American International University-Bangladesh (AIUB)

Department of Computer Science

Faculty of Science & Technology (FST)

DisasterGuard: Real-Time Natural Disaster Prediction and Safety Advisory App

A Software Engineering Project Submitted

By

Semester: Summer_23_24		**Section:**	**Group Number:**	
SN	Student Name	Student ID	Contribution (CO3+CO4)	**Individual Marks**
1	MD. SIFAT HOSEN	22-47001-1	20%	
2	FARJANA YESMIN OPI	22-47018-1	20%	
3	MD. ABU TOWSIF	22-47019-1	20%	
4	SABED BIN SARWAR	22-47030-1	20%	
5	MD. MASUM BILLAH NABIL	22-47031-1	20%	

The project will be Evaluated for the following Course Outcomes

CO3: *Select* appropriate software engineering models, project management roles and their associated skills for the complex software engineering project and evaluate the sustainability of developed software, taking into consideration the societal and environmental aspects	Total Marks	
Appropriate Process Model Selection and Argumentation with Evidence	[5 Marks]	
Evidence of Argumentation regarding process model selection	[5Marks]	
Analysis the impact of societal, health, safety, legal and cultural issues	[5Marks]	
Submission, Defense, Completeness, Spelling, grammar and Organization of the Project report	[5Marks]	
CO4: *Develop* project management plan to manage software engineering projects following the principles of engineering management and economic decision process	Total Marks	
Develop the project plan, its components of the proposed software products	[5Marks]	
Identify all the activities/tasks related to project management and categorize them within the WBS structure. Perform detailed effort estimation correspond with the WBS and schedule the activities with resources	[5Marks]	

Identify all the potential risks in your project and prioritize them to overcome these risk factors.	[5Marks]	

Description of Student's Contribution in the Project work

Student Name: MD. SIFAT HOSEN
Student ID: 22-47001-1
Contribution in Percentage (20%):
Contribution in the Project:

- Solution of the problem
- Functional requirements
- Activity diagram
- Process model selection, role identification and responsibilities

____

Signature of the Student

Student Name: FARJANA YESMIN OPI
Student ID: 22-47018-1
Contribution in Percentage (20%):
Contribution in the Project:

- Solution of the problem
- Functional requirements
- Use case diagram
- Process model selection, role identification and responsibilities

____

Signature of the Student

Student Name: MD. ABU TOWSIF
Student ID: 22-47019-1
Contribution in Percentage (20%):
Contribution in the Project:

- Solution of the problem
- Functional requirements
- Class diagram
- Process model selection, role identification and responsibilities

____

Signature of the Student

Student Name: SABED BIN SARWAR
Student ID: 22-47030-1
Contribution in Percentage (20%):
Contribution in the Project:

- Solution of the problem
- Functional requirements
- Sequence diagram
- Process model selection, role identification and responsibilities

____
Signature of the Student

Student Name: MD. MASUM BILLAH NABIL
Student ID: 22-47031-1
Contribution in Percentage (20%):
Contribution in the Project:

- Solution of the problem
- Functional requirements
- Activity diagram
- Process model selection, role identification and responsibilities

____
Signature of the Student

PROJECT PROPOSAL

Background to the Problem

Background Description:

Natural disasters such as earthquakes, floods, hurricanes, wildfires, and extreme weather events have always posed significant threats to human lives, property, and the environment. The increasing frequency and severity of these events, driven in part by climate change, have amplified the need for effective disaster prediction and management tools. Despite advancements in technology and forecasting methods, many existing systems fall short in providing timely and accurate predictions, comprehensive data integration, and personalized safety advice.

Root Cause of the Problem:

The root cause of this problem lies in the complexity and variability of natural disasters. Each type of disaster is influenced by a multitude of factors that interact in unpredictable ways. Climate change, for instance, is increasing global temperatures and altering weather patterns, which leads to more frequent and severe weather events. Geological activities, such as earthquakes and volcanic eruptions, are inherently difficult to predict due to the intricate dynamics of the Earth's crust. Similarly, floods and droughts are influenced by various factors including rainfall patterns, river flow rates, and soil saturation levels, all of which require sophisticated modeling to predict accurately. Human activities like urbanization and deforestation can also exacerbate the impact of natural disasters, complicating prediction efforts further.

Importance of addressing of the problem:

Addressing this problem is crucial because accurate and timely predictions can save lives by providing early warnings and allowing people to take protective actions. Natural disasters cause significant economic losses through damage to infrastructure, property, and businesses. Improved predictions can mitigate these losses by enabling better preparedness and resource allocation. Effective disaster management can reduce the environmental impact of disasters, preserving ecosystems and biodiversity. By enhancing the ability of communities to anticipate, prepare for, and respond to natural disasters, we can build more resilient societies. The interconnected nature of today's world means that the effects of natural disasters can have far-reaching consequences beyond the immediate area of impact.

Objective:

The objective of DisasterGuard is to enhance public safety by predicting natural disasters early and providing practical safety advice. This app will use advanced technology to forecast events like heat waves, hurricanes, earthquakes, floods, and wildfires, sending timely alerts and guidance to users so they can prepare and respond effectively. The solutions provided by the app include real-time data integration, AI and machine learning-driven predictions, dynamic forecasting and alerts, personalized safety advice, interactive maps and visualizations, and community support and resources.

