**Project Title: Emergency Coordination System  
Prepared by: Janani , Jayalakshmi**  
**Date: April 2025**

**"Problem Statement 7:**

**Scenario: During emergency situations, such as natural disasters, accidents, or medical emergencies, effective coordination among first responders, victims, and support services is crucial. However, the current methods of communication and coordination often fall short, leading to delays in response, overlapping efforts, and confusion. For example, in the aftermath of a natural disaster, rescue teams might struggle to locate victims due to a lack of real-time information, while victims may have difficulty accessing the help they need. Similarly, medical emergencies may see delays in care because responders and hospitals are not fully synchronized. The chaos that often accompanies emergencies exacerbates these issues, making it clear that a more streamlined, technology-driven approach is needed."**

**Business Requirements for Emergency Coordination System**

**1. Project Overview**

**Objective:**To develop a centralized, real-time emergency coordination platform that improves communication, response time, and collaboration among victims, first responders, coordinators (dispatchers), and hospitals during crises such as natural disasters, accidents, and medical emergencies.

**2.Proposed Solution**

A web and mobile-based system enabling:

* Victims to send alerts with real-time location.
* First responders to receive assignments and update rescue statuses.
* Coordinators to assign teams and monitor incidents.
* Hospitals to prepare in advance for incoming patients.

**3. System Users (Actors)**

| **Actor** | **Description** |
| --- | --- |
| Victim | Reports emergencies and shares location. |
| First Responder | Receives incidents and updates rescue progress. |
| Dispatcher | Assigns responders and oversees coordination. |
| Hospital Staff | Prepares resources for incoming patients. |

5. **Core Features and Services**

| **Feature** | **Description** |
| --- | --- |

|  |  |
| --- | --- |
| Emergency Alert System | Victims can report emergencies with optional medical details and location. |

|  |  |
| --- | --- |
| Real-time GPS Tracking | Victims and responders share live locations (lat/lon). |

|  |  |
| --- | --- |
| Incident Management | Dispatchers view and assign responders to reported emergencies. |

|  |  |
| --- | --- |
| Rescue Progress Updates | Responders update status ("En Route", "Resolved", etc.). |

|  |  |
| --- | --- |
| Hospital Coordination | Hospitals receive patient info and prepare accordingly. |

|  |  |
| --- | --- |
| WebSocket Communication | Real-time communication across all users. |

|  |  |
| --- | --- |
| Filtering & Search | Filter victims/responders by status, location, etc. |
| 6. **Technology Stack**   | **Layer** | **Technologies** | | --- | --- | | Frontend | React.js, Tailwind CSS, Socket.io | | Backend | Spring Boot (Microservice-ready) | | Database | MySQL with Spring Data JPA | | Realtime Layer | WebSockets (via Spring + Socket.io) | | Authentication | JWT (Optional toggle for production) | | Map Services | Google Maps API / Leaflet for GPS UI |   **7. Entity Structure**  **Victim**   * id, name, location, emergencyType, medicalCondition, latitude, longitude, timestamp, status   **Responder**   * id, name, location, assignedVictimId, latitude, longitude, status   **Hospital**   * id, name, location, capacity, availableResources, status   **IncidentReport**   * id, victimId, responderId, hospitalId, status, timestamp   **8. Modules Breakdown**   * **Victim Module**: Emergency alert, location tracker, update status * **Responder Module**: View assigned case, live location, update progress * **Dispatcher Module**: Assign responder, view map, communicate * **Hospital Module**: View incidents, resource planning   **9. Future Scope**   * Mobile App Integration (Android/iOS) * AI-based resource allocation * Admin Dashboard for analytics * Disaster Simulation Mode for training * SMS and Offline alert fallback |  |