

Complete Documentation

Project Title: Emergency Alert And Location Sharing App

Project Code: CS-[463]

Batch: Fall-2020

Project Advisor: Almas Ilyas

Project Team:

Hamza Mughal (Group Leader) 201400164

M.Fahad Zafar 201400137

Usman Sheikh 201400129

Ch Salman Ahsan 201400184

Submission Date:

September 11, 2024

Certificate by Supervisor

This Project work has been carried out under my direct supervision. It has been checked by me and has been found to be satisfactory regarding content, format, tools, and thus fulfills the requirements for the award of BS degree.

Name of Supervisor

Signature of Supervisor

Date

Document Information

Category	Information
Customer	General Audience
Project	<EAS>
Document	Final Year Project Report
Status	Final
Author(s)	<Hamza Mughal, Ch Salman Ahsan, M Fahad Zafar, Usman Sheikh>
Approver(s)	PM
Issue Date	August 10 th , 2024
Submission Date	August 16 th , 2024
Document Location	Gift University
Distribution	1. Advisor 2. Project Office 3. Team Members

Definition of Terms, Acronyms and Abbreviations

Term	Description
SRS	Software Requirements Specification.
ML	Machine Learning
GUI	Graphical User Interface.
DS	Design Specification
DD	Design Document
WBS	Work Breakdown Structure
TC	Test Cases
TP	Test Plan
TR	Test Report
FR	Functional Requirements
NFR	Non Functional Requirements

Table of Contents

Certificate by Supervisor.....	2
Document Information	3
Definition of Terms, Acronyms and Abbreviations	3
Software Requirement Specifications.....	7
1. Introduction.....	8
1.1 Purpose.....	8
1.2 Scope.....	9
1.3 Technology	9
1.4 Overall Description.....	10
1.5 Practical Applications and Comparative Analysis.....	12
1.6 Learning Outcomes	12
1.7 Work Breakdown Structure:.....	13
1.8 Gant Chart.....	14
2. Specific Requirements	15
2.1 Functionality	15
2.2 Usability.....	16
2.3 Reliability.....	16
2.4 Performance	17
2.5 Supportability and Compatibility.....	17
2.6 Design Constraints	17
2.7 On-line User Documentation and Help System Requirements.....	17
2.8 Purchased Components	18
2.9 Interfaces.....	18
2.10 Licensing Requirements.....	18
2.11 Legal, Copyright, and Other Notices	18
2.12 Applicable Standards	18
3. Supporting Information.....	19
3.1 Activity Detection.....	19
3.2 Geo location Technologies.....	20
4. Use Cases.....	21

4.1	Use Case Diagram.....	21
4.2	Use Case Description.....	22
	Software Design Specifications	31
5.	Introduction of Design Specifications.....	32
5.1	Purpose of Design Specifications	32
5.2	Design Objectives	32
6.	Design Considerations	32
6.1	Assumptions and Dependencies.....	32
6.2	Risks and Volatile Areas.....	33
7.	System Architecture.....	34
8.	System Detailed Design.....	35
8.1	Class diagram.....	35
8.2	Comprehensive Sequence diagram	36
8.3	Detailed Sequence diagrams with parameter list	37
8.4	Activity Diagram.....	42
8.5	Data Flow Diagram.....	43
8.6	ERD.....	44
9.	System Mockups/UI.....	45
10.	References.....	58
11.	Appendices.....	58
	Project Testing.....	59
12.	Introduction.....	60
12.1	Purpose.....	60
12.2	Scope of Testing.....	60
13.	Test Plan.....	61
13.1	Test Plan Strategy	61
13.2	Test Environment.....	64
13.3	Schedule	65
13.4	Control Activities.....	65
13.5	Resources	66
13.6	Major Deliverables.....	66
13.7	Risk and Assumptions.....	67

14. Test Report.....	68
14.1 Test Case Scenarios and Test Cases.....	68
14.2 Traceability Matrix	90
15. Exit Criteria.....	91
16. Sign off.....	92

Software Requirement Specifications

1. Introduction

This project Document outlines the goals and specifications for creating the location-sharing and emergency alert application. This application uses mobile technology and integrates with office or home equipment like cameras and alarms to satisfy the vital need for quick response during emergencies. The system backdrop highlights the need for a smooth communication channel with emergency services and designated contacts and describes the gap in current emergency communication techniques. The project's goals are to facilitate the integration of IOT devices, allow users to share their location in real-time, enable camera-based detection of suspicious activity, integrate real-time communication with emergency services, and enable users to initiate predefined emergency alerts. The document also describes requirements for reaction time, usability, security, and compatibility with iOS and Android devices utilizing React Native for front-end development. Reliability standards are also outlined for both camera activity detection and connection with emergency services. The overall goal of this project is to create a complete and intuitive solution that will enable users to communicate quickly and efficiently in emergency situations.

1.1 Purpose

In contemporary society, the need for rapid response during emergencies is paramount. Traditional communication methods may not always suffice in critical situations where every second counts. There is not any application that addresses this gap by leveraging mobile technology to provide a quick and efficient means for users to summon emergency services and their relatives or people which can take any rapid action for them, in the all that cases our manual contacting system of emergency services or relative fails that's why we are making a such system.

The Emergency Alert and Location Sharing Application is a mobile-based solution designed to empower users through Suspicious activity detection through ml model and in swiftly communicating with Contacts police, fire brigade, and ambulance services during emergencies as well as user can intergrade the home or office devices like alarms, cameras.

The Emergency Alert and Location Sharing Application stands as a critical solution in contemporary society, aiming to revolutionize emergency communication by leveraging mobile technology. It marks a new era of instant emergency communication, precise location sharing, and accessibility for diverse needs

1.2 Scope

The scope of this project encompasses the development of a cross-platform mobile application for Android. The application will feature a user-friendly interface, a quick response algorithm for automated messages, real-time location sharing using GPS technology, and secure communication channels. The focus will be on collaboration with local emergency services for seamless integration. This Application is for only Pakistan. The user sent messages predefine/Default i.e. emergency alerts Messages to contacts can be edited by the user if he or she has time to describe an emergency situation otherwise after click, it will be sent automatically.

The camera's suspicious activity detection feature aims to provide users with a proactive security solution, enabling them to respond swiftly to potential threats and contributing to the overall safety and well-being of the user and their property in their offices or home and the user can trigger the alert that can be connected with WIFI in the case activity.

Suspicious Activity Category Includes:

- a) Masked Face
- b) Wall Climbing
- c) Weapon

Exclusions:

- a) The system cannot be integrated with other devices except cameras or alarms.
- b) The system does not control or guarantee the response times or actions of external emergency services. It relies on the existing capabilities and responsiveness of these services.

1.3 Technology

Tools And Technology	Purpose
React Native	Front-end Development
Geolocation technology	For send live location
Python	For security camera
Firebase	Back-end development
Android studio	Tool for Development
Adobe XD 2020	For app interface design
IOT devices	Forget live footage, enable alarm
Machine learning Model	Activity Detection

1.4 Overall Description

Background and Problem Statement

In the Today ‘Society, people in need of immediate help usually had to telephone hotline services, which could cause delays and result in the loss of vital time in urgent situations. The location and specifics of the emergency were frequently communicated over the phone, which led to errors and inefficiencies that delayed the arrival of emergency personnel on time. Another issue was the widespread deployment of CCTV cameras in homes and workplaces, which frequently resulted in a lack of ongoing oversight of the recorded material. Without real-time activity monitoring, incidents can happen without prompt notice or action, exposing people and property at risk from a variety of dangers.

Proposed Solution and Objectives

In order to overcome these obstacles (discuss in problem statement), we suggest creating an application with a Suspicious Activity detection through CCTV live footage with Machine Learning model built in, giving consumers instant access to all necessary emergency services, contacts with Call and default message, along with live location and also allows user integration with Alarm for trigger at the time of critical situation. This streamlined procedure seeks to improve overall efficiency, security, and speed while drastically reducing reaction times, guaranteeing prompt aid during emergency situations.

Our app also integrates with security cameras, improving surveillance capabilities and allowing for the early identification of questionable activity. User will be able to react quickly and minimize losses when abnormalities, such Violence or theft, are detected thanks to this connection, which will provide users with instant notifications. Our program aims to alleviate the difficulties linked to manual monitoring and response by using technology to automate these procedures and deliver real-time notifications, thereby improving safety and security for both people and property owners.

Objectives:

- a) Use integrated security cameras to detect suspicious activity with 70% accuracy through machine learning Model.
- b) Provide thorough online documentation and user assistance resources directly within the program.
- c) Make a quick and efficient Communications system for user by using advance communication protocols.

The Emergency Alert and Location Sharing Application's main objective is to meet the demand for emergency response. This program uses technology to establish a line of communication between users and their designated contacts and emergency services. In life-threatening situations, the current system sometimes fails to provide assistance; hence a practical remedy is

required. The goal of this project is to address issues with inadequate security measures, poor response times, and communication channels. The app gap is caused by the lack of one that integrates proactive surveillance utilizing home or office devices, real-time location sharing, and emergency notifications of Suspicious Activity detection.

The goals include giving consumers an intuitive interface to facilitate speedy replies and including capabilities that guarantee communication with emergency services.

The Emergency Alert Application aims to give a best service to handle an emergency situation through its ml trained model for Suspicious activity detection in which the CCTV footage will integrate with our server and then application's algorithm will detect activity such as wall climbing, covered face, weapons on runtime if anyone can try to make violence and can record in footage and as well as system generate alert to user to see whether the situation is critical or not and the user can trigger IOT base alarm which is also connected to system, and can communicate the other contacts or Emergency Services like, police, fire brigade, ambulance. Designing a simple and intuitive interface that can be easily understood and operated by users of varying ages, abilities, and technical expertise, even under stressful conditions. Address potential privacy concerns regarding location tracking and intrusion detection by providing clear information about data collection, usage, and sharing practices, and offering user control over settings.

User have to use and set IOT devices cameras, alarm that is necessary for application, user can also send alert message to WhatsApp and in Apps own communication system, User can add a user as contact and can give alert message to that person for handle situation and also can share real time location. This constraint enables reliable and efficient communication between users and emergency conditions. Using Firebase for backend. Personal information like contact number and locations data collected by the application shall be securely stored. This constraint safe-guard's user privacy, it builds confidence in the application's usage.

The system should support integration with some IOT devices like home security devices, including cameras and alarms, based on standard protocols thorough Wi-Fi. Suspicious activity detection through ml model like wall climbing, covered masked, weapon detection through CCTV live footage. Integrate real-time location with communication System and enable connected cameras to detect suspicious activities and generate alerts. Include geographical coordinates in location information for emergency alerts in the form of calls to emergencies services. Allow users to define and manage a list of predefined contacts. Enable users to alert contacts through messaging as well as Hope to achieve 50 to 70 percent reliability of camera real time activity detection. Maintain an average response time of 3 or less seconds for emergency alerts and location sharing. Easy Design that can be useable for all and giving awareness and training to users about application, encrypt all communication between the application and external services using industry standard protocols. Securely store and handle personal information and location data in compliance with data protection regulations. Provide comprehensive online documentation and help resources within the application for user support as well as React native is using for Front End development.

1.5 Practical Applications and Comparative Analysis

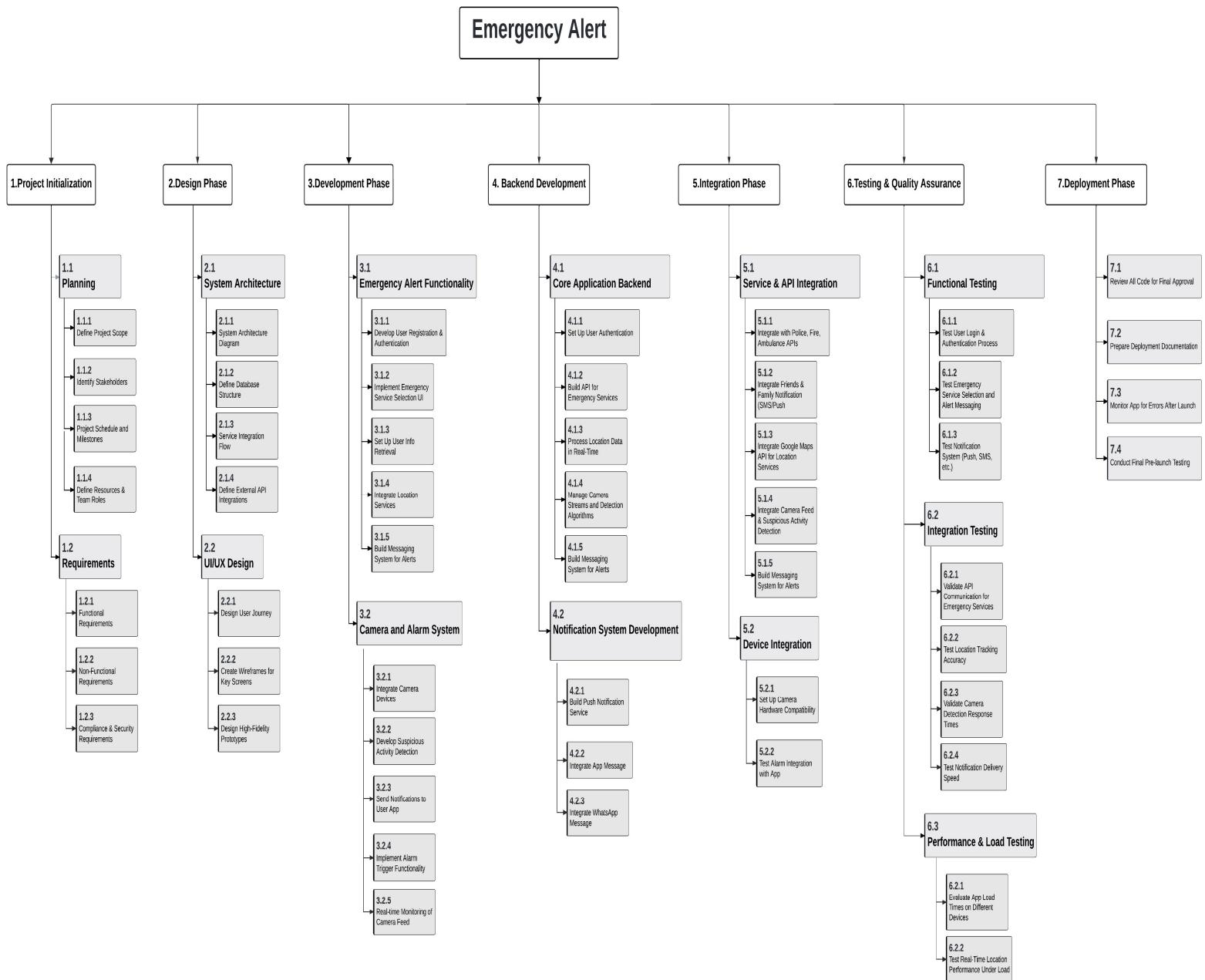
Application Name	Weakness	Proposed Project Solution
Emergency Plus	Bad user experience, No machine learning model ,No cross platform support and does not provide alert services	Our app will have Good UI/UX and provide alert services and machine learning model for activity detection through CCTV.
SOS App	This app does not gives alerts and do not support GPS support	Our app will have functionality to send alerts
Safe and security App	There's no support of IOT devices and GPS technology	Our app will have integration of different IOT devices and GPS support to fetch location

1.6 Learning Outcomes

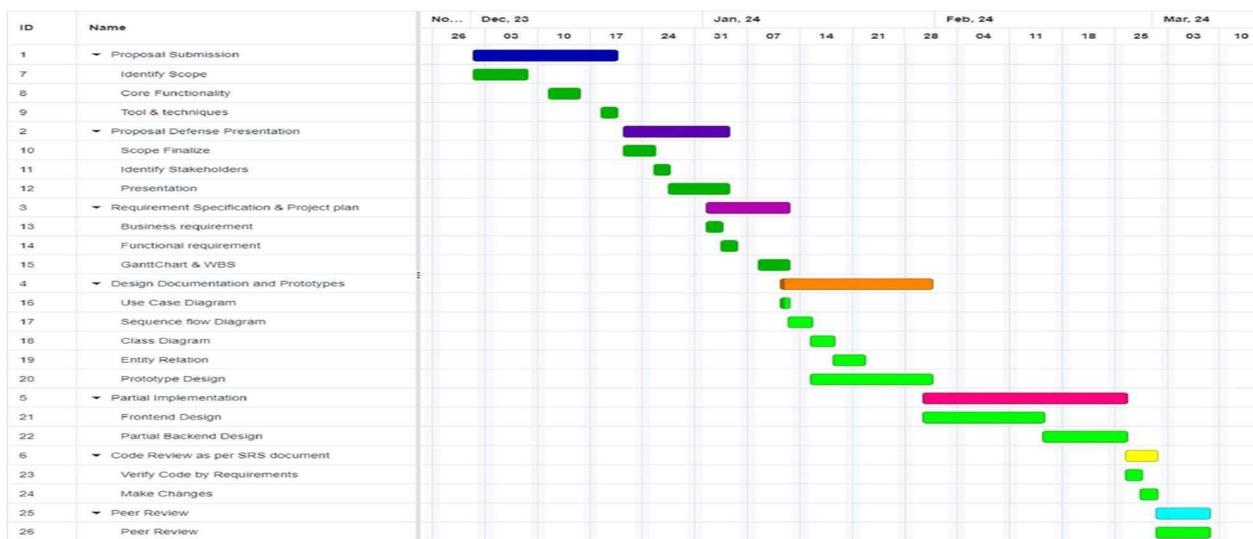
The proposed Emergency Alert and Location Sharing Application project encompasses a range of technical and practical aspects, providing a rich learning experience for those involved. The following learning outcomes are expected:

- 1) **Mobile Application Development:** how to develop a mobile app in react native.
- 2) **Geolocation Technologies and communication protocols:** Use of location coordinates and send it on run time.
- 3) **User Education and Awareness Campaigns:** Designing effective user education strategies, creating informative materials, and conducting awareness campaigns.
- 4) **IOT Devices Connectivity:** Learn how to use and connect/interact/operate devices through application.
- 5) **Suspicious Activity Detection:** Familiarity with Machine Learning Algorithms.

1.7 Work Breakdown Structure:



1.8 Gant Chart



2. Specific Requirements

2.1 Functionality

FR-01	System shall allow user to Sign Up.
FR-02	System should allow user to Sign in and out.
FR-03	System shall allow integration with IOT devices, such as home security cameras and alarms,
FR-04	System shall perform real-time analysis of camera streams, detecting suspicious activities
FR-05	System shall provide real-time communication with user added contacts to facilitate
FR-06	System shall also integrate with WhatsApp for send default massage along with live
FR-07	System shall allow user to customize predefined emergency messages before sending,
FR-08	System shall enable users to initiate emergency calls to police, fire brigade, and ambulance
FR-09	System shall allow users to share their real-time location during emergencies, including
FR-10	System shall allow the user to save and delete predefined contacts for quick alerting during
FR-11	System shall allow user to add/remove Alarms and cameras views.
FR-12	All communication shall be encrypted using industry-standard protocols to ensure security.
FR-13	System shall provide comprehensive online documentation and help resources shall be
FR-14	System shall allow user to pin any view on front.
FR-15	System shall allow user to Delete Account permanently.

