VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi – 590018

A MINI PROJECT REPORT ON

WOMEN'S SAFTEY APPLICATION

Submitted in the partial fulfillment of the requirement for the sixth semester of

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE ENGINEERING

Submitted By

THANUSH.P (1RR20IS043)

Under the Guidance of

Mr. Dinesh Kumar

Asst. Prof Dept. of ISE

DEPARTMENT OF INFORMATION SCIENCE ENGINEERING

RAJARAJESWARI COLLEGE OF ENGINEERING MYSORE ROAD, BANGALORE-560074

(An ISO 9001:2008 Certified Institute)

Accredited by NAAC with 'A+' grade

(2022-23)

RAJARAJESWARI COLLEGE OF ENGINEERING

MYSORE ROAD, BANGALORE-560074

(An ISO 9001:2008 Certified Institute)

Accredited by NAAC with 'A+' grade

(2022-23)

CERTIFICATE

Certified that project report entitled **"WOMEN'S SAFTEY APPLICATION"** carried out by **THANUSH.P (1RR20IS043),** bonafide student of **Rajarajeswari College of Engineering** in partial fulfilment for the award of **Bachelor of Engineering** of **Visvesvaraya Technological University**, during the year **2022-23**. It is certified that all corrections indicated for Internal Assessment have been incorporated in the report deposited in the dept. library. The project report has been approved as it satisfies the academic requirements in respect of the project prescribed for the said degree.

.....	.....
Signature of Guide	Signature of HOD
Mr. Dinesh Kumar	**Dr. J Amutharaj**
Asst. Prof .Dept. of ISE	HOD Dept. of ISE

EXTERNAL VIVA

Examiners	**Signature**
1. _____	_____
2. _____	_____

2.

DECLARATION

I am the student of 6th semester of ISE, Rajarajeswari College of Engineering, solemnly declare that my project work entitled “WOMEN’S SAFETY APPLICATION” is a bonafide work of my. My project is neither a copy nor by means a modification of any other engineering project. I also declare that this project was not entitled for submission to any other university in the past and shall remain the only submission made and will not be submitted by me to any other university in the future.

Submitted by

THANUSH.P (1RR20IS043)

ACKNOWLEDGEMENT

We are truly thankful and convey our sincere gratitude to the principal, **DR. R. BALAKRISHNA**, Rajarajeswari College of Engineering, Bangalore.

We convey our sincere gratitude to **DR J AMUTHARAJ**, HOD, Department of Information Science & Engineering for his meticulous support, continuing, co-operation, valuable suggestion and encouragement during the development of project we also extend our thanks to him for his invaluable guidance to imbibe the requisite knowledge for success of our project.

We convey our sincere gratitude to **MR. DINESH KUMAR**, Assistant Professor, Department of Information Science & Engineering, for his meticulous support continuing cooperation, valuable suggestion and encouragement during the development of project. We also extend our thanks to his for invaluable guidance to imbibe the requisite knowledge for success of our project, as an internal guide. We are very much obliged.

We also thank our parents who have encouraged us and supported in every stage of development of this project.

Last but not the least, our wishes to the entire Information Science Department for their help and guidance, encouragement, inspiration and co-operation at all stages of the development of this project without which this project wouldn't have been a success.

Finally, we express our heartfelt gratitude to all those who helped us to complete the mini project work successfully by providing support, suggestions, advise, guidance and much needed encouragement.

THANUSH.P (1RR20IS043)

ABSTRACT

This mini project aims to develop a mobile Android application focused on women's safety. The application will provide a platform for women to enhance their personal safety and seek immediate assistance in emergency situations. The app will feature various functionalities such as emergency contacts, location tracking, panic buttons, and a distress signal feature. It will also include features like self-defense tips, safety reminders, and a community forum for women to connect and share their experiences. The primary goal of this project is to empower women by providing them with a reliable and user-friendly tool to enhance their safety and create a sense of security in their everyday lives.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Women's safety is a critical concern in today's society. To address this issue, a mobile Android application is developed to offer women a comprehensive safety solution. The application leverages the capabilities of mobile devices, such as GPS, internet connectivity, and user-friendly interfaces, to create a powerful tool for enhancing women's safety.

The application will leverage various features and functionalities to enhance personal safety, encourage prompt response in emergencies, and facilitate communication with trusted contacts and authorities. Our Woman Safety Mobile Application will serve as a comprehensive tool to address the safety concerns faced by women in their daily lives. It will empower them to navigate through potentially risky situations confidently, knowing that they have a reliable safety companion in their pockets. By combining cutting-edge technology with user-friendly design, the application will aim to make personal safety accessible to women from all walks of life.

1.1 Problem definition

In recent times, the safety and security of women have become significant concerns in our society. Women often face various forms of harassment, violence, or unsafe situations while commuting, traveling, or simply going about their daily lives. To combat these issues and empower women, there is a need for a mobile application that can assist them in distressful situations, enhance their personal security, and provide a sense of safety.

1.2 Objective of our Project

The application aims to offer features such as emergency contact alerts, real-time location tracking, and a panic button to immediately notify designated contacts or authorities in case of an emergency. Additionally, it may incorporate features like self-defense tutorials, safety tips, and a community forum for women to share their experiences and support each other. By leveraging technology, the application strives to enhance women's safety, raise awareness about potential dangers, and foster a sense of security within their communities.

CHAPTER 2

REQUIREMENT ANALYSIS

2.1. Introduction to System Analysis:

A system is an orderly group of interdependent components linked together according to a plan to achieve specific objectives. Its main characteristics are organization, interaction, interdependence, integration and central objectives.

Analysis is a detailed study of the various operations by a system and their relationships within and outside. One aspect of analysis is defining the boundaries of the system and determining whether or not a candidate system should consider the related systems. During analysis data are collected on the available files, decision points and transactions handled by the present system. This involves gathering information and using structured tools for analysis.

2.2. Existing System:

Some storage tools is a simple mobile based application which is useful for storing some data within the mobile using another database and these will also help to store respective data in clouds and many more options are available in this type of applications.

Notepad is capable of keeping anything in the storage locally, that is also free and available in any type of operating systems and many org own it and developed it.

2.3 Proposed System:

The main objectives of this application:

★ It saves memory space.

★ It saves user time.

★ Users can directly switch from one interface to another interface in a single application.

And using it may reduce the storage space it will lead us to use one application instead of two which mentioned above. And some notepad will not let us to do some CRUD operations like our Application does and deletion too.

2.4 Feasibility Study:

Feasibility is the determination of whether or not a project is worth doing.This type of study if a project can and should be taken. In the conduct of the feasibility study, the analyst will usually consider seven distinct, but interrelated types of feasibility.The process followed in making this determination is called feasibility study.

2.4.1 Technical Feasibility

This is considered with specifying equipment and software that will successful satisfy the user requirement the technical needs of the system may vary considerably but might include

★ The facility to produce output in a given time.

★ Response time under certain conditions.

★ Ability to process a certain column of transaction at a particular speed.

