Product Backlog for Evo 3

JavaFX Application for Administration

1. As an administrator, I want to use a JavaFX application to manage users, so that I can add, update, and remove user accounts efficiently.
    a. Sub-task 1: Design a user-friendly interface for user account management.
    b. Sub-task 2: Implement add, update, and remove functionalities for user accounts.
    c. Sub-task 3: Integrate these functionalities with the existing database to maintain data consistency.
2. As an administrator, I want to manage student enrollments and course/section details through the JavaFX application, to streamline the administrative processes.
    a. Sub-task 1: Design interfaces for adding, updating, and removing courses and sections.
    b. Sub-task 2: Implement functionality to enroll and manage students in courses and sections.
    c. Sub-task 3: Provide tools within the GUI to facilitate batch processing for course enrollments via file uploads (e.g., CSV).

GUI for Taking Attendance

1. As a faculty member, I want a GUI-based application to take attendance easily, allowing me to focus more on teaching and less on administrative tasks.
    a. Sub-task 1: Develop a simple and intuitive interface for faculty to select a session and record attendance.
    b. Sub-task 2: Implement keypress or click event handling to mark attendance quickly.
    c. Sub-task 3: Enable modifications to past attendance records within the GUI.
2. As a faculty member, I want to access and generate attendance reports from the GUI, to monitor student participation and adjust teaching strategies accordingly.
    a. Sub-task 1: Create functionality within the GUI to generate and view attendance reports.
    b. Sub-task 2: Design the report generation feature to allow customizations based on file format.
3. As a student, I want to view my attendance records through a user-friendly GUI, so I can keep track of my participation and address any discrepancies promptly.
    a. Sub-task 1: Design the student view interface to display individual attendance records.
    b. Sub-task 2: Implement functionality for students to manage notification preferences directly from the GUI.

Sprint Backlog for Evo3

Sprint 1: JavaFX Administration Application Development

Objective: Develop and deploy a JavaFX application for administrative tasks including user, student, and class management functionalities.
Tasks:
Task 1: Design the JavaFX GUI layout for administration tasks.
Task 2: Implement functionality for adding, removing, and updating user accounts within the JavaFX application.
Task 3: Implement functionality for managing class and section details (Add/Remove/Update) within the JavaFX application.
Task 4: Integrate existing backend services with the new JavaFX GUI to ensure all administrative actions are handled properly.

Sprint 2: GUI Development for Attendance Taking

Objective: Implement a GUI for faculty to take attendance and for students to view their attendance records, suitable for both desktop and potential mobile deployment.
Tasks:
Task 1: Design and implement the GUI for faculty to select a class, view student lists, and record attendance.
Task 2: Develop functionalities within the GUI for faculty to modify past attendance records and access attendance reports.
Task 3: Create the student interface within the GUI for viewing attendance records and managing notification preferences.
Task 4: Ensure compatibility and responsiveness of the GUI for both desktop and mobile platforms.

Stand-up Meeting Notes

Pre-Sprint Planning, April 13

Administration App Development

Aoli discusses plans to handle the student add, remove, and user management functionalities. Begins preliminary design.
Jiazheng outlines the approach for student enrollment, course management, and section management. Preparation for development is underway.

Attendance App Development

Xinyi prepares for the development of the faculty side of the PC app, focusing initially on main and login screens.
Can is setting up the groundwork for the student side Android app, beginning with structural setup.
Initial assignments for documentation responsibilities are clarified, with individual tasks scheduled close to the project's completion to reflect the most current system information.

Mid-Sprint Review, April 25

Administration App Development

Aoli reports progress on student and user management features, expecting to meet the April 28 deadline. Some challenges with interface responsiveness were noted and are being addressed.
Jiazheng has completed the core functionalities for course and section management and is now focusing on final integrations and testing.

Attendance App Development

Xinyi completed the faculty side main and login by April 19 and finished the taking attendance UI on April 25. Other functionalities are in progress with an expected completion by April 27.
Can has completed the student side structure and main and login UI. He is now focusing on completing the course and function selection UI by April 27.

No specific updates; reminder of the importance of starting documentation soon to meet the April 29 deadline.

Sprint Closure and Final Review, April 28

Administration App Development

Aoli is finalizing all features today, with testing completed by the end of the day. Ready to assist with any last-minute issues.
Jiazheng confirms that all his tasks are complete, with integration testing showing stable results.

Attendance App Development

Xinyi has finalized all faculty-related functionalities.
Can is wrapping up the student side app, with final adjustments based on today's testing.

Aoli, Jiazheng, Xinyi, and Can all confirm that their respective parts of the documentation are in progress, with final versions due tomorrow. Last-minute reviews are scheduled to ensure accuracy and completeness.