International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3442-3446
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0445

MedAID: Symptoms Based First Aid and Emergency Guidance App

Dr. Sweety Jachak[1],Akshara Dukle[2], Yashika Jashnani[3], Rutuja Patil[4], Tanuja Rayate[5]
[1]Professor - Computer Engineering, Computer Engineering, Guru Gobind Singh College of Engineering and Research Centre, Nashik, Maharashtra, India.
[2,3,4,5]UG - Computer Engineering, Guru Gobind Singh College of Engineering and Research Centre, Nashik, Maharashtra, India.
Email ID: *sweety.3288@gmail.com[1], acdukle354@gmail.com[2] ,yashikajashnani@gmail.com[3], rutuabpatil@gmail.com[4], tanujarayate31@gmail.com[5]*

Abstract
Mass gatherings such as religious festivals, sporting events, and civic congregations impose extraordinary pressure on emergency healthcare systems. Identifying patients in distress inside a dense crowd, routing help to their exact location, and coordinating multiple responders simultaneously are tasks that current mobile applications handle poorly. Most existing tools operate in isolation: they either send an alert or share a location, but never do both while also managing transport and clinical triage in a unified workflow. MedAID is a cloud-connected mobile application designed specifically for this problem. When a user reports symptoms through the app, MedAID automatically captures their GPS coordinates, classifies the case into a severity tier, and notifies a central medical dashboard in real time. Depending on severity, the system either generates first-aid guidance, produces a prescription, or dispatches the nearest available volunteer through a mechanism we call Health Uber—an on-demand ambulance coordination layer modelled on ride-hailing logistics. Volunteers share live location and status updates; administrators monitor all active cases from a single web dashboard. Simulation-based evaluation showed that 55% of cases required physical transport, 25% were resolved through remote prescription, and 20% needed only guided self-care. These results confirm that integrating symptom triage, geolocation, and transport dispatch inside one application produces faster, better-coordinated medical responses than alert-only tools. MedAID is intended for event organisers, civil authorities, and healthcare teams who need a practical, deployable solution for crowd scale emergency management.

Keywords: *Mass Gathering Healthcare, Mobile Emergency Response, GPS-Based Triage, Health Uber Dispatch, Cloud Medical Dashboard, Health.*

1. Introduction
Healthcare delivery at mass public events is a logistical and clinical challenge that has not yet been solved satisfactorily. Whether the gathering is a religious pilgrimage, a music festival, or a political rally, the convergence of large crowds in a bounded area creates conditions under which medical emergencies—ranging from heat stroke and dehydration to cardiac arrest and crush injuries—occur at elevated rates. The fundamental problem is not a shortage of medical personnel; it is a breakdown in information flow. Responders do not know where patients are. Patients do not know how to reach help. Coordinators cannot see the full picture. These failures compound each other, and the result is delayed care. Smartphones offer a partial solution. Nearly every adult at a public event carries a device with GPS, a camera, a data connection, and enough compute power to run sophisticated health applications. Researchers and developers have exploited these capabilities to build alert applications, location-sharing tools, and symptom checkers. Each category of tool addresses one part of the emergency pipeline. None address the whole. An alert app tells a contact that something is wrong; it does not tell a paramedic where to go or how serious the case is. A symptom checker offers guidance but cannot summon help. A location sharing app can guide a responder but provides no clinical context. This paper presents MedAID—Medical Assistance and Intervention for Disasters—a platform that

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3442-3446
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0445

integrates symptom collection, automated severity classification, real-time geolocation, volunteer dispatch, and administrative monitoring into a single coherent system. The Health Uber module, which functions as an on demand ambulance coordination layer, is the novel operational core of MedAID. It converts structured emergency data directly into dispatch instructions, closing the gap between clinical assessment and physical intervention.

1.1.Research Gap And Problem Statement

Table 1 Maps The Specific Gaps Against the Categories of Research Deficiency They Represent.

