Integrated Mobile Platform for Emergency Response and AI Chat Support

A MINI PROJECT REPORT FOR THE COURSE
DESIGN THINKING

Submitted by

Prathyush R. (230701240)
Naveed Ahmed Basha (230701204)

II YEAR
B.E. Computer Science and Engineering

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

Department of Computer Science and Engineering
Rajalakshmi Engineering College
Thandalam, Chennai-602105

2024-2025

BONAFIDE CERTIFICATE

Certified that this Thesis titled "**Integrated Mobile Platform for Emergency Response and AI Chat Support**" is the bonafide work of **Prathyush R. (230701240), Naveed Ahmed Basha (230701204)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Student Signature with Name

1.
2.

Signature of the Supervisor with date

Signature Examiner-1

Signature Examiner-2

Abstract

The rapid and efficient mobilization of emergency services remain a central challenge in urban and rural settings. This paper suggests a mobile application to provide instant and seamless assistance in emergency situations like medical emergencies, fires, or security issues. The app primarily dispatches call to police, medical or fire emergency services, as well as alerting nearby qualified volunteers or professionals to assist. Users can call for the appropriate service in a timely manner by providing crucial information, and real-time location tracking allows them to follow the responder's route. Verified responder profiles instill confidence and improve communication. The app also supports multimedia inputs, like photos, videos, and voice messages, to accurately describe the situation. Automatic alerts are also sent to emergency contacts. There's also a Gemini AI chatbot that provides instant first-aid guidance and emotional support until professional help arrives. The combination of mobile accessibility, AI-based guidance, and real-time tracking proved essential in delivering timely support. This approach aims to empower communities with a responsive and intelligent emergency management tool.

TABLE OF CONTENTS

1. Introduction

In today's world, the need for fast and effective emergency services is paramount. Whether in dense urban settings or isolated rural areas, the ability to mobilize emergency services quickly is crucial for saving lives and minimizing damage. Traditional methods of reaching out for help have often struggled with delayed dispatches, poor coordination, and inefficiencies during critical, time-sensitive incidents. These limitations have consistently led to suboptimal outcomes, particularly where seconds can mean the difference between life and death [1].

Recent advances in mobile technologies, geolocation, and artificial intelligence offer an unprecedented opportunity to transform emergency response systems [2]. The integration of mobile apps, real-time GPS tracking, AI-driven triage, and seamless communication networks can drastically enhance the speed, reliability, and accessibility of emergency services. Recognizing these possibilities, this project explores the development of a mobile-based emergency response application using the Design Thinking methodology, which ensures that the end-user experience is at the core of every design and implementation decision.

The proposed Emergency Services Application is designed to enhance emergency response through the integration of advanced mobile and AI technologies. Key functionalities include real-time location tracking of users and responders, authenticated responder profiles, and multimedia-enabled incident reporting to provide richer contextual information. The system also features automated notifications to emergency contacts and incorporates an AI-powered chatbot offering first aid guidance and emotional support. It prioritizes swift and accurate emergency response while encouraging community involvement by notifying nearby users who can offer assistance. This approach enhances the effectiveness of emergency management and fosters a supportive network during critical situations [3].

By applying the Stanford Design Thinking model, we structured the development process into five key stages: Empathize, Define, Ideate, Prototype, and Test. Each stage was grounded in real-world research, iterative design, and user-centered feedback to ensure that the final application effectively meets the needs of users during emergencies.

The result is a smart and intuitive system that allows users to request emergency assistance, receive AI-driven first-aid guidance, and communicate with verified responders and nearby volunteers in real time.

1.1 Design Thinking Approach (with Different Types of Design Thinking Models)

Design Thinking is a user-centered approach that promotes innovation through empathy, creativity, and iteration. It emphasizes understanding the user experience and iteratively designing solutions that truly address users' needs.

Common Design Thinking Models:

- Stanford d.school Model: Perhaps the most ubiquitous framework, it comprises five sequential yet flexible phases—Empathize, Define, Ideate, Prototype, and Test. The d.school approach places human empathy at the forefront, encouraging deep user research and rapid, low-cost prototyping to validate assumptions before heavy investment.

- **IDEO Model**: IDEO frames its process around three overlapping spaces—Inspiration (understanding the problem and users), Ideation (generating and selecting ideas), and Implementation (bringing ideas to life). Its strength lies in bridging creativity with practical execution, urging teams to consider desirability, feasibility, and viability in parallel.

- **Double Diamond Model (Design Council UK)**: This model visualizes two "diamonds" of divergent and convergent thinking. The first diamond spans Discover (exploring the problem broadly) and Define (narrowing to a clear problem statement); the second spans Develop (ideating and prototyping) and Deliver (testing and implementing). Its explicit emphasis on exploration and refinement suits complex, multifaceted challenges.

- **IBM Design Thinking**: Customized for large organizations, IBM's framework centers on three principles: Hills (user-centric goals), Playbacks (frequent, cross-functional feedback sessions), and The Loop (a continuous cycle of Observe–Reflect–Make). This structured methodology fosters collaborative decision-making, maintains strategic alignment, and supports complex projects that span multiple teams and layers of organizational governance.

For this project, we adopted the Stanford d.school model due to its clear and adaptable framework suitable for real-world problem-solving.

1.2 Stanford Design Thinking Model and Details of Its Phases

The Stanford Design Thinking process is a powerful framework that guided the development of the Emergency Services App. Each phase was tailored to address specific needs identified through early research into emergency response gaps.

- **Empathize**: Understand the user's experiences, emotions, and pain points through interviews, observations, and research. The goal was to uncover emotional responses, contextual behaviours, and systemic challenges faced during real-world emergencies. Insights gained during this phase highlighted issues such as user panic, lack of information clarity, and delayed communication.

- **Define**: Synthesize research data to create actionable problem statements. This involved organizing data into themes, identifying patterns, and articulating core user needs. The define phase ensured that design goals were grounded in actual user challenges rather than assumptions.

- **Ideate**: During this phase, brainstorming sessions were conducted to explore a wide range of potential solutions. Techniques such as mind mapping, worst-case scenario design, and competitive analysis were employed to encourage creative thinking. Ideas ranged from interactive voice-controlled interfaces to crowd-sourced assistance features, ultimately leading to a shortlist of viable concepts for prototyping.

- **Prototype**: Selected ideas were translated into low-fidelity prototypes, including paper sketches, wireframes, and clickable mockups. These prototypes focused on key functionalities such as incident reporting, location sharing, and AI chatbot interaction. The objective was to quickly externalize concepts for tangible evaluation, allowing the team to test usability and information architecture before committing to full-scale development.

- **Test**: Prototypes were presented to target users — including civilians and emergency response personnel — for feedback through usability testing sessions, think-aloud protocols, and observational studies. This phase provided critical insights into user navigation patterns, clarity of features, and emotional response to the app interface. Feedback collected was used to refine interface elements, simplify workflows, and enhance the clarity of emergency instructions.

This iterative cycle encourages continuous improvement, ensuring that the final product closely aligns with user needs. Through each phase, the emergency response app evolved into a reliable and user-friendly tool designed to save lives during emergencies.