2.4.2. Economic Feasibility

Economic analysis is the most frequently used technique evaluating the effectiveness of a proposed system. More commonly known as cost or benefit analysis. The procedure is to determine the benefits and savings expected from a proposed system and compare them with costs. It benefits out of the way costs; a decision is taken to design and implement the system will have to be made if it is to have a chance of being approved. There is an ongoing effort that improves in accuracy at each phase of the system life cycle.

CHAPTER 3

SYSTEM REQUIREMENTS AND SPECIFICATIONS

3.1. Introduction

One of the important activities in software development is the preparation of software requirement specification. Since the problems in the modern world are becoming more and more complex, it's becoming increasingly difficult for the developers to comprehend the problems fully and work exactly according to the predicted goal all through the work. Hence the need for a more rigorous requirement analysis is required. In the present time analysis phase is considered to be most critical and difficult.

3.2. Purpose

The aim of the software requirement specification document is to list out the user requirement in an organized manner. It defines all the constraints and software requirements needed to understand this application and documentation. It also has to give an overall design plan. The user should be in a position to incorporate some changes if required. Further be used asthe theme of software development.

3.3. Scope

Software requirement specification is the only written document that describes the requirements of the system. It is meant for the use by the developers and will be the basis for validating the final delivered system. Any changes made to the requirement in the future will have to go through a formal change approval process. The developer is responsible for clarifications, wherever necessary and will not make any alternatives without the permission of the client. It allows the user to analyze the future enhancement in the system.

3.4. Functional Requirements

The name functional requirements specify the statements of services that the system should provide and it performs. It shows how the system should react to particular inputs and behave in particular situations. The functionality of a software system thus defines the 'Functional Requirements'.

3.5. Non – Functional Requirements

These are constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process and standards. Non-functional requirements often apply to the system as a whole. They do not usually just apply individual system features or services.

The types of non-functional requirements are:

➔ PRODUCTION REQUIREMENT:

★ **Maintainability:** Maintainability is the case with which a program/specification can be corrected if an error occurs desires a change in requirements. Specify attributes of software that relate case of maintenance to software itself.

★ **Performance:** Perform specification plays an important part in the analysis of the system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environments, is measured in terms of output provided by the application.

3.6. Organizational Requirements

These requirements are derived from politics and procedures in the customers and developers organization. Examples include process standards that must be used, implantation requirements such as the programming language or design method used, and delivery requirements thatspecify when the product and its documentation are to be delivered.

3.7. Software Requirements

The software requirements specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description as functional representation, a representation of system behavior, an indication of performance requirements and design constraints, and appropriate criteria.

★ **Android Studio**

3.7.1. Android Studio:

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

3.8. Hardware Requirements

★ Android version 5 or higher

★ 2GB or higher RAM

★ Notification turned on

The standard output device can be computer, laptop or mobile. The keyboard, the main input function, should be functional. Apart from these hardware requirements, there should be minimum hard disk space and primary memory available for proper working of the protocol.

CHAPTER 4

SOFTWARE INSTALLATION

★ Head over to the https://developer.android.com/studio website and click on the download button.

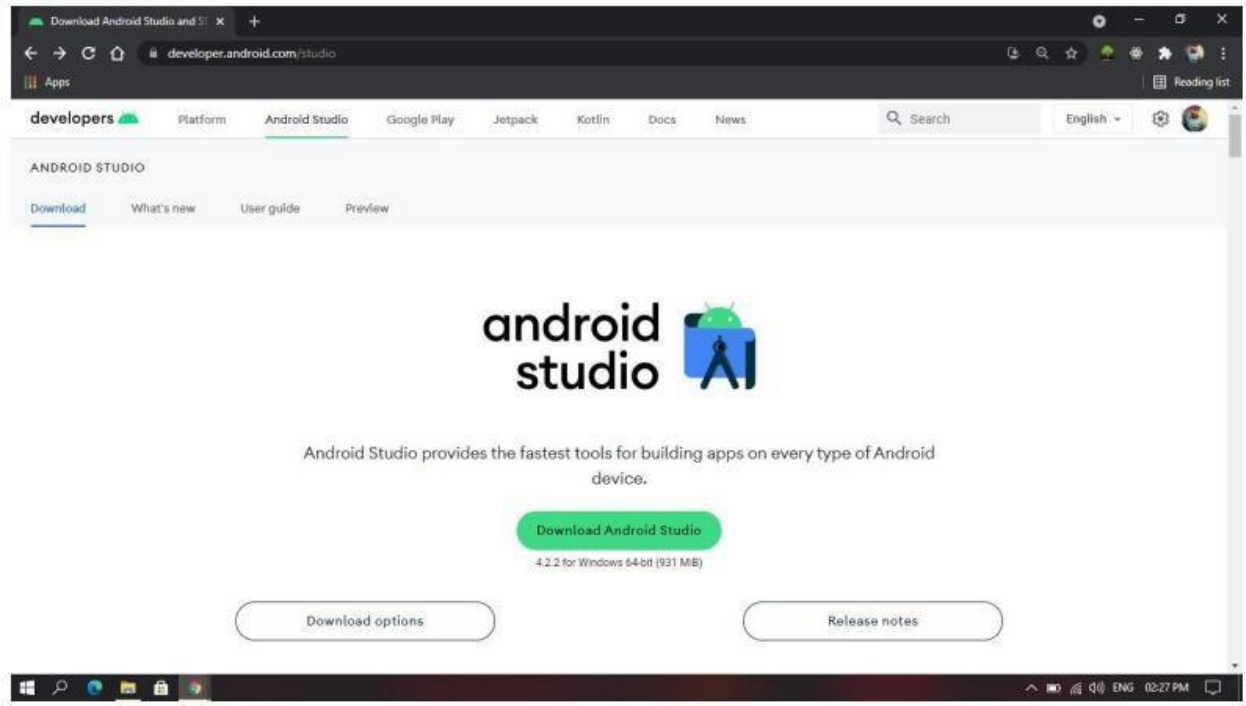


Figure 4.1

To install Android Studio on Windows, proceed as follows:

1. If you downloaded an .exe file (recommended), double-click to launch it. If you downloaded a .zip file, unpack the ZIP, copy the android-studio folder into your Program Files folder, and then open the android-studio > bin folder and launch studio64.exe (for 64-bit machines) or studio.exe (for 32-bit machines).

2. Follow the setup wizard in the Android Studio link as mentioned above and install any SDK packages that it recommends.

★ To start a new project click the software icon in the desktop and create the project by selecting the options you need.

CHAPTER 5

SYSTEM DESIGN

5.1. Design

Data flow design is as shown below - covering the flow of the data in the system. It describes the relation between user input and the system behavior.

