MAIN PROJECT REPORT

ON

SAFE PULSE : REAL TIME HUMAN TRACKING AND EMERGENCY ALERT SYSTEM

Submitted By

JITHIN T (CEC23MCA-2021)

in partial fulfillment for the award of the degree of

Master of Computer Applications

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ENGINEERING, CHERTHALA

ALAPPUZHA - 41

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

MAY 2025

MAIN PROJECT REPORT ON

SAFE PULSE : REAL TIME HUMAN TRACKING AND EMERGENCY ALERT SYSTEM

Submitted By

JITHIN T (CEC23MCA-2021)

in partial fulfillment of the requirements for the award of the degree of

Master of Computer Applications
of
APJ Abdul Kalam Technological University

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ENGINEERING, CHERTHALA

ALAPPUZHA-41

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

MAY 2025

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ENGINEERING, CHERTHALA

ALAPPUZHA-688541

C E R T I F I C A T E

This is to certify that, the project report titled **"SAFE PULSE : REAL TIME HUMAN TRACKING AND EMERGENCY ALERT SYSTEM"** is a bonafide record of the **20MCA246 Main Project** presented by **JITHIN T (CEC23MCA-2021)**, Fourth semester Master of Computer Application student, under our guidance and supervision, in partial fulfillment of the requirements for the award of the degree, **Master of Computer Application** of **APJ Abdul Kalam Technological University during the academic year 2024-2025**.

Guide	**Co-ordinator**	**HOD**
Dr. Preetha Theresa Joy	**Mrs.Renjusha Aravind**	**Dr. Preetha Theresa Joy**
Professor	Assistant Professor	Professor
Dept.of Computer Engg.	Dept.of Computer Engg.	Dept.of Computer Engg.

ACKNOWLEDGEMENT

This work would not have been possible without the support of many people. First and the foremost, we give thanks to Almighty God who gave us the inner strength, resource and ability to complete our project successfully.

I, Jithin T would like to thank **Dr. Jaya V.L**, our Principal, who has provided with the best facilities and atmosphere for the project completion and presentation.I would also like to thank our HoD **Dr. Preetha Theresa Joy** (Professor, Department of Computer Engineering), our project coordinator **Mrs.Renjusha Aravind** (Assistant Professor, Department of Computer Engineering), and our guide **Dr. Preetha Theresa Joy** (Professor, Department of Computer Engineering) for the help extended and also for the encouragement and support given to us while doing the project.

I would like to thank my dear friends for extending their cooperation and encouragement throughout the project work, without which we would never have completed the project this well.Thank you all for your love and also for being very understanding.

DECLARATION

I hereby declare that the project "**SAFE PULSE : REAL TIME HUMAN TRACKING AND EMERGENCY ALERT SYSTEM**" is a bonafide work done by me during the academic year 2024-2025 under the guidance of **Dr. Preetha Theresa Joy**, Assistant Professor at College of Engineering, Cherthala and this report has not been previously formed the basis for the award of any degree, diploma, fellowship or any other similar title or recognition in any other university.

JITHIN T (CEC23MCA-2021)

(07/04/2025)

ABSTRACT

SafePulse is a mobile-based safety application designed to provide real-time protection and emergency support for users, including children, elderly individuals, specially-abled persons, and those in high-risk situations. The app integrates features like fall detection, geo-fencing, SOS flashlight, emergency contact alerts, and unauthorized access detection. It leverages Android technologies, Google Maps API, Firebase, and SMS services to ensure quick response and continuous monitoring. SafePulse enhances personal safety through proactive alerts and seamless user interaction, offering a reliable platform for real-time emergency management.

Contents

List of Figures

Chapter 1

INTRODUCTION

SafePulse is a real-time human tracking and emergency alert system designed to enhance personal safety through smart monitoring and rapid response features. It combines fall detection, location tracking, and emergency communication to ensure timely assistance in critical situations. The app detects emergencies such as sudden phone falls and immediately sends alert messages along with the user's live location and travel history to pre-registered emergency contacts. If no response is received, the system initiates crowd-sourced assistance by notifying nearby SafePulse users within a specific radius, enabling community-driven support. Additionally, the app includes an SOS flashlight signal to help users silently alert others in emergencies.

To further strengthen safety, SafePulse offers Geo-Fencing for defining safe zones such as home, school, or workplace. The app monitors user movement and sends alerts if the user fails to return to these zones within a predefined timeframe—making it ideal for children, elderly individuals, or people in high-risk environments. It also includes an Unauthorized Access Detection feature, which captures and sends the intruder's photo and location to emergency contacts after multiple failed unlock attempts. With these features, SafePulse provides a comprehensive and proactive safety solution tailored for real-world use.

Chapter 2

PROBLEM STATEMENT

2.1 Problem Statement

This project focuses on developing an Android-based mobile application, SafePulse, for real-time human tracking and emergency alert management. Built using Android Studio and Kotlin, the system leverages sensor data and geolocation services to detect critical situations such as falls, unauthorized access, or unexpected device shutdowns. Upon detection, the app automatically alerts pre-registered emergency contacts with the user's live location and recent travel history, ensuring rapid response during emergencies.In addition, the system integrates features like Geo-Fencing for defining safe zones, an SOS flashlight signaling mechanism, and crowd-sourced assistance to involve nearby users when emergency contacts are unavailable. Unauthorized access attempts trigger the front camera to capture the intruder's image and send it along with location data to emergency contacts. By combining real-time safety monitoring, smart alerting, and community-driven support, SafePulse enhances personal security, especially for children, elderly individuals, and people in high-risk environments.

2.2 Objective

The objective of the SafePulse project is to develop a real-time safety and emergency alert mobile application using Android Studio and Kotlin. It aims to detect critical situations, send automatic alerts with live location, and provide proactive security through features like fall detection, geo-fencing, unauthorized access detection, and community support. The system aims to:

1. **To develop a real-time safety and emergency alert system :**using Android Studio and Kotlin that ensures quick and automatic response during critical situations such as falls and unauthorized access.

2. **To implement fall detection and emergency alert features:**that automatically notify emergency contacts with the user's live location and travel history without requiring manual input.

3. **To integrate Geo-Fencing functionality:**allowing users to define safe zones (e.g., home, school, workplace) and receive alerts if the user does not return to the designated area within a specified timeframe.

4. **To enhance device security through unauthorized access detection:** capturing and sending the intruder's image and location to emergency contacts after multiple failed unlock attempts.

