



DEVELOPMENT OF A MULTI-CHANNEL EMERGENCY RESPONSE SYSTEM

Adesoji Adedeji Adegbola, Adebayo Olamide Solomon, Gafaru King Oluwashina,
and Adewale Saheed Eniola.

Software Engineering, School of Computing, Babcock University, Nigeria.

Cite this article:

A. A., Adegbola, Adebayo, O. S., Gafaru, K. O., Adewale, S. E. (2026), Development of a Multi-Channel Emergency Response System. British Journal of Computer, Networking and Information Technology 9(2), 50-67. DOI: 10.52589/BJCNIT-FXDPOLVE

Manuscript History

Received: 29 Apr 2026

Accepted: 2 Jun 2026

Published: 23 Jun 2026

Copyright © 2026 The Author(s).

This is an Open Access article distributed under the terms of Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0), which permits anyone to share, use, reproduce and redistribute in any medium, provided the original author and source are credited.

ABSTRACT: *Across many parts of the world, the escalating frequency of insurgency, banditry, and kidnapping has stretched emergency response infrastructure to its limits, bringing to light deep-seated inefficiencies in how security threats are communicated and managed. Traditional emergency response systems frequently fail to preserve coordinated and timely communication, producing outcomes that are operationally inadequate due to their centralized, manual, and reactive processing model. In this study, we present the development of a mobile-based, multi-channel emergency security alert system built on a real-time location tracking and automated agency notification pipeline. The system integrates React Native for cross-platform mobile interaction, Django REST Framework for backend logic and API communication, and a multi-channel delivery model transmitting alerts via push notifications, SMS, and email. A custom acknowledgment feedback mechanism manages response status updates, while a layered architecture maintains separation of concerns across components, enabling scalability and modular extensibility. The system is evaluated using response time, alert delivery reliability, and usability metrics. Results confirm that the system fulfills its core functional requirements reliably. Unit and system testing returned a full pass rate across all critical test cases, encompassing authentication, priority computation, agency assignment, and end-to-end alert delivery. Multi-channel notification was successfully achieved across push notifications, SMS, and email, with alert status tracking progressing accurately through all defined response stages. These findings demonstrate the viability of a mobile-first, automated approach to real-time security alert management in high-risk environments.*

KEYWORDS: Emergency Response, Multi-Channel Communication, Real-Time Location Tracking, Automated Notification, Mobile Application, Layered Architecture.



INTRODUCTION

In recent years, the manner in which societies respond to emergencies has undergone considerable transformation. Advances in mobile computing, the Internet of Things (IoT), and managing crises with a speed and precision that conventional infrastructure cannot match. Traditional systems, including sirens, television broadcasts, and manual dispatch protocols, were not designed to meet the demands of this environment. They are limited in their ability to deliver location-specific alerts, inter-agency coordination remains fragmented, and dependence on manual workflows introduces delays with potentially fatal consequences (Bajwa, 2025; Indalkar et al., 2025; Smith et al., 2018). As urban populations expand and security threats grow more varied in nature, these structural limitations have become increasingly difficult to justify.

The technologies now available present a credible basis for reform. IoT sensor networks support continuous, real-time environmental monitoring across large geographic areas (Han Zhang et al., 2025; Patel et al., 2020), while machine learning models enable predictive risk assessment and more efficient allocation of emergency resources (Han Zhang et al., 2025). Mobile platforms such as SafeConnect have demonstrated that location tracking, instant alert dissemination, and community-level incident reporting can be consolidated within a single application, enabling individuals to participate meaningfully in their own safety (Han Zhang et al., 2025; Sairaj Jadhav et al., 2025). Empirical results support this trajectory: IoT-based systems have achieved alert latency below 500 ms in simulated deployment environments (Han Zhang et al., 2025), and mobile-integrated solutions have produced measurable reductions in emergency communication timelines (Bajwa, 2025; Indalkar et al., 2025).

What the existing literature has not adequately addressed, however, is the problem of integration. Most current platforms attend to individual components of emergency response without establishing meaningful connections between them. Multi-channel architectures capable of sustaining reliable communication across varied network conditions, including environments where standard data transmission is unavailable, remain uncommon (Han Zhang et al., 2025). Emergency agencies continue to operate from independent information systems, limiting the shared situational awareness that effective crisis coordination requires (Han Zhang et al., 2025). Of particular concern is the absence of robust mechanisms for automated, simultaneous notification across multiple security bodies, including law enforcement, emergency medical services, and fire departments (Chuckee B. Gatchalian et al., 2025). The practical consequences of these gaps are well documented: delayed interventions, misallocated resources, prolonged crises, and avoidable loss of life.

The system was developed in direct response to these deficiencies. The system is a mobile-based, multi-channel emergency security alert platform that enables real-time incident reporting, automated location tracking, severity-aware alert routing, and simultaneous notification of relevant security agencies with minimal reliance on manual input. Through a unified interface accessible on both mobile and web, the platform centralises situational awareness and establishes a more direct communication pathway between those in distress and the agencies responsible for their safety.

This paper presents the design, implementation, and evaluation of the system, covering its system architecture, core functionality, underlying technologies, and the technical challenges encountered during development. The broader objective is to contribute a scalable, dependable,



and practically viable emergency response solution suited to the evolving public safety demands of modern communities.

Statement of Problem

Current public safety systems are centralized, reactive, and rely on manual processes that cause significant delays in detecting and responding to emergencies. These systems operate in separate, disconnected networks that prevent first responders from accessing real-time data, while generic alerts often create confusion rather than helping people respond effectively. They also lack protection against network disruptions and fail to involve citizens as active participants in emergency response. The proposed solution addresses these limitations through an intelligent, real-time mobile application system that detects emergencies and delivers instant location-based alerts through multiple channels such as push notifications, SMS, and in-app messages. Key features include real-time location tracking for better coordination with authorities and strong backup mechanisms to ensure alerts are delivered even during network disruptions. This integrated approach empowers communities as active participants in their own safety while giving emergency responders the real-time information they need for effective decision-making.

Aim and Objectives

The aim of this study is to develop a multi-channel emergency security alert system that provides real-time location tracking and automated notification to relevant security agencies, in order to enhance rapid response to terrorism, banditry, and kidnapping incidents, thereby improving the safety and security of vulnerable individuals

The project will achieve its aim through the following generative objectives:

- To design a multi-channel emergency security alert system architecture
- To implement the proposed emergency alert system
- To evaluate the performance of the proposed system

LITERATURE REVIEW

Modern societies face increasingly complex emergency security challenges driven by urban infrastructure growth and the rising frequency of both natural and man-made disasters (Caprotti & Cowley, 2019). Traditional public safety systems, which rely on centralised infrastructures and manual processes, are hindered by delays, inefficiencies, and a lack of real-time situational awareness (Evgrafova et al., 2022). Conventional alert mechanisms such as sirens, radio broadcasts, and manual dispatch are often generic, lacking the personalisation needed to deliver precise, localised instructions to affected populations (Gojmerac et al., 2016). These limitations can cause confusion, panic, and secondary incidents that further complicate emergency response. Timely communication is paramount in mitigating such impacts; real-time alerts provide individuals with a heightened sense of security and enable faster personal responses (Leederman et al., 2020), while effective crisis communication strategies depend on delivering accurate information promptly (Lee, 2021). Technology plays a transformative role in



addressing these deficiencies by integrating real-time data acquisition, automated decision-making, and rapid communication into emergency frameworks, with AI-driven predictive analytics capable of anticipating risks and optimising resource allocation (Kumar & Yadav, 2022).

