

Software Requirements Specification

for

East Blue Portal

Version <1.0>

Group No.: Group 6

Tan Jun Xian	242UC2452B
Sokmaed Ong Yu Kang	242UC244V6
Lai Joon Li	242UC2452F
Muhammad Loqman Bin	1231301695
Ahmad Fariz	

Date: 23 May 2025

Table Content

1. Introduction	5
1.1. Purpose	5
1.2. Scope	6
1.3. Product Overview	7
1.3.1. Product Perspective	7
1.3.2. Product Functions	9
1.3.3. User Characteristics	10
1.3.4. Limitations	12
1.4. Definitions	13
2. References	14
3. Requirements	15
3.1. Functions	15
3.2. Performance Requirements	49
3.3. Usability Requirements	50
3.4. Interface Requirements	52
3.4.1. System Interfaces	52
3.4.2. User Interfaces	52
3.4.2.1. Student	52
3.4.2.2. Lecturer	64
3.4.2.3. Administrators	71
3.4.2.4. Parent	77
3.4.3. Hardware Interfaces	81
3.4.4. Software Interfaces	82
3.4.5. Communications Interfaces	82
3.5. Logical Database Requirements	83
3.6. Design Constraints	88
3.7. Software System Attributes	89
3.8. Supporting Information	91
3.8.1. Summary of Key Elicitation Findings	91
3.8.1.1. Type of Elicitation Used	91
3.8.1.2. Explanation of Elicitation Use	91
3.8.1.3. Benefits	92
3.8.1.4. Challenges Faced	92
3.8.1.5. Recommendations	92
3.8.2. Sample Input/Output Formats	93
4. Verification	95
4.1. Verification Approach	95
4.2. Verification Criteria	98
5. Appendices	104
5.1. Assumptions and Dependencies	104

Assumptions and Dependencies	104
5.2. Acronyms and Abbreviations	105

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Tan Jun Xian, Lai Joon Li, Sokmaed Ong Yu Kang, Muhammad Loqman Bin Ahmad Fariz	Initial complete draft of the Software Requirements Specification for East Blue Portal, outlining all proposed features and system requirements.	23/5/2025

1. Introduction

1.1. Purpose

Through the integration of the Student Information System (SIS) and the Learning Management System (LMS), the university aims to improve the efficiency of administrative procedures such as course registration, student enrolment, and academic record management. The goal of this integration is to decrease the frequency of manual data entry errors and remove redundant tasks.

In addition, the system aims to improve the user experience for students. Students can access administrative and educational resources through a single interface through to the integrated platform, which provides a smooth user interface. Students are more satisfied with the university's services overall as a result of the improved usability and convenience.

Other than students, administrative gain from the integration since it gives them access to all necessary course management resources in one place. This includes functions like grading, managing assignments, uploading course materials, and interacting with students. Effective course delivery and organisation are encouraged by this simplified system.

The university can get a complete view of student performance and engagement by combining academic activity from the LMS with student data from the SIS. By using this data for reporting and analysis, it will be easier to make well-informed decisions that will support student success programs and improve academic results.

1.2. Scope

The goal of the software product, called East Blue Portal, is to develop a single platform that connects the university's Learning Management System (LMS) and Student Information System (SIS).

Important university procedures like student enrolment, course registration, academic record management, SMS notification, and course material access will all be simplified by this integrated platform. The software product will increase productivity, reduce manual work, and reduce errors in administrative tasks linked to academic operations through combining the features of SIS and LMS into a single, integrated system.

The East Blue portal's development aims to reach a number of advantages, goals, and objectives:

- **Streamlined Procedures:** By automating and streamlining administrative duties linked to course registration, student enrolment, and academic record management, the software will reduce errors and manual work.
- **Greater User Experience:** The system will improve accessibility and user experience by giving administrators, lecturers, parents and students a single interface, which will increase productivity and satisfaction.
- **Improved Communication:** By integrating SIS and LMS, students and lecturers will be able to communicate more effectively and receive timely updates on assignments, course schedules, and academic progress.
- **Flexibility and Scalability:** The software will be built to grow with the university and adapt to its changing needs, guaranteeing flexibility and scalability over time.
- **Academic Excellence:** The East Blue Portal will help to improve the overall academic experience and promote excellence in teaching and learning by improving administrative procedures and facilitating enhanced access to learning materials.
- **Cost Savings:** The university will save money on time and resources by automating manual tasks and improving productivity.

1.3. Product Overview

1.3.1. Product Perspective

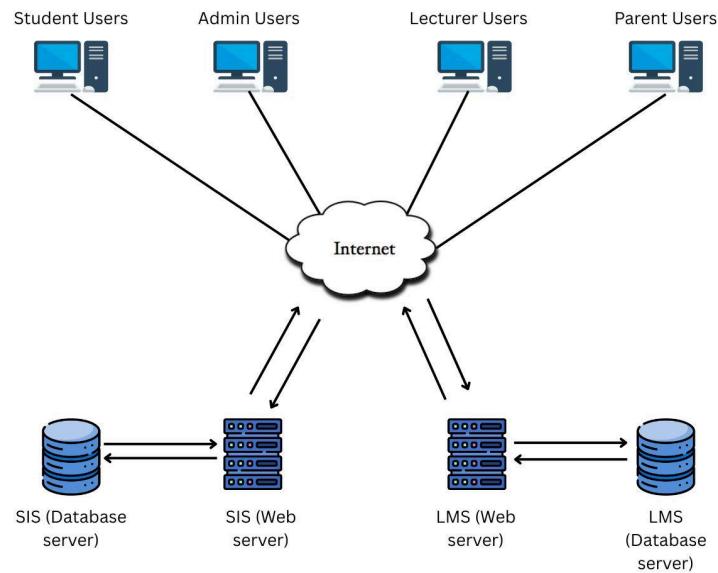


Figure 1.3.1.1 Overview Diagram

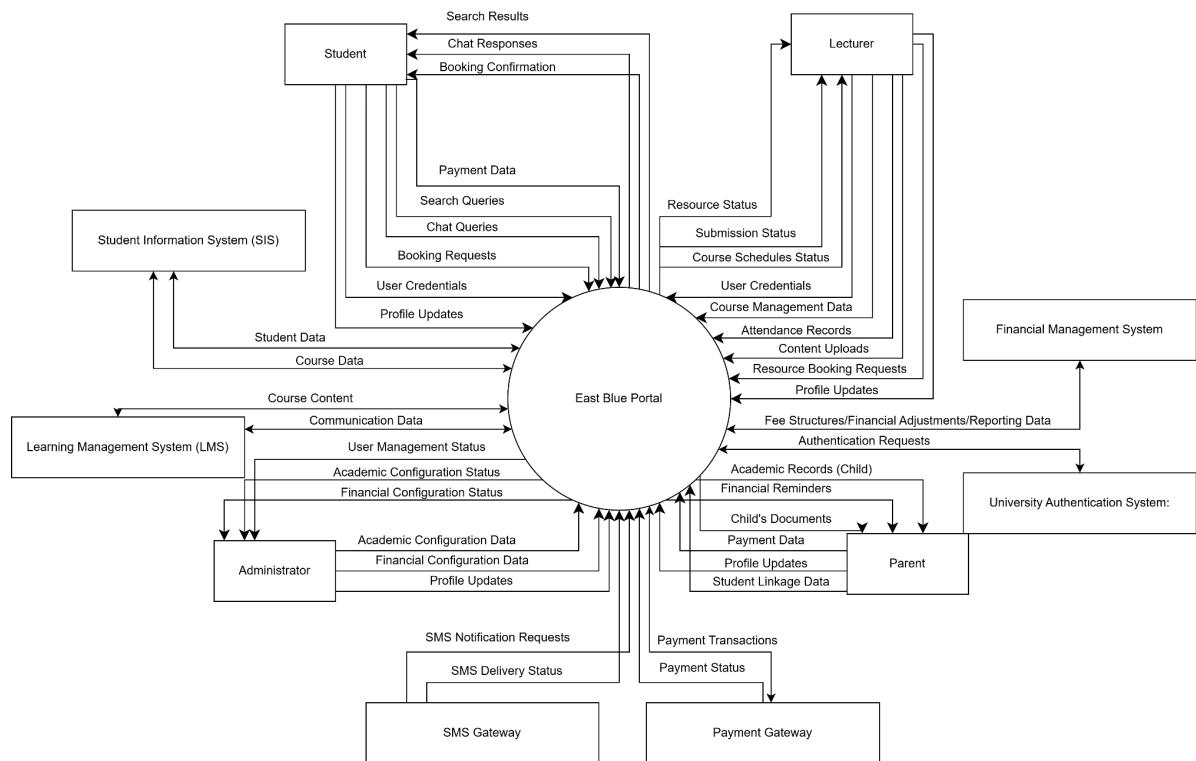


Figure 1.3.1.2 Context Diagram Diagram

<https://drive.google.com/file/d/1JnCAi-yHJR8-E8RWUKVrynEZBMQKvRA2/view?usp=sharing>

An integrated system called East Blue Portal was created to optimise academic procedures in a university setting. It connects the Learning Management System (LMS) and Student Information System (SIS) to streamline academic and administrative duties, making it an essential component of the wider educational ecosystem.

In reference to other products, East Blue Portal serves as a link between systems that are normally kept separate allowing for smooth data exchange and communication between course management and student records. Although it functions independently, it interfaces with the university's current databases, authentication systems, and outside services like SMS notification platforms.

[Diagram Description: The overview diagram depicts East Blue Portal at the center, surrounded by interconnected components. On the left side are student users, follow up administrative users. On the right side are lecturer users and parent users. The central East Blue Portal component connects to the university's databases, SIS, LMS, and external services.]

1.3.2. Product Functions

- **User Authentication and Role-Based Access:** Securely logs in users (Students, Lecturers, Parents, Administrators) and provides access to functionalities and data appropriate to their role.
- **Academic Management:** Allows students to view academic records (grades, CGPA, attendance), lecturers to manage courses (attendance, assignments, content), and administrators to manage academic structures.
- **Financial Information Management:** Enables students and parents to view financial balances, make payments, and access payment history. Administrators can manage fee structures and financial reporting.
- **Scheduling and Booking:** Provides timetable viewing for students, and allows students and lecturers to book resources such as consultations or rooms.
- **Communication and Notification:** Facilitates communication through features like an AI chatbot and sends SMS notifications for important events like deadlines or low attendance.
- **Profile and System Personalization:** Allows users to manage their profiles and customize their dashboard and notification preferences.

