INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
VOLUME: 04 ISSUE: 04 | APRIL – 2025 DOI: 10.55041/ISJEM02860
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

Disaster Response And Relief Co-ordination

Karan Didwagh
Department of Computer Engineering
Atharva College of Engineering
Mumbai, India
didwaghkaran-cmpn@atharvacoe.ac.in

Tanmay Gadekar
Department of Computer Engineering
Atharva College of Engineering
Mumbai, India
gadekartanmay-cmpn@atharvacoe.ac.in

Yogesh Gupta
Department of Computer Engineering
Atharva College of Engineering)
Mumbai, India
guptayogesh-cmpn@atharvacoe.ac.in

Darshan Jagtap
Department of Computer Engineering
Atharva College of Engineering
Mumbai, India
jagtapdarshan-cmpn@atharvacoe.ac.in

Prof. Ashwini Kachare
Department of Computer Engineering
Atharva College of Engineering
Mumbai, India
ashwinikachare@atharvacoe.ac.in

***Abstract*— Effective disaster response is often hindered by fragmented communication and uncoordinated relief efforts, resulting in delays, resource mismanagement, and increased casualties. This project presents the design and development of a mobile-based Disaster Response and Relief Coordination System built using React Native and Appwrite, aimed at centralizing and streamlining post-disaster operations. The system enables role-based access for volunteers, NGOs, and administrators, supporting features such as user registration with role verification, real-time disaster reporting, emergency alerts with GPS location, and availability tracking of essential supplies and services. Administrators can approve user roles, verify credentials of aid providers, and manage disaster-related posts. The app also provides disaster preparedness guidelines, and a map-based interface for locating nearby shelters and aid centers. By integrating real-time communication and verified information dissemination, the system enhances collaboration, reduces response time, and improves the overall efficiency of disaster relief efforts.**

Keywords*— *Disaster management, React Native, Appwrite, real-time alerts, emergency response, mobile application, resource coordination, role-based access.

I. INTRODUCTION

Disaster response is a highly complex and time-sensitive process involving multiple organizations, each contributing unique skills, resources, and expertise. These organizations play crucial roles in the aftermath of both natural disasters—such as earthquakes, floods, and wildfires—and man-made crises like industrial accidents or conflict-related emergencies. In such high-stakes environments, a coordinated and timely approach is essential for minimizing casualties and reducing damage.

However, current disaster management systems are often plagued by fragmented communication, lack of coordination, and inefficient resource allocation. Often, rescue agencies work in isolation, resulting in duplicated efforts, poor communication, and considerable delays in delivering aid to affected populations. The absence of a centralized, real-time communication and coordination platform exacerbates these challenges, limiting the ability of first responders to act swiftly and effectively.

To address these critical gaps, this project proposes the development of a centralized disaster response and coordination platform, supported by a robust database and an accompanying mobile application. The proposed system is designed to function as a unified communication hub, enabling real-time interaction and collaboration among users, volunteers, NGOs, and affected individuals. By enhancing operational efficiency through real-time information sharing, the platform aims to streamline resource allocation with greater transparency and accessibility.

Furthermore, it seeks to enable faster and more strategic decision-making during emergencies by providing a centralized and reliable source of information. Through this integrated approach, the platform has the potential to significantly improve the

effectiveness of disaster relief efforts, reduce response times, minimize resource duplication, and ultimately save more lives.

II. LITERATURE REVIEW

Numerous studies have highlighted the pivotal role of technology in enhancing disaster management through improved communication, coordination, and resource allocation. A prominent approach involves the development of mobile applications that centralize operations for rescue agencies. These applications offer features such as real-time updates, resource tracking, and integrated collaboration tools. Frameworks such as React Native and Firebase have been commonly employed in the development of such systems due to their cross-platform capabilities and ease of integration with cloud services.

Certain mobile and web-based solutions have been developed to address challenges such as information overload and shelter coordination by offering basic features like location tracking, timely notifications, and easy-to-use interfaces that can be understood by users. Additionally, the integration of emerging technologies such as big data analytics and cloud computing has further strengthened disaster response systems. These technologies support real-time data collection, predictive analysis, and efficient allocation of resources, proving beneficial in managing scenarios such as floods and emergency logistics.