The evolution of emergency response systems from rudimentary methods to technologically integrated frameworks has produced several key innovations. Multi-channel dissemination strategies now leverage SMS, mobile applications, social media, and traditional sirens, while modern mobile apps provide instant updates, panic buttons, and geo-fencing for location-based alerts (Chuckee et al., 2023; Han Zhang et al., 2025). The Internet of Things (IoT) provides granular, real-time data through sensor networks, and when augmented by machine learning, empowers systems with predictive analytics to forecast risks (Han Zhang et al., 2025). Real-time location tracking, powered by GPS and complemented by GSM for areas with limited connectivity, is foundational to modern emergency dispatch, though accuracy remains challenged by poor signal conditions and infrastructural gaps (Ali et al., 2024; Rao, Bapat, & Dandekar, 2012). Automated agency notification systems integrate geolocation APIs, GPS, and IoT to enable real-time tracking of distressed individuals and available emergency units, optimising resource deployment and significantly improving response times (Harshit et al., 2025). Modern system architectures employ multi-tiered structures comprising emergency detection units, processing units, and client applications, supported by cloud-based backends and edge computing for low-latency, scalable operations (Daniel et al., 2019; Vamsy et al., 2024). Evaluation metrics for such systems include response time, with leading platforms achieving alert delivery under 450 milliseconds, delivery success rates of 99.1%, and detection accuracy exceeding 95% across multiple event types (Bajwa et al., 2025).

Despite these advancements, significant research gaps persist. Public safety infrastructure in many regions remains outdated, siloed, and ill-equipped for contemporary emergencies (Bajwa, 2025). A substantial gap exists in comprehensive, modular frameworks adaptable to multi-hazard scenarios with seamless inter-agency coordination, while many systems lack resilience mechanisms against network disruptions, cyberattacks, or overload during large-scale disasters. Critical areas such as citizen integration, adaptive messaging, and participatory safety models remain underrepresented in current research (Bajwa, 2025). Communication challenges are particularly acute in developing contexts, where poor signal transmission in rural and semi-urban areas (Olapade, 2020; Eze & Adebayo, 2019) and incompatible protocols among emergency services hinder real-time information sharing (Muhammad et al., 2023). These gaps underscore the critical need for an integrated, scalable, and context-aware system combining real-time data acquisition, automated decision-making, and rapid communication—one that prioritises human-centric design by involving citizens as both sensors and responders through personalised, location-aware alerts and two-way communication channels (Bajwa, 2025). It is this convergence of identified limitations and technological possibilities that motivates the design of the proposed system, a multi-channel emergency security alert system with real-time location tracking and automated agency notification aimed at addressing these persistent challenges within the Nigerian security context.



METHODOLOGY

To achieve the stated aim of developing, implementing, and evaluating the Multi-Channel Emergency Security Alert System, this study adopts a structured, objective-driven methodological approach. The methodology is organised around three principal stages: requirements analysis and system design, system implementation, and performance evaluation. Each stage builds upon the preceding one to ensure that the final system is both technically sound and aligned with the practical demands of emergency response.

System Design

The first stage focuses on establishing a clear and comprehensive design foundation for the emergency alert system. It begins with a detailed analysis of system requirements to identify the core functionalities necessary for effective emergency reporting and coordinated agency response. These requirements include enabling civilians to submit emergency alerts with minimal friction, capturing real-time location data at the point of incident, automating communication with the appropriate security agencies, providing acknowledgment feedback to users, and ensuring that all personal and location data is handled securely throughout the process.

To give these requirements a concrete structural form, a Layered Architecture is designed, organising the system into four distinct layers: the presentation layer, the application logic layer, the integration layer, and the data layer. Each layer carries clearly defined responsibilities, and the architecture specifies how information flows between them during an active emergency scenario. This separation of concerns is intended to support both scalability and maintainability as the system evolves.

Several modelling techniques are employed to represent the system's structure and behaviour with greater precision. A conceptual diagram is developed to establish the high-level relationships between the system's core entities before implementation details are introduced. The system architecture diagram then provides a more technical view, illustrating how the various components, including the mobile client, backend services, notification channels, and database, relate to and communicate with one another. Use case diagrams are used to capture the interactions between the system's primary actors, namely civilians and security agencies, and the functions available to each. Sequence diagrams trace the step-by-step flow of emergency alerts, location data, and acknowledgment feedback as they pass through the defined architectural layers. Finally, an Entity-Relationship Diagram (ERD) defines the logical structure of the database, specifying the entities, their attributes, and the relationships between them that underpin the system's data management.

System Implementation

The second stage translates the design artifact produced in the first stage into a functional system. The platform is implemented as a mobile application, allowing civilians to report emergencies directly from their smartphones. The mobile frontend is built using React Native to support cross-platform deployment, with Redux employed for application state management. The backend infrastructure is developed with Django and Django REST Framework, handling API communication, alert processing, and the coordination of notification delivery. MySQL



serves as the database management system, storing user records, emergency alerts, real-time location coordinates, and acknowledgment data.

Real-time location tracking is implemented through mobile GPS APIs, which automatically capture and transmit the user's geographic coordinates at the time of an emergency. Automated, multi-channel notification delivery is achieved through the integration of Firebase Cloud Messaging for push notifications, SMS gateway services including Twilio and Africa's Talking for text-based alerts, and an optional email service retained for record-keeping purposes. These components are integrated to ensure that alerts are transmitted reliably and without unnecessary delay from the reporting civilian to the relevant security agencies.