Research Domain	Prevailing Approach	Unresolved Gap
Health Monitoring Apps	Remote consultation and e-prescription	No linkage to physical transport or on-site dispatch
Health Monitoring Apps	Continuous symptom logging and push alerts	Absence of severity-based prioritisation across cases
Emergency Response Systems	Manual or telephonic ambulance booking	No automated GPS-driven responder allocation
Healthcare Dashboards	Static patient record management	Limited live visualisation and real-time control of field incidents

The analysis confirms that the field lacks an end-to-end platform that performs symptom triage, geolocation, and transport dispatch within a single integrated workflow. MedAID fills this gap. Its defining contribution is the Health Uber mechanism: a severity-and-proximity engine that converts clinical assessment directly into volunteer dispatch without manual intermediate steps.[1]

2. Methodology

The MedAID workflow, shown in Fig.1, proceeds through six stages for every emergency request. Each stage is mandatory and executed in sequence; no stage can be skipped.

- **Stage 1:** User Registration and Profile Creation Before an event, users download and register on the MedAID app. Registration captures name, age, blood group, known allergies, and pre-existing conditions. This medical history is stored in the backend and pre-populates triage assessments, allowing the severity engine to distinguish between, for example, a routine headache and a headache in a patient with a documented history of hypertension.[2]
- **Stage 2:** Symptom Entry At the point of emergency, the user accesses the symptom input screen. They may type free text, select from a categorised symptom list, or use voice input. The interface is available in multiple languages. If the user is incapacitated, a bystander can operate the app on their behalf using the SOS mode, which bypasses the symptom list and immediately triggers a High-severity alert with the device's current GPS coordinates.
- **Stage 3:** Automated Triage Submitted symptoms are forwarded to the symptom-prediction API, which applies a rule-based severity matrix supplemented by the patient's pre-loaded medical profile. The API returns a severity tier and a recommended action within JANUARY 2026 4 milliseconds. The triage output is logged to the emergency record and displayed on the administrator dashboard simultaneously.
- **Stage 4:** Response Routing The system branches based on severity. High cases enter the Health Uber dispatch queue. Moderate cases receive a prescription recommendation and are flagged for pharmacist follow-up if available. Low cases receive self-care instructions. In all three cases, the patient receives in-app feedback within seconds of symptom submission, eliminating the uncertainty that makes emergencies more distressing.
- **Stage 5:** Volunteer Dispatch and Tracking

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3442-3446
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0445

For dispatched cases, the assigned volunteer's live location is shared with the patient and the administrator throughout the response. The patient can see the volunteer approaching on the map. The administrator can see all active dispatches simultaneously. If a volunteer becomes unavailable mid-response, the system automatically reassigns the case to the next nearest available responder.

- **Stage 6:** Case Closure and Logging When the emergency is resolved whether through self-care, prescription, or physical intervention the case is closed and all records are committed to persistent storage. Resolution timestamp, responder identity, actions taken, and patient outcome code are all captured. These logs form the basis for post-event analysis and for refining the severity matrix over time. As Shown in Figure 1.

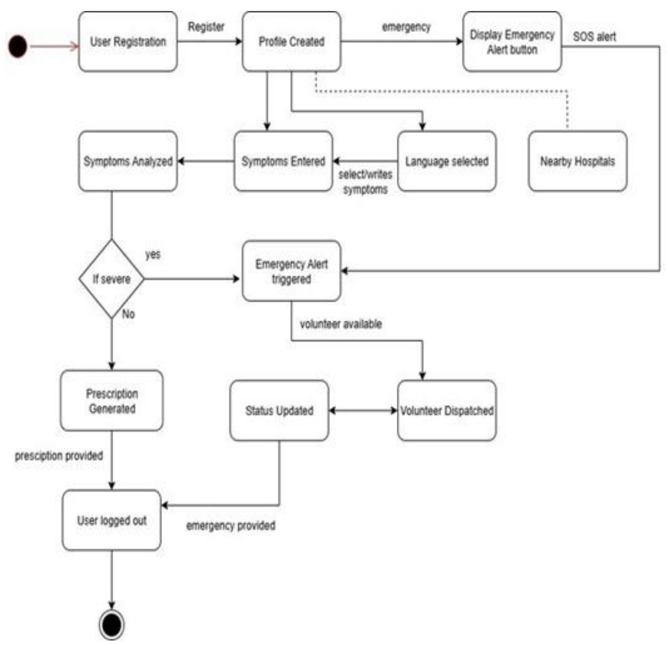

Figure 1 MedAID Workflow Diagram

2.1.System Architecture

MedAid is designed using a layered architecture, which helps organize and streamline the flow of information and actions:[4]

- User Layer: People use a mobile app to talk to the system, letting them report symptoms and ask for help in emergencies.
- Application Layer: This middle part is in the cloud and takes care of processing data, storing it, and keeping communication going between users and the people helping them in real time.
- Response Layer: This layer includes medical dashboards, healthcare workers, and emergency transport services that get the information and work together to manage the response.