Solution to the Problem:

Real-time data integration ensures that comprehensive information is aggregated from diverse sources, leading to more accurate assessments and timely responses. AI-driven predictions enhance accuracy by analyzing vast amounts of historical and current data to identify patterns, making the predictions more precise and adaptable. Dynamic forecasting and alerts offer immediate notifications, allowing users to take prompt action and personalize their responses based on their location and the nature of the threat. These solutions are feasible in meeting the business objective because they enhance public safety, leverage state-of-the-art technology, are scalable, engage users effectively, and align with public safety initiatives.

Reasons for Appropriateness of Solutions:

The solutions provided by DisasterGuard are appropriate for several reasons. First, the integration of real-time data from diverse sources ensures comprehensive and up-to-date information, which is crucial for accurate disaster prediction and management. This allows for immediate responses to changing situations, which is vital in disaster scenarios. Second, the use of AI for predictions enhances accuracy by analyzing large amounts of data and continuously learning from new information. This adaptability means predictions stay relevant and precise over time. Third, the app's dynamic forecasting and alert system offers real-time notifications and personalized alerts tailored to user locations, making the information more actionable and increasing user compliance.

Yes, the solutions outlined for this app are feasible in meeting the business objective. Here's why:

The proposed solutions are feasible and align with the business objectives in several ways. DisasterGuard enhances public safety by providing early and precise natural disaster predictions and actionable advice, addressing a significant need. The app leverages advanced technology, including AI, machine learning, and real-time data integration, ensuring it is capable of delivering timely alerts and guidance. The solution is scalable, meaning it can be adapted to different regions and various types of disasters, which allows for broad adoption and impact. Personalized notifications and interactive features improve user engagement, potentially attracting a wide user base and generating revenue through premium features and partnerships with emergency services and government agencies. Moreover, DisasterGuard's alignment with public safety initiatives makes it an attractive partner for emergency responders and local authorities. Thus, the app's feasibility is supported by its ability to address critical needs, its advanced technological foundation, scalability, potential for user engagement, and alignment with public safety goals.

Basic functionalities:

- Continuous monitoring of natural disaster indicator
- Provides real time updates on potential natural disasters with estimated times, locations and severity levels
- Offer short-term and long-term forecasts to help users prepare accordingly
- Offer personalized safety tips, emergency checklists and preparedness strategies to minimize the impact of disasters
- Displays interactive maps highlighting affected areas, safe zones, evacuation routes and shelter locations
- Contains a library of educational materials on natural disaster preparedness and response
- Contains critical information and safety guidelines which are accessible even without an internet connection

Deep insight on creative solution to the real-life problem:

The envisioned app integrates AI with real-time data streams to analyze and predict various calamities. Through predictive modeling and risk assessment, it offers personalized safety recommendations to users based on their location and anticipated threats. Community engagement and crowdsourced data contribute to enhancing the app's predictive capabilities, while real-time alerts and communication channels ensure timely dissemination of critical information. Continuous learning and improvement mechanisms enable the app to evolve over time, ultimately empowering individuals and communities to proactively prepare for and respond to natural disasters, potentially saving lives and mitigating the impact of such events.

Target group:

The target group for DisasterGuard includes the general public, especially those living in disaster-prone areas who need timely information and advice. It also includes emergency responders who require a tool for better coordination and information dissemination, local authorities and governments to support public safety initiatives and disaster management plans, and organizations and businesses to ensure employee safety and business continuity.

Contributions to the development of Scientific Results:

DisasterGuard contributes to the development of scientific results by integrating multidisciplinary data sources and advancing predictive analytics. It will integrate data from various sources, including meteorological data, geological data, satellite imagery, and social media feeds, providing a comprehensive view of potential natural disaster scenarios. Collaboration with meteorologists, geologists, data scientists, and emergency management experts fosters interdisciplinary research and innovation. The app will use advanced machine learning algorithms and AI to predict natural disasters like hurricanes, earthquakes, and floods, enhancing the scientific understanding of disaster patterns and improving prediction accuracy. Real-time analytics will demonstrate significant advancements in processing real-time data from sensors, satellites, and user inputs, contributing to scientific research on how real-time data can improve prediction accuracy and timeliness.

Literature Review

Existing Studies:

Several studies have examined natural disaster prediction and management, each focusing on specific aspects such as weather forecasting, seismic activity, and flood prediction.Numerous academic studies have focused on the prediction and management of natural disasters, each contributing valuable insights into specific types of disasters and methodologies. For instance, Chen et al. (2018) developed an integrated approach for flood prediction using machine learning techniques, demonstrating how real-time data integration can enhance the accuracy of flood forecasting. Similarly, McCloskey et al. (2016) investigated earthquake early warning systems, highlighting the challenges and potential of predicting seismic events in regions with high tectonic activity. Zeng et al. (2020) explored the use of AI in weather forecasting, showing significant improvements in the prediction of extreme weather events like hurricanes and heatwaves.

Existing software:

Existing software solutions, such as WeatherBug and AccuWeather, primarily focus on providing weather forecasts and severe weather alerts, but they often lack comprehensive disaster prediction capabilities and personalized safety advice. Earthquake apps like MyShake and QuakeAlertUSA provide immediate warnings but are limited to specific regions and disaster types. Flood monitoring apps such as FloodMap Mobile offer real-time data on flood risks, but their scope is often confined to specific areas or data sources.

DisasterGuard extends these existing studies by integrating a wider range of data sources, including meteorological, geological, and satellite data, into a unified platform. Unlike current systems that often focus on a single type of disaster, DisasterGuard provides a comprehensive approach to predicting multiple natural disasters. The app employs advanced AI and machine learning to enhance the accuracy and timeliness of predictions, while also offering personalized safety advice tailored to individual user locations and circumstances. This multifaceted approach not only builds on the limitations of current solutions but also provides a more holistic and user-centric model for disaster preparedness and response, thus filling a critical gap in the existing landscape of disaster management tools.