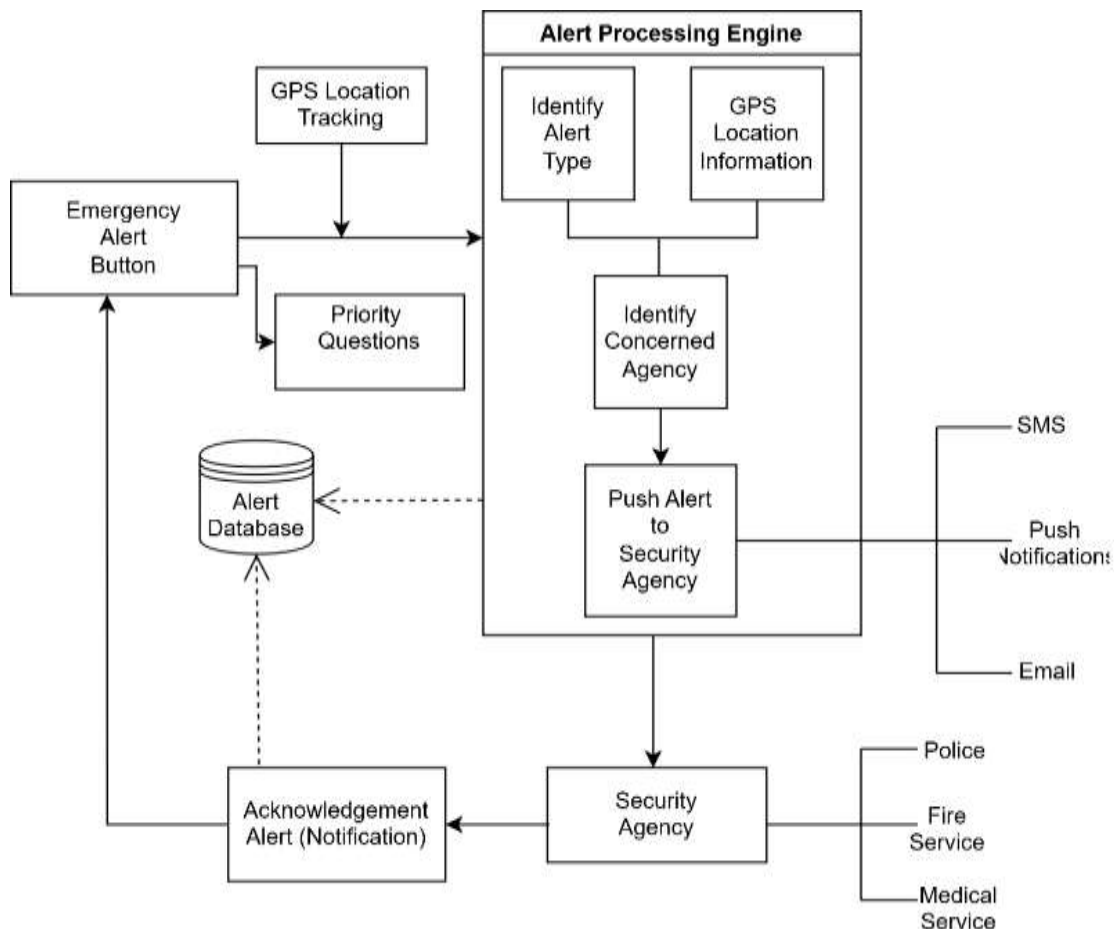
Performance Evaluation

The final stage assesses whether the implemented system meets the objectives established at the outset. Evaluation is conducted through controlled simulations representing a range of emergency scenarios, with performance measured across four key metrics: response time, alert delivery reliability, multi-channel notification effectiveness, and overall system functionality. Usability testing is carried out to evaluate the ease with which users can navigate the application, submit alerts, and interpret the acknowledgment feedback they receive. A security evaluation is also conducted to confirm that user data and location information are transmitted and stored in a manner consistent with data protection expectations. The findings from these evaluations are analysed to determine the extent to which the system fulfils its design objectives and to surface areas warranting further refinement.

Conceptual Diagram

The proposed emergency alert system follows a multi-channel notification architecture designed to ensure rapid and comprehensive dissemination of emergency information. The system is initiated when an emergency event occurs, which first presents a Priority Questions dropdown — a set of contextual questions including whether people are trapped, the speed of fire spread, proximity of fuel or chemical hazards, and the current severity of injuries at the scene. The responses to these questions are used to determine the priority level of the alert before it is passed forward for processing. This triggers the Alert Processing Engine. This module processes the alert data type, GPS location information, identify concerned agency, then push the processed information to notification service to be disseminated to the appropriate security agency.

The Notification Service employs a parallel processing approach, simultaneously routing alerts through four distinct communication channels: SMS, Email, and Push Notification. This multi-channel strategy ensures that alerts reach end users through multiple touchpoints, significantly increasing the probability of successful message delivery and user awareness during critical situations.

Figure 1: Conceptual Design for a Multi-Channel Emergency Response System

System Architecture

The system architecture diagram illustrates the structural organization and interaction of the core components of the Multi-Channel Emergency Alert System based on a Layered Architecture. The architecture is organized into four layers: the Presentation Layer, Application Layer, Integration Layer, and Data Layer, each responsible for specific system functions.

The Presentation Layer consists of the User and the Mobile Application, which provides the interface through which civilians initiate emergency alerts, capture real-time location information using mobile GPS, and receive alert status and acknowledgment feedback.

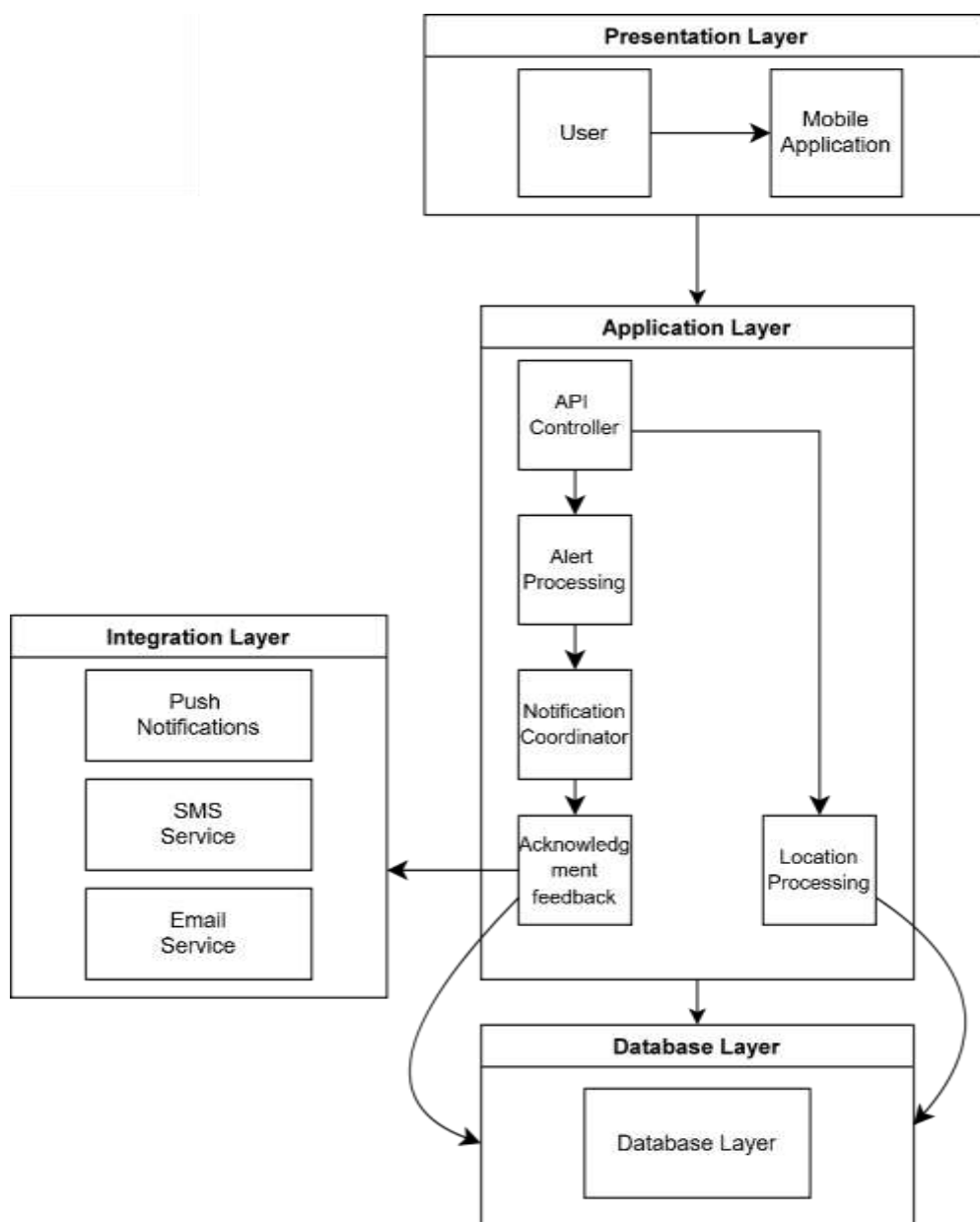
The Application Layer coordinates all core system operations and is composed of logical components responsible for receiving requests from the mobile application, processing emergency alerts and location data, selecting appropriate security agencies, coordinating multi-channel notification delivery, and managing acknowledgment workflows. This layer also controls communication between the other layers of the system.