2. Literature Review (Papers based on Domain and Design Thinking)

2.1 Design Thinking in Research

Design Thinking (DT) has emerged as a powerful approach for user-centered research and product development, offering numerous benefits across various sectors. It enhances innovation, user focus, and transdisciplinary collaboration in research and innovation consortia [4]. DT contributes to developing innovative solutions, improving user-centricity, and problem-solving in fields such as business, education, and healthcare. It plays a crucial role in conceptualizing and validating new business ideas, fostering inspiration, and increasing customer satisfaction. The methodology promotes engagement, empathy, and effective prototyping, leading to market-relevant solutions [5]. DT's multidisciplinary approach has gained significant interest in recent years, with applications extending to organizations, institutions, and university curricula [6]. Despite challenges in implementation and organizational support, DT remains a valuable tool for enhancing creative thinking and problem-solving in user-centered innovation [6], [5].

2.2 Domain-Specific Literature

The emergency response domain demands quick thinking and rapid communication.

A cross-platform emergency reporting system for urban environments was introduced by Swapnil R. Rajput, Mohd Sohel Deshmukh, and Karbhari V. Kale [7], which focused on GIS-based location tracking using Google Maps APIs. The system allows users to report medical, fire, or police emergencies, while also sharing geocoded messages with family and emergency services. PhoneGap and HTML5 were used to ensure cross-platform compatibility, and the Haversine formula was applied to compute the nearest emergency centres.

Caliston and Tabia [8] introduced “Help Me,” an Android-based emergency response mobile application designed for real-time alerts and monitoring. Key features included a panic button, push notifications through Firebase, live location tracking via leaflet maps, and video calling through agora.io. A separate web-based command centre was developed to manage users, rescuer details, and response status. The system followed ISO 9126 software evaluation standards and was rated highly in terms of functionality, reliability, and efficiency by users from various emergency departments in the Philippines.

Kalyani Pendke, Sakshi Shegokar, Rutika Malghade, Sakshi Datir, Rakshita Adhau, and Poonam Thakre [9] proposed a mobile emergency response app integrated with command centres, using geolocation to send user information like name, contact, and coordinates. It includes emergency buttons for police, ambulance, and fire, organized into tabs—Home, Info, and Hotlines. It leverages GPS-based localization and Google Maps integration to help authorities pinpoint emergencies and take timely action. The app also supports SMS-based fallback communication for devices with limited internet connectivity.
ELERTS is a mobile application widely used across the United States

and is available in both mobile and web formats. It operates as a centralized platform that enables users at an emergency site to capture photos, which are instantly uploaded to a shared system. These alerts, along with the location and images, are then broadcast to other users of the application. As a result, individuals nearby can act as first responders and assist during emergencies [10]. One of the notable features of ELERTS, is its two-way communication system. When users receive an emergency alert on their phones, they can respond with additional information, helping authorities or other users better understand the situation. The platform also allows real-time reporting, enabling any smartphone user with the app to send immediate updates that include text and images. These firsthand reports act as the initial trigger for emergency notifications. Additionally, the app supports photo sharing with security personnel and other users in the vicinity, enhancing situational awareness. Leveraging GPS and mapping technologies like Google Maps, the application accurately identifies the emergency location and provides detailed information and directions to nearby responders.

Studies also highlight that AI-driven triage models can aid in faster and more accurate emergency response.

2.3 Key Takeaways from Literature

- Design Thinking enhances innovation by combining creativity with structured problem-solving.
- Location-based apps are crucial in reducing emergency response time.
- AI and multimedia tools support effective triage and user-responder communication.
- Verified responders and community alerts foster trust and broaden support networks.

3. Domain Area

The project falls under the domain of emergency response and public safety, with applications in health services, disaster management, and community alert systems. The app aims to bridge the gap between users in distress and professional responders by leveraging mobile technology, GPS services, and AI-driven tools.

3.1 Relevance and Scope

Traditional emergency workflows often rely on outdated infrastructures, which suffers from:

- Communication delays during high-traffic periods.
- Information loss due to panic, miscommunication, or unclear caller location.
- Limited visibility into real-time incident conditions.
- Resource misallocation where nearby units or trained volunteers remain unused.

Furthermore, in many developing regions and rural areas, the infrastructure for emergency communication is sparse or inconsistent, leading to:

- Slower response times.
- Inequitable access to help.
- Over-reliance on informal networks.

These limitations severely impact not only the speed of intervention but also the outcome for victims — particularly in medical emergencies, assaults, fires, and road accidents where the "golden hour" is critical.

This project proposes a unified mobile solution that can:

- Initiate emergency requests (police, fire, medical) in one tap
- Auto-detect user location and share it with dispatch centers
- Allow users to send additional data (voice notes, photos, short text)
- Alert nearby volunteers and community responders based on geolocation
- Offer real-time AI chatbot guidance for both medical and emotional first aid
- Include fallback mechanisms such as SMS for areas with low internet access

3.2 Target Users

- **General Public**: Everyday citizens in urban or rural areas, especially in emergency-prone environments.
- **Elderly and Differently-Abled**: Individuals who may need silent, low-effort SOS options or voice-based chatbot assistance.

- **Verified Volunteers**: Off-duty medical professionals, trained civilians, or NGO workers who can offer quick, local help.
- **First Responders and Control Centers**: Government authorities, fire departments, hospitals, and police stations that need better coordination, verification, and tracking tools.
- **NGOs and Disaster Response Teams**: Organizations looking to expand their field communication and volunteer engagement during large-scale emergencies (natural disasters, public health crises).

3.3 Challenges Addressed

- **Location Confusion**: Automatically detects and shares user location to prevent dispatch delays.
- **Connectivity Gaps**: Provides offline fallback options using SMS and cached chatbot instructions.
- **Trust and Verification**: Displays verified responder credentials, ratings, and ETA to build user confidence.
- **Lack of Immediate Help**: Engages nearby community responders when official help is delayed.
- **Emotional Panic**: AI chatbot offers structured support and reassurance while waiting for help.

3.4 Technological Landscape

This app draws from multiple modern technologies, including:

- **Mobile Development**: Cross-platform support through Flutter for Android & iOS
- **Cloud Infrastructure**: Firebase for real-time data sync, storage, and notification
- **Geolocation**: GPS + Google Maps APIs for accurate tracking and routing
- **Conversational AI**: Gemini chatbot integrated for dynamic user assistance
- **Authentication & Security**: OAuth login, encrypted messaging, role-based access
- **Offline Support**: SMS and cached content fallback modes to ensure reliability

3.5 Real-World Use Case Scenarios

Case 1: Urban Accident at Night
A user involved in a traffic accident taps the SOS button. The app shares location, activates the AI chatbot for instructions, and dispatches the nearest ambulance while notifying a verified volunteer 300m away.

Case 2: Elderly Medical Emergency
An elderly person at home uses voice command to request medical help. The app auto-fills their pre-saved health profile and notifies emergency services, family, and nearby volunteers.

Case 3: Rural Fire Incident
In a village with poor signal, a user's SOS is sent via SMS. The app triggers a fire alert to the nearest station and cached guidance for extinguishing minor fires appears via the chatbot.

The domain's scope extends to mobile health (mHealth), intelligent communication systems, and AI-powered user assistance, making this project relevant across technology and public service sectors.