2.2 Usability

UR-01	User can register itself.
UR-02	User can Sign in and out.
UR-03	User can integrate app with IOT devices, such as home security cameras and alarms, using standard protocols through Wi-Fi.
UR-04	User can receive alert when ml model detect activity (Wall climb, Gun, Covered Face).
UR-05	User can make real-time communication with other user, added to facilitate efficient response on system's own SOS alert module.
UR-06	User can also send default massage along with live location in the critical situation with a single click on WhatsApp.
UR-07	User can customize predefined emergency messages before sending, ensuring accurate and detailed information is conveyed on WhatsApp.
UR-08	User can initiate emergency calls to police, fire brigade, and ambulance services with a single click.
UR-09	User can pin camera view on front Screen.
UR-10	System shall allow user to add/remove Alarms and cameras views.
UR-11	User can read comprehensive online documentation and help resources shall be provided within the application for user support.
UR-12	User can Delete Account permanently.

2.3 Reliability

- o The system shall have a 99.9% uptime, ensuring continuous availability for users in emergency situations.
- o In the event of a system failure, the application shall automatically attempt to reconnect to emergency services
- o The integration with emergency services shall demonstrate a reliability high level, ensuring a robust connection during critical situations. o The camera module shall aim for a reliability level of 50 to 70 percent in real-time activity detection.

2.4 Performance

- o The system shall handle multiple simultaneous users without a significant degradation in performance.
- o The application shall load within 3 seconds on Android devices.
- o The System should have fast communication.
- o The system shall maintain an average response time of 3 seconds or less for both emergency alerts and location sharing.

2.5 Supportability and Compatibility

- o The system architecture shall be scalable to accommodate potential increases in the user base and additional IOT devices.
- o The system shall support over-the-air updates for seamless software upgrades without user intervention.
- o The application shall be compatible with Android (version 8 and above) platforms

2.6 Design Constraints

- o The application shall be compatible with mobile Android platforms.
- o The system shall exclude integration with devices other than cameras or alarms, maintaining focus on home and office security.
- o The design of the application shall be user-friendly and easily usable for individuals of all ages and technical expertise.

2.7 On-line User Documentation and Help System Requirements

- o The system shall provide comprehensive online documentation and help resources within the application for user support.
- o The application shall include interactive training modules to educate users on the proper use of emergency features.

2.8 Purchased Components

- o The system may utilize third-party components for specific functionalities, and their reliability and compatibility shall be thoroughly tested such as cameras, Alarms.

2.9 Interfaces

- a) User Interfaces: The application shall have a consistent and visually appealing user interface across Android platforms.
- b) Hardware Interface: The system shall seamlessly interface with various IOT devices, including cameras and alarms, through standard protocols over Wi-Fi.
- c) Software Interfaces: The application shall integrate with messaging platforms like WhatsApp for quick alerting of predefined contacts.
- d) Communications Interfaces: The system shall encrypt all communication between the application and external services using industry-standard protocols.

2.10 Licensing Requirements

- o The application may operate under a freemium model, offering basic features for free and premium features with a paid subscription.

2.11 Legal, Copyright, and Other Notices

- o The system shall comply with all relevant local and international laws, including data protection regulations and privacy laws.

2.12 Applicable Standards

- o The development and operation of the Emergency Alert and Location Sharing Application shall adhere to industry standards, including IEEE and relevant security standards.

3. Supporting Information

3.1 Activity Detection

Human activity detection using cameras involves the process of identifying specific actions or behaviors performed by individuals within the field of view of a camera. This technology leverages various computer vision and artificial intelligence techniques to analyze video frames and detect activities such as walking, running, falling, or more complex interactions like fighting, using an object, etc. Here's an overview of the key aspects of camera-based human activity detection:

1. Video Capture and Preprocessing

The first step involves capturing video footage using one or more cameras. This footage is then preprocessed, which may include steps like resizing, normalization, and sometimes converting to grayscale to reduce computational complexity, although color can be a crucial feature for certain activities.

2. Human Detection and Tracking

Before analyzing activities, the system needs to detect and possibly track individuals across frames. This can be achieved using methods like background subtraction, or more sophisticated deep learning models like Convolutional Neural Networks (CNNs) or Region-Based CNNs (R-CNNs) designed for object detection and tracking.

3. Feature Extraction

This process involves identifying and extracting relevant features from the video data that can help in distinguishing different types of activities. Features could be based on the shape, posture, and motion of humans. Optical flow, which captures the motion between video frames, is a commonly used feature for activity recognition.

4. Activity Recognition Models

With features extracted, the next step is to classify the activity. This is typically done using machine learning models or deep learning architectures. Popular choices include:

Support Vector Machines (SVMs): Useful for simpler activity recognition tasks with well-defined feature sets.

Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks: Effective for sequential data like video, capturing temporal dynamics.

3D CNNs: Capable of capturing spatial and temporal features directly from video frames.

3.2 Geo location Technologies

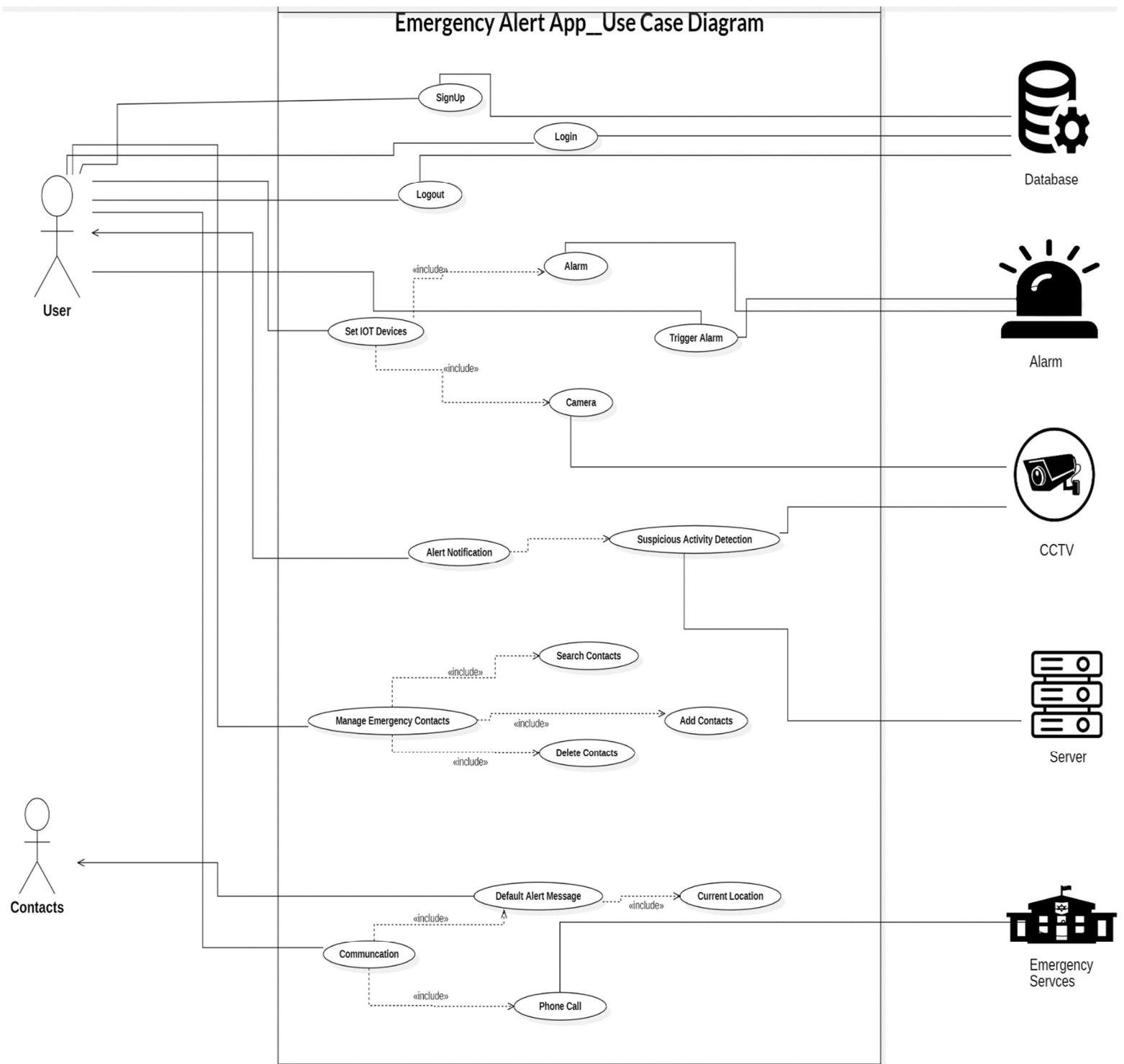
GPS is a satellite-based system that provides precise location data. It works by using signals from a network of satellites orbiting the Earth to determine the location of a GPS enabled device.

Navigation systems, mapping applications, location-based services, emergency response. Typically, accurate within a range of 5 to 10 meters, depending on conditions.

This method determines a device's location based on its IP address. It is less precise but useful for determining general locations, such as the city or region where a device is located.

4. Use Cases

4.1 Use Case Diagram



4.2 Use Case Description

ID: UC-01		
Actors	<i>User</i>	
Feature	<i>User Registering</i>	
Use case	<i>Sign-Up</i>	
Pre-condition	User started the App	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User taps on the "Sign Up" button.</i>	
		<i>Sign Up Screen Opens</i>
1.2	<i>User Adds Name</i>	
		<i>Name Entered Successfully</i>
1.3	<i>User Adds Email</i>	
		<i>Email Adds Successfully</i>
1.4	<i>User Adds Phone</i>	
		<i>Phone No. Entered Successfully</i>
1.5	<i>User Adds Address</i>	
		<i>Address Entered Successfully</i>
1.6	<i>User Save Form</i>	
		<i>Saved Successfully</i>
1.7	<i>User Verify Account</i>	
		<i>Verify Successfully</i>
1.8	<i>User Enter Password</i>	
		<i>Password Set Successfully</i>
Post Condition	User registered into app database	
Alternative	User can Login if he have already Signup	
Use Case Cross referenced	<i>UC-02 (Login)</i>	
User Interface reference	<i>Sign-Up Screen</i>	

ID: UC-02		
Actors	<i>User</i>	
Feature	<i>User Login to App</i>	
Use case	<i>Login</i>	
Pre-condition	<i>User Already Sign Up</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Enters Email</i>	
		<i>Email Enters Successfully</i>
1.2	<i>User Enters Password</i>	
		<i>Password Entered Successfully</i>
1.3	<i>User Click Login</i>	
		<i>Login Successfully</i>
Post Condition	<i>User Login into app database</i>	
Alternative	<ul style="list-style-type: none"> o If the user forgets their password: o The user clicks on the "Forgot Password" link. o The system prompts the user to enter their registered email address. o The user enters their email address and submits the request. o The system sends a password reset link to the user's email address. o The user follows the link to reset their password and regain access to the system. 	
User Interface reference	<i>Sign-In Screen</i>	

ID: UC-03		
Actors	<i>User</i>	
Feature	<i>User Logout to App</i>	
Use case	<i>Log-Out</i>	
Pre-condition	<i>User Already Sign In</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Open Settings</i>	
		<i>Open Successfully</i>
1.2	<i>User Click Logout</i>	
		<i>Logout Successfully</i>
Post Condition	<i>Logout from app</i>	
User Interface reference	<i>Settings Screen</i>	

ID: UC-04		
Actors	<i>User</i>	
Feature	<i>User Sets IOT Devices</i>	
Use case	<i>Set IOT Devices</i>	
Pre-condition	<i>User Already Sign In</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Click Config Devices</i>	
		<i>Open IOT Configuration Settings Screen</i>
1.2	<i>User Enters Details</i>	
		<i>Entered Successfully</i>
1.3	<i>User Click Saved</i>	
		<i>Saved Successfully</i>
Post Condition	<i>User Connected IOT Devices.</i>	
Alternative	<i>If not sets, so checks IOT connectivity</i>	
User Interface reference	<i>Configuration Screen</i>	

ID: UC-05		
Actors	<i>User</i>	
Feature	<i>User Sets Cameras CCTV</i>	
Use case	<i>Camera</i>	
Pre-condition	<i>User Already Sign In</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Click Set Camera</i>	
		<i>Open Camera Configuration Settings Screen</i>
1.2	<i>User Fill Details</i>	
		<i>Entered Successfully</i>
1.3	<i>User Click Saved</i>	
		<i>Saved Successfully</i>
Post Condition	<i>User Connected Camera Devices.</i>	
Alternative	<i>If not sets, so checks IOT connectivity</i>	
User Interface reference	<i>Configuration Screen</i>	

ID: UC-06		
Actors	<i>User</i>	
Feature	<i>User Sets Alarm</i>	
Use case	<i>Alarm</i>	
Pre-condition	User Already Sign In	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Click Set Alarm</i>	
		<i>Open Alarm Configuration Settings Screen</i>
1.2	<i>Enters Name</i>	
		<i>Enters Successfully</i>
1.3	<i>User Enters IP No.</i>	
		<i>Entered Successfully</i>
1.4	<i>User Click Saved</i>	
		<i>Saved Successfully</i>
Post Condition	User Connected Camera Devices.	
Alternative	If not sets, so checks IOT connectivity	
User Interface reference	<i>Configuration Screen</i>	

ID: UC-07		
Actors	<i>User</i>	
Feature	<i>Suspicious Activity Detection with ML model</i>	
Use case	<i>Suspicious Activity Detect</i>	
Pre-condition	User Already Sign In	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Sets Camera</i>	
		<i>Configuration</i>
1.2	<i>Server Detects Wall Climb</i>	
		<i>Detect Successfully</i>
1.3	<i>Server Detects Masked Face</i>	
		<i>Detect Successfully</i>
1.4	<i>Server Detects Gun</i>	
		<i>Detect Successfully</i>
Post Condition	System Detect Successfully.	
Alternative	If not sets, so checks server	
User Interface reference	<i>Camera Streams Screen</i>	

ID: UC-08		
Actors	<i>User</i>	
Feature	<i>Alert Notifications on users screens</i>	
Use case	<i>Alerts</i>	
Pre-condition	<i>Models detects Activity</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>System Detects Suspicious Activity</i>	
		<i>Alerts Notifications to user</i>
Post Condition	<i>User Received Notifications.</i>	
Alternative	<i>Emergency Alert Not Confirmed</i>	
User Interface reference		<i>Notification Screens Screen</i>

ID: UC-09		
Actors	<i>User</i>	
Feature	<i>Manage Emergency Contacts (User can delete , add and search others)</i>	
Use case	<i>Manage Emergency Alert</i>	
Pre-condition	<i>User Already Sign In</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Sets Contacts</i>	
		<i>Saved/Update in database</i>
User Interface reference		<i>Add User Screen</i>

ID: UC-10		
Actors	<i>User</i>	
Feature	<i>Manage Contacts</i>	
Use case	<i>Search Other users</i>	
Pre-condition	<i>User Cannot Saved Contacts</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Opens Add User Screen</i>	
		<i>Opened</i>
1.2	<i>User Enter Others Users Email</i>	
		<i>Entered Successfully and system searched</i>
Post Condition	<i>User Searched Emergency Contact.</i>	
Alternative	<i>If not sets, so checks user is registered or not.</i>	
User Interface reference	<i>Add User Screen</i>	

ID: UC-11		
Actors	<i>User</i>	
Feature	<i>Manage Contacts</i>	
Use case	<i>Add Other users</i>	
Pre-condition	<i>User Searched Contact.</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Clicked Add When Searched</i>	
		<i>Saved</i>
Post Condition	<i>User Add Emergency Contact.</i>	
Alternative	<i>If not sets, so checks user is registered or not.</i>	
User Interface reference	<i>Add User Screen</i>	

ID: UC-12		
Actors	<i>User</i>	
Feature	<i>Manage Contacts</i>	
Use case	<i>Delete Other User</i>	
Pre-condition	User Saved Contacts	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Opens Chats Screen</i>	
		<i>Opened</i>
1.2	<i>User Clicks Delete</i>	
		<i>Contact Deleted</i>
Post Condition	User Deleted Emergency Contact.	
Alternative	If not sets, so checks app connectivity.	
User Interface reference		<i>Chat Screen</i>

ID: UC-13		
Actors	<i>User</i>	
Feature	<i>User Communication with others.</i>	
Use case	<i>Communication</i>	
Pre-condition	User Saved Contacts	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Click SOS Button</i>	
		<i>Alert Message sends to all</i>
1.2	<i>Other User sends Message</i>	
		<i>Receive Message</i>
Post Condition	User Communicate With other Emergency Contact.	
Alternative	If not sets, so checks user is registered or not.	
User Interface reference		<i>Dashboard Screen</i>

ID: UC-14		
Actors	<i>User</i>	
Feature	<i>User Starts Alarm at the time of Emergency</i>	
Use case	<i>Trigger Alarm</i>	
Pre-condition	<i>User Already Sign In</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Click Alarm</i>	<i>Alarm Triggering</i>
Post Condition	<i>User Generated Alarm.</i>	
Alternative	<i>If not sets, so checks IOT connectivity</i>	
User Interface reference		<i>Dashboards Screen</i>

ID: UC-15		
Actors	<i>User</i>	
Feature	<i>User Communication with others.</i>	
Use case	<i>Default Message</i>	
Pre-condition	<i>User Saved Contacts</i>	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Click SOS Button</i>	<i>Alert Message sends to all</i>
Post Condition	<i>User Shares Message with other Emergency Contact.</i>	
Alternative	<i>If not sets, so checks user is registered or not.</i>	
User Interface reference		<i>Dashboard Screen</i>

ID: UC-16		
Actors	<i>User</i>	
Feature	<i>User Shares Real Time location In Communication with others.</i>	
Use case	<i>Current Location</i>	
Pre-condition	User Saved Contacts	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Allows GPS Locations to App</i>	
		<i>Get Location</i>
1.2	<i>User Click SOS Button</i>	
		<i>Alert Message sends to all</i>
Post Condition	User Shares Message along with location with other Emergency Contact.	
Alternative	If not sets, so checks user is registered or not.	
User Interface reference		<i>Dashboard Screen</i>

ID: UC-17		
Actors	<i>User</i>	
Feature	<i>User Phone Contact With Emergency Services .</i>	
Use case	<i>Phone call</i>	
Pre-condition	User Saved Contacts	
Scenarios		
Step #	Action	Software Reaction
1.1	<i>User Clicks Phone Buttons</i>	
		<i>Opened</i>
1.2	<i>User Click Police button</i>	
		<i>Dialer screen opened for initiate call</i>
1.3	<i>User Click Fire-Brigade button</i>	
		<i>Dialer screen opened for initiate call</i>
1.4	<i>User Click Ambulance button</i>	
		<i>Dialer screen opened for initiate call</i>
Post Condition	User Initiate Call With Emergency Services.	
Alternative	If not sets, so checks balance.	
User Interface reference		<i>Phone Screen</i>

Software Design Specifications

5. Introduction of Design Specifications

5.1 Purpose of Design Specifications

The Design of Emergency Alert Applications main objective is to define a detailed application with diagrams, in which there is full Functionality, Flow, Activities, shown and also UI Images to show our application real interface.

