

Crowdsourced Disaster Relief and Assistance Platform

Bhat Geetalaxmi Jairam¹, Yajna D J², Meghana N R³, Jashwanth R V⁴, Seemith S Shetty⁵ and Tushar S Bevinamarad⁶

¹⁻⁶Department of Information Science and Engineering, The National Institute of Engineering, Mysore, Karnataka, India
Email: bgj@nie.ac.in, 2022is_yajnadj_d@nie.ac.in, megahananr@nie.ac.in, 2022is_jashwanthrv_d@nie.ac.in, 2022is_seemithsshetty_d@nie.ac.in, 2022is_tusharsbevinamarad_d@nie.ac.in

Abstract— Disaster management systems face persistent challenges such as delayed communication, inefficient coordination, and limited scalability during large-scale emergencies. The proposed Crowdsourced Disaster Relief & Assistance Platform addresses these issues by providing a resilient, cloud-enabled ecosystem that integrates victims, volunteers, NGOs, donors, and government authorities. Affected individuals can raise real-time requests embedded with geolocation data, which are automatically prioritized using AI-driven urgency detection techniques. The platform leverages scalable cloud infrastructure and real-time analytics dashboards to empower authorities with actionable insights for decision-making. Volunteers and NGOs receive instant notifications, enabling faster mobilization of resources, while donors are offered a transparent medium for targeted contributions. By combining crowdsourcing, geospatial intelligence, and AI-based prioritization, the system ensures reliability and responsiveness in disaster relief operations. This holistic framework not only accelerates emergency response but also enhances long-term disaster resilience through data-driven planning and coordination.

Index Terms— Crowdsourcing, geo-spatial intelligence, real-time analytics, emergency response, disaster resilience.

I. INTRODUCTION

Floods, earthquakes, cyclones, and pandemics frequently lead to extensive damage that overwhelms traditional aid efforts. In these situations, slow communication, missing up-to-date info, and mixed-up responses make things worse, leaving people without quick support. Traditional disaster response relies heavily on government agencies and non-government organizations, which frequently face issues such as insufficient information flow, inadequate human resources, and logistical difficulties. The increasing use of mobile devices, internet access, and cloud services necessitates a cohesive digital environment capable of enhancing rapid, data-based, and transparent disaster response strategies.

The Crowdsourced Disaster Relief & Assistance Platform enables victims to send alerts with their location, quickly identifying troubled spots. Volunteers and NGOs get immediate notifications, so they can quickly gather supplies when needed. Donors have an easy way to donate openly, helping them do their part more easily. The platform offers analytics dashboards that authorities can use to watch real-time data, check relief efforts, and make smart choices. Leveraging crowdsourcing, cloud computing, and AI-driven urgency detection, this solution guarantees scalability, reliability, and resilience in disaster management.

Traditional methods of handling disasters often face issues such as delayed communication, limited coordination, and lack of real-time monitoring, leading to ineffective relief distribution. Along with increasing mobile technologies, cloud infrastructure and crowdsourced solutions, there is a strong opportunity to develop smart platforms for emergency management. The proposed crowdsourced disaster relief and aid platform detected geolocation, real-time analytics, and AI-operated urgency to provide an integrated ecosystem for victims, volunteers, NGOs, donors and officials. This ensures rapid response, better transparency and better disaster flexibility.

II. LITERATURE WORK

In examining crowdsourced disaster relief efforts, scholarly works predominantly concentrate on creating tools and frameworks that leverage collaborative wisdom for quicker and more precise emergency management and reconstruction processes through instantaneous information exchange among volunteers. Scientists investigated various methodologies for analyzing information in crisis situations using diverse approaches such as multiple analytical tools and advanced artificial intelligence models.

In Dingqi Yang [1] 2015 publication "Providing Real-Time Assistance in Disaster Relief by Leveraging Crowdsourcing Power," he explored on-site rescue staff can largely benefit from these crowd sourcing data, due to the rapidly evolving situation at the disaster site, they usually encounter various difficulties and have requests which need to be resolved in a short time.

