JU-Six-S Team - National Disaster Response System (NDRS)

Sprint 1 Retrospective Meeting Minutes

Date:	November 02, 2024
Start Time:	10:00 PM
End Time:	10:45 PM
Meeting Platform:	Discord Stream Room
Attendees:	Nasrin Akter Shimu (NA), Syeda Faria Sithi (SF), Musfikus Salihin Sifat (MS), Sunirmol Mollik (SMN), Sadman Sakib Sarkar (SS), Sovon Mallick (SM)
Meeting Chair:	Syeda Faria Sithi (SF)

Sprint 1 Evaluation

Archive Data (Past Incidents) - SS

- **Successes:**
  - Designed an intuitive UI for displaying previous incident data.
  - Built robust logic for data presentation and implemented features for admin data upload and updates.
- **Challenges:**
  - Utilized dummy data in place of a real database, which limited functionality.
  - Encountered multiple errors while running unit tests, affecting progress.
- **Improvements:**
  - Gain more experience with similar projects.
  - Regular use of time-tracking tools like Toggl and practice with unit testing frameworks to improve efficiency.

Emergency Contact Number - MS

- **Successes:**
  - Successfully implemented the Model, MainActivity, and ViewModel classes, adhering to coding standards.
  - Achieved functional contact capabilities for emergency services via the Android app.
- **Challenges:**
  - Faced difficulties with initial UI integration and unit testing.
  - Ensured consistent naming conventions across the project.

- **Improvements:**
  - Enhance comprehensive testing by integrating a real-time database.
  - Ensure clearer separation of UI and business logic for maintainability.

Manage News Board - NA

- **Successes:**
  - Developed a UI for posting new news articles.
  - Successfully generated documentation using Javadoc.
- **Challenges:**
  - Encountered difficulties retrieving all news from the database and displaying them in the UI.
- **Improvements:**
  - Focus on generating unit test cases to validate news retrieval and display functionalities.

Facilitate Training - SM

- **Successes:**
  - Successfully fetched and displayed training sessions.
  - Implemented user registration for sessions and established a functional feedback mechanism with proper data storage.
- **Challenges:**
  - Faced initial issues retrieving training materials post-registration.
  - Encountered inconsistencies in marking and logging attendance for online sessions.
- **Improvements:**
  - Improve error handling to provide specific user guidance for registration errors.
  - Optimize data loading performance to reduce wait times for training sessions and materials.

Get Guidelines about Incidents - SF

- **Successes:**
  - Developed the `getGuidelines` function to filter and retrieve guidelines based on incident type, location, and severity, enhancing usability.
- **Challenges:**
  - Used dummy data instead of a real database, resulting in null pointer exceptions during unit testing.

  - Testing and debugging took longer than anticipated due to dependencies and configuration issues.
  - Occasionally forgot to track time with Toggl.
- **Improvements:**
  - Focus on implementing robust error handling within functions like `getGuidelines`.
  - Enhance skills in writing and executing unit tests to identify issues early.
  - Learn advanced mocking techniques for testing ViewModel functions and use Toggl more consistently for time tracking.

Inform Media - SMN

- **Successes:**
  - Successfully designed and implemented the media inform functionality, enabling effective communication of important information.
  - Established a user-friendly interface for media updates, ensuring easy access for users.
- **Challenges:**
  - Faced difficulties in integrating real-time media feeds due to API limitations.
  - Encountered issues with media format compatibility, leading to delays in displaying updates.
- **Improvements:**
  - Enhance collaboration with media providers to streamline integration processes.
  - Improve error handling for unsupported media formats to ensure smoother user experience.
  - Conduct user feedback sessions to gather insights for further refining the media inform feature.

Learning Outcomes

- Developed proficiency in using GitHub for version control, including managing branches, resolving merge conflicts, and utilizing GitHub's issue tracking for seamless team collaboration.
- Enhanced task management and project tracking on Trello, leading to effective sprint planning, streamlined workflows, and clear project milestones.
- Acquired skills in creating well-structured documentation using Javadoc, generating comprehensive Software Requirements Specifications, API documentation, and user guides.

- Gained experience with testing frameworks like `JUnit` and `Mockito` for performing unit tests, simulating dependencies, and ensuring robust, reliable code quality.
- Strengthened communication and teamwork through consistent use of Discord for real-time collaboration, asynchronous problem-solving, and effective feedback sharing.
- Gained practical skills with the Model-View-ViewModel (MVVM) architecture, enabling separation of concerns and promoting clean, maintainable code.
- Mastered Android's Jetpack components, such as LiveData, ViewModel, and RecyclerView, to create responsive, interactive, and data-driven user interfaces.
- Improved Android UI design skills, adhering to Material Design principles to ensure a cohesive and visually appealing user experience.
- Learned to efficiently manage Android resources, including styles, themes, and localization, enhancing the app's versatility and user accessibility.
- Strengthened debugging and troubleshooting abilities through use of Android Studio's debugging tools, Logcat, and rigorous testing cycles.
- Built proficiency with Toggl time tracking tools to monitor productivity, set realistic goals, and analyze time allocation for better project time management and efficiency.
- Developed skills in Requirements Engineering, translating user stories into functional requirements and technical specifications for a structured development process.
- Adopted Agile development practices, including regular sprint reviews and stand-up meetings, fostering iterative improvement and adaptability.

Prepared by: Sovon Mallick
Date: November 02, 2024