1.3.3. User Characteristics

Student Users

- **Educational level:** Undergraduate students of bachelor's degrees and graduate students pursuing master's or doctoral degrees are among the student users; their educational backgrounds may have an impact on how experienced they are in academic terms and procedures.
- **Experience:** Depending on their educational background and exposure to digital learning environments, students' levels of experience with educational technology tools differ.
- **Technical Proficiency:** Although many students possess strong technological abilities, some may lack these abilities, especially if it comes to difficult software systems. To allow for users with different levels of technical expertise, the system should be designed with a user interface that is simple and clear instructions.

Administrative Users

- **Educational Background:** Most administrative users have a higher level of education and frequently hold bachelor's, master's, or doctoral degrees.
- **Experience:** They have worked in educational institutions before, and their knowledge of administrative procedures and educational technology tools.
- **Technical Expertise:** Although certain administrative users possess advanced technical skills, others might only possess basic technical knowledge. As a result, the system needs to be easy to use and requires minimal instruction to operate efficiently.

Lecturer Users

- **Educational Background:** A Master's degree is generally the minimum requirement and PhD is often preferred.
- **Experience:** They have experience in the educational field, and their knowledge of teaching students and using technology to prepare educational materials such as computers or tablets.

- **Technical Expertise:** Although many lecturers have strong technological skills, some may not, particularly when it comes to complex software systems. The system should have an easy-to-use interface with clear instructions for users who have different technical skill levels.

Parent Users

- **Educational Background:** Parents can use a school portal without having any particular educational background. Regardless of their own educational background, the main goal of these portals is to give parents access to information about their child's education, including grades, attendance, and important school communications.
- **Experience:** Depending on their learning skill and exposure to digital learning environments, parents' levels of experience with educational technology tools differ.
- **Technical Expertise:** In general, parents who use portals must be familiar with online navigation and possess basic computer skills. Logging in, navigating menus, viewing information and replying to messages are all included in this. It's not always necessary to have a deep understanding of digital literacy because the portal should be easy to use and must be clear.

1.3.4. Limitations

- **SMS Notification Delays:** While the system is designed to send timely SMS notifications, delivery to the end-user is subject to the operational status and potential delays of third-party SMS gateway services and telecommunication networks.
- **Scope of AI Chatbot:** The AI chatbot's knowledge base is limited to information explicitly managed within the East Blue Portal and integrated SIS/LMS. It cannot provide information or answer queries outside this defined scope.
- **Offline Functionality:** While the desktop application may support limited offline access for some features, most functionalities, especially those requiring real-time data (e.g., viewing current grades, submitting assignments, making bookings), require an active internet connection.
- **Customization Scope:** Dashboard widget customization is limited to the predefined set of available widgets and layout options provided by the system. Users cannot create entirely new widget types without system updates.
- **Supported Languages:** While multi-language support is a feature, the initial release will be limited to English, Bahasa Melayu and Chinese. Additional languages will be considered in future releases based on demand and resources.

1.4. Definitions

- **East Blue Portal:** The integrated software solution that serves as the central platform for students, lecturers, parents, and administrators to access academic, financial, and administrative services at the university.
- **Student Information System (SIS):** The university's core system responsible for managing student data, including personal details, enrollment, registration, and academic records. The East Blue Portal interfaces with the SIS.
- **Learning Management System (LMS):** The university's platform for delivering and managing educational content, including course materials, assignments, quizzes, and communication tools. The East Blue Portal interfaces with the LMS.
- **User Role:** A classification assigned to a user (e.g., Student, Lecturer, Administrator, Parent) that dictates their access rights and permissions within the East Blue Portal .
- **Dashboard:** A personalized user interface within the East Blue Portal that provides an overview of relevant information, announcements, and quick access to frequently used functions, often customizable by the user.
- **Widget:** A modular component within the East Blue Portal dashboard that displays specific information or provides a particular function (e.g., Timetable widget, Financial Summary widget)
- **Use Case:** A description of a sequence of actions that the system performs to yield an observable result of value to a particular actor (user).
- **CGPA (Cumulative Grade Point Average):** A measure of a student's overall academic performance throughout their program of study.
- **SMS Gateway:** An external service used by the East Blue Portal to send Short Message Service (SMS) notifications to users' mobile devices

2. References

1. Lane, G. K. (n.d.). *How to write an SRS Document (Software Requirements Specification Document)*. Perforce Software.
<https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document>
2. *Requirement analysis techniques*. (n.d.).
<https://www.visual-paradigm.com/guide/requirements-gathering/requirement-analysis-techniques/>
3. GeeksforGeeks. (2024, July 24). *Requirements Engineering Process in Software Engineering*. GeeksforGeeks.
<https://www.geeksforgeeks.org/software-engineering-requirements-engineering-process/>
4. Pragmatic institute. (n.d.). *Prioritizing Software Requirements with Kano Analysis*.
<https://www.pragmaticinstitute.com/resources/articles/product/prioritizing-software-requirements-with-kano-analysis/>
5. What is the kano model? meaning, working and categories. Young Urban Project. (2025, January 28).
<https://www.youngurbanproject.com/what-is-the-kano-model/#:~:text=The%20Kano%20Model%20helps%20businesses,what's%20most%20important%20for%20custom>
6. airfocus (n.d.) *What Is the Kano Model? Kano Model Definition, Origin & FAQ*
<https://airfocus.com/glossary/what-is-the-kano-model/>
7. *Enhance Customer Satisfaction with the Kano Model: Learn More on Our Blog*. (2024, August 2). HEFLO. <https://www.heflo.com/blog/kano-model>
8. Kano's model and five categories of quality attribute. | Download Scientific Diagram Using (n.d.) *Citation Generators Responsibly*
https://www.researchgate.net/figure/Kanos-model-and-five-categories-of-quality-attribute_fig1_233473957

3. Requirements

3.1. Functions

The Use Case Diagram below illustrates the main functions of the **East Blue Portal** and how different users interact with the system. There are four key actors:

- Student
- Lecturer
- Administration
- Parent

Each actor interacts with the system to perform specific tasks based on their role. The diagram shows major system functions such as login, viewing dashboard, managing academic and financial data, and booking services. The system also includes supportive features like Multi-language Support, AI Chatbot, and SMS Notifications.

To represent relationships between use cases, the diagram uses:

- <<include>>: Indicates that one use case always uses the behavior of another. For example, Online Booking includes both Venue Booking and Consultation Booking.
- <<extend>>: Indicates optional behavior. For instance, Profile Settings may extend into Change Password and Change Theme based on user actions.



Figure 3.1.1 Use Case Diagram

<https://drive.google.com/file/d/1VfQDRXVtONjeqw3sMYvtWZPUVFw5bcPK/view?usp=sharing>

3.1.1. Use Case Specification

3.1.1.1. Student/Lecturer/Parent/Admin Login

Table 3.1.1.1.1 Student/Lecturer/Parent/Admin Login

Use Case ID	REC_UC_0001
Use Case Name:	Login
Description:	Allows users to securely log into the system based on their role.
PrimaryActor:	Student/Lecturer/Parent
Precondition:	<ol style="list-style-type: none"> 1. The user must be registered and have valid login credentials.
Postcondition:	<ol style="list-style-type: none"> 1. The user is redirected to their respective dashboard or shown an error.
MainSuccessScenario:	<ol style="list-style-type: none"> 1. User accesses the login page. 2. System checks for saved credentials in cookies. 3. Pre-filled login form is displayed (if data exists). 4. User submits login form. 5. System authenticates credentials via the database. 6. If valid, login is successful: <ul style="list-style-type: none"> - Credentials saved in cookies. - User redirected to dashboard. 7. If invalid, error message is shown.
AlternativeScenario:	<ol style="list-style-type: none"> 1. Admin accesses the login page. 2. System checks user role. 3. If Admin detected, shows blank login form. 4. Admin submits login credentials. 5. Credentials are validated with the database. 6. If valid, Admin is redirected to the Admin dashboard. 7. If invalid, error message is displayed.
ExceptionScenario:	<ol style="list-style-type: none"> 1. Invalid credentials: Error message is displayed 2. Database or AuthService is unavailable: Login fails gracefully with error

3.1.1.2. Student Dashboard

Table 3.1.1.2.1 Student Dashboard

Use Case ID	REC_UC_0002
Use Case Name:	View Student Dashboard
Description:	Allows the student to personalize their dashboard by arranging

	and adding widgets.
PrimaryActor:	Student
Precondition:	<ol style="list-style-type: none"> 1. Student is logged in 2. Dashboard UI is available 3. Database is online
Postcondition:	<ol style="list-style-type: none"> 1. Updated layout and widget preferences are saved and reflected on the dashboard
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Student accesses dashboard 2. System retrieves saved preferences 3. Dashboard is displayed 4. Student repositions a widget 5. New layout is saved 6. System updates and reflects the change
AlternativeScenario:	<ol style="list-style-type: none"> 1. If no preferences exist, default layout is shown 2. If widget is unavailable, a message is shown
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database is unreachable: error is shown 2. Widget Manager fails to update: changes are not saved

3.1.1.3. Calendar View & SMS Notification for Student

Table 3.1.1.3.1 AI Calendar View & SMS Notification for Student

Use Case ID	REC_UC_0003
Use Case Name:	View/Manage Calendar
Description:	This use case allows a student to view their academic calendar, receive important notifications (via SMS), and get automated course recommendations based on preferences.
PrimaryActor:	Student
Precondition:	<ol style="list-style-type: none"> 1. Student is logged in 2. Calendar, Notification, and Enrollment modules are accessible 3. Database and SMS gateway are operational
Postcondition:	<ol style="list-style-type: none"> 1. Student sees calendar events and details 2. SMS notifications are sent if certain criteria are met 3. Student is optionally enrolled in recommended courses
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Student accesses calendar 2. Calendar UI retrieves and displays schedule 3. Notification service checks for deadlines, attendance, and finance info 4. SMS reminders are sent if needed

