

Day 1: Month January (1st-2nd week)

On the first day, we were informed by the department that the project groups would be formed. We went to college that day, and I, Sk Soyal, along with Arkamita Saha, Tamalika Paria, and Suvajit Das, decided to work together on this project. The four of us selected our project title: "Women Safety Mobile App (My SafetyPin)." It was decided by the department that Madam Aditi Chakrabarty would be our project guide.

Day 2: January (3rd -4th Week)

In the second week of January, we presented our project, "Women Safety Mobile App," outlining six key features: Login, Register, SOS, Online Complaint System, Ambulance Services, and Profile Edit. Our mentor appreciated our effort and positively encouraged us, saying, "You have made a very good decision."

Day 3: Month February (1st-2nd week)

This week all went to college together and drew a Flow Chart based on our pre-determined project. We drew the definition, features and diagrams of the Flow Chart.

Day 4: Month February (3rd-4th week)

This week all went to college together and drew a DFD based on our pre-determined project. We drew the definition, features and diagrams of the DFD.

Day 5: Month March (1st-2nd week)

For implementing the project step by step we draw a basic diagram of our project. At first we implement the front page layout using XML.

Day 6: Month March (3rd week)

We implement demo the of Front page layout. Seeing this, Madam Aditi Chakraborty said that some changes need to be made here. Then we make these changes as per madam's instructions.

Day : 7 Month March- April (4th-1th week)

Over the next two weeks, we began coding the Women Safety Android app's frontend using Figma for design and XML for implementation. Key screens like SOS, location tracking, and emergency contacts were developed for usability. The result was a simple, responsive interface ensuring quick access to essential safety features.

Day : 8 Month April (2nd week)

- In the following two weeks of March, we focused on backend development of the Women Safety Android App using Java in Android Studio. We implemented key features

like SOS alerts, current location find, and background services. Despite challenges with permissions and location accuracy, Java's debugging tools helped ensure a reliable backend foundation.

Day : 9 Month April (3rd Week)

This week, we established database connectivity, linking all data and tables using SQLite. This connection is vital for real-time data insertion, updates, and queries, ensuring efficient data management. It enhances system performance, maintains data integrity, and supports smooth interaction between the application and backend with robust error handling.

Day : 10 Month April (4th Week)

This week, we conducted the first round of testing for our project, encountering both successes and errors. We documented results, fixed issues, and improved for the next round. The collaborative effort our technical skills, teamwork, and communication, providing valuable experience in software development and quality assurance for future projects.

Day : 11 Month May (1st Week)

After testing, we sought our mentor's guidance for code modifications, image insertion, and error checking. She taught us new coding techniques and helped improve our work. With her advice, we completed both frontend and backend coding. After a week of effort, we successfully launched the project, thanks to her support.

Day : 12 Month May (2nd Week)

We corrected all mistakes and took the project to our mentor for review. After confirming everything was fixed, we focused on remaining tasks like printing, formatting tables, and verifying images.

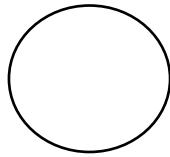
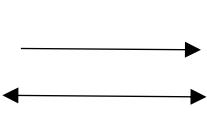
LIST OF ABBREVIATION

The following Abbreviation are used in the dissertations –

<u>Abbreviation</u>	<u>Full Form</u>
GPS	Global Positioning System
SMS	Short Message Service
API	Application Programming Interface
SOS	Save Our Souls / Emergency Signal
XML	Extensible Markup Language
SDK	Software Development Kit
IDE	Integrated Development Environment
DBMS	Database Management System
ML	Machine Learning
AI	Artificial Intelligence
NCIB	National Crime Investigation Bureau
UI	User Interface
SDLC	Software Development Life Cycle
SQLite	Structured Query Language Lightweight or Structured Query Language-Lite

LIST OF COMMON SYMBOLS

The Following Symbols are used in the dissertations –

<u>SL. No</u>	<u>Symbol</u>	<u>Notation</u>	<u>Dissertations</u>
1		External Entity	The source or destination of data outside the system.
2		Process	Step-by-Step instructions are followed that transform inputs into outputs (a computer or person or both doing the work).
3		Data Flow	Communication between an external entity and a process as the process is executing. It work with unidirectional or bidirectional.
4		Data Store	Information stored for a certain period and that is not directly flowing between External Entity.

LIST OF FIGURE

<u>Figure No.</u>	<u>Figure name</u>	<u>Page</u>
1	Opening animation page	65
2	Permission request pages	65
3	Blank signup page	66
4	Fill out the entire signup form and click the “Sign Up” button to create an account	66
5	Blank Login page	66
6	Enter mobile number and password, then click the “Login” button to log in.	66
7	Home Page	67
8	Overview of the features on the home page	67
9	SOS button	67
10	Fetch the current location	67
11	Send Alert SMS to the Guardian	68
12	Send Email to the Women Safety Dept.	68
13	Blank online complaint form	68
14	Fill out the complaint form, then click the submit button. The fetched details are sent to the NCIB	69
15	When a complaint is submitted, an email is automatically sent to the NCIB	69
16	Ambulance Services Home Page	70
17	When the Emergency button is clicked, it fetches the current location and sends SMS to the ambulance service team, who take immediate action.	70
18	Blank Ambulance Form	71
19	Complete the Booking Ambulance form and click “Book Now”; the system will automatically send all the details and current location to the ambulance services team	71
20	When an ambulance booking form is submitted, a SMS is automatically sent to the ambulance service team.	72
21	Profile Page	72
22	Profile Details	72
23	Edit Profile Details	72
24	Blank Add Guardian number page	73
25	Add Guardian phone number	73

INTRODUCTION

Women are adept at mobilizing diverse groups for a common cause. They often work across ethnic, religious, political, and cultural divides to promote peace. We are all aware about importance of safety of women's but we must realize that they should be properly protected. Women's are not as physically strong as men, in an emergency situation a helping hand would be a relief for them. The best way to minimize your chances of becoming a victim of violent crime (robbery, sexual assault, rape, domestic violence) is to identify and call on resources to help you out of dangerous situations. Whether you're in immediate trouble or get separated from friends during a night out and don't know how to get home, having these apps on your phone can reduce your risk and bring assistance when you need it. Although several were originally developed for students to reduce the risk of sexual assault on campus, they are suitable for all women in the light of recent outrage in India which shook the nation and woke us to the safety issues for our daughters, people are gearing up in different ways to fight back. A host of new apps have been developed to provide security systems to women on their phones.

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 user can activate this service by adding the emergency contacts using the emergency contacts icon in the app. While in emergency the user would have to shake up his/her handset, after that a distress signal (SOS) will automatically got generated from the user end and send SMS to those contacts which has been saved at the time of registration and updation. The SMS contains your default help message and your exact location.

AIMS / BENEFITS OF THE PROJECT

This system is for women's safety and overcomes existing systems. It consists of a GPS device, an Android phone. The unit will provide status information such as latitude and longitude of the user. The proposed App is based on advanced sensors. A user makes a phone call, sent alert to the contacts that have been added to the emergency call and send email to the women safety dept. The proposed system will be implemented with the help of android application. Which will alert the nearby people who having this application by sending alert messages to the guardian mobile on shaking of victim mobile. Also sends messages to the saved contacts in the application and police station. Which also show the location of the victim with the help of current location. Which also make message in guardian mobile when his/her mobile in silent mode.

In today's world, people using smart phones have increased rapidly and hence, a smart phone can be used efficiently for personal security or various other protection purposes. The heinous incident that outraged the entire nation have waken us to go for the safety issues and so a host of new apps have been developed to provide security systems to women via their phones. This paper presents women security an Android Application for the Safety of Women and this app can be activated this app by a shaking the mobile, whenever need arises. This app identifies the location of place through GPS and sends a message comprising this location URL to the registered contacts and also send messages to near by mobile which are having this app.

MOTIVATION OF THE PROJECT

In today's world, it is not safe for a person to travel alone at night especially for women; it will be high time to travel alone because a woman is not highly strong as men to protect herself from them. The good way to reduce chances in becoming a victim of violent crime (robbery, sexual assault, rape, domestic violence) is to identify and call on resources to help you out of unsafe situations. Whether you are in instant trouble or got separated from friends during night and do not know to reach home. having these apps on your phone can diminish our risk and bring assistance when we require it. In this paper, we present Security Alert an application for smart phones working over android platform.

LITERATURE REVIEW

As a part of literature survey, we investigated some applications of women safety that already exist in market. The aim is to observe how these applications work and to see how they can be improved and how are they different to date it is identified that the following Android Apps of women security are good and are offering relatively similar service.

2.1.1 Existing research or related work.

- **Personal Safety Apps for Women:**
 - **bSafe** : This is a popular safety app that allows users to send alerts with their location, activate an SOS feature, and share their real-time location with friends and family.
 - **Life360** : Primarily a family locator app, Life360 allows users to stay connected with family members and friends. Women often use it for tracking each other's locations in real-time for safety purposes.
 - **Guardly** : A safety app that connects users directly to emergency contacts and local authorities in case of danger. It uses features like geolocation and direct communication to ensure that help is sent immediately.
- **Emergency Alert Systems:**
 - **Panic Button Apps** : Apps that offer a one-tap emergency feature, which immediately sends an alert to designated contacts or emergency services. These types of apps are critical for situations where the user cannot manually make a call.
Ex – MySafetipin.
- **Geolocation and Tracking**
 - **Find My Friends** : Apple's built-in geolocation app is widely used, but similar solutions exist for Android users. These apps enable women to share their locations with trusted friends and family members in real-time providing an added layer of security.
 - **Google Maps Location Sharing** : Google Maps offers a location-sharing feature that allows users to share their location in real-time with selected contacts.
- **Preventive Measures**
 - **Self-Defense and Safety Tips** : This app provides a customizable panic button feature that sends emergency messages and location information to predefined contacts, helping users take preventive action before a situation escalates.

2.1.2 Limitations Of Existing System

Despite their benefits, existing women's safety apps on Android face several limitations that hinder their effectiveness in real-world emergency situations. These limitations can be broadly categorized into the following areas :

- **Geolocation Accuracy** : While most safety apps rely on GPS for real-time location tracking, GPS signals can be unreliable in certain environments, such as urban canyons, indoor spaces, or rural regions with poor satellite reception. This can lead to inaccurate or delayed location sharing, hindering timely responses in emergencies.
- **Battery Drain** : Continuous tracking of a user's location can drain the phone's battery quickly. In a situation where a woman's phone is low on battery or dies during an emergency, the app's primary function.
- **Delayed Response from Authorities** : Although many safety apps can connect users with emergency services, the response time can vary greatly.
- **Data Consumption** : Continuous location tracking and background application app activity significantly drain the device's battery and use mobile data.
- **Poor Internet Connectivity Issues**: Many apps require a stable internet connection to send alerts or share locations. IN low-network areas or during outages, the apps may fail when needed the most.
- **Privacy and Data Security Concerns** : Constant location sharing can lead to misuse or stalking if data is not well protected. Some apps may store or share user data without transparent policies, raising privacy issues.
- **Cost of Premium Features** : Some safety apps offer a free version with limited features, but essential functionalities, such as access to real time location sharing or emergency response features, may require users to pay for a premium version. This can exclude women who cannot afford the subscription fee.
- **Complex User Interface** : Many safety apps are designed with a range of features, but the user interface (UI) may be too complex.

2.1.3 Justification for the chosen approach

- Our chosen approach focuses on three core and essential features: sending a text message, making an emergency call, and sharing location via Google Maps. These features were selected based on their reliability, simplicity, and effectiveness in real-world emergency scenarios, particularly for users in both urban and rural areas.
- **Simplicity and Reliability**
 - In high-stress or emergency situations, users may not have time or clarity to navigate complex features. Our application focuses on essential actions that can be performed quickly and with minimal effort. By limiting the scope to texting, calling, and GPS location sharing, we ensure:
 - **Faster response times**
 - **Less confusion during emergencies**
 - **Lower chance of app malfunction or user error**
- **Offline and Low Connectivity Support**
 - Many advanced safety apps rely heavily on internet connectivity, which may not always be available — especially in rural or remote areas. By using SMS and phone calls, our app remains functional even without an active internet connection, making it more dependable in real-world situations.
- **Critical First Response Tools**
 - The three features we included are aligned with what users need most in the first few seconds of a threat:
 - **Text Message:** Sends a pre-set emergency alert along with the user's current location to trusted contacts.
 - **Call Function:** Directly dials a saved emergency number (like a family member or police) for immediate voice communication.
 - **Google Maps Integration:** Helps recipients track the user's real-time location, aiding in faster rescue or support.
 - These actions cover awareness, communication, and location tracking, which are the three pillars of personal safety response.
- **User Accessibility and Speed**
 - The app is designed for ease of use, especially for users who may not be tech-savvy. With a minimal UI, large buttons, and optional shortcut triggers (like tapping a widget or shortcut), the app allows for quick action under stress.

▪ **Resource Efficiency**

- By keeping the feature set focused, the app uses less battery, less storage, and fewer background processes, ensuring it remains lightweight and efficient — ideal for low-end Android devices.

▪ **Scalability**

- While the current version includes core features, the clean and modular design allows for future enhancements, such as automatic triggers, voice activation, or cloud-based emergency alerts — without rebuilding the app from scratch.

SYSTEM ANALYSIS

3.1.1 Hardware and Software Requirements

❖ Software Requirements -

▪ Minimum Software Requirement

- To build, run, and maintain the Women Safety Android Application project efficiently, the following software components are required. These tools support both front-end and back-end development, as well as database management and testing.

▪ Operating System (OS)

- Minimum - 64-bit Windows 10
- macOS 12
- Any 64-bit Linux distribution that support android Studio(IDE)

❖ Hardware Requirements -

▪ Minimum Hardware Requirements (For Developers Only)

- To implement and run this project efficiently, certain minimum hardware specifications are required. These requirements are based on factors such as system complexity, expected user load, and the technologies used in the project

1. Processor (CPU)

- The performance of the CPU plays a key role in handling development tasks and running the project smoothly.
- Type: Intel Core i5 or equivalent
- Speed: 2.0 GHz or higher
- Cores: Quad-core or above

2. Memory (RAM)

- The amount of RAM affects the system's ability to manage multiple processes and tools during development.
- Required: 8 GB For Studio & 16 GB For Emulator
(Handling development tasks and running the project smoothly)

3. Graphics Card (GPU)

- GPU requirements depend on the visual aspects and design complexity of the project.

- Recommended Options:
 - Nvidia Geforce 20 series or newer
 - AMD Radeon or newer with the latest drivers.

4. Storage (Hard Disk)

- Storage needs are determined by the size of the project files, resources, and local databases.
- Minimum Required Space:
 - Studio - 10 GB of free space. (approximate)
 - Emulator - 20 GB of free space. (approximate)

3.1.2 Concepts Used in the Proposed System

❖ Android Studio –

Android Studio is the official Integrated Development Environment (IDE) for android application development. Android Studio provides more features that enhance our productivity while building Android apps

Features of Android Studio -

- It has a flexible Gradle-based build system.
- It has a fast and feature-rich emulator for app testing.
- Android Studio has a consolidated environment where we can develop for all Android devices.
- Apply changes to the resource code of our running app without restarting the app.
- Android Studio provides extensive testing tools and frameworks.
- It provides build-in supports for Google Cloud Platform. It makes it easy to integrate Google Cloud Messaging and App Engine

❖ XML

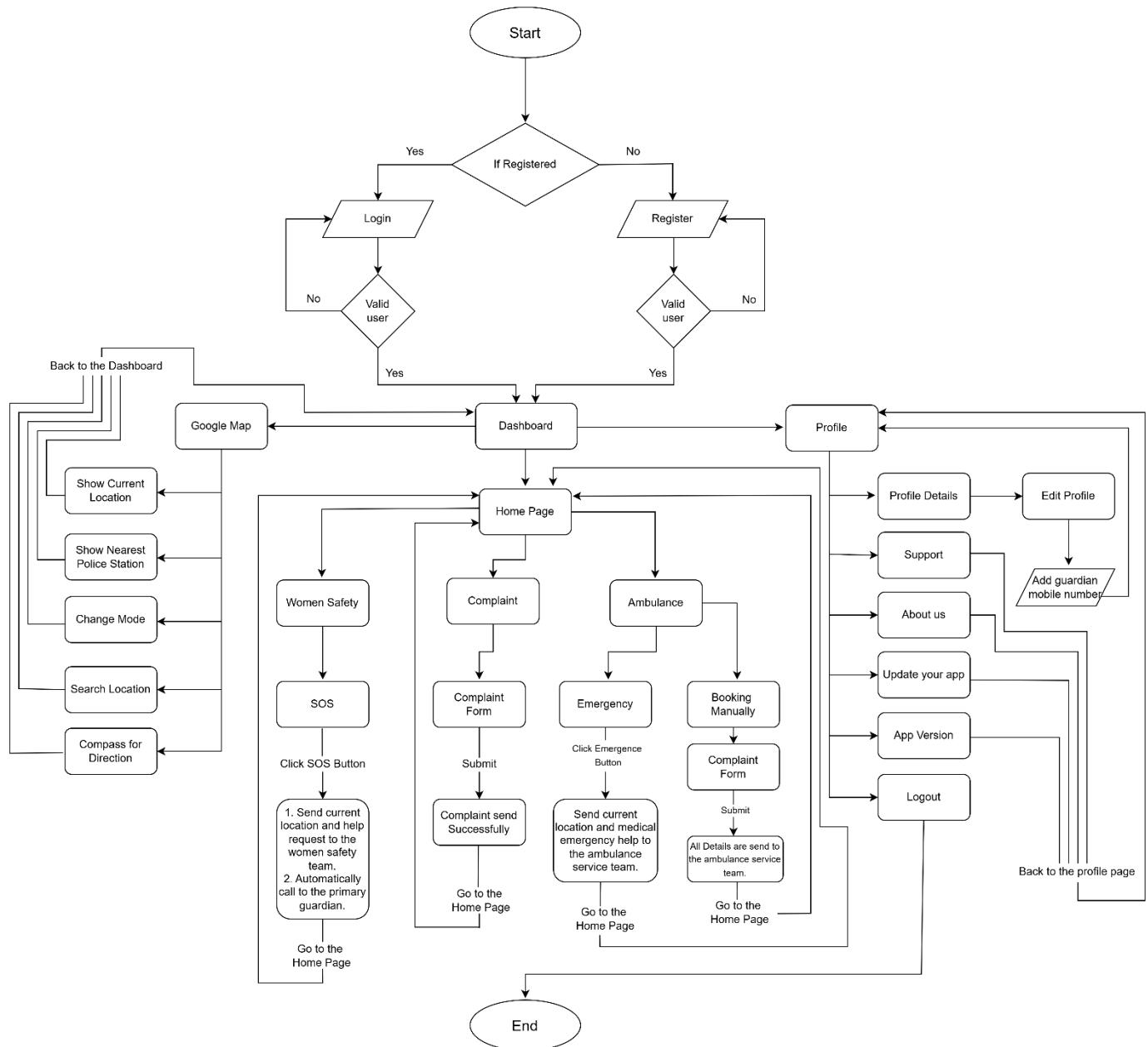
- XML stands for Extensible Markup Language.
- XML is a markup language much like HTML used to describe data.
- XML tags are not predefined in XML. We must define our own Tags.
- XML as itself is well readable both by human and machine. Also, it is scalable and simple to develop.
- In Android we use XML for designing our layouts because XML is lightweight language so it doesn't make our layout heavy.