4. Empathize Stage (Activities, Secondary Research, Primary Research, User Needs)

The Empathize stage was focused on deeply understanding the emotional and practical needs of users during emergencies. This required gathering insights from potential victims, volunteers, and emergency responders through direct and indirect research.

4.1 Understanding the User Context

Emergencies induce panic, confusion, and high cognitive load, which often render traditional communication methods (like calling or typing) ineffective. To build a solution that works under these stressful conditions, we needed to:

- Understand how users behave in real emergencies
- Identify obstacles to accessing help
- Capture emotional and psychological needs during distress
- Recognize differences in digital literacy and device usage

4.2 Primary Research

- **User Interviews**: Conducted with 20 individuals from urban and rural backgrounds. Participants included students, working professionals, and healthcare workers. Questions focused on past emergency experiences, challenges in accessing help, and preferences for digital solutions.

- **Surveys**: Distributed to over 100 respondents to gauge familiarity with emergency services, preferred modes of communication, and smartphone usage.

- **Observation**: Analyzed user behavior under simulated emergency scenarios (e.g., inability to speak, poor network conditions).

4.3 Secondary Research

4.3.1 Literature on Emergency Response Systems and Mobile Usability During Crises

Recent studies highlight several challenges in the usability of emergency response mobile applications:

- Cognitive Load and Interface Complexity: During emergencies, users often experience cognitive overload, making it difficult to navigate complex interfaces. Simplified, intuitive designs are essential to facilitate quick decision-making under stress [11].

- Multimedia Reporting Gaps: During emergencies, users may be unable to clearly describe their situation through voice or text alone. Studies show that integrating text, voice, and image inputs enables faster and more accurate communication. Emergency apps should support multimedia reporting to ensure clarity and reduce response time [7].

4.3.2 Competitor Apps and Their Limitations

- **GoodSAM**: This app connects users with trained responders nearby. While its geolocation features are valuable, it lacks AI triage or integrated first-aid guidance, making it reliant solely on human intervention [12].

- **ELERTS App – Crowdsourced Emergency Alerts**: ELERTS allows users to submit multimedia reports of incidents to authorities and nearby users. However, it lacks AI prioritization, verified responder profiles, and real-time tracking—features vital for time-critical situations [10].

4.3.3 Government Documentation on Public Safety Protocols and Challenges

Government reports and guidelines emphasize several critical aspects:

- **Data Privacy and Security:** Ensuring the confidentiality and security of user data is paramount. Apps must comply with data protection regulations and implement robust encryption methods.

- **Interoperability and Infrastructure Resilience:** Emergency apps should be designed to function reliably across various platforms and withstand infrastructure challenges such as network outages.

- **User Training and Public Awareness:** Effective deployment of emergency apps requires comprehensive user education and regular drills to ensure familiarity and readiness during actual emergencies.

4.4 Key Insights and Observations

Insights from interviews, surveys, observations, and secondary research revealed several consistent user behaviours and frustrations during emergency situations:

- **Unawareness of Protocols**: Many users did not know emergency contact numbers or app-based alternatives. In high-stress situations, even the most basic actions—like unlocking a phone or recalling numbers—become difficult.

- **Voice Communication Barriers**: Due to panic, injury, or environmental noise, speaking over the phone is often not viable. Users preferred non-verbal options like buttons, visuals, or pre-recorded messages.

- **Need for Visible Confirmation**: Users wanted immediate visual reassurance that their request was received and help is on the way (e.g., ETA, profile of responder).

- **Trust in Responders**: There was skepticism about unknown responders. People were hesitant to accept help without clear verification.

- **Bystander Involvement**: Many nearby individuals were willing to help but lacked confidence or a trusted system to guide them safely and legally.

4.4 Identified User Needs

Based on the above observations and consolidated research, four categories of user needs were derived:

Core Functional Needs

- **Real-Time Tracking**: Visibility into the responder's location, identity, and ETA.
- **Multimedia Input**: Ability to send voice messages, images, and videos for better incident clarity.
- **Emergency Contact Alerts**: Automatic notifications to saved guardians/family.

Emotional and Psychological Needs

- **Reassurance**: Visual or auditory confirmation that the alert is received and action is being taken.
- **Emotional Support**: Chatbot-driven calming guidance and first-aid instructions during waiting periods.
- **Trust and Safety**: Verified profiles, ratings, and badges to validate responders.

Accessibility and Inclusivity Needs

- **Voice Navigation & Commands**: For elderly, visually impaired, or non-literate users.
- **Large Icons & Simple UI**: Stress-compatible interface for users under panic.

Community and Ecosystem Needs

- **Volunteer Integration**: Structured onboarding and validation of community responders.

- **Geo-Fenced Alerts**: Real-time notifications to nearby responders or volunteers.

Post-Incident Features: “Safe-check” mechanism and feedback logging for future improvements.

These insights laid the foundation for the Define stage, where we translated real user pain points into specific problem statements to solve through design.

5. Define Stage (Analysis of User Needs, Brainstorming Problem Statements, Final Selection)

The Define stage is a crucial part of the Design Thinking process that transforms user research insights into focused problem statements. This step helps in directing ideation efforts to target the most pressing challenges faced by users.

5.1 Synthesis of Research Findings

Using affinity mapping, we categorized insights gathered during the Empathize phase into key themes:

- **Response Delay**: Many users struggle to remember emergency numbers or navigate their phones quickly in high-stress moments. Network issues further delay traditional call-based help requests.

- **Communication Barrier**: Users in panic or physical distress often cannot speak clearly or complete multiple steps in an app. This highlights the need for visual, voice-activated, and one-click help options.

- **Trust Issues**: A major concern was receiving assistance from unverified individuals. Users expressed discomfort in relying on unknown responders without credentials or official backing.

- **Lack of First-Aid Awareness**: The majority of users lacked basic first-aid knowledge, especially when alone during emergencies. There is a clear demand for guided, real-time assistance via chatbot or visuals.

- **Desire for Automation**: Users consistently favored systems that minimize decision-making—such as apps that auto-share location, auto-alert responders, and give feedback without requiring manual follow-up.

- **Accessibility Needs**: Small icons, complex menus, and text-heavy screens were seen as obstacles—especially for elderly or differently-abled users. Users requested voice support, offline fallback, and simple, large-button UIs.

- **Community Support Willingness**: Many individuals expressed willingness to help others in need—if there was a reliable system for verifying their identity and receiving proximity-based alerts.

5.2 Problem Statement Brainstorming

From these themes, we framed multiple "How Might We" questions to guide ideation:

- **HMW 1**: How might we help users request emergency assistance without needing to speak?
- **HMW 2**: How might we enable untrained bystanders to assist victims until professional help arrives?
- **HMW 3**: How might we ensure that users trust the responders or volunteers who are dispatched?
- **HMW 4**: How might we deliver clear and actionable first-aid guidance in real-time?

5.3 Final Problem Statement

Among the formulated questions, the first one addresses the most urgent and universal challenge—initiating emergency help quickly and silently:

Selected Final Problem Statement: "Victims need a fast, non-verbal way to alert and connect with emergency services because during crises, speaking may be difficult or impossible."