	<ol style="list-style-type: none"> 5. Student configures preferences for auto-enrollment 6. System recommends courses 7. Student confirms enrollment 8. System updates schedule
AlternativeScenario:	<ol style="list-style-type: none"> 1. No upcoming events: calendar appears empty 2. No contact info found: skip SMS sending 3. Student skips enrollment recommendation
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database or SMS service fails: display error or retry 2. Enrollment fails due to course capacity or prerequisite issues

3.1.1.4. View Academic

Table 3.1.1.4.1 View Academic

Use Case ID	REC_UC_0004
Use Case Name:	View Academic
Description:	This use case enables users (primarily students) to access a comprehensive academic dashboard that includes grades, attendance, enrollment status, CGPA tracking, and achievements.
PrimaryActor:	Student, Parent
Precondition:	<ol style="list-style-type: none"> 1. User is logged in 2. Academic data is available in the database 3. Student is enrolled in at least one course
Postcondition:	<ol style="list-style-type: none"> 1. User can view up-to-date academic data in a readable and interactive format
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Student navigates to academic dashboard 2. System retrieves complete academic record (current + past grades) 3. CGPA progression chart is generated 4. Course enrollment and degree tracker are displayed 5. Attendance data for all courses is shown 6. Achievements and recognitions are listed
AlternativeScenario:	<ol style="list-style-type: none"> 1. No academic achievements: section is hidden or shows a placeholder 2. No attendance data: system informs user data is pending or unavailable
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database error or missing records: system shows fallback error message 2. Chart rendering issue: system provides tabular alternative

3.1.1.5. Booking Venue/Consultation

Table 3.1.1.5.1 Booking Venue/Consultation

Use Case ID	REC_UC_0005
Use Case Name:	Online Booking
Description:	This use case describes the process by which a student books an appointment, either for a venue or a consultation, through the system.
PrimaryActor:	Student
Precondition:	<ol style="list-style-type: none"> 1. Student is logged in 2. Booking system and database are operational.
Postcondition:	<ol style="list-style-type: none"> 1. Appointment details are stored in the system. 2. Confirmation is provided to the student.
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Student access the booking page. 2. System displays the appropriate booking form (venue or consultation). 3. Student fills in the required details and submits the form. 4. Booking request is sent to the Booking Service. 5. Booking Service stores the appointment in the database. 6. Confirmation of the booking is shown to the student.
AlternativeScenario:	<ol style="list-style-type: none"> 1. If required form fields are missing or invalid, the system prompts the user to correct them.
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database is unreachable or fails during booking save. 2. Booking Service encounters an internal error.

3.1.1.6. AI Chatbot Assistance

Table 3.1.1.6.1 AI Chatbot Assistance

Use Case ID	REC_UC_0006
Use Case Name:	AI Chatbot
Description:	Allows the student to ask academic or administrative questions to an AI chatbot and receive answers.
PrimaryActor:	Student
Precondition:	<ol style="list-style-type: none"> 1. Student is logged in 2. Chatbot and database are active
Postcondition:	<ol style="list-style-type: none"> 1. Student receives accurate information based on their queries

MainSuccessScenario:	<ol style="list-style-type: none"> 1. Student asks chatbot a question 2. Chatbot queries database 3. Chatbot provides relevant response 4. Follow-up queries are answered appropriately
AlternativeScenario:	<ol style="list-style-type: none"> 1. Chatbot clarifies or rephrases if question is ambiguous
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database failure: chatbot informs student of service error 2. Chatbot unavailable: fallback message shown

3.1.1.7. AI System Search

Table 3.1.1.7.1 AI System Search

Use Case ID	REC_UC_0007
Use Case Name:	AI Search
Description:	Enables the student to search across the system using an AI-powered global search bar.
PrimaryActor:	Student, Lecturer, Administration, Parent
Precondition:	<ol style="list-style-type: none"> 1. Student is logged in 2. Search component and DB are functional
Postcondition:	<ol style="list-style-type: none"> 1. Search results are retrieved and displayed to the student
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Student enters query 2. Search system receives it 3. Database is queried 4. Results returned and shown
AlternativeScenario:	<ol style="list-style-type: none"> 1. No results found: message shown to student
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database/search component is down: error message displayed

3.1.1.8. Manage Finance

Table 3.1.1.8.1 Manage Finance

Use Case ID	REC_UC_0008
Use Case Name:	Manage Finance
Description:	Enables user to manage their financial records, including viewing current balances, accessing payment history and receipts, checking financial aid information, and making online payments.
PrimaryActor:	Student, Parent

Precondition:	<ol style="list-style-type: none"> 1. User is logged in 2. Finance service, database, and payment gateway are operational.
Postcondition:	<ol style="list-style-type: none"> 1. User can view and update their financial records, initiate payments, and receive notifications.
MainSuccessScenario:	<ol style="list-style-type: none"> 1. User opens the Finance section. 2. System displays balance and pending payments. 3. User views payment history and downloads receipts. 4. User reviews financial aid/scholarship info. 5. User chooses to make a payment and completes it through the payment gateway. 6. System sends payment deadline reminders if applicable.
AlternativeScenario:	<ol style="list-style-type: none"> 1. No aid or scholarships: system shows "no records available." 2. - No payment history: show "no previous transactions."
ExceptionScenario:	<ol style="list-style-type: none"> 1. Payment gateway failure: show error message, allow retry. 2. - Database offline: show error, disable further actions. 3. - Notification system down: no reminders sent.

3.1.1.9. Edit Profile

Table 3.1.1.9.1 Edit Profile

Use Case ID	REC_UC_0009
Use Case Name:	Edit Profile
Description:	This use case allows users to personalize their profile by updating notification settings, appearance preferences, and language/regional options.
PrimaryActor:	Student, Lecturer, Administration, Parent
Precondition:	<ol style="list-style-type: none"> 1. User is logged in 2. Profile and settings modules are accessible 3. Database is operational
Postcondition:	<ol style="list-style-type: none"> 1. Updated preferences are saved and applied immediately or on next session
MainSuccessScenario:	<ol style="list-style-type: none"> 1. User accesses profile settings 2. User updates notification preferences (email, SMS, in-app) 3. User customizes appearance (theme, layout, color scheme) 4. User selects preferred language and regional settings

	5. System saves all updates to the database 6. Confirmation is displayed to the user
AlternativeScenario:	1. User makes changes but cancels before saving: no update is applied
ExceptionScenario:	1. Database error while saving settings: system notifies user and retains old preferences 2. UI error: system reloads settings and prompts retry with error

3.1.1.10. User Management

Table 3.1.1.10.1 User Management

Use Case ID	REC_UC_0010
Use Case Name:	Manage User
Description:	This use case enables an admin to manage all user accounts, including creating, editing, deactivating, bulk importing/exporting users, and managing account security such as password resets.
PrimaryActor:	Admin
Precondition:	1. Admin is logged into the system. 2. Database and user services are operational. 3. Admin has the necessary permissions.
Postcondition:	1. User accounts are created, updated, or deactivated. 2. Bulk user actions are completed. 3. Passwords are reset or security settings updated.
MainSuccessScenario:	1. Admin accesses User Management. 2. Admin creates, edits, or deactivates user accounts. 3. Admin performs bulk import/export. 4. Admin resets passwords and manages security.
AlternativeScenario:	1. If a user already exists, system shows "duplicate account" error.
ExceptionScenario:	1. Database failure during save: system shows error. 2. File format error during bulk import: system rejects file with reason.

3.1.1.11. Academic Administration

Table 3.1.1.11.1 Academic Administration

Use Case ID	REC_UC_0011
Use Case Name:	Manage Academic Administration
Description:	This use case allows the admin to configure academic structure, including academic periods, faculties/departments, course management, and academic policies such as grading scales and enrollment rules.
PrimaryActor:	Admin
Precondition:	<ol style="list-style-type: none"> 1. Admin is logged into the system. 2. System and database are operational. 3. Admin has academic configuration permissions.
Postcondition:	<ol style="list-style-type: none"> 1. Academic structure and settings are successfully configured and saved. 2. Courses and programs are properly scheduled and updated.
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Admin configures academic periods (e.g., semesters). 2. Admin sets up faculties, departments, and programs. 3. Admin creates courses and assigns schedules. 4. Admin overrides enrollment/prerequisites as needed. 5. Admin configures grading scales and academic policies.
AlternativeScenario:	<ol style="list-style-type: none"> 1. If a course conflicts with an existing schedule, an error is shown. 2. If a department or program already exists, system prevents duplication.
ExceptionScenario:	<ol style="list-style-type: none"> 1. System error while saving configuration. 2. Database connection failure. 3. Invalid grading scale or schedule format.

3.1.1.12. Financial Administration

Table 3.1.1.12.1 Financial Administration

Use Case ID	REC_UC_0012
Use Case Name:	Manage Finance Administration
Description:	This use case allows the admin to manage and configure financial systems including fee structures, payment gateways, reporting, and financial adjustments such as refunds.
PrimaryActor:	Admin
Precondition:	<ol style="list-style-type: none"> 4. Admin is logged into the system. 5. Financial systems and database are operational.

Postcondition:	4. Payment systems, fee structures, and schedules are updated. 5. Reports and logs are generated. 6. Refunds or adjustments are recorded.
MainSuccessScenario:	5. Admin configures payment systems and gateways. 6. Admin defines or updates fee structures and schedules. 7. Admin generates reports and views audit logs. 8. Admin processes refunds or applies adjustments.
AlternativeScenario:	2. Gateway setup fails due to misconfiguration. 3. Attempt to generate report with no data returns warning. 4. Duplicate fee structure entry prevented.
ExceptionScenario:	3. Database error during update. 4. Payment gateway unreachable. 5. Invalid refund amount or permissions denied.