Nikolaos Papapenos, Claire Ellul, Amanda Shakir, Glen Hart [2] conducted research on 2018 on the publication "Exploring the use of crowdsourced geographic information in defence: challenges and opportunities," examining how geographic data are used by United Kingdom (UK) defence for purposes including peacekeeping, humanitarian aid and disaster relief, and fighting wars. The geographic extent of defence data covers the world, with greater focus directed towards areas considered to be of current interest.

Christian William Callaghan [3] (2016), conducted research on the publication "Disaster management, crowdsourced R&D and probabilistic innovation theory: Toward real time disaster response capability," introduced general agreement exists effective disaster management faces constraints related to knowledge sharing and a need for real-time research responses.

III. METHODOLOGY

The Crowdsourced Disaster Relief & Assistance Platform is structured using a three-tier architecture that emphasizes flexibility, dependability and smooth system expansion. The Frontend or Presentation Layer is developed using React.js, providing an intuitive interface where victims, volunteers, NGOs, donors, and authorities access personalized dashboards based on their roles. The Application Layer, built on Node.js, manages the core operational logic, including user authentication, request processing, and notification handling, while also supporting geographical visualization through OpenStreetMap. Finally, the Data Layer uses MongoDB to store user profiles, emergency requests, timeline updates, and system logs in a secure and scalable repository, enabling efficient data retrieval and reliable performance.

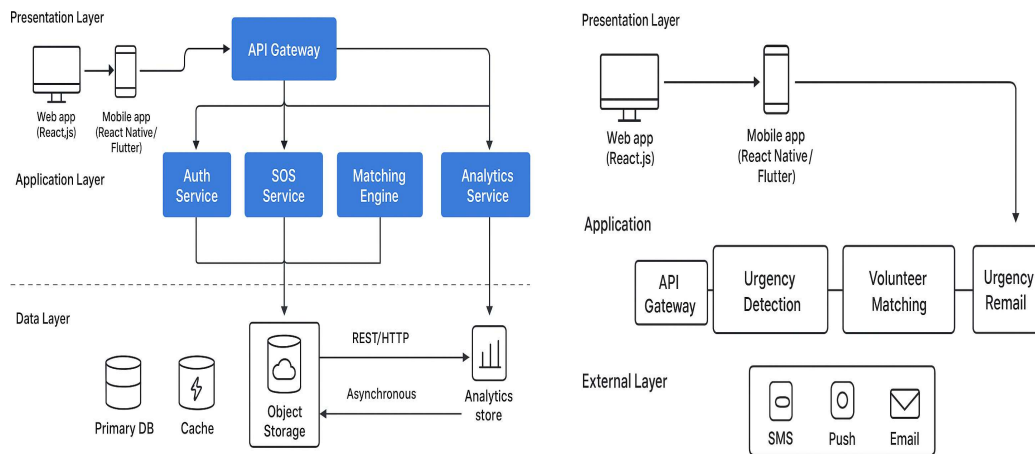


Figure 1 System architecture and workflow diagram

The system workflow follows a streamlined, event-driven sequence that connects people in need with responders and decision-makers. Users begin by registering on the platform, and once authenticated, they are granted permissions according to their specific role. Victims can create requests by selecting the type of assistance required such as food, medical support, shelter, or rescue while their live location is automatically mapped using OpenStreetMap integration. The platform then issues real-time alerts to volunteers and NGOs located nearby, and donors are informed when critical shortages or urgent needs arise. NGOs and volunteers continuously update the status of each request to reflect on-ground progress. Meanwhile, authorities access a unified dashboard that displays live disaster activity, resource deployment, and the status of all ongoing requests. This coordinated workflow enhances communication among all stakeholders and supports timely, data-informed disaster management.