5.2 Design Objectives

- o User Friendly Interface
- o Clarity and Usability

6. Design Considerations

The design phase of the Emergency Alert Application involves careful consideration of various factors that influence the overall architecture and functionality of the system. This section outlines the crucial aspects that must be addressed or resolved prior to finalizing a comprehensive design solution. These considerations form the foundation for developing a robust, flexible, and efficient system.

6.1 Assumptions and Dependencies

- a) Technological Dependencies: The system's design may rely on specific technologies, platforms, or frameworks. Any changes or updates in these dependencies could impact the design and necessitate adjustments.
- b) External Services: If the application depends on external APIs or services (e.g., location services, notification systems, IOT Devices), the availability and functionality of these services are crucial. Changes to these services could require design modifications.
- c) User Base Assumptions: The design might be predicated on assumptions about the user base size, behavior, or device usage patterns. Variations in these assumptions could affect scalability and performance considerations

6.2 Risks and Volatile Areas

System Response Time:

If our system will not very fast or speedy, and user cannot handle the Emergency situation, then it might be possible it fails.

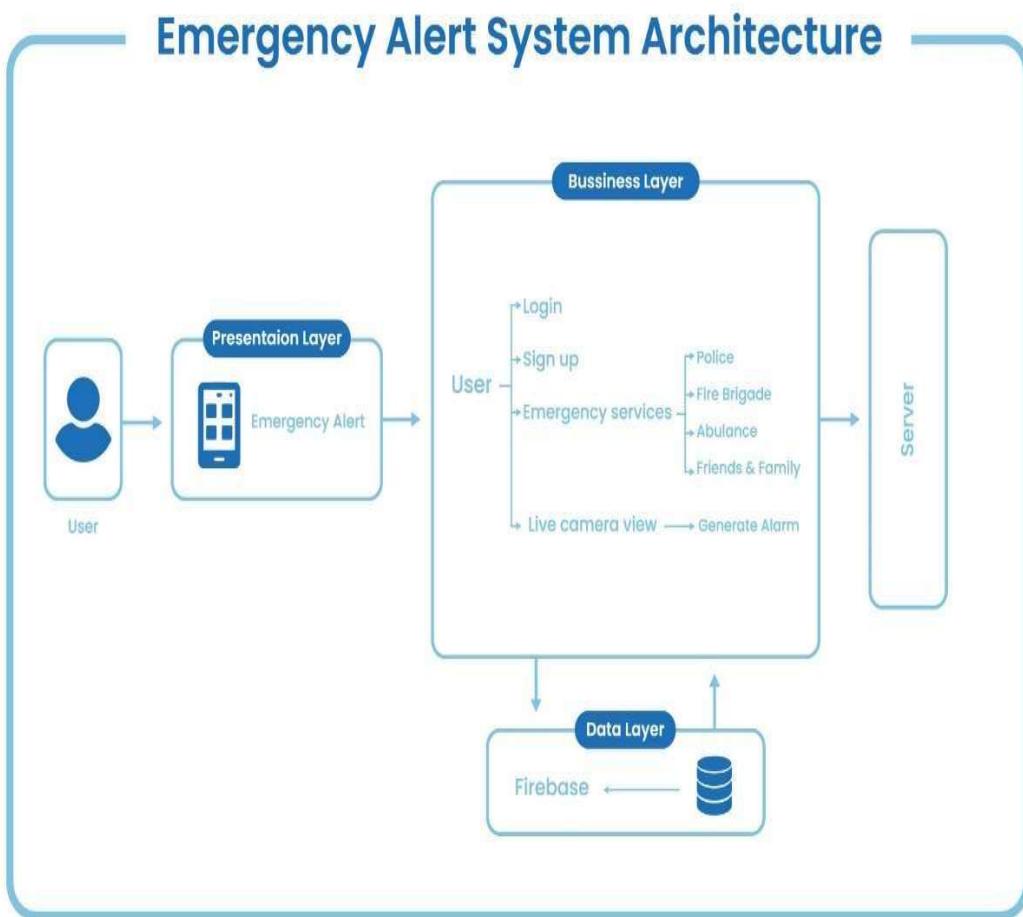
External Emergency Services Relief:

If the external emergency services could not work on the computer generated call, then user can panic.

Camera Activity Detection:

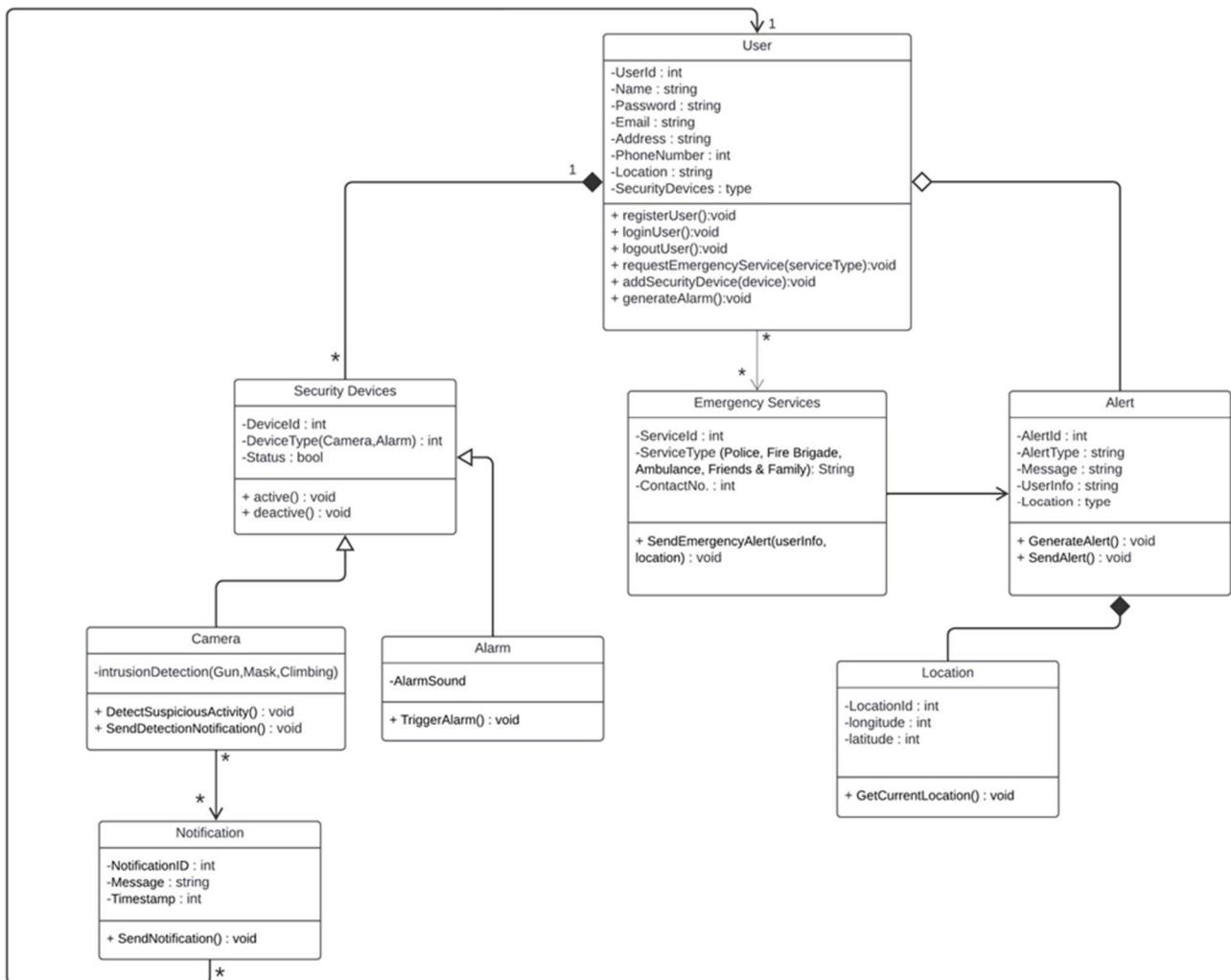
If our camera module should not work properly, so our system can fail to resolve the users' situation.