The Integration Layer supports multi-channel communication by providing external notification services, including Push Notification, SMS, and Email services, which are used to deliver emergency alerts to security agencies.

The Data Layer manages data storage and persistence, including user information, emergency alerts, location data, and acknowledgment records.

Overall, the layered architecture provides a structured separation of responsibilities that enhances system maintainability, reliability, and efficient emergency alert delivery.

Figure 2: **Software Architectural Design for a Multi-Channel Emergency Response System**



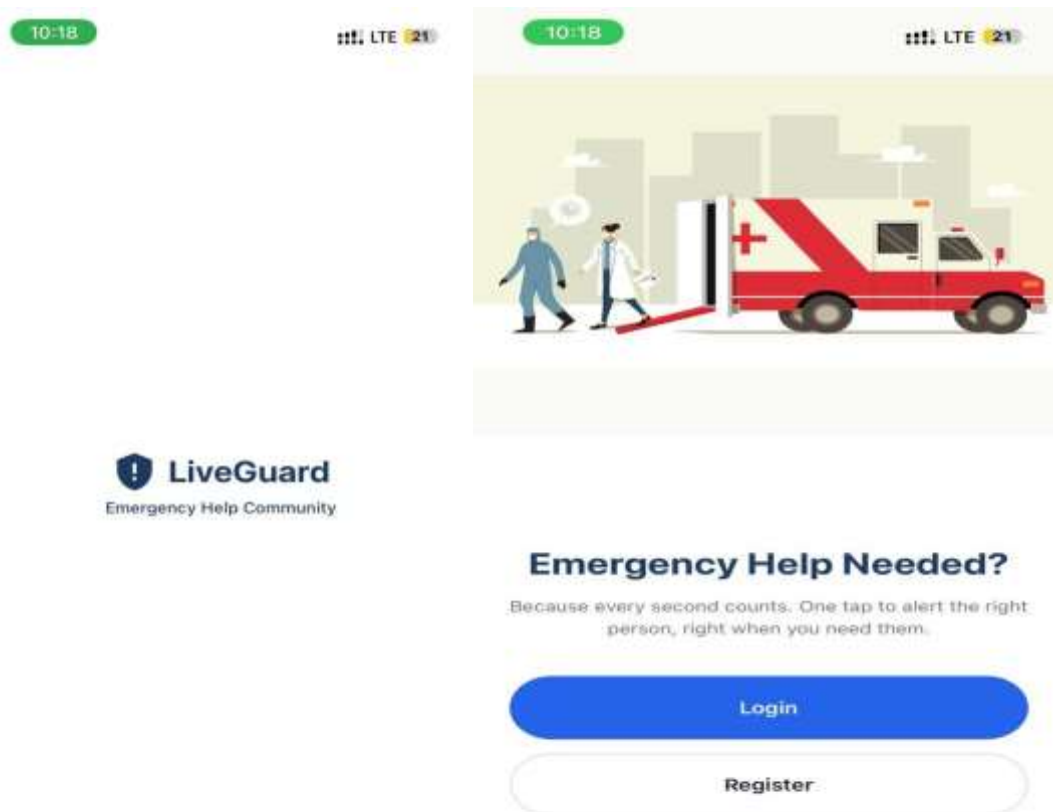
RESULTS

The system was implemented as a multi-component platform comprising a React Native mobile application for civilian users, a Django-based backend API, and two web dashboards serving security agencies and system administrators respectively. This section presents the key interfaces of the implemented system across all three application types, demonstrating the complete emergency alert workflow from initial submission through agency notification and acknowledgment feedback.

User Mobile Application

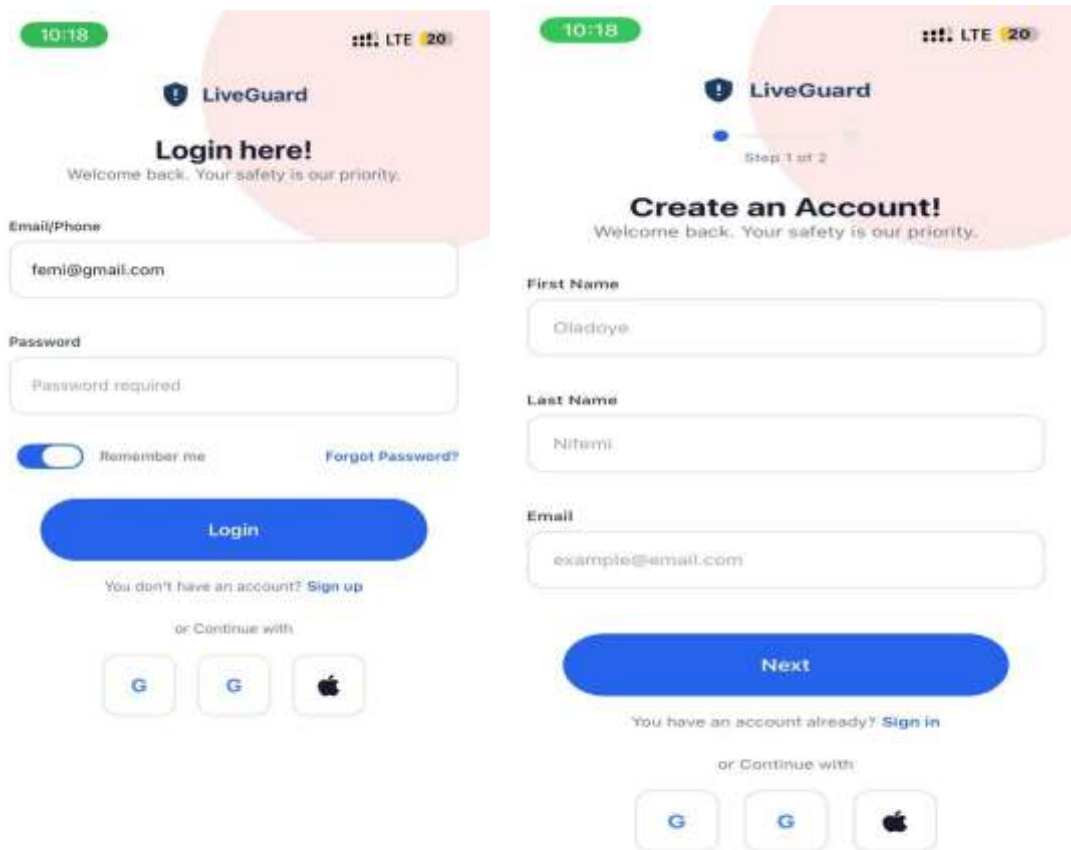
The splash screen displays the the system logo and application name upon launch, followed by an onboarding sequence that introduces the core features of the system to new users, including emergency alert submission, real-time location tracking, and acknowledgment feedback.

