Software Requirements Specification (SRS)

Project Title: Proactive Ambulance Alert and Routing System

Prepared by: Team Mirth Makers

---

1. Introduction

1.1 Purpose

This SRS document defines the software functionalities, design constraints, and system behavior for the Proactive Ambulance Alert and Routing System. It aims to reduce emergency response time through accident detection, geolocation services, geofencing notifications, and hospital coordination, using Angular for the frontend and Spring Boot with Java for the backend.

1.2 Scope

The system detects accidents via smartphone sensors, triggers an SOS signal, and alerts emergency services. It guides ambulances through optimized routes using traffic APIs, alerts nearby vehicles using geofencing, and informs hospitals of patient arrival and status, including real-time bed availability.

1.3 Intended Audience

- Full-stack Developers
- UI/UX Designers
- Emergency Service Providers
- Healthcare IT Managers
- QA/Testers
- Technical Review Boards

1.4 Definitions and Acronyms

- **SOS**: Emergency Signal
- **ETA**: Estimated Time of Arrival
- **GPS**: Global Positioning System
- **API**: Application Programming Interface
- **FCM**: Firebase Cloud Messaging
- **JWT**: JSON Web Token
- **UI**: User Interface

---

2. Overall Description

2.1 Product Perspective

This system is a centralized application with multiple service modules. It connects mobile applications, ambulance systems, traffic control units, and hospital information systems via RESTful APIs.

2.2 Product Functions

- Detect accidents through phone sensors
- Auto-trigger SOS and notify the nearest ambulance
- Send geofenced alerts to drivers along ambulance paths
- Automatically notify hospitals with live ETA and patient data
- Provide live traffic control integration for green corridor passage
- Auto-check and update hospital bed/resource availability

2.3 User Classes and Characteristics

- **Drivers:** Receive alerts via mobile UI or system-level notifications
- **Ambulance Personnel:** Access routing and patient monitoring dashboard
- **Hospital Staff:** Get patient status updates and resource alerts
- **Admin Panel Users:** Oversee operations and manage configurations

---

3. System Features

3.1 Accident Detection

- Detect crash using gyroscope and accelerometer
- AI verification before SOS activation
- GPS coordinates, impact strength, and time sent via Spring Boot API

3.2 Emergency Notification and Ambulance Dispatch

- Nearest ambulance is located via backend logic
- Notification sent via FCM
- ETA and optimized route fetched using Google Maps API

3.3 Driver Alert System via Geofencing

- Angular backend interfaces with Google Maps to create geofences
- Alerts sent through FCM or SMS using Twilio/Nexmo
- SMS fallback for offline users

3.4 Smart Traffic Management

- REST APIs connect to traffic light systems
- Lights prioritized for ambulance route based on dynamic traffic data

3.5 Hospital Notification

- API checks hospital capacity and resource availability
- Status dashboard updates in Angular frontend
- Real-time fallback to next facility if needed

3.6 Patient Monitoring in Transit

- Patient vitals sent from ambulance equipment or mobile interface
- Data encrypted and pushed to hospital dashboard
- Family members receive tracking link and ETA

4. External Interface Requirements

4.1 User Interfaces

- **Angular Web App** for Admins and Hospital Staff
- **Mobile App (Android/iOS)** for Drivers and Ambulance Crew

4.2 Hardware Interfaces

- Smartphones with sensor suite
- Ambulance medical monitoring units

4.3 Software Interfaces

- Google Maps API
- Firebase Cloud Messaging
- Spring Boot Backend (Java)
- Twilio or Nexmo for fallback SMS

4.4 Communication Interfaces

- **HTTPS** for secure data transmission
- **WebSockets** for live ambulance updates

5. Non-Functional Requirements

- **Performance:** System responds within 3 seconds
- **Uptime:** 99.9% SLA
- **Security:** JWT, TLS encryption, role-based access control
- **Scalability:** Microservice-friendly architecture
- **Compliance:** Adheres to GDPR, HIPAA, and local data laws

6. Assumptions and Dependencies

- All users have mobile phones with GPS and internet access
- City traffic infrastructure supports external API hooks
- Hospital systems are integrated with RESTful endpoints
- Government collaboration for access to real-time traffic light systems

---

7. Appendices

- [A] Use Case Diagrams (to be developed)
- [B] Data Flow Diagrams (DFD Level 0, Level 1)
- [C] API Schemas (Spring Boot Swagger)
- [D] UI Mockups (Angular Wireframes)

End of SRS Document