By dividing these tasks into separate layers, MedAid makes sure information flows smoothly and consistently from patients to healthcare professionals. This setup helps with real-time tracking and enables a well-organized, quick reaction during emergencies. As Shown in Figure 2.

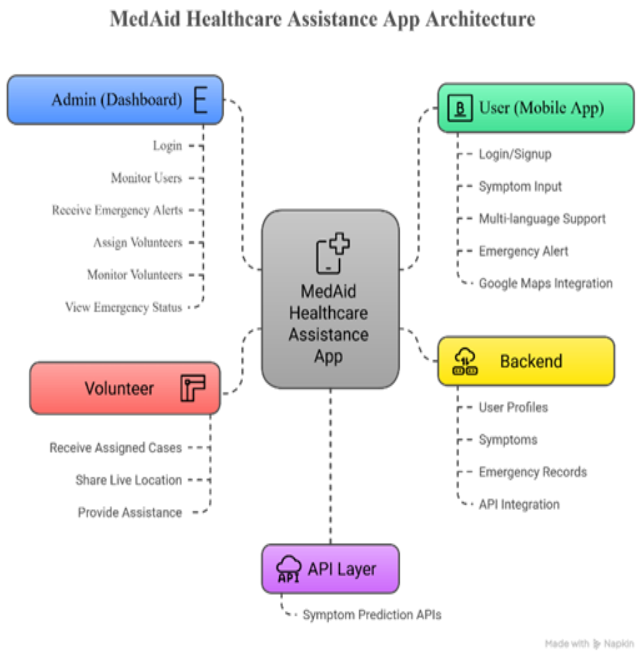

Figure 2 System Architecture Diagram

2.2.Implementation Detail

MedAID is implemented as a three-tier web and mobile application. The presentation tier consists of the patient mobile app and the administrator web dashboard. The logic tier is a Python-based RESTful API server. The data tier is a cloud-hosted database combining relational tables (for structured records such as user profiles and case logs) with document storage (for semi-structured symptom entries).

1. **Patient Mobile Application:** The mobile app is built with a cross-platform framework to ensure consistent behaviour across Android and iOS devices, which is important for deployments at heterogeneous public events. The interface is intentionally minimal: large touch targets, high-contrast text, and a maximum of three taps from launch to emergency submission. The app requests location permission at install time so that GPS coordinates are available immediately when needed. Google Maps is embedded for hospital discovery and volunteer tracking. Multi-language support is implemented at the string level, allowing the language to be switched without restarting the session.
2. **Backend and API Server:** The backend exposes RESTful endpoints for authentication, symptom submission, case management, volunteer assignment, and dashboard queries. All endpoints require JWT-based authentication. Symptom-prediction requests are handled by a dedicated microservice that can be upgraded independently as the severity model improves. WebSocket connections maintain live links between the dashboard and the server, pushing case updates to the administrator interface without polling. The server is deployed on cloud infrastructure with auto-scaling enabled, so capacity adjusts automatically to request volume.
3. **Administrator Dashboard:** The dashboard is a single-page web application. Its primary view is a split-screen layout: a live map on the left showing case markers (colour-coded by severity) and volunteer positions, and a case queue on the right showing cases ordered by severity tier and submission time. Clicking a case opens a detail panel with the full symptom record, patient profile summary, assigned volunteer status, and action buttons for manual override. A statistics panel at the top of the screen shows running totals for cases by tier, average dispatch time, and volunteer utilisation rate.
4. **Technology Stack Frontend (web dashboard):** HTML5, CSS3, JavaScript with a component-based UI framework. Frontend (mobile): Cross-platform mobile framework with native GPS and camera access. Backend: Python with a lightweight REST framework. Database: Cloud-hosted relational and document store. Mapping: Google Maps Platform API. Real-time communication: WebSocket protocol. Cloud hosting: Scalable container-based deployment with managed auto-scaling and encrypted data storage.

