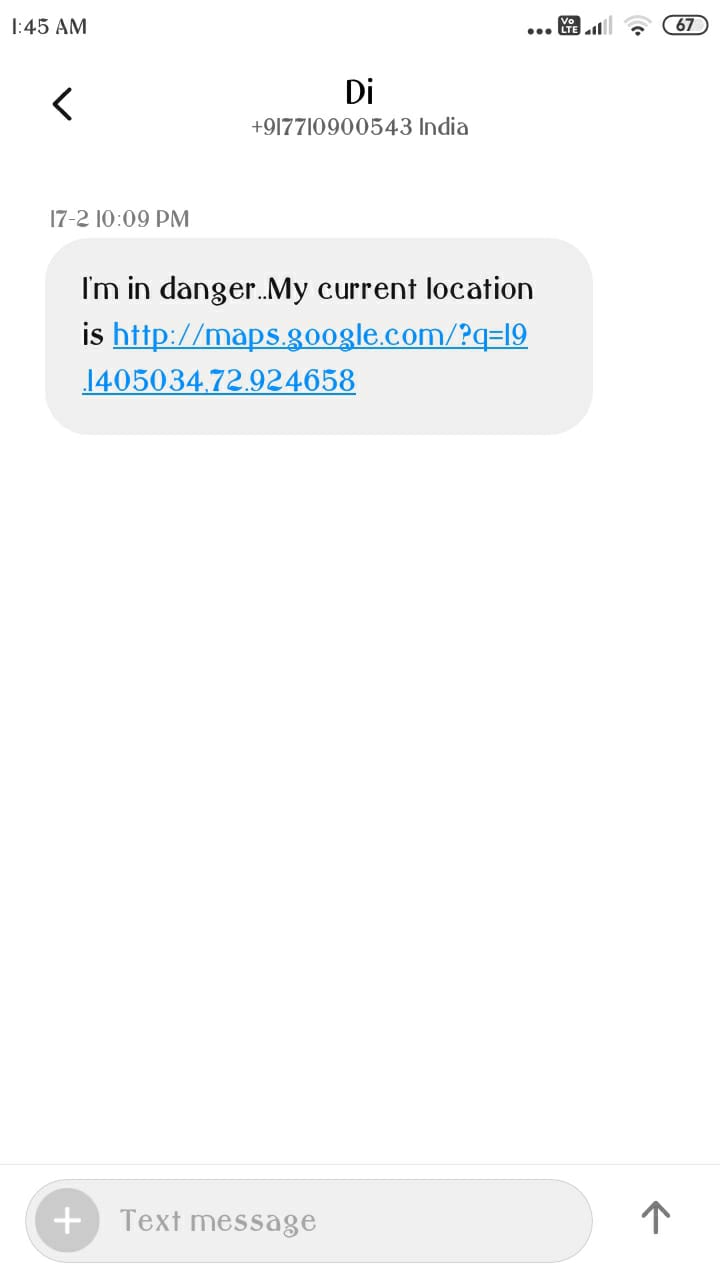
**# OnePlusSafety-App**

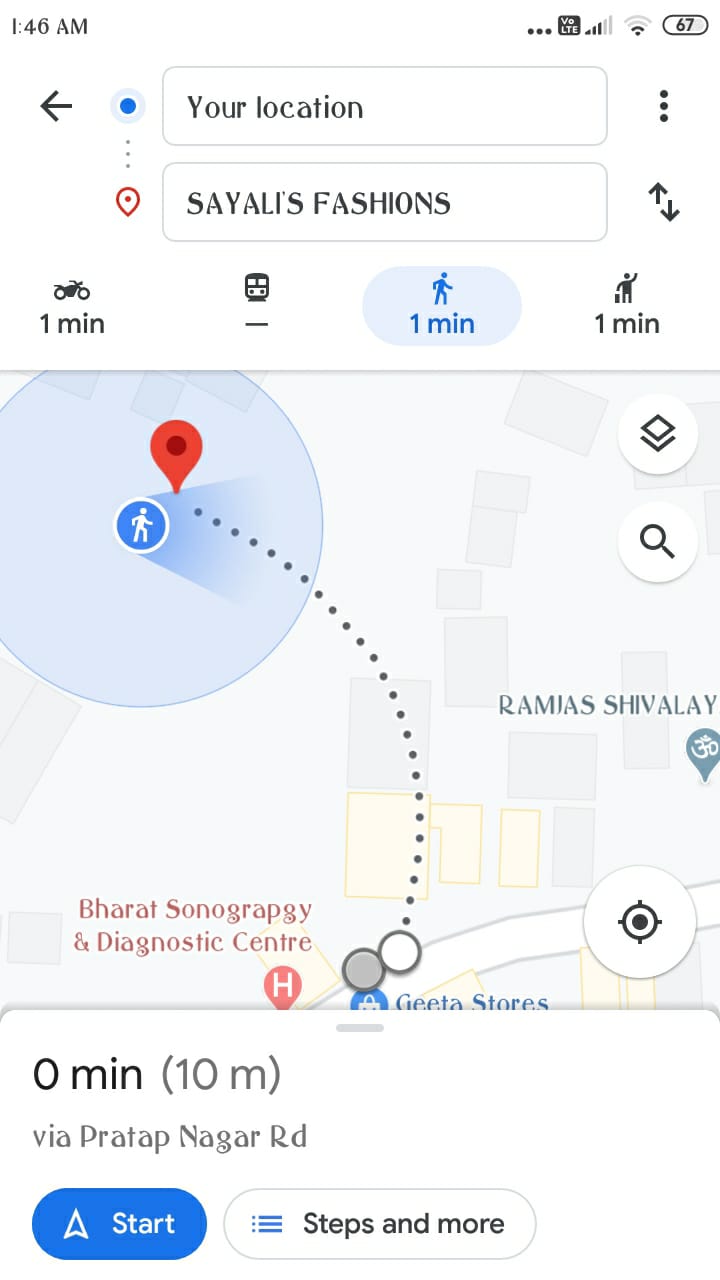
An Android App made for the women safety

**Features of the app**-

**1. SOS/Emergency-**