7. System Architecture

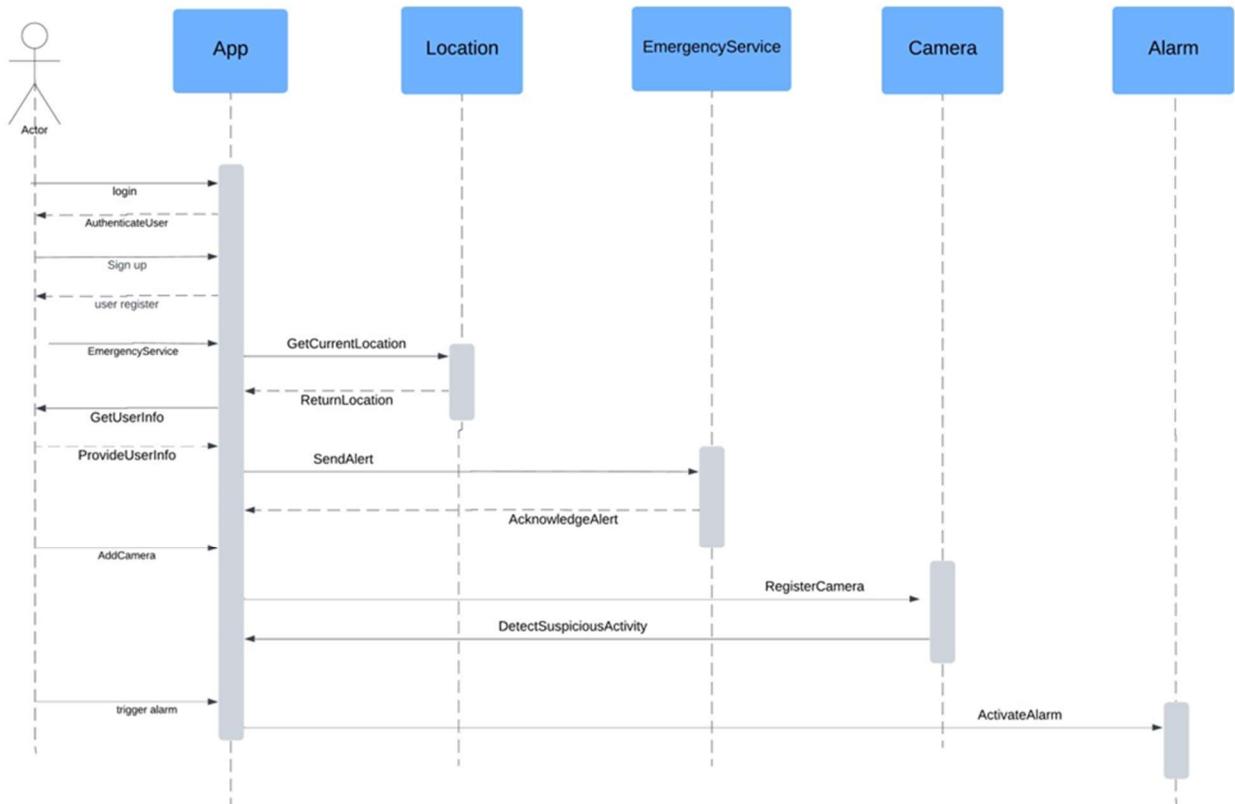


8. System Detailed Design

8.1 Class diagram

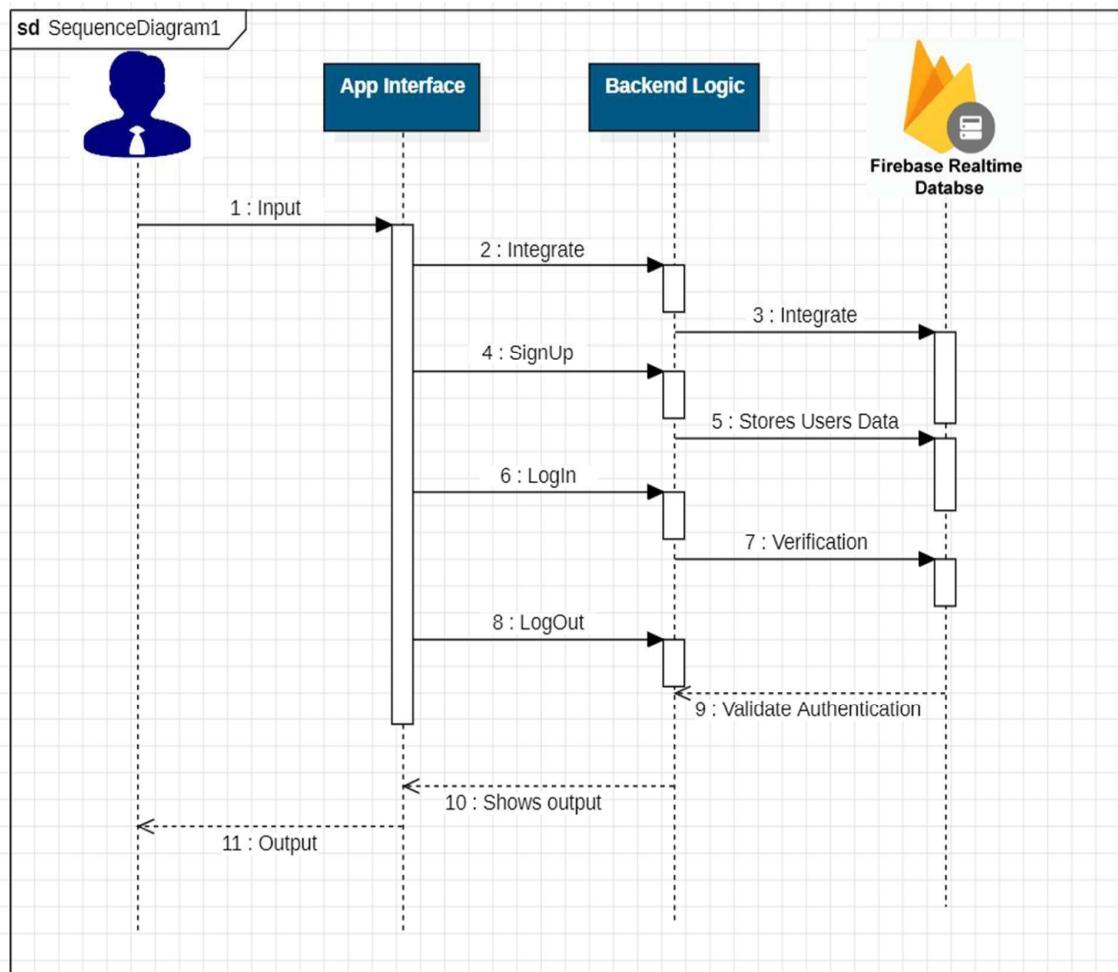


8.2 Comprehensive Sequence diagram

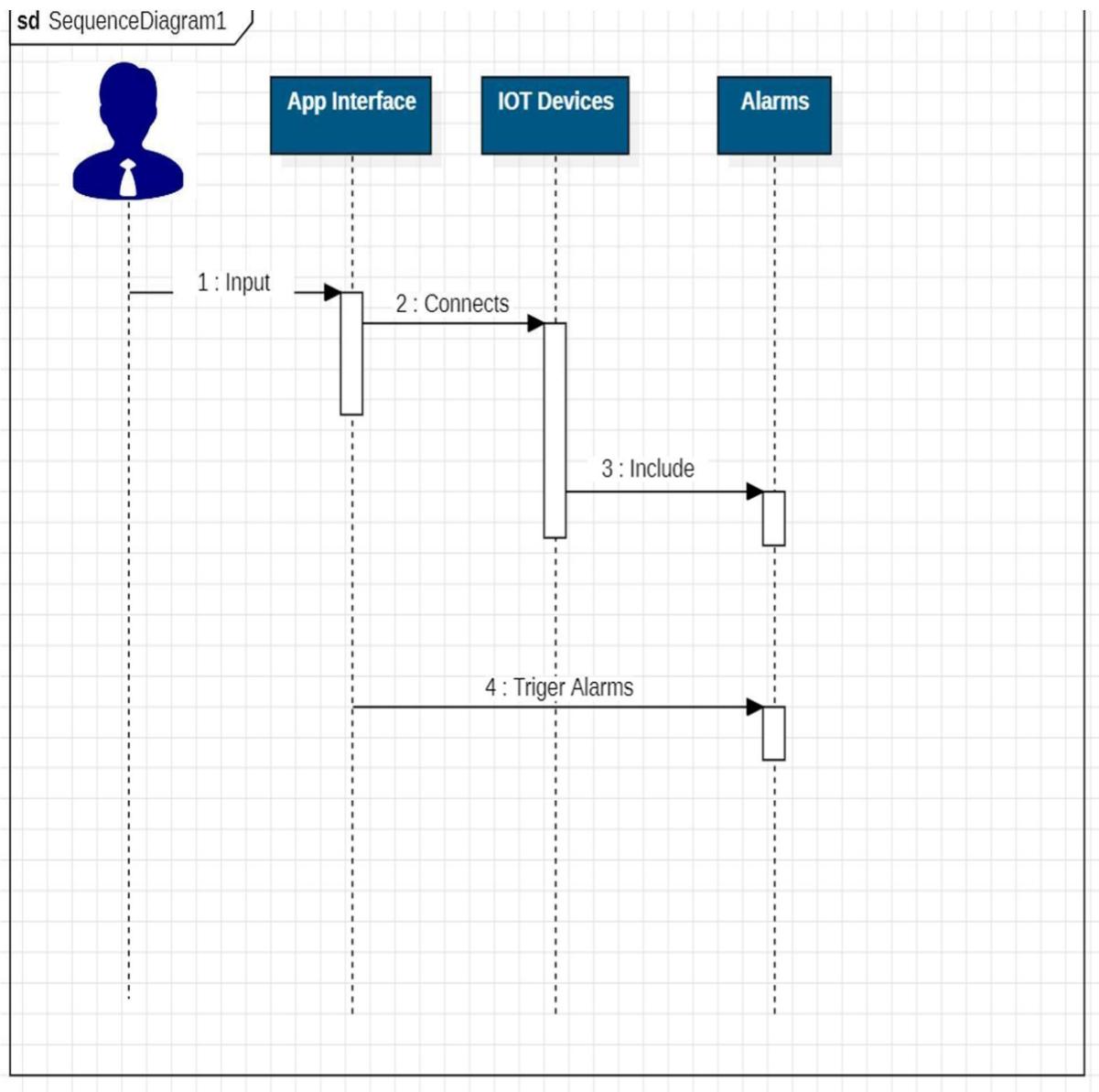


8.3 Detailed Sequence diagrams with parameter list

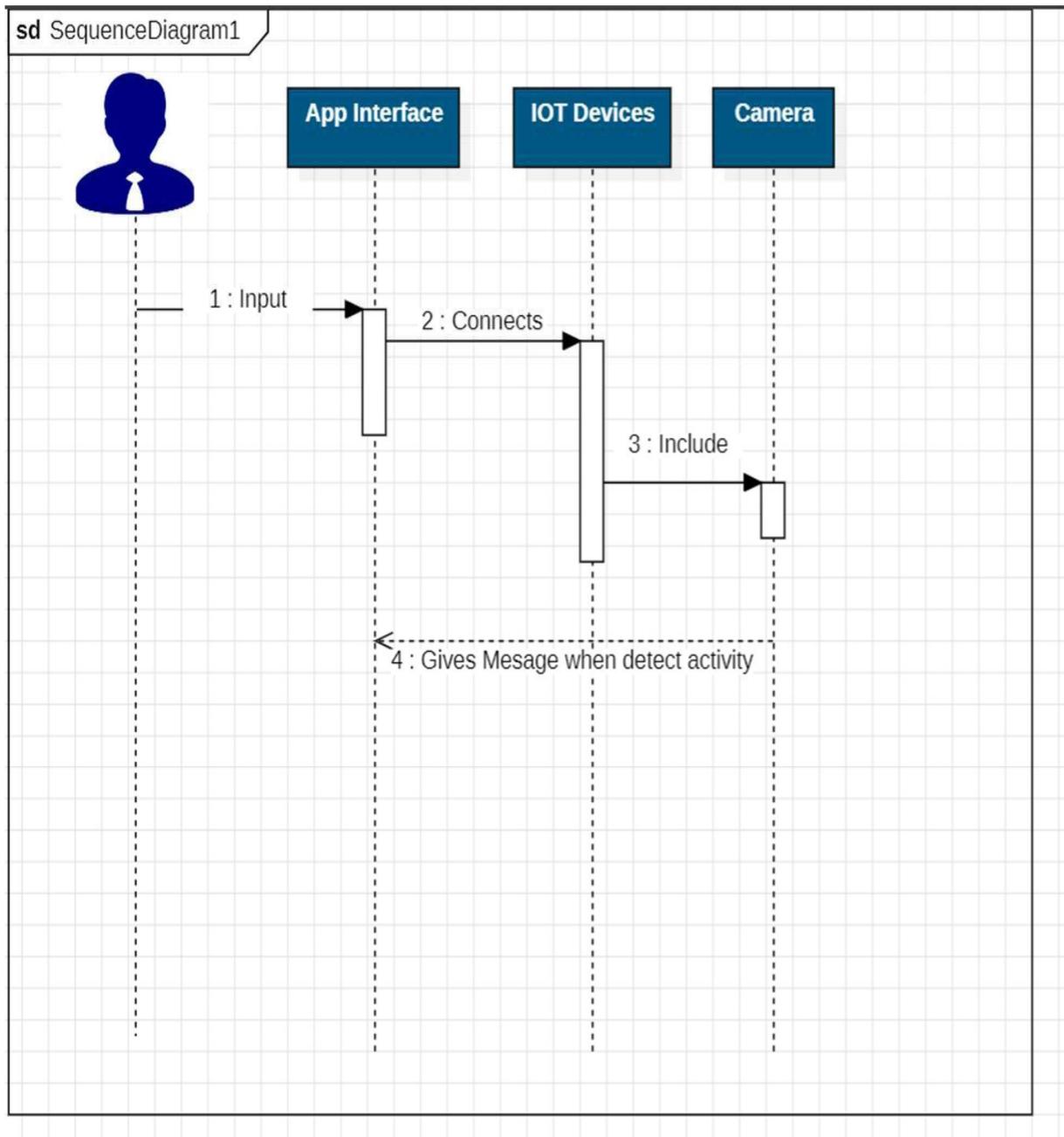
o User Authentication:



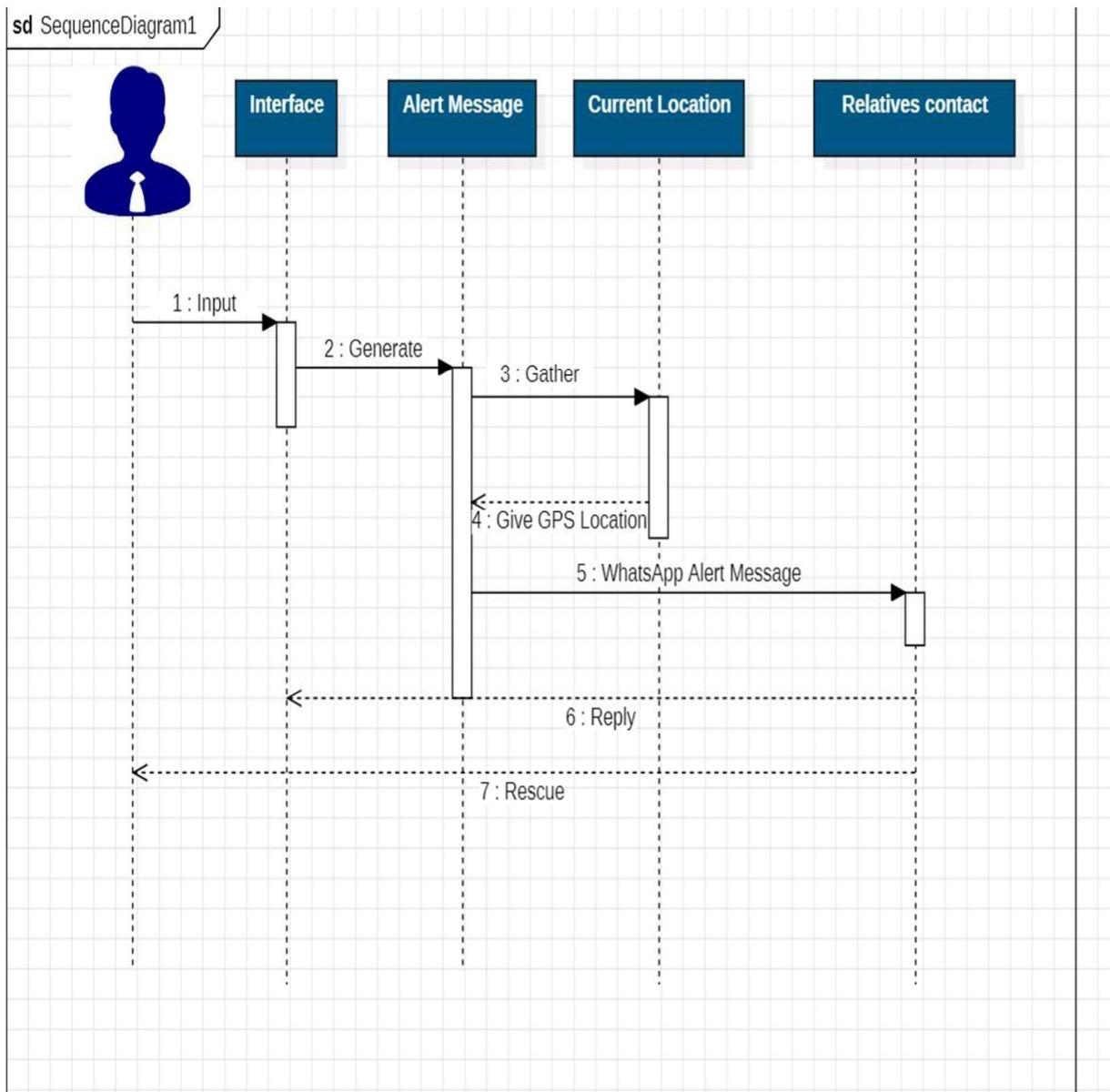
o IOT Devices (Alarms):



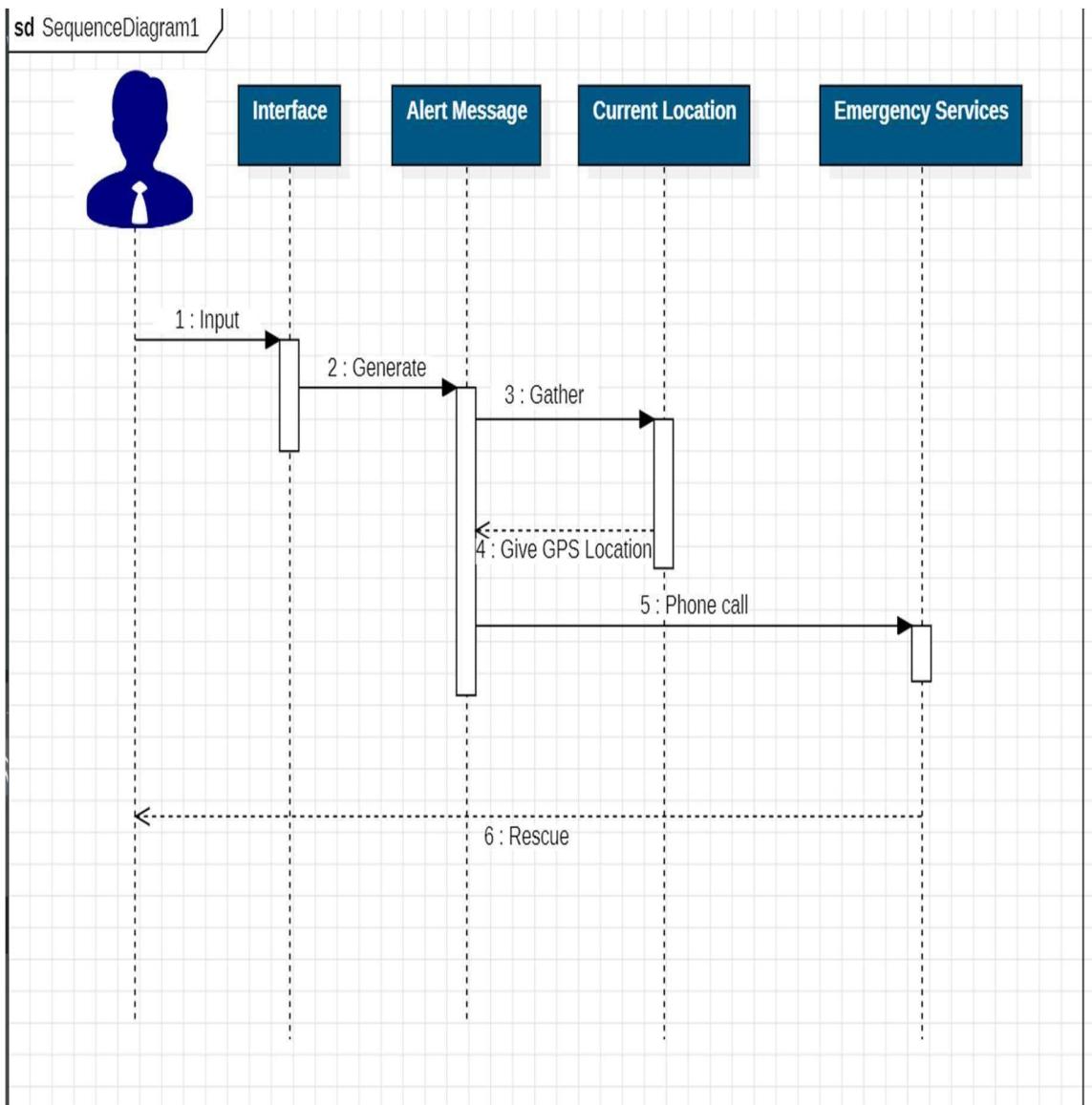
- o IOT Devices (Camera) and Suspicious Activity Detection:



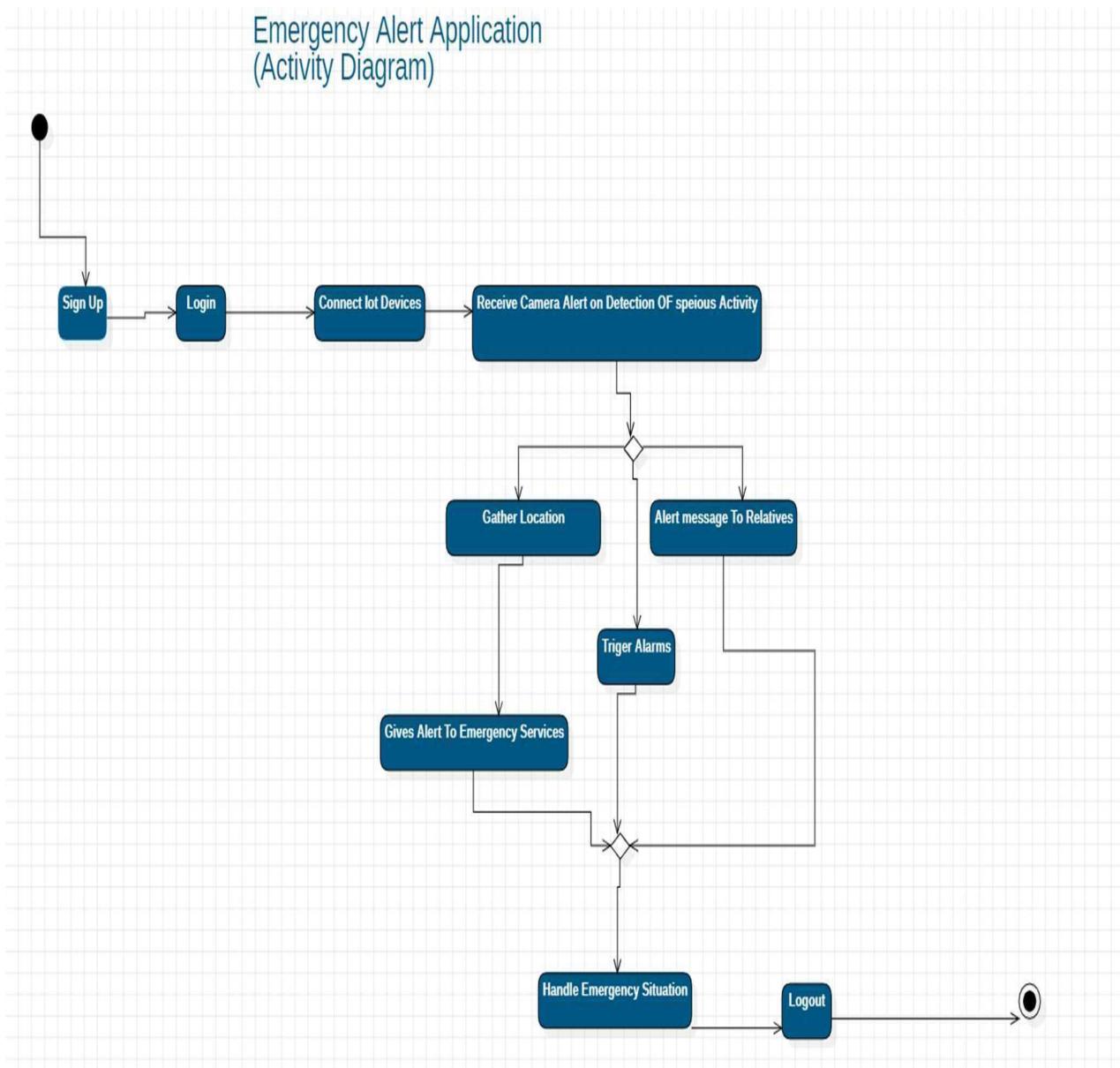
o Alert to Relatives:



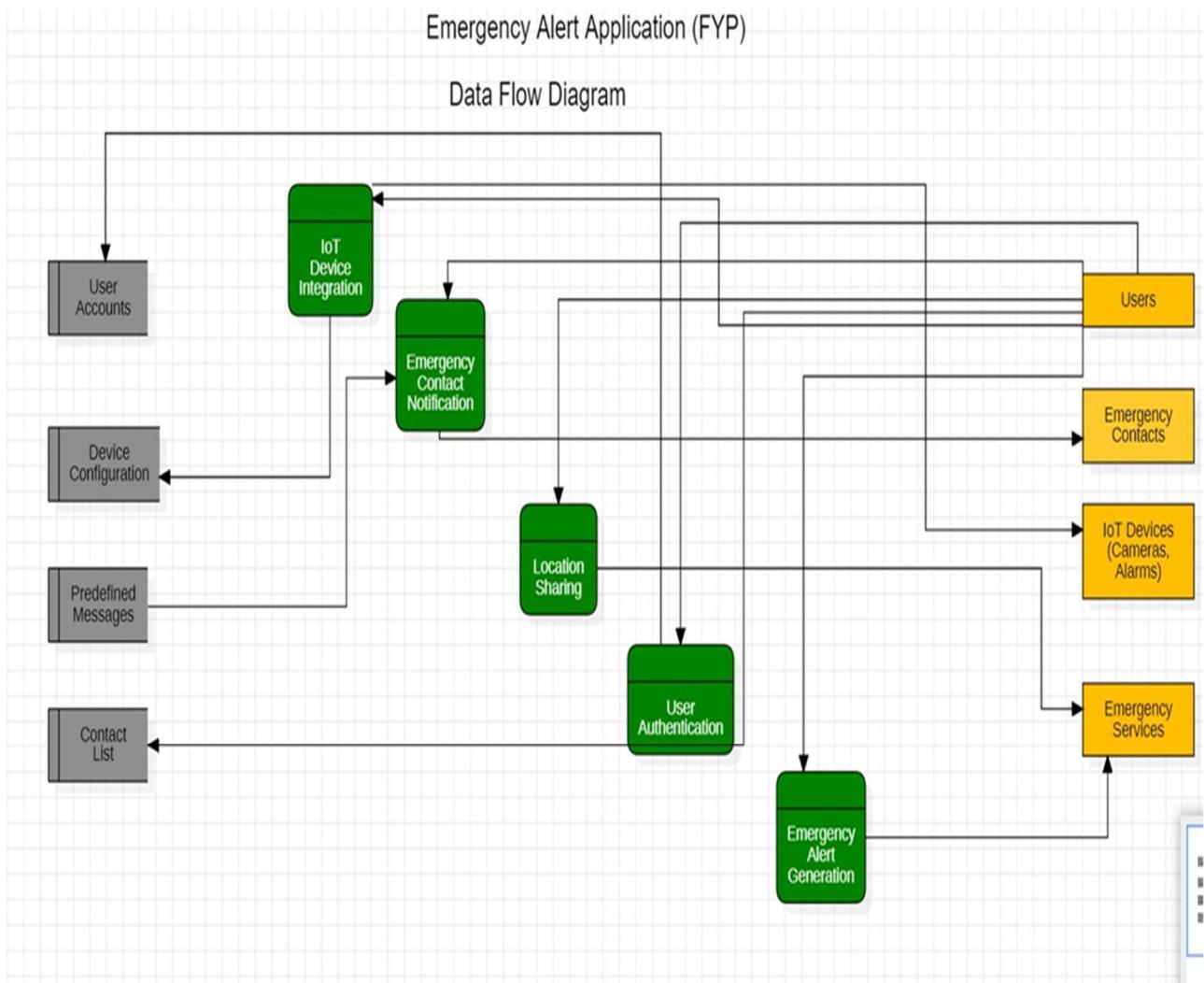
o Alert call to Emergency Services:



