Product Backlog

Client/Server Architecture

1. As an administrator, I want to deploy a server on a Linux VM to centralize the attendance data, so that the system can provide reliable service to multiple clients simultaneously.
2. As a developer, I want the client applications to interact with a centralized server for data retrieval and updates, ensuring that no major data processing is performed on the client side, to maintain system efficiency and security.
3. As an administrator, I want to ensure the data is stored in a relational database management system compatible with MySQL, to leverage existing infrastructure without incurring additional costs.

Multiple User Registration

1. As an administrator, I want to add, remove, and update user accounts through a standalone application, to manage access for faculty and students securely.
2. As a faculty member, I need to authenticate with the system to access my assigned classes and student attendance records, ensuring data privacy and integrity.
3. As a student, I need to log in to the system to view my attendance records and manage notification preferences, ensuring I receive relevant updates for my classes.

Multiple Classes and Sections

1. As a faculty member, I want to add, remove, and update classes and their sections, to accurately reflect the current academic offerings.
2. As an administrator, I want to enroll students into classes and sections, either manually or in batches, to efficiently manage class rosters.

Faculty Access

1. As a faculty member, I want to select a class to take attendance, view student names, and record attendance with a simple keypress, to minimize time spent on administrative tasks.
2. As a faculty member, I want to modify attendance records for previous days, ensuring that late arrivals or administrative errors can be corrected.
3. As a faculty member, I want to generate attendance reports showing participation rates, including adjustments for tardy students, to monitor and assess class engagement.

Student Application

1. As a student, I want to change my notification preferences for each class, so I can receive critical updates according to my needs.
2. As a student, I want to access detailed reports of my attendance, so I can track my participation and address any discrepancies promptly.

Sprint Backlog

Sprint 1: Infrastructure and User Management

Objective: Establish client/server architecture, complete user registration, and management functionalities.

Deploy and Configure Server

Task 1: Provision Linux VM on the university server.
Task 2: Install necessary software packages (e.g., MySQL).
Task 3: Set up server security (secure shell (SSH) access).

Design and Implement Database Model

Task 1: Design a relational database schema for users, classes, and attendance records.
Task 2: Implement the schema in MySQL or another compatible RDBMS.
Task 3: Test database connections and CRUD (Create, Read, Update, Delete) operations.

User Authentication System

Task 1: Define user roles and permissions (students, faculty, administrators).
Task 2: Implement user registration, login, and authentication mechanisms.
Task 3: Develop a user interface for account management (add, remove, update users).

Client-Server Communication

Task 1: Define DAO functions for data retrieval and updates.
Task 2: Implement client-side request handling and server-side response logic.
Task 3: Ensure data encryption and secure data transfer.

Sprint 2: Class Management and Attendance Functionality

Objective: Implement class and section management, attendance recording, and provide reporting and notification services.

Class and Section Management

Task 1: Develop functionalities to add, remove, and update class information.
Task 2: Implement section management within classes.
Task 3: Create a batch import feature for enrolling students using CSV files.

Attendance Recording Functionality

Task 1: Design the UI for selecting a class and recording attendance.
Task 2: Implement keypress event handling for marking attendance.
Task 3: Develop functionalities for modifying past attendance records.

Attendance Reporting Generation

Task 1: Create reports generating functionality for individual student attendance.
Task 2: Adjust participation rates in reports for students.
Task 3: Ensure faculty can easily access and generate these reports.

Student Application Development

Task 1: Develop the login and menu interface for students.
Task 2: Implement notification preference settings and management.
Task 3: Create functionalities for students to view attendance summary and export detailed reports.

Stand-Up Meeting Notes

April 3, 2023

Discussion:
Can & Jiazheng reported starting on client and server connection tasks, targeting completion by April 4.
Xinyi took responsibility for setting up the database, with an April 4 deadline.
Aoli was assigned to modify the user system, including the administrator functions, also due April 4.

Action Items:
All members focus on initial setup tasks and report any roadblocks immediately.

April 4, 2023

Discussion:
Can & Jiazheng confirmed completion of client and server connection setup.
Xinyi reported early completion of the database setup on April 3.
Aoli completed modifications to the user system as scheduled.
Next tasks were distributed: Aoli to work on userOperator and DAO tasks; Xinyi on CourseDAO, SessionDAO, and others; Can on StudentTextPlayer and database integration; Jiazheng on course management app enhancements.

Action Items:

Begin work on assigned tasks for April 5 deadline.
Can and Jiazheng to start planning the course management app additions.

April 5, 2023

Discussion:
Aoli completed the userOperator sign-in task ahead of schedule and started on DAO implementations.
Xinyi finished SectionDAO, EnrollmentDAO ahead of schedule, and continued with other DAO tasks.
Can and Jiazheng proceeded with their respective tasks without reporting completion yet.

Action Items:
Aoli and Xinyi to continue with DAO tasks.
Can focus on integrating functionalities with the database.
Jiazheng to continue work on the course management app, targeting April 7 for completion.

April 6, 2023

Discussion:
Progress update on DAO tasks by Aoli and Xinyi, both on track.
Xinyi completed additional tasks related to session management and status changes.
Aoli began work on the section list for faculty.
Discussion on tasks related to generating reports and managing course enrollments, yet to be started.

Action Items:
Xinyi and Aoli to finalize their current tasks.
Can and Jiazheng to catch up on pending tasks and focus on upcoming deliverables for April 7.

April 7, 2023

Discussion:
Jiazheng completed the course management app's add/remove functionality and moved on to updates and sections.
Aoli finished the userAdmin app tasks related to adding/removing users and viewing users.
Xinyi and Aoli reported progress on their tasks, with some completions noted.
Discussion on the final stretch of the sprint, focusing on report generation and course management functionalities.

Action Items:
All team members prioritize remaining tasks, aiming for completion by the sprint's end on April 11.

Prepare for the final push, ensuring all functionalities are integrated and tested.

April 9, 2023

Discussion:

Xinyi finished tasks on getting the section list for students and generating reports in JSON/XML formats. Also fixed the changeStatus functionality.

Can needs to work on sending JSON and XML files to clients. This is crucial for enabling report functionality.

Jiazheng is focusing on course management tasks, including enrolling and deleting students in sections. Details on progress are awaited.

Action Items:

Can prioritize sending report files to ensure functionality is tested and integrated.

Jiazheng to update the team on course management tasks and push for completion.

Xinyi and Aoli to start on documentation tasks immediately.