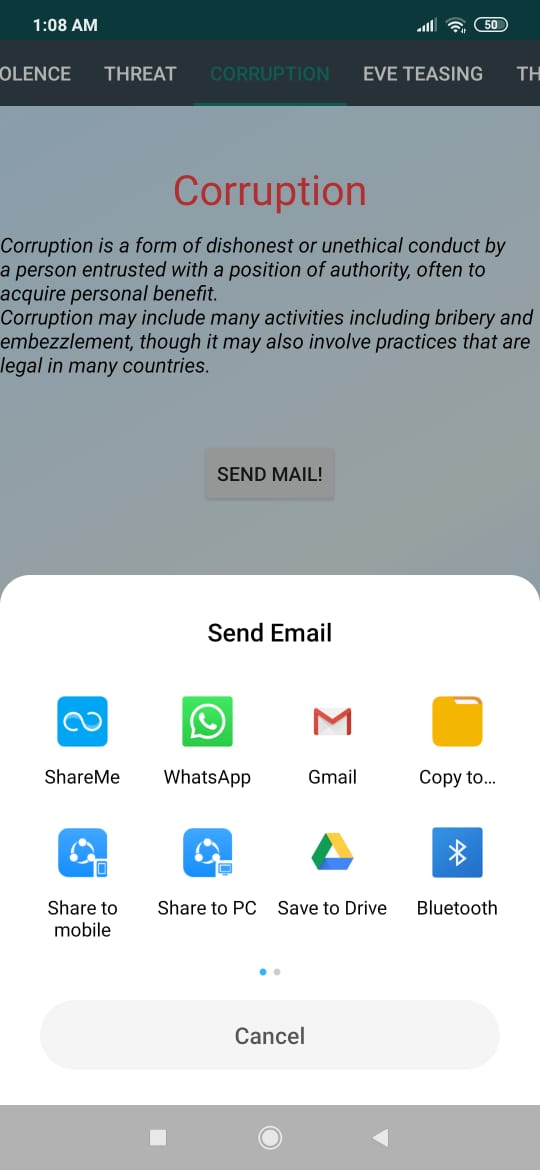
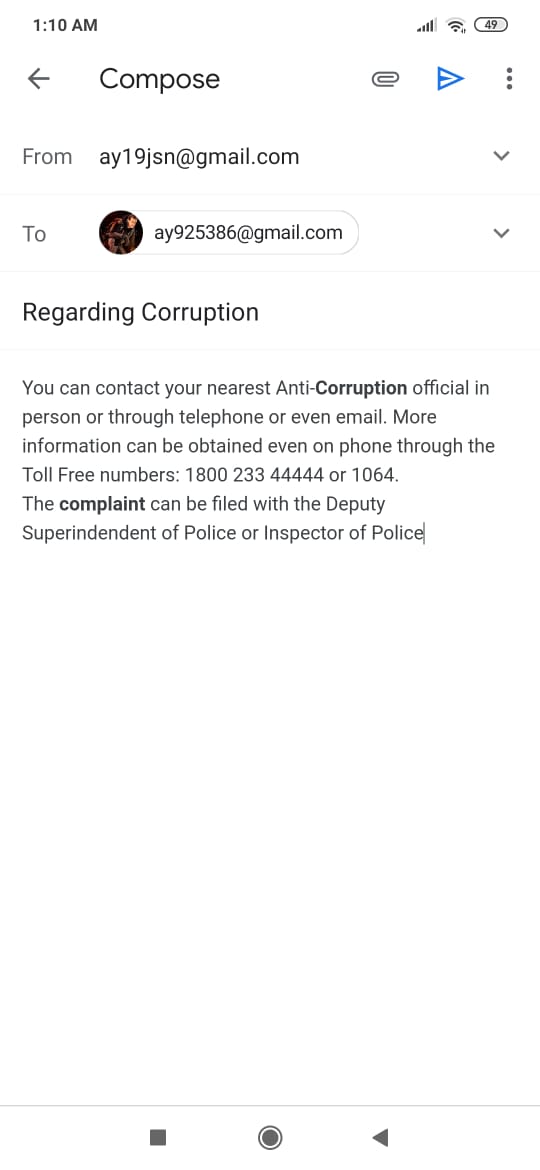
Send a SMS reading- "I am In Danger" to selected 3 contacts and your GPS location to track you with just a click of a button.





