Project Report: Volunteer Service Platform Application Development Integrating Jira

Fu Boyu

4IR Groupe A2

Project Overview:

This project aims to develop a service platform application based on Java Swing, designed to help volunteers, validators, and hospitalized individuals within an organization handle and respond to help requests efficiently. Using MySQL to store user data and help requests, the application integrates with the Jira platform to manage task assignments and status updates. This tool enhances the efficiency of volunteer services, ensuring that volunteers can respond quickly to the needs of hospitalized individuals.
Due to personal circumstances (repeating a year), I completed this project individually. Despite the absence of a team, I maintained strict time management and constant self-feedback to ensure the successful completion of the project. During development, I used the gei005 database to ensure data integrity and consistency.
The project is hosted on GitHub, and the complete source code can be found at the following link: [GitHub Project Link]. Task management and progress tracking were handled through the Jira platform, where I could effectively manage the project progress.

Project Goals:

1. **Provide a User-Friendly Request Submission Platform**: Allow hospitalized individuals to easily submit help requests.
2. **Volunteer Response System**: Enable volunteers to proactively respond to users' needs, ensuring quick assistance.
3. **Iterative Development**: Utilize an Agile development approach to enable rapid adjustments to user feedback and evolving requirements.
4. **User-Centered Interface Design**: Design an intuitive graphical interface to simplify interactions among users, volunteers, and validators.

Agile Project Management Approach:

The project followed Agile principles, which is particularly important given the solo nature of the development. Agile practices allowed for flexibility, iterative progress, and close collaboration with stakeholders. The following are key aspects of Agile applied throughout the project:
1. Individuals and Interactions:
Although the project was completed individually, I engaged regularly with mentors and peers for feedback. This provided valuable insights and guidance, simulating the collaborative aspect of Agile development. Every iteration was followed by a review and adjustments to the system to meet the expected quality.
2. Working Software:
Rather than spending extensive time on documentation, the focus was on delivering working software that met user needs. This approach ensured that each version of the application added functional value and could be tested with real users.
3. Customer Collaboration:

Agile emphasizes constant collaboration with customers. For this project, I closely collaborated with the target users—volunteers, validators, and hospitalized individuals. This ensured that the application closely matched their needs and that any changes were made quickly and effectively.

4. Responding to Change:

The project was designed with flexibility in mind. As requirements evolved or new features were requested, the system was adapted accordingly. For instance, adjustments were made to user interfaces and the functionality of help requests based on user feedback, making the development process dynamic and responsive to change.

Project Architecture:

1. User and Request Classes:

- **Utilisateur Class**: Stores user login credentials and profile information. Each user has a unique identifier for system authentication.
- **Demande Class**: Stores details of help requests, including type, description, and current status (e.g., pending, accepted, rejected).

2. Database Connection Class: DatabaseConnection:

This class handles communication with the MySQL database, providing methods like authenticate for user authentication. It ensures secure data retrieval and handling.

3. Graphical User Interface Classes:

- **HomeBenevolat**: Main interface for volunteers, showing all pending help requests and allowing them to accept or reject them.
- **HomeValidateur**: Main interface for validators, showing requests to be processed and enabling status updates.
- **HomeHospitalise**: Main interface for hospitalized individuals, allowing them to submit new requests and view their submitted requests.

4. Main Class: Main:

The Main class serves as the entry point to the application. It launches the authentication interface, allowing users to log in and access their respective home screens.

Key Features:

1. User Authentication:

Users must authenticate using their username and password. The system validates the credentials using the information stored in the database.

2. Help Requests:

Hospitalized individuals can submit new requests via the HomeHospitalise interface, describing the type and specifics of their needs. Volunteers in the HomeBenevolat interface can view and respond to pending requests.

3. Status Updates:

Validators can update the status of requests, either accepting or rejecting them. If rejected, they must provide a reason for the rejection.

4. Information Display:

Each user's personal information (name, email, role) and associated requests are displayed on their respective home screen.

Challenges Encountered and Solutions:

1. Slow Database Responses:

At times, the database would respond slowly, particularly during complex queries. To address this, I optimized the SQL queries by adding indexes and improving the structure of the tables, which significantly improved query performance.

2. Jira Task Deletion:

During integration with the Jira platform, tasks were accidentally deleted, leading to the loss of information and difficulties in tracking progress. To mitigate this, I

implemented better error handling and ensured regular backups of task data to prevent similar issues.

3. Incomplete Tasks:

Some tasks were not completed on time due to underestimating the complexity or the time required. I adjusted my planning process by revising task estimations and ensuring more detailed scheduling for the remaining work.

Project Results:

The project successfully met its core objectives, providing a functioning platform for managing help requests and facilitating efficient interaction between volunteers and hospitalized individuals. Despite being a solo effort, I achieved all planned features through disciplined time management and the iterative nature of Agile development.

Key Accomplishments:

- Completed user authentication and role-based access.
- Implemented help request submission and volunteer response functionality.
- Enabled task status management, allowing validators to handle requests efficiently.

Improvements and Future Steps:

While the project successfully met the initial goals, there are areas for further enhancement:

1. **User Interface Optimization**: Continuous improvements to the graphical interface, especially in terms of mobile responsiveness and usability.
2. **Notification System**: Adding real-time notifications to alert users and volunteers about task status changes.
3. **Enhanced Security**: Incorporating multi-factor authentication and stronger encryption to ensure user data protection.
4. **Performance Optimization**: As the user base grows, further optimization of database queries and backend processing will be necessary to ensure the system can handle higher loads.

Conclusion:

The project achieved significant success through the application of Agile development principles, despite being a solo effort. By focusing on flexibility, iterative improvement, and constant user feedback, I was able to create a functional, user-friendly platform for managing volunteer services. The Agile approach not only helped me maintain a steady development pace but also ensured that the system met the dynamic needs of its users.

By applying Agile techniques, I was able to adapt quickly to challenges, learn continuously, and deliver a high-quality solution that satisfied the project's requirements.

Annexes：

GitHub Project Link:
https://github.com/Sichuich/ProjectBenevolat

Jira Project Management Link:
https://boyufu.atlassian.net/jira/software/projects/PDLA/summary