IV. IMPLEMENTATION

The Crowdsourced Disaster Relief and Assistance Platform was implemented using a modular architecture that integrates a React.js-based frontend with a Node.js and Express.js backend, supported by a MongoDB database. The frontend provides role-based dashboards where users can submit requests, view live disaster maps through OpenStreetMap, and track request progress in real time. The backend functions as the central API layer, managing all business logic, authentication, request handling, role verification, and notification services. JWT-based authentication secures user sessions, while NGOs undergo an additional admin-issued access code verification process[9]. Requests are processed through dedicated services that classify urgency, match volunteers or NGOs, update timelines, and send automated notification emails when needed. All data including user profiles, requests, timelines, and logs is stored in MongoDB with structured schemas ensuring consistency and fast retrieval. The system operates through a streamlined request-response workflow where frontend actions trigger API calls, backend controllers process the requests, data is stored or retrieved from the database, and updated information is rendered instantly on the user interface[7]. The entire application was deployed and tested locally, with React and Node servers running on localhost and MongoDB hosted through a local instance, enabling a secure, responsive, and fully functional disaster assistance platform.

The implementation integrates multiple system components into a unified platform capable of supporting real-time disaster coordination. The backend manages all operational logic, including request processing, user role verification, and volunteer assignment, while the frontend focuses on delivering an intuitive interface for interacting with these services. Each request flows through a structured pipeline creation, validation, storage, status updates, and timeline recording ensuring clear traceability throughout its lifecycle. The use of OpenStreetMap enables accurate geolocation mapping without external API dependencies, supporting spatial analysis of active emergencies. All modules were tested and executed in a controlled local environment, ensuring stable communication between the frontend, backend, and database during end-to-end operations.[4]

V. RESULTS AND DISCUSSION

The implemented Crowdsourced Disaster Relief and Assistance Platform (CRISISGUARD) successfully produced a functional and interactive system capable of supporting real-time emergency coordination. The user-facing components such as the landing page, user login, NGO login, and signup interfaces which allow seamless role-based access and straightforward navigation. Users can submit requests, track their cases, and view disaster updates through an intuitive dashboard, while NGOs access a specialized interface for verifying requests, assigning volunteers, and updating delivery status. Automated email notifications for new NGO registrations demonstrate the backend's integration with SMTP services, improving verification workflows. Across all interfaces, the system ensures secure authentication using JWT tokens and maintains accurate data retrieval through MongoDB, resulting in a cohesive and reliable user experience.[12]

The platform also includes advanced features such as the Help Requests Feed, Live Map Visualization, and Request Timeline, enabling transparent and coordinated disaster response[6]. The feed displays all active assistance requests with clear details on urgency, location, and contact information, enabling volunteers or NGOs to respond instantly. The live map powered by Leaflet.js and OpenStreetMap provides spatial awareness by plotting each request as a marker with real-time updates, assisting decision-makers in identifying critical zones. The timeline module records every status transition, offering a complete historical trail from request creation to final resolution, which strengthens accountability and operational clarity. Overall, these results highlight the system's capability to integrate data visualization, role-based operations, and real-time updates to support efficient disaster management.