3.1.1.13. Dashboard for Admin, Lecturer, and Parent

Table 3.1.1.13.1 Dashboard for Admin, Lecturer, and Parent

Use Case ID	REC_UC_0013
Use Case Name:	Dashboard
Description:	This use case allows different user types (Admin, Lecturer, Parent) to view a role-specific dashboard containing relevant widgets and summaries based on their responsibilities and permissions.
PrimaryActor:	Admin, Lecturer, Parent
Precondition:	4. The user is logged in 5. Dashboard components are configured and data is available.
Postcondition:	1. The user sees a dashboard view with relevant information.
MainSuccessScenario:	1. User logs in and accesses the dashboard. 2. System loads dashboard layout based on role. 3. System fetches and displays real-time data (e.g., summaries, alerts, charts).
AlternativeScenario:	1. System shows the default layout.
ExceptionScenario:	1. Dashboard fails to load due to a system error. 2. Data fetch timeout or database unreachable.

3.1.1.14. Course Management for Lecturer

Table 3.1.14.1 Course Management for Lecturer

Use Case ID	REC_UC_0014
Use Case Name:	Manage Course
Description:	This use case allows lecturers to manage their assigned courses, including viewing upcoming class schedules, editing course details, and monitoring student enrollment and status.
PrimaryActor:	Admin, Lecturer, Parent
Precondition:	<ol style="list-style-type: none"> 1. Lecturer is logged in 2. Courses are assigned to the lecturer. 3. Class schedules and student records exist in the database.
Postcondition:	<ol style="list-style-type: none"> 1. Course data is displayed and/or updated. 2. Student and schedule information is retrieved and shown.
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Lecturer accesses the Course Management interface. 2. System loads assigned courses. 3. Lecturer views course details including upcoming classes and student stats. 4. Lecturer edits course information if needed. 5. System saves and reflects the changes.
AlternativeScenario:	<ol style="list-style-type: none"> 1. No courses assigned: display empty state or warning. 2. Lecturer edits only partial course info.
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database or schedule service is unavailable. 2. Save/edit operation fails due to validation errors.

3.1.1.15. Attendance tracking for Lecturer

Table 3.1.15.1 Attendance tracking for Lecturer

Use Case ID	REC_UC_0015
Use Case Name:	Track Attendance
Description:	This use case allows lecturers to record and monitor student attendance, view attendance statistics, identify students at risk, send attendance-related reminders, and generate reports.
PrimaryActor:	Lecturer
Precondition:	<ol style="list-style-type: none"> 1. Lecturer is logged in 2. Course and student enrollment data are available. 3. Attendance system is operational.
Postcondition:	<ol style="list-style-type: none"> 1. Attendance is recorded or updated. 2. Attendance statistics are generated and shown.

	<p>3. Notifications may be sent to students with issues..</p>
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Lecturer accesses the attendance module. 2. Selects a course and session. 3. Marks attendance for each student. 4. System saves the attendance. 5. Lecturer views statistics and can generate reports.
AlternativeScenario:	<ol style="list-style-type: none"> 1. Attendance already recorded: system allows update or blocks changes. 2. Lecturer only views stats without recording.
ExceptionScenario:	<ol style="list-style-type: none"> 1. Database/service failure. 2. Student/course data unavailable. 3. Notification service fails to send reminders.

3.1.1.16. Assignment & Assessment Management for Lecturer

Table 3.1.1.16.1 Assignment & Assessment Management for Lecturer

Use Case ID	REC_UC_0016
Use Case Name:	Manage Assignment
Description:	This use case allows lecturers to create and manage assignments, set deadlines, grading rubrics, track submissions, assign grades, give feedback, and analyze student performance.
PrimaryActor:	Lecturer
Precondition:	<ol style="list-style-type: none"> 1. Lecturer is logged in 2. Assigned courses are available. 3. Students are enrolled.
Postcondition:	<ol style="list-style-type: none"> 1. Assignments are created/updated. 2. Grades and feedback are stored. 3. Performance metrics are accessible.
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Lecturer creates or edits an assignment. 2. Sets deadline and grading rubric. 3. Views and tracks student submissions. 4. Grades submissions and adds feedback. 5. Views performance analytics.
AlternativeScenario:	<ol style="list-style-type: none"> 1. Edit existing assignment. 2. Regrade with updated rubric. 3. Download submissions offline.
ExceptionScenario:	<ol style="list-style-type: none"> 1. Submission data not loading. 2. Assignment creation fails due to system/database error.

3.1.1.17. Content Creation for Lecturer

Table 3.1.1.17.1 Content Creation for Lecturer

Use Case ID	REC_UC_0017
Use Case Name:	Create Content
Description:	Lecturers create, upload, and manage learning resources such as lecture notes, videos, and readings. Content is structured into modules/topics and visibility is controlled. Student engagement is tracked via analytics.
PrimaryActor:	Lecturer
Precondition:	<ol style="list-style-type: none"> 1. Lecturer is logged in 2. Lecturer is assigned to a course
Postcondition:	<ol style="list-style-type: none"> 1. Content is uploaded and structured 2. Visibility settings are applied 3. Engagement data is tracked
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Lecturer accesses course content module 2. Uploads files or creates resource entries 3. Organizes materials by topic/module 4. Sets visibility for each item 5. Views student interaction metrics
AlternativeScenario:	<ol style="list-style-type: none"> 1. Edit or delete existing content 2. Schedule content visibility for future dates
ExceptionScenario:	<ol style="list-style-type: none"> 1. Upload fails due to network or format error 2. Engagement data not available.

3.1.1.18. Resource Scheduling for Lecturer

Table 3.1.1.18.1 Resource Scheduling for Lecturer

Use Case ID	REC_UC_0018
Use Case Name:	Resource Scheduling
Description:	Lecturers can schedule resources such as lecture halls, labs, projectors, and manage requests for teaching equipment. They can also set up office hours and view the availability and booking status of resources.
PrimaryActor:	Lecturer
Precondition:	<ol style="list-style-type: none"> 1. Lecturer is logged in 2. Access to scheduling module is granted
Postcondition:	<ol style="list-style-type: none"> 1. Booking is recorded in system

	<ol style="list-style-type: none"> 2. Resource status is updated 3. Conflicts are avoided
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Lecturer accesses scheduling page 2. Searches for available resources 3. Selects resource and time slot 4. Submits booking request 5. Views updated schedule including confirmed/pending bookings
AlternativeScenario:	<ol style="list-style-type: none"> 1. Set recurring bookings (e.g., weekly classes) 2. Modify or cancel bookings 3. Set office hours
ExceptionScenario:	<ol style="list-style-type: none"> 1. Resource already booked (conflict) 2. Equipment unavailable 3. Booking not confirmed due to admin hold

3.1.1.19. Document Access for Parent

Table 3.1.1.19.1 Document Access for Parent

Use Case ID	REC_UC_0019
Use Case Name:	Access Document
Description:	Allows parents to access a centralized repository of documents related to their child's academic and financial activities. This includes academic reports, transcripts, fee statements, official notices, and parent resources.
PrimaryActor:	Parent
Precondition:	<ol style="list-style-type: none"> 1. Parent is logged in 2. Parent is linked to student account(s) 3. Documents have been uploaded by the system/admin
Postcondition:	<ol style="list-style-type: none"> 1. Parent views/downloads documents
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Parent logs in and accesses Document Center 2. System fetches list of categorized documents 3. Parent views/downloads selected document
AlternativeScenario:	<ol style="list-style-type: none"> 1. Filter by document type or date 2. View only unread/new documents 3. Access archived documents
ExceptionScenario:	<ol style="list-style-type: none"> 1. No documents available 2. Access denied due to missing student linkage 3. Document server unavailable

3.1.1.20. Manage Student for Parent

Table 3.1.1.20.1 Manage Student for Parent

Use Case ID	REC_UC_0020
Use Case Name:	Manage Student
Description:	Allows parents to add, edit, or remove linked student profiles (typically their children) to access academic, financial, and communication services.
PrimaryActor:	Parent
Precondition:	<ol style="list-style-type: none"> 1. Parent is logged in 2. System allows parent-child linking
Postcondition:	<ol style="list-style-type: none"> 1. Student list is updated (add/edit/delete) 2. Linkage reflected in backend for document/academic access
MainSuccessScenario:	<ol style="list-style-type: none"> 1. Parent accesses Manage Student page 2. Parent views current linked students 3. Parent adds or edits a student profile (e.g., name, student ID) 4. System updates the database and confirms the action
AlternativeScenario:	<ol style="list-style-type: none"> 1. Parent removes a student 2. Validation on student info (e.g., must exist in university system) 3. Set primary/default student
ExceptionScenario:	<ol style="list-style-type: none"> 1. Entered student ID is invalid or not found 2. Parent not authorized to manage student 3. Database update fails