5.2. System Architecture

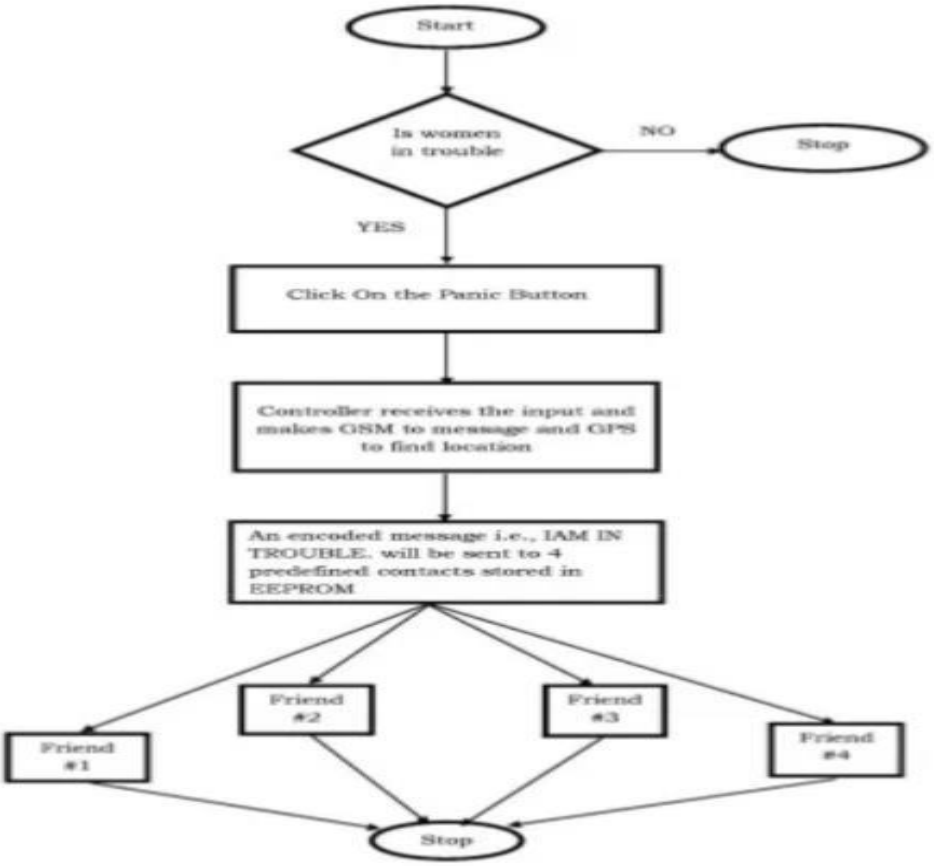


Figure 5.1

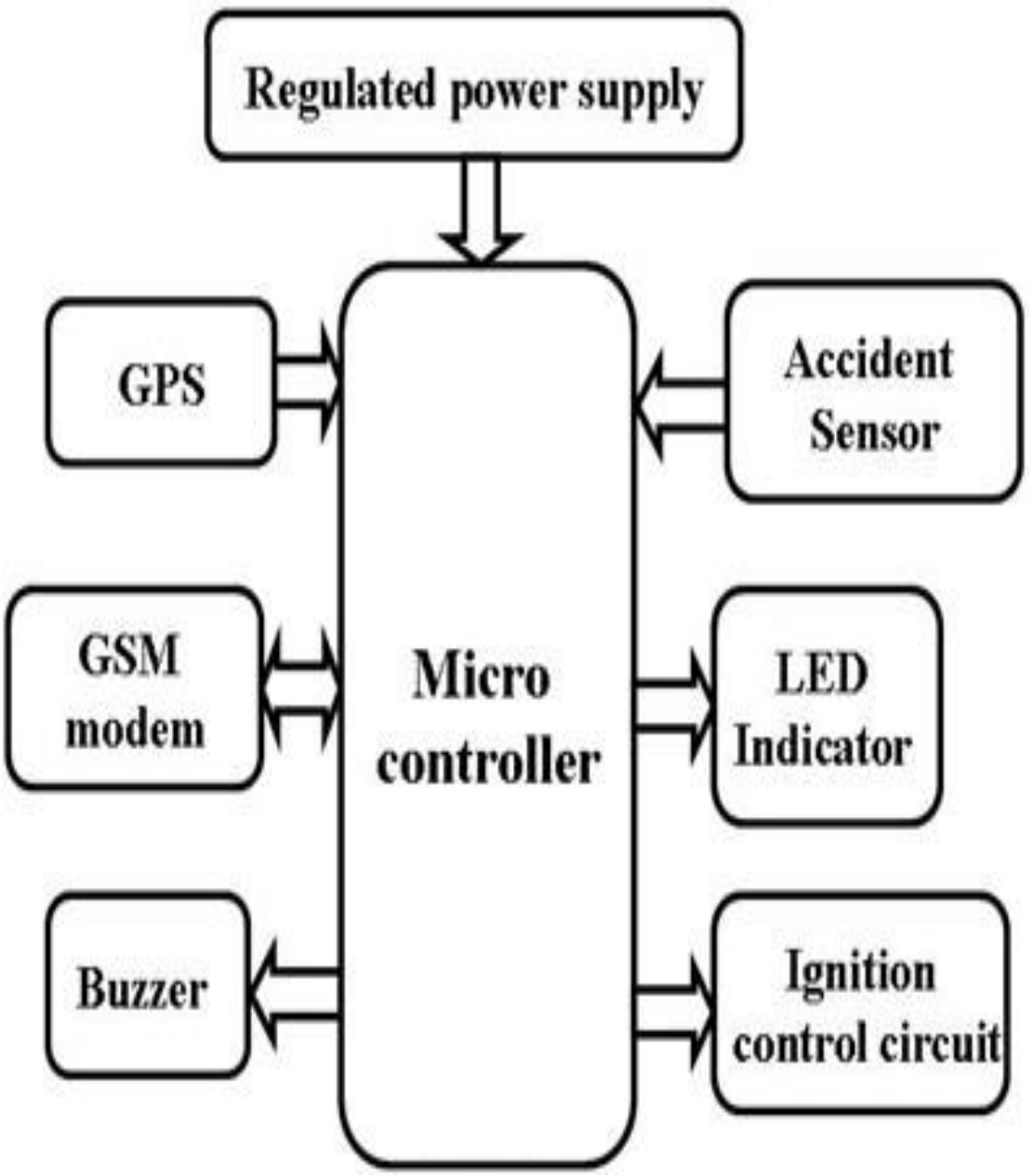

Regulated power supply
GPS
Accident Sensor
GSM modem
Micro controller
LED Indicator
Buzzer
Ignition control circuit

Figure 5.2

CHAPTER 6

IMPLEMENTATION

There is a series that need to follow to build this Women's Saftey App. The implementation steps include:

1: Open your Android Studio and click on Create a Project.

2: Select Empty Activity and proceed.

3: Now, keep a name to your application and select Java and proceed.

4: Now, you need to browse the res —> layout —> open activity_main.xml. You need to define your design as follows.

5: Now open your Main Activity file

6: Now, you need to create two files. One will be the adapter class, and the other is your Modal Class. For this, you can right-click on your package and click on new, and then select Class.