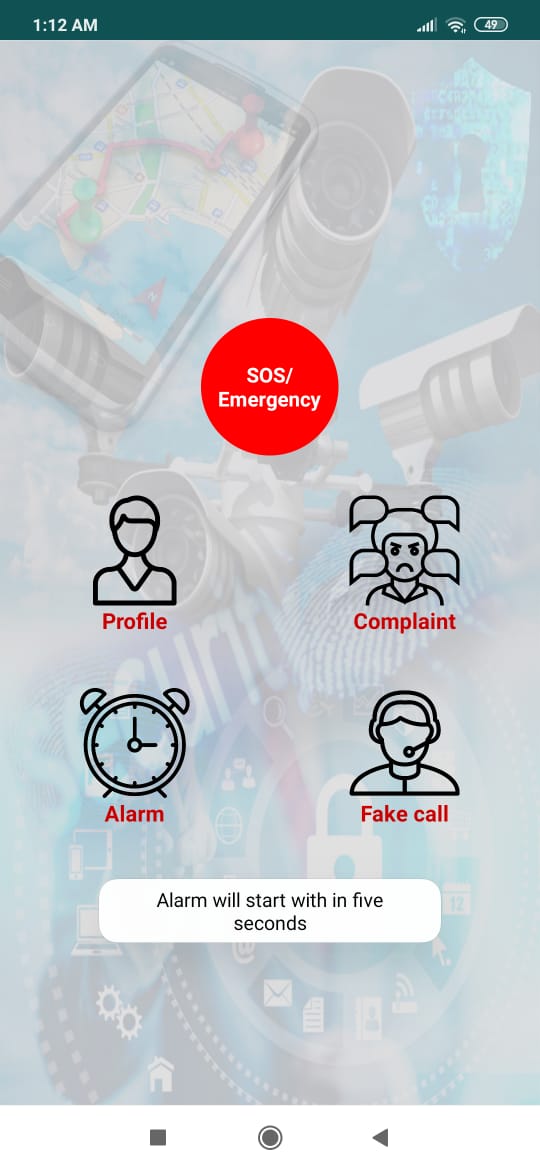
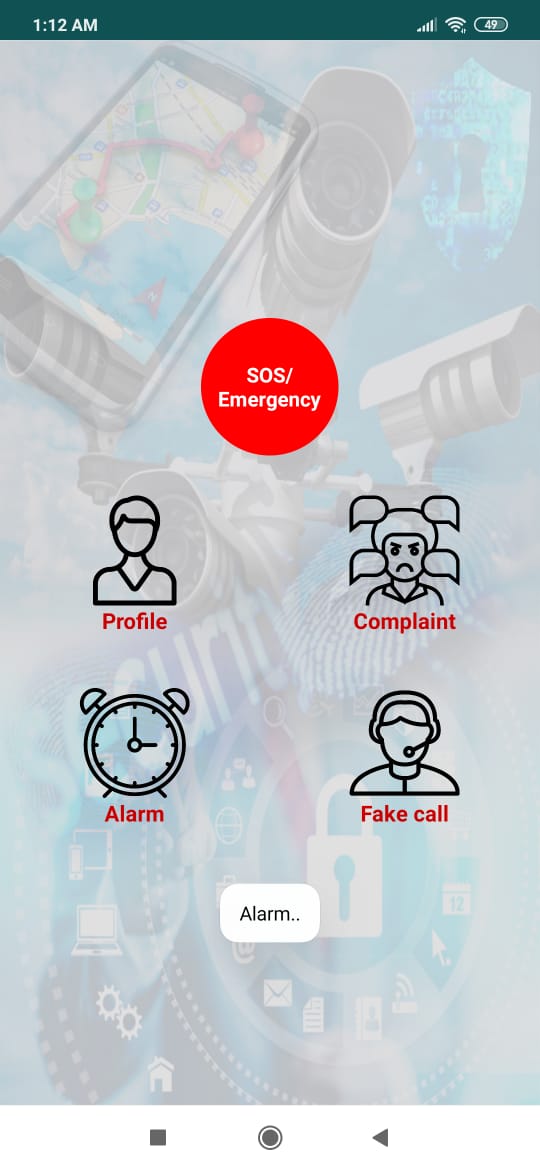
**2. Complaint-**