3.1.2. Sequence Diagram

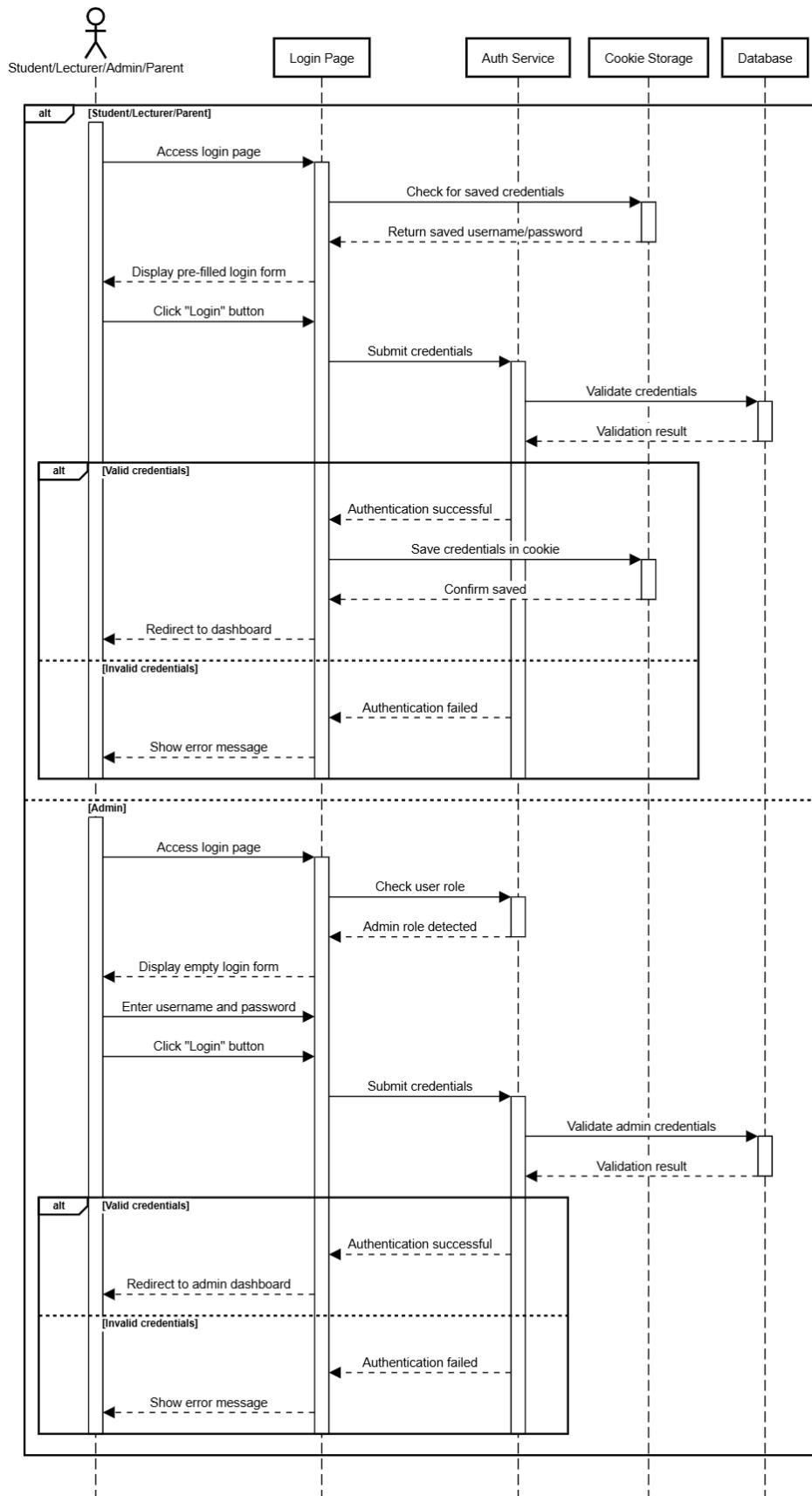
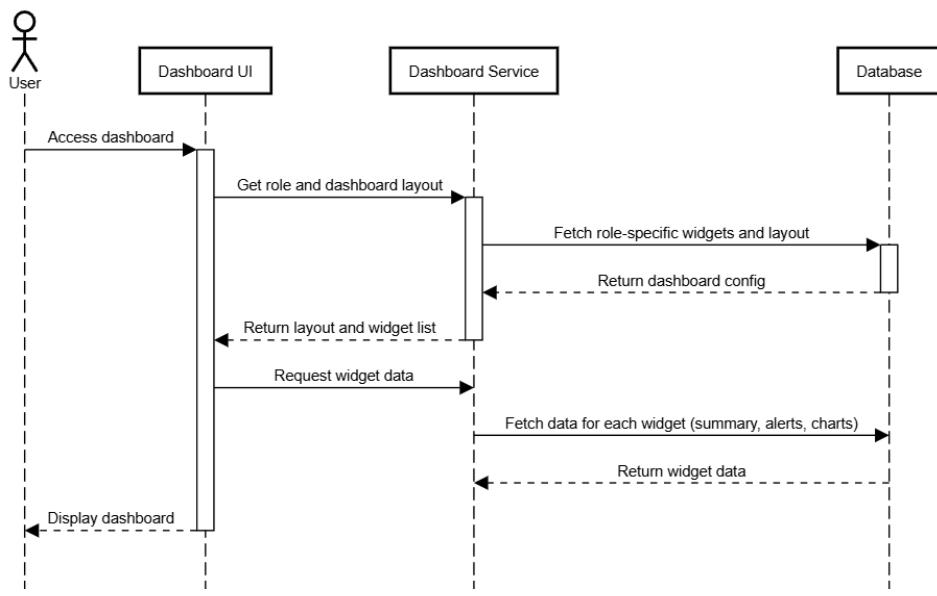