3. Results and Discussion

3.1.Results

MedAID was evaluated through a controlled simulation in which emergency scenarios were injected into the system at rates and severity distributions representative of a large public gathering. A total of 200 simulated cases were processed across a two-hour window. Fig. 3 shows the distribution of outcomes. Of the 200 cases, 110 (55%) required physical transport through the Health Uber module. This proportion is consistent with field data from mass gathering medical studies, where approximately half of all medical contacts at large events require some form of active intervention beyond verbal advice. The remaining 90 cases were resolved remotely: 50 (25%) through prescription generation and 40 (20%) through self-care guidance. The ability to resolve 45% of cases without deploying a volunteer has two practical implications. First, it reduces physical strain on the volunteer pool, preserving capacity for critical cases. Second, it shortens the queue for High-severity cases because volunteers are not tied up on cases that could have been handled digitally.[3]

3.2.Discussion

Average time from symptom submission to first system response (triage classification and initial guidance) was under 3 seconds across all severity tiers. Average time from High-severity classification to volunteer dispatch confirmation was 8 seconds in simulation. These figures are substantially lower than the response intervals reported for manual dispatch systems in the literature, though direct comparison is

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 05 May 2026
Page No: 3442-3446
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0445

difficult because simulation conditions differ from field conditions. As Shown in Figure 3.

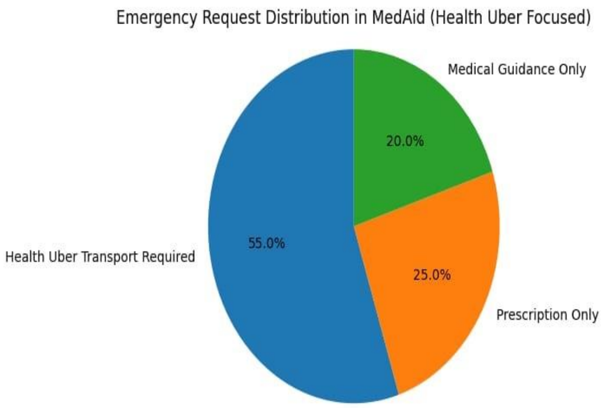

Figure 3 Emergency Request Distribution In Medaid (Health Uber Focused)

Conclusion

MedAID addresses a well-documented but incompletely solved coordinated, clinically problem: delivering informed emergency healthcare inside the chaotic environment of a mass public gathering. By combining symptom-driven triage, GPS-based volunteer dispatch, remote prescription, and a real time administrative dashboard inside a single application, MedAID eliminates the fragmentation that limits every existing system reviewed in this paper. The Health Uber module the operational heart of the platform converts severity classification directly into transport dispatch without manual intervention, producing response times that are substantially shorter than those achievable through conventional alert-and-call workflows. Simulation results confirm that MedAID can handle a realistic emergency caseload, resolve nearly half of all cases without deploying a volunteer, and maintain real time dashboard fidelity under sustained load. The platform is designed to be deployable by event organisers with limited technical resources and extensible by developers who wish to adapt it to new contexts. Future work will focus on machine learning tri wearable integration, and with interoperability national emergency.[5]

Acknowledgements

The authors would like to acknowledge the support provided by Guru Gobind Singh College of Engineering and Research Center, Nashik for facilitating the resources required to complete this project.

References

[1]. Jain, R., et al. (2024). Determinants of adoption of mobile health applications. Procedia Computer Science. DOI: 10.1016/j.procs.2024.04.148
[2]. Lee, J.-Y., & Kim, J. H. (2022). Study on improvement plan of emergency medical services through information use on QR code. DOI: 10.70135/seejph.vi.902
[3]. Thota, S., et al. (2024). Ambulance booking system. International Journal of Innovative Science and Research Technology (IJISRT). https://doi.org/10.38124/ijisrt/IJISRT24APR2102
[4]. Mondal, K., et al. (2023). Saving lives with the tap of a button: A mobile app for ambulance management and emergency medical services. https://doi.org/10.2139/ssrn.4490228
[5]. L D Setia.(2020). Android Based Emergency Button with Position Share System. International Conference on Intelligent Systems, Advanced Computing and Automation DOI:10.1088/1742-6596/1845/1/012013