Evaluation

1. Thorough Understanding of the Need, Problem, or Opportunity

Yes, the team has demonstrated a comprehensive understanding of the critical need for enhanced natural disaster prediction and management. They have identified the increasing frequency and severity of disasters, such as hurricanes, earthquakes, and floods, which are exacerbated by climate change and other factors. The team has extensively researched the limitations of existing systems, which often lack the integration of diverse data sources and advanced predictive capabilities. Evidence of their research includes references to studies by major meteorological and geological institutions, which underscore the gaps in current predictive technologies. This understanding highlights the pressing need for a more effective and comprehensive solution, which the project aims to address.

2. Clear Objective with Relevant Benefits and Target Market

Yes, the project has a clearly defined objective, which is to improve public safety by providing accurate and timely predictions of natural disasters along with actionable safety advice. The benefits of the project are multifaceted, including enhanced preparedness, reduced risk of harm, and minimized economic losses due to improved disaster response. The target market includes individuals living in disaster-prone areas, emergency responders who require real-time data for better coordination, and local authorities responsible for public safety initiatives. Additionally, the project aims to benefit organizations and businesses by ensuring the safety of employees and continuity of operations during disasters. The clarity of the objective and the identification of relevant benefits and target audiences ensure that the project is well-aligned with the needs of its users.

3. Solutions we are going to propose to deal with the problem:

Real-time data integration ensures that comprehensive information is aggregated from diverse sources, leading to more accurate assessments and timely responses. AI-driven predictions enhance accuracy by analyzing vast amounts of historical and current data to identify patterns, making the predictions more precise and adaptable. Dynamic forecasting and alerts offer immediate notifications, allowing users to take prompt action and personalize their responses based on their location and the nature of the threat. These solutions are feasible in meeting the business objective because they enhance public safety, leverage state-of-the-art technology, are scalable, engage users effectively, and align with public safety initiatives.

4.Understandable Purpose and Basic Functionality

Yes, the project’s purpose and basic functionality are easy to understand and are clearly communicated. The main goal of the project is to predict various natural disasters and provide personalized safety recommendations, which can help users prepare and respond effectively. The app integrates real-time data from multiple sources, uses advanced AI for accurate predictions, and offers user-specific alerts and guidance. Features such as interactive maps for safe zones and evacuation routes, along with educational materials on disaster preparedness, make the app's functionality straightforward and user-friendly. This clear explanation of the project's purpose and functionalities ensures that potential users and stakeholders can easily grasp the value and operational aspects of the app.

Requirement Analysis

1. Location and Notification Access:

Functional Requirements:

1. The app shall request permission to access the user's location.
2. The app shall request permission to send push notifications.
3. There will be one popup message box requesting for the location and notifications access   from the user.
4. There will be yes/no button to communicate with this page.

Priority Level: High

Preconditions: User's device must have location and notification services enabled.

2. Software Signup:

Functional Requirements:

1. The app shall provide a signup page for new users.
2. The signup page will include Text fields for username, email, and password.
3. The app shall verify the uniqueness of the chosen username and email.
4. If the given username or email is already available in the database, then there will be one popup message saying to insert different usernames and email.
5. Upon successful signup, the user's credentials shall be stored securely in the database.

Priority Level: Medium

Preconditions: None

3. Software Sign In:

- Functional Requirements:

1. The app shall provide a sign-in page for existing users.
2. The sign-in page shall include Text fields for username and password.
3. The app shall verify the correctness of the entered credentials against the database.
4. Upon successful sign-in, the user shall be redirected to the app's home screen.

- Priority Level: Medium

- Preconditions: User must have a registered account.

4. Continue without sign in/sign up

- Functional Requirements:

1. The app shall provide a 'continue without sign in/ sign up' option to the users.

2. Users can directly go to the apps home screen by clicking on the 'continue without sign in/sign up' option.

3. It will be a text in which users will click to go to the home screen.

- **Priority Level**: Medium

- **Preconditions:** Users have already given the location and notification access.

5. Forgot Password:

- **Functional Requirements:**

1. The app shall provide a "Forgot Password" option on the sign-in page.

2. It will be a separate page users can recover their account when tey forgot their password.

3. The user shall be prompted to enter their email address.

4. The app shall send a password reset link to the user's email address.

5. Users have to click on the link sent to their email address in order to recover thier

- **Priority Level:** Medium

- **Preconditions:** User must have a registered email address.

6. Reset Password:

- **Functional Requirements:**

1. The app shall provide a page for users to reset their password.

2. The page shall include fields for entering a new password and confirming it.

3. The app shall validate the new password and update it in the database upon submission.

- **Priority Level**: Medium

- **Preconditions:** User must have received a password reset link.

7. Real-Time Temperature Monitoring:

- Functional Requirements:

1. The app shall display real-time temperature data for the user's current location.
2. The current location will be fetched from the GPS system.
3. The app shall update temperature data at regular intervals.
4. This feature will be displayed in the home screen in the app.
5. Users can also view temperature and weather of other locations and cities.

- Priority Level: High

- Preconditions: User must have granted location access.