Figure 3: Splash Screen and Onboarding

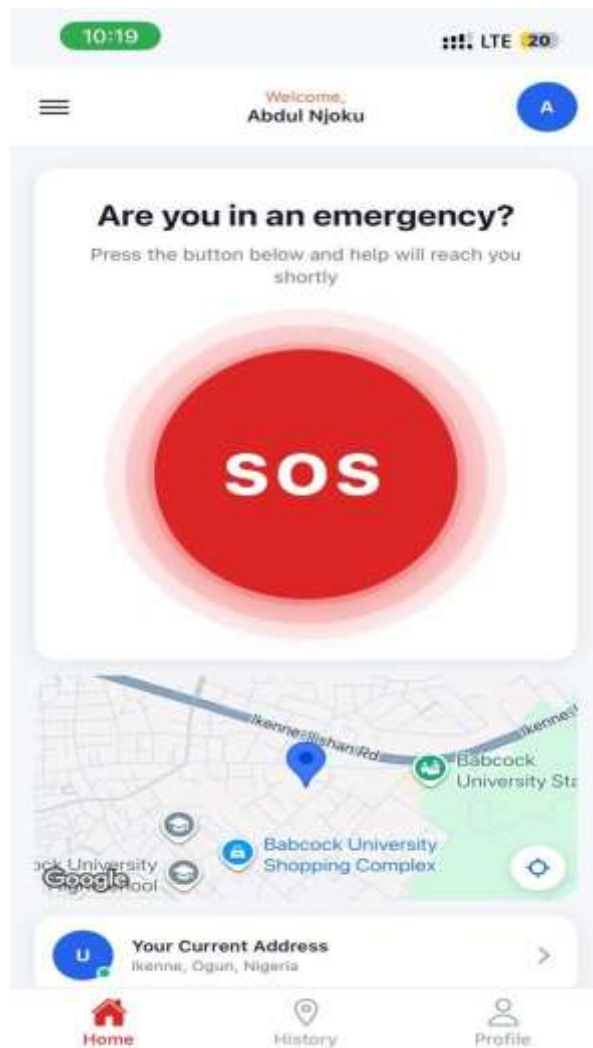


The registration screen allows new users to create an account by providing their full name, email address, phone number, and a secure password, with an optional field for emergency contact information. The login screen provides returning users with email and password authentication, secured through token-based session management.

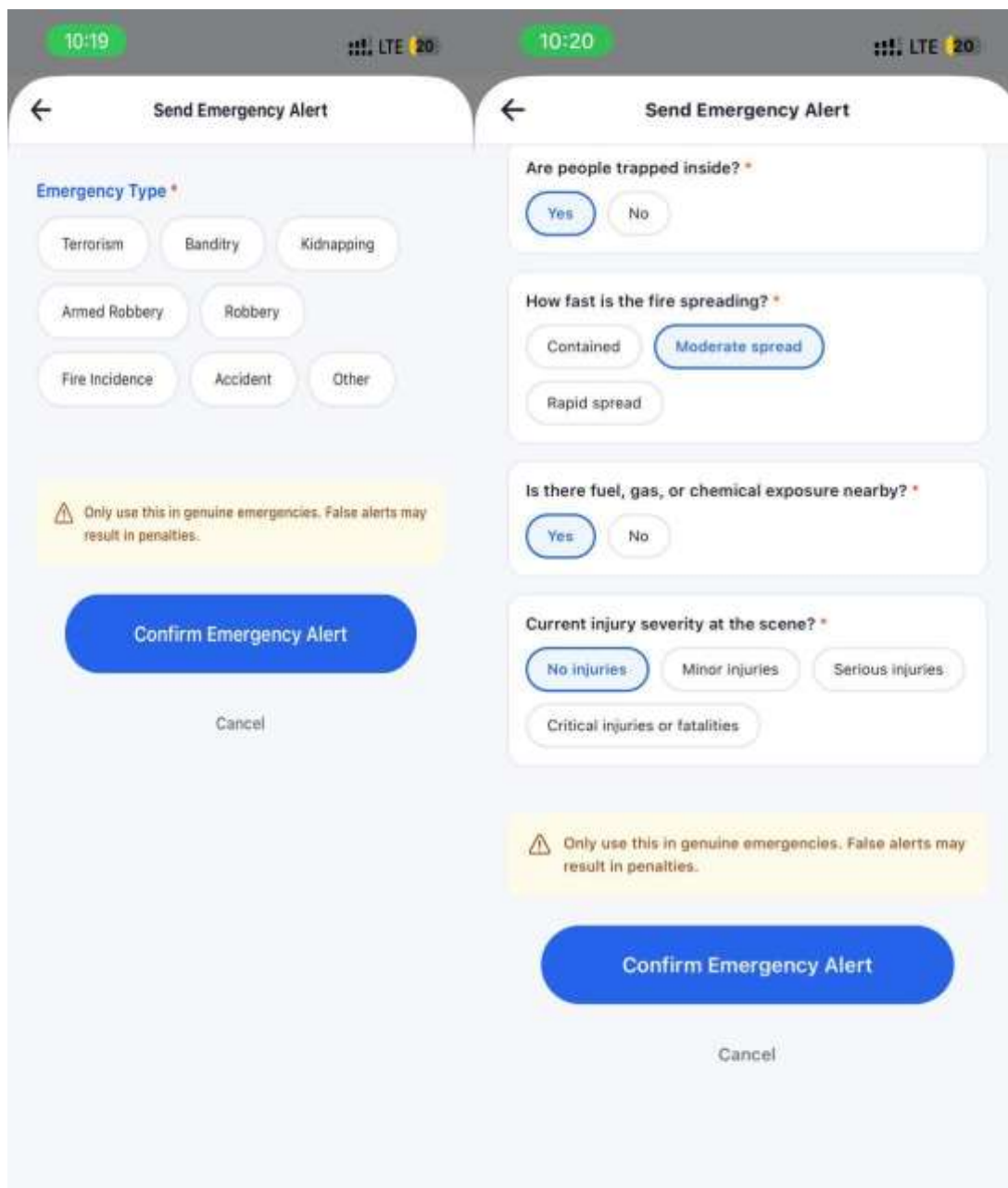
Figure 4: User Registration and Login Screens



The home dashboard serves as the primary navigation hub of the mobile application. It features a prominent SOS button for initiating emergency alerts, quick-access options for viewing alert history, managing profile settings, and accessing emergency contact information. The dashboard also displays the user's current location status and any active alert notifications.

Figure 5: Home Dashboard

Upon pressing the SOS button, the user is presented with the emergency type selection interface, where they choose from options including terrorism, banditry, kidnapping, or other. Following type selection, the user is presented with a set of contextual priority questions covering whether people are trapped, the speed of hazard spread, proximity of fuel or chemical exposure, and current injury severity at the scene. The responses to these questions are used by the system to automatically compute and assign a priority level to the alert before submission. GPS coordinates are captured in the background throughout this process.

Figure 6: Emergency Type Selection and Priority Questions

The figure displays two sequential screenshots of a mobile application interface for sending an emergency alert. Both screenshots show the time as 10:19 and 10:20, and the network as LTE 20.