5. **To provide an SOS flashlight signaling feature:**allowing users to visually signal distress in silent or critical situations.

Chapter 3

EXISTING SYSTEM

The existing system typically include basic safety and emergency features, but lack the comprehensive, integrated, and proactive safety mechanisms that SafePulse offers. Here are some points on previously existing systems.

3.1 Basic SOS Apps:

- Sent emergency messages manually via button press.
- No fall detection or automatic alert triggering.

3.2 GPS Tracking Apps:

- Provided location sharing only when user actively sent location.
- No geofencing or time-bound safety zone alerts.

3.3 No Unified Platform:

- Most safety apps offer basic SOS buttons and manual emergency alerts only.
- No intruder photo capture or ride start alert mechanisms.

Chapter 4

LITERATURE SURVEY

4.1 Base Paper 1

Enhanced Security Fencing System with Geolocation Tracking[1]- The paper titled "Enhanced Security Fencing System with Geolocation Tracking" presents a modern approach to surveillance by integrating traditional fencing systems with geolocation-based contact connecting mapping technology. It focuses on enhancing security in sensitive areas such as healthcare facilities and jail asylums, where monitoring individual movements and preventing unauthorized access or escape is critical. The system enables real-time location tracking and mapping of individuals within a confined area, providing a proactive solution to safety concerns. It also addresses challenges such as threats to surrounding individuals, unauthorized movement, and system maintenance. Overall, the paper highlights how combining physical barriers with smart tracking technologies can significantly improve situational awareness and security management.

4.2 Base Paper 2

Security in Mobile Network: Issues, Challenges, Solutions[2]- The paper titled "Security in Mobile Network: Issues, Challenges, Solutions" explores the growing security risks associated with mobile devices and applications, highlighting the rapid increase in mobile usage and its vulnerability to various cyber threats. It provides a comprehensive overview of common mobile security problems, such as data breaches, privacy concerns, and unauthorized access, along with defensive mechanisms to safeguard user data and maintain trust. This is highly relevant to the SafePulse project, as it underscores the importance of integrating strong security measures—such as unauthorized access detection, secure data transmission, and privacy protection—within mobile safety applications. By addressing these concerns, SafePulse ensures that its emergency alert system not only provides real-time assistance but also safeguards the sensitive location and personal data of its users.

4.3 Base Paper 3

Smart Device Fall Detection System for Real-Time Monitoring and Emergency Response for Specially Abled Individuals[3]- The paper titled "Smart Device Fall Detection System for Real-Time Monitoring and Emergency Response for Specially Abled Individuals" presents a sensor-integrated system designed to detect falls in real time and respond rapidly with emergency alerts. Using an ESP32 microcontroller connected to various sensors (accelerometer, gyroscope, pressure, cardiotachometer, and GPS), the system achieves high accuracy and a fast response time of 1.45 seconds, making it effective for protecting elderly and specially-abled individuals. This research is closely related to the SafePulse project, which also incorporates real-time fall detection and emergency alerting features. While the paper focuses on wearable hardware, SafePulse implements fall detection via smartphone sensors using Android-based logic, delivering alerts with live location to emergency contacts and nearby users. Both systems share the common goal of enhancing user safety through automated monitoring and quick response, making this study a strong technical foundation for validating and improving SafePulse's fall detection functionality.

4.4 Base Paper 4

Fall Detector Device for Improved Safety and Independence[4]- The paper titled "Fall Detector Device for Improved Safety and Independence" introduces a wearable wrist-worn fall detection device designed to support the elderly and individuals with limited mobility. It uses advanced sensors like gyroscopes and vibration motion shock sensors to detect falls accurately while minimizing false alarms. Integrated with an Android application, it ensures real-time alerts are sent to caregivers or emergency services, enhancing responsiveness and user safety. This study aligns closely with the SafePulse project, which also provides real-time fall detection and emergency notifications through an Android app. While the paper focuses on hardware wearables, SafePulse achieves similar goals using mobile phone sensors, offering a more accessible solution for users without specialized equipment. Both approaches emphasize timely intervention, independence, and improved quality of life for vulnerable populations.

4.5 Base Paper 5

The European emergency number 112: Exploring the potential of crowd-sourced information for emergency management[5]- The study "The European Emergency Number 112: Exploring the Potential of Crowd-Sourced Information for Emergency Management" highlights the importance of combining structured emergency call systems with citizen-generated data to enhance situational awareness during crises. It found that while social media offers large volumes of real-time information, it often lacks accuracy and requires significant processing. In contrast, data collected through emergency numbers like 112 is more reliable and actionable. This insight reinforces the approach taken by the SafePulse app, particularly its "Crowd-Sourced Assistance" feature, which combines automated fall detection and real-time location alerts with a network of nearby users within a 5 km radius. By structuring alerts through sensor-based inputs and verifying responses through app-based confirmations, SafePulse ensures that crowd-sourced help is both timely and trustworthy. Integrating lessons from the 112 system, SafePulse emphasizes a hybrid emergency response model that leverages both mobile technology and community support to improve safety and responsiveness for vulnerable users.

Chapter 5

PROPOSED SYSTEM

5.1 Solution

SafePulse app is a real-time safety and emergency response system designed to protect specially-abled individuals, elderly people, children, and those in high-risk situations. The system leverages mobile technology, smart sensors, and community support features to provide timely assistance and improve personal safety.The core of the proposed system integrates the following smart features:

1. **Fall Detection System:**

   Using built-in mobile sensors like accelerometers and gyroscopes, the app detects sudden falls. Upon detection, it automatically sends real-time alerts and the user's live location to up to five emergency contacts and nearby SafePulse users within a 5 km radius for rapid assistance.

2. **Unauthorized Access Detection:**

   The app includes a 4-digit security lock. After three failed unlock attempts, it captures a photo of the intruder using the front camera, saves it to the phone gallery, and sends the image via SMS to registered emergency contacts.

3. **Journey Monitoring with Start Ride Feature:**

   When the user taps the “Start Ride” button, the app sends a message to emergency contacts with the user’s current location and indicates the journey has begun. This ensures someone is informed of the user’s movement in real-time.

4. **Geo-Fencing and Safe Zones:**

   Users can define safe zones such as home, school, or workplace using Google Maps. If the user does not enter the safe zone , an alert message is sent to emergency contacts. The system also alerts when the user enters a safe zone.