Figure 3.1.2.1 Sequence Diagram for Student/Lecturer/Parent Login



*Figure 3.1.2.3 Sequence Diagram for **View Student Dashboard***

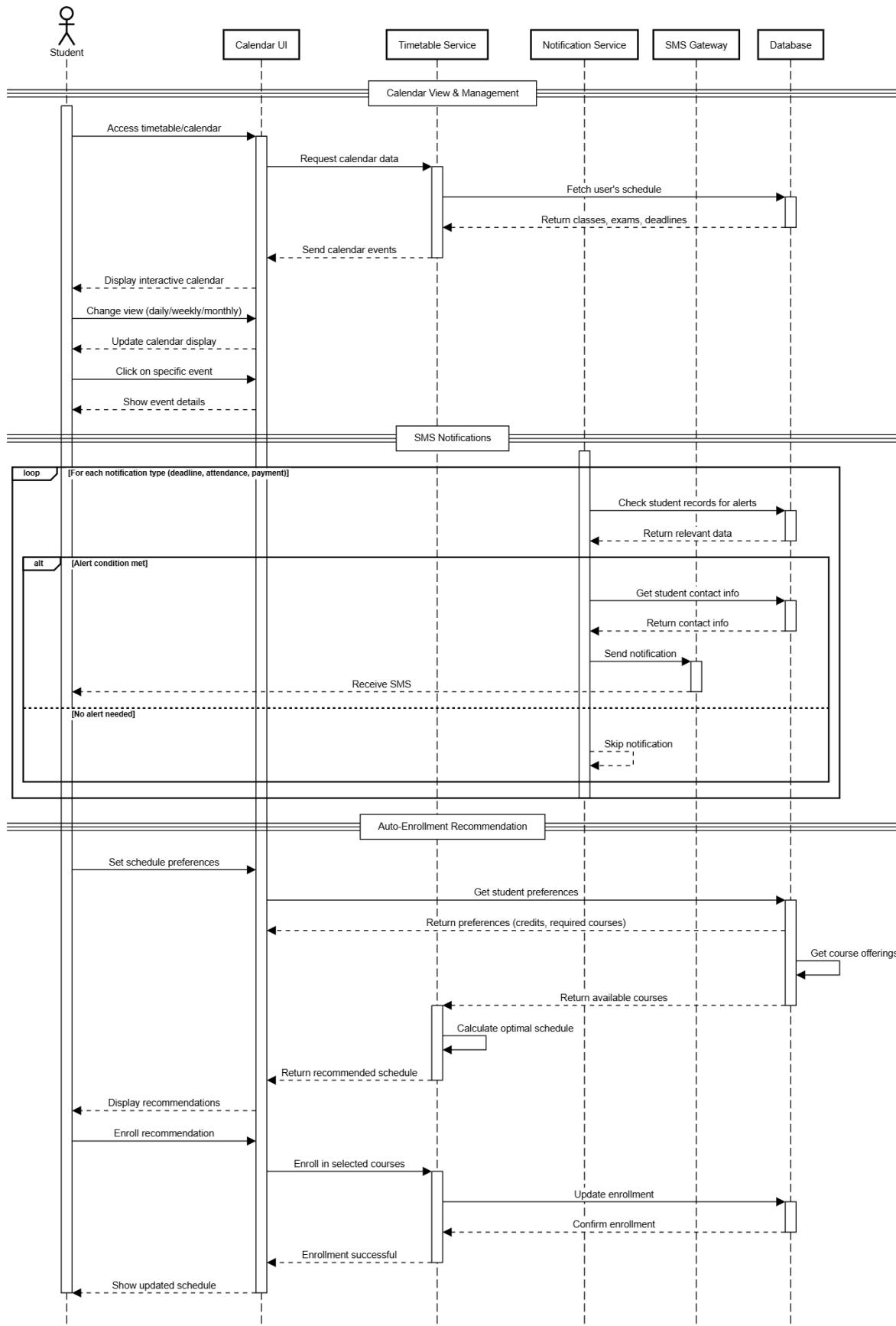


Figure 3.1.2.4 Sequence Diagram for **View/Manage Calendar**

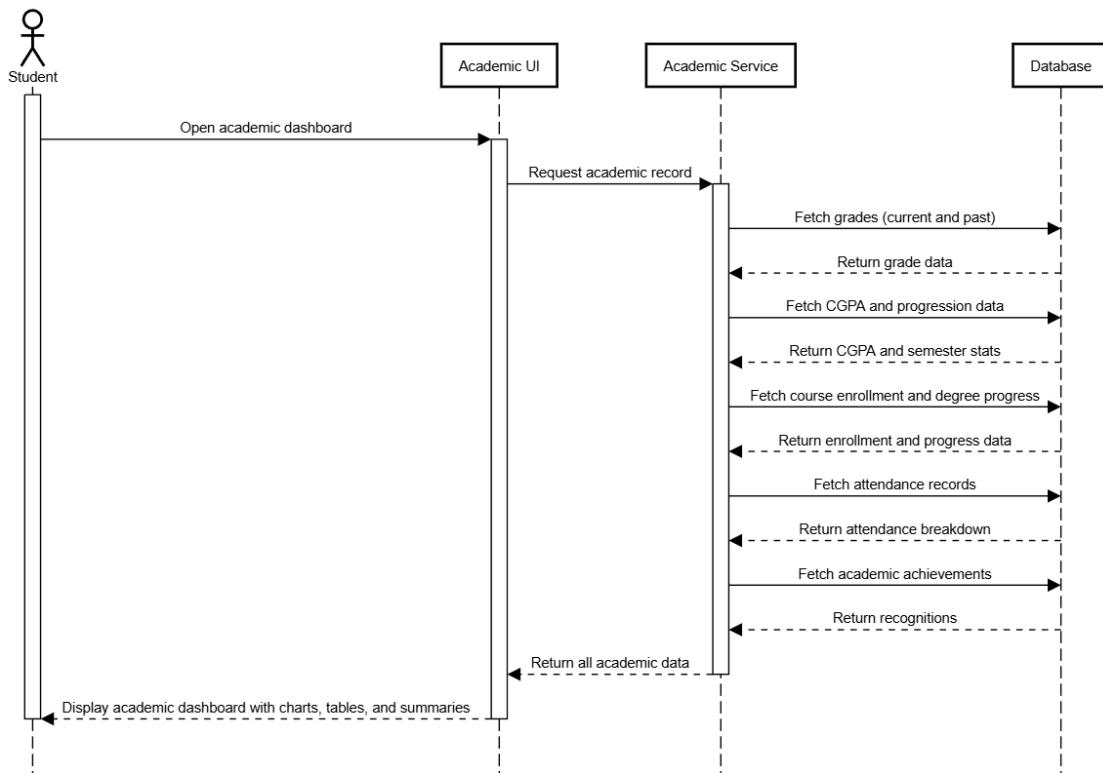


Figure 3.1.2.5 Sequence Diagram for **View Academic**

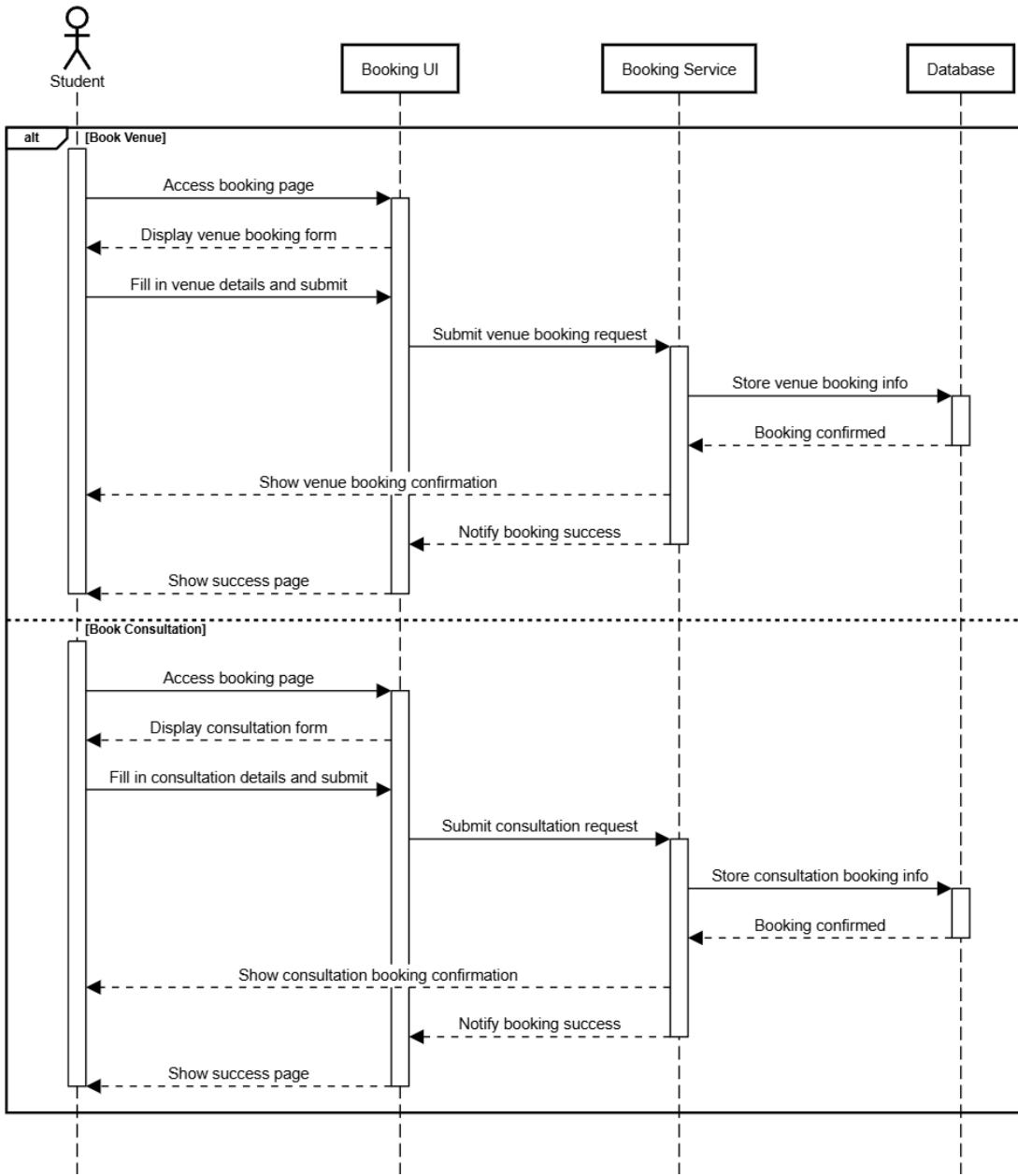


Figure 3.1.2.6 Sequence Diagram for **Online Booking**

