PS Code: 1440

PS Title: An application under which all rescue agencies are registered and which can display the location of other rescue relief agencies during natural/ man made calamities

Organization: Ministry of Home Affairs

Category: Software

Domain Bucket: Disaster Management

Description: To build such an application. there is need to create a central database where all rescue agencies can register their information, including their location, contact details, and areas of expertise. This information could be entered manually by agency administrators, or automated using GPS or other location tracking technologies. Once the database is populated, the application would need to be designed to display this information in an easy-to-use interface: It could include a map that shows the locations of all registered rescue agencies, along with filters that allow users to narrow down the results based on specific criteria, such as the type of disaster, the resources available, or the time since the last reported activity. In addition to displaying the locations of rescue agencies, the application could also include features for communication and collaboration. For example, agencies could send alerts or requests for assistance to each other directly through the application, or collaborate on shared resources such as medical equipment or transportation. Security and privacy would be major considerations in building such an

application. It would be important to ensure that only authorized users have access to the database, and that sensitive information such as personal contact details is protected. Overall, building an application that allows rescue agencies to coordinate their efforts and provide aid more effectively could be a valuable tool for responding to natural or man-made disasters.

IDEA:

User:

We have made an app and website through which user can login and report emergencies during any calamities.
Our app will have English, Hindi and a Local Language.
In an emergency, the app notifies the nearest rescue camp based on the user's location.
Users have the option to report emergencies even when they have no internet connectivity.
Users can view the real-time locations of nearby rescue camps on a map, so that they can ask help from them.
Notification section for him to verify the help.
Extra Features:
Quick access to emergency numbers like 100, 101, 102.
A dedicated SOS feature to send msg to friends and family while in Emergency.
Information and tips on precautionary measures for natural calamities.

Admin:

Admins can manage their own resources and availability status.
Admins' busy or available status is visible only to district-level administrators on the map.

Admins can contact different agencies for alerts or resource sharing.
Details and available resources of different agencies are visible on the map, so that they can send alert or share resources.
A section for receiving emergency cases.
A section for receiving normal cases.
Verification of Help: Admins can send notifications to users to verify that assistance has been provided.
A section for tracking and managing resolved cases. (Sort by emergency or normal)
Admins can search for different organizations or agencies.

District Authority:
District Authority can contact different agencies for alerts or resource sharing, especially for offline emergency cases.
District Authority can search for different organizations.
A section for receiving emergency cases.
A section for receiving normal cases.
He will not get the option for sending a notification to user to verify the help.
A section for tracking and managing resolved cases. (Sort by emergency or normal)

USE CASES:
User:
Emergency Reporting by Citizens and Automated Dispatch to Nearest Rescue Camp:
Use Case: A user witnesses a building collapse during an earthquake. They use the app to report the incident, specifying the type of calamity and location.

After the user reports, the app identifies the closest rescue team and notifies them.

Offline Reporting in Remote Areas:
Use Case: A user in a remote mountainous region reports a landslide through the app while offline.

Real-Time Tracking of Rescue Camps:
Use Case: During a flood, a user stranded on a rooftop checks the app to find the nearest rescue camp's location for assistance.

Access to Emergency Numbers:
Use Case: A user witnesses a car accident and uses the app to quickly find and dial the emergency services number (e.g., 100).

Precautionary Measures and Information:
Use Case: Before an approaching hurricane, a user accesses the app to learn about evacuation routes and safety precautions.

SOS Feature for Immediate Help:
Use Case: Users can send an SOS message for to family and friends during life-threatening situations.

Admin:
Resource Management by Admins:
Use Case: A fire department administrator updates the availability of firefighting equipment, ensuring it is ready for use during a fire outbreak.

Inter-Agency Communication and Collaboration:

Use Case: During a flood, the local emergency management agency contacts nearby volunteer rescue teams for assistance and resources.

Search Function for Admins & District Authorities:
Use Case: A district administrator searches for available medical facilities within their district to coordinate medical aid during a health crisis.

These use cases demonstrate how the disaster management application can serve both citizens and administrators in effectively responding to emergencies, coordinating rescue efforts, and disseminating critical information during calamities.

DEPENDENCY:
Data Collection and Population:
The development of the application relies on the availability of accurate and up-to-date data about rescue agencies, their locations, contact details, and resources. This data may need to be entered manually or integrated through automated processes.

GPS and Location Services Integration:
The application's capability to determine users' locations and rescue agency locations in real-time depends on the successful integration of GPS and location tracking technologies.

Security and Privacy Implementation:
Security measures, such as user authentication and access controls, need to be in place before sensitive information like agency contact details is entered into the database.

Inter-Agency Collaboration:
For features like resource sharing and mutual aid requests to work, rescue agencies must actively use the application and engage in collaborative efforts.

Testing and Quality Assurance:
Before deployment, the application must undergo extensive testing to identify and resolve any bugs or issues. The quality assurance process ensures that the application functions as intended.

TECH STACK:
Web Front End: HTML, CSS, JS, React
Android Front End: Kotlin
Back End: Node.js, Express.js
Database: MongoDB
Location Services: Google Maps API
Input Sanitization: Zod
Authentication and Authorization: Passport.js, JSON Web Tokens (JWT)
Real-time Communication: Socket.io
Security: HTTPS
Cloud Hosting: AWS or Azure
Version Control: Git
DevOps and Continuous Integration/Continuous Deployment (CI/CD)
Documentation and Collaboration: JIRA, or GitHub Wiki
Testing and Quality Assurance: Jest and Postman