5. **SOS Flashlight Functionality:**

   The SOS Flashlight feature in the SafePulse app is designed to provide a silent yet powerful way to signal distress in critical situations. When activated, the phone’s flashlight blinks in the internationally recognized SOS Morse code pattern (··· —— ···), enabling users to visually alert nearby individuals or rescuers even in complete darkness or low-visibility environments. This functionality is especially useful when the user is unable to speak or move freely, offering a quick and accessible method to draw attention and indicate the need for urgent help.

6. **Crowd-Sourced Assistance:**

   In emergencies, especially when falls are detected, the app notifies SafePulse users within a 5 km radius along with the emergency contacts. This local network of users helps increase the chances of someone nearby providing assistance quickly.

Chapter 6

SYSTEM REQUIREMENT SPECIFICATION

6.1 Overall Description

SafePulse is a real-time safety and emergency response app designed for specially-abled individuals, elderly people, children, and those in high-risk situations. It uses mobile sensors, GPS, and SMS to detect emergencies like falls or unauthorized access and automatically notifies emergency contacts with real-time location updates. Features include fall detection, safe zone alerts, ride monitoring, crowd-sourced assistance, and an SOS flashlight that blinks in Morse code. The app is user-friendly, reliable, and built to provide proactive safety support anytime, anywhere.

6.1.1 Product Perspective

SafePulse is a standalone Android application that functions as a personal safety tool. It integrates with smartphone hardware such as GPS, accelerometer, gyroscope, and flashlight to monitor user activity and detect emergency events. The app does not rely on external hardware but can operate alongside cloud services (e.g., Firebase) for storing emergency contacts and sending alerts. Designed for ease of use and real-time responsiveness, SafePulse aims to enhance personal safety without requiring continuous user interaction.

6.1.2 Product Function

The system provides the following key functions:

- User Registration and Authentication
- Emergency Contact Management
- Fall Detection System
- Geo-Fencing for Safe Zones
- Unauthorized Access Detection
- SOS Flashlight Functionality
- Crowd-Sourced Assistance
- Start Ride Alert System

6.2 System Requirements

- **Operating System:** Android 8.0 or Higher
- **Development Platform:** Android Studio (latest version)
- **Programming Language:** Kotlin
- **Backend:** Firebase (Authentication, Realtime Database, Firestore, Cloud Functions)
- **Map Integration:** Google Maps API
- **SMS And Location:** Android Location Services, SMS Manager API
- **Camera Access:** Android CameraX API

6.2.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Android application development, developed by Google. It offers a complete suite of tools for designing, coding, testing, and debugging Android apps. SafePulse is developed using Android Studio due to its powerful features such as real-time code editing, XML-based UI design, built-in Android emulator, and seamless integration with Firebase and Google APIs. These capabilities enable efficient development and testing of SafePulse's safety features, including location tracking, emergency alerts, and sensor-based functionalities.

6.2.2 Kotlin

Kotlin is a modern, concise, and statically typed programming language officially supported by Google for Android app development. It offers interoperability with Java, enhanced readability, and safer code with reduced boilerplate. SafePulse is developed using Kotlin to take advantage of its robust features, which support efficient implementation of real-time safety functionalities, such as emergency alerts, location tracking, and sensor integrations.

6.2.3 Firebase

Firebase is a platform developed by Google that provides a variety of tools and services for app development, including real-time databases, authentication, cloud messaging, and analytics. In the SafePulse app, Firebase is used to manage user data securely, authenticate users, store emergency contact information, and support real-time notifications and alerts, ensuring a responsive and reliable safety experience.

6.2.4 Google Maps API

Google Maps API is integrated into the SafePulse app to support real-time location tracking and geofencing functionalities. It enables users to define Safe Zones using map-based selection and helps the app monitor whether the user enters or exits these predefined zones. This feature enhances user safety by providing timely alerts based on location changes, especially for children, elderly individuals, and those in high-risk environments.

6.2.5 SMS and Location Services

The SafePulse app uses Android Location Services to accurately track the user's real-time location. In case of an emergency, it leverages the SMS Manager API to send automated alert messages, along with the live location link, to the user's pre-added emergency contacts. This ensures timely communication even without internet access, enhancing reliability in critical situations.

6.2.6 Android CameraX API

The SafePulse app utilizes the Android CameraX API to capture images discreetly in response to unauthorized access attempts. When multiple failed app lock attempts are detected, the front camera is triggered to take a photo of the intruder. This image is then saved to the device gallery and sent to the user's emergency contacts, enhancing security and evidence collection.

6.3 Hardware Requirements

To ensure optimal functionality, the SafePulse app requires a smartphone with essential hardware components, including a gyroscope and accelerometer for fall detection, GPS module for real-time location tracking, a camera (front-facing) for intruder capture, flashlight for SOS signaling, and internet connectivity (Wi-Fi or mobile data) for cloud integration and communication. The device should also support SMS capabilities to send alerts to emergency contacts.

Chapter 7

SYSTEM DESIGNS

7.1 Use-Case Diagram

The use case diagram for the SafePulse app illustrates the interaction between the user and the core safety features of the system. It shows how users can perform actions such as adding emergency contacts, starting a ride, setting a geofence, triggering an SOS alert, and more. It also represents automated system actions like fall detection, unauthorized access alerts, and location tracking.

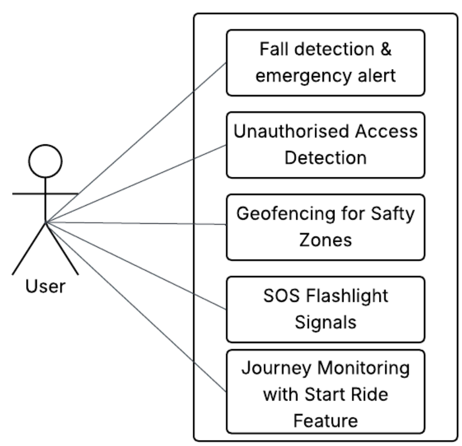

Fig. 7.1: Use case diagram

7.2 System Architecture

- The SafePulse system architecture is designed with a layered approach to ensure efficient interaction between the user interface, application logic, hardware sensors, and external services. Users interact with key safety features such as Fall Alert, Start Ride, SOS Flashlight, Set Safe Zone, and App Lock. These inputs trigger various application logic components including fall detection, intruder detection, geo-fencing timers, ride monitoring, and SMS alert mechanisms.