The left screenshot, titled "Send Emergency Alert", shows the "Emergency Type" selection screen. It features a grid of buttons for selecting the emergency type: Terrorism, Banditry, Kidnapping, Armed Robbery, Robbery, Fire Incidence, Accident, and Other. Below the grid is a warning message: "Only use this in genuine emergencies. False alerts may result in penalties." At the bottom, there is a large blue "Confirm Emergency Alert" button and a "Cancel" option.

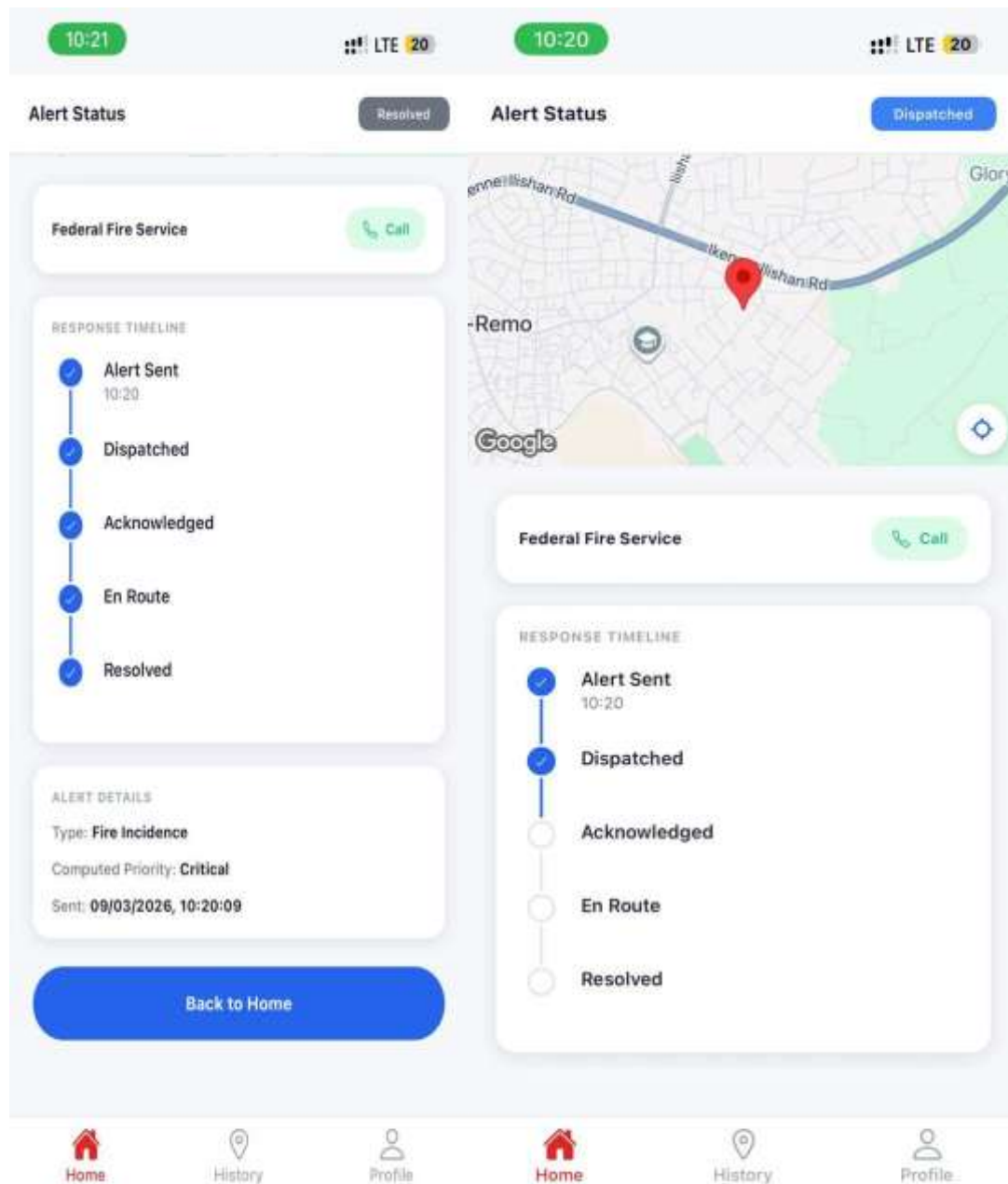
The right screenshot, also titled "Send Emergency Alert", shows the "Priority Questions" screen. It contains four questions with radio button options:

- "Are people trapped inside?" with options "Yes" and "No".
- "How fast is the fire spreading?" with options "Contained", "Moderate spread", and "Rapid spread".
- "Is there fuel, gas, or chemical exposure nearby?" with options "Yes" and "No".
- "Current injury severity at the scene?" with options "No injuries", "Minor injuries", "Serious injuries", and "Critical injuries or fatalities".

Below the questions is another warning message: "Only use this in genuine emergencies. False alerts may result in penalties." At the bottom, there is a large blue "Confirm Emergency Alert" button and a "Cancel" option.

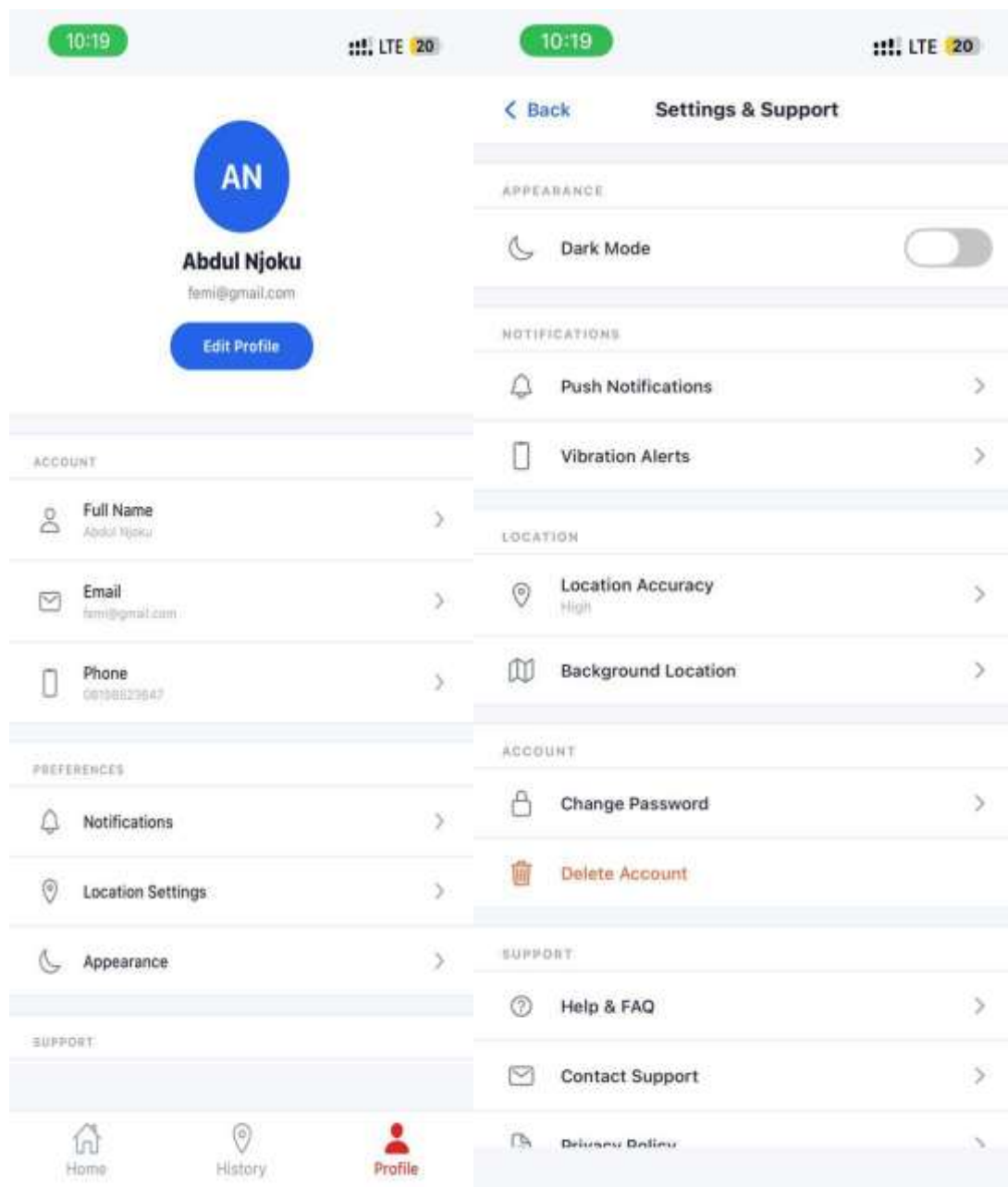
Following alert submission, the user is directed to the alert status screen, which provides real-time updates on the progress of their alert through the system, including confirmation when security agencies acknowledge receipt. A cancel option is available for users who need to retract a false alarm. The alert history screen presents a chronological list of all past alerts, including status, timestamp, emergency type, and responding agency. For resolved alerts, users are prompted to submit a satisfaction rating, providing feedback on the quality and timeliness of the emergency response received.