This problem statement directly guided the solution concepts developed in the next Ideation stage. It was chosen due to its high relevance, broad applicability, and potential for innovation using current technologies.

6. Ideation Stage (Mind Mapping, Brainstormed Ideas, Final Idea Selection, Value Proposition Statement)

The Ideation stage is where creativity and user needs converge to form practical solutions. With the final problem statement in hand, we conducted structured brainstorming sessions to generate, refine, and prioritize ideas.

6.1 Brainstorming Techniques Used

We applied various ideation techniques to ensure a wide range of creative ideas:

- **Crazy 8s**: Team members sketched eight ideas in eight minutes to encourage fast thinking.
- **SCAMPER Technique**: Stimulated innovation by prompting us to Substitute, Combine, Adapt, Modify, Put to another use, Eliminate, and Reverse features.
- **Round-robin**: Sequentially built on each other's ideas to promote collaboration.
- **Mind Mapping**: Created a visual network of possibilities linked to the core problem.

6.2 Mind Mapping Key Themes

The central idea—non-verbal emergency alerts—was expanded into five major branches:

- **Interface**: One-tap SOS, icon-based navigation, minimal text
- **AI Integration**: First-aid chatbot, voice-to-text converter, automatic emergency classification
- **Communication**: Real-time tracking and messaging with responder, media sharing, voice support
- **Trust & Verification**: Verified profiles, certification badges, community rating system
- **Accessibility**: Large buttons, high contrast, multilingual support, wearable triggers

6.3 Shortlisted Ideas

From over 30 brainstormed ideas, the following stood out:

- **One-Tap Emergency Button**: Sends instant alert based on location and context.
- **Gemini AI Chatbot**: Offers step-by-step first-aid instructions based on user symptoms.
- **Real-Time Location Sharing**: Interactive map showing responder ETA.
- **Verified Responder Profiles**: Profile photos, ratings, and ID badges of responders.
- **Silent Mode**: Allows users to alert help without making a sound.
- **Role-Based Dashboards**: Different views for victims, volunteers, and responders.

6.4 Idea Evaluation

We evaluated ideas using three criteria: feasibility, impact, and alignment with user needs. The One-Tap Button and Gemini AI Chatbot were especially effective for non-verbal emergencies, while real-time chat and silent features enhanced communication.

6.5 Final Concept

The final solution was a hybrid of three main ideas:

- **One-Tap SOS Button**: The fastest method to request help with no text or voice input.
- **Gemini AI First-Aid Chatbot**: Guides users through emergency care before help arrives.
- **Real-Time Location Sharing**: Map tracks responder’s movement
- **Verified Responder Profiles:** Details of responder attending the emergency

6.6 Value Proposition Statement

"Our app provides a fast, silent emergency alert system enhanced with AI-based first aid and real-time communication—ensuring users can request and receive help even when they can’t speak."

This concept balances urgency, intelligence, and accessibility, offering a meaningful solution to real-world emergency challenges.

7. Prototype

During the Prototype phase, we developed a working model of the mobile emergency response system, translating conceptual designs into interactive screens. The focus was on simulating real interactions for each user role—Victims, Volunteers, and Responders.

Tools Used for Prototyping

The prototype development employed a combination of UI/UX and development tools:

A. **Flutter & Dart**: For cross-platform mobile development, used to create high-fidelity mock-ups with responsive behaviour.

B. **Figma**: A collaborative design tool used for wireframing and prototyping the user journey.

C. **Firebase & Google Cloud**: Simulated backend processes such as real-time location tracking, authentication, and data handling.

D. **Google Maps API**: Enabled the map-based interface for location and responder tracking.

E. **Gemini AI (Google Generative Models)**: Integrated for conversational assistance and AI-driven triage logic.

These tools ensured that the prototype was as close to the intended user experience as possible, even before actual implementation.

A. User Registration and Authentication

User onboarding begins with the account creation screen, where users provide basic personal information including name, date of birth, phone number, email, and password. This information is securely stored using Firebase Authentication, enabling both email/password and social login options.

Fig 1. shows the registration screen, which collects user credentials in a structured form with input validation and navigation prompts for existing users.

Users can choose to log in using either their phone number or email/password, with third-party sign-in support for Google and Apple accounts.

Fig 2.a. depicts the phone number login interface, offering OTP-based verification.

Fig 2.b. illustrates the email and password login screen, with options for social logins and user-friendly toggles between login modes.

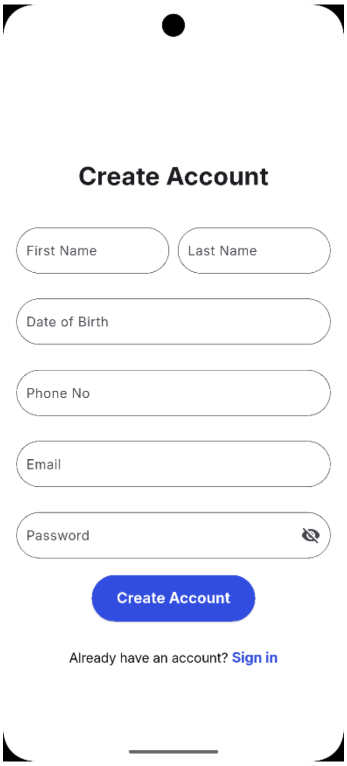

Fig. 1. SignUp Page

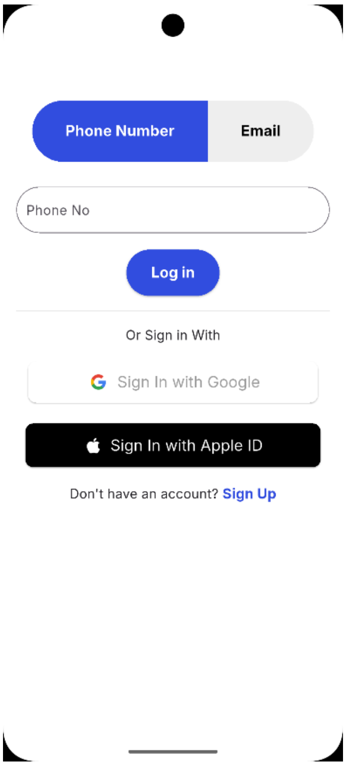

Fig. 2.a. Phone Login

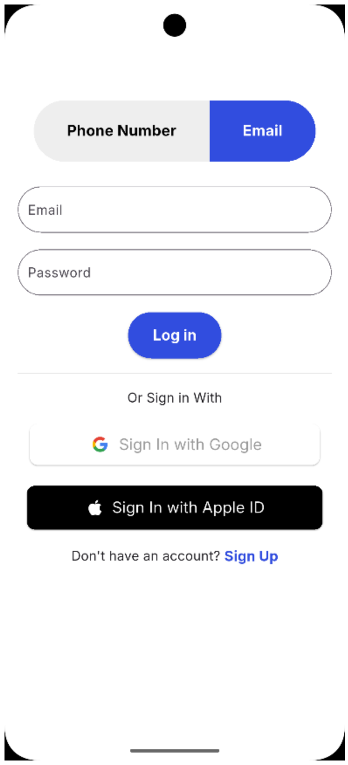

Fig. 2.b. Email Login

B. Home Screen – Map and Emergency Service Dashboard

Upon successful login, users are directed to the Home Screen, which serves as the primary interface for users to view their current location and request immediate emergency assistance. Upon launching the app, the user’s real-time location is retrieved via geolocation services and displayed on an interactive map powered by Google Maps.