  The app leverages device hardware like the gyroscope, GPS, front camera, and flashlight for real-time monitoring and action. In parallel, it integrates with external services such as Firebase Realtime Database, Google Maps API, Android Location Services, SMS Manager API, and CameraX API for data storage, location tracking, and message dispatch. This architecture ensures real-time safety alerts and efficient emergency response with minimal user effort.

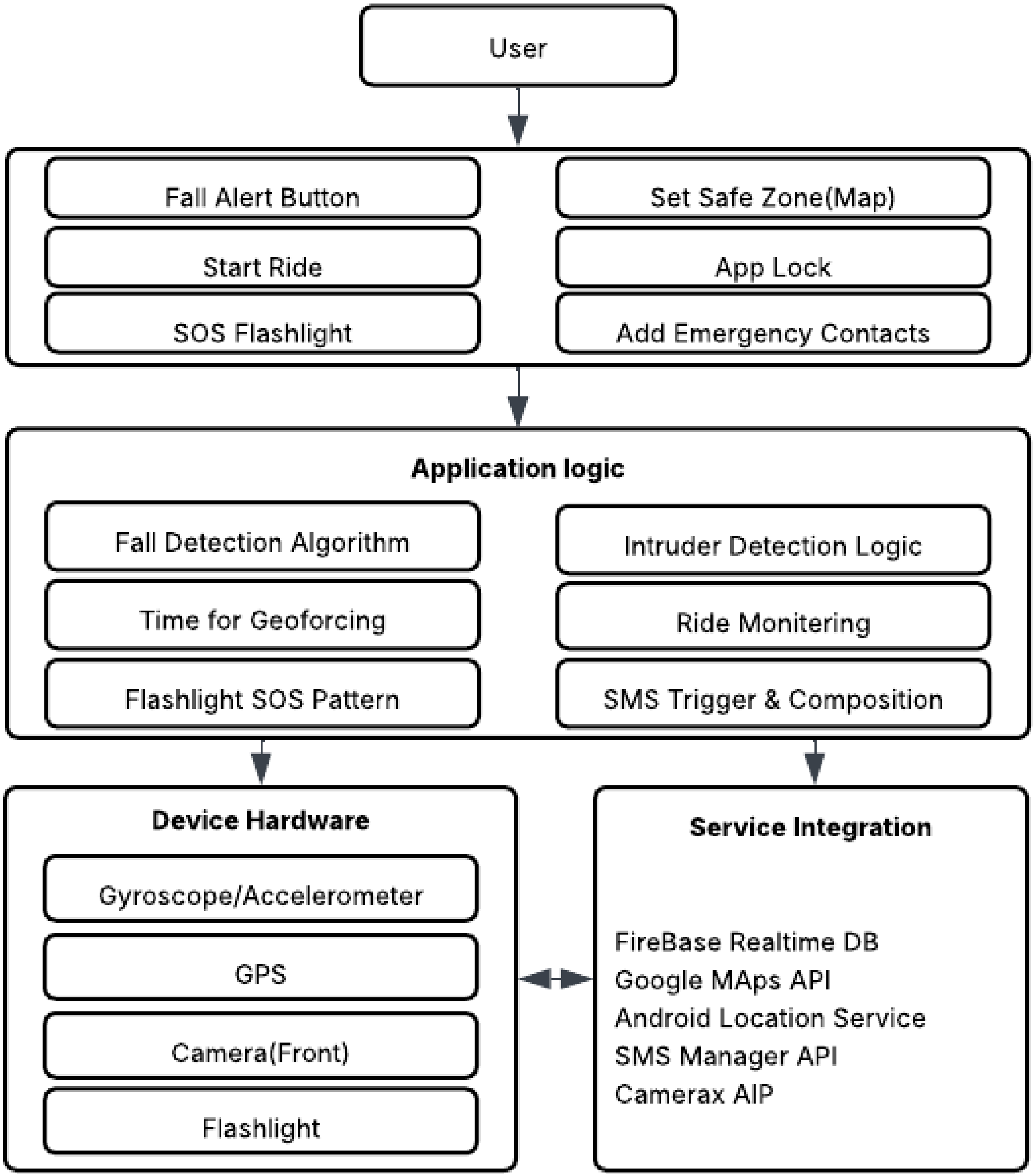

Fig. 7.2: Architecture

Chapter 8

IMPLEMENTATION

The implementation of the SafePulse app was carried out using Android Studio with Kotlin as the primary programming language. The app integrates multiple Android APIs such as CameraX for unauthorized access detection, SMS Manager and Location Services for sending alerts with real-time location, and the Google Maps API for geofencing and tracking purposes. Firebase is used for user authentication and cloud data storage. The modular approach in the app's architecture allows each feature—such as fall detection, SOS flashlight, Safe Zone alerts, and crowd-sourced assistance—to function independently while sharing critical data. The UI is designed to be intuitive and accessible for children, elderly users, and people with special needs, ensuring ease of use in emergency situations.

8.1 Coding Environment Used

The SafePulse application was developed using Android Studio, the official Integrated Development Environment (IDE) for Android development. The project utilized Kotlin as the primary programming language, offering modern syntax and improved safety over Java. The app leverages Android SDK tools, Emulators, and Gradle for build automation. Firebase services were integrated for backend functionalities such as authentication and real-time database management. The development environment also incorporated Google Maps API, CameraX, SMS Manager, and Location Services APIs for implementing safety and alert features.