❖ JAVA

- Java plays an important role in development of Android applications because business logic is written in Java.
- You can say that knowledge of core Java is must for the development of android application. Knowledge of advance Java is a plus point for the development. With the knowledge of advance Java, you can add new features to the application.

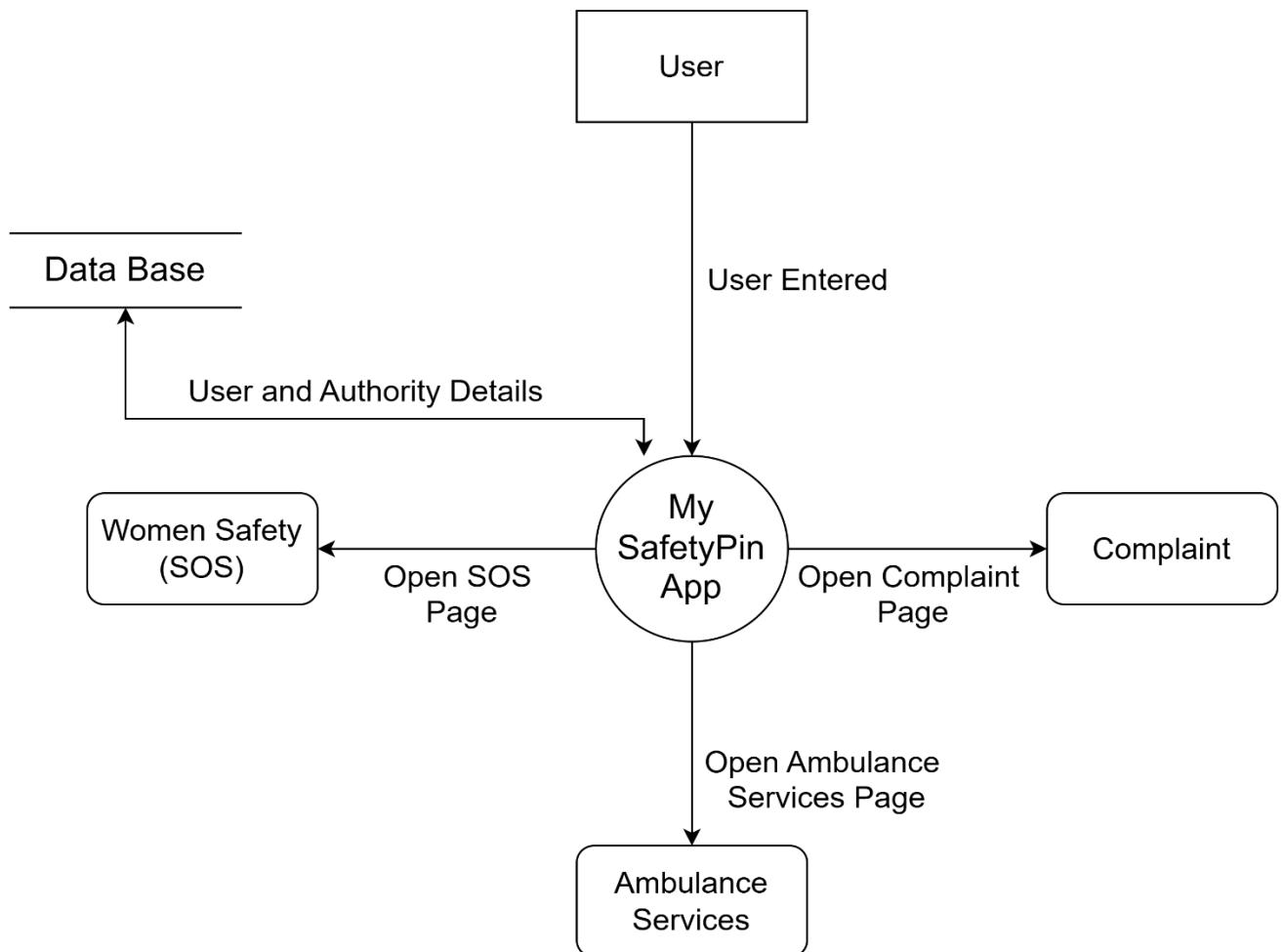
SYSTEM DESIGN

4.1.1 Flow Chart

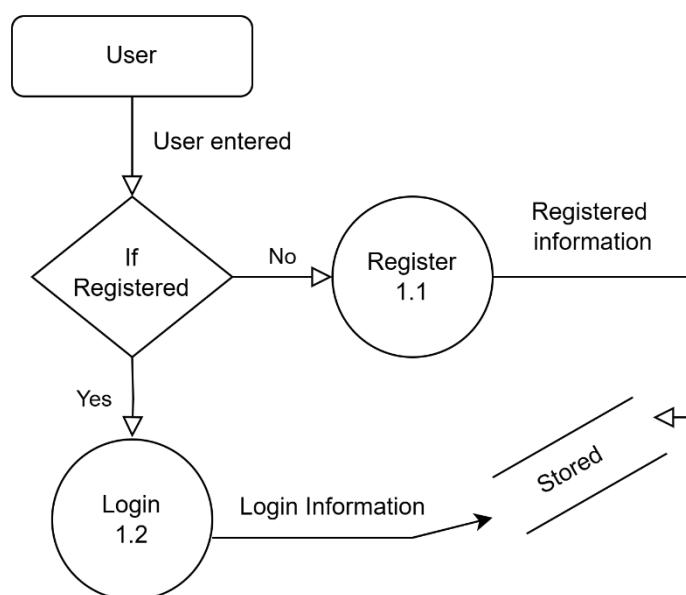


4.1.2 Data Flow Diagrams

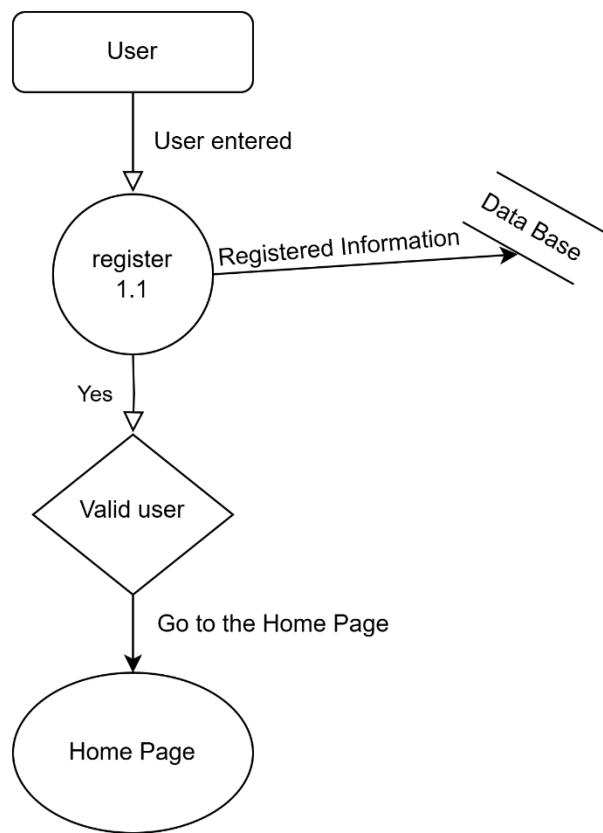
■ Level 0



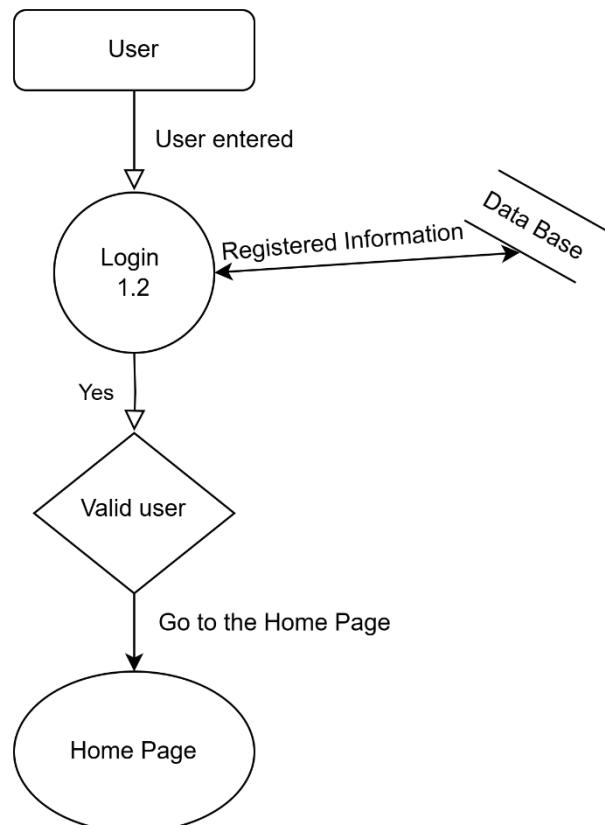
■ Level 1



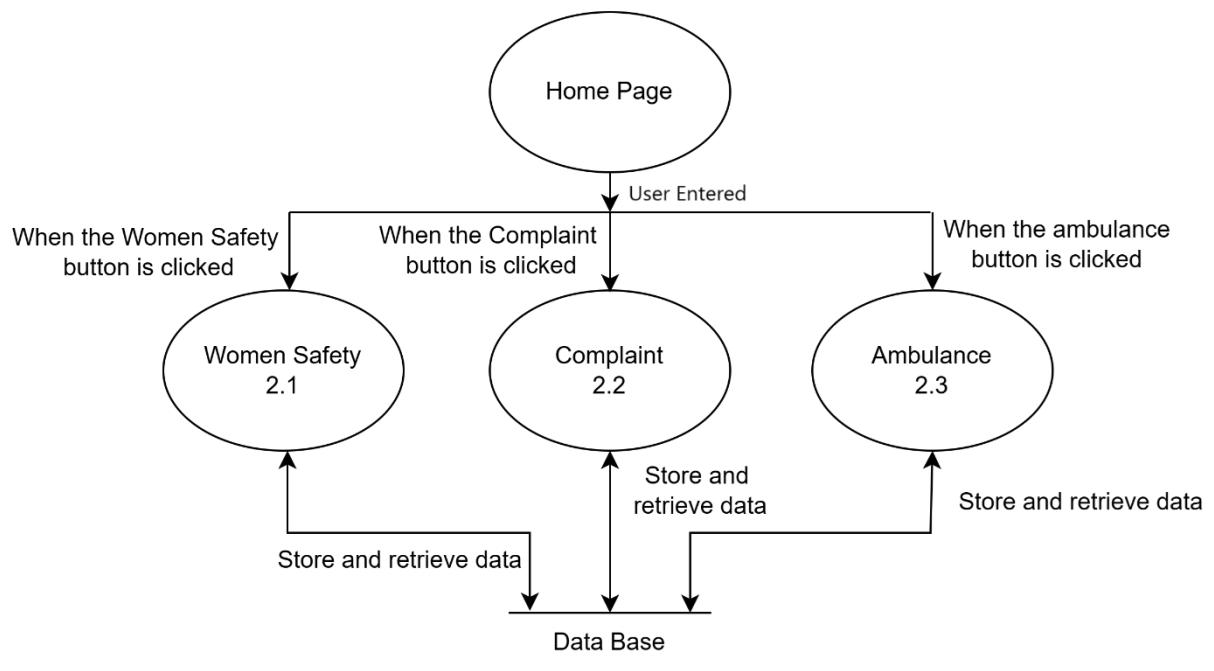
■ Level 1.1



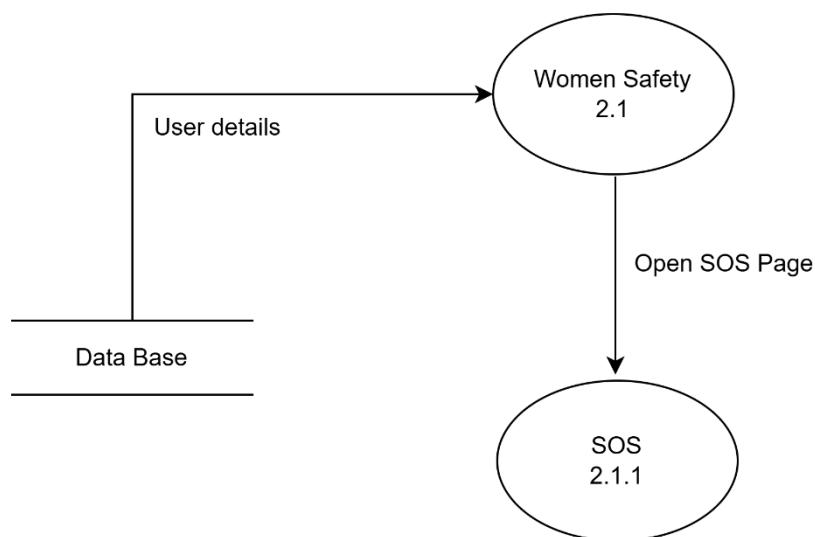
■ Level 1.2



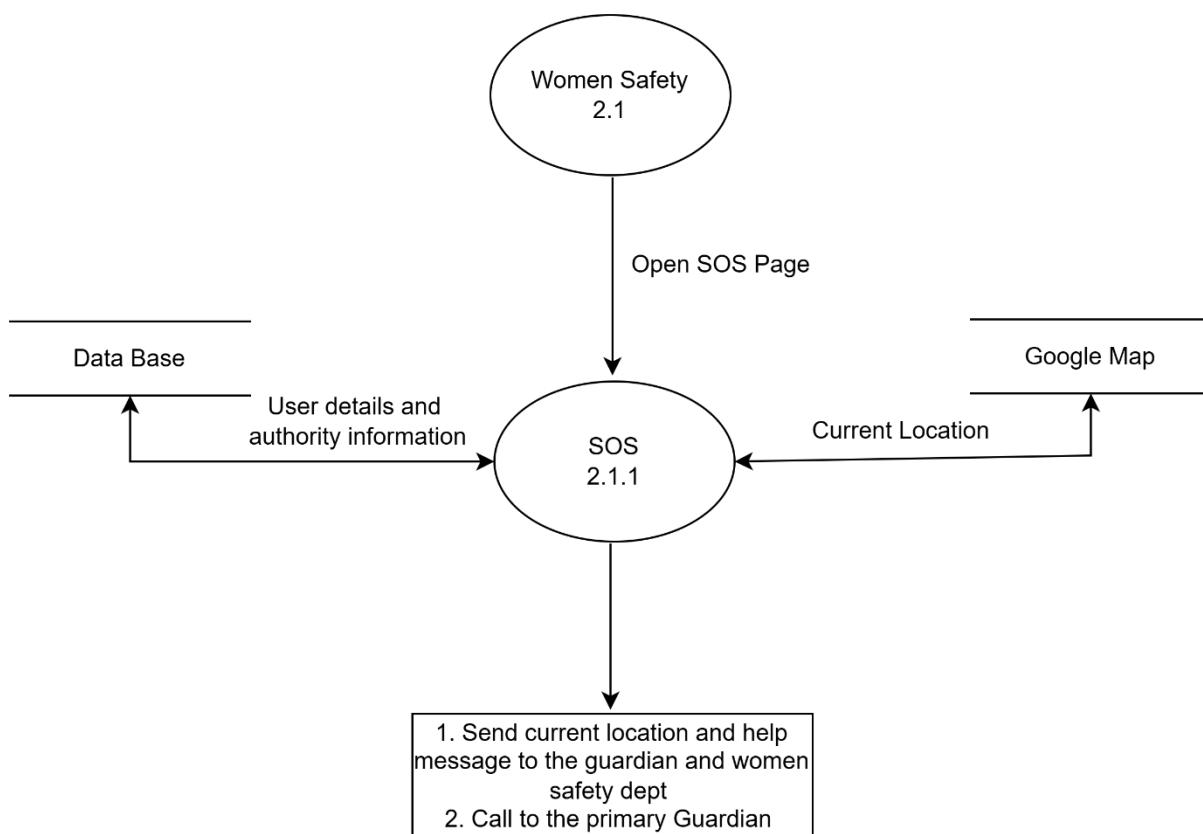
■ Level 2



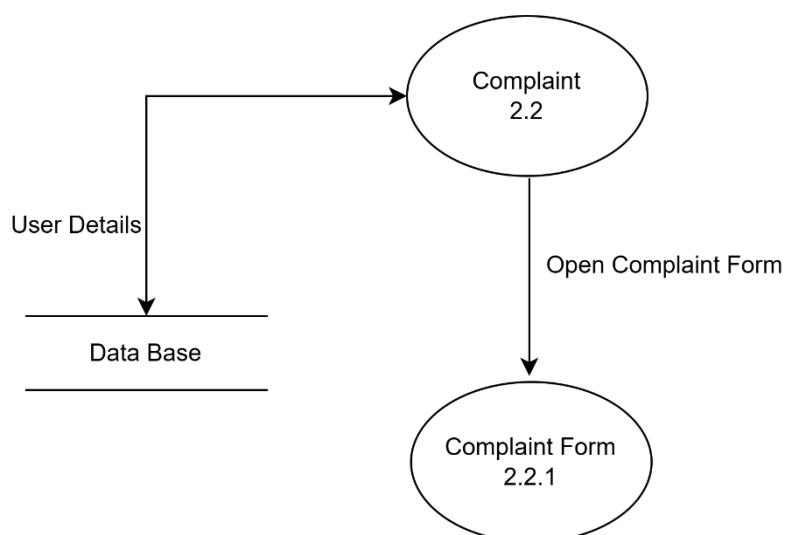
■ Level 2.1



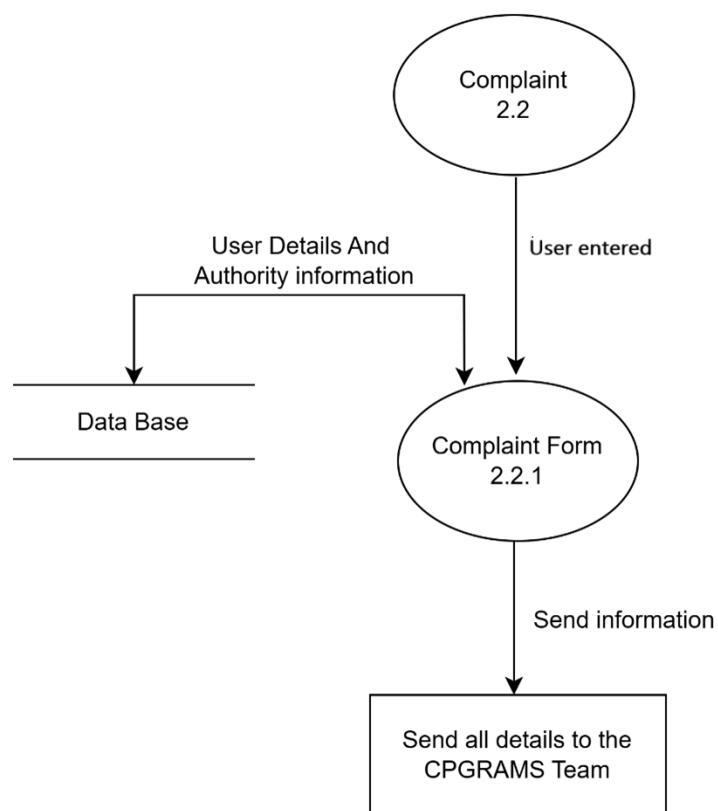
■ Level 2.1.1



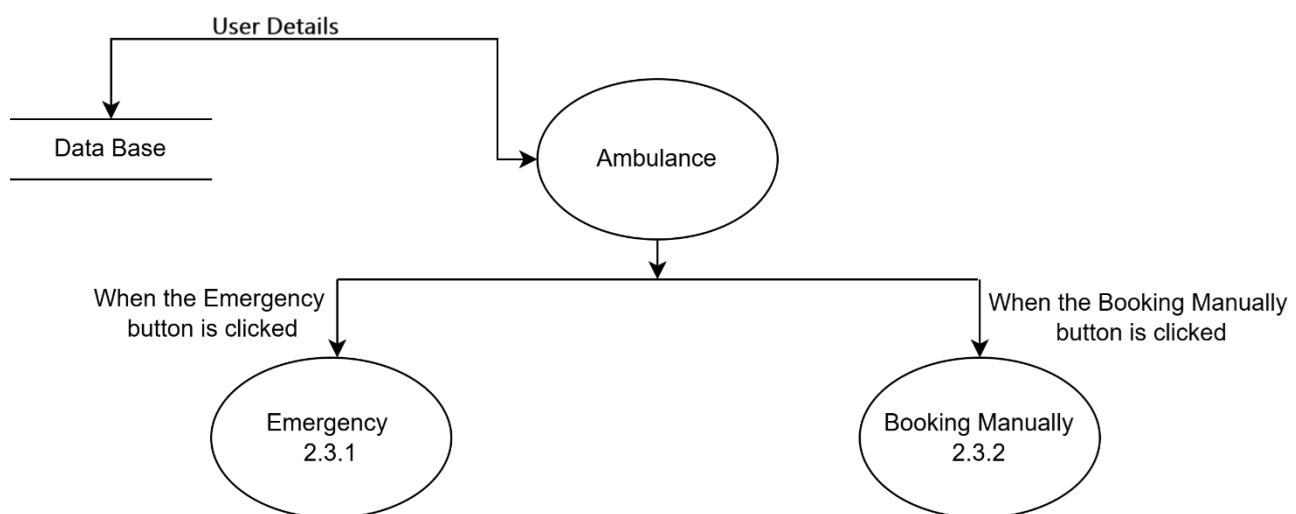
■ Level 2.2



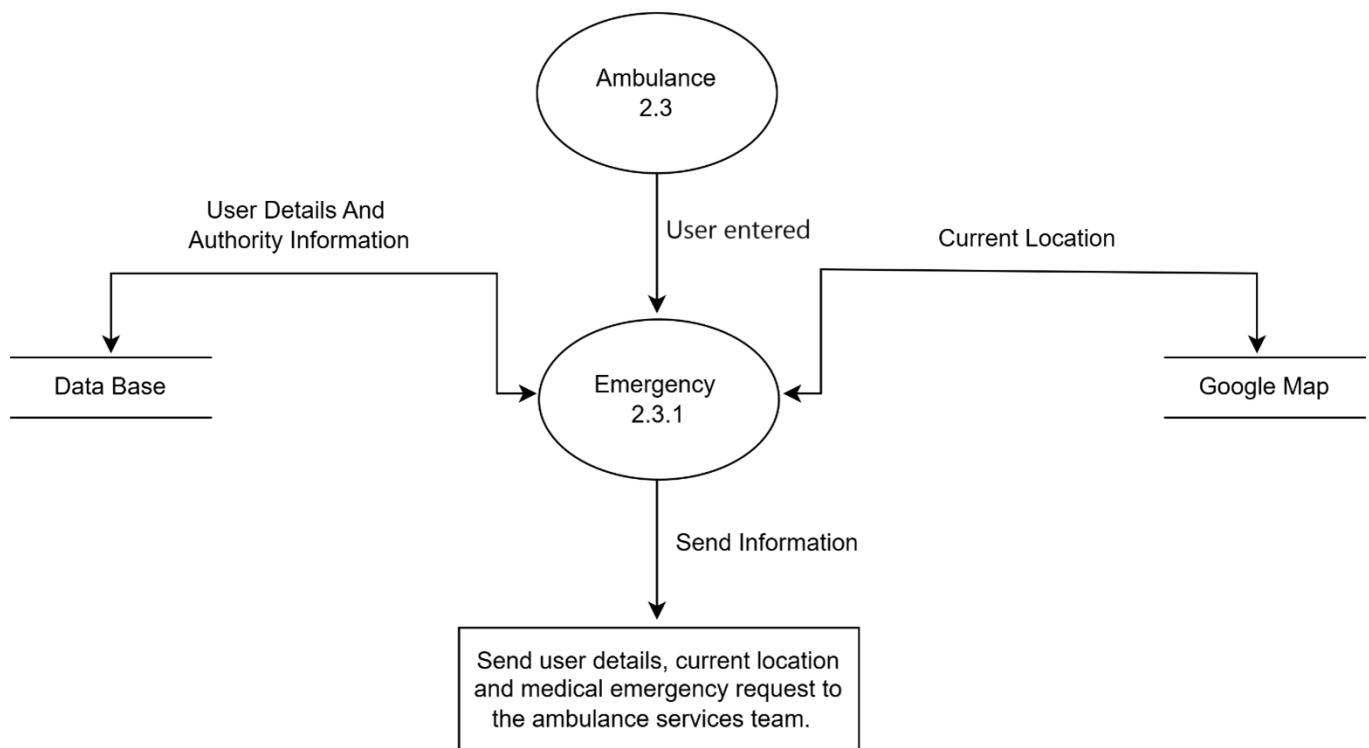
■ Level 2.2.1



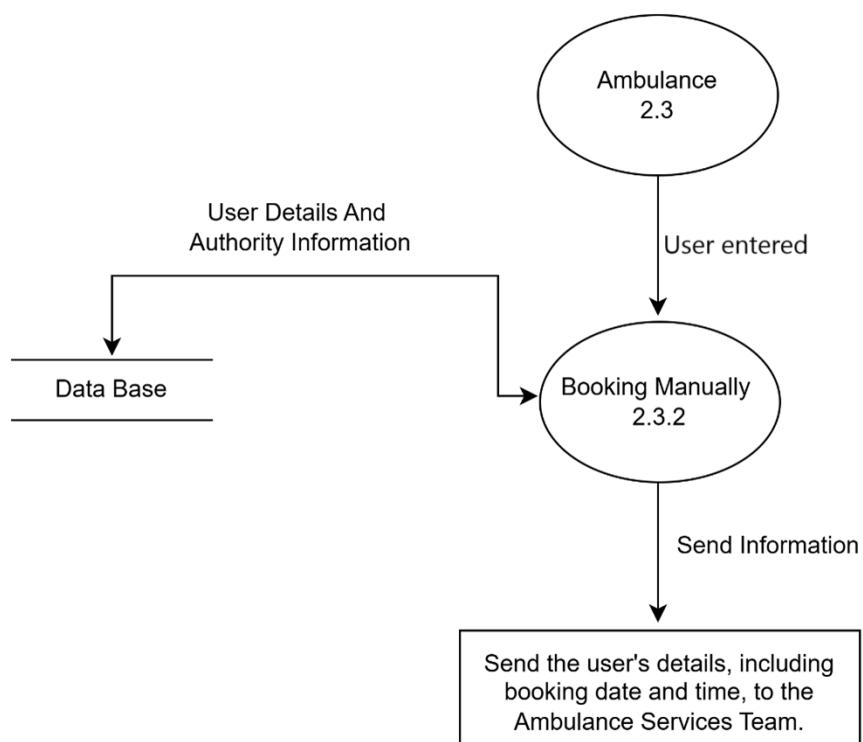
■ Level 2.3



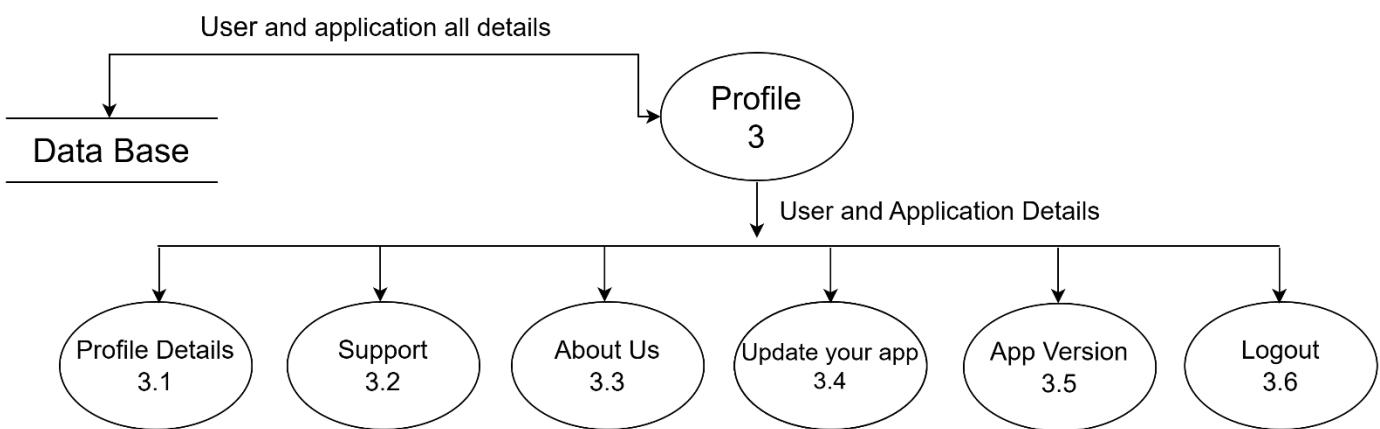
■ Level 2.3.1



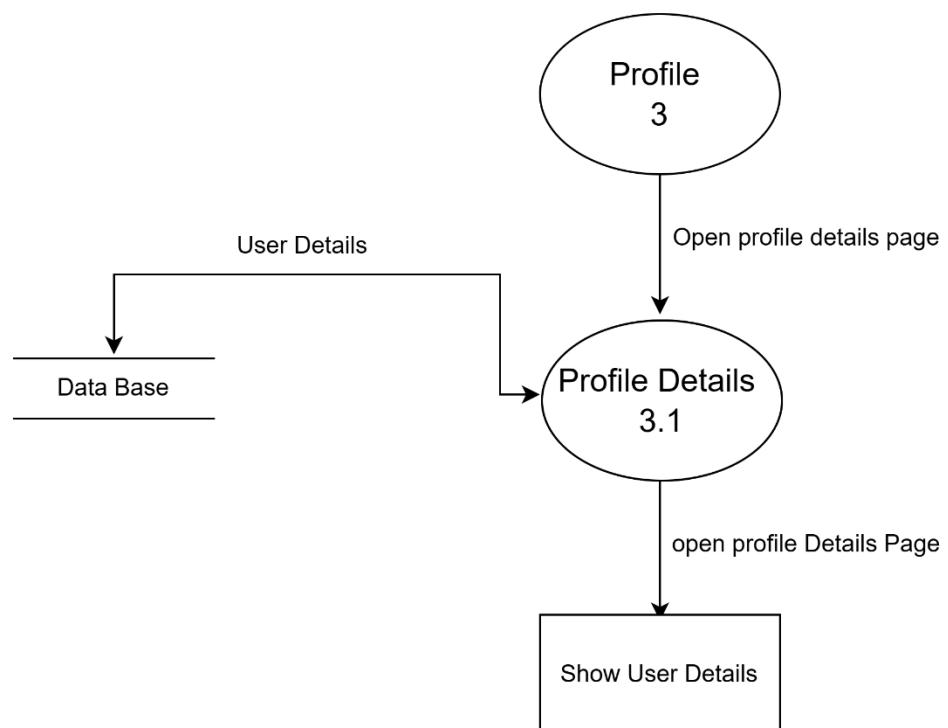
■ Level 2.3.2



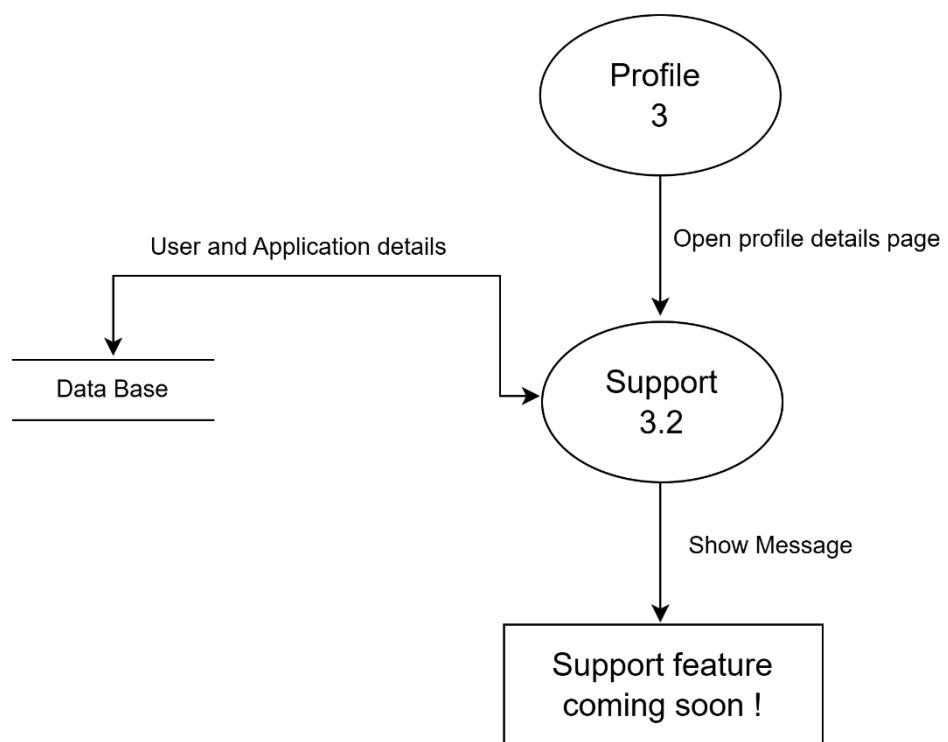
■ Level 3



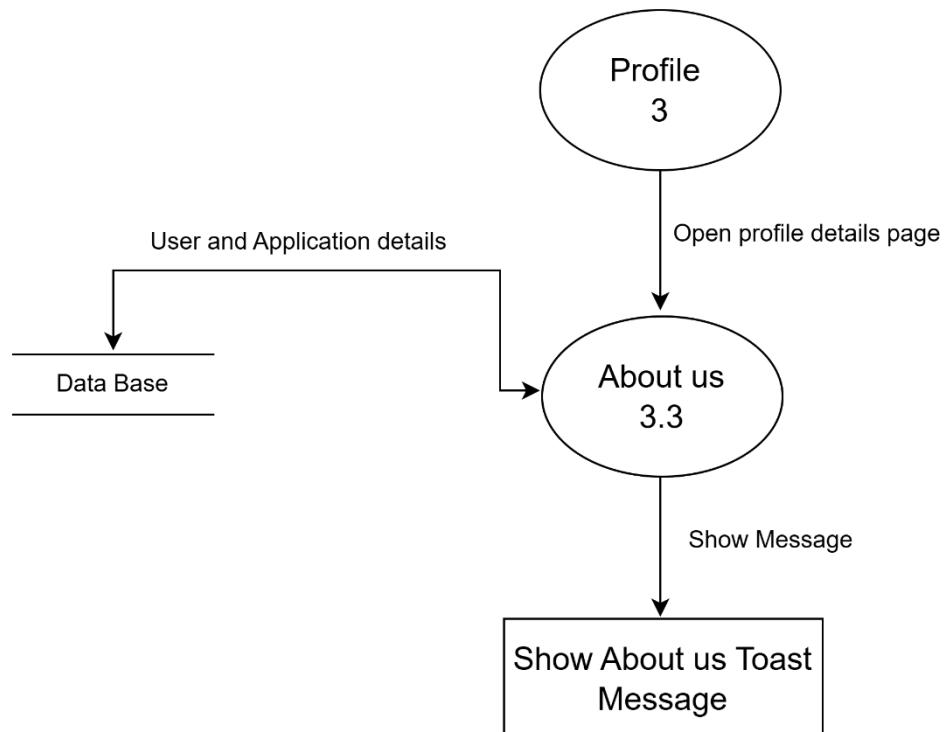
■ Level 3.1



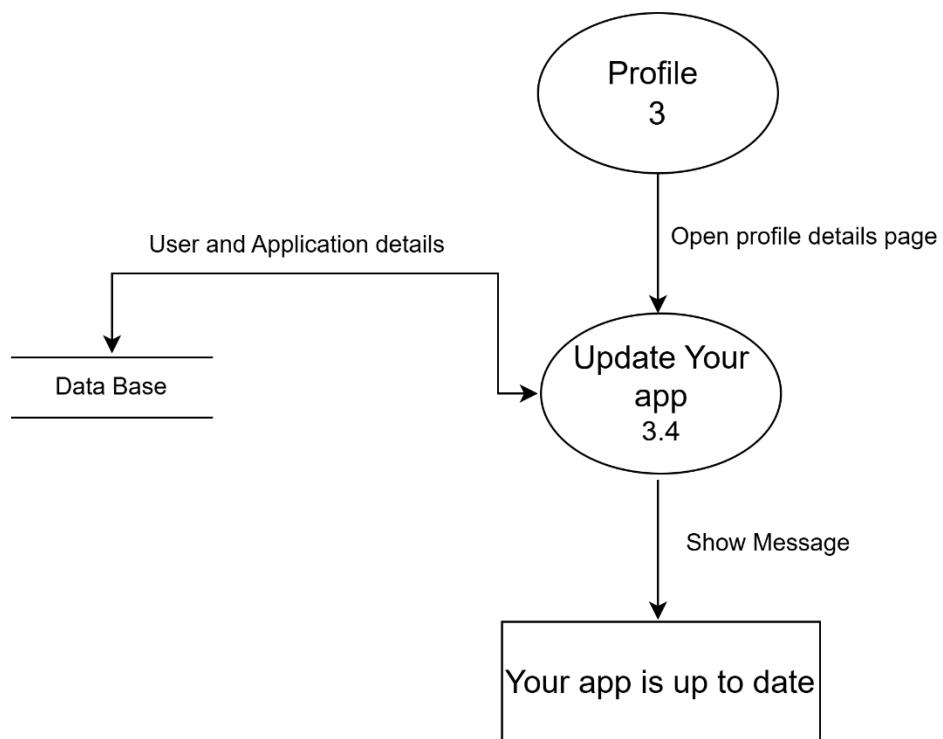
■ Level 3.2



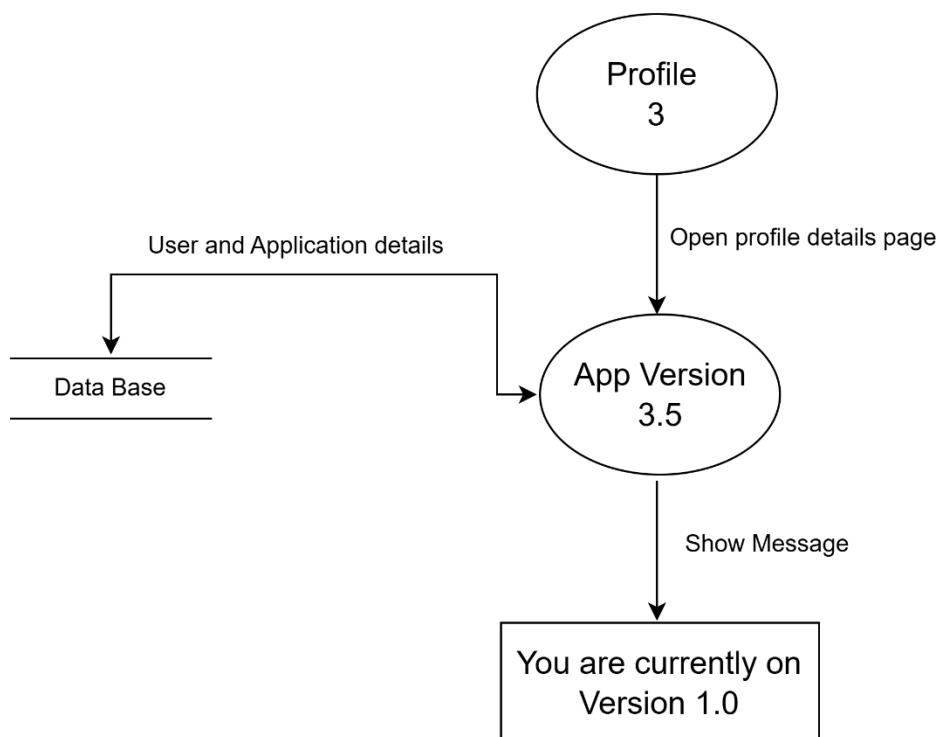
■ Level 3.3



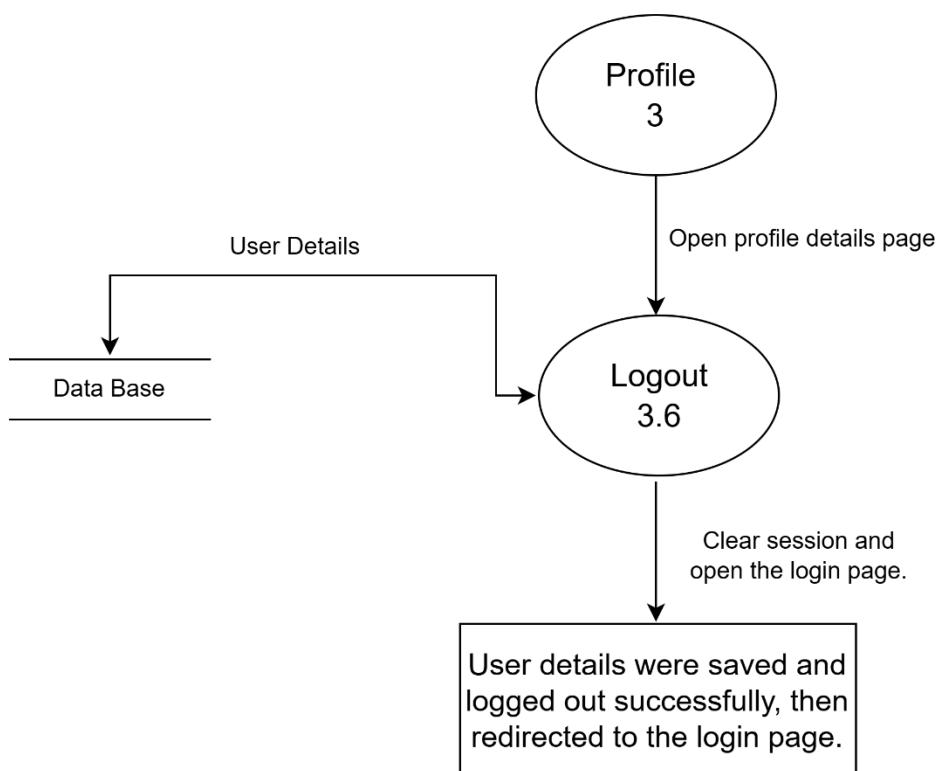
■ Level 3.4



■ Level 3.5



■ Level 3.6



TECHNOLOGY USED

5.1.1 Programming Languages and Frameworks

- **Frontend Framework**
 - Android XML layout or XML
- **Backend Framework**
 - Android SDK with JAVA or JAVA

5.1.2 Development Tools

- **Android Studio**

5.1.3 Testing Device

- **Emulator**
 - Google Pixel 7
 - Google Pixel 9
 - POCO
 - Vivo V30e

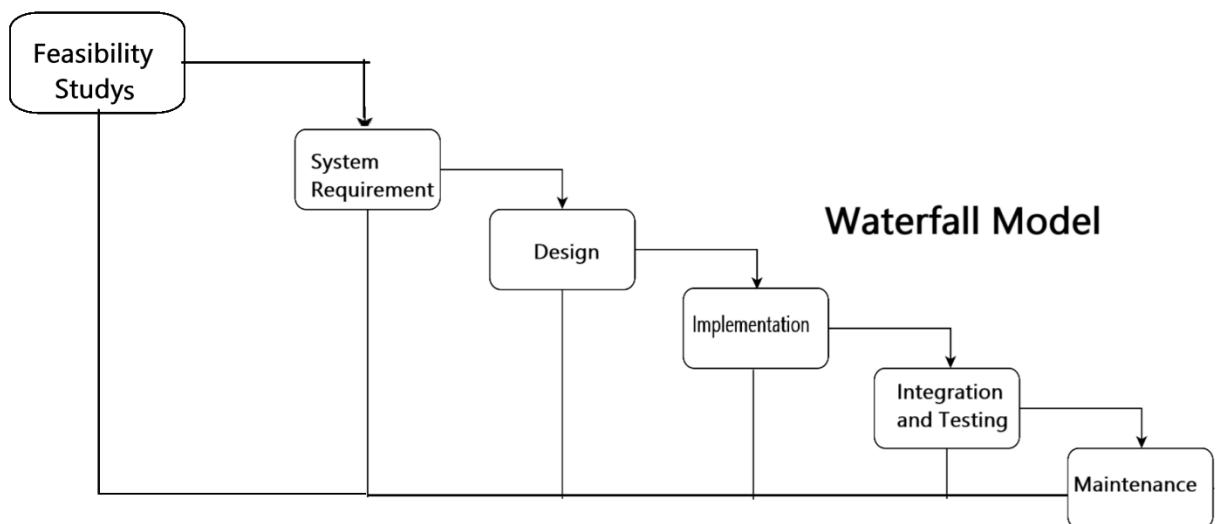
5.1.4 Database

- **SQLite**

IMPLEMENTATION

6.1.1 Waterfall Model

The waterfall model was introduced by Win Royce in Lockheed in the 1970s. It is named waterfall because it can be graphically represented as a waterfall from requirements definition to design creation, program implementation, system testing, and delivery to the customer (Jayaswal & Patton, 2006). The waterfall model is a software development method that is depicted like a waterfall through various phases as outlined in figure-



This method is described as a traditional engineering method used in software engineering. With this method, once the phase is started or completed, it is neither modified nor revised, which is the main disadvantage. Due to this inflexibility in a waterfall model, most studies have been criticized and therefore suggested more flexible alternatives. It is therefore not an attractive method, to some extent related to projects that exceed the budget and has been replaced by more flexible and versatile methods created specifically for software development.

The waterfall method is very suitable for supporting inexperienced project managers and teams or for creating volatile project teams. Progress in the development of the waterfall system is inflexible, slow, and expensive. The project moves forward and moves only slightly backwards. It is impossible to test the produced system until it is nearly totally coded and difficult to repair or respond to modifications. Changes made later in a person's life cycle are more pricey and so are not recommended. The Waterfall is often difficult to read and understand.

6.1.2 Frontend code

1. Activity home page

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout 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/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white"
    android:orientation="vertical"
    tools:context=".First_Activity.Home_page_Activity">
    <FrameLayout
        android:id="@+id/id_home_page_framlayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="@+id/id_bottom_navigation"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0" />

    <com.google.android.material.bottomnavigation.BottomNavigationView
        android:id="@+id/id_bottom_navigation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="parent"
        app:menu="@menu/bottom_navigation_bar_icons" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

2. Activity_first_opening_animation

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout 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/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".First_Activity.first_opening_animation">

    <com.airbnb.lottie.LottieAnimationView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
```

```
tools:ignore="MissingClass"
app:lottie_rawRes="@raw/first_opening_lottie_animation"
app:lottie_autoPlay="true"
app:lottie_loop="true" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

3. Fragment_sign_up

```
<?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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".First_Activity.Fragment_sign_up"
    android:orientation="vertical">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/top_background"
        android:layout_marginBottom="-35dp"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="30dp"
            app:cardCornerRadius="30dp"