8. Disaster Alerts and Forecast:

- Functional Requirements:

1. The app shall provide alerts for various types of natural disasters based on real-time data.
2. The color of the specific container or widget displaying the alerts will be changed according to the emergency level.
3. The app shall display forecasts for upcoming natural disasters.
4. There will be information in the alerts regarding which type of disaster it will be, the intensity level of the disaster and also contains the alert title.

- Priority Level: High

- Preconditions: None

9.Location-Based Alerts:

- Functional Requirements:

1. The app shall send alerts specific to the user's current location.

2. This will be not a notification alert but a in app alert.
3. Specific alert text/ option will be present in the app.
4. Users can click on the alert option and see upcoming / ongoing alert based on their current location.
5. The app shall use GPS technology to determine the user's location.

- Priority Level: High

- Preconditions: User must have granted location access to avail this feature.

10. Safety Guidelines and Procedure:

- Functional Requirements:

1. The app shall provide safety guidelines and procedures for various natural disasters.
2. It will provide necessary guidelines and precautions need for a specific disaster.
3. The guidelines shall include evacuation routes, emergency contacts, and safety tips.

- Priority Level: High

- Preconditions: None

11. User Reporting in Emergency:

- Functional Requirements:

1. The app shall allow users to report emergencies or hazards they encounter.
2. There will be a option of 'report emergency' inside the app. Users can click on the option
3. and a new page will be open. User can there write the emergency they faced or facing and submit to the app.
4. The app shall provide a option and a form for users to submit relevant information.

- Priority Level: Medium

- Preconditions: None

12. Multi-Language Support:

- Functional Requirements:

1. The app shall support multiple languages for user interface and content.
2. Users shall be able to select their preferred language from the app settings.
3. Multiple languages will be available in a list view.
4. Users can select their desired language from the list view language option.

- Priority Level: Medium

- Preconditions: None

13. Regular Notification Alerts:

- Functional Requirements:

1. The app shall send regular notifications for important updates and reminders.
2. Regularly the app will send notification of weather updates and forecast.
3. There will be some specific time in which notifications will be pushed to the users like in morning.
4. Users shall have the option to enable or disable these notifications in the settings.

- Priority Level: High

- Preconditions: User must have granted notification access.

14. Emergency Notifications with Guidelines:

Functional Requirements:

1. The app shall send emergency notifications with relevant safety guidelines to the user.

2. The guidelines shall be specific to the type and severity of the emergency.

3. Again this notification option can of enable or disable by the users.

- **Priority Level**: High

- **Preconditions:** None

15. Rescue Team in Emergency situation upon users request

- Functional Requirements:

1. The app shall send emergency rescue team to the user upon users request for emergency help.

2. The app will verify the users request using real time data.

3. Then the rescue team will track the location of the victim user and will go to rescue .

4. The app will track the real time location of the victim to communicate with the rescue team and victim

- **Priority Level**: High

- **Preconditions**: None

16. Real time map with accurate data of weather and natural disaster

- Functional Requirements:

1. The app shall have a real time map in which the weather condition of all over the world.

2. The users can see the real time weather conditions inside the map.

3. Using real time map, users can see and stay alert of upcoming extreme weather and disasters.

- **Priority Level**: High

- **Preconditions**: User must have given location access

17. Offline Mode for Critical Information

- Functional Requirements:

1. The app shall store essential safety information and guidelines offline so that users can view them even in offline mode.

2. Users shall have access to critical safety tips, evacuation routes, and emergency contacts without an internet connection.

3. The app shall periodically sync data when connected to the internet to ensure information is up to date.

- Priority Level: High

- Preconditions: Users must initially download the necessary data while online.

18. Community Reporting and Feedback

Functional Requirements:

1. The app shall include a feature for users to report on-ground conditions and provide feedback.

2. Users can upload photos, videos, and descriptions of local conditions, which will be shared with other users and emergency services.

3. The app shall display these reports on a map to provide real-time updates on local disaster conditions.

- Priority Level: Medium

- Preconditions: Users must grant access to their device's camera and storage for uploading media.

19. Emergency Contact List

- Functional Requirements:

1. The app shall allow users to create and manage a list of emergency contacts.
2. Users can quickly call or send a notification to their contacts during an emergency.
3. The app shall automatically notify selected contacts when a user is in a disaster-affected area.

- Priority Level: Medium

- Preconditions: Users must input and save contact information within the app.

20. Disaster Preparation Checklists

- Functional Requirements:

1. The app shall provide checklists for users to prepare for various types of natural disasters.
2. Users can customize these checklists to fit their personal needs and circumstances.
3. The app shall send reminders to users to complete preparation tasks as deadlines approach.

- Priority Level: Medium

- Preconditions: None

21. Weather and Disaster Educational Resources

- Functional Requirements:

1. The app shall include a library of educational materials on weather patterns, climate change, and disaster preparedness.

2. Users can access articles, videos, and interactive tutorials to better understand the causes and impacts of natural disasters.

- **Priority Level**: Low

- **Preconditions**: None

22. Backup and Data Sync

- **Functional Requirements:**

1. The app shall provide a feature to back up user data and settings to the cloud.

2. Users can sync their data across multiple devices for continuity and easy recovery.

3. The app shall offer options for manual and automatic backup scheduling.

- **Priority Level**: Medium

- **Preconditions**: Users must create an account for cloud backup and syncing.

23. Volunteer Coordination and Support

- **Functional Requirements:**

1. The app shall include a feature for coordinating volunteer efforts and support during disasters.

2. Users can sign up to volunteer, track available opportunities, and receive updates on volunteer needs.

3. The app shall provide a communication platform for volunteers and organizations to coordinate efforts.

- **Priority Level**: Low

- **Preconditions**: Cooperation with volunteer organizations is required.