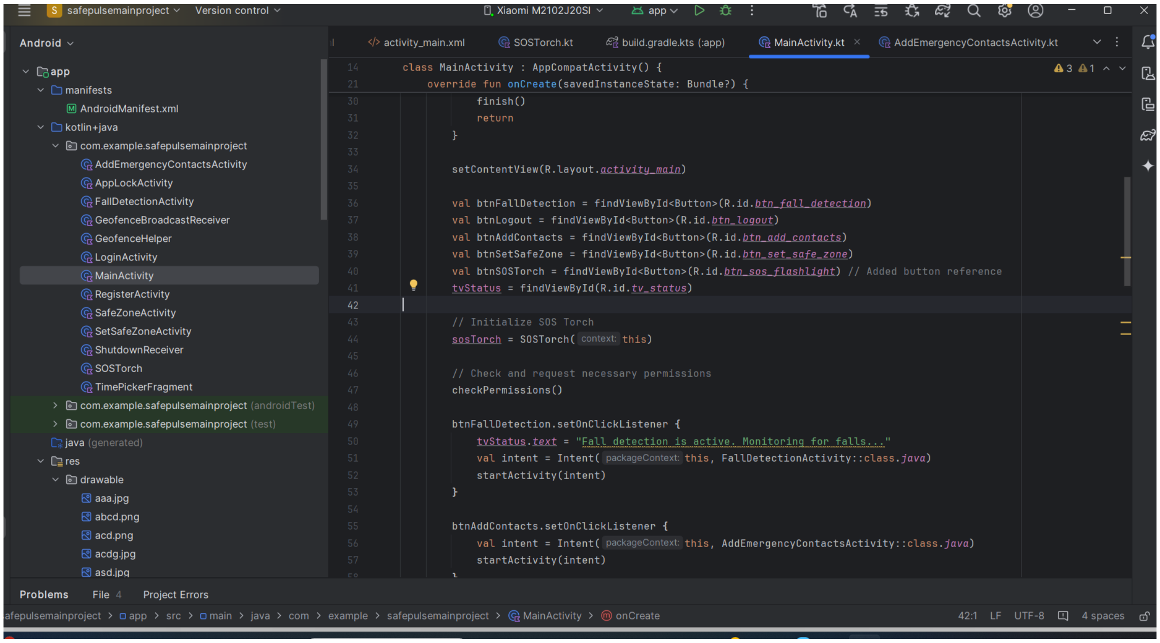

Fig. 8.1: Android Studio Interface

8.2 Backend

The backend of the SafePulse application is powered by Firebase, which offers real-time database management, cloud storage, authentication, and messaging services. Firebase enables seamless user data synchronization, location updates, emergency contact management, and real-time alert notifications. It also supports push notifications for fall detection and SOS alerts through Firebase Cloud Messaging (FCM). The integration with Firebase ensures a secure and scalable backend infrastructure that enhances the app's responsiveness and reliability in emergency scenarios.

Fig. 8.2: Backend Connectivity

8.3 Frontend

The frontend of the SafePulse app is developed using Kotlin in Android Studio, offering a user-friendly and accessible interface tailored for elderly users, specially-abled individuals, children, and users in high-risk environments. The design emphasizes simplicity, clear navigation, and large buttons for critical features such as SOS activation, emergency contact access, geofencing setup, and ride tracking. The integration of Google Maps API, CameraX, and Android Location Services ensures real-time interaction, while the Material Design components enhance usability and visual consistency across different devices.

```
<LinearLayout
        android:layout_width="48dp"
        android:layout_height="48dp"
        android:src="@drawable/hhh"/>

    <Button
        android:id="@+id/btn_fall_detection"
        android:layout_width="match_parent"
        android:layout_height="35dp"
        android:backgroundTint="#8FF00505"
        android:text="Start Fall Detection"
        android:background="@color/colorDivider"
        android:textColor="@color/background_color"
        android:textSize="13sp"
        android:fontFamily="sans-serif-medium"/>
</LinearLayout>
```

Fig. 8.3: frontend

Chapter 9

SOFTWARE TESTING

The SafePulse application underwent rigorous software testing to ensure its functionality, reliability, and user safety. Testing included unit testing, integration testing, and user acceptance testing (UAT). Each module—such as SOS alerts, geofencing, ride tracking, and unauthorized access detection—was individually tested for correctness. Real-time features like live location sharing and SMS alerts were validated in various network conditions to ensure consistency. Additionally, manual testing on multiple Android devices was performed to verify compatibility and UI responsiveness. Edge cases, including power loss, sensor misfires, and permission denials, were simulated to ensure graceful error handling and app recovery.The testing includes:

- Unit testing
- Integration testing
- User acceptance testing

9.1 Unit testing

Unit Testing in the SafePulse app was conducted to ensure the reliability and correctness of individual components and features. Each core functionality—such as fall detection, SOS flashlight activation, geo-fencing alerts, emergency SMS sending, and unauthorized access detection—was tested independently to verify its logic and behavior. Mock data and simulated inputs were used to test sensor responses, location accuracy, camera activation, and SMS delivery without relying on external dependencies.

Test ID	**Function Tested**	**Expected Output**
UT01	Emergency contact input validation	Valid contact added to database
UT02	Fall detection algorithm trigger	Fall event correctly identified
UT03	SOS flashlight activation	Flashlight blinks in SOS pattern
UT04	Geo-fencing time check logic	Alert triggered if user not in safe zone
UT05	Unauthorized access detection	Intruder photo captured and SMS sent

Table 9.1: Unit Testing Cases for SafePulse App

9.2 Integration testing

Integration Testing for the SafePulse app was performed to verify that different modules and components work seamlessly together as a unified system. After ensuring that individual units like fall detection, SOS flashlight, geo-fencing, camera capture, and SMS alert were functioning correctly through unit testing, integration testing was used to assess the interactions between these components.

Test ID	Modules Integrated	Expected Result
IT01	Contact input + Firebase database	Contact saved and retrievable
IT02	Fall detection + SMS module	SMS sent when fall detected
IT03	Geo-fence + Google Maps API	Alert triggered outside safe zone
IT04	App Lock + Camera module	Photo captured after failed attempts
IT05	SOS button + Flashlight	Flashlight blinks in SOS pattern

Table 9.2: Integration Testing Cases for SafePulse App

9.3 User acceptance testing

User Acceptance Testing (UAT) was conducted to ensure that the SafePulse app meets user expectations and performs effectively in real-world scenarios. The app was tested by a sample group including elderly individuals, specially-abled users, parents of children, and users in high-risk professions. Testers evaluated core functionalities such as fall detection, SOS flashlight signaling, geo-fencing, camera-based intruder detection, and emergency SMS alerts.

Feedback focused on ease of use, reliability, and responsiveness of the system during emergencies. Based on user input, minor UI adjustments and feature refinements were made to improve clarity and performance. UAT confirmed that the app aligns well with its goal of enhancing personal safety and emergency response.

9.4 Test Case

The SafePulse app was tested for key functionalities like app launch, emergency contact saving, fall detection, SOS flashlight, geo-fencing alerts, and unauthorized access detection. These tests ensured reliable and timely emergency responses.