Despite these advancements, key challenges remain in the implementation of disaster management technologies. Many existing systems lack proper integration of features such as centralized coordination and consistent resource tracking, which limits the ability of volunteers and rescue teams to respond effectively. Additionally, earlier solutions often do not provide reliable location tracking, timely alerts, or simple interfaces suitable for users from different language backgrounds. Other persistent issues include system interoperability, data privacy concerns, and the absence of robust offline functionality.

In addition to studies focusing on India's disaster management ecosystem, international solutions have showcased the importance of centralized coordination systems during emergencies. The European-Mediterranean Seismological Centre's (EMSC) LastQuake application demonstrated the role of real-time user-generated reports and safety tip dissemination following earthquakes, improving public awareness and reducing panic. Similarly, the Federal Emergency Management Agency (FEMA) app in the United States offers disaster alerts, shelter locations, and preparedness resources, establishing a precedent for mobile platforms supporting disaster relief efforts on a national scale. The insights from these global implementations informed several design choices in the proposed system.

III. PROPOSED SYSTEM

The proposed system is a centralized, mobile-first Disaster Response and Relief Coordination Platform designed to improve situational awareness, streamline resource distribution, and ensure effective communication during emergencies. Developed using React Native for cross-platform deployment and Appwrite as the backend-as-a-service (BaaS), the application enables interaction between victims, volunteers, NGOs, and administrators. By integrating real-time updates, role-based access, geolocation features, and aid management functionalities, the system addresses key operational challenges in disaster scenarios.

A. Functional Capabilities

1) User Roles and Authentication: Users must register by selecting a designated role—Victim, Volunteer, NGO, or Admin. Each registration undergoes admin approval before access is granted, ensuring only verified users can interact with the platform. The system provides secure login, logout, and password recovery mechanisms to protect user accounts and maintain data confidentiality.

2) User Roles and Permissions: The platform implements distinct user roles with specific permissions and functionalities:

- The **Victim** role allows disaster-affected individuals to register, send emergency alerts with geolocation to admins, volunteers, and NGOs, access preparedness guidelines and emergency contacts, and search for available resources like food and shelter based on type and location.
- The **Volunteer** role allows individuals to register, get verified by admins, view and post live updates on disasters, post available resources, contribute based on skills, and communicate with admins and victims for better coordination.
- The **NGO** role allows organizations to register with admin verification, post available resources like food, clothing, and medical aid, respond to emergency alerts, coordinate with volunteers and admins, and manage their resource listings.
- The **Admin** role is assigned to authorized individuals who oversee the platform's operations. They handle user registration approvals, verify NGO credentials, monitor activity, manage user access, validate and manage aid postings, and perform

system-level tasks to ensure security and smooth functioning.

This role-based access model ensures that users can only access the features and data necessary for their role, minimizing the risk of unauthorized access and enhancing system security.

3) Disaster Updates and Emergency Alerts: Volunteers can actively post and view real-time updates regarding ongoing disasters, helping to keep the community informed. In emergency situations, victims can send alerts with their geolocation, which are automatically shared with admins, volunteers, and NGOs.

4) Preparedness Guidelines: The app includes detailed preparedness information tailored to various disaster types. It provides users with emergency contact numbers, medical assistance links, and safety tips to minimize risks during crises. These resources empower users to stay informed and act responsibly in the face of disasters.

5) Aid and Resource Management: NGOs, Admins, and volunteers can post available resources like food, shelter, clothing, and medical aid. Admins verify, approve, update, or remove posts to ensure accuracy. Victims can search for aid based on type and location for faster access.

6) Disaster Assistance Chatbot: The platform integrates an AI-powered chatbot that delivers instant disaster preparedness information and emergency guidance. It assists users by answering queries about safety measures, first-aid procedures, and emergency contacts for various disaster types. The chatbot helps victims, volunteers, and NGOs access reliable, pre-defined advice, ensuring timely support even before human assistance is available.

B. Technical Solution Overview

To ensure efficiency and accuracy during disaster response, the system integrates the following technological solutions:

1) Centralized Database: All user roles and resource data are stored in a centralized backend, ensuring real-time synchronization and access across all devices.

2) Dynamic Map Interface: A live map visualizes affected zones, active alerts, and available resources. This allows for strategic resource allocation based on urgency and proximity.

3) Volunteer Coordination: Volunteers independently select where to contribute based on real-time disaster updates, resource needs, and their own availability and skills, ensuring flexible yet informed participation.