            app:cardElevation="20dp">

            <ScrollView
                android:layout_width="match_parent"
                android:layout_height="wrap_content">

                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:orientation="vertical"
                    android:gravity="center"
                    android:padding="24dp">

                    <TextView
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:text="Sign up"
                        android:textSize="40dp" />
                    <!--Edit Text Button Code Start.....-->

                    <com.google.android.material.textfield.TextInputLayout
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
```

```
        android:hint="Enter Name"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="5dp">>

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:inputType="text"
            android:id="@+id/up_name_et"
            android:layout_height="wrap_content" />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Mobile Number"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="5dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:id="@+id/up_phone_et"
            android:layout_height="wrap_content"
            android:inputType="phone"/>

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Email Id"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="5dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:id="@+id/up_email_et"
            android:inputType="textEmailAddress"
            android:layout_height="wrap_content" />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Guardian's Mobile Number "
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="5dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:id="@+id/up_guardian_et"
            android:inputType="phone"
            android:layout_height="wrap_content" />

    </com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Password "
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="5dp"
        app:passwordToggleEnabled="true" >

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:id="@+id/up_password_et"
            android:inputType="textPassword"
            android:layout_height="wrap_content"/>

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Pin Code"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="5dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:id="@+id/up_pin_code_et"
            android:layout_height="wrap_content"
            android:inputType="number"/>

    </com.google.android.material.textfield.TextInputLayout>

<!--Edit Text Botton Code End.....-->
<CheckBox
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="I accept all terms and conditions"
    android:id="@+id/up_check_box"
    android:paddingRight="13dp"/>

<Button
    android:id="@+id/btn_register_go"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Signup"
    android:backgroundTint="@color/purple_200" />

</LinearLayout>
</ScrollView>
</androidx.cardview.widget.CardView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Already have an Account ? "

```

```

        android:textSize="20dp"/>

<TextView
    android:id="@+id/id_tv_sign_in_page"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    android:textSize="20dp"
    android:textStyle="bold"
    android:textColor="@color/blue" />
</LinearLayout>
</LinearLayout>
</LinearLayout>

```

4. Fragment home page

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".First_Activity.Fragment_home_page"
    android:background="@color/white">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/top_bg_layout">

        <ImageView
            android:id="@+id/top_bg_picture"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/top_background"
        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:orientation="vertical"
            android:paddingTop="240dp">

            <LinearLayout
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/first_layout_home_page_btn"
                android:layout_below="@+id/top_bg_layout"
                android:paddingBottom="0dp"
                android:gravity="center">
                <androidx.cardview.widget.CardView
                    android:id="@+id/cardView1_safe_women"
                    android:layout_width="150dp"
                    android:layout_height="150dp"
                    android:layout_alignParentStart="true"

```

```
        android:layout_alignParentBottom="true"
        android:layout_margin="12dp"
        android:layout_marginStart="12dp"
        android:layout_marginBottom="729dp"
        app:cardCornerRadius="25dp"
        app:cardElevation="10dp">>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:gravity="center"
        android:paddingBottom="20dp">

        <ImageView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_gravity="center"
            android:src="@drawable/safe_image"
            android:layout_weight="1"/>

        <TextView
            android:layout_width="match_parent"
            android:layout_height="35dp"
            android:layout_gravity="center_horizontal"
            android:text="women safety"
            android:textColor="@color/black"
            android:textSize="20sp"
            android:textStyle="bold"
