CRISIS AID & RELIEF ENGINE

Ashutosh
BTech ,IT dept.
NIET,
Greater Noida, India
ashutoshkundra84@gmail.com

Shailja la
BTech, IT dept.
NIET,
Greater Noida, India
shailjalal2002@gmail.com

Kundan Saxena
BTech, IT dept.
NIET,
Greater Noida, India
er.kundansaxena@gmail.com

Akshay Mohan Srivastava
BTech, IT dept.
NIET,
Greater Noida, India
Akshaym1003@gmail.com

Ankur Kumar Varshney
Assistant professor, IT dept.
NIET,
Greater Noida, India
Ankur.varshney@niet.co.in

ABSTRACT

Disasters—both natural and man-made—demand coordinated, rapid, and resource efficient response systems. In today's digital era, technology-driven platforms have become central to effective disaster management. This research presents CARE (Crisis Aid & Relief Engine), a web-based coordination system that integrates NGOs, volunteers, donors, and administrators through a real-time, role-based interface. Building upon learnings from existing tools like Sahana, Ushahidi, and AI-integrated disaster analytics, CARE enhances situational awareness, resource mapping, and stakeholder coordination. The paper explores its technical architecture, compares it with prior models, and proposes scalable features like mobile integration, IoT inputs, and AI-driven triage—offering a roadmap for future-ready disaster management ecosystems..

KEYWORDS: *Disaster Management,Digital ,Disaster Response, Crowdsensing, Mobile Crowdsensing,IoT Sensors, AI in Disaster Zones, Disaster Response System, Disaster Relief Coordination, Crisis Aid & Relief Engine, Role-based Dashboards, NGO Relief Requests,Volunteer Management, Donor Integration, Real-time Data Cloud Computing, Cloud4BigData*

1. INTRODUCTION

Disasters, by their very nature, bring unpredictability, urgency, and often an overload of disconnected information. Traditional disaster response systems often struggle with coordination gaps, delayed resource distribution, and ineffective communication between stakeholders. Technological interventions like crowdsensing, geospatial mapping, and real-time alerting have shown promise in addressing these issues [1][2].

The **CARE (Crisis Aid & Relief Engine)** platform is designed to act as a digital bridge between multiple stakeholders—NGOs, volunteers, donors, and government bodies. Its design incorporates modular workflows, real-time dashboards, and mobile-first strategies to optimize disaster mitigation, response, and recovery.

2. LITERATURE REVIEW

2.1 Mobile Crowdsensing

Crowdsourced data from users' mobile devices has proven highly effective in enhancing situational awareness during crises. Studies [2] show that real-time reporting from ground-level volunteers can assist authorities in resource allocation, evacuation, and risk estimation.

2.2 Cloud-based IoT Monitoring

The use of IoT sensors (e.g., water levels, fire alerts) combined with cloud computing enables real-time data ingestion and decision support systems. The "Cloud4BigData" architecture exemplifies the potential of scalable cloud platforms in disaster response [3].

2.3 Artificial Intelligence in Crisis Zones

AI has been applied for damage assessment via satellite imagery, chatbot-based citizen support, and triage prediction using predictive models. Tools like Salesforce Einstein and UN OCHA's AI pilot demonstrate the shift towards data-driven response [4].

2.4 Open-Source and Civic Platforms

Open platforms such as Sahana, Ushahidi, and GDACS have pioneered digital disaster response through collaborative mapping, SMS-based reporting, and centralized emergency feeds [5]. CARE builds upon these learnings to provide a unified solution tailored for local-level execution.

3. CARE System Overview

3.1 Architectural Stack

CARE is designed using a full-stack architecture:

- **Frontend**: React + Tailwind CSS
- **Backend**: Django REST Framework
- **Database**: SQLite for development; PostgreSQL for scale
- **Hosting**: Cloud-ready with GitHub CI/CD support
- **Roles**: Admin, NGO, Donor, Volunteer

3.2 Workflow

1. NGO submits aid request with metadata (urgency, location, resource need).
2. Admin reviews and approves or rejects requests.
3. Volunteers view active tasks, apply, and get deployed.
4. Donors view approved campaigns and contribute digitally.
5. Status updates are visualized through a dashboard for each role.

4. System Features and Enhancements

4.1 Integrated Volunteer Crowdsensing

Inspired by mobile crowdsensing systems, CARE allows volunteers to report field-level updates, geotagged photos, and status flags that feed into the Admin dashboard in real-time.

4.2 AI-Assisted Damage Mapping

By integrating satellite APIs (future scope) and machine learning models, CARE can automate priority ranking of affected areas, aiding quicker decision-making.

4.3 Centralized Role-Based Dashboards

Unlike scattered response tools, CARE consolidates multi-role coordination into a single interface with dynamic data cards, filters, and live updates.

4.4 Donation Analytics and Transparency

CARE logs donor history, sends automated receipts, and optionally integrates payment gateways like Razorpay for seamless financial tracking.

5. Case Study: CARE in a Flood Simulation

In a simulated flood scenario in Bihar, India:

- CARE's IoT module recorded rising water levels.
- Affected NGOs raised aid requests via the platform.

- Admins triggered SMS alerts to registered volunteers.
- Donors contributed within minutes to specific needs.
- Real-time data was shared with civic bodies for coordination.

This simulation highlighted CARE's agility, centralized response, and user-friendly interface under pressure.

6.Future Directions

Mobile App Deployment via React Native for offline- first usage

AI-based Chat Support for citizen queries during disasters

Blockchain Integration for transparent donation tracking

Open API Support to integrate with NDMA and state bodies

SMS + USSD interface for non-smartphone access.

7. Conclusion

CARE represents a significant step toward smarter disaster coordination. By learning from global systems while being tailored to local infrastructure, CARE offers a scalable, inclusive, and tech-powered solution to disaster relief challenges. Its architecture and feature set demonstrate how full-stack engineering, community participation, and data intelligence can work in unison to save lives and optimize humanitarian response.

References

1. Khan, S. M. et al. (2023). A Systematic Review of Disaster Management Systems, Land Journal.

2. Ganti, R., & Ye, F. (2011). Mobile Crowdsensing: Current State and Future Challenges.

3. Gaire, R. et al. (2017). Cloud4BigData: IoT

Platform for Scalable Disaster Monitoring.

4. Raj, A. et al. (2024). Generative AI Applications in Disaster Risk Zones, Springer AI Review