7: Now, you need to code your Modal Class. Below is the code which you can paste into your modal class.

8: Now, you need to create a layout for each item in the recycler view list. So for this, go to res —> design and right-click. Then click new and select Layout Resource File. After this name the file as women_saftey_file.xml

9: Now, you need to add the reminder on your SQLite Database. For this, create a new empty activity called a Reminder.

10: Now, you need to open the WomenActivity.java file.

11: Now, you need to create a Database Manager that will help you to connect your application to the SQLite Database. Using the database manager, you can execute all your SQL queries. For this, create a dbManager.java file.

12: Now, you need to set up the Alarm Manager for your application and register the broadcast receiver. Create a java class called AlarmBroadcast.java.

13: Now, as the last step, you need to edit your Manifest File. Here you need to add some permissions and some declarations.

CHAPTER 7

SCREENSHOTS

Code Snippets:

MapActivityPermissionsDispatcher.java | build.gradle (:app)

Files under the "build" folder are generated and should not be edited.

```
// This file was generated by PermissionsDispatcher. Do not modify!
package com.mad.womensafety;

import ...

final class MapActivityPermissionsDispatcher {
  private static final int REQUEST_GETMYLOCATION = 0;

  private static final String[] PERMISSION_GETMYLOCATION = new String[] {"android.permission.ACCESS_FINE_LOCATION","android.permission.ACCE

  private static final int REQUEST_STARTLOCATIONUPDATES = 1;

  private static final String[] PERMISSION_STARTLOCATIONUPDATES = new String[] {"android.permission.ACCESS_FINE_LOCATION","android.permissi

  private MapActivityPermissionsDispatcher() {
  }

  static void getMyLocationWithPermissionCheck(@NonNull MapActivity target) {
    if (PermissionUtils.hasSelfPermissions(target, PERMISSION_GETMYLOCATION)) {
      target.getMyLocation();
    } else {
      ActivityCompat.requestPermissions(target, PERMISSION_GETMYLOCATION, REQUEST_GETMYLOCATION);
    }
  }

```

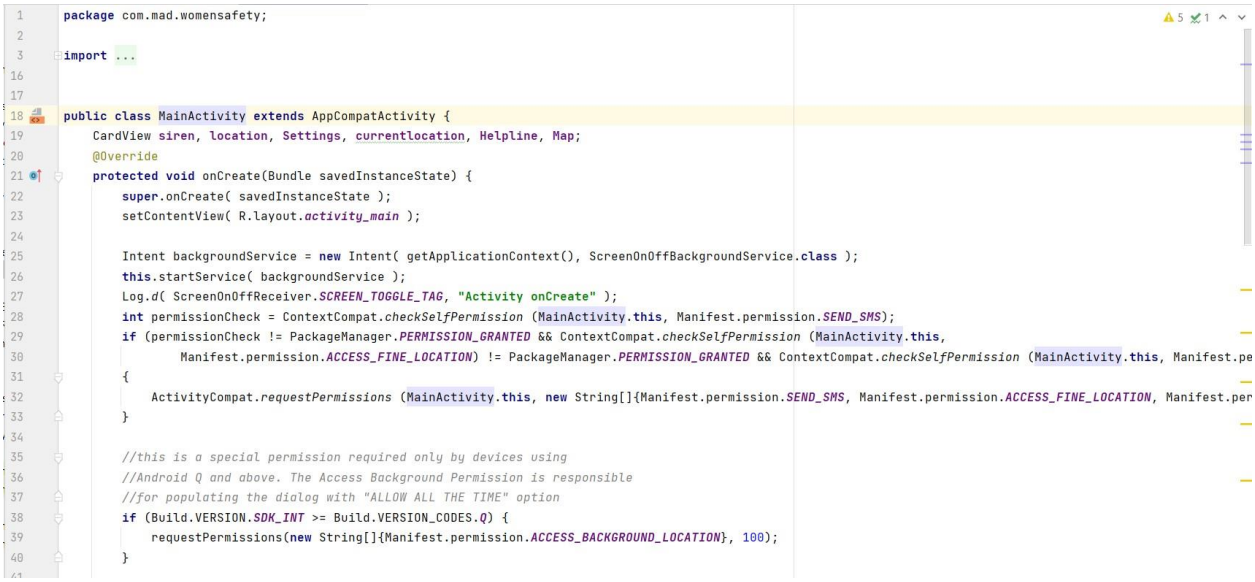
Figure 7.1

```

static void startLocationUpdatesWithPermissionCheck(@NonNull MapActivity target) {
  if (PermissionUtils.hasSelfPermissions(target, PERMISSION_STARTLOCATIONUPDATES)) {
    target.startLocationUpdates();
  } else {
    ActivityCompat.requestPermissions(target, PERMISSION_STARTLOCATIONUPDATES, REQUEST_STARTLOCATIONUPDATES);
  }
}

static void onRequestPermissionsResult(@NonNull MapActivity target, int requestCode,
    int[] grantResults) {
  switch (requestCode) {
    case REQUEST_GETMYLOCATION:
    if (PermissionUtils.verifyPermissions(grantResults)) {
      target.getMyLocation();
    }
    break;
    case REQUEST_STARTLOCATIONUPDATES:
    if (PermissionUtils.verifyPermissions(grantResults)) {
      target.startLocationUpdates();
    }
    break;
    default:
    break;
```

Figure 7.2



```
package com.mad.womensafety;

import ...


public class MainActivity extends AppCompatActivity {
    CardView siren, location, Settings, currentlocation, Helpline, Map;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate( savedInstanceState );
        setContentView( R.layout.activity_main );

        Intent backgroundService = new Intent( getApplicationContext(), ScreenOnOffBackgroundService.class );
        this.startService( backgroundService );
        Log.d( ScreenOnOffReceiver.SCREEN_TOGGLE_TAG, "Activity onCreate" );
        int permissionCheck = ContextCompat.checkSelfPermission (MainActivity.this, Manifest.permission.SEND_SMS);
        if (permissionCheck != PackageManager.PERMISSION_GRANTED && ContextCompat.checkSelfPermission (MainActivity.this,
                Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED && ContextCompat.checkSelfPermission (MainActivity.this, Manifest.pe
        {
            ActivityCompat.requestPermissions (MainActivity.this, new String[]{Manifest.permission.SEND_SMS, Manifest.permission.ACCESS_FINE_LOCATION, Manifest.per
        }

        //this is a special permission required only by devices using
        //Android Q and above. The Access Background Permission is responsible
        //for populating the dialog with "ALLOW ALL THE TIME" option
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q) {
            requestPermissions(new String[]{Manifest.permission.ACCESS_BACKGROUND_LOCATION}, 100);
        }
```