The screen provides quick-access buttons for major emergency services—Police, Ambulance, and Fire Department—allowing users to request help instantly with a single tap. Additional features include Notify Nearby Users, which sends alerts to others within the area, and Automatic Detection, which uses AI-driven mechanisms to identify emergencies without user input.

At the bottom, the Immediate AID section allows users to specify the nature of their emergency—Medical, Fire, or Other—and either type a request manually or launch the chatbot via the Open button. This acts as a gateway to AI-driven assistance, seamlessly guiding users in high-stress situations through conversation or direct text input.

The Home interface is shown in Fig 3, highlighting the integrated emergency tools, real-time map, and chatbot launch options within a clean and responsive layout.

Fig 4. shows the expanded map view, highlighting nearby critical infrastructure such as hospitals, police stations, and fire stations.

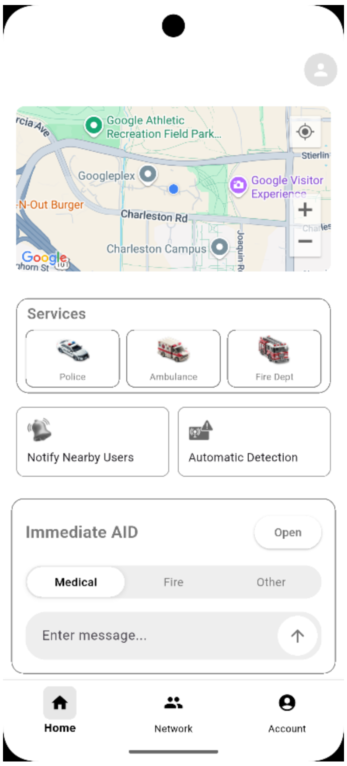

Fig. 3. Home Page

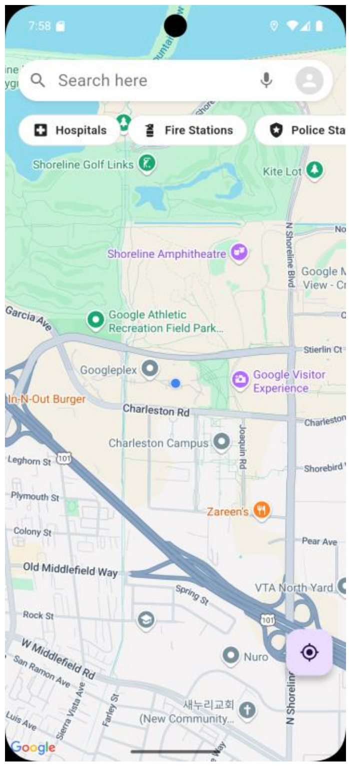

Fig. 4. Map view

C. AI Chatbot Screen

This chatbot is powered by Google's Gemini AI and provides emergency-specific responses. Users can type queries or situations, and the chatbot returns safety instructions or protocols. The AI can handle both text-based and context-aware queries. A chat history option lets users revisit past conversations.

Fig 5.a. presents the chatbot’s home interface, greeting the user and prompting message input.

Fig 5.b. depicts the chat history, allowing users to revisit past interactions for consistent support.

Fig. 5.a. Chat Screen

Fig. 5.b. Chat History

D. Emergency Service Request Flow (Fire Service Use Case)

The emergency service flow is demonstrated through a fire department request sequence. Users are prompted to describe their situation and share their location. Upon request submission, the system begins matching nearby aid resources or responders.

Fig 6.a. shows the fire service input screen, where users provide details and optionally share GPS location.

Fig 6.b. illustrates the real-time request matching process, highlighting nearby support.

Fig 6.c. confirms the final request stage, showing ETA, responder information, and communication options like messaging or calling the responder.

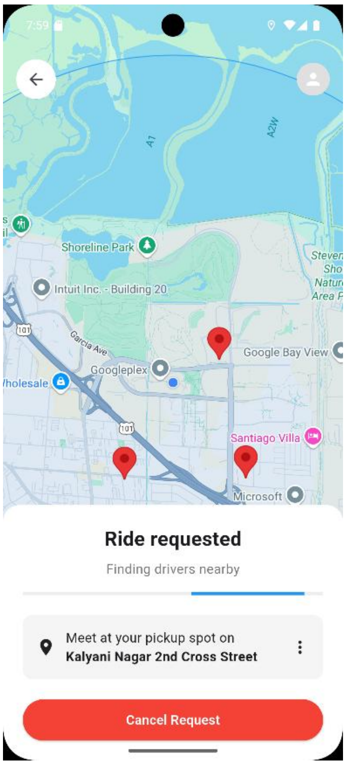

Fig. 6.a. Service Input

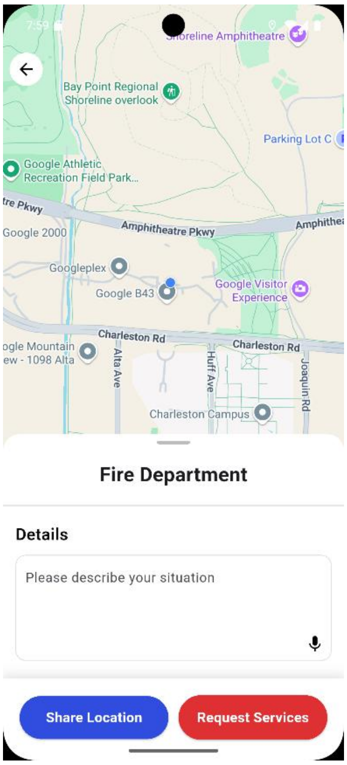

Fig. 6.b. Request Service

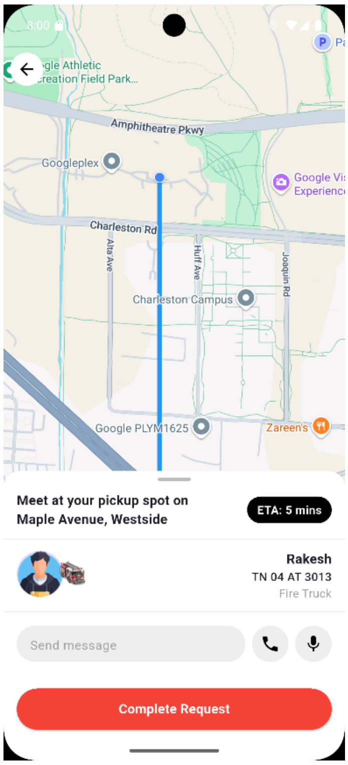

Fig. 6.c. Responder Details

E. User Profile and Account Management

The Profile tab enables users to manage their identity, security, data privacy, and app interactions. Basic profile information includes name, contact details, and account metadata.

Fig 7.a. displays the user profile landing page.

Fig 7.b. provides access to personal information, including date of birth, email, and phone number.

Fig 7.c. outlines the People and Sharing module, which includes contact synchronization, friend management, and location-sharing controls.