ID	Description	Expected Result
TC01	App launch from device	Home screen displayed
TC02	Add emergency contact	Contact saved successfully
TC03	Fall detection alert	SMS sent to emergency contacts
TC04	SOS Flashlight trigger	Flashlight blinks SOS pattern
TC05	Geo-fence	Alert sent to emergency contacts
TC06	Unauthorized access detection	Intruder photo captured and sent

Table 9.3: Key Test Cases for SafePulse App

9.5 Test Result

9.5.1 Test Case 1

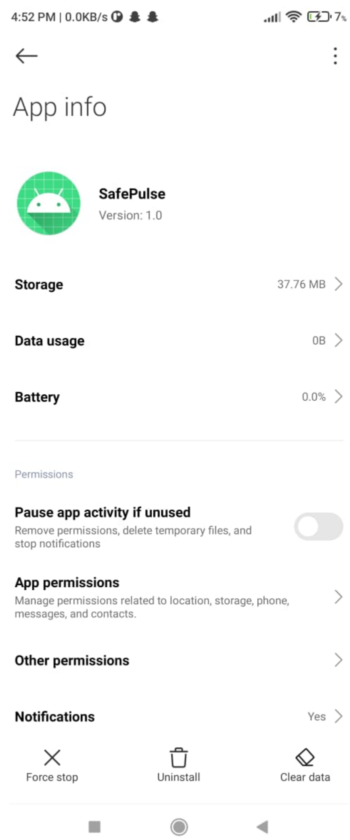

Fig. 9.1: App Info

Fig 8.1 shows the result of Test case 1. App launched from the device. So, test case 1 is successfully passed.

9.5.2 Test Case 2

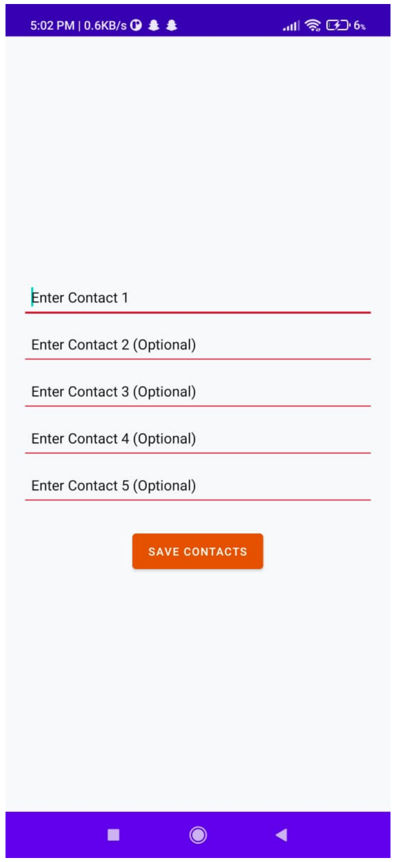

Fig. 9.2: Emergency Contact

Fig 8.2 shows the result of Test case 2. Emergency contact added. So, test case 2 is successfully passed.

9.5.3 Test Case 3

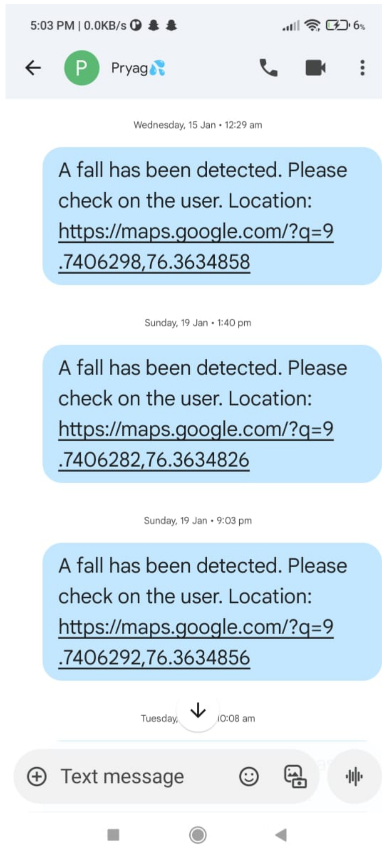

Fig. 9.3: Fall Detection Alert

Fig 8.3 shows the result of Test case 3. Fall detection alert sent successfully. So, test case 3 is successfully passed.

9.5.4 Test Case 4

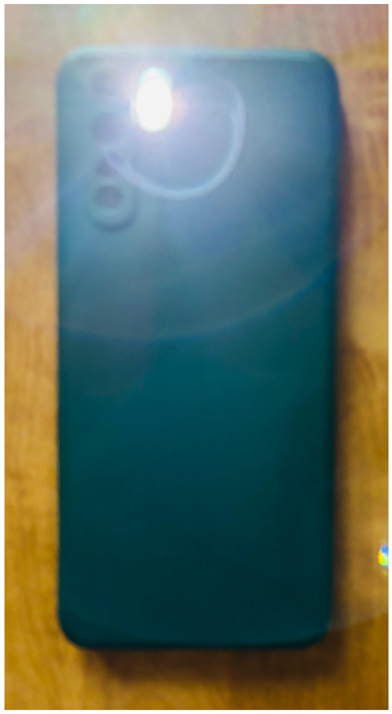

Fig. 9.4: SOS Flashlight Trigger

Fig 8.4 shows the result of Test case 4. SOS Flashlight triggered. So, test case 4 is successfully passed.

9.5.5 Test Case 5

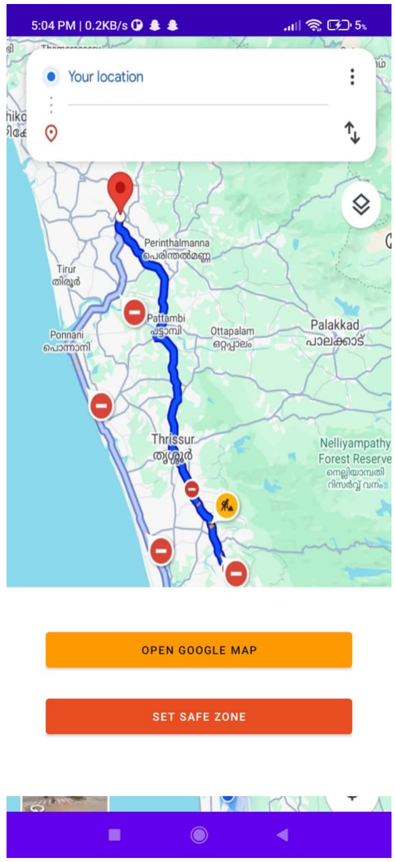

Fig. 9.5: Geofence

Fig 8.5 shows the result of Test case 5. Geofence added successfully. So, test case 5 is successfully passed.