            android:gravity="center_horizontal"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView2_online_complain"
    android:layout_width="150dp"
    android:layout_height="150dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentBottom="true"
    android:layout_margin="12dp"
    android:layout_marginStart="12dp"
    android:layout_marginBottom="729dp"
    app:cardCornerRadius="25dp"
    app:cardElevation="10dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:gravity="center"
        android:paddingBottom="20dp">

        <ImageView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_gravity="center"
            android:src="@drawable/police_report"
```

```
        android:layout_weight="1"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="27dp"
        android:layout_gravity="center_horizontal"
        android:text="Complaint"
        android:textColor="@color/black"
        android:textSize="20sp"
        android:textStyle="bold"
        android:gravity="center_horizontal"
        android:layout_marginBottom="5dp"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/second_layout_home_page_btn"
    android:layout_below="@+id/first_layout_home_page_btn"
    android:paddingBottom="10dp"
    android:gravity="center">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView3_ambulance"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:layout_alignParentStart="true"
        android:layout_alignParentBottom="true"
        android:layout_margin="12dp"
        android:layout_marginStart="12dp"
        android:layout_marginBottom="729dp"
        app:cardCornerRadius="25dp"
        app:cardElevation="10dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:gravity="center"
            android:paddingBottom="20dp">

            <ImageView
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:layout_gravity="center"
                android:src="@drawable/ambulance_icon"
                android:layout_weight="1"/>

            <TextView
                android:layout_width="match_parent"
                android:layout_height="35dp"
                android:layout_gravity="center_horizontal"
                android:gravity="center_horizontal"
                android:text="Ambulance"
                android:textColor="@color/black"
```

```

        android:textSize="20sp"
        android:textStyle="bold" />
    </LinearLayout>
</androidx.cardview.widget.CardView>
</LinearLayout>
</LinearLayout>
</RelativeLayout>

```

5. Fragment_home_btn_safe_women

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    tools:context=".First_Activity.Fragment_home_btn_safe_women"
    android:background="#4D60C3">
    <ImageButton
        android:id="@+id/sos_button"
        android:layout_width="250dp"
        android:layout_height="250dp"
        android:background="@drawable/sos_button_icon"
        android:layout_gravity="center"
        android:padding="53dp">
    </ImageButton>
</FrameLayout>

```

6. Activity_google_maps

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center_horizontal"
    android:orientation="vertical"
    tools:context=".First_Activity.MapsActivity">
    <SearchView
        android:id="@+id/searchView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginHorizontal="10dp"
        android:background="@color/white"
        android:iconifiedByDefault="false"
        android:queryHint="Search"
        map:layout_constraintEnd_toEndOf="parent"
        map:layout_constraintStart_toStartOf="parent"
        map:layout_constraintTop_toTopOf="parent" />
    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="406dp"

```

```

        android:layout_height="713dp"
        map:cameraTargetLat="22.9868"
        map:cameraTargetLng="87.8550"
        map:cameraTilt="30"
        map:cameraZoom="13"
        map:layout_constraintBottom_toBottomOf="parent"
        map:layout_constraintEnd_toEndOf="parent"
        map:layout_constraintStart_toStartOf="parent"
        map:layout_constraintTop_toBottomOf="@+id/searchView"
        map:layout_constraintVertical_bias="0.0"
        map:uiCompass="true"
        map:uiRotateGestures="true"
        map:uiScrollGestures="true"
        map:uiTiltGestures="true"
        map:uiZoomControls="true"
        map:uiZoomGestures="true" />
    <ImageButton
        android:id="@+id/police_button"
        android:layout_width="50dp"
        android:src="@drawable/baseline_accessibility"
        android:layout_height="50dp"
        android:layout_marginTop="16dp"
        android:textColor="#F21E1E"
        map:layout_constraintEnd_toEndOf="@+id	btn"
        map:layout_constraintHorizontal_bias="0.434"
        map:layout_constraintStart_toStartOf="@+id	btn"
        map:layout_constraintTop_toBottomOf="@+id	btn" />
    <Button
        android:id="@+id	btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="MODES"
        map:layout_constraintBottom_toBottomOf="parent"
        map:layout_constraintEnd_toEndOf="parent"
        map:layout_constraintHorizontal_bias="1.0"
        map:layout_constraintStart_toStartOf="parent"
        map:layout_constraintTop_toTopOf="parent"
        map:layout_constraintVertical_bias="0.229" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