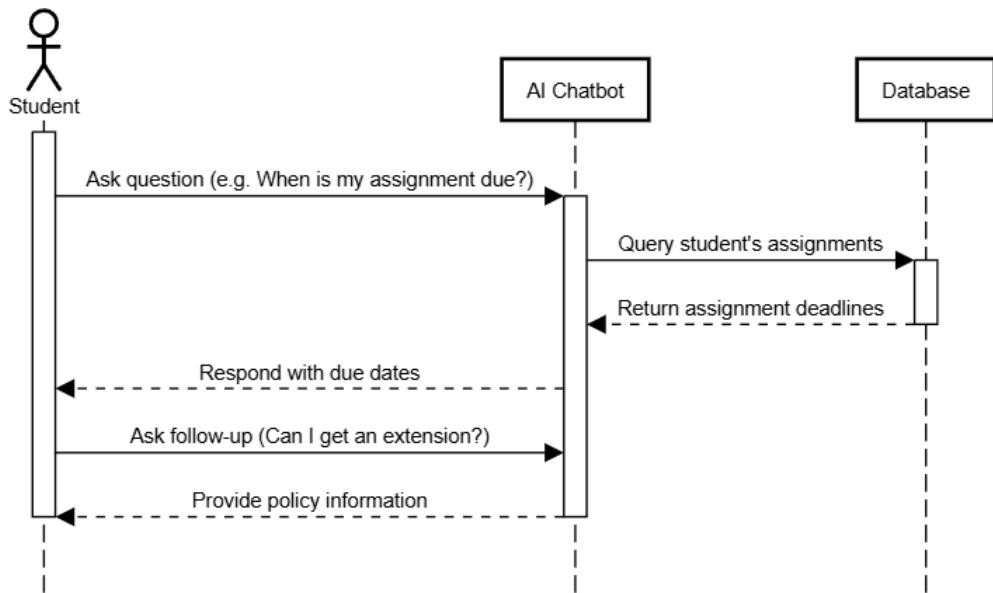


Figure 3.1.2.7 Sequence Diagram for **AI Chatbot**

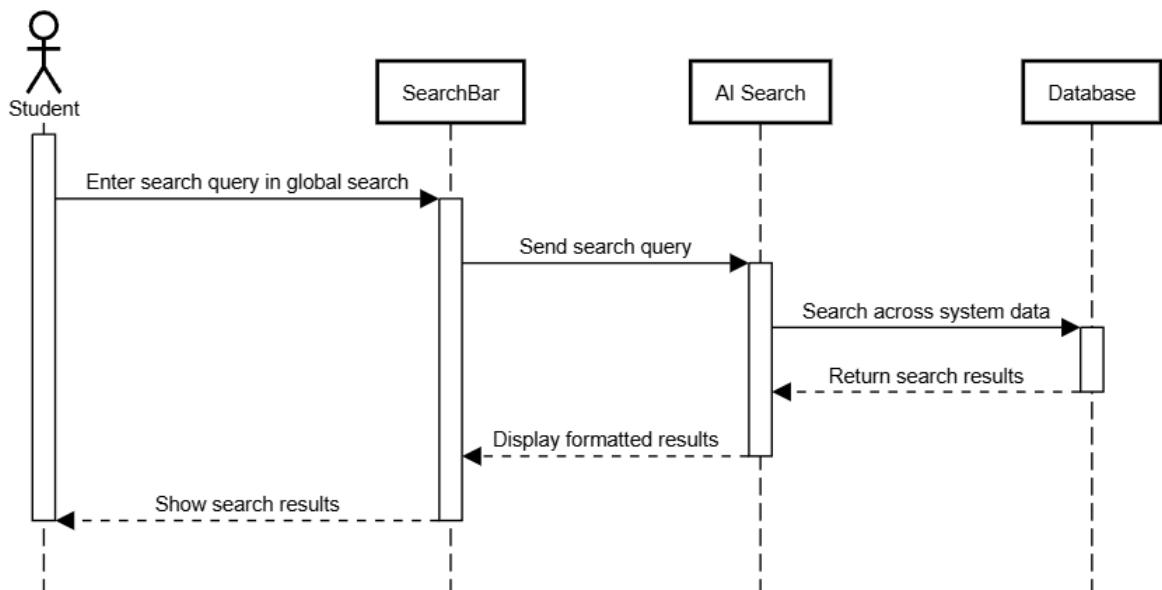


Figure 3.1.2.8 Sequence Diagram for **AI Search**

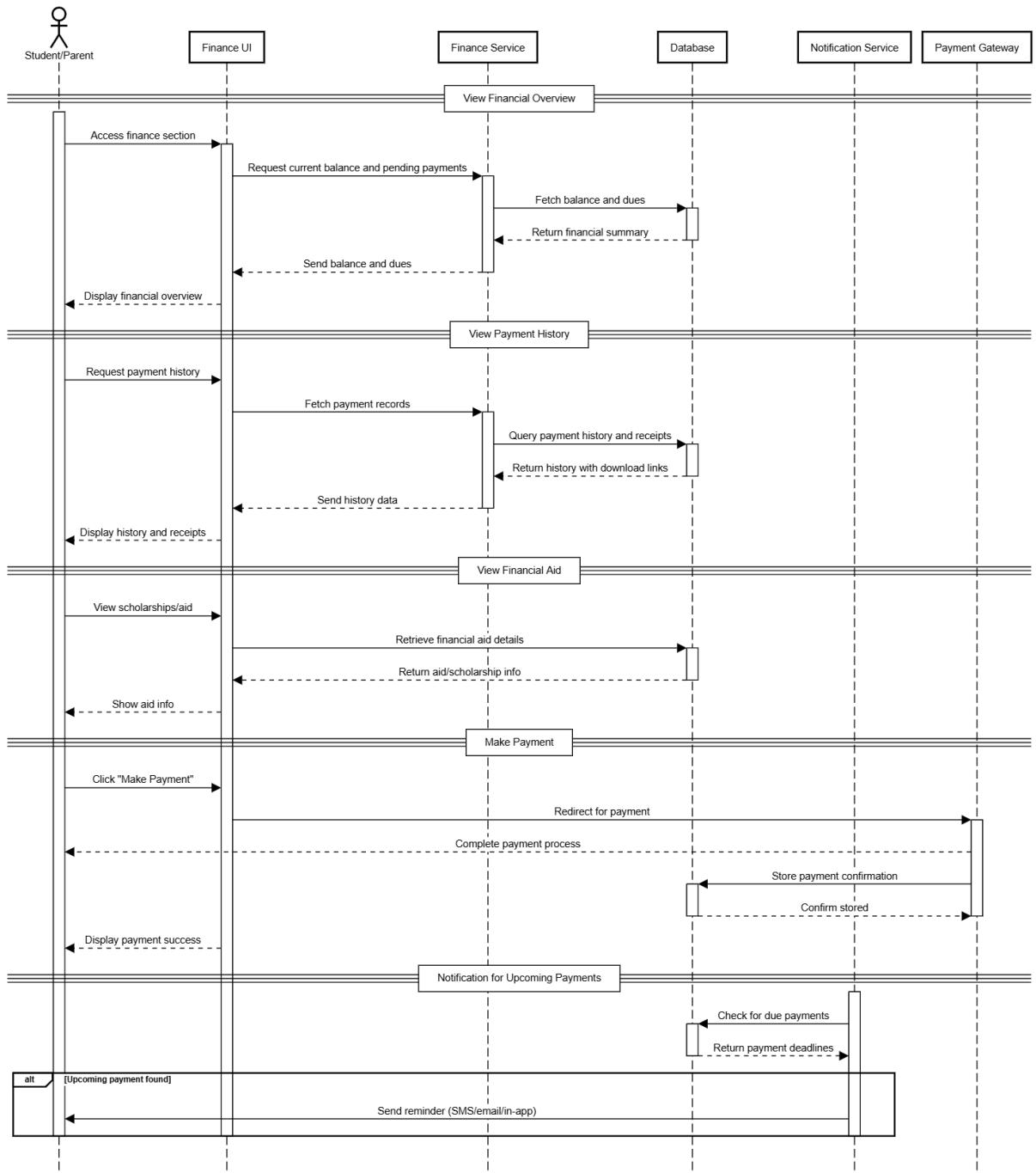


Figure 3.1.2.9 Sequence Diagram for Manage Finance

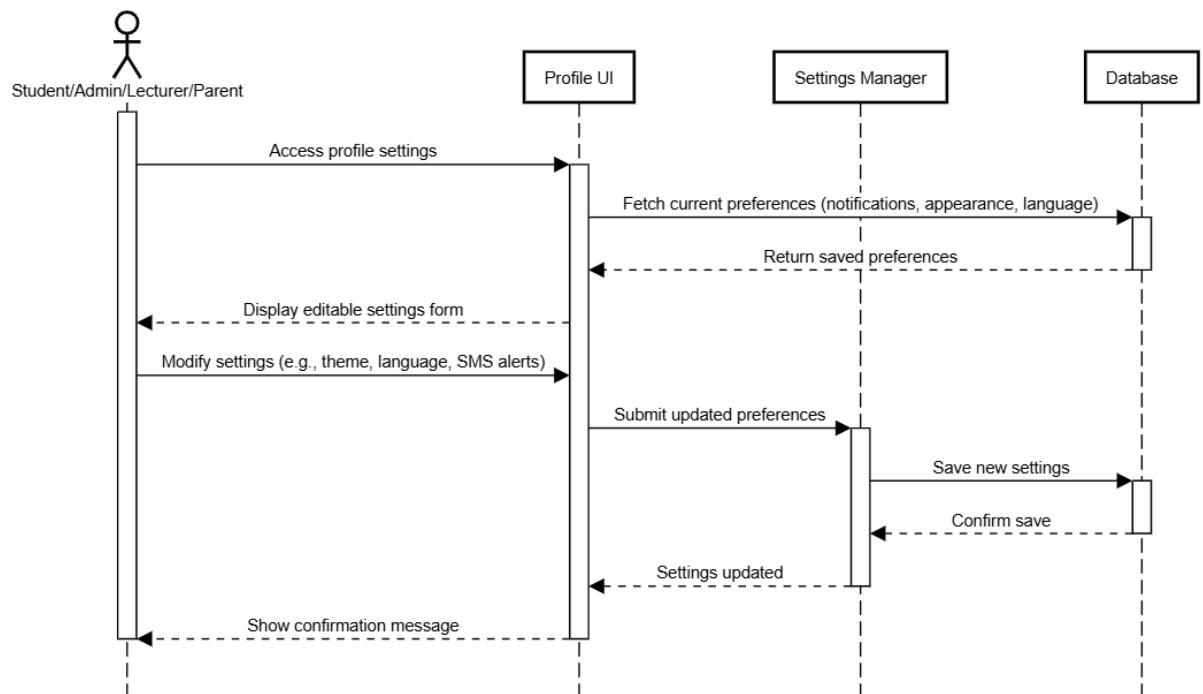


Figure 3.1.2.10 Sequence Diagram for **Edit Profile**