C. Operational Logic

1) Role-Based Access Control (RBAC): Custom backend logic assigns permissions and accessible modules based on user roles, enforcing operational boundaries.

2) Geolocation-Based Emergency Alerts: Victim alerts include real-time GPS coordinates, which are transmitted using geospatial APIs to authorized users.

3) Verification of NGO & Volunteer Credentials: Submitted credentials are manually or semi-automatically verified by admins prior to granting posting privileges.

4) Dynamic User Management: Admins monitor platform activity and can approve, suspend, or revoke access through status flags and action logs.

5) Search and Filter Mechanisms: Implements efficient querying and contextual filtering of disaster updates, aid listings, donation records, and volunteer data based on parameters such as category, geolocation, and timestamp.

Figure 1 illustrates the key interactions between user roles—Admin, Victim, Volunteer, and NGO—and the system. Admins manage users and aid posts; Victims send emergency alerts and seek aid; Volunteers post disaster updates and access aid; NGOs post and respond to aid requests. These interactions enable effective disaster response and coordination.

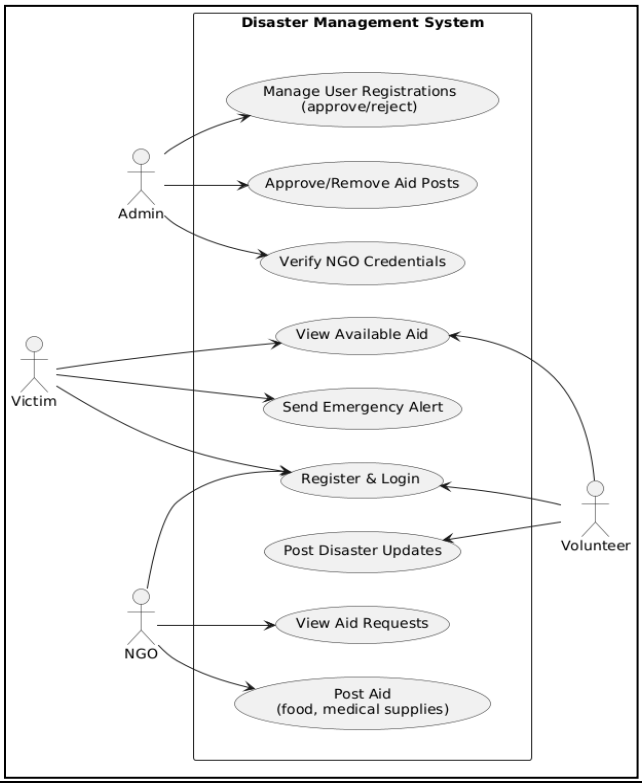


Figure 1: Use Case Diagram of the Disaster Response and Relief Coordination System

IV. SYSTEM ARCHITECTURE

The architecture of the proposed Disaster Response and Relief Coordination System adopts a modular, layered design that ensures scalability, real-time data synchronization, secure access control, and efficient management of resources and updates. The system architecture is divided into four primary layers: Frontend Interface Layer, Backend Function Layer, Appwrite Services Layer, and Deployment & Testing Layer.

A. Frontend Interface Layer: This layer, built with React Native, delivers a role-based mobile interface for Android devices. Admins manage users and aid posts; victims access resources, send alerts, and view guidelines; volunteers, and NGOs post aid and respond to updates. The interfaces are securely integrated with Appwrite's Authentication and Real-time Database for dynamic, role-specific experiences.

B. Backend Function Layer: This layer manages core application logic, handling actions like disaster updates, emergency alerts, and aid management. It features Role-Based Access Control (RBAC), geolocation tracking, search and filter for aid discovery, and alert prioritization. Serverless Appwrite Functions power backend processes, ensuring scalable and efficient execution.