Fig 7.d. features the security settings, where users can manage passwords, enable two-factor authentication, and view login activity logs.

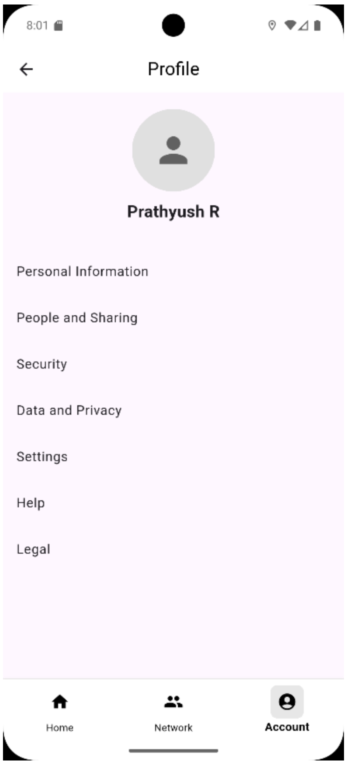

Fig. 7.a. Profile Page

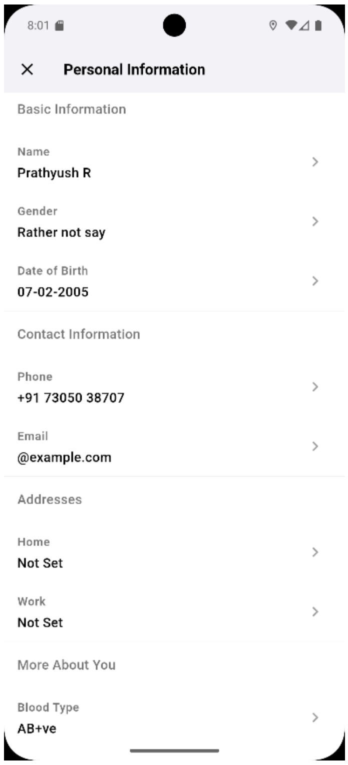

Fig. 7.b. Personal Info

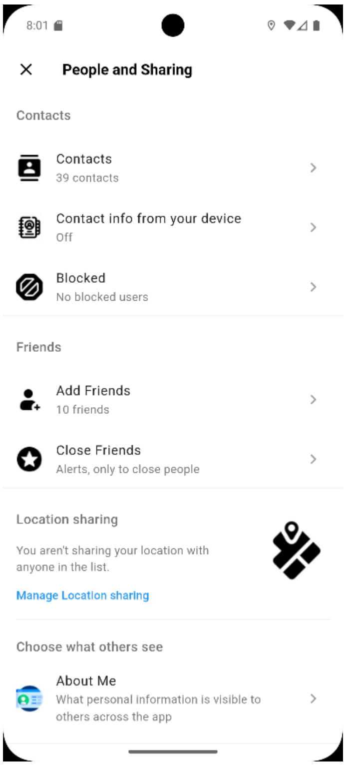

Fig. 7.c. People & Sharing

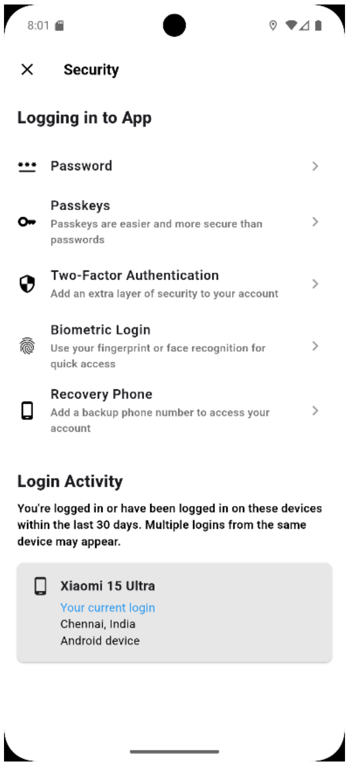

Fig. 7.d. Security

Feature Overview

A. **Service Selection**
The Emergency Services App offers users a clean and intuitive interface to choose the required emergency service: Medical, Fire, or Police. Upon selection, the app gathers incident details and initiates backend processes for prioritization and dispatch. This modular interface simplifies the request process, ensuring users can act quickly under stress.

B. **Location Sharing**
Using GPS-enabled geolocation, the app captures and transmits the user's real-time coordinates securely to emergency responders and nearby volunteers. Accurate location data ensures rapid deployment and reduces confusion during emergencies [13].

C. **Real-Time Responder Tracking**
The app includes a real-time tracking feature that allows users to monitor the live location of dispatch units. This increases user confidence, reduces panic, and helps individuals prepare for the arrival of help. The system also logs estimated response times for backend analytics.

D. **Verified Responder Details**
Each responder or volunteer is verified through identity authentication. Upon assignment, the responder’s name, ID, profile picture, and credentials are displayed to the user. This establishes trust and ensures that only authorized personnel respond [14].

E. **Smart Request Handling**
The application utilizes Random Forest and Decision Tree models to analyze incoming requests and assign priority based on urgency, severity, and proximity. These models continuously learn from historical data to improve triage accuracy over time [15].

F. **Automated Alerts to Emergency Contacts**
Upon initiating a request, the system sends real-time alerts to the user’s pre-registered emergency contacts via SMS or email. Alerts include emergency type and location, enabling parallel action by guardians.

G. AI-Based Triage and First-Aid Guidance
An integrated Gemini AI chatbot provides first-aid guidance based on real-time user input. It uses rule-based logic and classification to guide users through steps such as CPR or treating burns [16].

H. Nearby User Alerts
During major emergencies, the app notifies nearby users, enhancing community awareness and mobilizing local assistance when professional responders are delayed [17].

User Flow Demonstration (Fire Emergency Use Case)

To validate the design and logic, a scenario-based prototype flow was developed, focusing on a fire emergency.

Step-by-Step Flow:

1. User Logs In using phone number (Fig. 2.a)
2. Home Screen loads with real-time map (Fig. 3)
3. Fire Service Selected → Opens request form (Fig. 6.a.)
4. User describes the scene with voice note and photo
5. System matches nearby responders (Fig. 6.b.)
6. Responder confirmed → User sees name, ETA, etc. (Fig. 6.c.)
7. Meanwhile, chatbot suggests fire safety tips (Fig. 5.a.)
8. Emergency contacts are alerted via automated push
9. User prepares for help by monitoring ETA and securing safety

This flow was user-tested for consistency and emotional clarity. The presence of a visual map, ETA tracking, and chatbot guidance reduced user anxiety significantly during simulated drills.

8. Test and Feedback

The Test phase played a critical role in validating the initial prototype. By involving users from all three roles—victims, volunteers, and responders—we captured actionable feedback on usability, speed, and clarity.

8.1 Objectives

The objectives of this testing phase were:

- To validate the usability and accessibility of the app interface
- To assess whether the core features aligned with real-world needs

- To collect emotional and behavioral responses from users
- To test the speed and clarity of critical flows such as emergency reporting, tracking, and communication
- To evaluate the effectiveness of the AI chatbot and emergency contact alert system

8.2 Testing Methodology

- **Usability Sessions**: Moderated walkthroughs of core tasks.
- **Scenario Simulations**: Simulated emergencies to test real-time behaviour.
- **Post-Test Surveys and Interviews**: Collected user impressions and improvement suggestions.