9.5.6 Test Case 6

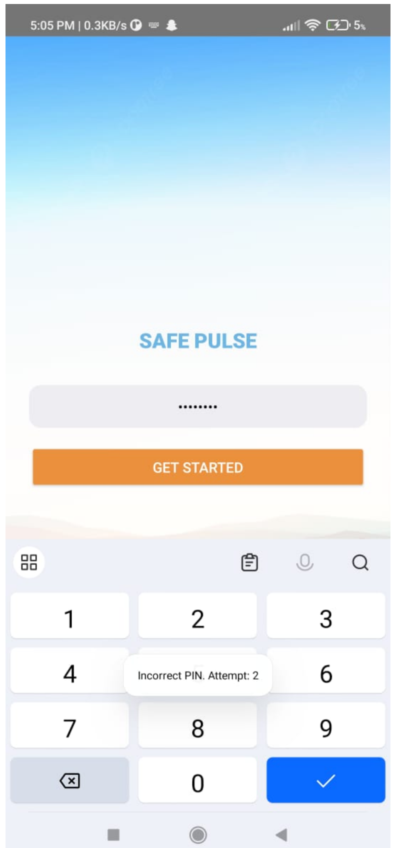

Fig. 9.6: Unauthorized Access Detection

Fig 8.6 shows the result of Test case 6. Unauthorized access detection triggered. So, test case 6 is successfully passed.

Chapter 10

RESULT AND ANALYSIS

10.1 Result

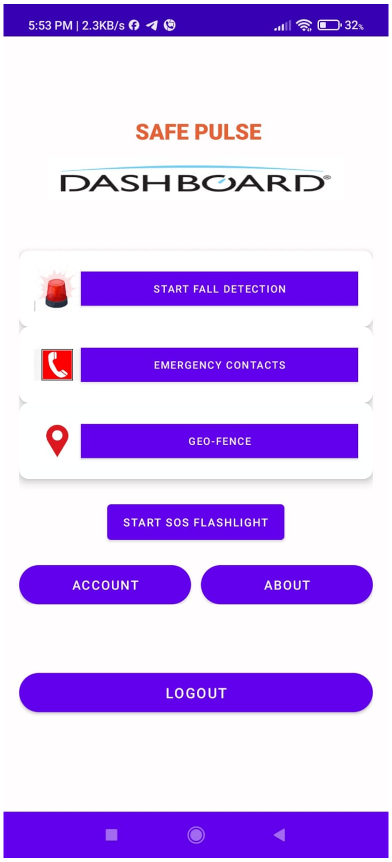

Fig. 10.1: Home Page

The Home Page of the SafePulse app serves as the central hub, offering users quick access to key safety features such as fall detection, SOS flashlight, emergency contacts, ride monitoring, and geo-fencing. With a simple and intuitive interface, users can activate or manage essential safety tools directly from the home screen, ensuring immediate response during emergencies.

Fig. 10.2: Fall Detection

Fall Detection is integrated into the SafePulse home page, allowing users to activate it easily. Once enabled, the app monitors sudden movements and triggers alerts in case of a suspected fall, ensuring quick help by notifying emergency contacts automatically.

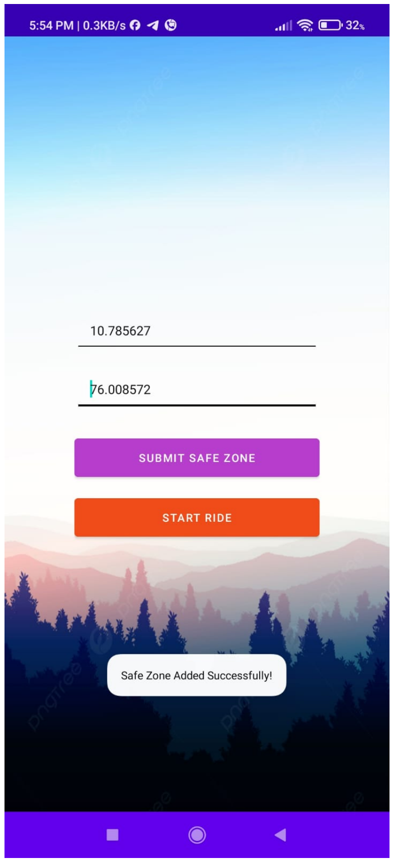

Fig. 10.3: Geofence

The Geofence feature allows users to set a safe zone using Google Maps. If the user doesn't reach this zone, the app sends an automatic alert to emergency contacts. The Start Ride button sends a message with the user's live location at the beginning of a journey, helping contacts monitor their safety in real time. Both features work together to ensure timely alerts and location tracking.

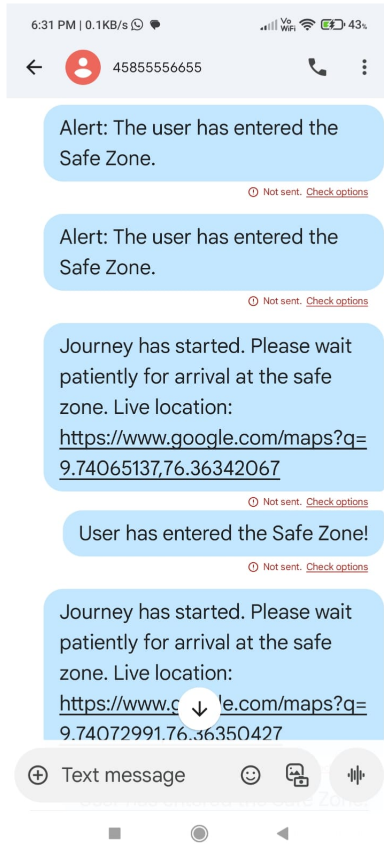

Fig. 10.4: Alert Message

Automated SMS alerts from the SafePulse app include messages for two core features: Geofence Alert (notifying that the user has entered the Safe Zone) and Start Ride (sending live location links when the journey starts). These messages help emergency contacts track the user's safety and movement in real time.

The SafePulse app is a safety-focused mobile application designed to assist specially-abled individuals, elderly users, children, and those in high-risk environments. It integrates various smart features such as fall detection, SOS flashlight, geo-fencing, emergency SMS alerts, and live location sharing to ensure user safety.