7. Add others guardian's number

```

<?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:orientation="vertical">
    <androidx.appcompat.widget.AppCompatButton
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:background="@drawable/back_btn_account"
        android:backgroundTint="@color/black"
        android:layout_margin="14dp"
        android:id="@+id/add_guardian_to_edit_profile_page_back"/>
    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```

        android:hint="Enter Guardian Mobile Number"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:paddingHorizontal="15dp"
        android:layout_marginTop="15dp">
        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:inputType="phone"
            android:id="@+id/id_add_guardian_mobile_number_et"
            android:layout_height="wrap_content" />
    </com.google.android.material.textfield.TextInputLayout>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="ADD"
        android:layout_marginTop="7dp"/>
</LinearLayout>

```

6.1.3 Backend Code

1. Home page Activity

```

package com.example.xsefety.First_Activity;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.FrameLayout;
import androidx.activity.EdgeToEdge;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import com.example.xsefety.R;
import com.google.android.material.bottomnavigation.BottomNavigationView;
public class Home_page_Activity extends AppCompatActivity {
    FrameLayout id_home_page_framlayout;
    BottomNavigationView id_bottom_navigation;
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_home_page);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
        id_home_page_framlayout =(FrameLayout) findViewById(R.id.id_home_page_framlayout);
        id_bottom_navigation = (BottomNavigationView) findViewById(R.id.id_bottom_navigation);
    }
}

```

```

getSupportFragmentManager().beginTransaction().replace(R.id.id_home_page_framlayout,new
Fragment_home_page()).commit();
        id_bottom_navigation.setOnNavigationItemSelected(new
BottomNavigationView.OnNavigationItemSelectedListener() {
    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        int itemId = item.getItemId();
        if (itemId == R.id.home_btn_navigation_bar)
        {
            getSupportFragmentManager().beginTransaction().replace(R.id.id_home_page_framlayout,new
Fragment_home_page()).addToBackStack(null).commit();
        }
        else if (itemId == R.id.map_btn_navigation_bar)
        {
            Intent intent = new Intent(getApplicationContext(), MapsActivity.class);
            startActivity(intent);
        }
        else if (itemId == R.id.profile_btn_navigation_bar)
        {
            getSupportFragmentManager().beginTransaction().replace(R.id.id_home_page_framlayout,
new Fragment_Account_Details()).addToBackStack(null).commit();
        }
        return true;
    }
});
}
}

```

2. First_opening-animation

```

package com.example.xsefety.First_Activity;
import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.os.Handler;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import com.example.xsefety.R;
public class first_opening_animation extends AppCompatActivity {
    public String[] permissions;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        // EdgeToEdge.enable(this);
        setContentView(R.layout.activity_first_opening_animation);
        permissions = new String[]{Manifest.permission.SEND_SMS,

```

```

Manifest.permission.ACCESS_COARSE_LOCATION,
Manifest.permission.CALL_PHONE,
Manifest.permission.ACCESS_FINE_LOCATION,
Manifest.permission.READ_EXTERNAL_STORAGE,
Manifest.permission.ACCESS_NETWORK_STATE,
Manifest.permission.INTERNET
};

if (ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED &&
    ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED &&
    ActivityCompat.checkSelfPermission(this, Manifest.permission.CALL_PHONE) != PackageManager.PERMISSION_GRANTED &&
    ActivityCompat.checkSelfPermission(this, Manifest.permission.SEND_SMS) != PackageManager.PERMISSION_GRANTED &&
    ActivityCompat.checkSelfPermission(this, Manifest.permission.READ_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED) {

    ActivityCompat.requestPermissions(this, permissions, 1);
} else {
    HANDLER();
}
ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
    Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
    v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
    return insets;
});
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == 1) {
        if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED &&
            grantResults.length > 1 && grantResults[1] == PackageManager.PERMISSION_GRANTED &&
            grantResults.length > 2 && grantResults[2] == PackageManager.PERMISSION_GRANTED &&
            grantResults.length > 3 && grantResults[3] == PackageManager.PERMISSION_GRANTED &&
            grantResults.length > 4 && grantResults[4] == PackageManager.PERMISSION_GRANTED)
        {
            HANDLER();
        } else
        {
            ActivityCompat.requestPermissions(first_opening_animation.this, permissions, 1);
        }
    }
}
void HANDLER() {
    new Handler().postDelayed(new Runnable() {
        @Override
        public void run() {
            Intent intent = new Intent(first_opening_animation.this, First_Activity_01.class);
            startActivity(intent);
        }
    }, 1000);
}

```

```
        finish();
    }
}, 3000);
}
}
```

3. Fragment_sign_up

```
package com.example.xsefety.First_Activity;
import android.content.Intent;
import android.os.Bundle;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentTransaction;
import android.text.Editable;
import android.text.TextWatcher;
import android.util.Patterns;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.TextView;
import android.widget.Toast;
import com.example.xsefety.R;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
public class Fragment_sign_up extends Fragment {
```

```
    View view;
    TextView id_tv_sign_up_page;
    public String name,password,guardian_phone_number,self_phone_number,pin;
    boolean
name_boolean=false,self_phone_number_boolean=false,guardian_phone_number_boolean=false,pin_number_boolean=false,password_boolean=false;
    com.google.android.material.textfield.TextInputEditText
up_name_et,up_phone_et,up_pin_et,up_email_et,up_password_et,up_guardian_et;
    CheckBox up_check_box;
    Button btn_register_go;
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view= inflater.inflate(R.layout.fragment_sign_up, container, false);
        up_check_box=(CheckBox) view.findViewById(R.id.up_check_box);
        up_email_et=(com.google.android.material.textfield.TextInputEditText)
view.findViewById(R.id.up_email_et);
        up_guardian_et=(com.google.android.material.textfield.TextInputEditText)
view.findViewById(R.id.up_guardian_et);
        up_name_et=(com.google.android.material.textfield.TextInputEditText)
view.findViewById(R.id.up_name_et);
```

```

        up_pin_et=(com.google.android.material.textfield.TextInputEditText)
view.findViewById(R.id.up_pin_code_et);
        up_phone_et=(com.google.android.material.textfield.TextInputEditText)
view.findViewById(R.id.up_phone_et);
        up_password_et=(com.google.android.material.textfield.TextInputEditText)
view.findViewById(R.id.up_password_et);

// check name validation
up_name_et.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int after) {
    }
    @Override
    public void onTextChanged(CharSequence s, int start, int before, int count) {
        name=up_name_et.getText().toString();
        if (!(name.isEmpty()))
        {
            if (name.length() >= 2 && isAlphaValid(name))
            {
                if (isNum_name_check(name))
                {
                    if(isSpecial(name))
                    {
                        name_boolean = true;
                    }
                    else
                    {
                        up_name_et.setError("Special character not allowed.");
                    }
                }
                else
                {
                    up_name_et.setError("digit is not valid");
                }
            }
            else{
                if(isAlphaValid(name))
                {
                    up_name_et.setError("Please enter your full name");
                }
                else {
                    up_name_et.setError("digit is not valid");
                }
            }
        }
        else
        {
            up_name_et.setError("Empty!");
        }
    }
    @Override
    public void afterTextChanged(Editable s) {

```

```

    }

});

// check phone number
up_phone_et.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int after) {
    }
    @Override
    public void onTextChanged(CharSequence s, int start, int before, int count) {
        self_phone_number=up_phone_et.getText().toString();

        if(self_phone_number.startsWith("+91"))
        {
            if (validateMobile(self_phone_number.substring(3))) {
                self_phone_number_boolean = true;
            } else {
                up_phone_et.setError("Invalid mobile number");
            }
        }
        else
        {
            if (validateMobile(self_phone_number)) {
                self_phone_number_boolean = true;
                self_phone_number="+91"+self_phone_number;
            } else {

                up_phone_et.setError("Invalid mobile number");
            }
        }
    }
});

@Override
public void afterTextChanged(Editable s) {
}

});

// check guardian phone number
up_guardian_et.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int after) {
    }
    @Override
    public void onTextChanged(CharSequence s, int start, int before, int count) {
        guardian_phone_number=up_guardian_et.getText().toString();
        if(!self_phone_number.contains(guardian_phone_number))
        {
            if (guardian_phone_number.startsWith("+91"))
            {
                if (validateMobile(guardian_phone_number.substring(3)))
                {

```

```

        guardian_phone_number_boolean = true;
    }
    else
    {
        up_guardian_et.setError("Invalid mobile number");
    }
}
else
{
    if (validateMobile(guardian_phone_number)) {
        guardian_phone_number_boolean = true;
        guardian_phone_number = "+91" + guardian_phone_number;
    } else {
        up_guardian_et.setError("Invalid mobile number");
    }
}
else
{
    up_guardian_et.setError("Enter guardian number");
    up_guardian_et.requestFocus();
}
});
}

// check pin number validation
up_pin_et.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int after) {
    }
    @Override
    public void onTextChanged(CharSequence s, int start, int before, int count) {
        pin=up_pin_et.getText().toString();

        if(isNum_pin(pin) && pin.length()==6)
        {
            pin_number_boolean=true;
        }
        else
        {
            up_pin_et.setError(" ");
            up_pin_et.requestFocus();
        }
    }
    @Override
    public void afterTextChanged(Editable s) {
    }
});
//check password validation

```

```

        up_password_et.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence s, int start, int count, int after) {
            }

            @Override
            public void onTextChanged(CharSequence s, int start, int before, int count) {
                password=up_password_et.getText().toString();
                if (validatePassword(password)) {
                    password_boolean = true;
                } else {
                    up_password_et.setError("length 8\n use uppercase\nlowercase\nand special symbol");
                }
            }

            @Override
            public void afterTextChanged(Editable s) {
            }
        });

        id_tv_sign_up_page = (TextView) view.findViewById(R.id.id_tv_sign_in_page);
        btn_register_go =(Button) view.findViewById(R.id.btn_register_go);
        // register button submit
        btn_register_go.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (name_boolean) {
                    if (self_phone_number_boolean) {
                        if (validateEmail(up_email_et.getText().toString())){
                            if(guardian_phone_number_boolean) {
                                if(password_boolean) {
                                    if(pin_number_boolean)
                                    {
                                        if(up_check_box.isChecked()) {
                                            // SQLite data store
                                            MyDbHelper myDbHelper=new MyDbHelper(view.getContext());

                                            myDbHelper.InsertUserHelper(name,self_phone_number,up_email_et.getText().toString(),password,pin);
                                            myDbHelper.InsertGuardianHelper(guardian_phone_number);

                                            //SharedPreferences data store
                                            sessionManager s=new sessionManager(view.getContext());
                                            s.createSession(name,self_phone_number);
                                            Intent intent = new Intent(getApplicationContext(), Home_page_Activity.class);
                                            startActivity(intent);
                                        }
                                        else
                                        {
                                            Toast.makeText(view.getContext(),"Accept terms and
condition",Toast.LENGTH_SHORT).show();
                                        }
                                    }else
                                    {
                                        up_pin_et.setError(" ");
                                    }
                                }
                            }
                        }
                    }
                }
            }
        });
    }
}

```

```

        up_pin_et.requestFocus();
    }
}else{
    up_password_et.setError(" ");
    up_password_et.requestFocus();
}
}else{
    up_guardian_et.setError(" ");
    up_guardian_et.requestFocus();
}

} else {
    up_email_et.setError(" ");
    up_email_et.requestFocus();
}
} else {
    up_phone_et.setError(" ");
    up_phone_et.requestFocus();
}
} else {
    up_name_et.setError("");
    up_name_et.requestFocus();
}
});
id_tv_sign_up_page.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Fragment open_sign_in = new Fragment_sign_in();
        FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.frameLayout_first,
open_sign_in,null).addToBackStack(null).commit();

    }
});
return view;
}

}

boolean validateMobile(String phone) {
    Pattern p = Pattern.compile("[6-9][0-9]{9}");
    Matcher m = p.matcher(phone);
    return m.matches();
}
// check password validation function
boolean validatePassword(String password) {
    Pattern p = Pattern.compile("^"
+ "(?=.*[0-9])" //minimum one number
+ "(?=.*[a-z])" //minimum one lower case character
+ "(?=.*[A-Z])" //minimum one UPPER case character
+ "(?=.*[a-zA-Z])" //any character

```

```

+ "(?=.*[@#$%^&+=])" //minimum one special character
+ "(?=\\S+$)" // no white spaces
+ ".{8,}" // on white spaces
+ "$"); //minimum length 6 characters
Matcher m = p.matcher(password);
return m.matches();
}

//check name contain digit or not
boolean isNum_name_check(String name)
{
    Pattern p = Pattern.compile("^"
+ "(?=.*[0-9])"
+ ".{2,}" // on white spaces
+ "$"); //minimum length characters
Matcher m = p.matcher(name);
return !(m.matches());
}
//check pin number contain only number
boolean isNum_pin(String name)
{
    Pattern p = Pattern.compile("^"
+ "(?=.*[0-9])"
+ ".{7,}" // on white spaces
+ "$"); //minimum length 6 characters
Matcher m = p.matcher(name);
return !(m.matches());
}

// check name contain special character or not
boolean isSpecial(String name)
{
    Pattern p = Pattern.compile("^"
+ "(?=.*[@#$%^&+=/?()^_~.,;\\|])"
+ ".{2,}" // on white spaces
+ "$"); //minimum length 6 characters
Matcher m = p.matcher(name);
return !(m.matches());
}

// check name contain only alphabetic character or not
boolean isAlphaValid(String name)
{
    Pattern p = Pattern.compile("^"
+ "(?=.*[a-z])" //minimum one lower case character
// + "(?=.*[A-Z])" //minimum one UPPER case character
+ "(?=.*[a-zA-Z])" //any character
+ ".{3,}" // on white spaces
+ "$"); //minimum length 6 characters
Matcher m = p.matcher(name);
return m.matches();
}

```

```

// Email validation code
boolean validateEmail(String email)
{
    if(email.isEmpty())
    {
        up_email_et.setError("Email is required. Can't be empty.");
        return false;
    } else if (!Patterns.EMAIL_ADDRESS.matcher(email).matches()) {

        up_email_et.setError("Email is invalid. Enter valid email address");
        return false;
    }
    else
    {
        return true;
    }
}
}

```

4. Fragment home page

```

package com.example.xsefety.First_Activity;
import android.os.Bundle;
import androidx.cardview.widget.CardView;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentTransaction;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import com.example.xsefety.R;
public class Fragment_home_page extends Fragment{
    View view;
    CardView cardView1_safe_women;
    CardView cardView2_online_complain;
    CardView cardView3_ambulance;
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_home_page, container, false);
    }
    // Safe women clickable button backend code
    cardView1_safe_women =(CardView) view.findViewById(R.id.cardView1_safe_women);
    cardView1_safe_women.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Fragment open_sign_up = new Fragment_home_btn_safe_women();
            FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
            fragmentTransaction.replace(R.id.id_home_page_framlayout,
open_sign_up,null).addToBackStack(null).commit();
        }
    });
}

```

```

// online complaint clickable button backend code
cardView2_online_complain=(CardView) view.findViewById(R.id.cardView2_online_complain);
cardView2_online_complain.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Fragment open_sign_up = new Fragment_online_complaint_system();
        FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.id_home_page_framlayout,
open_sign_up,null).addToBackStack(null).commit() ;
    }
});
// ambulance booking clickable button backend code
cardView3_ambulance =(CardView) view.findViewById(R.id.cardView3_ambulance);
cardView3_ambulance.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Fragment open_sign_up = new Fragment_booking_ambulance_home_page();
        FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.id_home_page_framlayout,
open_sign_up,null).addToBackStack(null).commit() ;
    }
});
return view;
}
}

```

5. Fragment home btn safe women

```

package com.example.xsefety.First_Activity;
import android.os.Bundle;
import androidx.cardview.widget.CardView;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentTransaction;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import com.example.xsefety.R;
public class Fragment_home_page extends Fragment{
    View view;
    CardView cardView1_safe_women;
    CardView cardView2_online_complain;
    CardView cardView3_ambulance;
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_home_page, container, false);
    }
    // Safe women clickable button backend code
    cardView1_safe_women =(CardView) view.findViewById(R.id.cardView1_safe_women);
    cardView1_safe_women.setOnClickListener(new View.OnClickListener() {

```

```

@Override
public void onClick(View v) {
    Fragment open_sign_up = new Fragment_home_btn_safe_women();
    FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
    fragmentTransaction.replace(R.id.id_home_page_framlayout,
open_sign_up,null).addToBackStack(null).commit() ;
}
});

// online complaint clickable button backend code
cardView2_online_complain=(CardView) view.findViewById(R.id.cardView2_online_complain);
cardView2_online_complain.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Fragment open_sign_up = new Fragment_online_complaint_system();
        FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.id_home_page_framlayout,
open_sign_up,null).addToBackStack(null).commit() ;

    }
});

// ambulance booking clickable button backend code
cardView3_ambulance =(CardView) view.findViewById(R.id.cardView3_ambulance);
cardView3_ambulance.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Fragment open_sign_up = new Fragment_booking_ambulance_home_page();
        FragmentTransaction fragmentTransaction =
getActivity().getSupportFragmentManager().beginTransaction();
        fragmentTransaction.replace(R.id.id_home_page_framlayout,
open_sign_up,null).addToBackStack(null).commit() ;
    }
});

return view;
}
}

```

6. Activity google maps

```

package com.example.xsefety.First_Activity;
import androidx.fragment.app.FragmentActivity;
import android.os.Bundle;
import com.example.xsefety.R;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.example.xsefety.databinding.ActivityMapsBinding;
import static java.lang.Math.abs;

```

```

import androidx.annotation.NonNull;
import androidx.core.app.ActivityCompat;
import android.Manifest;
import android.app.Activity;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.graphics.Canvas;
import android.graphics.drawable.Drawable;
import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.ImageButton;
import android.widget.SearchView;
import android.widget.Toast;
import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.model.BitmapDescriptor;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;
    private ActivityMapsBinding binding;
    private final int FIND_PERMISSION_CODE = 1;
    Location currentLocation;
    FusedLocationProviderClient fusedLocationProviderClient;
    //SearchView for map searchview
    private SearchView mapSearchView;

    boolean
    b=false,midnapore=false,kgp=false,debra=false,kolaghat=false,panskura=false,pingla=false,daspur=false,ghat
    al=false,keshpur=false,tamluk=false;

    ImageButton ImageButton;
    double distance=0.18;
    Handler handler;
    Runnable runnable;

    LatLng
    latLngKeshpur,latLngKgp,latLngMidnapur,latLngKolaghat,latLngPanskura,latLngGhatal,latLngPingla,latLngTaml
    uk,latLngDebra,latLngDaspur;