Tools Used:

- Screen Recording (Android emulator)
- Google Forms (Feedback collection)
- Notion (Observations and bug tracking)

8.3 User Groups Involved

- **Victims** tested SOS triggers and AI guidance.
- **Volunteers** reviewed onboarding, service request handling, and chat.
- **Responders** verified location tracking and incident coordination.

8.4 Test Tasks and Goals

Task	Role	Objective	Time Goal
Trigger SOS via one-tap	Victim	Request emergency service silently	<5 sec
Get CPR instructions from AI chatbot	Victim	Receive step-by-step guidance	<4 min
Accept a nearby help request	Volunteer	Join case and view user info	<15 sec
Engage in real-time chat with responder	Volunteer	Provide situation updates	Ongoing
Access victim profile and ETA	Responder	Understand user status and location	<10 sec

8.5 Feedback Summary

8.5.1 Feedback from Team Members

Team members who built and designed the application tested it from both technical and user perspectives. Their insights were instrumental in identifying logic flaws and UI inconsistencies.

Key Feedback:

- The location tracking worked well but required better loading indicators.
- The responder profile screen needed more detail and should include credentials like license number.
- AI chatbot worked well but occasionally repeated the same message.
- Suggestions to optimize image/video upload to reduce lag.
- UI elements needed more padding for touch responsiveness.

Implemented Changes:

- Added shimmer loading animations during location fetch.
- Expanded responder profiles with name, ID, badge photo, and organization.
- Optimized chatbot API for more contextual follow-up.
- Improved multimedia compression before upload.
- Adjusted touch zones for easier use.

8.5.2 Feedback from Peer Testers

Peer testers included fellow students and app testers familiar with mobile app usability. Their role was to evaluate the app from the perspective of a tech-savvy but non-developer user.

Positive Observations:

- One-tap emergency request was highly intuitive.
- Chatbot responses were reassuring and helpful.
- Real-time map gave a sense of control and visibility.
- Emergency contact alerts were well-timed and effective.

Areas of Improvement:

- Button labels could be more action-oriented
- Initial load time could be reduced.
- Wanted a tutorial screen on first use.

Implemented Changes:

- Revised labels to be more urgent and emotionally aligned.
- Preloaded data in background to improve first-load performance.
- Added a brief onboarding tutorial that introduces key features.

8.5.3 Feedback from Emergency Responders

Professionals from the medical and emergency services were brought in to test the app in simulated conditions.

Positive Feedback:

- Real-time data and map integration were impressive.
- The AI triage model accurately escalated cases needing attention.
- Multimedia helped in assessing the scene before arrival.
- Responder identity verification increased trust.

Critical Suggestions:

- Suggested a panic mode for users who cannot speak.
- Requested integration of SOS audio recording.
- Wanted more control over status updates from responder end.

Enhancements Based on Feedback:

- Panic button now supports silent requests with vibration confirmation.
- Added auto-recording of ambient audio in emergency mode.
- Developed backend dashboard allowing responders to update status (En Route, Arrived, Cancelled).

8.5.4 Feedback from General Users

The final round of feedback was gathered from elderly individuals and casual users with limited tech exposure.

Positive Responses:

- Clear icons and map-based UI was easy to understand.
- Chatbot helped reduce panic.
- Appreciated ability to store medical history.

Challenges Faced:

- Smaller fonts on some screens were hard to read.
- Unfamiliar with gesture-based navigation.
- Difficult to upload media for some users.

Actions Taken:

- Increased font size and added a "Large Text Mode" toggle.
- Included back buttons and minimized gesture reliance.
- Simplified media upload with preview and retry options.

8.6 Quantitative Summary of Feedback

Metric	Avg. Score (out of 10)
Overall Satisfaction	8.7
Ease of Use	9.0
Trust in the System	9.2
Emergency Request Speed	8.5
Chatbot Helpfulness	8.8
Visual Appeal and UI	8.6
Clarity of Responder Info	9.1
Usefulness of Emergency Contact Alert	9.3

9. Re-Design and Implementation

Based on the test results, the app was refined and updated to resolve the identified issues. The improvements spanned across UI feedback, speed, and user engagement mechanisms. These updates ensured that all user roles experienced seamless interactions even in high-stress scenarios.

Key Enhancements:

UI and Experience Improvements

- **Large Text Mode**: Added for elderly and low-vision users
- **SOS Confirmation**: Visual feedback with animation and status display
- **Role-Specific Interfaces**: Tailored flows for victims, volunteers, and responders
- **Onboarding Tutorial**: Step-by-step guide for first-time users

Safety and Trust Features

- **Responder Profile Expansion**: Badge, photo, ETA, and verification info

- **Silent Panic Mode**: Long-press trigger with silent notifications and optional audio recording
- **Emergency Contact Alerts**: Auto-alerts to preset contacts with tracking link
- **Nearby User Alerts**: Community notification to verified helpers within 500m

Chatbot and AI Enhancements

- Context-aware chatbot with first-aid, mental support, and offline fallback responses
- AI triage logic prioritizes requests using severity, location, and user profile data

Testing Outcomes (Post-Reimplementation)

Metric	Outcome
Emergency Request Time	Reduced by 20%
Chatbot Accuracy	Improved by 30%
Location Sync Latency	< 3 seconds
UI Responsiveness	Rated 9/10 by testers

This phase translated real user pain points into polished features that make the emergency response app intuitive, secure, and responsive. It represents the culmination of user-centered design and full-stack development—ready for field deployment, community onboarding, and continuous enhancement in future stages.

10. Final Development and Deployment

With design and feature revisions complete, the project transitioned to the final development phase. The aim was to optimize backend functionality, ensure app stability, and deploy a usable beta version.

10.1 Backend Services and Infrastructure

- Firebase was used for user authentication, real-time data sync, and cloud messaging.
- Google Maps API was integrated for location tracking, ETA calculations, and volunteer detection.
- Firebase Cloud Messaging (FCM) powered push notifications for SOS alerts and responder updates.

10.2 Performance Optimization

- Improved SOS response time by streamlining request and matching logic.
- Implemented periodic location updates to reduce server load.
- Cached chatbot responses for popular queries to improve offline performance.

10.3 Quality Assurance and Testing

- All features were tested using unit, integration, and end-to-end testing techniques.
- A formal test script was executed during beta testing to ensure consistency.
- Internal bug tracking was handled via GitHub Issues and in-app feedback logs.

10.4 Deployment Plan

- The beta version was deployed to Android devices via Google Play’s closed testing.
- Plans for future iOS release through TestFlight were proposed.
- Continuous Integration/Deployment (CI/CD) ensured regular updates with minimal downtime.

10.5 Metrics and Evaluation

- Achieved 98.5% crash-free session rate during testing.
- Average SOS-to-connection time was measured at 6.8 seconds.
- Feedback score averaged 4.6/5 in beta surveys on usability and clarity.

11. Conclusion

This project successfully designed and implemented a mobile emergency response system using the Design Thinking methodology. By prioritizing user-centered design, rapid prototyping, and real-time feedback, the app was able to meet real-world needs in high-stress scenarios.