Instant complaining to the police via E-mail. Complaints can be of any type varying from Domestic Violence to Eve Teasing.



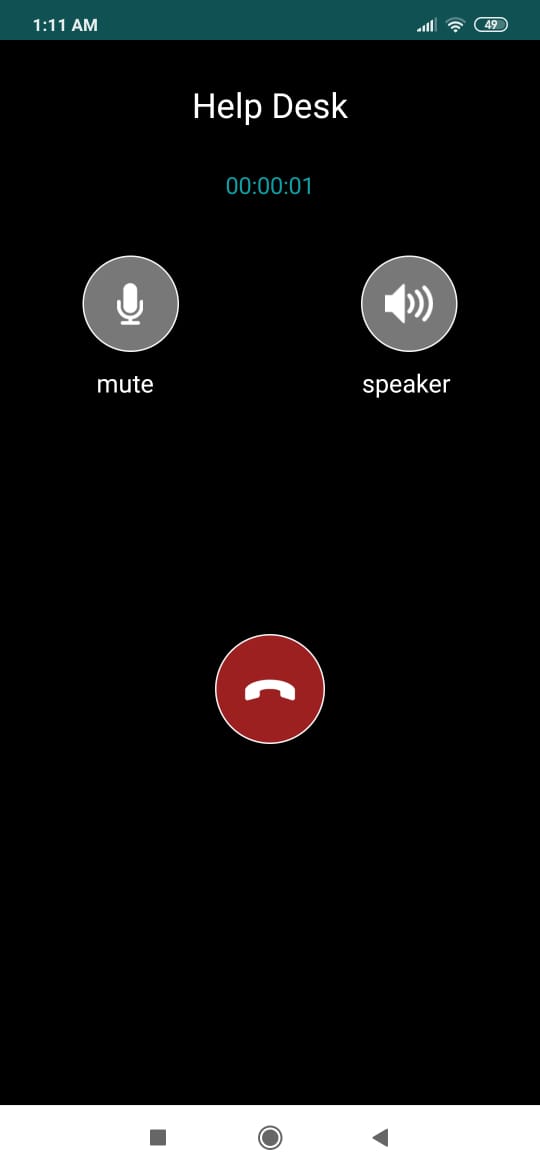
**3. Alarm-**

An alarm (Polics Siren) is set up within 5 seconds. It may help to scare the pursuer.



**4. Fake Call-**

How often, do we think that if we had an incoming call right now, it can save us from the situation we are in right now.