```

```

Button btn;
int c=0;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    binding = ActivityMapsBinding.inflate(getLayoutInflater());
    setContentView(binding.getRoot());
    location_calculator location_calculator=new location_calculator();
    btn=(Button) findViewById(R.id.btn);
    btn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            switch (c)
            {
                case 0: mMap.setMapType(GoogleMap.MAP_TYPE_SATELLITE);
                    btn.setText("satellite");
                    c++;
                    break;
                case 1:mMap.setMapType(GoogleMap.MAP_TYPE_TERRAIN);
                    btn.setText("Terrain");
                    c++;
                    break;
                case 2: mMap.setMapType(GoogleMap.MAP_TYPE_HYBRID);
                    btn.setText("Hybrid");
                    c++;
                    break;
                case 3:mMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);
                    btn.setText("Normal");
                    c=0;
                    break;
            }
        }
    });
    ImageButton=ImageButton findViewById(R.id.police_button);
    mapSearchView = (SearchView) (findViewById(R.id.searchView));
    mapSearchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String query) {
            Geocoder geocoder=new Geocoder(getApplicationContext());
            List<Address> list=new ArrayList<>();
            try {
                list=geocoder.getFromLocationName(query,1);
            } catch (IOException e) {
                e.printStackTrace();
            }
            if(list.size()>0)
            {
                Address address=list.get(0);
                LatLng latLngSearch=new LatLng(address.getLatitude(),address.getLongitude());
                mMap.addMarker(new MarkerOptions().position(latLngSearch).title("Hi"));
                mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLngSearch,16));
            }
        }
    });
}

```

```

    }
    return true;
}
@Override
public boolean onQueryTextChange(String newText) {
    return false;
});
//current location
fusedLocationProviderClient= LocationServices.getFusedLocationProviderClient(this);
handler=new Handler();
handler.postDelayed(runnable=new Runnable() {
    @Override
    public void run() {
        handler.postDelayed(runnable,2000);
        getLastLocation();
    }
},2000);
}
private void getLastLocation()
{
    if (ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,new String[]{Manifest.permission.ACCESS_FINE_LOCATION},FIND_PERMISSION_CODE);
    }
    else {
        try {
            Task<Location> task = fusedLocationProviderClient.getLastLocation();
            task.addOnSuccessListener(new OnSuccessListener<Location>() {
                @Override
                public void onSuccess(Location location) {
                    if (location != null) {
                        currentLocation = location;
                        location_calculator.setLat(currentLocation.getLatitude());
                        location_calculator.setLng(currentLocation.getLongitude());
                        SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map);
                        if(mapFragment==null)
                            return;
                        mapFragment.getMapAsync(MapsActivity.this);
                    }
                }
            if (mMap != null)
            {
                mMap.clear();
                LatLng sydney = new LatLng(currentLocation.getLatitude(),currentLocation.getLongitude());
                mMap.addMarker(new MarkerOptions()
                    .position(sydney)
                    .title("mylocation")
                    .flat(true));
            }
        }
    }
}

```

```

        mMap.getUiSettings().setZoomControlsEnabled(true);
        mMap.getUiSettings().setMyLocationButtonEnabled(true);
        mMap.getUiSettings().setCompassEnabled(true);
        mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));
        if (ActivityCompat.checkSelfPermission(MapsActivity.this,
Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(getApplicationContext(),
Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
            return;
        }
        mMap.setMyLocationEnabled(true);
    }
}
});
} catch (Exception e)
{
    finish();
}
}
@Override
public void onMapReady(@NonNull GoogleMap googleMap) {
    mMap = googleMap;
    if(b)
    {
        LatLngKeshpur=new LatLng(22.5558,87.4613);
        if(keshpur)
        {
            MarkerOptions markerOptions4=new MarkerOptions().position(latLngKeshpur).title("Keshpur
police station") .icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
            mMap.addMarker(markerOptions4);
            mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLngKeshpur,16));
        }
        LatLngKgp=new LatLng(22.3380,87.3099);
        if(kgp)
        {
            MarkerOptions markerOptions5=new MarkerOptions().position(latLngKgp).title("Kharagpur police
station") .icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
            mMap.addMarker(markerOptions5);
            mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLngKgp,16));
        }
        LatLngMidnapur=new LatLng(22.430889,87.321487);
        if(midnapore)
        {
            MarkerOptions markerOptions7=new MarkerOptions().position(latLngMidnapur).title("Midnapur
police station") .icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
            mMap.addMarker(markerOptions7);
            mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLngMidnapur,16));
        }
        LatLngGhatal=new LatLng(22.6648,87.7282);
        if(ghatal)
        {

```

```

        MarkerOptions markerOptions9=new MarkerOptions().position(latLngGhatal).title("Ghatal police
station").icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
        mMap.addMarker(markerOptions9);
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLngGhatal));
    }
    latLngPingla=new LatLng(22.2780,87.5862);
    if(pingla)
    {
        MarkerOptions markerOptions10=new MarkerOptions().position(latLngPingla).title("Pingla police
station").icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
        mMap.addMarker(markerOptions10);
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLngPingla));
    }
    latLngTamluk=new LatLng(22.3,87.92);
    if(tamluk)
    {
        MarkerOptions markerOptions6=new MarkerOptions().position(latLngTamluk).title("Tamluk police
station").icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
        mMap.addMarker(markerOptions6);
        mMap.animateCamera(CameraUpdateFactory.newLatLng(latLngTamluk));
    }
    latLngDaspur=new LatLng(22.6074,87.7232);
    if(daspur)
    {
        MarkerOptions markerOptions8=new MarkerOptions().position(latLngDaspur).title("Daspur police
station").icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
        mMap.addMarker(markerOptions8);
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLngDaspur));
    }
    latLngDebra=new LatLng(22.3630,87.5519);
    if(debra)
    {
        MarkerOptions markerOptions1=new MarkerOptions().position(latLngDebra).title("Debra police
station").icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
        mMap.addMarker(markerOptions1);
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLngDebra));
    }
    latLngKolaghat=new LatLng(22.4354,87.8505);
    if(kolaghat)
    {
        MarkerOptions markerOptions3=new MarkerOptions().position(latLngKolaghat).title("Kolaghat
police station").icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility));
        mMap.addMarker(markerOptions3);
        mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLngKolaghat,16));
    }
    latLngPanskura=new LatLng(22.3970,87.7139);
    if(panskura)
    {
        MarkerOptions markerOptions2=new MarkerOptions()
            .position(latLngPanskura)
            .title("Panskura police station");
    }
}

```

```

// .icon(setIcon(MapsActivity.this,R.drawable.baseline_accessibility_new_24));
mMap.addMarker(markerOptions2);
mMap.moveCamera(CameraUpdateFactory.newLatLng(latLngPanskura));
}
double lat=currentLocation.getLatitude();
double log=currentLocation.getLongitude();
if(abs(lat-((double) latLngDebra.latitude))<=distance && abs(log-((double)
latLngDebra.longitude))<=distance)
{
    debra=true;
}
if(abs(lat-((double) latLngKgp.latitude))<=distance && abs(log-((double)
latLngKgp.longitude))<=distance)
{
    kgp=true;
}
if(abs(lat-((double) latLngMidnapur.latitude))<=distance && abs(log-((double)
latLngMidnapur.longitude))<=distance)
{
    midnapore=true;
}
if(abs(lat-((double) latLngPanskura.latitude))<=distance && abs(log-((double)
latLngPanskura.longitude))<=distance)
{
    panskura=true;
}
if(abs(lat-((double) latLngKolaghat.latitude))<=distance && abs(log-((double)
latLngKolaghat.longitude))<=distance)
{
    kolaghat=true;
}
if(abs(lat-((double) latLngKeshpur.latitude))<=distance && abs(log-((double)
latLngKeshpur.longitude))<=distance)
{
    keshpur=true;
}
if(abs(lat-((double) latLngGhatal.latitude))<=distance && abs(log-((double)
latLngGhatal.longitude))<=distance)
{
    ghatal=true;
}
if(abs(lat-((double) latLngDaspur.latitude))<=distance && abs(log-((double)
latLngDaspur.longitude))<=distance)
{
    daspur=true;
}
if(abs(lat-((double) latLngTamluk.latitude))<=distance && abs(log-((double)
latLngTamluk.longitude))<=distance)
{
    tamluk=true;
}
}

```

```

imageButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(b)
        {
            b=false;
            Toast.makeText(getApplicationContext(),"wait for hide police
station",Toast.LENGTH_LONG).show();
        }
        else
        {
            b=true;
            Toast.makeText(getApplicationContext(),"wait for show police
station",Toast.LENGTH_LONG).show();
        }
    }
});
//Dependency to find current location (implementation("com.google.android.gms:play-services-
location:21.3.0"))
}
public BitmapDescriptor setIcon(Activity context,int drawable)
{
    Drawable drawable1=ActivityCompat.getDrawable(context,drawable);
    drawable1.setBounds(0,0,drawable1.getIntrinsicWidth(),drawable1.getIntrinsicHeight());
    Bitmap
    bitmap=Bitmap.createBitmap(drawable1.getIntrinsicWidth(),drawable1.getIntrinsicHeight(),Bitmap.Config.A
RGB_8888);
    Canvas canvas=new Canvas((bitmap));
    drawable1.draw(canvas);
    return BitmapDescriptorFactory.fromBitmap(bitmap);
}
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[]
grantResults, int deviceId) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults, deviceId);
    if(requestCode==FIND_PERMISSION_CODE) {
        if(grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED)
        {
            getLastLocation();
        }
        else
        {
            Toast.makeText(this,"Location permission is denied, please allow the permission to
access",Toast.LENGTH_LONG).show();
        }
    }
}

```

7. Add others guardian's number

```

package com.example.xsefety.First_Activity;
import android.content.Intent;
import android.os.Bundle;

```

```

import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentTransaction;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.RelativeLayout;
import android.widget.TextView;
import android.widget.Toast;
import com.example.xsefety.R;
import java.util.ArrayList;

public class Fragment_Account_Details extends Fragment {
    View view;
    RelativeLayout id_account_details_btn ,id_app_version_btn ,id_logout_btn ,id_update_your_app_btn ,
    id_support_btn;

    TextView profile_name;
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment__account__details, container, false);
        //required for profile name (showing name in account)
        profile_name=(TextView)view.findViewById(R.id.account_text_name);
        MyDbHelper myDbHelper=new MyDbHelper(view.getContext());
        sessionManager s=new sessionManager(view.getContext());
        ArrayList<UserModel> al=new ArrayList<>();

        al.addAll(myDbHelper.getUserAllData(s.getSessionDetails("key_session_phone")));
        if(al!=null) {
            profile_name.setText(al.get(0).getName());
        }
        //profile details clickable button backend code
        id_account_details_btn = (RelativeLayout) view.findViewById(R.id.id_account_details_btn);
        id_account_details_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Fragment account_details_btn = new Fragment_profile_account_details();
                FragmentTransaction fragmentTransaction =
                getActivity().getSupportFragmentManager().beginTransaction();
                fragmentTransaction.replace(R.id.id_home_page_framlayout,
                account_details_btn,null).addToBackStack(null).commit();
            }
        });
        //app update clickable button backend code
        id_update_your_app_btn =(RelativeLayout) view.findViewById(R.id.id_update_your_app_btn);
        id_update_your_app_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Toast.makeText(view.getContext(), "Enter valid Phone Number & Passwrod",
                Toast.LENGTH_SHORT).show();
            }
        });
    }
}

```

```

        Toast.makeText(view.getContext(),"Your App Is Up to Date",Toast.LENGTH_SHORT).show();
    }
});
//app Support clickable button backend code
id_support_btn =(RelativeLayout) view.findViewById(R.id.id_support_btn);
id_support_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //  Toast.makeText(view.getContext(), "Enter valid Phone Number & Passwrod",
        Toast.LENGTH_SHORT).show();

        Toast.makeText(view.getContext(),"Currently Support Not Available",Toast.LENGTH_SHORT).show();
    }
});
//app version clickable button backend code
id_app_version_btn =(RelativeLayout) view.findViewById(R.id.id_app_version_btn);
id_app_version_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(view.getContext(),"You are currently on 1.0 version",Toast.LENGTH_SHORT).show();
    }
});
//logout clickable button backend code
id_logout_btn =(RelativeLayout) view.findViewById(R.id.id_logout_btn);
id_logout_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        startActivity(new Intent(view.getContext(), first_opening_animation.class));
        getActivity().finish();
        sessionManager s=new sessionManager(view.getContext());
        s.logoutSession();
    }
});
return view;
}
}

```

8. MYDBHelper

```

package com.example.xsefety.First_Activity;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import java.util.ArrayList;

public class MySqlHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME="x_sefety";
    private static final String GUARDIAN_TABLE="guardian_table";
    private static final String USER_TABLE="user_table";

```

```

private static final int DATABASE_VERSION=1;
public MyDbHelper(Context context) {
    super(context, DATABASE_NAME, null, DATABASE_VERSION);
}

@Override
public void onCreate(SQLiteDatabase db) {

    String CREATE_USER_TABLE_QUERY="CREATE TABLE "+USER_TABLE+"(id INTEGER PRIMARY KEY
AUTOINCREMENT,name TEXT,phone TEXT,email TEXT,password TEXT,pin TEXT)";
    String CREATE_GUARDIAN_TABLE_QUERY="CREATE TABLE "+GUARDIAN_TABLE+"(id INTEGER PRIMARY
KEY AUTOINCREMENT,phone TEXT,UNIQUE(phone) ON CONFLICT REPLACE)";
    db.execSQL(CREATE_USER_TABLE_QUERY);
    db.execSQL(CREATE_GUARDIAN_TABLE_QUERY);
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

    db.execSQL("DROP TABLE IF EXISTS "+USER_TABLE);
    db.execSQL("DROP TABLE IF EXISTS "+GUARDIAN_TABLE);
    onCreate(db);
}

public boolean InsertUserHelper(String name1,String phone1,String email1,String password1,String pin1)
{
    SQLiteDatabase sqLiteDatabase=this.getWritableDatabase();
    ContentValues contentValues=new ContentValues();
    contentValues.put("name",name1);
    contentValues.put("phone",phone1);
    contentValues.put("email",email1);
    contentValues.put("password",password1);
    contentValues.put("pin",pin1);
    long l=sqLiteDatabase.insert(USER_TABLE,null,contentValues);
    sqLiteDatabase.close();
    return l > 0;
}
public boolean InsertGuardianHelper(String phone1)
{
    SQLiteDatabase sqLiteDatabase=this.getWritableDatabase();
    ContentValues contentValues=new ContentValues();
    contentValues.put("phone",phone1);
    long l=sqLiteDatabase.insert(GUARDIAN_TABLE,null,contentValues);
    sqLiteDatabase.close();
    return l > 0;
}

public ArrayList<UserModel> getUserAllData(String phone1)
{
    ArrayList<UserModel> al=new ArrayList<>();
    SQLiteDatabase sqLiteDatabase=this.getReadableDatabase();
    String query="SELECT * FROM "+USER_TABLE+" WHERE phone='"+phone1+"'";

```

```

Cursor cursor=sqLiteDatabase.rawQuery(query,null);
if(cursor.moveToFirst())
{
    String name=cursor.getString(1);
    String phone=cursor.getString(2);
    String email=cursor.getString(3);
    String password=cursor.getString(4);
    String pin=cursor.getString(5);
    UserModel userModel=new UserModel();
    userModel.setName(name);
    userModel.setPhone(phone);
    userModel.setEmail(email);
    userModel.setPassword(password);
    userModel.setPin(pin);
    al.add(userModel);
}
return al;
}

public ArrayList<UserModel_two> GetAllGuardianPhone()
{
    ArrayList<UserModel_two> al = new ArrayList<UserModel_two>();
    SQLiteDatabase sqLiteDatabase=this.getReadableDatabase();
    Cursor cursor=sqLiteDatabase.rawQuery("SELECT * FROM "+GUARDIAN_TABLE,null);
    if (cursor.moveToFirst())
    {
        do {
            UserModel_two userModel_two=new UserModel_two();
            String phone = cursor.getString(1);
            userModel_two.setGuardian_phone(phone);
            al.add(userModel_two);
        }while (cursor.moveToNext());
    }
    return al;
}
public boolean deleteHelper(String phone1)
{
    SQLiteDatabase sqLiteDatabase=this.getWritableDatabase();
    int i=sqLiteDatabase.delete(GUARDIAN_TABLE,"phone=?",new String[]{phone1});
    sqLiteDatabase.close();
    return i>0;
}
}

```

Testing and Evaluation

7 Testing

The aim of program testing is to help realise and identify all defects in a program. However, in practice, even after satisfactory completion of the testing phase, it is not possible to guarantee that a program is error free. This is because the input data domain of most programs is very large, and it is not practical to test the program exhaustively with respect to each value that the input can assume. We must remember that careful testing can expose a large percentage of the defects existing in a program, and therefore provides a practical way of reducing defects in a system.

A software product is normally tested in three stages:

- **Unit testing:**

Unit testing is undertaken after a module has coded and reviewed. This activity is typically undertaken by the coder of the module himself in the coding phase. Before carrying out unit testing, the unit test cases have to be designed and the test environment for the unit under test has to be developed.

- **White Box Texting:**

- White box testing is an important type of unit testing. A large number of white box testing strategies exist. Each testing strategy essentially designs test cases based on analysis of some aspect of source code and is based on some heuristic.

- **Black Box Testing:**

In black-box testing, test cases are designed from an examination of an input/output values only and no knowledge of design or code is required. The following are the two main approaches available to design black box test cases:

- **Integration testing:**

- Integration testing is carried out after all (or at least some of) the modules have been unit tested. Successful completion of unit testing, to a large extent, ensure that the unit (or module) as a whole work satisfactorily.