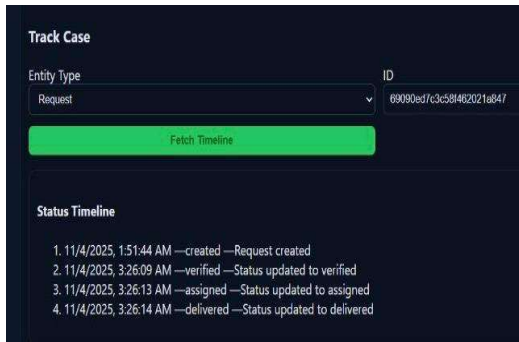


Fig 2. Screenshot 1 Request tracking timeline

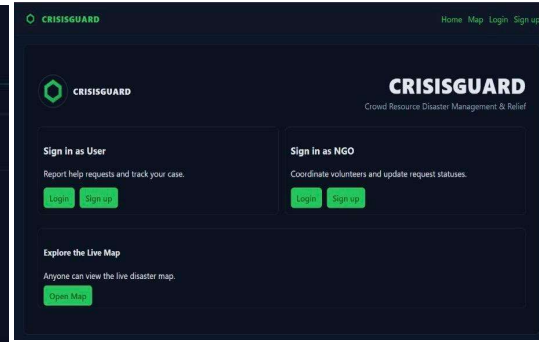


fig 3. Screenshot 2 NGO Dashboard

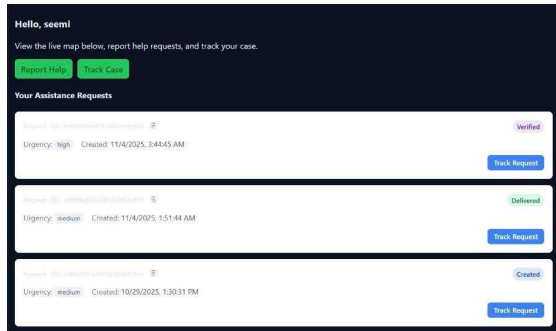


Fig 4. Screenshot 3 Automated email notification

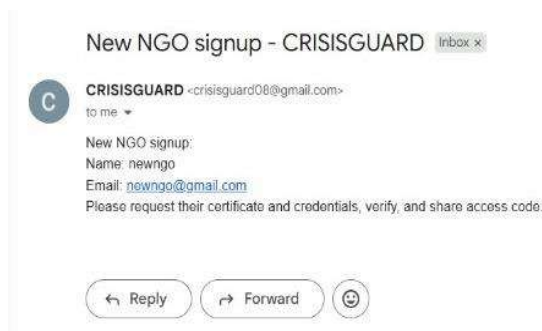


Fig 5. Screenshot 4 User dashboard

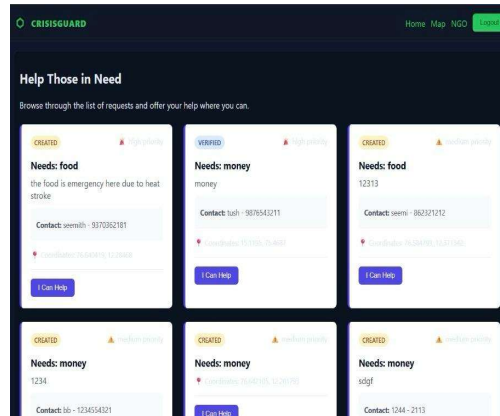


Fig 6. Screenshot 5 Live map

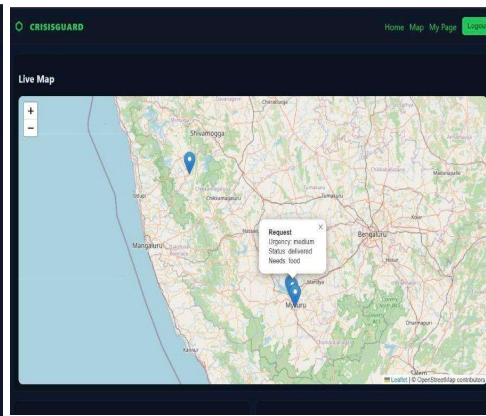


Fig 7. Screenshot 6 Help request feed

A. Discussion

The results clearly demonstrate that the system fulfills all functional and non-functional requirements outlined during the design phase. Key achievements include:

- **Seamless Integration:** The backend (Node.js/Express.js) and frontend (React) communicate effectively through RESTful APIs, ensuring smooth data exchange.
- **Real-Time Updates:** Map visualization and request timelines ensure dynamic synchronization between users and NGOs.[5]
- **Security and Authentication:** JWT-based login and access codes for NGOs provide multi-level security.
- **Automation and Notifications:** The email verification system enhances trustworthiness and operational efficiency.
- **Usability and Accessibility:** A dark-themed, responsive UI ensures optimal readability across devices.

Overall, the platform demonstrates a robust, user-centric disaster management solution, combining data visualization, secure operations, and community participation to enhance relief coordination efficiency.[11]

VI. CONCLUSION

The Crowdsourced catastrophe relief and assistance Platform become successfully designed and implemented to deal with the demanding situations confronted throughout emergency and disaster conditions with the aid of connecting sufferers, volunteers, and authorities on a unified virtual platform. The mission correctly integrates modern-day net technology to offer a actual-time, at ease, and scalable solution for crisis communique and coordination.

The gadget's architecture is constructed using Node.js and specific.js at the backend, React.js at the frontend, and MongoDB because the database, ensuring efficient statistics control and a responsive experience through the use of JWT-based totally authentication and based API integration, the platform guarantees cozy statistics alternate among customers and the server.