Figure 7.3

```
    Map = findViewById( R.id.map );
    siren.setOnClickListener( new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity( new Intent( getApplicationContext(), Flashing.class ) );
        }
    } );
    location.setOnClickListener( new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity( new Intent( getApplicationContext(), Instructions.class ) );
        }
    } );
    Settings.setOnClickListener( v -> startActivity( new Intent( getApplicationContext(), SmsActivity.class ) ) );
    currentlocation.setOnClickListener( new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity( new Intent( getApplicationContext(), ChoosenActivity.class ) );
        }
    } );
    Helpline.setOnClickListener( new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity( new Intent( getApplicationContext(), HelplineActivity.class ) );
        }
    } );
```

Figure 7.4

Home Page:

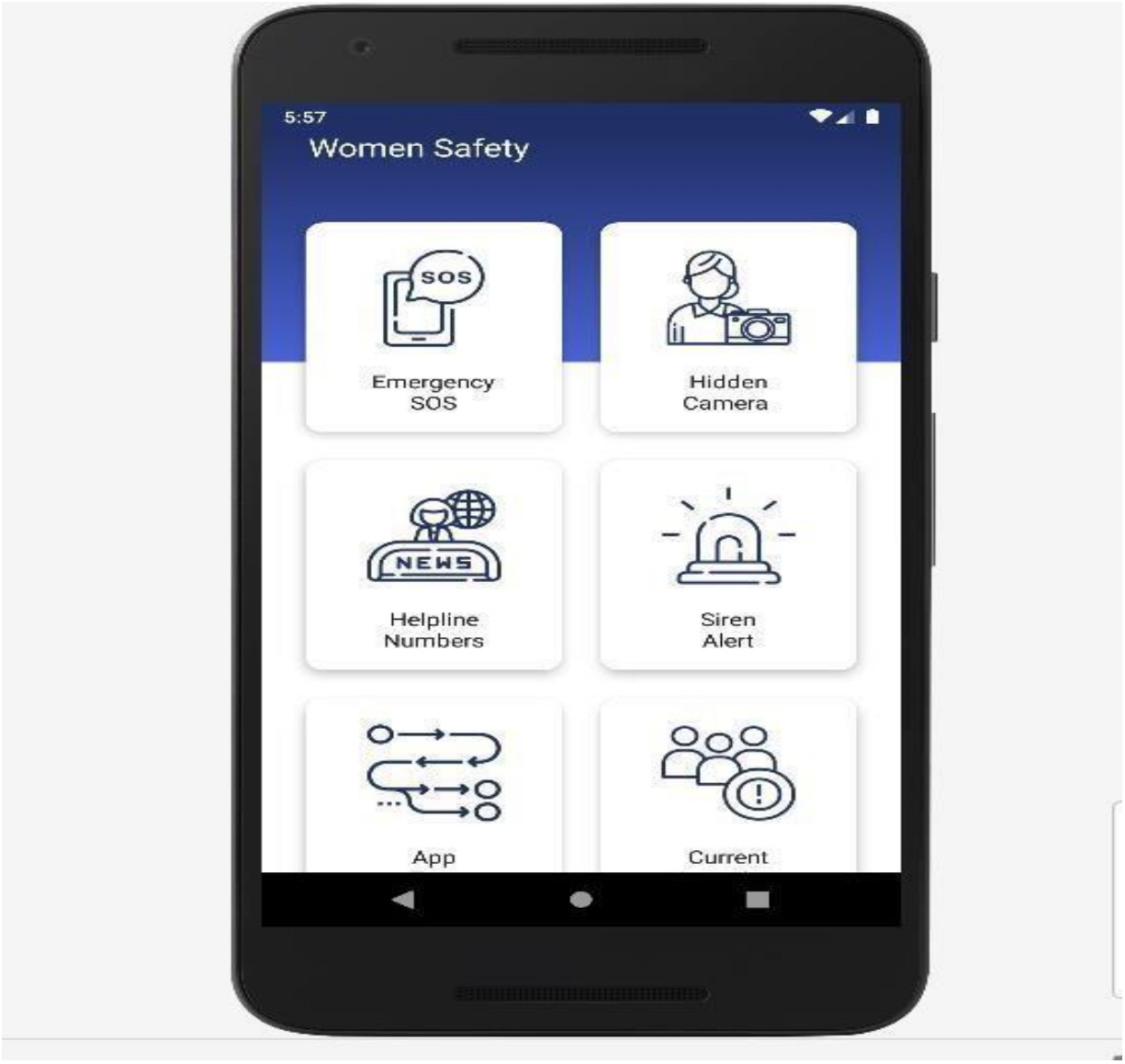


Figure 7.5

Emergency SOS:

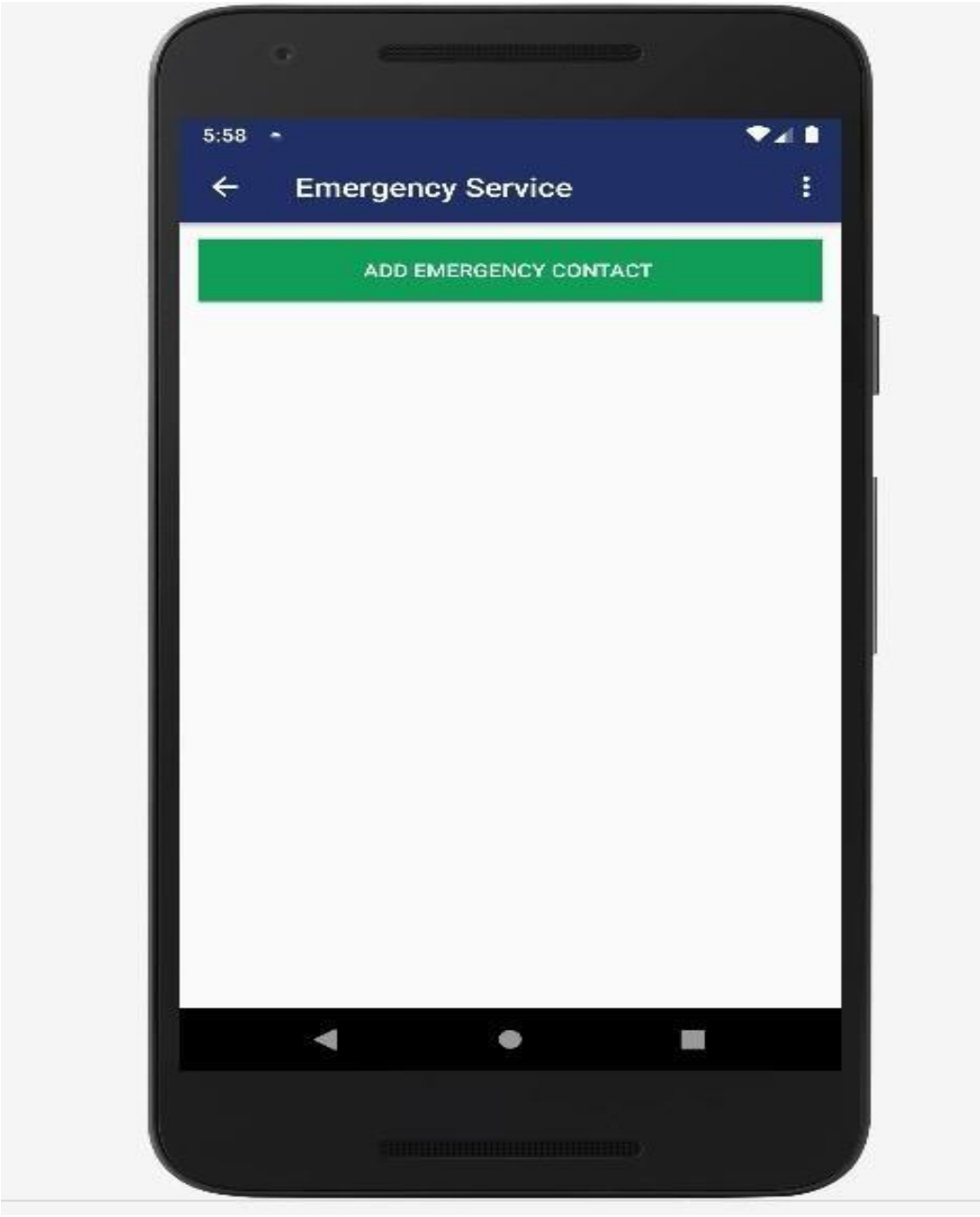


Figure 7.6

Helpline Numbers:

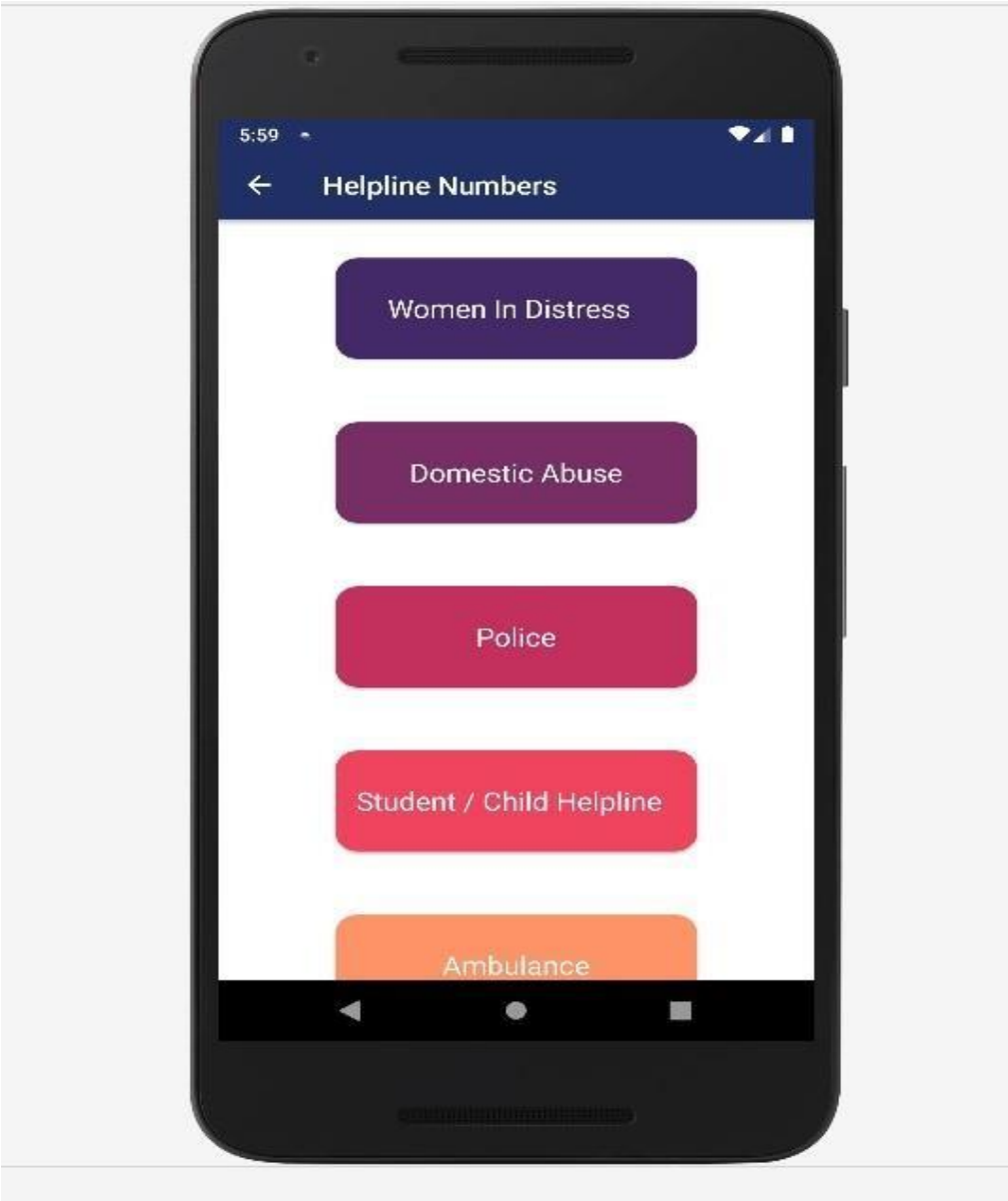


Figure 7.7

Alarm:

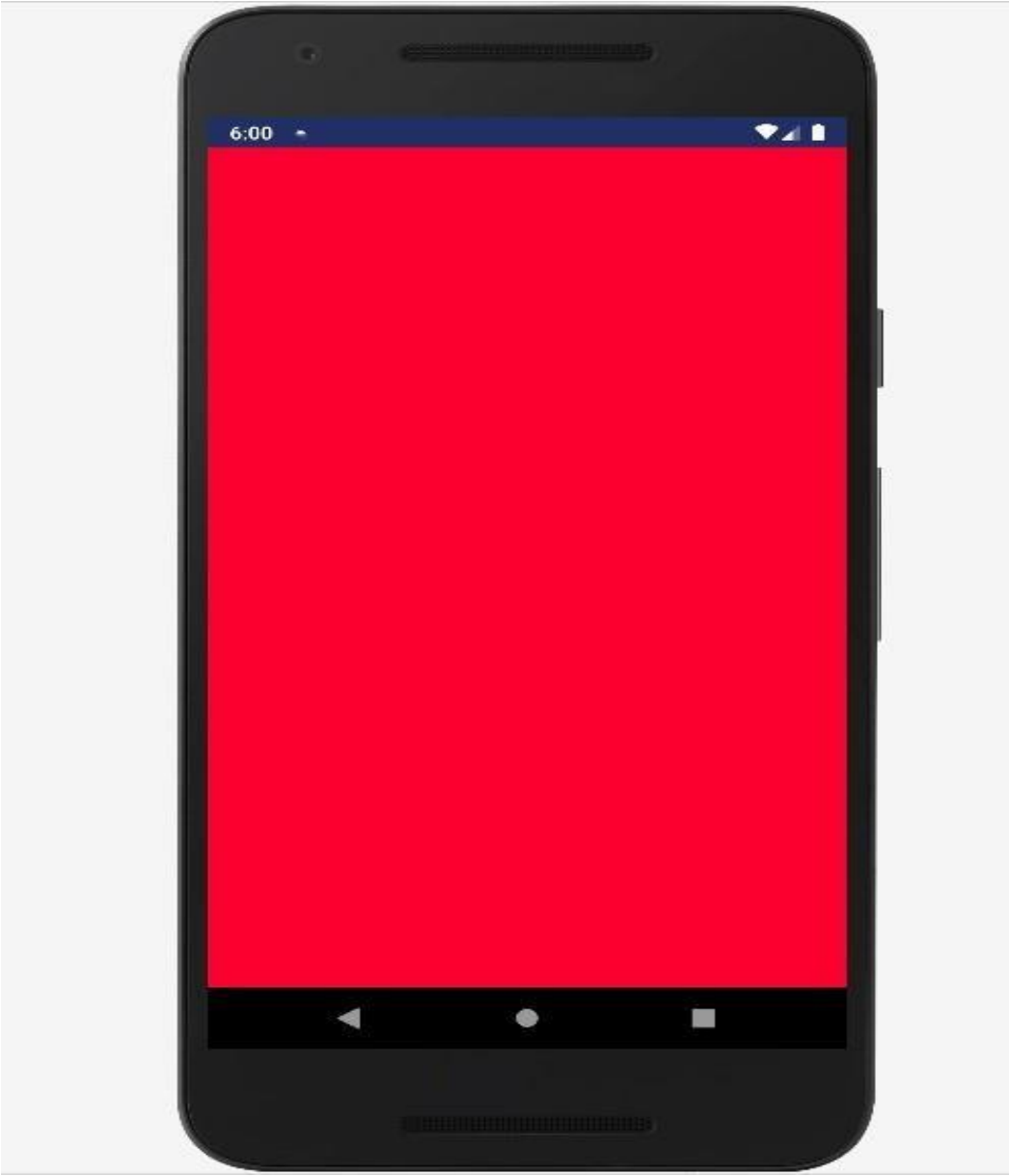


Figure 7.8

App Tour:

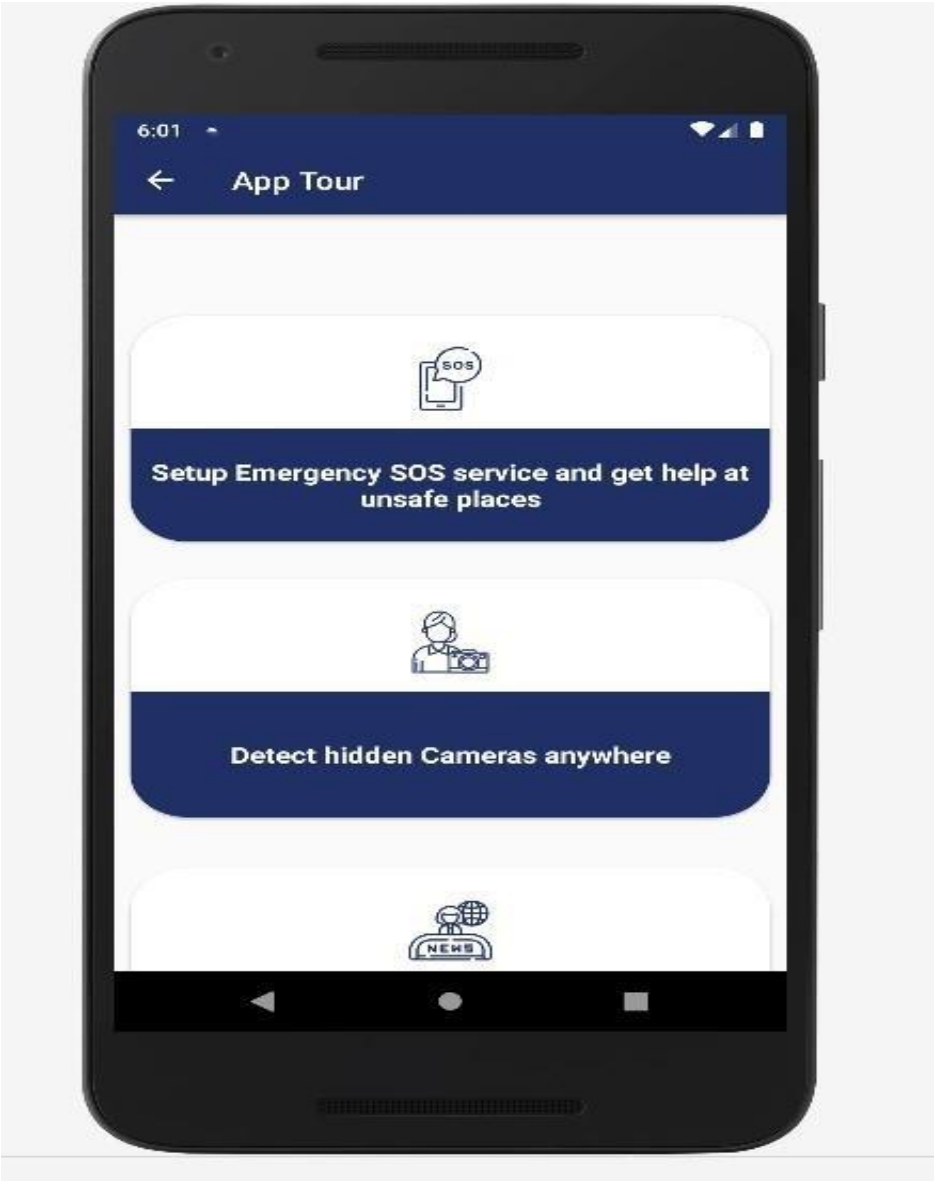


Figure 7.9

Current Location:

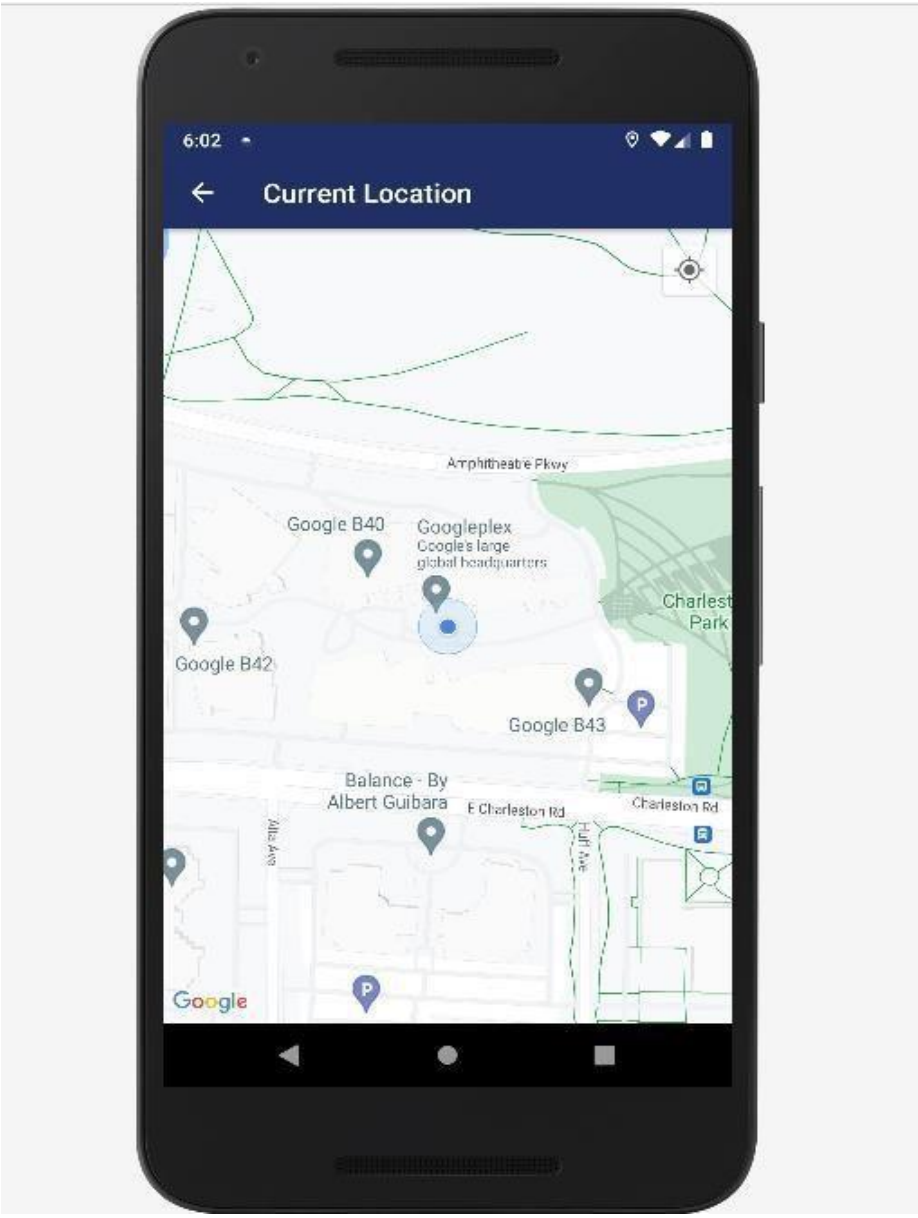


Figure 7.10

Camera:

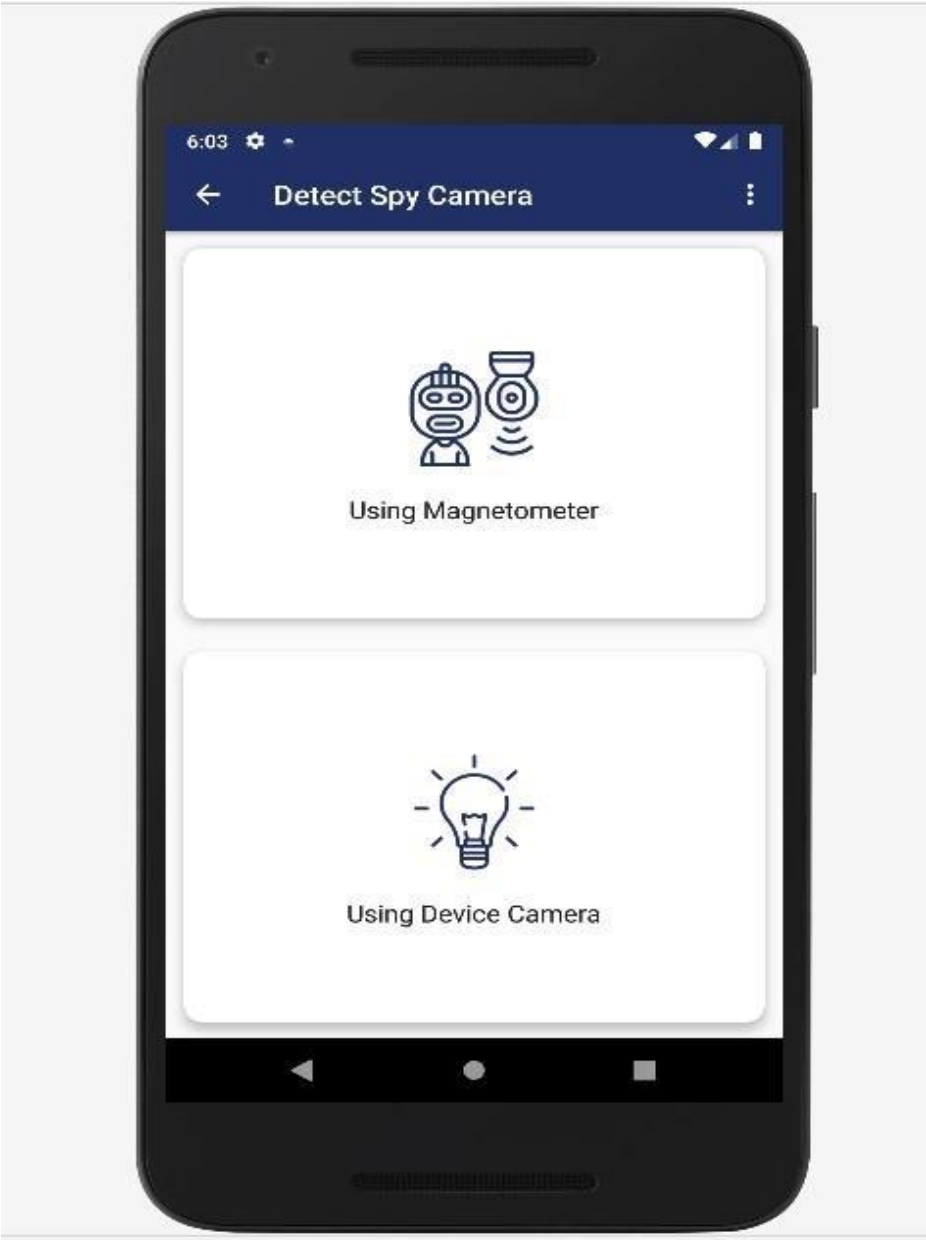


Figure 7.11

CONCLUSION

The development of the project is not an easy process as it involves lot of challenges in different stages of software analysis, design, coding and testing. Having understood the requirements properly and implementing the solutions as per the expectation as brought to the closure of the project. We have tried our best to make this project very realistic, so that the user does not face any trouble when switching over from any real-life android project to this highly useful one.

REFERENCES

[1] https://developers.android.com/index.html

[2] https://www.openhandsetalliance.com/android_overview .html

[3] https://www.android.com

[4] https://www.4shared.com/office/0RX_5-iE/file.html [5]

https://youtube.com