8.1.1 Testing Table

NO	Test Cases	Expected Output	Status
1	Register	Successfully save all details and go to the profile page.	Pass
2	Login	Successfully log in and go to the profile page.	Pass
3	Click on "Home" (Main) Button	Successfully open home page.	Pass
4	Click on "Maps" (Main) Button	Successfully open google map.	Pass
5	Click on "Profile" (Main) Button	Successfully open profile page.	Pass
6	Click on "Women Safety" Button	Successfully open SOS page.	Pass
7	Click on "SOS" Button	Successfully get victim current location and call to the first guardian phone, send SMS to all guardian phones and inform to the women safety dept.	Pass
8	Click on "Complaint" Button	Successfully open online complaint Form.	Pass
9	Fill online complaint form then click on "Submit" Button	Successfully send the email to the NCIB.	Pass
	Click on "Ambulance" Button	Successfully open Ambulance Booking (Ambulance Services) page.	Pass
10	Click on "Emergency" ambulance booking button	Successfully booking ambulance. User register details and fetch user current location then all details automatically send to the ambulance service team.	Pass
11	Click on "Booking manually" ambulance booking button	Successfully booking ambulance with manual filling form and all details automatically send to the ambulance service team.	Pass
12	Click on "profile details" button.	Successfully show user all details.	Pass
13	Click on Edit "profile details" button.		Pass
14	Add other guardians phone numbers.	Successfully add guardians mobile number (Maximum four number are added).	Pass

15	Click on “Support” button	Successfully show one toast message for support details.	Pass
16	Click on “About Us” button	Successfully show one toast message for About Us.	Pass
17	Click on “Update Your App” button	Successfully show one toast message “Up To Date”.	Pass
18	Click on “App Version” button	Successfully show one toast message “version 1.0”.	Pass
19	Click on “Log Out” button	Successfully log out the profile page and open the Sign in page.	Pass

Result and discussion

8.1.1 Output and Screenshots



Fig - 1. Opening Animation Page

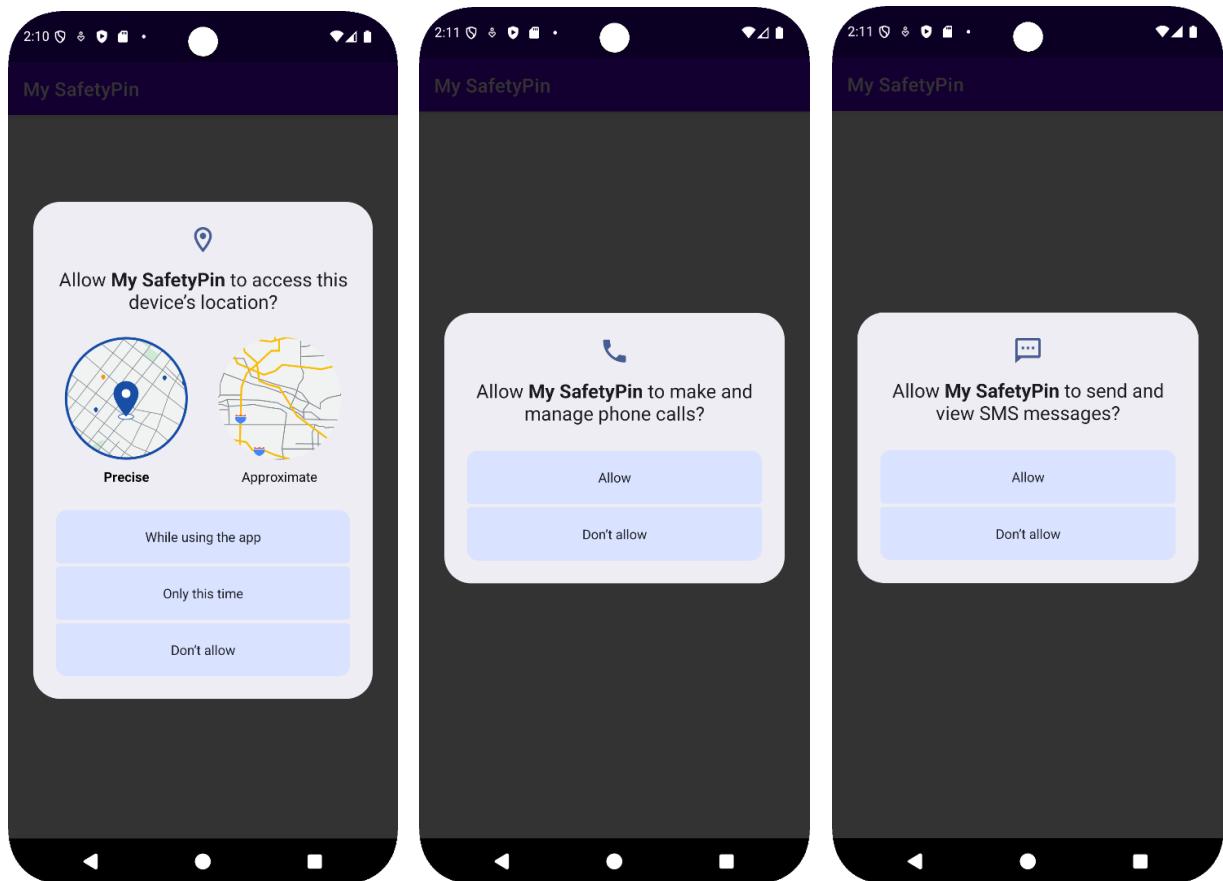


Fig – 2. Permission request pages

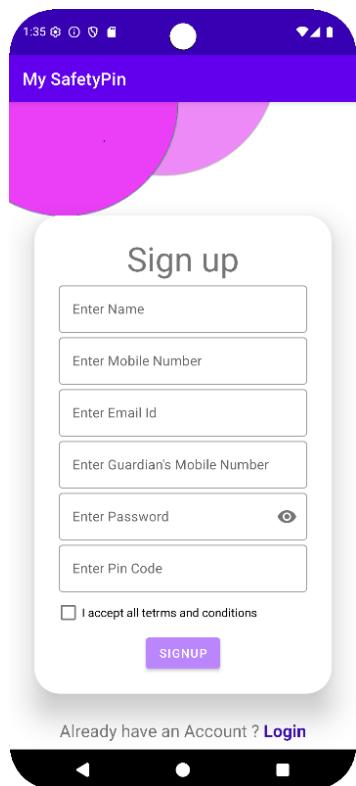


Fig – 3. Blank Signup Page

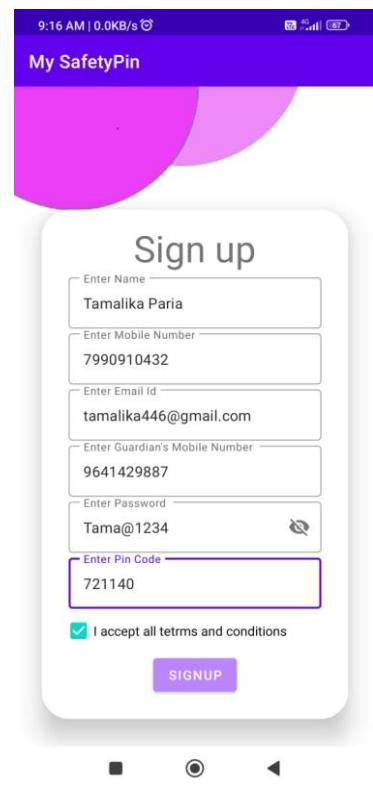


Fig – 4. Fill out the entire sign up form and click the 'Sign Up' button to create an account

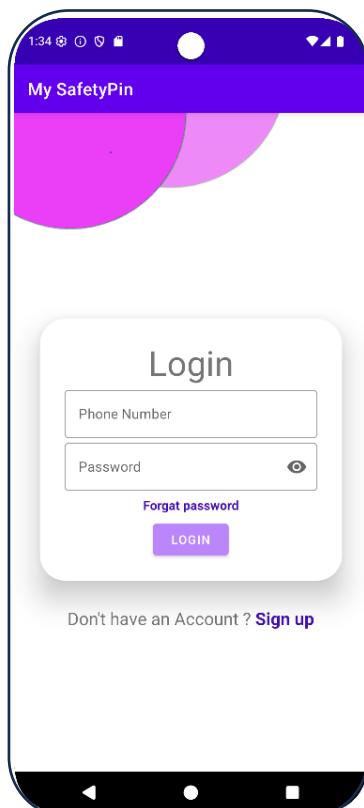


Fig – 5. Blank Login Page

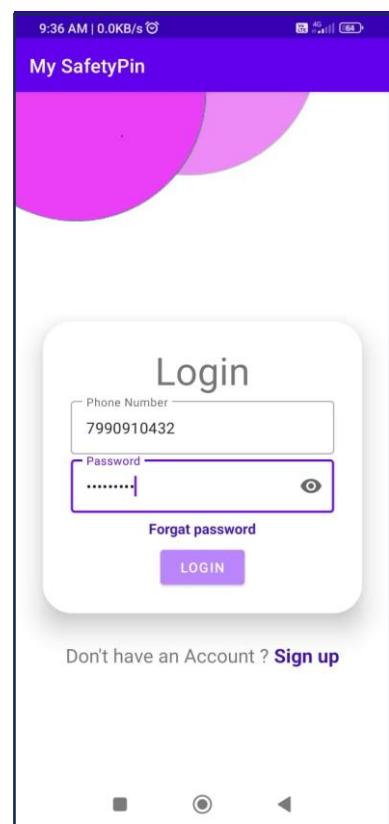


Fig – 6. Enter mobile number and password, then click the 'Login' button to log in.

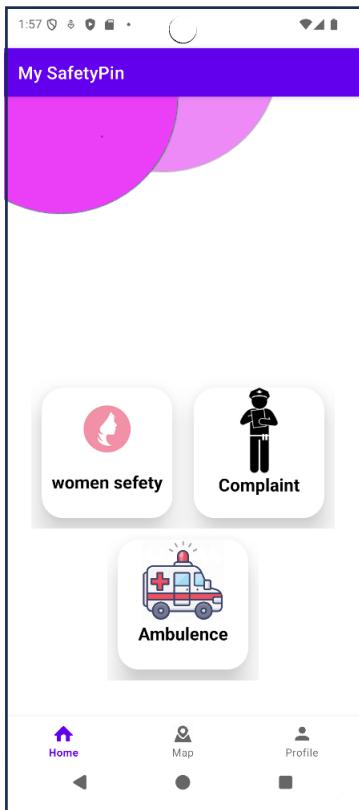


Fig – 7. Home Page

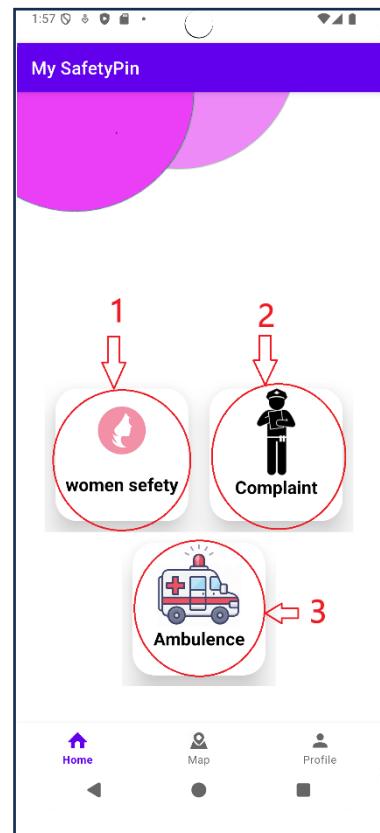


Fig – 8. Overview of the features on the home page

8. If the user clicks the 'Women Safety' Button

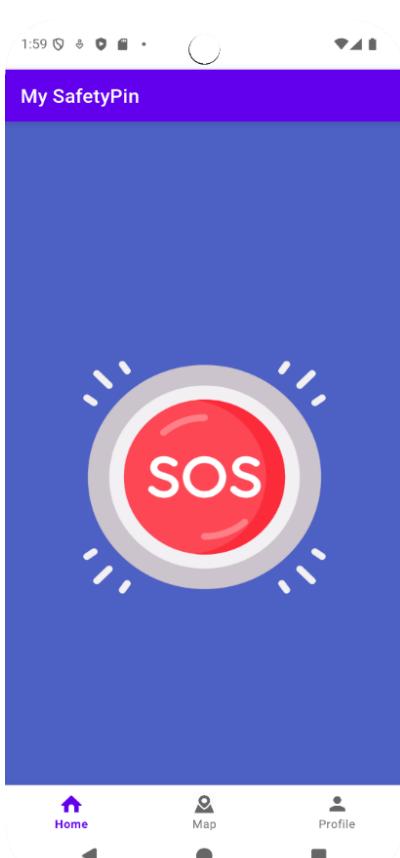


Fig- 9 . SOS Button

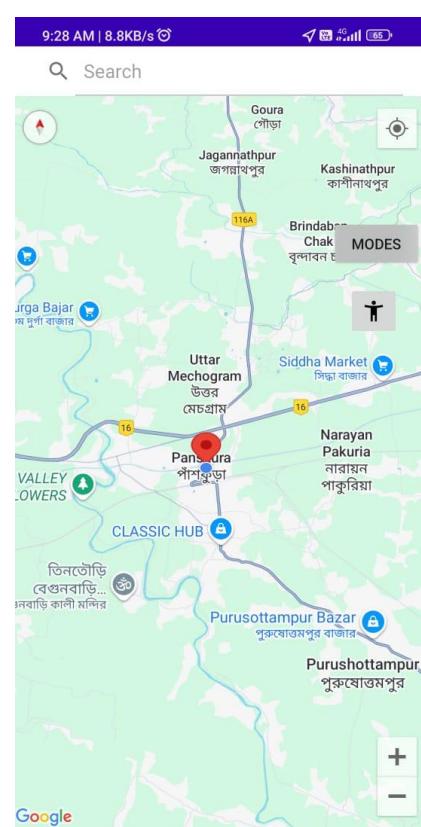


Fig – 10. Fetch Current Location

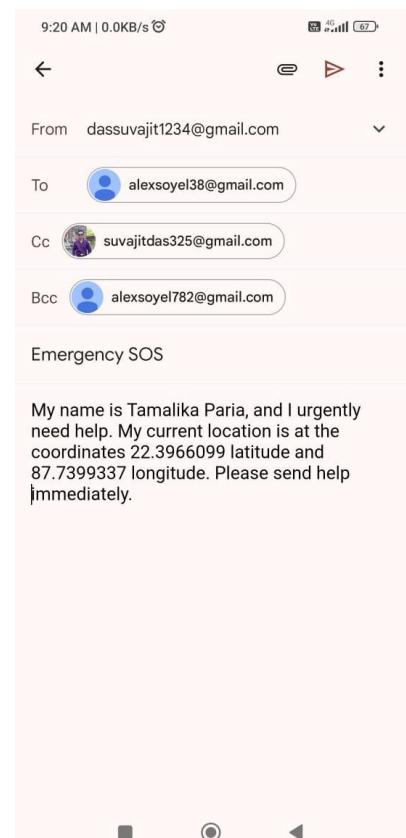
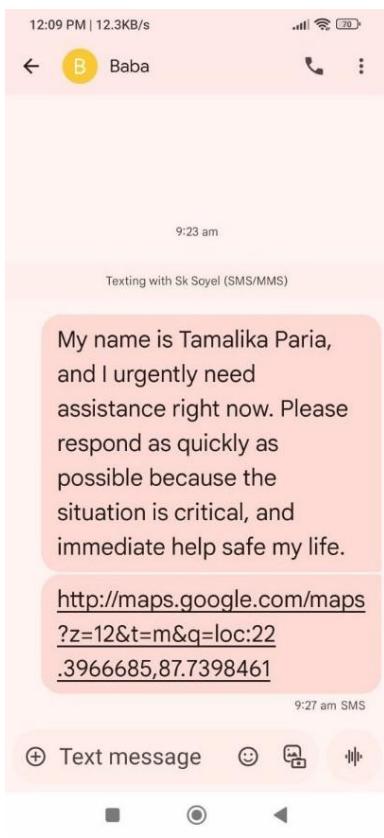


Fig-11. Send Alert SMS to the Guardian.

Fig – 12. Send Email to the Women Safety Dept.

9. If the user clicks the 'Complaint' Button

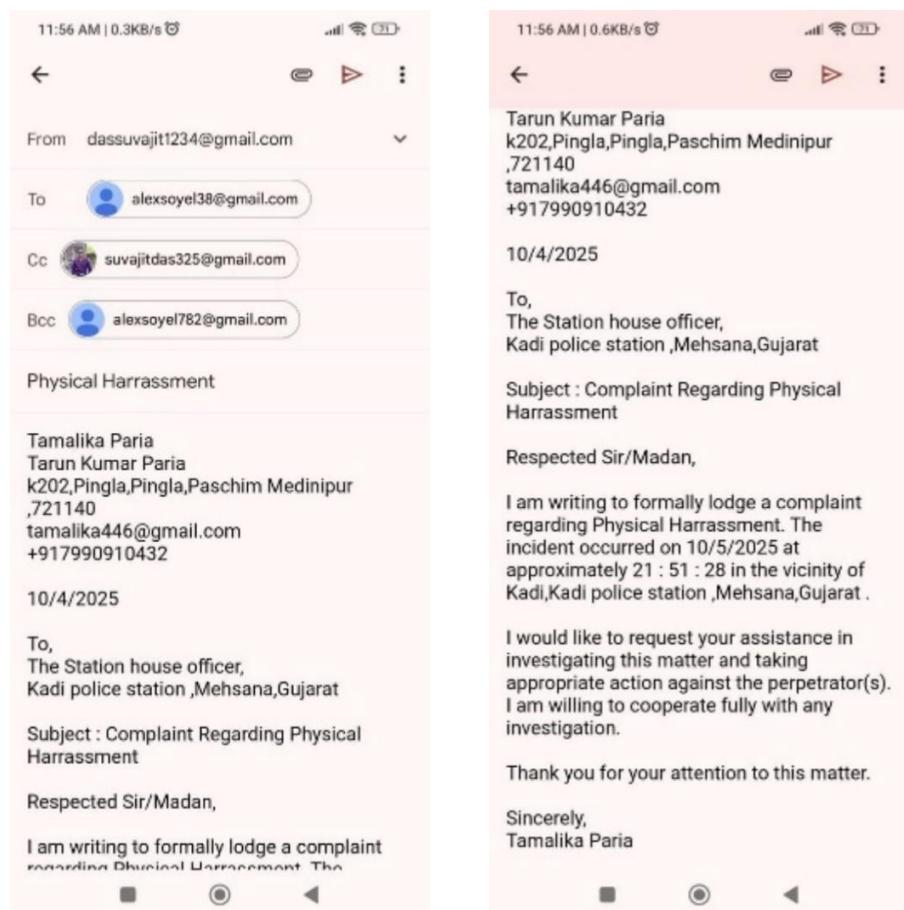
The screenshots show a progression of filling out a 'Complaint Form' on a mobile device. Each screen has a purple header bar with the text 'My SafetyPin'.