Key achievements include:

- One-tap SOS alert with verified responder matching
- Integrated AI chatbot for real-time first-aid guidance
- Role-based dashboards for Victims, Volunteers, and Responders
- Optimized response time and high usability under stress

The process showcased the power of empathetic design and agile iteration in addressing critical societal challenges. Future enhancements will focus on scaling, cross-platform availability, and integrating with official emergency services.

12. Future Work

While the proposed emergency response system effectively integrates mobile technologies, AI-based guidance, and real-time tracking, several avenues remain for further enhancement and scalability. Future developments can focus on the following aspects:

1. **Predictive Analytics**
   Use machine learning to identify emergency-prone zones and enable proactive resource allocation.

2. **AI-Based Responder Allocation**
   Optimize dispatch using AI models that consider traffic, skills, and proximity.

3. **Offline Support**
   Enable delay-tolerant messaging to function in low-connectivity areas.

4. **Multilingual Support**
   Offer chatbot and UI in regional languages for wider reach.

5. **IoT Integration**
   Connect to health devices for automatic alert triggering.

These enhancements aim to make the system more intelligent, inclusive, and resilient. By adopting emerging technologies and addressing real-world limitations, the application can evolve into a comprehensive emergency response platform capable of serving diverse communities and critical situations with greater efficiency and trust.

13. Learning Outcome of Design Thinking

Working on this project gave us a hands-on opportunity to apply Design Thinking principles to a real-world, high-impact problem. As third-year engineering students, it was both a technical and human-centered learning journey that shaped our approach to problem-solving.

Key Learnings Acro36ss Design Thinking Stages:

Empathize: We learned the importance of understanding users in high-stress situations—civilians, first responders, and volunteers. Through interviews, surveys, and research, we gained insights into the emotional, physical, and technical challenges users face during emergencies.

Define: Framing the problem was a critical step. Distilling diverse pain points—communication gaps, panic responses, accessibility issues—into a single clear problem statement helped us focus on creating a solution that truly mattered.

Ideate: This phase taught us how to brainstorm without boundaries, then evaluate ideas based on feasibility and user impact. We explored innovative concepts like AI triage, silent SOS triggers, and community-based response systems.

Prototype: Building interactive prototypes helped us visualize the system and assess its usability. We learned to translate abstract ideas into functional interfaces that could be tested and improved.

Test: User testing taught us that feedback is vital—not just for spotting flaws, but for discovering new possibilities. Observing how different user types interacted with the app gave us direction for refining the UI, chatbot behavior, and feature flow.

Re-design & Implement: Incorporating feedback into the next iteration taught us resilience and adaptability. We made meaningful changes—like adding panic modes and offline support—that directly aligned with real user needs.

Overall, this project showed us how to design for urgency and empathy. We discovered that even under constraints, thoughtful design powered by AI and mobile technology can make a life-saving difference. More than just technical learning, this journey gave us the confidence that technology, when built with users at the center, can solve complex societal problems and create meaningful impact.

14. References

[1] A. Alanazy, Stuart Wark, John Fraser, Amanda Nagle, "Factors Impacting Patient Outcomes Associated with Use of Emergency Medical Services Operating in Urban Versus Rural Areas: A Systematic Review," Int. J. Environ. Res. Public Health, vol. 16, no. 10, 2019.

[2] N.C. Simpson and P.G. Hancock, "Fifty years of operational research and emergency response," J. Oper. Res. Soc., vol. 60, no. 1, pp. 126–139, 2009.

[3] G. Pettet, Ayan Mukhopadhyay, Mykel J. Kochenderfer, Abhishek Dubey, "On Algorithmic Decision Procedures in Emergency Response Systems in Smart and Connected Communities," Adaptive Agents and Multi-Agent Systems, 2020.

[4] A. Gonera and R. Pabst, "The Use of Design Thinking in Transdisciplinary Research and Innovation Consortia: Challenges, Enablers and Benefits," *Journal of Innovation Management*, vol. 7, no. 3, pp. 96–122, 2019.

[5] S. Palavesh, "The Role of Design Thinking in Conceptualizing and Validating New Business Ideas," *Journal of Informatics Education and Research*, 2024.

[6] E. E. I. Barcellos and G. Botura, “Design Thinking: User-Centered Multidisciplinary Methodology Based on People and Innovation,” in *Proc. Int. Conf. on Innovation Methods*, 2017.

[7] S. R. Rajput, M. S. Deshmukh, and K. V. Kale, “Cross-platform Smartphone Emergency Reporting Application in Urban Areas using GIS Location-based and Google Web Services,” Int. J. Comput. Appl., vol. 130, no. 12, pp. 1–6, Nov. 2015.

[8] N. P. Caliston and G. C. Tabia, “HELP ME: An Emergency Response Mobile Application,” J. Eng. Sci. Technol. Special Issue on ICITE2021, pp. 47–63, Oct. 2021, © School of Engineering, Taylor’s University.

[9] K. Pendke, S. Shegokar, R. Malghade, S. Datir, R. Adhau, and P. Thakre, “Mobile Application for Emergency Services,” JETIR, vol. 8, no. 5, May 2021.

[10] ELERTS Corporation, “Official ELERTS website,” Jan. 2013. [Online]. Available: http://elerts.com

[11] D. Amores, M. Vasardani, and E. Tanin, "Smartphone Usability for Emergency Evacuation Applications (Short Paper)," *Proc. COSIT 2019*, vol. 142, pp. 2:1–2:7, 2019.

[12] GoodSAM, "Alerter Volunteer (AV)," *GoodSAM*, Accessed: Apr. 20, 2025. [Online]. Available: https://www.goodsamapp.org/AV

[13] A. Khan, F. Muhammad, M. Ullah, H. Khan, M. Shoaib, and M. A. Khan, “Accident detection and smart rescue system using Android smartphone with real-time location tracking,” in Proc. Int. Conf. on Advances in Computing, Communication Control and Networking (ICACCCN), 2018, pp. 1–5.

[14] S.-Y. Chiou and J.-W. Lin, "A real-time, automated and privacy-preserving mobile emergency-medical-service network for informing the closest rescuer to rapidly support mobile-emergency-call victims," IEEE Access, vol. 6, pp. 68962–68974, 2018.

[15] H. Mutegeki, R. S. Rajamani, D. Thirumalai, and M. Kumar, “Interpretable Machine Learning-Based Triage for Decision Support in Emergency Care,” Proc. 7th Int. Conf. Trends Electron. Inform. (ICOEI), 2023, pp. 1234–1239.

[16] N. Ouerhani, M. Ben Halima, and A. M. Alimi, “Towards a Chatbot Based Smart Pervasive Healthcare Medical Emergency Cases,” in Proc.

2020 Int. Conf. on Advanced Systems and Emergent Technologies (IC_ASET), 2020, pp. 288–293.

[17] A. Valeriano, M. Van Heer, C. Perel, and D. Prieto Merino, “Crowdsourcing to Save Lives: A Scoping Review of Bystander Alert Technologies for Out-of-Hospital Cardiac Arrest,” Resuscitation, vol. 152, pp. 164–172, 2020.