24. Accessible Interface for Users with Disabilities

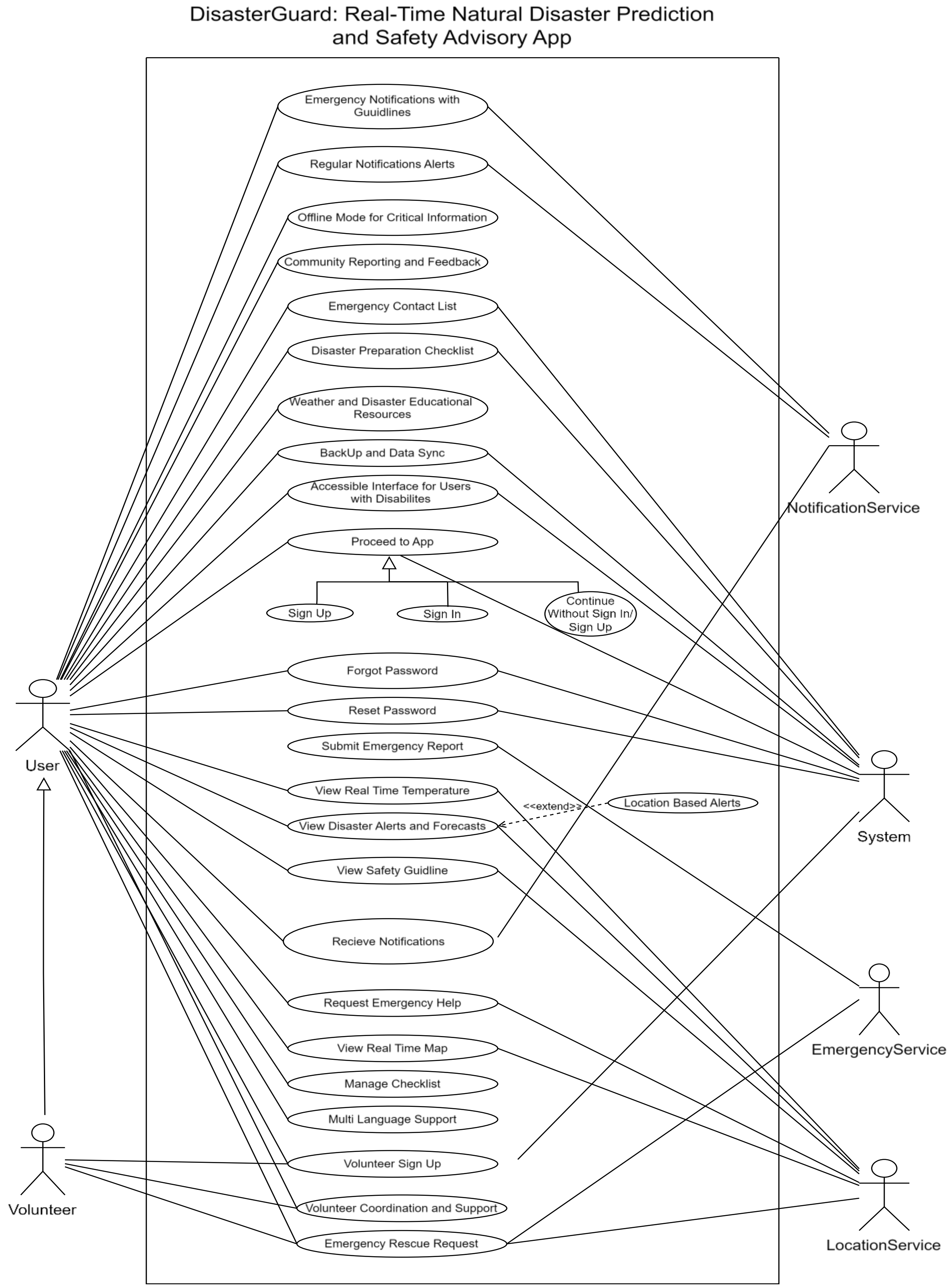
- **Functional Requirements:**

  1. The app shall include features to support users with disabilities, such as text-to-speech, high-contrast modes, and customizable font sizes.
  2. Users can enable accessibility features from the settings menu.
  3. The app shall comply with accessibility standards to ensure usability for all users.