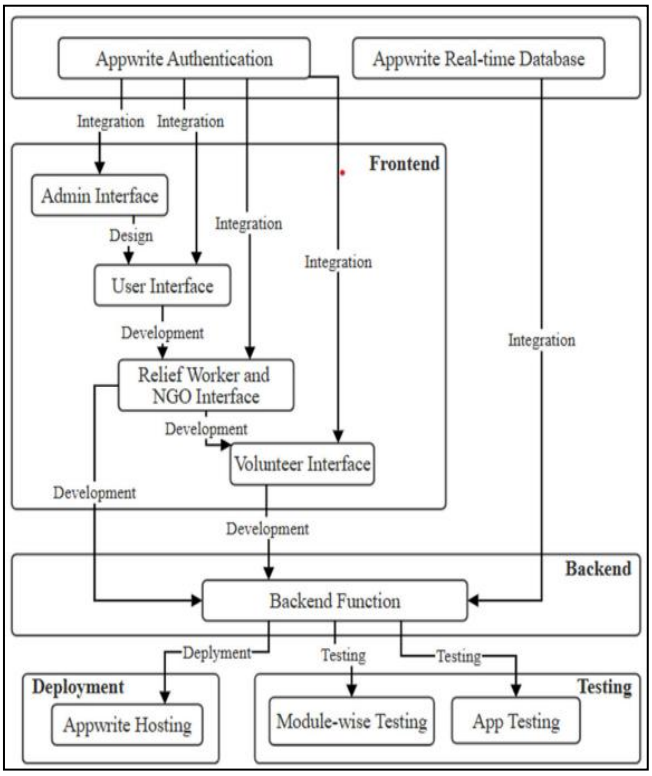


Figure 2: System Architecture

C. Appwrite Services Layer: Appwrite serves as the core backend service provider, offering essential services such as secure authentication for login, password resets, and role-based user verification. Its real-time document-based database manages users, aid posts, disaster alerts, and preparedness guidelines efficiently. The storage service handles image and media uploads related to aid and disaster events. Additionally, Appwrite Functions enable custom backend logic for tasks like user verification, alert routing, and analytics, ensuring the application remains robust, scalable, and secure.

D. Deployment & Testing Layer: This ensures system stability and performance through structured processes. Backend services are deployed using Appwrite Hosting, enabling scalable and secure backend integration. Module-wise testing is performed to validate core functionalities such as authentication, alert handling, and aid management. End-to-end testing is carried out on Android devices to ensure consistent performance, UI responsiveness, and functional integrity across user roles.

V. METHODOLOGY

The development of the Disaster Response and Relief Coordination App followed an agile and modular approach, enabling iterative testing and integration. The methodology involved setting up a secure and controlled localhost environment for both frontend and backend development before deploying to a cloud or public test environment.

1. Project Initiation: The project was initiated to overcome common issues in disaster response and relief coordination. One of the team members focused on identifying inefficiencies in existing systems and contributed to shaping a digital solution aimed at improving communication, volunteer coordination, and resource allocation. This phase included a feasibility study and setting clear project objectives based on practical disaster response needs.

2. Requirements Gathering: This phase involved in-depth research through academic papers, real-life case studies, and feedback via structured questionnaires. The goal was to understand frequent challenges like delayed communication and poor inter-agency coordination. The team documented both functional requirements (e.g., real-time alerts, role-based access, resource tracking) and non-functional requirements (e.g., scalability, performance, and interoperability).

3. System Design: The system followed a modular design approach to support future scalability and easier maintenance. Core elements included role-based user

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
VOLUME: 04 ISSUE: 04 | APRIL – 2025 DOI: 10.55041/ISJEM02860
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

management, real-time notifications, and Geographic Information System (GIS) integration for mapping disaster zones. Tools like Draw.io were used for high-level architecture diagrams, while Figma was used to create wireframes that helped align frontend and backend development efforts

4. User Interface (UI) & User Experience (UX) Design: A user-centric design approach was followed to ensure the app remains intuitive and accessible, especially during disaster situations. The interface focuses on simplicity, readability, and role-specific workflows, allowing users to quickly navigate and perform critical actions such as sending alerts, locating aid, or posting updates. Icons, color schemes, and layout structures were carefully selected to enhance clarity and minimize cognitive load.

5. Development: Development was done using Agile methodology with weekly sprints to enable iterative progress and quick feedback loops. The mobile app was built with React Native and TypeScript, ensuring type safety and cross-platform compatibility. Appwrite was chosen as the backend solution for its integrated features like authentication, database, storage, and real-time capabilities. Git and GitHub were used for version control and team collaboration.

6. Testing and Validation: Testing involved multiple layers including unit tests, integration tests, and user acceptance testing (UAT), simulating real-life disaster situations to ensure the app's reliability and usability. API endpoints and database operations were tested using Postman to validate data integrity and request-response handling. These efforts helped identify and fix critical issues, ensuring the system was stable and production-ready.