****

## System Testing

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

### Testing the whole system

System testing is performed on the entire system in the context of a

Functional Requirement Specification(s) (FRS) and/or a System Requirement Specification (SRS). System testing tests not only the design, but also the behavior and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software/hardware requirements specification(s).

### Validation

#### Test Cases, Test Results and Test Data

What is Test Case?

“A Test Case has a component that describe an input, action or event expected response, to determine if a feature of an application is working correctly.” Software testing can be stated as the process of validating and verifying that a computer program/application/product:

* Meets the requirements that guided its design and development.
* Works as expected
* Can be implemented with the same characters.
* And satisfies the needs of Stakeholders.

Why we Write Test Case?

A Test Case in Software Engineering is a set of conditions or variables under which a tester will determine whether an application, software system or one of its features is working as it was originally established for it to do. Test Cases bring some sort of standardization and minimize the ad-hoc approach in testing.

#### Test Case :

##### **Case-1**

|  |  |
| --- | --- |
| **Purpose:** | Check for Functionaliy of trigger Button. |
| **Assumptions:** | Application has been launched. |
| **Steps:** | * Click on Trigger button and assure your mobile connection and GPS. * Wait for the output to be generated based on your input in form of Message. |
| **Expected Result:** | Gps based Send a SMS reading- "I am In Danger" to selected 5 contacts and your GPS location to track you with just a click of a button. |
| **Actual Result:** | **Database, GPS (Global Positioning System), GSM (Global System for Mobile)**  based input and output is working fine. |

##### **Case-2**

|  |  |
| --- | --- |
| **Purpose:** | Check for Performance. |
| **Assumptions:** | Application has been launched. |
| **Steps:** | * Choose any option as per choice. **SOS/Emergency**   **Complaint**  **Alarm**  **Fake Call.**   * Wait for the output to be generated in form of GPS location to track you with just a click of a button.   and simultaneously display the output. |
| **Expected Result:** | * Processing time should be less. * Results should be displayed simultaneously. |
| **Actual Result:** | * Processing time is fine. * Results is displayed after Smartphone, Registered contacts,   Database, GPS (Global Positioning System), GSM (GlobalSystem for Mobile) based output. |

**Black Box Testing :**

Black Box Testing, also known as Behavioral Testing, is a software testing method in which the internal structure/ design/ implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional. This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see. This method attempts to find errors in the following categories

1. Incorrect or missing functions

2. Interface errors

3. Errors in data structures or external database access

4. Behavior or performance errors

5. Initialization and termination errors

The testing of Android Mobile application is generally done inside the development tools. Initially emulators are used for this and final testing is done with hardware devices. However, for this application the development and testing is extended out on the specific hardware model. It is essential to have android device embedded in the hardware model.

## Conclusion and Future Enhancement

## Unfortunately the safety in doubt and security is not concerned. Many headlines still coming across against women indicates that increasing trends of such sexual assault, rapes still happening in today's generation. percent the realization of freedom. So we are trying to contribute little efforts towards women which will ensure the safety she also have rights to grow equally like men.

## This mobile application is very much helpful for anyone. GPS the current address will be fetched and send it to any contact depending on user. Here the user can take precautions before coming to the actual danger.

## Limitations

## 1. You can’t send message to any contacts if there is no balance.

## 2. If your mobile is not connected to network , the current location will not be fetch.

## 

## Future Enhancement

* This mobile application is helpful in future when any problem arises in travelling or any kind of situations.
* As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment.
* Because it is based on object-oriented design, any further changes can be easily adaptable.
* Based on the future security issues, security can be improved using emerging technologies

## References

* <http://www.aiktcdspace.org:8080/jspui/bitstream/123456789/2053/1/aiktcDspace2053>
* <https://www.youtube.com/watch?v=14FcNnjixcY>
* <https://www.tutorialspoint.com/android/android_location_based_services.htm>
* <https://www.youtube.com/watch?v=Ak8uRvlpGS0>
* <https://www.youtube.com/watch?v=8FJ3oOpHszc>
* <https://www.youtube.com/watch?v=UDwj5j4tBYg>
* <https://developer.android.com/guide/topics/media-apps/volume-and-earphones>
* <https://stackoverflow.com/questions/11318933/listen-to-volume-changes-events-on-android/30441548>
* <https://www.javatpoint.com/how-to-send-email-in-android-using-intent>