- **Priority Level**: Medium

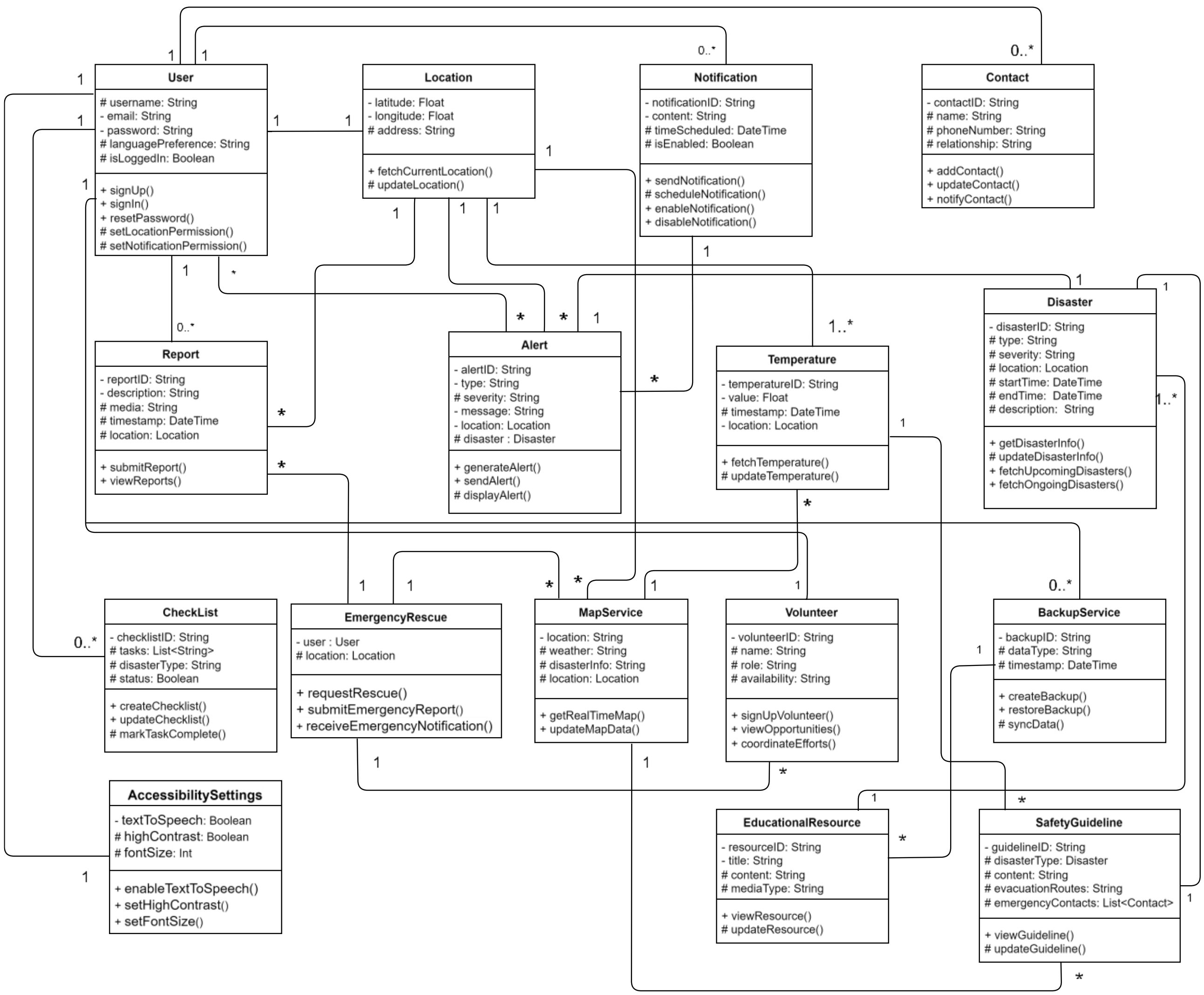
- **Preconditions**: None

Use Case Diagram:

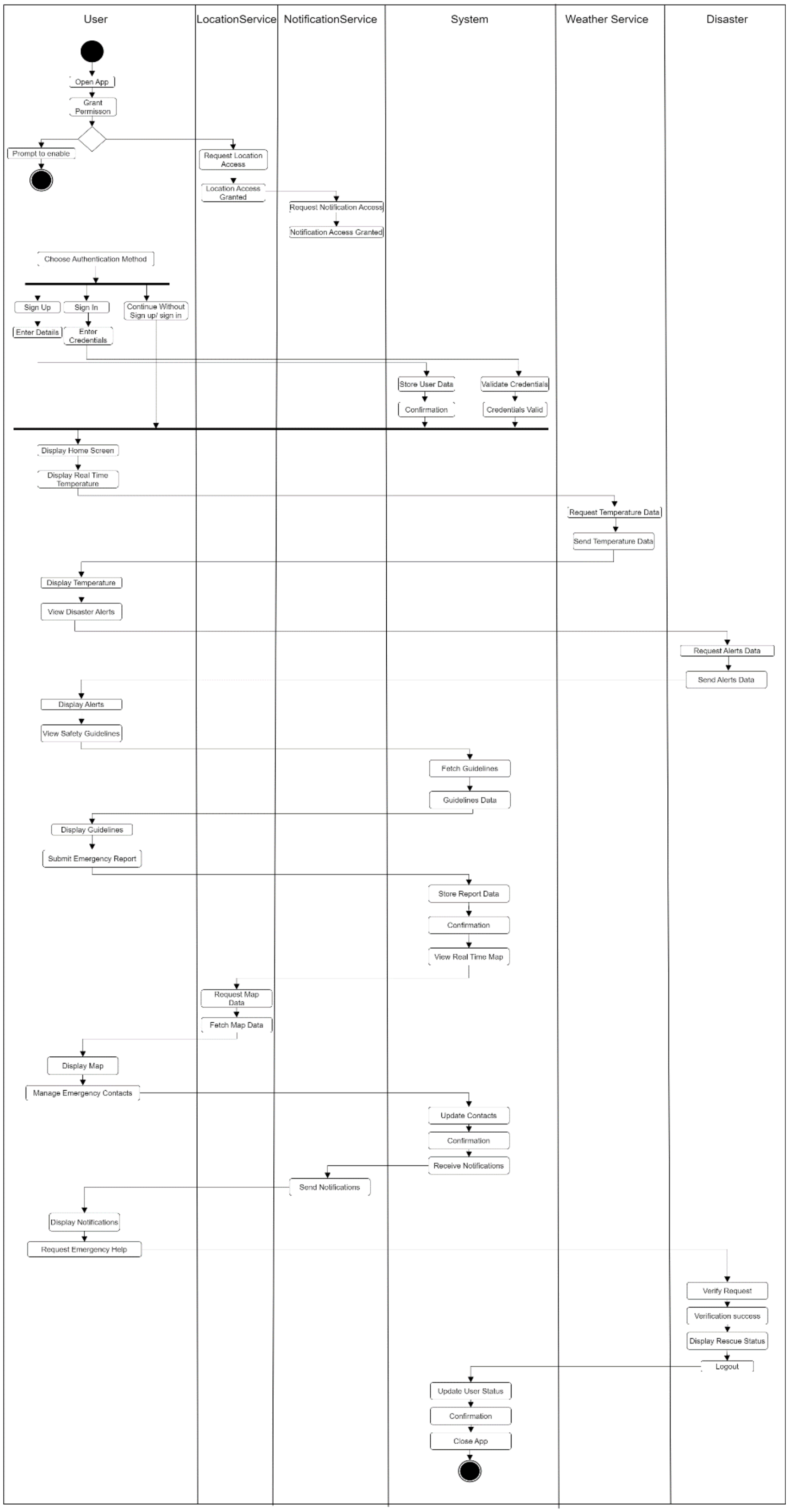


DisasterGuard: Real-Time Natural Disaster Prediction and Safety Advisory App
Emergency Notifications with Guuidlines
Regular Notifications Alerts
Offline Mode for Critical Information
Community Reporting and Feedback
Emergency Contact List
Disaster Preparation Checklist
Weather and Disaster Educational Resources
BackUp and Data Sync
Accessible Interface for Users with Disabilites
Proceed to App
Sign Up
Sign In
Continue Without Sign In/ Sign Up
Forgot Password
Reset Password
Submit Emergency Report
View Real Time Temperature
<<extend>>
Location Based Alerts
View Disaster Alerts and Forecasts
View Safety Guidline
Recieve Notifications
Request Emergency Help
View Real Time Map
Manage Checklist
Multi Language Support
Volunteer Sign Up
Volunteer Coordination and Support
Emergency Rescue Request
User
Volunteer
NotificationService
System
EmergencyService
LocationService

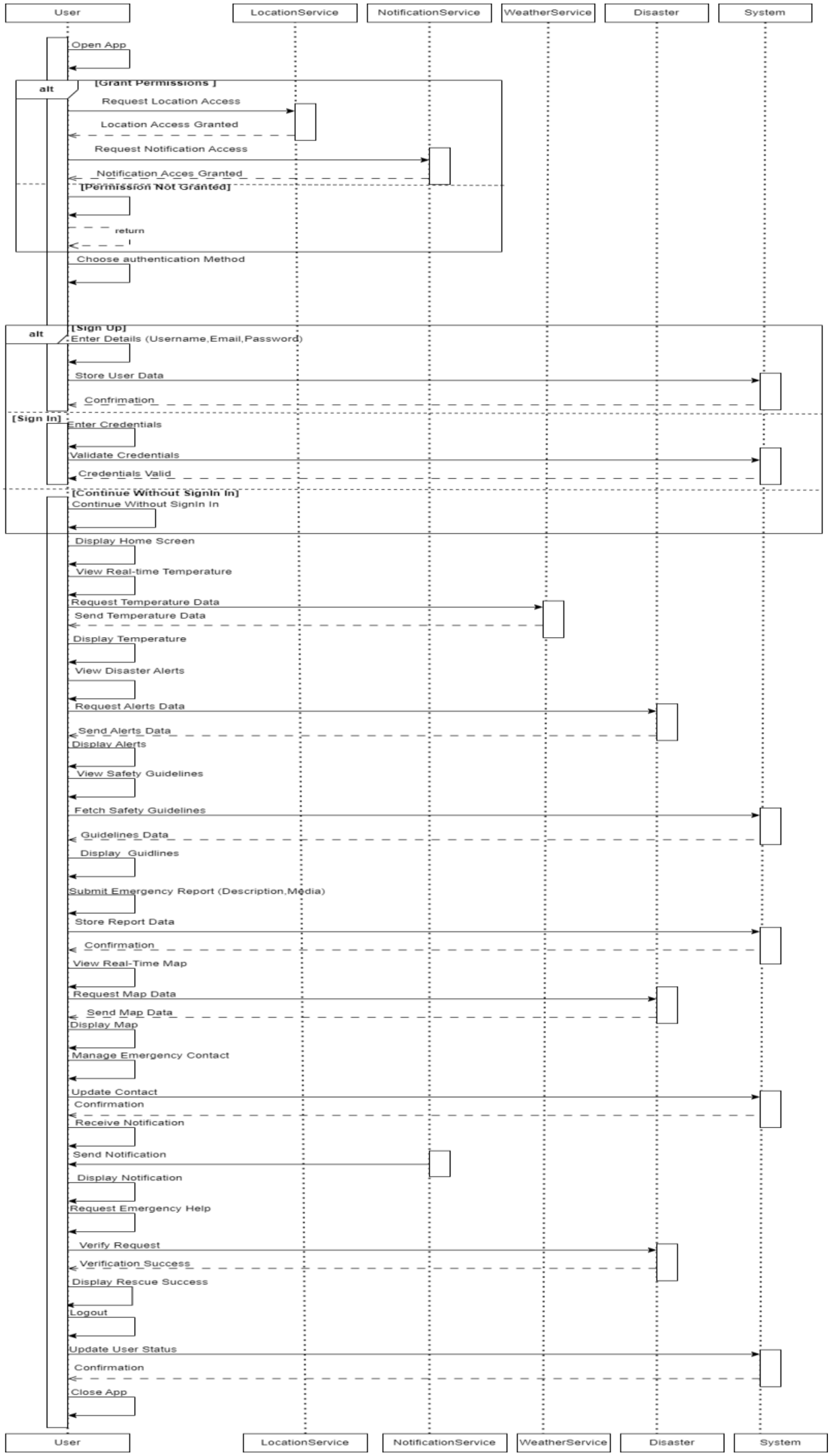
Class Diagram:


DisasterGuard: Real-Time Natural Disaster Prediction and Safety Advisory App

Activity Diagram:



Sequence Diagram:



2. SOFTWARE DEVELOPMENT LIFE CYCLE

2.1. Software Process Model

Scrum is the ideal choice for developing DisasterGuard due to its flexibility and adaptability, which are essential for an application that must quickly respond to new disaster scenarios and evolving user needs. Scrum's iterative approach allows for rapid adjustments, ensuring the app remains relevant and effective as circumstances change. Additionally, the frequent releases facilitated by Scrum's short sprints ensure that the app is continuously updated with the latest disaster information and safety guidelines, providing users with current and reliable data. This frequent updating process ensures that the app evolves in line with user expectations, as Scrum incorporates user feedback throughout the development cycle. The emphasis on user-centered development helps refine the app's functionality and usability, making it more effective in real-world applications.

Furthermore, developing DisasterGuard requires seamless coordination among developers, disaster management experts, and end-users. Scrum's focus on regular communication and collaboration ensures that all stakeholders are aligned and working towards common goals, enhancing the overall quality and effectiveness of the app. The iterative process of Scrum also facilitates continuous risk management, allowing the team to identify and address potential issues early, thus ensuring the app can handle unpredictable events effectively. High-quality standards are maintained through Scrum's emphasis on quality assurance and regular testing, ensuring each release is reliable and robust, which is crucial for an app used in emergency situations. Lastly, Scrum's product backlog prioritization ensures that the most valuable features are developed and delivered first, maximizing the benefits to users and ensuring that limited resources are used efficiently.

Why Not other process Models?

We are not using other models like Waterfall, V-Model, Incremental, or XP because they do not offer the same level of flexibility, adaptability, and customer-centric focus as Scrum. Waterfall and V-Model follow a rigid, linear approach that does not accommodate changes well once the project is in progress, making them less suitable for projects with evolving requirements like the DisasterGuard app. Incremental development, while more flexible, does not emphasize iterative learning and feedback to the same extent as Scrum. XP focuses on technical excellence but lacks the structured project management framework provided by Scrum. Scrum's iterative cycles, frequent reassessment, and emphasis on customer feedback ensure that the product remains aligned with user needs and can adapt quickly to new information or changes in the environment, making it the best choice for dynamic and complex projects like DisasterGuard.

Project Roles and Responsibilities

Product Owner

The Product Owner is a crucial role responsible for the product's success.

Responsibilities

The Product Owner defines and prioritizes the product backlog, ensuring it reflects user and stakeholder needs. They communicate the project's vision to the development team and decide on features and priorities based on feedback and business objectives. As the primary contact for stakeholders, they ensure requirements are met throughout the project.

Scrum Master

The Scrum Master ensures the team follows Scrum practices and removes obstacles.

Responsibilities:

The Scrum Master facilitates Scrum ceremonies, ensures smooth workflow by removing barriers, and guides the team in following Scrum principles. They coach the team on self-organization and continuous improvement to foster a collaborative and productive environment.

Scrum Team

The Development Team creates the product increment and collaborates to achieve sprint goals.

Responsibilities:

Their responsibilities include designing, developing, testing, and delivering product increments during each sprint, ensuring that the deliverables are of high quality. The team self-organizes to determine the best approaches for completing tasks and overcoming challenges. They actively participate in sprint planning, reviews, and retrospectives to continuously enhance the development process and meet the project's objectives.

Customers

The customers of the DisasterGuard app play a pivotal role in its development and success.

Responsibilities:

Customers provide clear requirements, participate in user testing, and give feedback throughout development. They help prioritize features based on critical needs and the app's use in disaster management. Their involvement ensures the app meets real-world needs and enhances usability and relevance.

Management

The management team overseeing the development of the DisasterGuard app holds several critical responsibilities.

Responsibilities:

Management sets clear goals, allocates resources, and maintains stakeholder communication. They oversee risk management and provide strategic guidance to ensure the app meets quality standards and user expectations. By fostering collaboration, management supports the successful delivery of the DisasterGuard app as an effective disaster management tool.

By leveraging the Scrum framework and clearly defined roles, the DisasterGuard app can be developed efficiently and effectively, ensuring timely delivery of a high-quality product that meets the dynamic needs of users during emergencies.

Socio Economic Effect of DisasterGuard

The DisasterGuard app brings significant socioeconomic benefits by improving disaster preparedness and response. Socially, it enhances community resilience by enabling quick communication and coordination during emergencies, fostering a sense of safety and solidarity among users. Economically, it reduces the financial burden on governments and individuals by minimizing damage through timely alerts and preventive measures. By facilitating faster recovery and reducing downtime, it supports economic stability in disaster-prone regions. Overall, DisasterGuard not only strengthens societal safety nets but also promotes economic resilience by mitigating losses and enabling efficient resource allocation during crises.