VI. RESULT

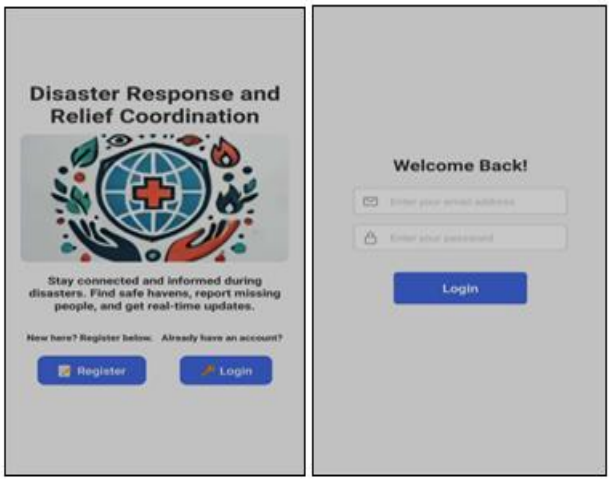


Figure 3.1: User Authentication and Onboarding Screen

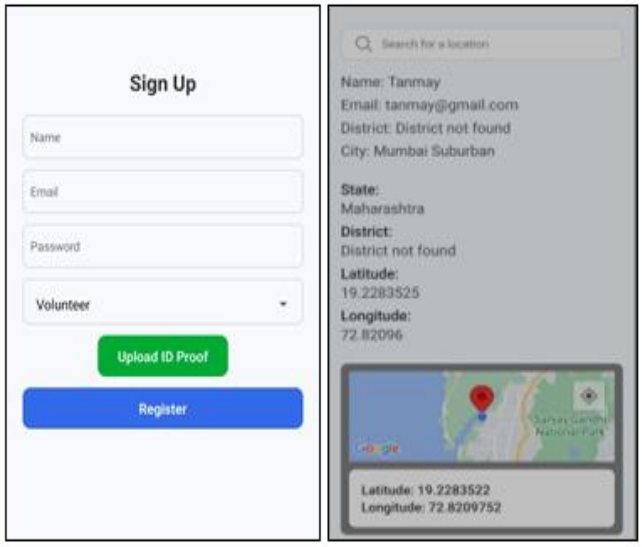
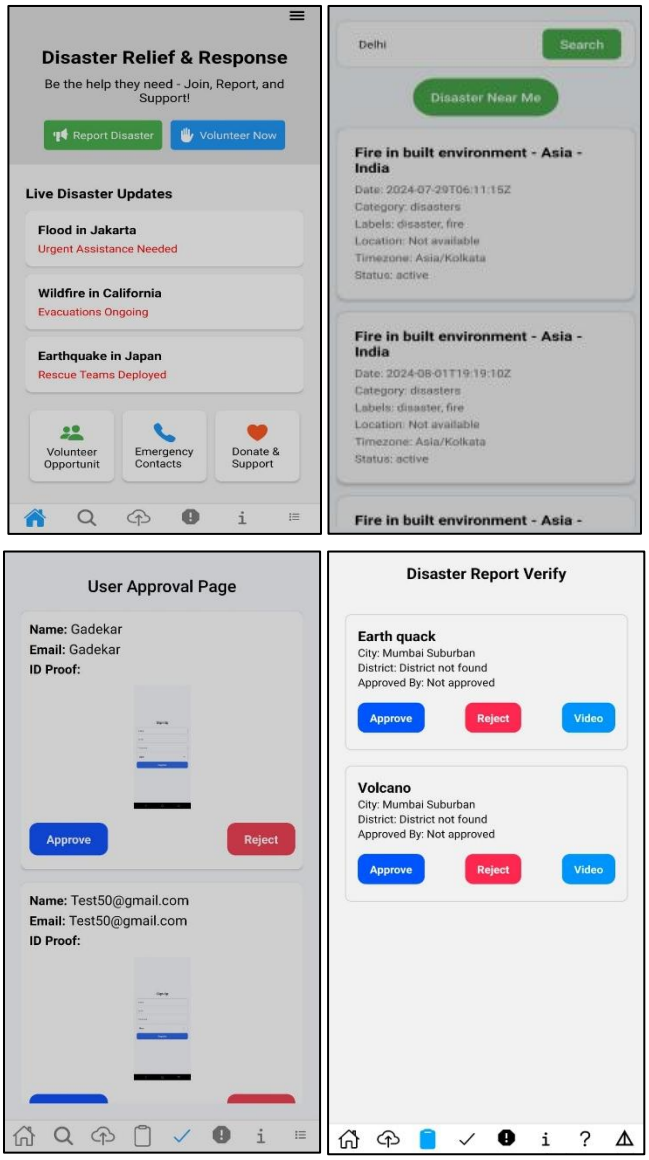