Figure 7: Alert Status and Alert History



The profile screen allows users to update their personal information, manage emergency contacts, adjust notification preferences, and modify account settings, ensuring that contact details remain accurate for effective emergency communication.

Figure 8: Alert Status and Alert History Screens

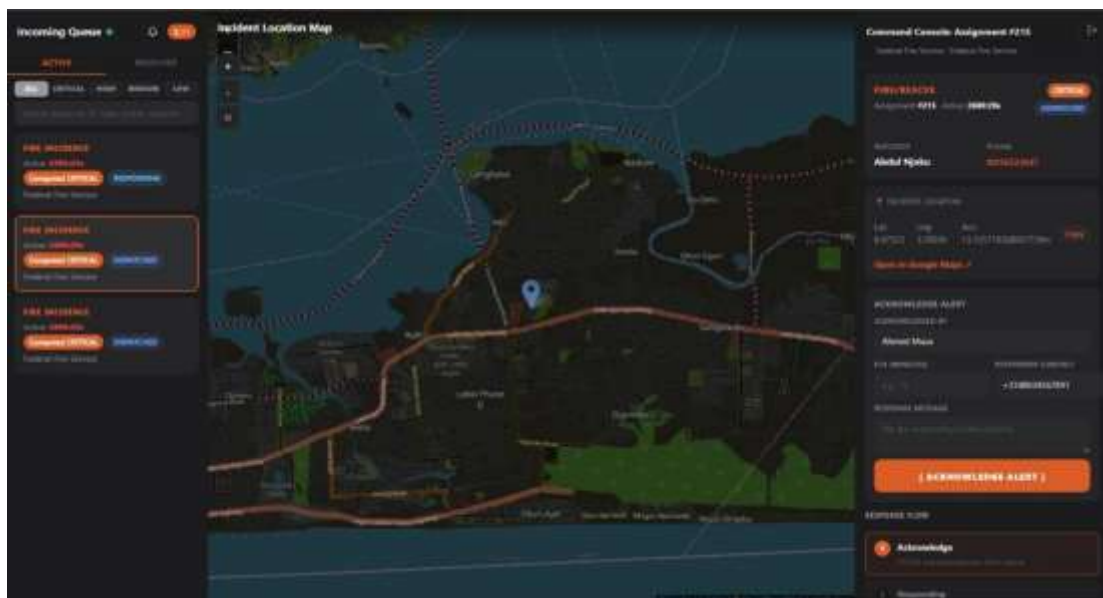


Agency Web Application

The agency web dashboard provides security agency personnel with a dedicated desktop interface for managing incoming emergency alerts. The dashboard displays a real-time feed of assigned alerts with key details including emergency type, computed priority level, location coordinates, and submission timestamp. Agency users can acknowledge alerts, update alert statuses as the response progresses, view location data on an integrated map, and access the full history of alerts assigned to their agency. This web-based interface supports the operational

requirements of agency dispatch teams who require a persistent, multi-alert view that would be impractical on a mobile device.

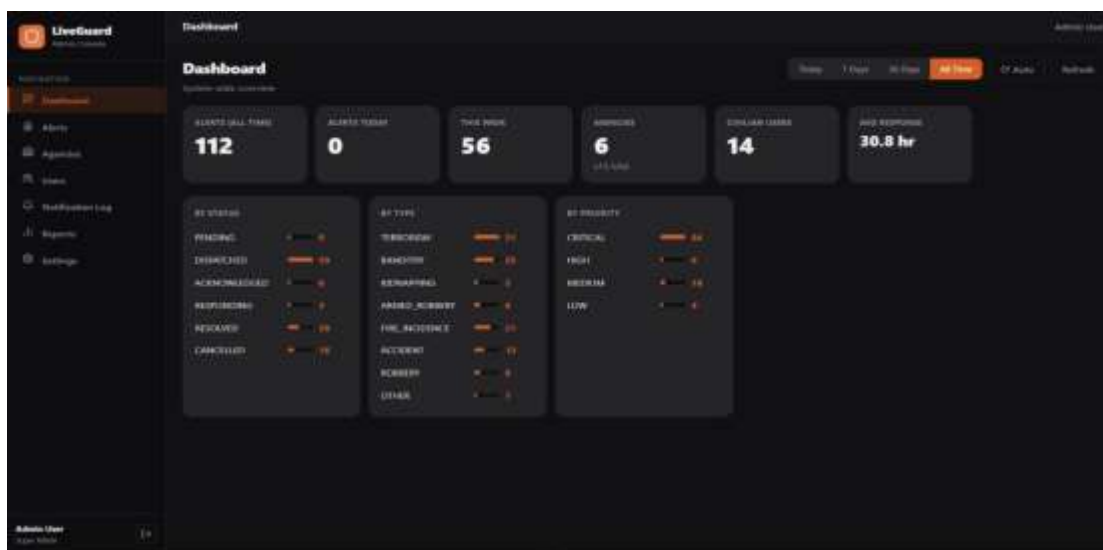
Figure 9: Agency Web Dashboard



Admin Web Application

The admin web dashboard provides system administrators with tools for overseeing all system operations. Administrators can manage user and agency accounts, monitor system-wide alert activity, view notification delivery logs across all channels, and generate summary reports. The dashboard also provides system health indicators and performance monitoring capabilities that support ongoing system evaluation and maintenance.

Figure 10: Admin Web Dashboard





Collectively, these interfaces demonstrate the complete emergency alert workflow from user registration through alert submission, automated priority computation, multi-channel agency notification, acknowledgment feedback, and post-resolution rating, realising the design objectives established in the earlier chapters of this study.