- Stage 1:** Shows fields for 'Enter Name', 'Enter Father Name', 'Enter Mobile Number', 'Enter Email Id', and 'Select Gender' (with options MALE, Female, Custom). Below these are fields for 'House/Door/Flat No', 'Village/Town/city', and 'Police Station'.
- Stage 2:** Adds fields for 'District' and 'Pin Code' under 'YOUR DETAILS'. It also includes 'Victims Details' with 'Enter Victims Name' and 'Mobile Number' fields, and 'House/Door/Flat No' and 'Village/Town/city' fields under 'Complaint Form'.
- Stage 3:** Adds 'Incident Details' with 'Incident Place - Vilage/Town/City', 'Incident Police Station', 'Incident District', and 'Incident State' fields. It also includes an 'Explain incident' text area.
- Stage 4:** Adds 'SET DATE' and 'SET TIME' buttons below the 'Explain incident' area. A large 'SUBMIT' button is located at the bottom right of the form.

At the bottom of each screen are navigation icons for 'Home', 'Map', and 'Profile'.

Fig – 13. Blank Online complaint form.

Fig - 14. Fill out the complaint form, then click the Submit button. The fetched details are sent to the NCIB.



10. If the user clicks the 'Ambulance' Button

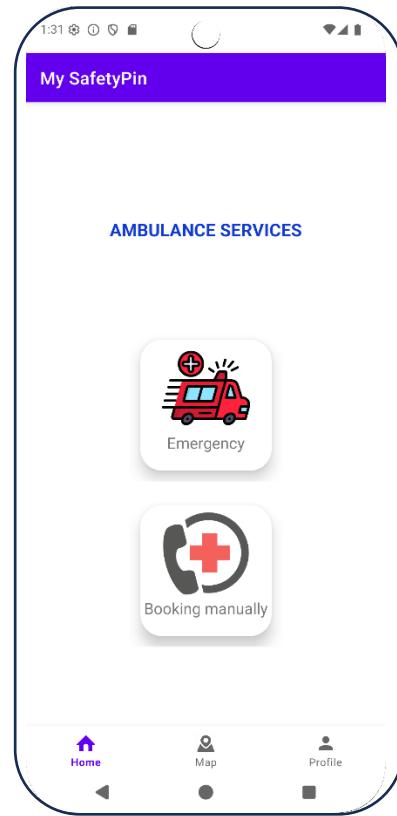


Fig - 16. Ambulance Services Home page.

- If the user clicks the 'Emergency' Button

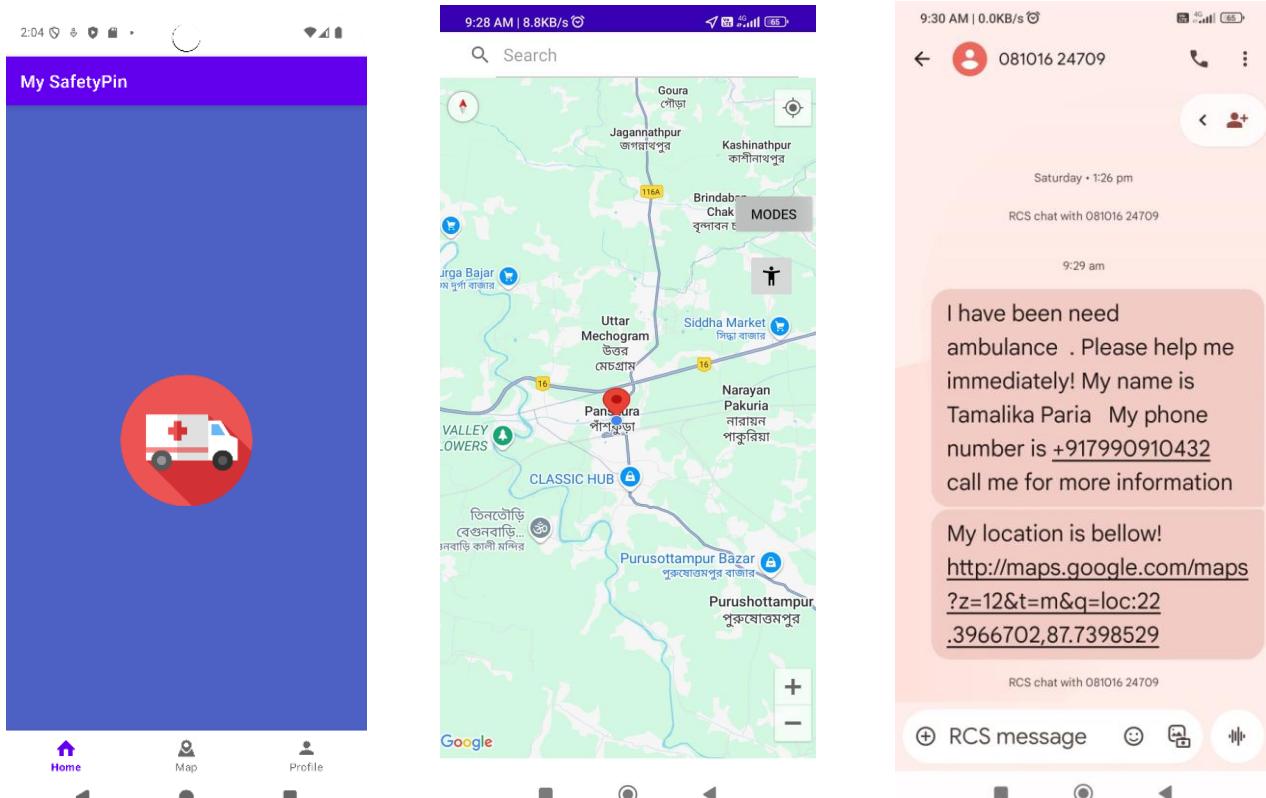


Fig – 17. When the Emergency button is clicked, it fetches the current location and sends a message to the ambulance services team, who take immediate action.

- If the user clicks the 'Booking Manually' Button.

Booking Ambulance

Enter Patient Name
Enter Mobile Number
Enter Email Id
House/Door/Flat No
Village/Town
Police Station
District
Enter Pin Code

Booking Ambulance

House/Door/Flat No
Village/Town
Police Station
District
Enter Pin Code

SET DATE
SET TIME
 I accept all terms and conditions

BOOK NOW

Home Map Profile

Fig – 18. Blank Ambulance Form.

Booking Ambulance

Enter Patient Name
Bela rani patia
Enter Mobile Number
7990910432
Enter Email Id
tamalikaparia24@gmail.com
House/Door/Flat No
101a
Village/Town
pingla
Police Station
pingla
District
paschim Medinipur
Enter Pin Code
721140

Booking Ambulance

House/Door/Flat No
101a
Village/Town
pingla
Police Station
pingla
District
paschim Medinipur
Enter Pin Code
721140

SET DATE 21/5/2025
SET TIME 10 : 10
 I accept all terms and conditions

BOOK NOW

Home Map Profile

Fig- 19. Complete the Booking Ambulance form and click 'Book Now'; the system will automatically send all the details and current location to the ambulance services Authority.

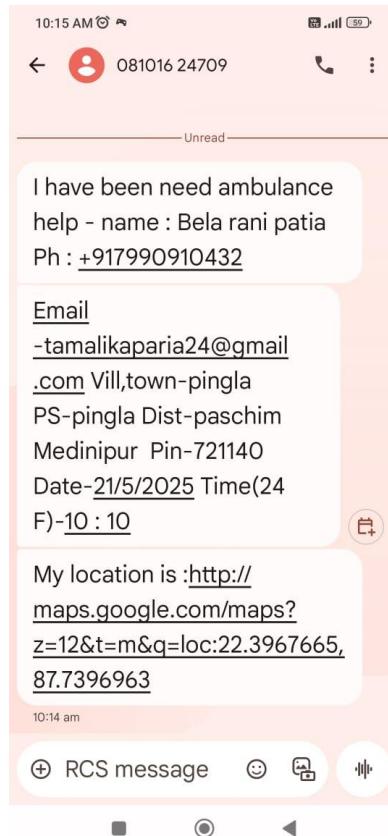


Fig- 20. When an ambulance booking form is submitted, a message is automatically sent to the ambulance services Authority.

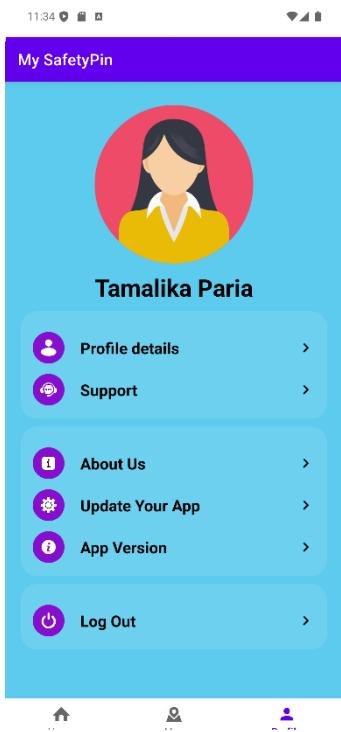


Fig- 21. Profile Page



Fig- 22. Profile Details

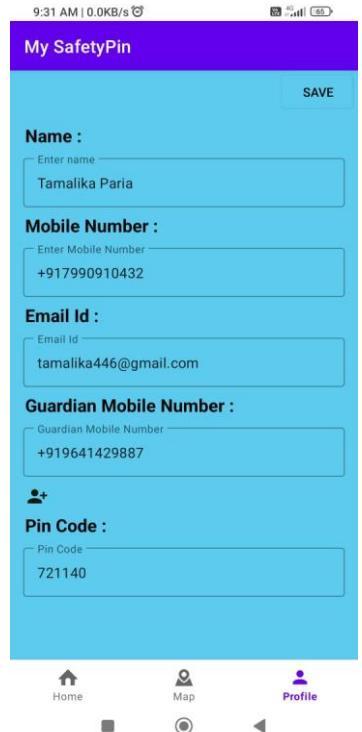
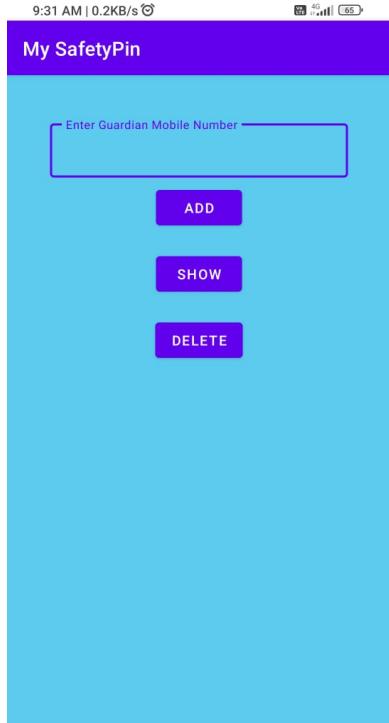


Fig- 23. Edit Profile Details



Home Map Profile

■ ○ <



Home Map Profile

■ ○ <

Fig- 24. Blank Add Guardian number page.

Fig- 25. Add Guardian phone number

9.1 Advantages of Proposed Work

- The proposed women safety Android application is designed with simplicity, speed, and reliability in mind. By focusing on essential functions—**emergency SMS alerts, quick calling, and real-time location sharing via Google Maps**—it offers several key advantages:
 - Quick and Easy to Use
 - The app provides **one-tap or shortcut-based activation**, allowing users to send an alert or make a call **within seconds** during emergencies.
 - Simple user interface (UI) ensures accessibility for users of **all ages and technical backgrounds**.
 - Offline Functionality
 - Since the app relies on **SMS and phone calls**, it **does not require an active internet connection**, making it highly reliable in **low-network or rural areas**.
 - Real-Time Location Sharing
 - Integration with **Google Maps** allows users to share their current location, enabling **faster assistance** by trusted contacts.
 - Minimal Resource Usage
 - The application is **lightweight**, using very little storage, memory, or battery—ideal for low-end Android devices.
 - It doesn't run heavy background services, preserving phone performance.
 - Affordable and Scalable
 - Developed using **freely available Android tools and APIs** (e.g., Google Maps, SMS, Telephony), the app is **cost-effective** to build and maintain.
 - It can be **easily scaled** or customized for specific regions, institutions, or organizations.

- Increased Personal Safety and Peace of Mind
 - Knowing they can quickly alert someone with their location increases **users' confidence and sense of security**.
 - This empowerment is especially critical for women traveling alone, late at night, or in unfamiliar areas.
- No Internet Dependency
 - Unlike many advanced apps that fail without Wi-Fi or mobile data, this app remains **functional as long as there's cellular signal**—making it more dependable in critical situations.
- Customizable for Various Use Cases
 - Can be adapted for use not just by women, but also by elderly people, children, or workers in high-risk environments.

9.2 Limitations

- Dependence on Network Availability
 - Although the app is designed to work without internet by relying on SMS and calls, it still requires a mobile network signal to function. In areas with poor or no cellular coverage:
 - Text messages may not be delivered.
 - Emergency calls may not connect.
 - Location sharing may be delayed or inaccurate.
- Lack of Automation or Smart Triggers
 - The current system depends on manual user input to send alerts or make calls. In situations where the user is physically unable to interact with their phone (e.g., unconscious or restrained), the app cannot function unless externally triggered.
- No Multimedia Evidence Collection
 - Unlike some advanced safety apps, this application does not support audio or video recording, which can be helpful.
- Limited Integration with Authorities
 - The app sends alerts to personal emergency contacts but lacks direct integration with police departments, hospitals, or official emergency services. This may delay professional intervention in critical situations.

- No Feedback or Confirmation Mechanism
 - Once an SOS message or call is sent, the app does not confirm whether:
 - The message was received successfully.
 - The contact has responded.
 - The user is now safe.
 - This lack of feedback loop limits the app's usefulness in managing ongoing safety status or escalating alerts if needed.
- Not Context-Aware
 - The app does not use AI, behavior patterns, or environmental inputs to detect potential danger or preemptively suggest action (e.g., entering a high-risk area). This limits its ability to offer preventive safety features.
- Deterring threats.
- Collecting evidence for legal follow-up.
- Alerting responders to the environment/context of the emergency.

9.3 Future work

- Artificial Intelligence for Threat Detection
 - Future versions can integrate AI models to analyse user behaviour patterns and detect potential threats in real-time. For example, sudden changes in location or voice tone (panic or distress) can automatically trigger alerts.
- Integration with Wearable Devices
 - By connecting the app with smartwatches or fitness bands, emergency features like SOS can be activated via gesture recognition (e.g., tap or shake), making distress calls more discreet.
- Real-Time Location Sharing and Tracking with Authorities
 - Enhancing the app to share live GPS data directly with nearby police stations or verified authorities during emergencies can drastically improve response time.
- Crowdsourced Safety Reporting
 - Users can report unsafe areas or incidents in real-time, helping build a crowdsourced "safety map" of the city. Other users can then avoid these areas or take precautions.
- Voice-Activated Emergency Triggers
 - In situations where users can't touch their phone, voice commands (e.g., saying a specific phrase) can be used to silently trigger SOS alerts.

- Augmented Reality (AR) Navigation to Safe Spots
 - AR-based navigation can guide users to the nearest safe zone, police station, hospital, or shelter, particularly useful in unfamiliar environments.
- Offline Emergency Mode
 - Implementing an offline SOS mode using SMS and low-energy Bluetooth can help in areas with poor internet connectivity.
- AI Chatbot for Immediate Guidance
 - An in-app AI chatbot can offer immediate guidance during a crisis, such as recommending actions, calming the user, or connecting with support services.
- Multi-Language and Accessibility Features
 - Expanding support for regional languages and adding features for visually or hearing-impaired users will make the app more inclusive and accessible.
- Confirmation from the ambulance service authority
 - Confirmation from the ambulance service authority, along with real-time ambulance location tracking for the user.

CONCLUSION

This application is still under trial process and needs to go through the non-functional test which will be carried out soon. However, three objectives in this project which are to design a women's safety mobile application (My SafetyPin) using the Geolocation method to track the current location, to develop an android application for women's safety and to test the functionality of this mobile application successfully achieved. There are still some limitations to this application. One of the limitations is that an internet connection is required to use the application. Internet is essential when using this application because it uses GPS to get the user's current location. If there is no internet, GPS was unable to read the location. The second limitation is to always make sure the phone has sufficient credit balance (if prepaid or Postpaid) to send SMS and make emergency calls. Otherwise, the user cannot send the message and calls to all the contacts saved in the application. In future, there are some recommendations that can be added to the application for improvement. First recommendation is that this application can be used even without an internet network.

BIBLIOGRAPHY

8 Books & Research Papers:

- Tiwari, A., & Parihar, S. (2020). *Women Safety Mobile Applications: A Survey of Android Apps*. International Journal of Innovative Research in Computer and Communication Engineering, 8(5), 2675-2680.
- Sharma, R., & Kaur, G. (2018). *An Intelligent Android App for Women Safety Based on Location Tracking and SMS Alert*. International Journal of Computer Sciences and Engineering, 6(3), 157–162.
- Singh, D., & Joshi, A. (2017). *Review Paper on Women's Safety App*. International Journal of Advanced Research in Computer Science, 8(5), 1212–1215.

9 Android Development & Technical Resources:

- Android Developers. (n.d.). *Building Your First App*. Retrieved from <https://developer.android.com>
- Vogella, L. (2018). *Android Programming: Develop Android Apps Using Java and Eclipse*. Pearson Education.
- Android Developers. (n.d.). *Location and Maps*. Retrieved from <https://developer.android.com/training/location>

10 Websites and Reports:

- Ministry of Women and Child Development, Government of India. (n.d.). *Women Safety Initiatives*. Retrieved from <https://wcd.nic.in>
- National Crime Records Bureau (NCRB). (2022). *Crime in India*. Retrieved from <https://ncrb.gov.in>
- UN Women. (n.d.). *Safe Cities and Safe Public Spaces*. Retrieved from <https://www.unwomen.org>

11 Apps and Case Studies:

- Google Play Store. (n.d.). *Safetipin - Personal Safety App*. Retrieved from <https://play.google.com/store/apps/details?id=com.safetipin.android>
- Raksha – Women Safety Alert App. (n.d.). Retrieved from <https://play.google.com/store/apps/details?id=com.raksha.app>