Figure 3.1: User Registration, and Location Access Interface



INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
VOLUME: 04 ISSUE: 04 | APRIL – 2025 DOI: 10.55041/ISJEM02860
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

Figure 3.2: Home Dashboard Overview and Admin Control Interface

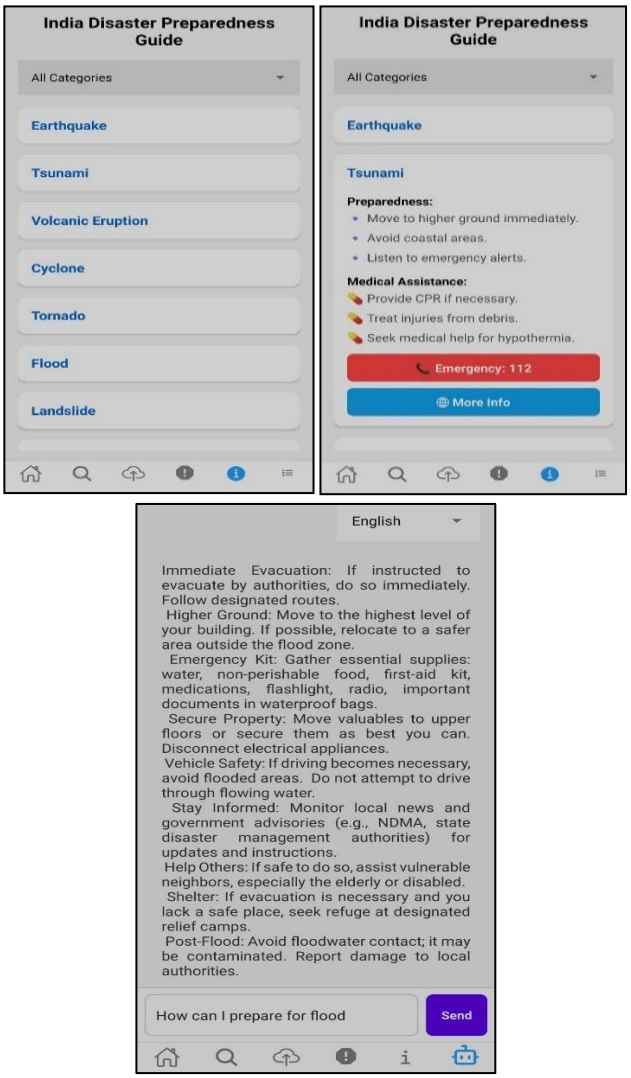


Figure 3.3: The interface showing disaster preparedness guidelines and an AI-powered chatbot providing emergency advice to assist users during disaster scenarios.

VII. INSIGHTS AND SYSTEM ADVANCEMENTS

The Disaster Response and Relief Coordination System presented in this study demonstrates a structured approach to enhancing situational awareness, resource allocation, and communication efficiency during disaster scenarios. The system's modular architecture, role-based access control, geolocation-enabled alerts, and real-time resource updates contribute significantly to improving operational coordination among victims, volunteers, NGOs, and administrators.

However, real-world disaster environments present certain practical challenges that were identified during the system design and simulated testing phases. One of the primary challenges is the dependency on stable network connectivity, which can be severely disrupted during natural calamities due to infrastructure failures or network congestion. To address this, future enhancements will focus on implementing offline-first data caching strategies and asynchronous update mechanisms, ensuring the system retains essential functionalities such as disaster guidelines, emergency contacts, and pre-fetched resource listings, even in the absence of continuous connectivity.

Furthermore, the system's current design assumes users possess a basic level of digital literacy and familiarity with mobile applications. In real-world scenarios, disaster-affected populations often include individuals from diverse age groups, varying educational backgrounds, and differing levels of technical proficiency. Recognizing this, future development will focus on enhancing accessibility by introducing simplified user interfaces, multilingual content support, visual guidance cues, and voice-assisted navigation to ensure the platform remains usable and effective for all users, including those with limited technological experience.

