HackToFuture 3.0

Syntax Squad

- Pavan Kumar H R
- Kiran Kumar
- K Jeevan Kumar
- Padmaraj Kurundwade

pavankumarhr1207@gmail.com
Alva's Institute of Engineering and Technology

RapidResQ – Emergency Response System

- RapidResQ is a smart emergency response platform.
- It connects citizens with the nearest ambulances, hospitals, and doctors in real-time.

Key Features :

- One-tap SOS alert for instant help.
- Real-time location tracking of emergency services.
- AI-powered route optimization for faster response.
- Live updates to users and responders.

RapidResQ

T1PS2

Problem Statement :

The current emergency response system faces several challenges that cause delays in medical assistance:

No Real-Time Tracking – Difficulty in locating the nearest ambulance or hospital.
Inefficient Communication – Poor coordination between citizens, responders, and hospitals.
Traffic Delays – Congested roads and poor route planning slow down emergency vehicles.

These issues result in delayed medical care, impacting patient safety and survival.

Solution Overview

The emergency response system aims to improve medical assistance by addressing key challenges through smart technology and real-time coordination.

- **One-Tap SOS:** Instantly send emergency alerts for quick response.
- **Live Tracking:** Locate nearby ambulances and hospitals in real time.
- **Fast Routes:** AI-powered navigation finds the quickest path.
- **Live Updates:** Receive real-time notifications on emergency status.
- **Medical Info Sharing:** Helps doctors provide better treatment.
- **Easy Access:** Available via app, SMS, and voice commands.
- **Collaboration with Hospitals & Govt.:** Ensures better emergency support.

Tech Stack

- Frontend : React js ,Tailwind CSS, Google Maps, API,Redux / context API
- Backend: Node.js with Express.js / Django, MySQL / MongoDB,RESTful APIs
- Real-Time Features: WebSockets / Firebase Realtime Database ,AI-based Route Optimization
- Notifications & Communication: Firebase Cloud Messaging (FCM) ,Twilio / SendGrid.
- Security: JWT / OAuth 2.0 ,HTTPS & Data Encryption

Implementation

1. **User Sends SOS Alert :**

- User taps the SOS button on the web/app.
- The request is sent to the backend server via REST API.

2 . **Backend Processes Request :**

- The server (Node.js/Express or Django) receives the request.
- It fetches the nearest ambulance/hospital using Google Maps API.
- Stores the emergency details in MySQL/MongoDB.

Implementation

3. Real-Time Tracking & Route Optimization :

- The system uses WebSockets to update live location.
- AI-based route optimization finds the fastest route for responders.

4 . Notifications & Alerts Sent

- Firebase Cloud Messaging (FCM) sends push notifications to responders.
- Twilio SMS API sends alerts to users and emergency contacts.

5. Emergency Responder Reaches User

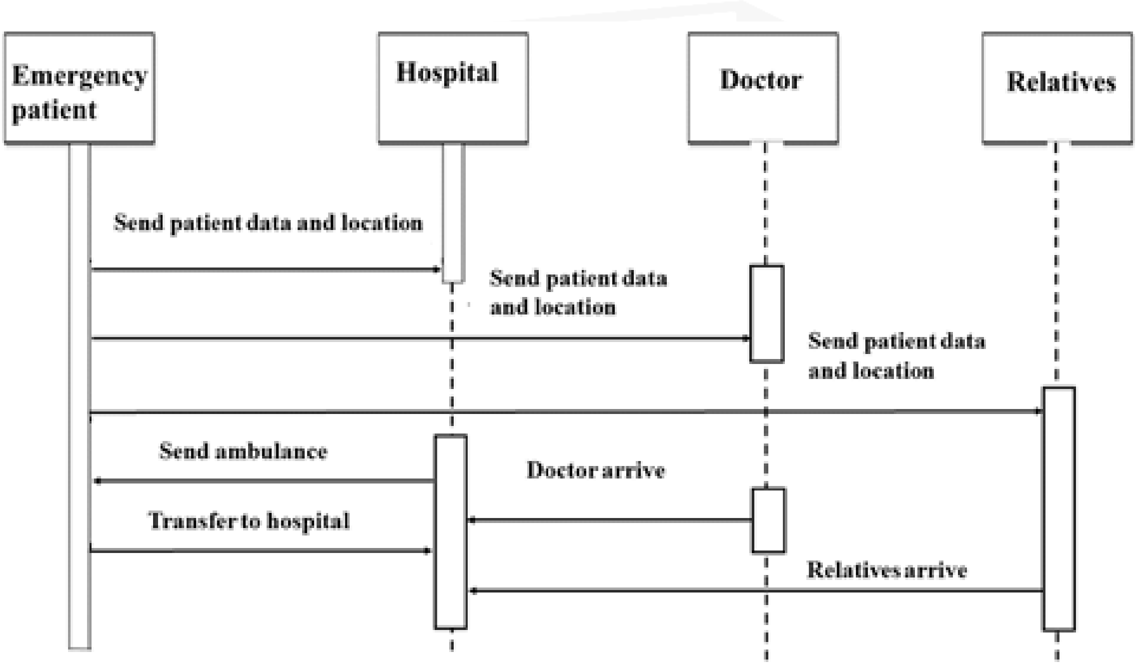
- The ambulance follows the optimized route.
- The responder can access the user’s medical history from the database.

Implementation

6 . Post-Emergency Actions

- The system logs details for future analysis.
- User and responder receive feedback forms.

Sequence Diagram



Thank You !!!