Chapter 11

SAMPLE CODES

11.1 Home page

```
setContentView(R.layout.activity_main)

val btnFallDetection = findViewById<Button>(R.id.btn_fall_detection)
val btnLogout = findViewById<Button>(R.id.btn_logout)
val btnAddContacts = findViewById<Button>(R.id.btn_add_contacts)
val btnSetSafeZone = findViewById<Button>(R.id.btn_set_safe_zone)
val btnSOSTorch = findViewById<Button>(R.id.btn_sos_flashlight) // Added button reference
```

Fig. 11.1: Mainactivity.kt

11.2 Geofence

```
private fun sendJourneyStartAlert() {
    val message = "Journey has started. Please wait patiently for arrival at the safe zone. Live location: https://www.google.com/ma
    sendSmsToEmergencyContacts(message)
}

private fun sendSafeZoneAlert() {
    val message = "User has entered the Safe Zone!"
    sendSmsToEmergencyContacts(message)
    showNotification( title: "Safe Zone Alert", message)
}
```

Fig. 11.2: setsafezoneactivity.kt.kt

11.3 Fall Detection

```
private fun getLocationAndSendSMS() {
    // Check if permission is granted
    if (ContextCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION) == PackageManager.PERMISSION_GRANTED)
        fusedLocationClient.lastLocation.addOnSuccessListener { location ->
            if (location != null) {
                val sharedPreferences = getSharedPreferences( name: "SafePulsePrefs", MODE_PRIVATE)
                val contacts = sharedPreferences.getStringSet("emergency_contacts", emptySet()) ?: emptySet()

                if (contacts.isEmpty()) {
                    Toast.makeText( context: this,  text: "No emergency contacts saved!", Toast.LENGTH_SHORT).show()
                    return@addOnSuccessListener
                }

                val message = "EMERGENCY!!! Please check on the user. Location: https://maps.google.com/?q=${location.latitude},${locat

                for (contact in contacts) {
                    sendSMS(contact, message)
                }
```

Fig. 11.3: Falldetectionactivity.kt

11.4 Unauthorized Access Detection

```
for (contact in contacts) {
    try {
        val intent = Intent(Intent.ACTION_SEND)
        intent.type = "image/jpeg"
        intent.putExtra(Intent.EXTRA_STREAM, uri)
        intent.putExtra(Intent.EXTRA_TEXT,  value: "🚨 Unauthorized access detected! See attached image.")
        intent.setPackage("com.whatsapp") // Ensure it only opens WhatsApp
        intent.addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION)

        startActivity(intent)
    } catch (e: Exception) {
        Log.e( tag: "SendWhatsApp",  msg: "Failed to send image via WhatsApp: ${e.message}")
        Toast.makeText( context: this,  text: "WhatsApp is not installed or the file format is not supported.",
    }
}
```

Fig. 11.4: Applockactivity.kt

11.5 SOS Flashlight Signal

```kotlin
for (contact in contacts) {
    try {
        val intent = Intent(Intent.ACTION_SEND)
        intent.type = "image/jpeg"
        intent.putExtra(Intent.EXTRA_STREAM, uri)
        intent.putExtra(Intent.EXTRA_TEXT, value: "🚨 Unauthorized access detected! See attached image.")
        intent.setPackage("com.whatsapp") // Ensure it only opens WhatsApp
        intent.addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION)

        startActivity(intent)
    } catch (e: Exception) {
        Log.e( tag: "SendWhatsApp", msg: "Failed to send image via WhatsApp: ${e.message}")
        Toast.makeText( context: this, text: "WhatsApp is not installed or the file format is not supported.",
    }
}
```

Fig. 11.5: sostorch.kt

Chapter 12

CONCLUSION & FUTURE SCOPE

The SafePulse app successfully addresses the growing need for personal safety solutions, especially for vulnerable groups such as children, elderly individuals, and specially-abled users. By integrating features like fall detection, SOS flashlight, unauthorized access alerts, geo-fencing, and emergency SMS with live location, the app offers a proactive and real-time safety response system. Its user-friendly interface and background functionalities ensure that users receive help when they need it most, without any manual effort.

Future enhancements to the system will focus on several key areas:

- **Voice-Activated Emergency Triggers:** Voice-activated emergency triggers allow users to initiate an SOS alert using specific voice commands without physically interacting with their phone. This feature is especially helpful for individuals who may be in distress, immobilized, or visually impaired. When a predefined keyword or phrase (e.g., "Help me" or "SafePulse SOS") is detected, the app immediately sends an alert with the user's live location to their emergency contacts, enhancing hands-free safety and response time.
- **Integration with Wearable Devices:** SafePulse integrates with wearable devices such as smartwatches and fitness bands to enhance accessibility and real-time emergency response. This allows users to trigger SOS alerts, monitor vital signs, and detect falls directly from their wearable device. The integration ensures continuous safety monitoring, even when the

phone is not easily accessible, making it particularly valuable for children, elderly individuals, and those in high-risk environments.

- **AI-Based Movement and Behavior Prediction:** SafePulse can incorporate AI algorithms to analyze users' movement patterns and predict unusual or risky behaviors, such as sudden stops, irregular walking, or deviations from routine paths. This predictive feature enhances safety by triggering proactive alerts before an emergency occurs, enabling faster response and prevention.
- **Real-time Video Streaming to Emergency Contacts:** Enable users to stream live video to their emergency contacts during distress situations for better situational understanding.
- **Safe Route Recommendation:** Use AI and crowd-sourced data to guide users through safer paths or areas based on real-time crime or incident reports.

SafePulse stands as a reliable companion for personal safety, offering proactive protection and timely assistance when it matters most.

References

[1] Giuseppe Lelow- The European Emergency Number 112 Exploring The Potential Of Crowd Sourced Information For Emergency Management DOI:10.1109/ICALTER61411.2023.10372884

[2] Medha Wyawhahare- Enhanced Security Fencing System With Geolocation Tracking, 2024. DOI:10.1109/ACCESS.2021.3113172

[3] Ruby Dahiya- Security In Mobile Network: Issues, Challenges And Solutions (Unau thorized Access Detection), 2024. DOI:10.1109/I2CT61223.2024.10543337

[4] NRajesh- Smart Device Fall Detection System For Real Time Monitoring And Emergency Re- sponse For Specially Abled Individuals, 2024. DOI: https://aclanthology.org/2021.nllp-1.11

[5] Tunkunorizan- Fall Detector Device For Improved Safety For Independence, 2024. DOI: 10.1109/IATMSI60426.2024.10502550