CONCLUSION

This study presented the design and implementation of the system, a multi-channel emergency security alert system with real-time location tracking and automated agency notification, developed to address the limitations of traditional emergency response systems in Nigeria. Existing public safety infrastructures are centralized, reactive, and plagued by fragmented communication, manual workflows, and information silos that introduce critical delays during emergencies such as terrorism, banditry, and kidnapping.

The proposed system employs a layered architecture and is built using React Native and Django REST Framework, enabling civilians to trigger emergency alerts via a single SOS action that automatically captures GPS coordinates and dispatches notifications simultaneously through push notifications, SMS, and email to relevant security agencies. An acknowledgment feedback mechanism closes the communication loop by confirming to users that their alert is being acted upon. The multi-channel strategy ensures redundancy in alert delivery, which is particularly significant in the Nigerian context where network instability is prevalent, mitigating the risk of undelivered alerts and enhancing timely, location-accurate emergency communication.

FUTURE WORK

Future work should focus on field deployment and real-world testing, integration of AI-based predictive analytics for proactive threat detection, offline alert capabilities for low-connectivity regions, and inter-agency data sharing dashboards to further strengthen the system's utility as a comprehensive emergency management platform.

REFERENCES

- Agoylo, J. C. Jr., Narciso, R. S., Ortega, J. G. D., Gula, R. O., & Silvio, A. B. (2024). Municipal emergency response system (MERS) using GMaps and SMS. <https://doi.org/10.5281/zenodo.11189470>
- Akinsanmi, O., Omodunbi, B., Ayeoribe, O. P., Esan, A., & Oluwole, A. S. (2025). Impact of inadequate signal transmission on Ekiti State's emergency response systems. *The Journal of Communication Technology*.
- Alavi, A., Bogu, G. K., Wang, M., Rangan, E. S., Brooks, A. W., Wang, Q., Higgs, E., Celli, A., Mishra, T., Metwally, A. A., & Snyder, M. P. (2022). Real-time alerting system for COVID-19 and other stress events using wearable data. *Nature Medicine*. <https://doi.org/10.1038/s41591-021-01593-2>



- Arnob, R. M. (2025). SafeConnect: An inclusive personal safety, connectivity and real-time emergency alert app [Bachelor's project]. Brac University.
- Bajwa, A. (2025). AI-based emergency response systems: A systematic literature review on smart infrastructure safety. *American Journal of Advanced Technology and Engineering Solutions*, 1(1), 174–200. <https://doi.org/10.63125/xcxwpv34>
- Chavula, P., Kayusi, F., Lungu, G., & Uwimbabazi, A. (2025). The current landscape of early warning systems and traditional approaches to disaster detection. *LatIA*, 3, 77. <https://doi.org/10.62486/latia202577>
- Costa, D. G., Vasques, F., Portugal, P., & Aguiar, A. (2020). A distributed multi-tier emergency alerting system exploiting sensors-based event detection to support smart city applications. *Sensors*, 20(1), 170. <https://doi.org/10.3390/s20010170>
- Damaševičius, R., Bacanin, N., & Misra, S. (2023). From sensors to safety: Internet of Emergency Services (IoES) for emergency response and disaster management. *Journal of Sensor and Actuator Networks*, 12, 41.
- Dande, A., Hava, S., Rajmani, M., Sarangi, P., & Kalbhor, P. (2023). Emergency alerting system. *IOSR Journal of Computer Engineering*, 25(3), 33–37.
- Faheem, M., Butt, R. A., Raza, B., Ashraf, M. W., Ngadi, M. A., & Gungor, V. C. (2019). A multi-channel distributed routing scheme for smart grid real-time critical event monitoring applications in the perspective of Industry 4.0. *International Journal of Ad Hoc and Ubiquitous Computing*, 32(4), 236–256.
- Gatchalian, C. B., Pascual, K. F. C., & Domingo, M. S. (2023). Enhancing community resilience through the government text alert system: A case study in Baranggay San Miguel, Bulacan. *International Research Journal of Advanced Engineering and Science*, 8(3).
- Hasanuzzaman, M., Hossain, S., & Shil, S. K. (2023). Enhancing disaster management through AI-driven predictive analytics: Improving preparedness and response. *International Journal of Advanced Engineering Technologies and Innovations*, 1(1).
- Indalkar, A. S., Tamgave, P. A., Khalate, S. P., Gurav, S. D., & Pawar, S. T. (2025). Enhancing community safety through real-time mobile applications: Emergency alerts, live tracking, and monitoring systems. *Journal of Emerging Technologies and Innovative Research*, 12(4).
- Jadhav, S., Jadhav, S., Ghodella, M., Jain, H., & Mondal, D. (2025). Smart regional alert and navigation system. *International Journal on Advanced Computer Theory and Engineering*, 14(1).
- Jolhe, R., Hadawale, S., Bagul, P., Bhoir, R., & Kamble, P. (2025). Quick-Rescue: A smart emergency response application. *International Journal of Engineering Applied Sciences and Technology*, 9(11), 49–57.
- Kar, S. K., Prity, F. S., Fahad, N., Hasan, M. M., Morol, M. K., & Al Jubair, M. A. (2024). Rokkha: Enabling offline emergency location sharing via SMS. *Proceedings of ICCA 2024, Dhaka, Bangladesh*. <https://doi.org/10.1145/3723178.3723190>
- Krishna, B. V., Sai, B. B., Vineela, B., Praneetha, B., & Priya, B. L. V. (2024). IoT based disaster alerting system for smart city environment. *Journal of Emerging Technologies and Innovative Research*, 11(4).
- Ma, M., Bartocci, E., Lifland, E., Stankovic, J., & Feng, L. (2021). A novel spatial-temporal specification-based monitoring system for smart cities. *arXiv:2104.04904*.



- Mounika, J., Charanjit, N., Saitharun, B., & Vashista, B. (2021). Accident alert and vehicle tracking system using GPS and GSM. *Asian Journal of Applied Science and Technology*, 5(2), 81–89.
- Odulaja, G. (2025). Intelligent GPS and sensor controlled tracker and locator for human trafficking, kidnapping, banditry and theft control in Nigeria. *MUST Journal of Research and Development*, 6(2). <https://doi.org/10.62277/mjrd2025v6i20013>
- Olajuyigbe, A. E., Adegboyega, S. A., & Adenigba, A. D. (2015). Spatial analysis of factors responsible for spread of crime activities in Akure, Nigeria, using GIS techniques. *International Journal of Criminology and Sociological Theory*, 8(1), 1–19.
- Pandey, N., Rani, P., & Kundu, S. (2025). An intelligent disaster management system: Integrating technology for effective response and recovery. *Proceedings of the 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC 2025)*.
- Tiwari, H. P., Kumar, D., Jha, K., Priyanka, Umar, S. W., Upadhyay, P., & Singhal, V. (2025). Real-time emergency response system using geolocation, data analytics, and machine learning. *IJIRAE: International Journal of Innovative Research in Advanced Engineering*, 12(6), 287–291. <https://doi.org/10.26562/ijirae.2025.v1206.01>
- Zhang, H., Zhang, R., & Sun, J. (2025). Developing real-time IoT-based public safety alert and emergency response systems. *Scientific Reports*. <https://www.nature.com/scientificreports>