The platform lets in customers to register, authenticate, put up requests, view disaster updates, and get hold of timely notifications[8] government and volunteers can get right of entry to validated statistics, prioritize reaction activities, and monitor help progress correctly. The undertaking demonstrates the potential of a crowdsourced model in enhancing disaster control efficiency, decreasing conversation delays, and making sure that assist reaches affected people faster.

The final touch of this venture validates the realistic applicability of full-stack web technologies in building actual-time, network-driven applications. It additionally emphasizes the significance of software program structures that leverage collaboration, scalability, and information integrity for public welfare and emergency support. The consequences acquired via implementation and checking out verify that the device meets all its intended objectives and can be in addition extended for real-global deployment.

FUTURE WORK

While the current version of the Crowdsourced Disaster Relief and Assistance Platform fulfills its core objectives, there remains significant potential for enhancement in terms of functionality, scalability, and real-world adaptability. Future improvements can focus on integrating intelligent systems, improving accessibility, and strengthening communication between users, volunteers, and authorities. The following points highlight the key directions for future development:

A. Machine Learning-Based Alert Prediction:

Incorporating machine learning algorithms can enable predictive analysis of disasters. By analyzing past data, weather patterns, and social media inputs, the system could automatically identify potential risk zones and generate early alerts for authorities and users.

B. Mobile Application Development:

A dedicated mobile application can be developed using frameworks like React Native or Flutter. This would make the system more accessible, especially for users in disaster-affected areas where desktop access is limited, ensuring quick SOS requests and real-time notifications.

C. Blockchain-Based Data Security:

To enhance trust and transparency, blockchain technology can be used to create immutable records of SOS requests, volunteer activities, and aid distribution. This would ensure data integrity and accountability across all stakeholders.[10]

D. Scalability and Cloud Deployment:

The platform can be deployed on scalable cloud environments like AWS, Google Cloud, or Microsoft Azure. This will help handle high traffic volumes during large-scale disasters, ensuring uninterrupted service and performance reliability.

REFERENCES

- [1] Dingqi Yang, Daqing Zhang, Korbinian Frank, Patrick Robertson, Edel Jennings, Mark Roddy, Michael Lichtenstern, "Providing Real-Time Assistance in Disaster Relief by Leveraging Crowdsourcing Power," 2015.

- [2] Nikolao Papapetros, Claire Ellul, Amanda Shakir, Glen Hart," Exploring the use of crowdsourced geographic information in defense: challenges and opportunities," Crossmark, 17 December 2018.
- [3] Christian William Callaghan," Disaster management, crowdsourced R&D and probabilistic innovation theory: Toward real time disaster response capability," Elsevier,2016.
- [4] Juliana Rotich,"Ushahidi: Empowering Citizens through Crowdsourcing and Digital Data Collection", Field Actions Science Reports,2017.
- [5] C.Reuter,A. Kaufhold,"Fifteen years of social media in emergencies: A systematic review," Information Systems Frontiers, Springer, 2018.
- [6] K. Prasad, S. Singh,"GIS-Based Disaster Management and Real-Time Mapping," International Journal of Disaster Risk Reduction, Elsevier, 2019.
- [7] M. Lwin, B. Lu,"Optimized Volunteer Request Matching in Crowdsourced Crisis Response Platforms," ACM Transactions on Intelligent Systems, 2022.
- [8] Panagiotopoulos,L.Barnett,"Crisis management and crowdsourcing: A case study on digital volunteerism," Government Information Quarterly, Elsevier, 2015.
- [9] S. Saha, R. Kundu,"Secure Communication and Authentication in Web-Based Emergency Systems," IEEE Transactions on Dependable and Secure Computing, 2021.
- [10] S.Patil, V. Bhosale,"Blockchain-Based Disaster Relief and Resource Tracking Systems," Springer Communications in Computer and Information Science (CCIS), 2021.
- [11] Mark T. Riccardi," The power of crowdsourcing in disaster response operations," December 2016.
- [12] B.Yu, L. Ashworth, P. Brown,"A Real-Time Disaster Response System Using Messaging and Volunteer Coordination," IEEE Global Humanitarian Technology Conference, 2021.