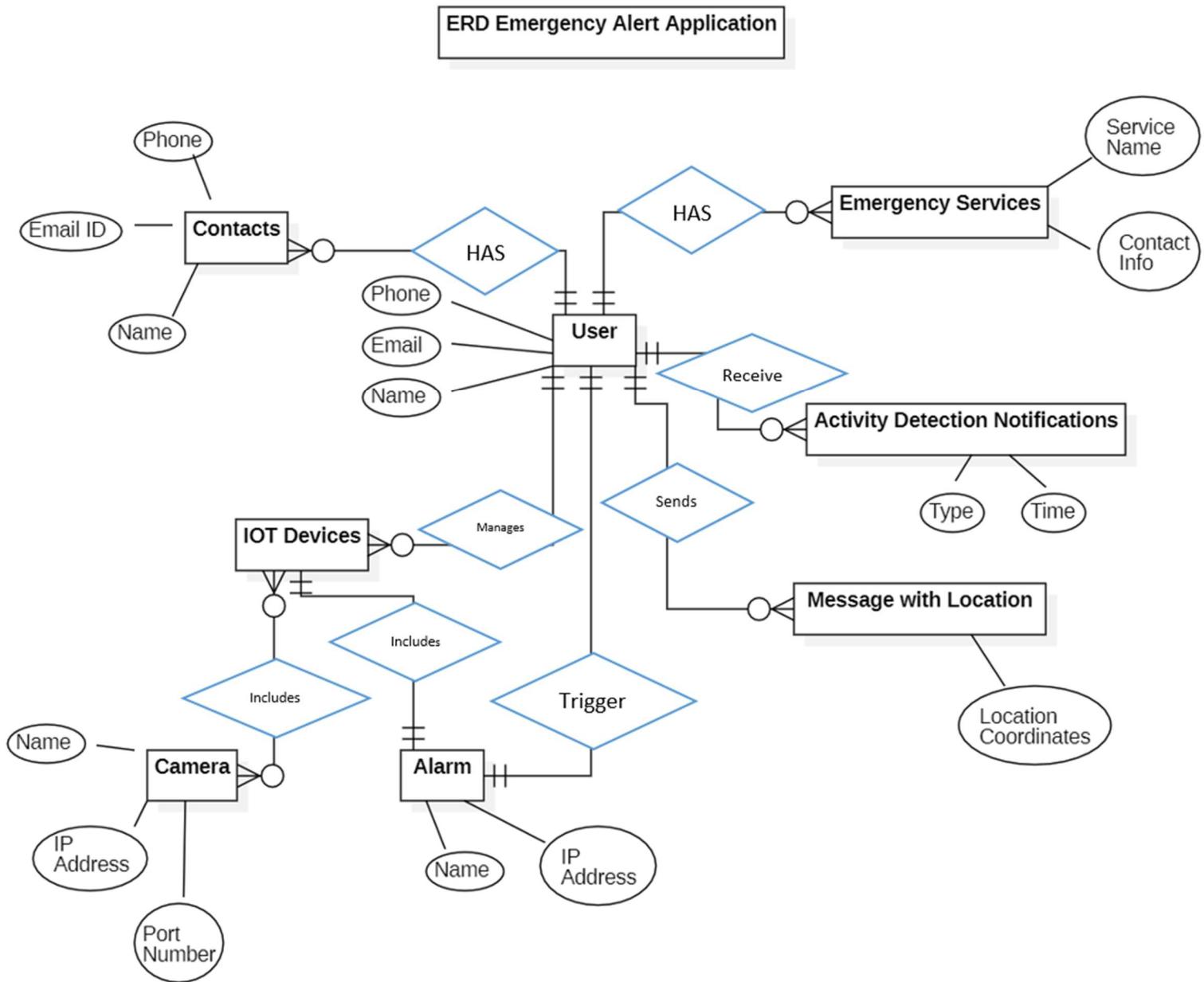
8.4 Activity Diagram



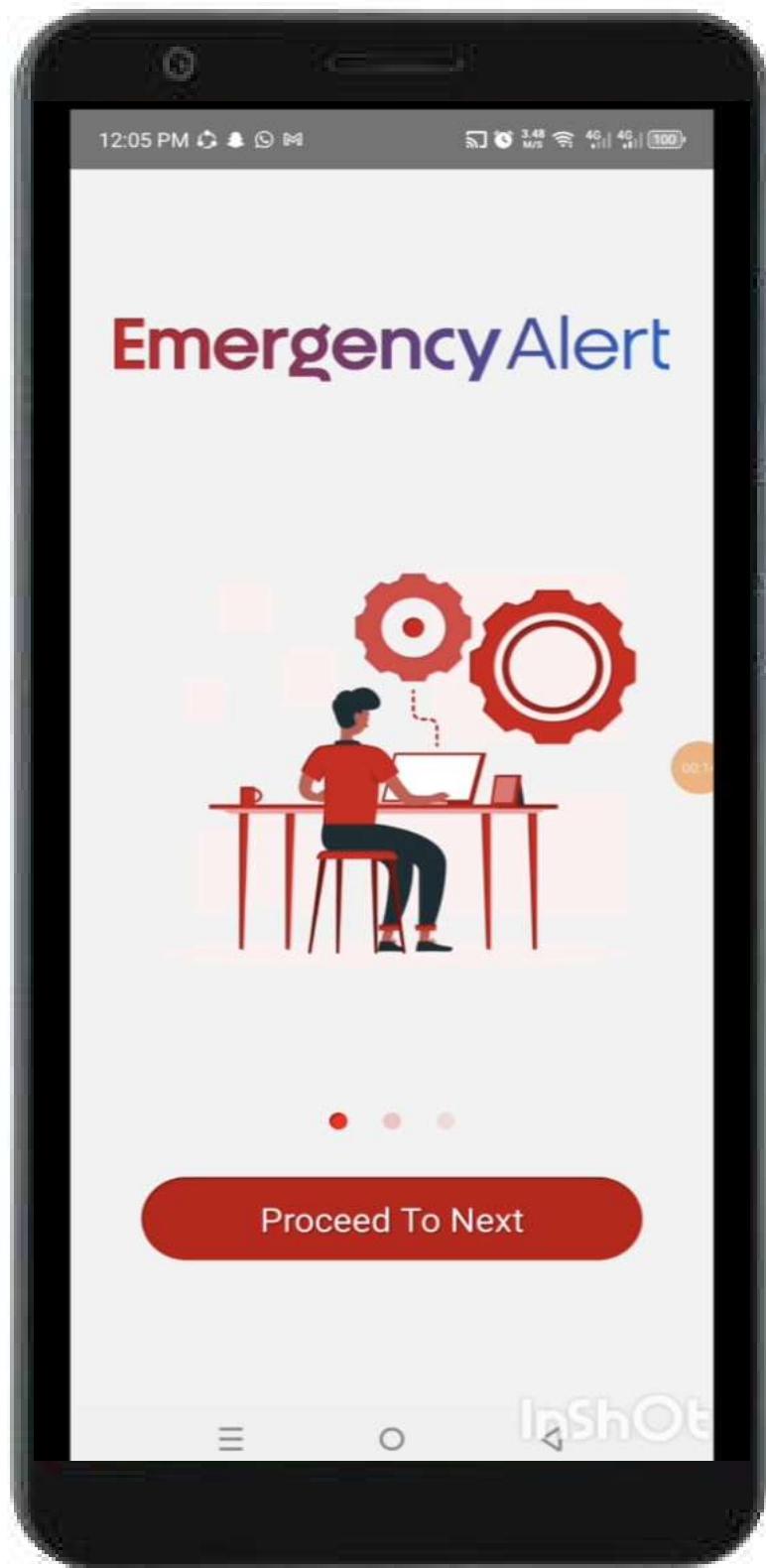
8.5 Data Flow Diagram

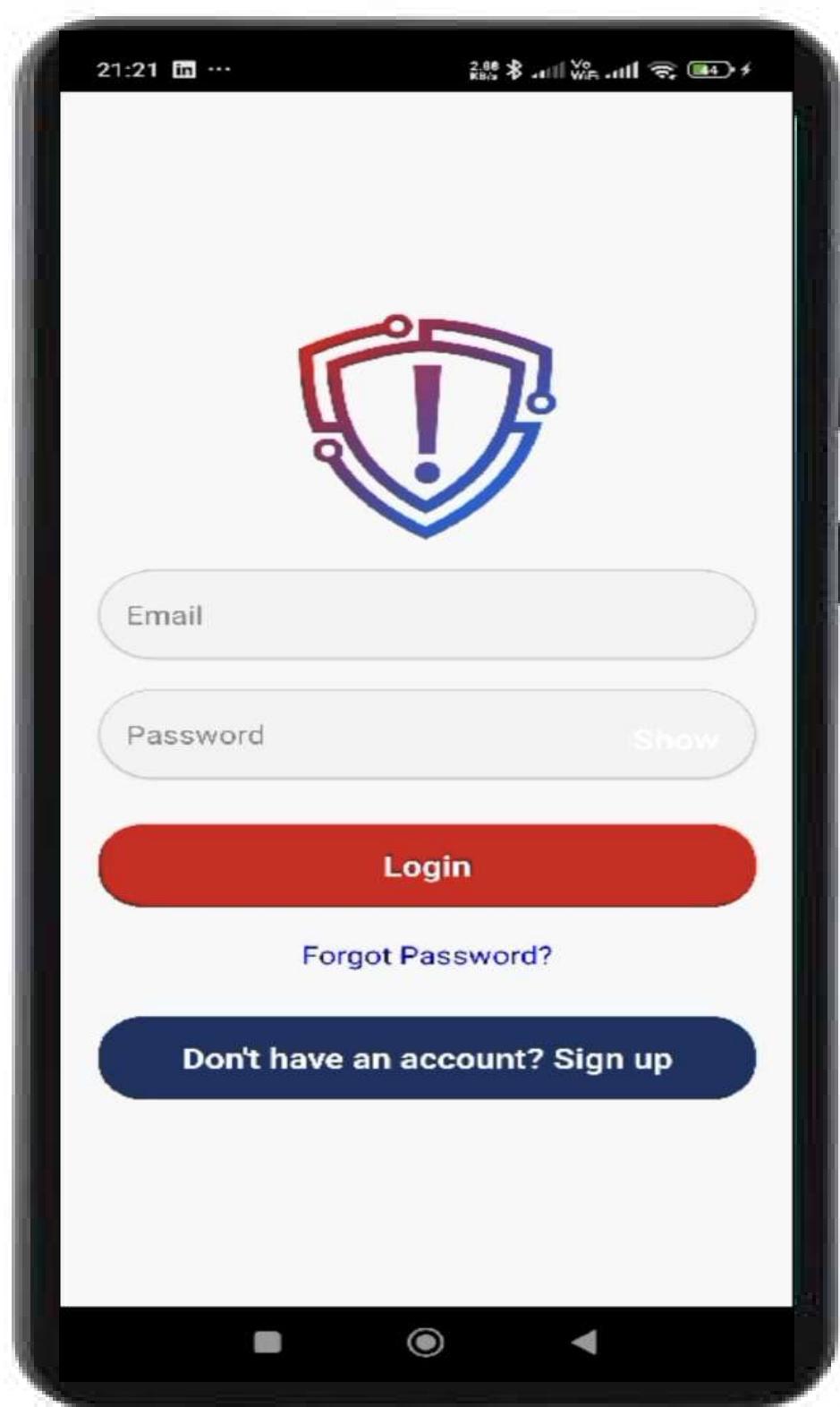


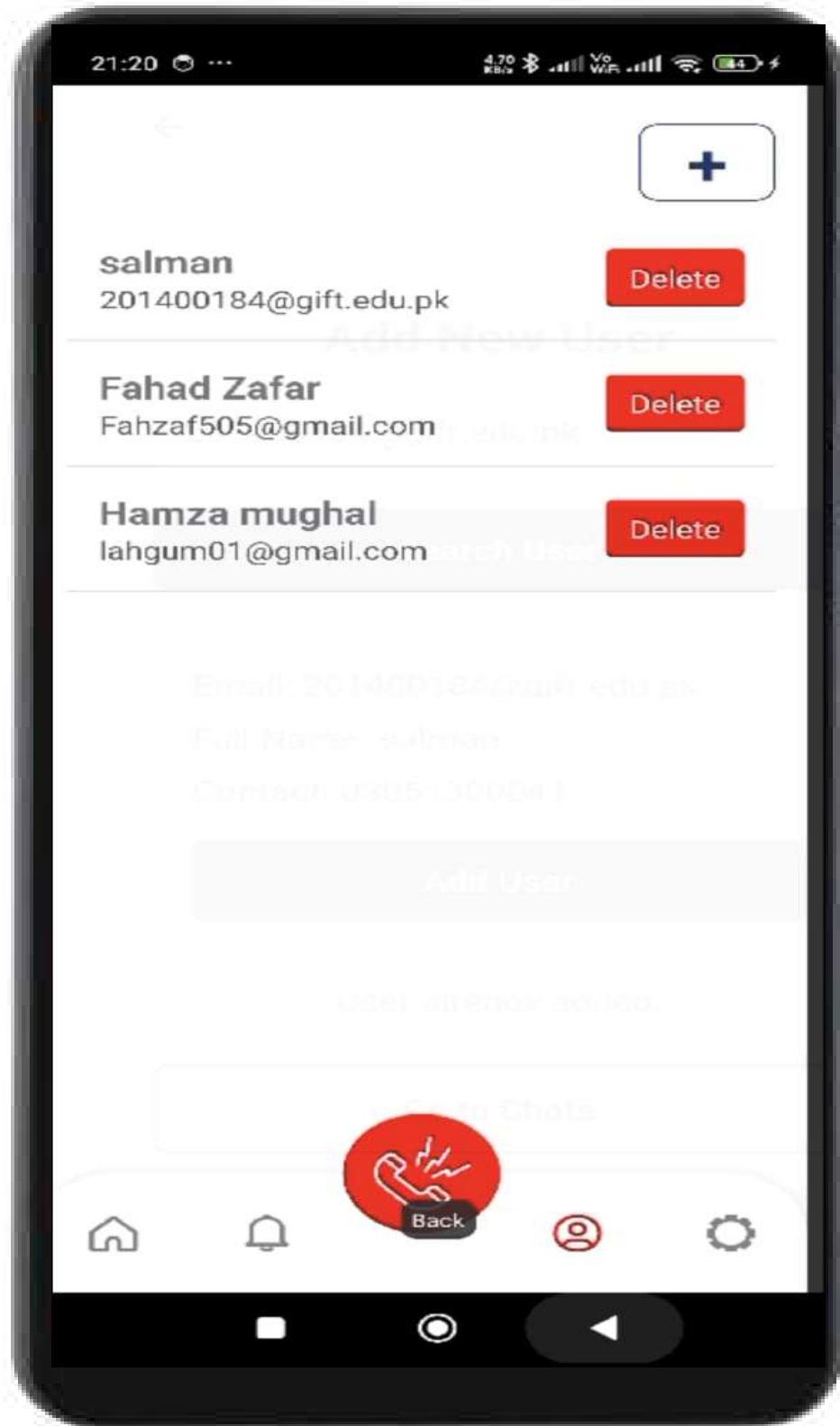
8.6 ERD

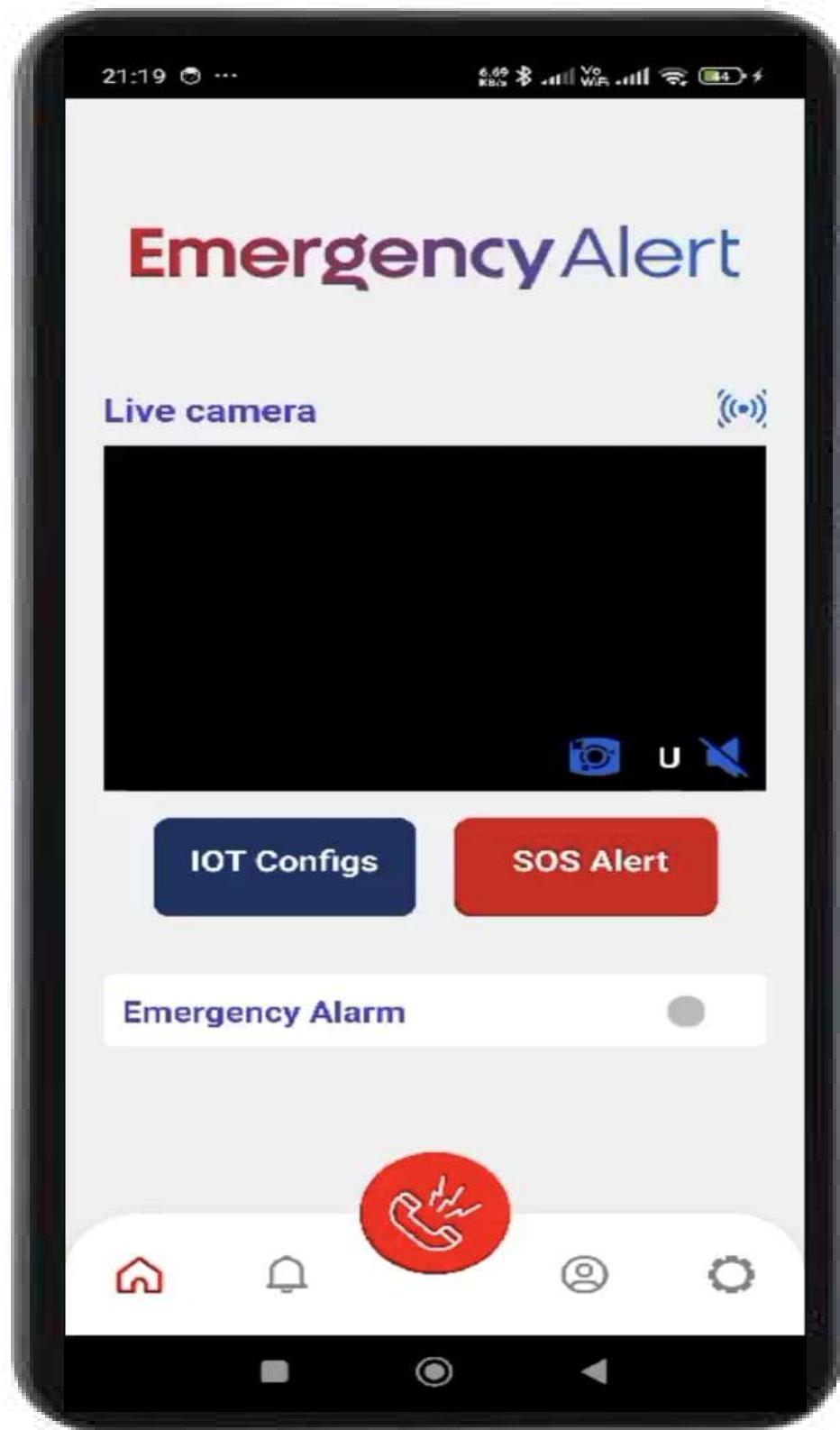


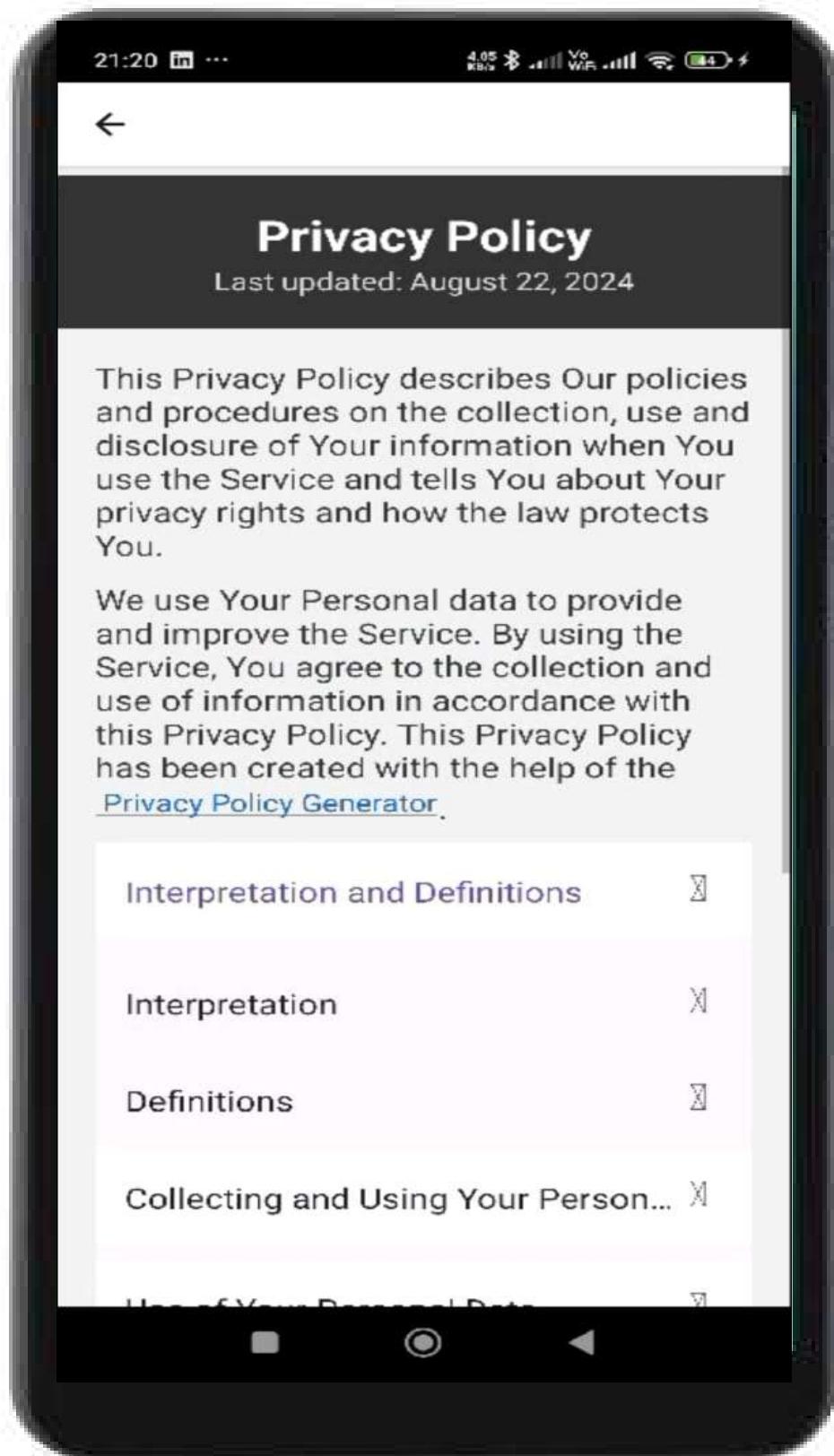
9. System Mockups/UI

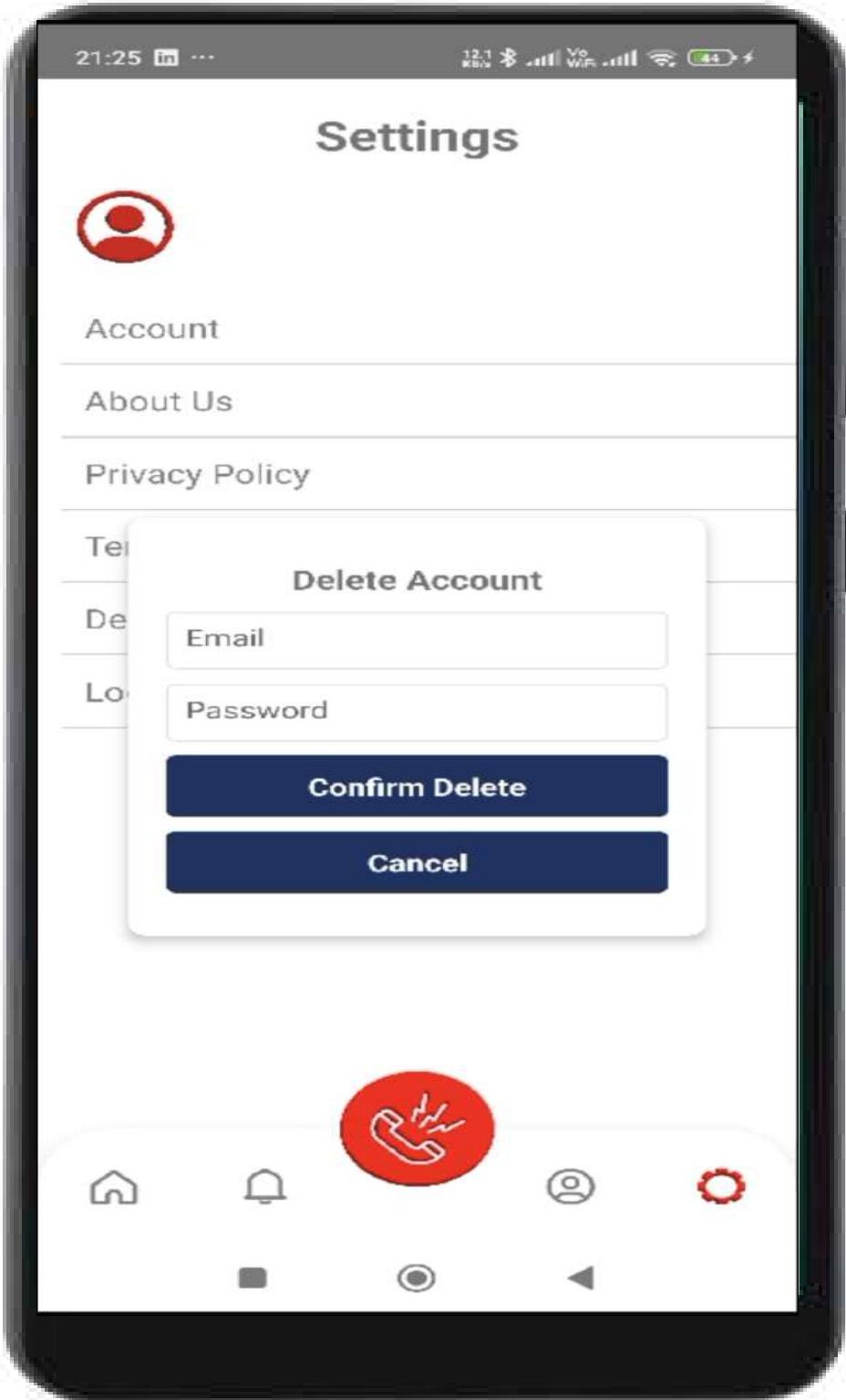


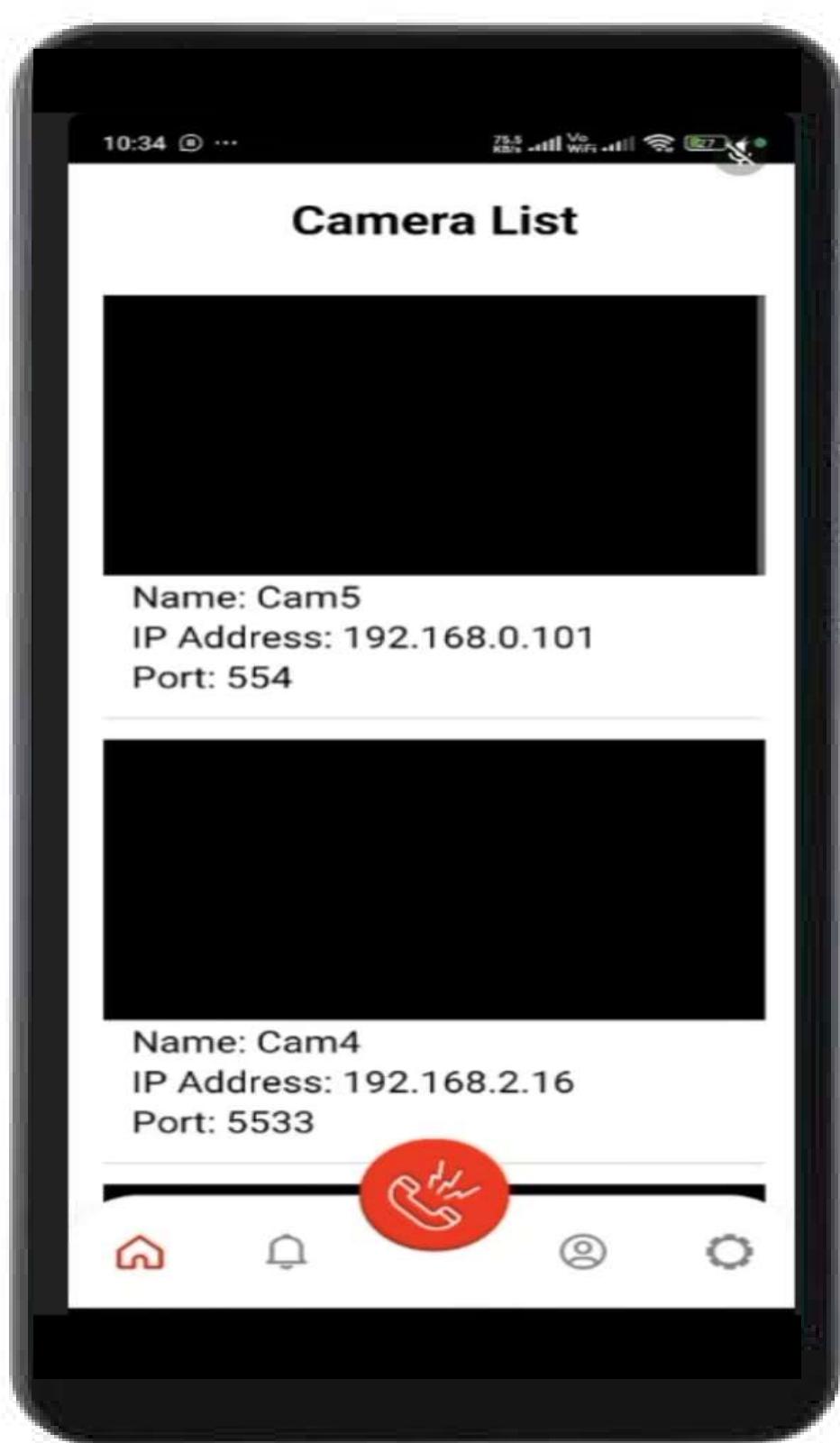




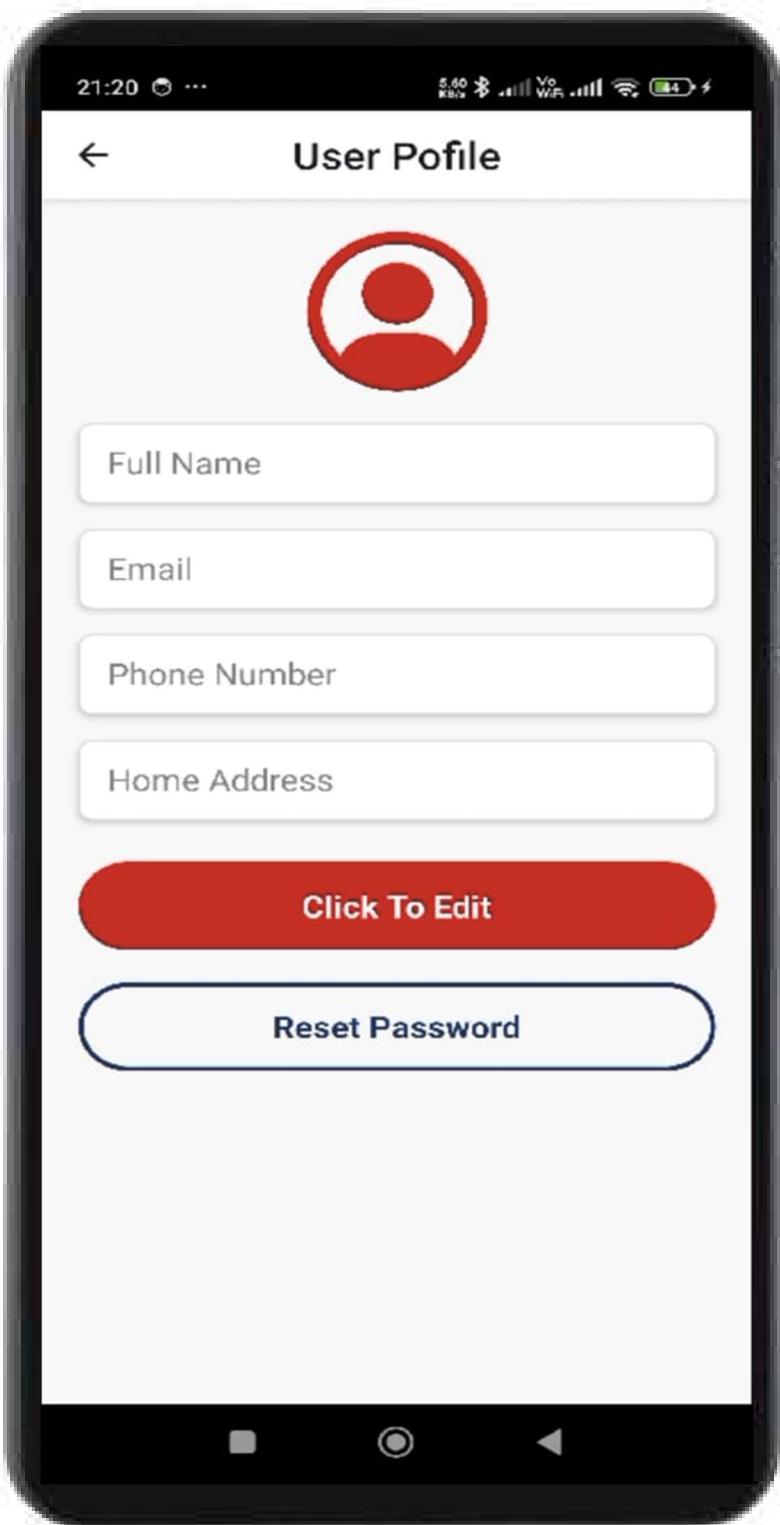


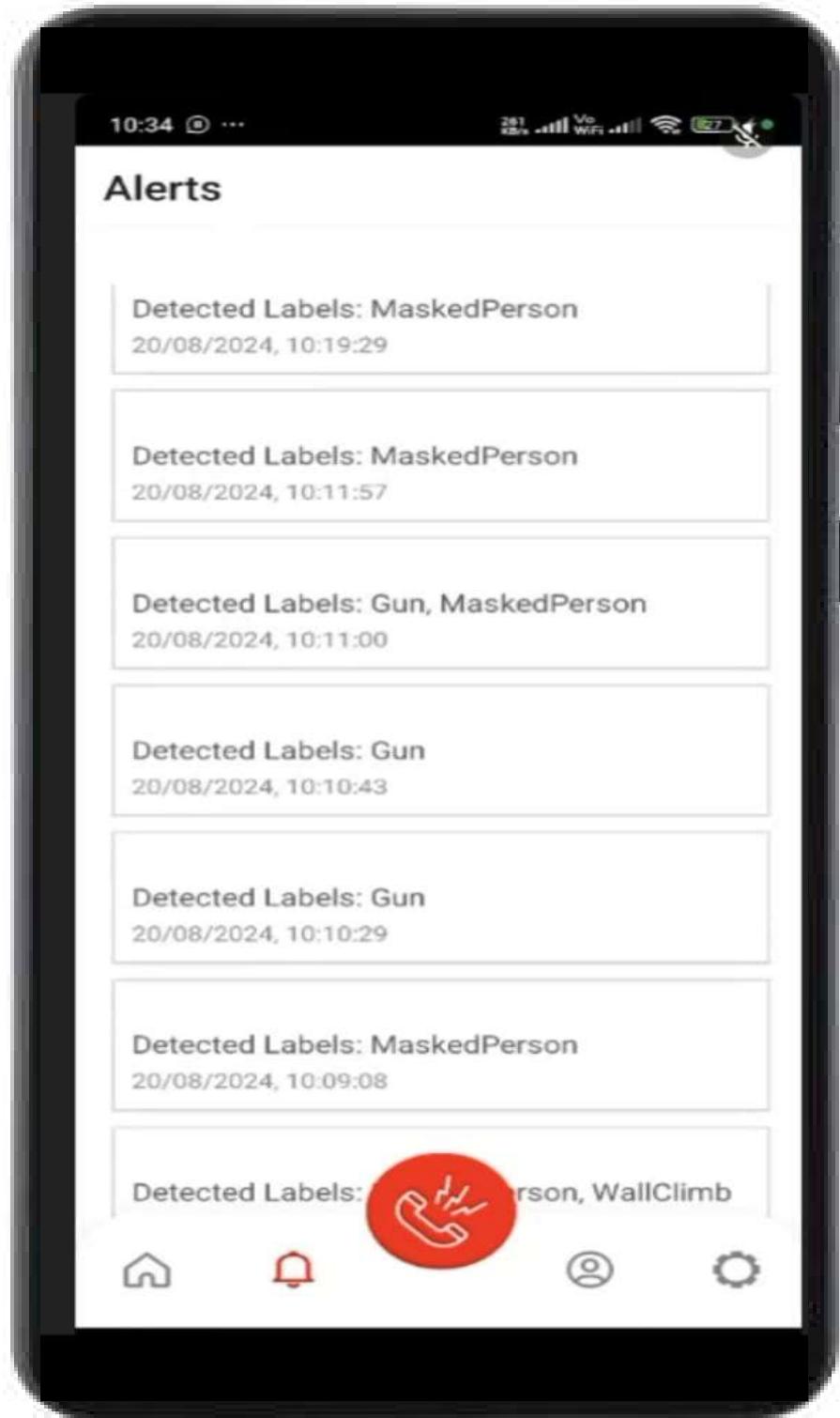


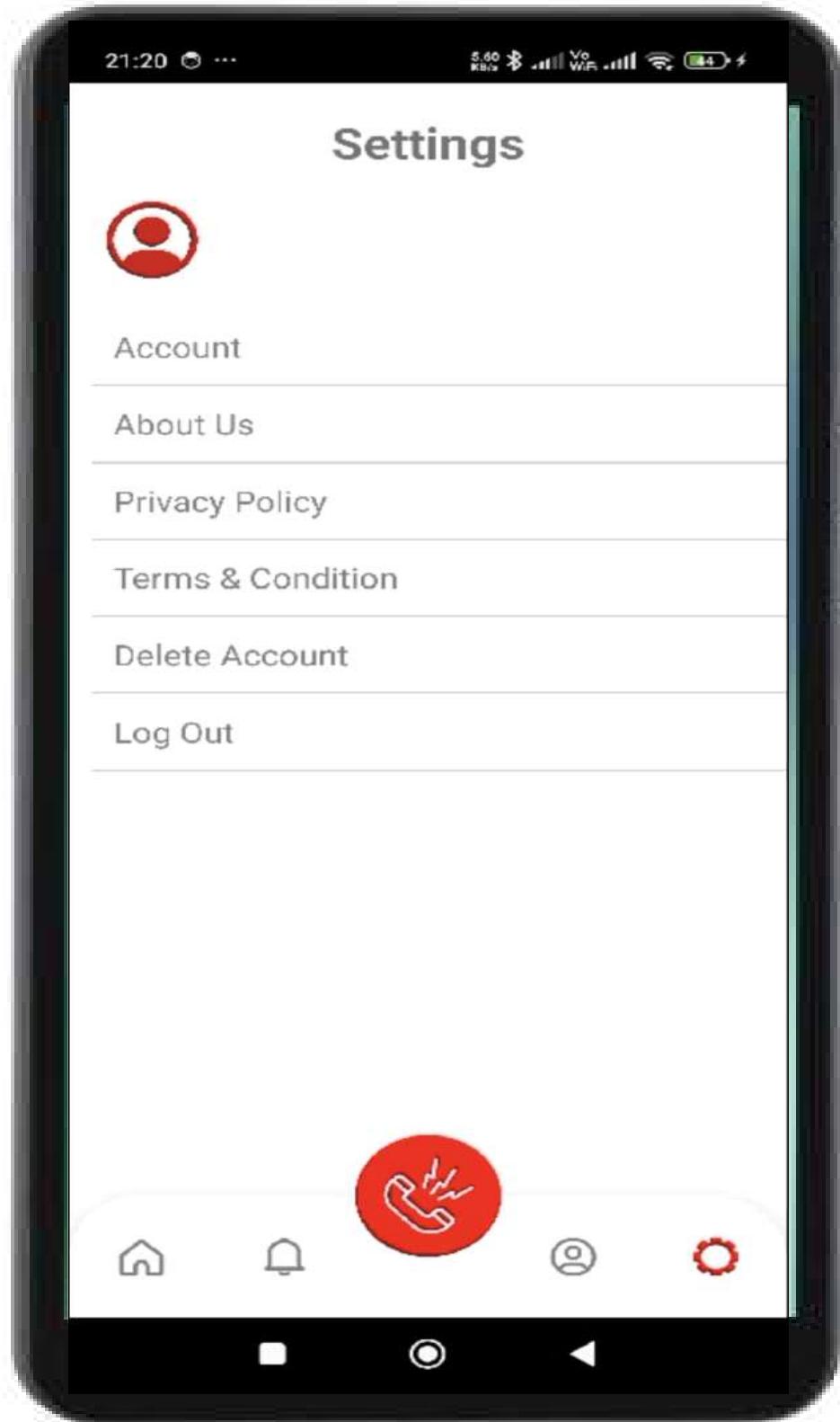


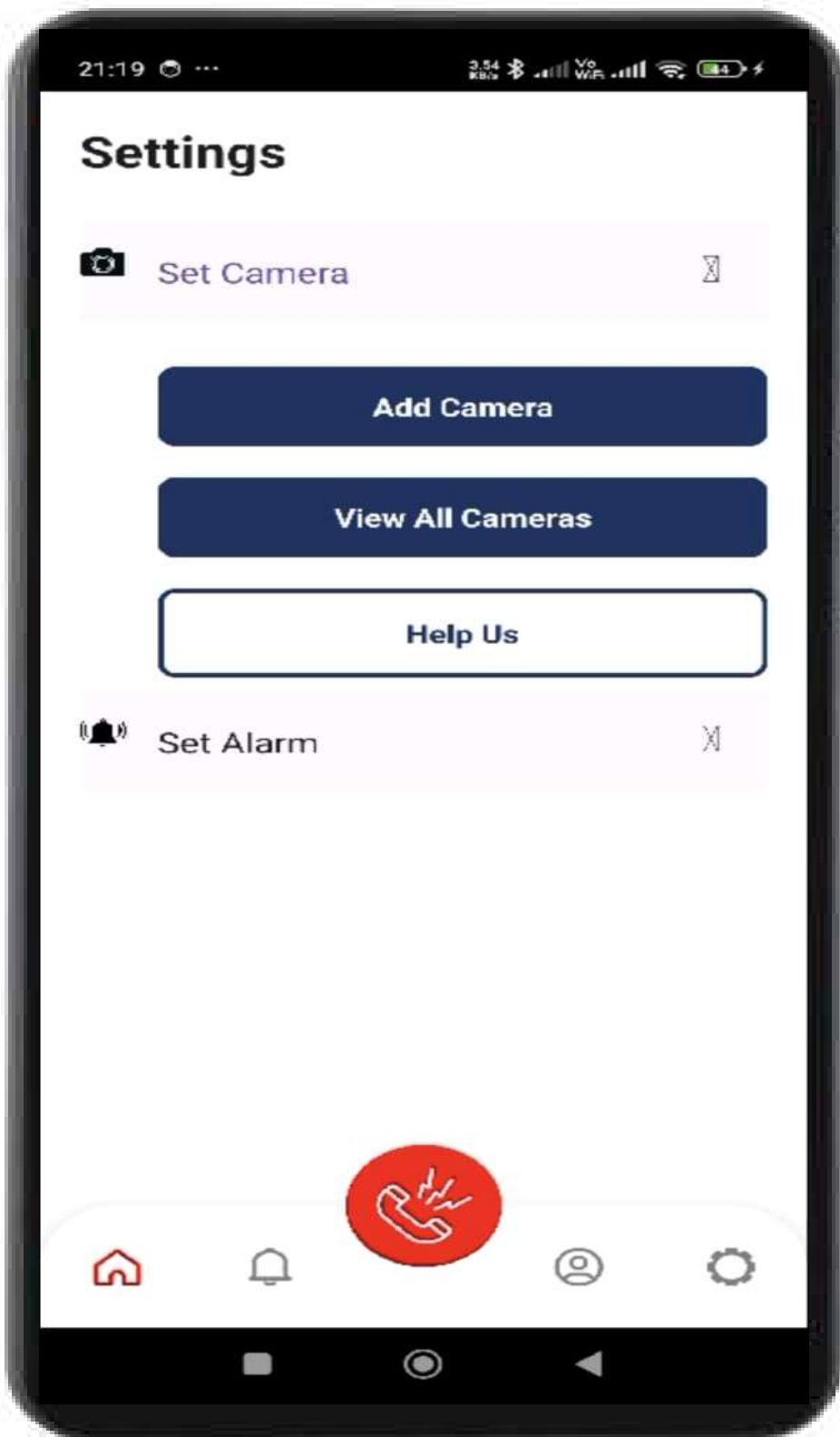


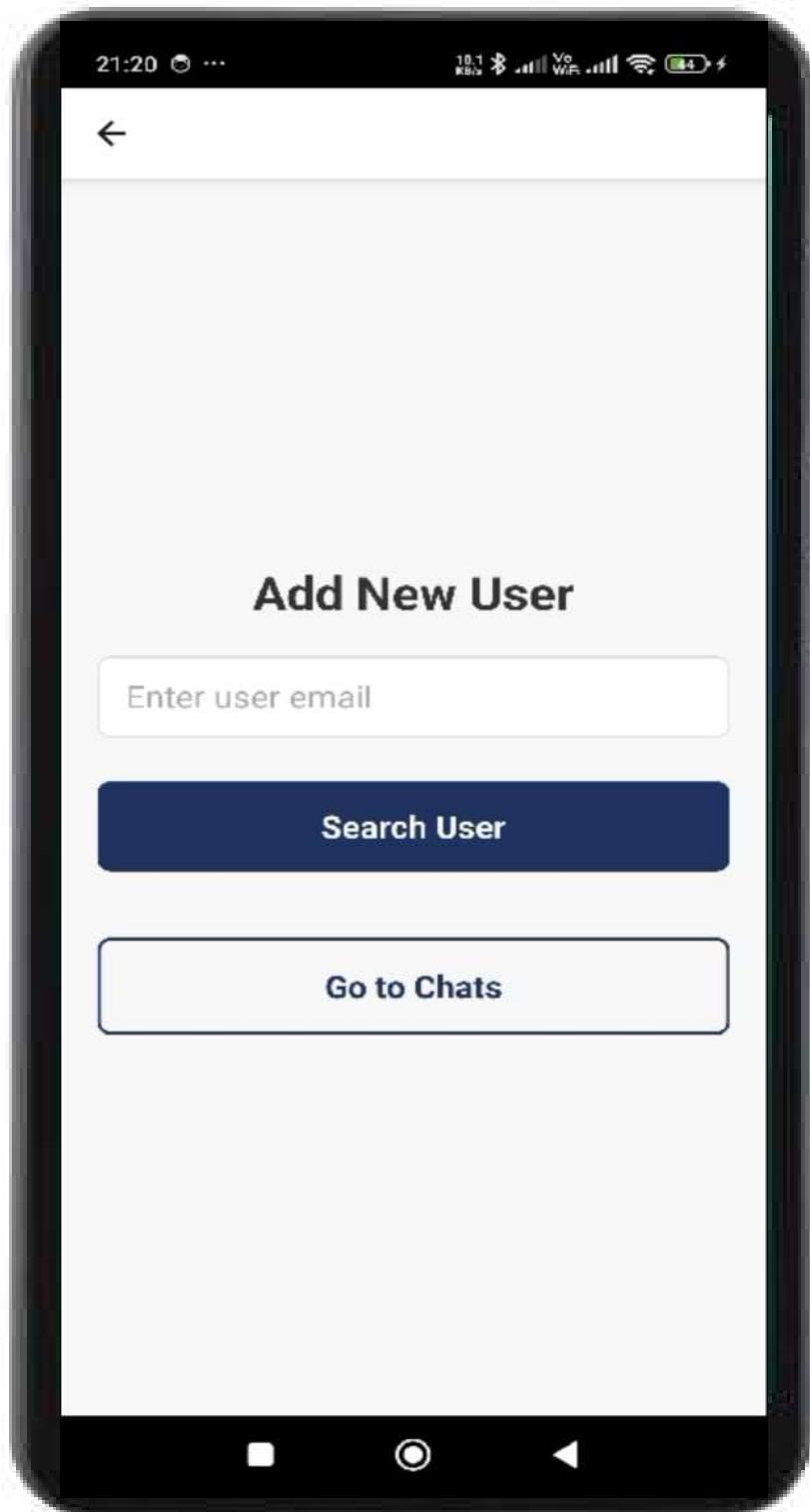












10. References

- i. M. A. Al-Taee, K. Al-Nuaimy, and W. Al-Ataby, "Mobile-Based Emergency Response System: Design and Implementation," in *Journal of Information Technology & Software Engineering*, vol. 5, no. 1, 2015.
- ii. F. Restuccia, S. D'Oro, and T. Melodia, "Securing the Internet of Things in the Age of Machine Learning and Software-defined Networking," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4829-4842, Dec. 2018.
- iii. R. Lu, X. Lin, X. Liang, and X. Shen, "A Secure Handshake Scheme with Symptoms-Matching for Healthcare Social Network," *Mobile Networks and Applications*, vol. 16, no. 6, pp. 683-694, 2011.
- iv. "Guidelines for Developing Location-Based Applications," World Wide Web Consortium (W3C), [Online]. Available: <https://www.w3.org/TR/location-API/>
- v. Z. Zhou, M. Dong, K. Ota, G. Wang, and L. T. Yang, "Secure and Privacy Preserving Protocol for Cloud-based Vehicular DTNs," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 6, pp. 1299 1314, 2015.

11. Appendices

Appendix A: User Interface Mockups

High-fidelity designs and wireframes of the application's user interface.

Appendix B: Emergency Scenarios and Response Protocols

Detailed descriptions of emergency scenarios and the corresponding app behavior.

Project Testing

12. Introduction

12.1 Purpose

The purpose of this test plan is to ensure the comprehensive testing and verification of the Emergency Alert and Location Sharing Application. This document will be utilized by the entire testing team to confirm that the system functions as intended and meets all specified requirements. It serves as a guide to ensure that the application is thoroughly tested, identifying and addressing any potential bugs or issues before deployment. The ultimate goal of this document is to guarantee that the final product is reliable, secure, and fully operational, providing users with a dependable emergency response tool.