Finally, the current version of the system has been evaluated primarily through functional and integration testing in controlled simulated environments. While these tests confirmed the core features such as role-based authentication, emergency alert delivery, and resource coordination, real-world deployment remains a future objective. Field testing with disaster management agencies or NGOs is planned for subsequent phases, which will allow further assessment of system resilience, scalability, and usability in practical disaster scenarios.

VIII. CONCLUSION

The Disaster Response and Relief Coordination App effectively facilitates real-time disaster updates, emergency alerts, and seamless coordination among volunteers, NGOs, and affected individuals. Through its role-based user management and robust admin controls, the app ensures secure access and organized response workflows.

Key functionalities such as disaster reporting, real-time location tracking, resource management, and volunteer coordination significantly enhance the efficiency of disaster response.

By leveraging a scalable, mobile-first design with a secure backend powered by Appwrite and a responsive frontend built with React Native, the application delivers accessible and reliable functionality across devices. Overall, this system provides a powerful foundation for improving communication, coordination, and decision-making in disaster

scenarios, ultimately enabling faster and more effective relief efforts.

IX. REFERENCES

[1] Dr. Bhanumathi, Palavali Harsha Sai, Rakesh, Tribhuvan K, Uday Kiran H, "Rescue Relief Agency Management for Disaster Recovery," International Journal of Advanced Research in Science, Communication and Technology (IJARSCT) International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal Volume 4, Issue 1, May 2024

[2] Shivansh Srivastava, Rohan Kumar, "DESIGN AND IMPLEMENTATION OF DISASTER MANAGEMENT APPLICATION USING REACT-NATIVE," :04/Issue:04/April-2022.

[3] Laure Fallou, Laura Petersen, Rémy Bossu, Fréderic Roussel, "Efficiently allocating safety tips after an earthquake -lessons learned from the smartphone application LastQuake," WiPe Paper — Ethical, Legal, and Social Issues, Proceedings of the 16th ISCRAM Conference - València, Spain May 2019.

[4] Maulik Shah, Akash Shah, Viral Parmar, Neel Dedhia, Smita Bansod, "e-KrisenSeva: A Web-App based Portal for Real Time Disaster Management System," International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 03 | Mar 2019.

[5] Saad Mazhar Khan, Imran Shafi, Wasi Haider Butt, Isabel de la Torre Diez, Miguel Angel López Flores, Juan Castanedo Galán, Imran Ashraf, "A Systematic Review of Disaster Management Systems: Approaches, Challenges, and Future Directions," July 2023. Land 12(8):1514

[6] Vibhas Sukhwani, Rajib Shaw, "Operationalizing crowdsourcing through mobile applications for disaster management in India," Progress in Disaster Science Volume 5, January 2020.

[7] K. Jason, Jishnu V K, Remya G, "Web Application for Disaster Management," International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol. 8, Issue 2, February 2019.

[8] Victor James Escolano, Arlene R Caballero, Erlito Albina, Alexander A Hernandez, Ronaldo A. Juanatas, "Acceptance of Mobile Application on Disaster Preparedness: Towards Decision Intelligence in Disaster Management," Conference: 2023 8th International Conference on Business and Industrial Research At: Bangkok, Thailand, May 2023.

[9] Sangeetha.P, Divya.A, "Mobile App for Disaster Management & Information Technology in Emergency Preparedness and Response," Indo-Iranian Journal of Scientific Research (IIJSR) Peer-Reviewed Quarterly International Journal, Volume 2, Issue 2, Pages 81-85, April-June 2018.

[10] Lucía Navarro de Corcuera, María del Mar Barbero-Barrera, Ana Campos Hidalgo, Jorge Recio Martínez, "Assessment of the adequacy of mobile applications for disaster reduction," Environ Dev Sustain. 2022; 24(5): 6197–6223.

[11] Vasileios Linardos, Maria Drakaki, Panagiotis Tzionas, Yannis L. Karnavas, "Machine Learning in Disaster Management: Recent Developments in Methods and Applications", May 2022. Machine Learning and Knowledge Extraction 4(2):446-473

[12] Omar Cheikhrouhou, Anis Koubaa, Anis Zarrad, "A CloudBased Disaster Management System", Published on January 13 2020.