In the Today ‘Society, people in need of immediate help usually had to telephone hotline services, which could cause delays and result in the loss of vital time in urgent situations. The location and specifics of the emergency were frequently communicated over the phone, which led to errors and inefficiencies that delayed the arrival of emergency personnel on time. Another issue was the widespread deployment of CCTV cameras in homes and workplaces, which frequently resulted in a lack of ongoing oversight of the recorded material. Without real-time activity monitoring, incidents can happen without prompt notice or action, exposing people and property at risk from a variety of dangers.

12.2 Scope of Testing

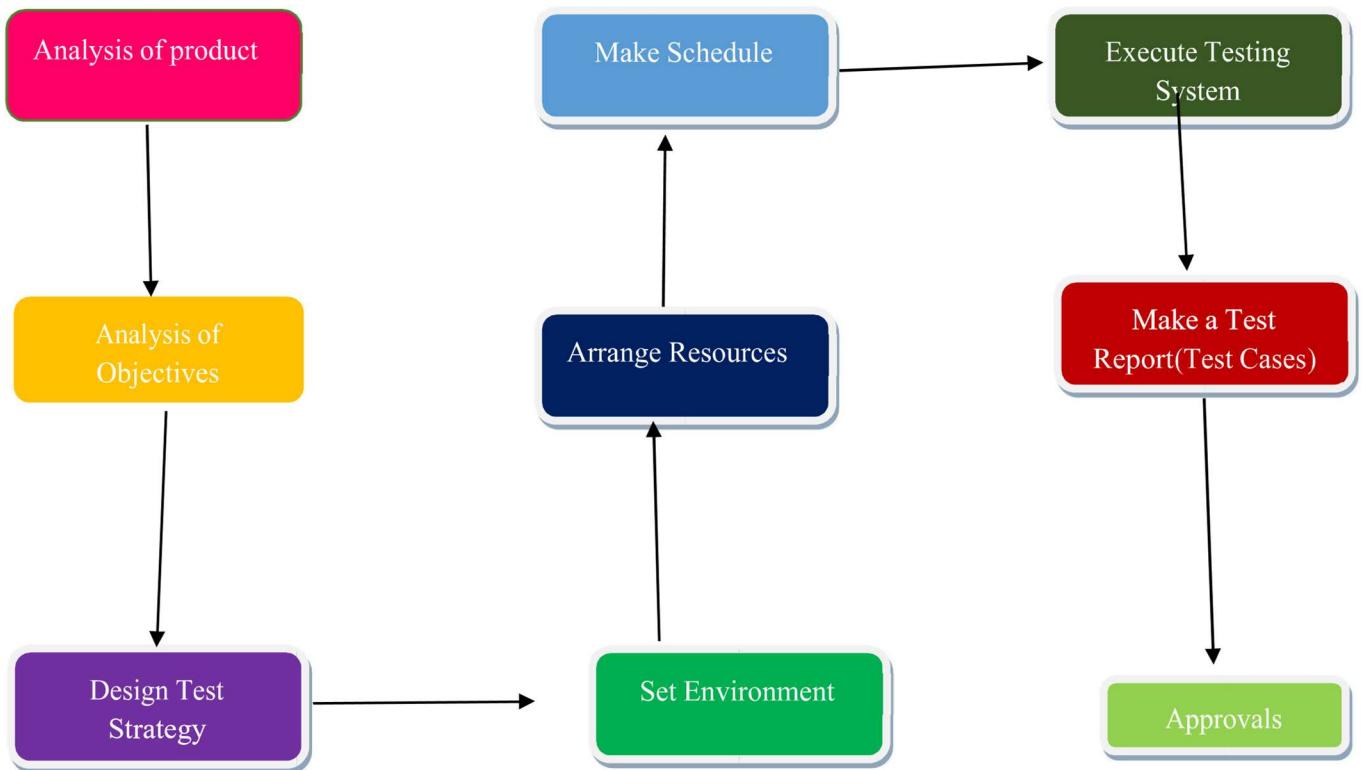
The scope of testing for the Emergency Alert and Location Sharing Application focuses on ensuring that all functionalities related to emergency handling, suspicious activity detection, communication, and data security work as intended. The testing will cover both functional and non-functional aspects of the application, ensuring that it meets all specified requirements and delivers a reliable and user-friendly experience, even under stressful conditions.

The Emergency Alert Application aims to give a best service to handle an emergency situation through its ml trained model for Suspicious activity detection in which the CCTV footage will integrate with our server and then application’s algorithm will detect activity such as wall climbing, covered face, weapons on runtime if anyone can try to make violence and can record in footage and as well as system generate a alert to user to see whether the situation is critical or not and the user can trigger IOT base alarm which is also connected to system, and can communication the other contacts or Emergency Services like, police, fire brigade, ambulance. Designing a simple and intuitive interface that can be easily understood and operated by users of varying ages, abilities, and technical expertise, even under stressful conditions. Address potential privacy concerns regarding location tracking and intrusion detection by providing clear information about data collection, usage, and sharing practices, and offering user control over settings.

13. Test Plan

13.1 Test Plan Strategy

The objective of this test plan is to ensure that there is a uniform understanding among anyone who reads the test plan, regarding the test methodology as well as the number of resources, schedules, etc.



Unit Testing

Definition:

Unit testing involves testing individual functions, programs, or subroutines in isolation from the entire system. Each function or module is tested independently to verify that it performs as expected, in a standalone environment. This process helps to ensure that the smaller building blocks of the software are functioning correctly before they are combined and tested as part of the larger system.

Participants:

- Group Members

Methodology:

- a. Initial Unit Test: Unit tests will be conducted concurrently with development, where each function or module is tested immediately after it is written.
- b. Modular Testing: As modules are completed, unit tests will be conducted on these modules to identify and rectify any inconsistencies or errors.
- c. Individual Module Testing: Each module will undergo independent testing to verify that it

performs its designated task accurately, without reliance on other modules or systems.

Integration Testing

Definition:

Integration testing involves combining individual software modules that have already been unit tested and then testing them as a group. The goal is to identify any issues that arise from the interaction between modules. Integration testing bridges the gap between unit testing and system testing, ensuring that different components work together correctly before the system is fully integrated.

Participants:

Group Members Methodology:

- a. Module Integration: In this phase, all unit-tested modules are integrated with each other, creating larger aggregates of the system.
- b. Group Testing: The integrated modules will be provided with various inputs, and their outputs will be evaluated to ensure that they work together as expected.

System Testing

Definition:

System testing is conducted on the complete, integrated system to evaluate its compliance with the specified requirements. It is a black-box testing method, meaning that it tests the system without needing to understand the internal code or logic. System testing aims to detect any defects within the entire system, both in terms of the software components and their interaction with any applicable hardware systems.

Participants:

- Group Members

Methodology:

- a. Full System Integration: In this phase, all previously integrated modules are combined into one complete system.
- b. Comprehensive System Testing: The entire system is subjected to various inputs to observe its behavior and ensure that it produces the expected outputs.
- c. Issue Resolution: If the system generates any unfavorable outputs, further investigation will be conducted to determine whether the issue lies within the system's integrated modules or within the system as a whole. Necessary adjustments will be made to resolve these issues.

Performance Testing:

- o Testing the application's performance under various conditions, including load testing, stress testing, and testing under poor network conditions to ensure the system is responsive and stable.

Security Testing:

- o Ensuring that the application securely handles user data, including location information and personal contacts, and that it protects against unauthorized access and data breaches.

Usability Testing:

- o Testing the user interface to ensure it is intuitive and accessible for users of all ages and technical expertise, even in stressful situations.

Regression Testing:

- o Re-testing the application after any modifications to ensure that new changes do not introduce new bugs or negatively impact existing functionality.

User Acceptance Testing (UAT):

- o Final testing conducted with end-users to ensure the application meets their expectations and requirements in real-world scenarios.

13.2 Test Environment

The test environment for the Emergency Alert and Location Sharing Application is designed to simulate real-world conditions as closely as possible to ensure the reliability and effectiveness of the application. Below are the details of the test environment setup

Set Hardware's

- o Camera
- o Alarm

Set Application

- o Install and Initiate Android (latest version and several previous versions)
- o Windows 11 for server-side application management and monitoring

Testing Tools

- o Mask, Gun

Network Configuration

- o High-latency networks to test the application's real-time features.
- o Scenarios with intermittent connectivity to ensure the application handles network drops gracefully.

Test Data

- o User Data: Simulated user profiles, including contact information and location data, to test privacy controls.
- o Test cases involving multiple user scenarios to ensure the system correctly handles various emergency situations.
- o Machine Learning Model Training Data: Pre-loaded test scenarios to validate the accuracy of suspicious activity detection (e.g., mock CCTV footage with simulated incidents).

13.3 Schedule

Testing Activities	Begin	End	Person Responsible
Designing Test Cases	05-07-2024	16-07-2024	Group Leader
Executing Test Cases	17-07-2024	23-07-2024	Group Leader
Unit Testing	23-07-2024	26-07-2024	Group Leader
Integration Testing	27-07-2024	29-07-2024	Group Leader
System Testing	29-07-2024	01-08-2024	Group Leader

13.4 Control Activities

Date	Activity
05-08-24	Initial project kickoff meeting held to discuss objectives and outline project milestones.
10-08-24	Design specifications for UI/UX and IoT device integration are reviewed and approved.
15-08-24	Development team completes the integration of the machine learning model with CCTV footage.
18-08-24	Preliminary test cases for machine learning-based suspicious activity detection are created.
22-08-24	First round of functional testing on emergency dialer and communication system is executed.
25-08-24	Integration testing for IoT-based alarm systems and camera feed completed.
28-08-24	Feedback from initial usability testing session is reviewed and action items identified.
01-09-24	Performance and security testing begins, focusing on data encryption and network stability.
01-09-24	Final design adjustments made based on feedback; test case execution continues.
01-09-24	Comprehensive testing of privacy controls and user data management is conducted.

01-09-24	Project status meeting to review progress and resolve any outstanding issues.
01-09-24	Final test case execution and system validation are completed.
01-09-24	Results of testing activities reviewed and documented in preparation for final review.
01-09-24	Test plan, along with testing results and analysis, is submitted for final approval.

13.5 Resources

Hardware

- GPU Computer
- Mobile Phone
- Camera
- Alarms Speakers
- Wi-Fi modems
- WAN, LAN Wires

Software

- Google Colab
- CAVIT
- VS code
- Android Studio
- Kaggle online model train

13.6 Major Deliverables

These are the major deliverables of the testing phase.

- Test plan
- Test cases

13.7 Risk and Assumptions

- **System Response Time:**

If our system will not very fast or speedy, and user cannot handle the Emergency situation, then it might be possible it fails.

- **External Emergency Services Relief:**

If the external emergency services could not work on the computer generated call, then user can panic.

- **Camera Activity Detection:**

If our camera module should not work properly, so our system can fail to resolve the users' situation.

14. Test Report

14.1 Test Case Scenarios and Test Cases

No.	Chunks	Test Scenarios	Test Cases	Responsible
01	Registering	TS-01	TC_1-4	Group Members
02	User Log-In	TS-02	TC_5-7	Group Members
03	CCTV Camera Module	TS-03	TC_8-10	Group Members
04	Alarm Module	TS-04	TC_11-13	Group Members
05	Check Suspicious Activity Detection	TS-05	TC_14-17	Group Members
06	SOS Alert	TS-06	TC_18-21	Group Members
07	Initiate calls to emergency services	TS-07	TC_22-24	Group Members
08	View and Update Profile and deleted account	TS-08	TC_25-27	Group Members
09	Check terms and conditions , about us, privacy policy	TS-09	TC_28-31	Group Members
10s	User Log-Out	TS-10	TC_32	Group Members

Registering

ID	TS-01		
Description	User Can Successfully Registered		
Pre-Condition	User cannot registered		
Post-Condition	User is registered		
Test Steps	Test data	Expected Result	Actual Result
I. Verify that a User can successfully register with valid information.	Id:201400137@gift.edu.pk Password: Fahad1234	Pass	Pass
II. Verify that a User cannot register with invalid.		Pass	Pass
III. Verify that a User cannot register with Incomplete.		Pass	Pass
IV. Check for duplicate registrations with the same email or phone number		Pass	Pass

T_ID:	001	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Sign-Up Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify that a User can successfully register with valid information.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Click Sign UP button	Signup form Should be open	As Expected	Pass	
02	Enter data	Successfully data will enter	As Expected	Pass	
03	Register	Not give any error	As Expected	Pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	002	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Sign-Up Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify that a User cannot register with invalid.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Click Sign UP button	Signup form Should be open	As Expected	Pass	
02	Enter Invalid Data	Give error	As Not Expected	Fail	
03	Click Register Button	Give alert or error for invalid Info	As Expected	Pass	
Net Result	Pass				
Reviewer Comments	Good				

T_ID:	003	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Sign-Up Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify that a User cannot register with Incomplete. .				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click Sign UP button		Signup form Should be open	As Expected	Pass
02	Enter Incomplete Data		Give error	As Not Expected	Fail
03	Click Register Button		Give alert or error for Incomplete Info	As Expected	Pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	004	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Sign-Up Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Check for duplicate registrations with the same email or phone number				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click Sign UP button		Signup form Should be open	As Expected	Pass
02	Enter same data again in form		Entered	As Expected	Pass
03	Click Register data		Give alert or error for same	As Expected	Pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

User Log-In

ID	TS-02		
Description	User can Login to Application		
Pre-Condition	User cannot already login		
Post-Condition	User is logged-In Successfully.		
Test Steps	Test data	Expected Result	Actual Result
I. Verify the registered details in Login form II. Verify the wrong registered details in Login form III. Verify User can update password by click on forget password	Id:201400137@gift.edu.pk Pass:Fahad1234	Pass User cannot sign In User can update	Pass Pass Pass

T_ID:	005	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open Sign-In Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Verify the registered details in Login form						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
Enter Id	Entered Successfully	As Expected	pass	pass			
Enter Pass	Entered Successfully	As Expected	pass	pass			
Click Sign In	Successfully Signed In	As Expected	pass	pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	006	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/09/2023		
P_NO.	Prerequisites						
1	Initiate App						
2	Open Sign In Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Verify the wrong registered details in Login form						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Enter Id	Entered Successfully	As Expected	pass			
02	Enter wrong Password	Entered Successfully	As Expected	pass			
03	Click Sign In	Not Successful Signed In / Give alert error	As Expected	pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	007	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open Sign In Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Verify User can update password by click on forget password						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click Forget Password	Entered Successfully	As Expected	pass			
02	Enter Email	Entered Successfully	As Expected	pass			
03	Get new password	Set Successfully	As Expected	pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

Camera Module

ID	TS-03		
Description	User Can Integrate the CCTV Camera Footage to App and remove it successfully.		
Pre-Condition	User cannot added camera		
Post-Condition	Camera module running successfully.		
Test Steps	Test data	Expected Result	Actual Result
I. User can Add the camera.	IP Address, Port Number, camera name	Pass	Pass
II. User can see the main camera view in dashboard screen and in all views screen.		Pass	Pass
III. Remove Camera		Pass	Pass

T_ID:	008	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open add camera Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	User can Add the camera.						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click Camera Settings	Open Settings	As Expected	Pass			
02	Click add camera	Open Add Camera details	As Expected	Pass			
03	User add camera details	Successfully Entered	As Expected	Pass			
04	User click on save button	Successfully saved	As Expected	Pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	009	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Dashboard Screen, All Views Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	User can see the main camera view in dashboard screen and in all views screen.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Click Dashboard Screen	Camera View Enabled	As Expected	Pass	
02	Click All View Screen	All Views of CCTVs enabled	As Expected	Pass	
03	Click pin button	View will be on dashboard	As Expected	Pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	010	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open All View screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Remove Camera				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Click remove camera view button	The video player will remove	As Expected	Pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

Alarm Module

ID	TS-04		
Description	Verify that User can add, trigger alarm and remove successfully.		
Pre-Condition	Alarm is not added		
Post-Condition	Alarm module is running successfully.		
Test Steps	Test data	Expected Result	Actual Result
I. Add Alarm	Alarm details	Pass	Pass
II. Trigger Alarm		Pass	Pass
III. Remove Alarm		Pass	Pass

T_ID:	011	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open Add Alarm Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Add Alarm						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click add alarm	Open form	As Expected	Pass			
02	Add Alarm details	Alarm Added	As Expected	Pass			
02	Add Alarm again (Already Added)	Show Error	As Expected	Pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	012	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open Dashboard Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Trigger Alarm						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click the button in right side of Emergency Alarm Screen	Trigger Alarm	As Expected	Pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	013	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open IOT Config Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Remove Alarm						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click the button in right side of alarm.	Delete Alarm	As Expected	Pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

Check Specious Activity Detection and alert generation

ID	TS-05		
Description	Verify the algorithm detect specious activity accurately		
Pre-Condition	Algorithm is not detecting		
Post-Condition	Algorithm detect specious activity and generate notification.		
Test Steps	Accuracy	Expected Result	Actual Result
i. Verify system can detect wall climbing successfully and give alert in notification.	65-75%	Pass	Pass
ii. Verify system can detect Gun successfully and give alert in notification.	80%	Pass	Pass
iii. Verify system can detect covered face successfully and give alert in notification.	76%	Pass	Pass
iv. Verify system cannot detect other activity and give alert in notification.	90%	Pass	Pass

T_ID:	014	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Notification Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	I. Verify system can detect wall climbing successfully and give alert in notification.				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Do wall climb in the front of cctv.		Give alert	As Expected	pass
Test Case Net Result:		Pass			
Reviewer Comments		Good			

T_ID:	015	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Notification Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify system can detect Gun successfully and give alert in notification.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Show Gun in the front of cctv.	Give Alert	As Expected	pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	016	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Notifications Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify system can detect covered face successfully and give alert in notification.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Show covered face in the front of cctv.	Give alert	As Expected	pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	017	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Notifications Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify system cannot detect other activity and give alert in notification.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Do some activity in the front of CCTV	Cannot Give Alert	As Expected	pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

SOS Alert

ID	TS-06	
Description	User can add other user as contact and can send messages to it	
Pre-Condition	Application is logged in	
Post-Condition	Message sent	
Test Steps	Expected Result	Actual Result
i. Search User and Add User	Pass	Pass
ii. Sends Alert Message	Pass	Pass
iii. Initiate message to what's app contacts	Pass	Pass
iv. Edit WhatsApp message	Pass	Pass

T_ID:	018	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Add User Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Search User and Add User				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click Search bar		Typing	As Expected	pass
02	Type Other User Contact		Searched	As Expected	pass
03	Click on add user		User Added	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	019	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Sends Message				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click on send SOS alert		Message sent	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	020	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Initiate message to what's app contacts						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click WhatsApp button	Link to WhatsApp	As Expected	pass			
02	Choose contacts	Chosen	As Expected	pass			
03	Send Message (location & Default Message)	Sent	As Expected	pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	021	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Edit WhatsApp message						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click WhatsApp button	Link to WhatsApp	As Expected	pass			
02	Choose contacts	Chosen	As Expected	pass			
03	Change and Send Message (location & Default Message)	Sent	As Expected	pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

Initiate calls to emergency services

ID	TS-07		
Description	User can initiate calls to emergency services		
Pre-Condition	User click the phone button in the center bottom of dashboard and see many options.		
Post-Condition	User can initiate call successfully.		
Test Steps	Expected Result		Actual Result
I. Initiate call to Ambulance	Pass		Pass
II. Initiate call to police	Pass		Pass
III. Initiate call to fire brigade	Pass		Pass

T_ID:	022	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Phone Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Initiate call to Ambulance				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click On Ambulance button		Link to phone and already dialed No	As Expected	pass
02	Click Phone		Initiate call	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	023	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Phone Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Initiate call to police				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click On Police button		Link to phone and already dialed No	As Expected	pass
02	Click Phone		Initiate call	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	024	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Phone Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Initiate call to fire-brigade				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click On Fire-brigade		Link to phone and already dialed No	As Expected	pass
02	Click Phone		Initiate call	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

View, update profile and delete account.

ID	TS-08		
Description	User can Log-out to Application		
Pre-Condition	User already login		
Post-Condition	User info update and account deleted successfully.		
Test Steps	Test data	Expected Result	Actual Result
I. Verify the user info is correct.	Id:201400137@gift.edu.pk Pass:Fahad1234	Pass	Pass
II. Verify the user can Update Account Info.		Pass	Pass
III. Verify user can delete account permanently.		Pass	Pass

T_ID:	025	Test By:	Hamza	Created By:	Fahad		
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024		
P_NO.	Prerequisites						
1	Initiate App						
2	Open Settings Screen						
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234				
Objective	Verify the user info is correct.						
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended			
01	Click on Accounts	Open Details	As Expected	pass			
Test Case Net Result:	Pass						
Reviewer Comments	Good						

T_ID:	026	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Account Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify the user can Update Account Info.				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click on accounts		Open details	As Expected	pass
02	Change details in form		Changed	As Expected	pass
03	Click Save/Update		Successfully saved	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	027	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/09/2023
P_NO.	Prerequisites				
1	Initiate App				
2	Open Settings Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify user can delete account permanently.				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click Delete permanently		Give alert for permission	As Expected	pass
02	Click ok		Deleted Successfully	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

Check terms and conditions, about us, privacy policy and help material

ID	TS-09		
Description	Checks the Helping material for user		
Pre-Condition	System Logged In		
Post-Condition	Information is showing successfully.		
Test Steps	Expected Result		Actual Result
I. Verify the information in terms and conditions is given and accurate.	Pass		Pass
II. Verify the information in about us is given and accurate.	Pass		Pass
III. Verify the information in privacy policy is given and accurate.	Pass		Pass
IV. Verify the information in Help material is given and accurate.	Pass		Pass

T_ID:	028	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Settings Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify the information in terms and conditions is given and accurate.				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click Terms and Condition		Open Article	As Expected	pass
Test Case Net Result:		Pass			
Reviewer Comments		Good			

T_ID:	029	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Settings Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify the information in about us is given and accurate.				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click About Us		Show Article	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	030	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Settings Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify the information in privacy policy is given and accurate.				
Step#	Step Details		Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
01	Click Privacy Policy		Open Article	As Expected	pass
Test Case Net Result:	Pass				
Reviewer Comments	Good				

T_ID:	031	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open IOT Config Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify the information in Help material is given and accurate.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Click help me	Open Web Page	As Expected	pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

User Log-Out

ID	TS-12		
Description	User can Log-out to Application		
Pre-Condition	User already login		
Post-Condition	User is Log out Successfully.		
Test Steps	Test data	Expected Result	Actual Result
I. Verify the user can logout.	Id:201400137@gift.edu.pk Pass:Fahad1234	Pass	Pass

T_ID:	032	Test By:	Hamza	Created By:	Fahad
Reviewed By:	Almas Ilyas	Initial Action:	App Install	Test Date:	14/08/2024
P_NO.	Prerequisites				
1	Initiate App				
2	Open Settings Screen				
Test Validation Data	Id:201400137@gift.edu.pk		Pass:Fahad1234		
Objective	Verify the user can logout.				
Step#	Step Details	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended	
01	Click Log-Out	Going to Sign In Screen	As Expected	pass	
Test Case Net Result:	Pass				
Reviewer Comments	Good				

14.2 Traceability Matrix

No.	Chunks	Test Scenarios	Test Cases	Expected Result	Results
01	Registering	TS-01	TC_1 TC_2 TC_3 TC_4	Pass Pass Pass pass	Pass Pass Pass Pass
02	User Log-In	TS-02	TC_5 TC_6 TC_7	Pass Pass Pass	Pass Pass Pass
03	CCTV Camera Module	TS-03	TC_8 TC_9 TC_10	Pass Pass Pass	Pass Pass Pass
04	Alarm Module	TS-04	TC_11 TC_12 TC_13	Pass Pass Pass	Pass Pass Pass
05	Check Suspicious Activity Detection	TS-05	TC_14 TC_15 TC_16 TC_17	Pass Pass Pass Pass	Pass Pass Pass Pass
06	SOS Alert	TS-06	TC_18 TC_19 TC_20 TC_21	Pass Pass Pass Pass	Pass Pass Pass Pass
07	Initiate calls to emergency services	TS-07	TC_22 TC_23 TC_24	Pass Pass Pass	Pass Pass Pass
08	View and Update Profile and deleted account	TS-08	TC_25 TC_26 TC_27	Pass Pass Pass	Pass Pass Pass
09	Check terms and conditions , about us, privacy policy	TS-09	TC_28 TC_29 TC_30	Pass Pass Pass	Pass Pass Pass
10s	User Log-Out	TS-10	TC_32	Pass	Pass

15. Exit Criteria

The exit criteria for the testing phase of the Emergency Alert and Location Sharing Application define the conditions that must be met before the testing phase can be considered complete and the application can move to the deployment stage. The following criteria must be satisfied:

No High-Priority Problems Are Open:

- o All identified high-priority issues or bugs must be resolved, and no critical defects should remain open. This ensures that the application is stable and reliable for end-users, particularly in emergency situations where functionality is crucial.

Successful Completion of All Test Cases:

- o All test cases, including functional, non-functional, security, and performance tests, must be executed successfully with results that meet or exceed the predefined acceptance criteria.

Performance Benchmarks Met:

- o The application must meet all performance benchmarks, such as response times, processing speed, and resource utilization, as defined in the test plan.

No Unresolved Medium or Low Priority Bugs that Affect Usability:

- o Any remaining medium or low-priority bugs must be documented, and their impact on the user experience must be negligible. If necessary, these can be scheduled for future updates, but they must not affect the core functionality or user safety.

User Acceptance Testing (UAT) Completed:

- o User Acceptance Testing should be completed with positive feedback from test users, indicating that the application meets their needs and expectations in real-world scenarios.

Documentation Completed:

- o All necessary documentation, including user manuals, help guides, and technical documentation, must be completed and reviewed.

16. Sign off

This document has been formally approved before System Test can commence by:

*Project Advisor: Almas Ilyas
(GIFT University Gujranwala)* _____