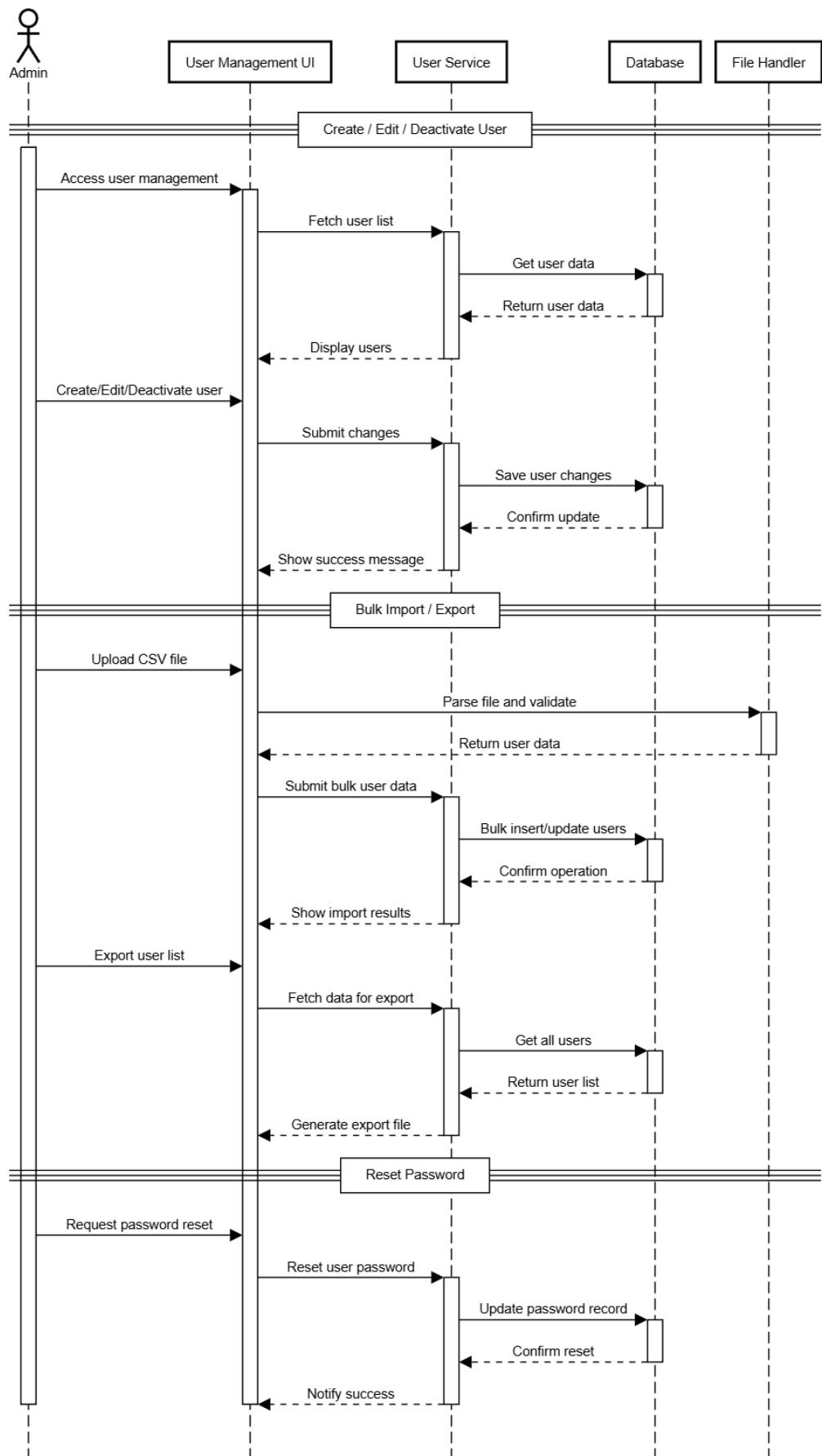


Figure 3.1.2.11 Sequence Diagram for **User Management**

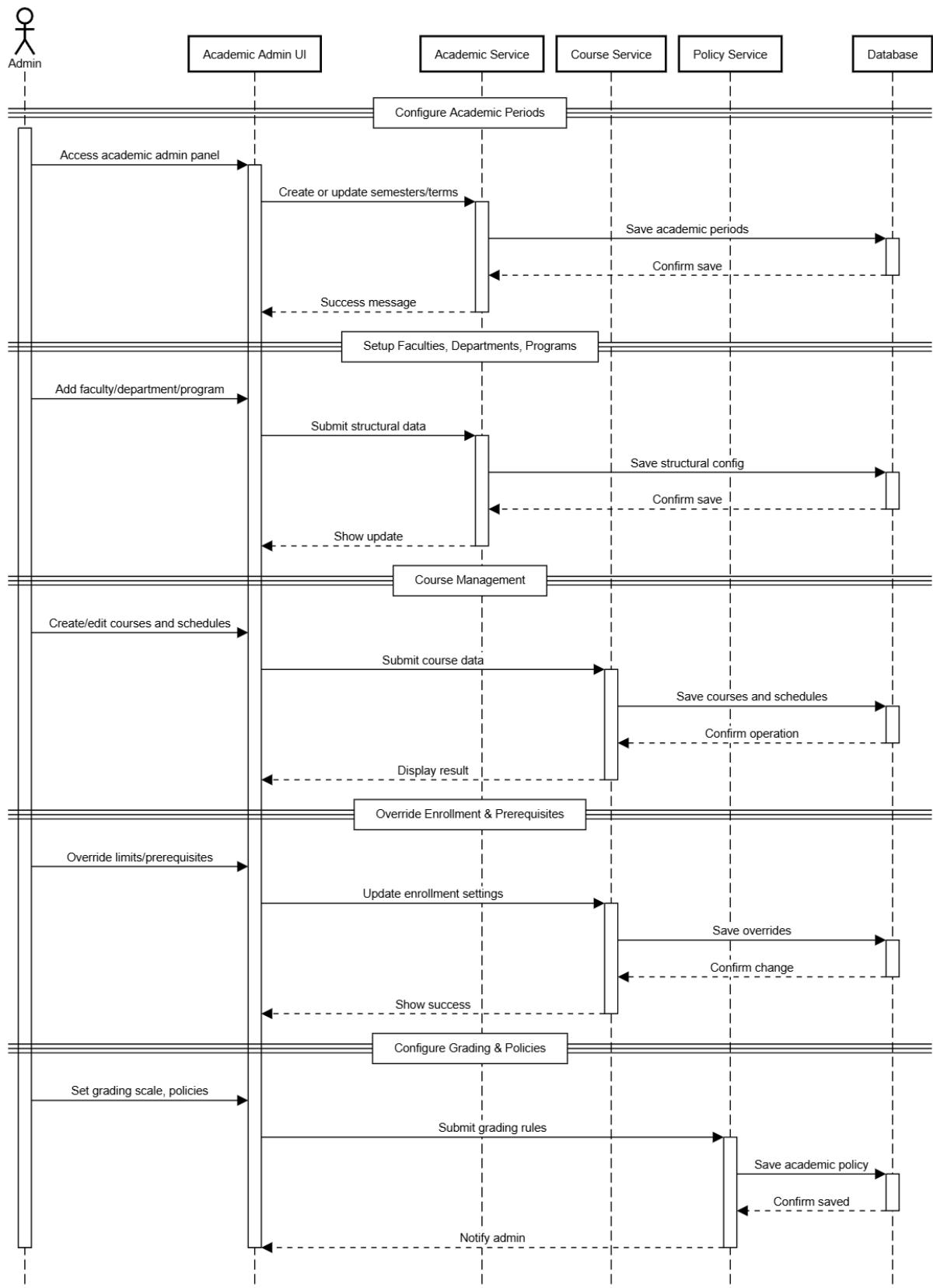


Figure 3.1.2.12 Sequence Diagram for **Manage Academic Administration**

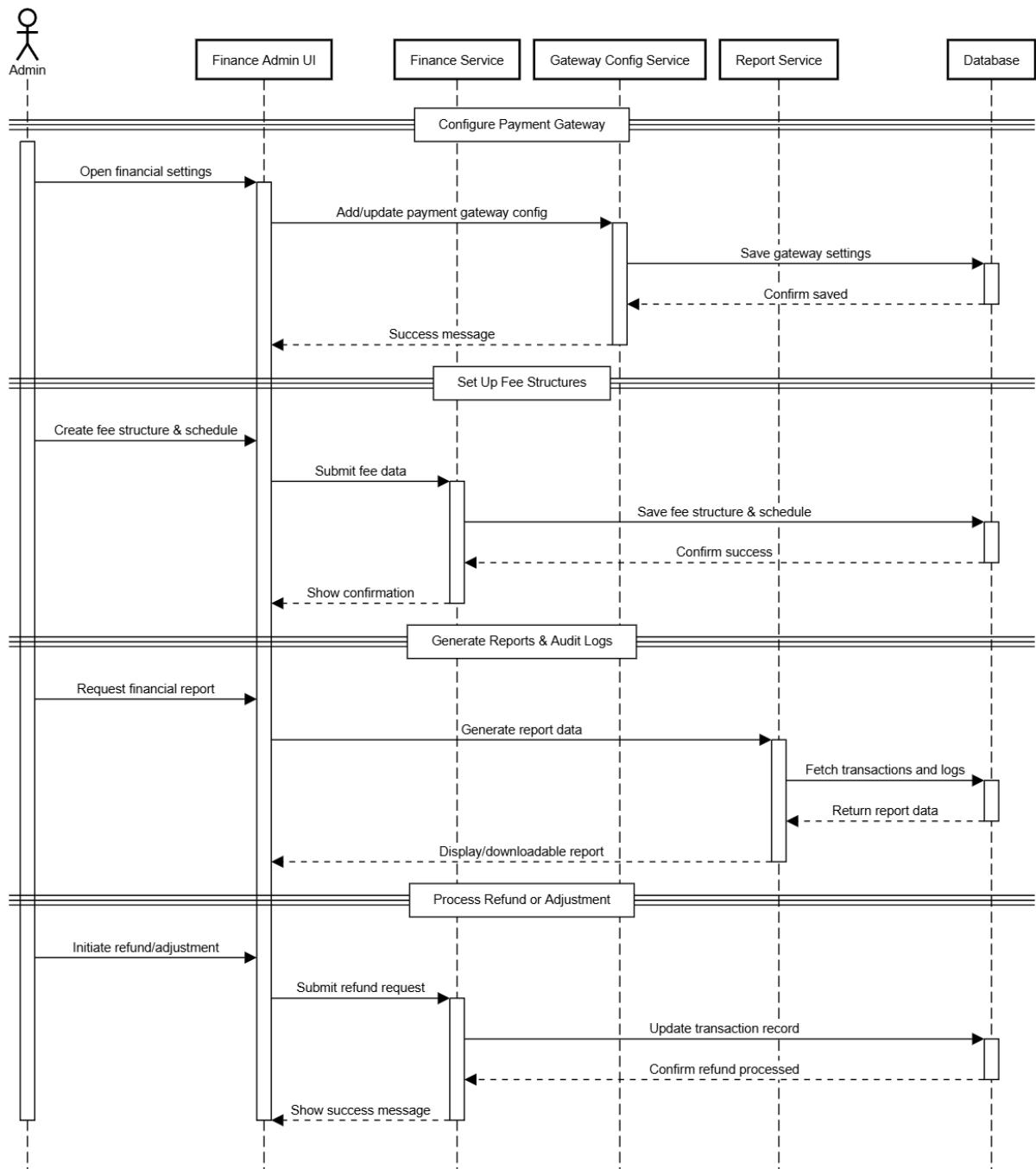


Figure 3.1.2.13 Sequence Diagram for Manage Finance Administration

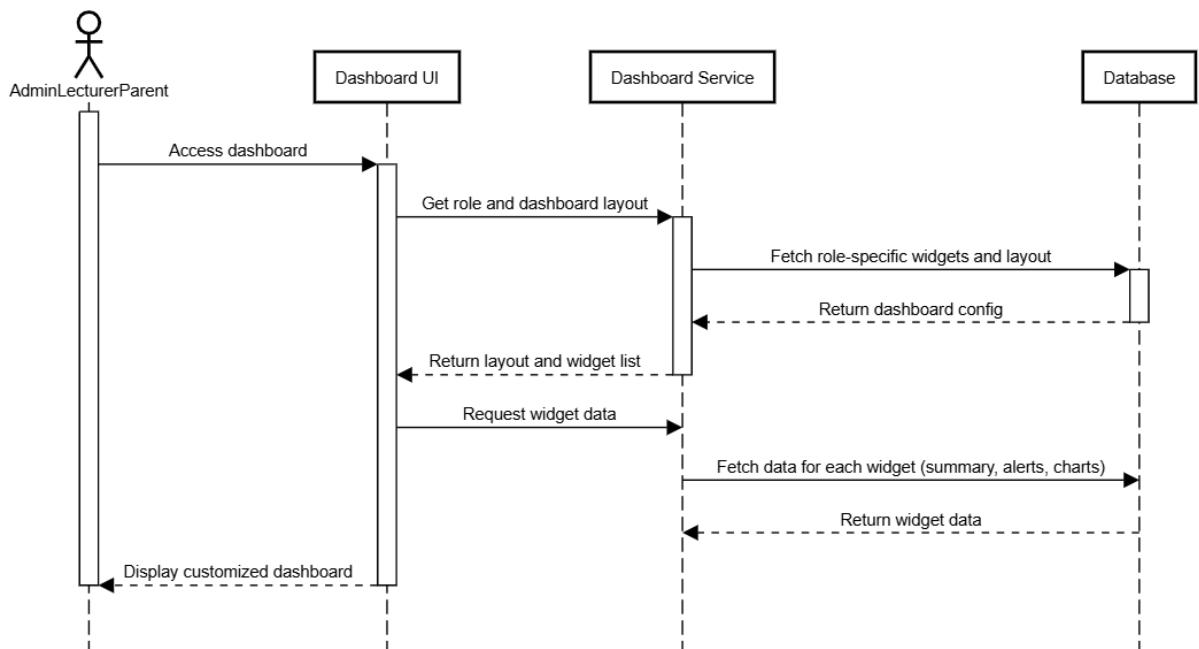


Figure 3.1.2.14 Sequence Diagram for **Dashboard**

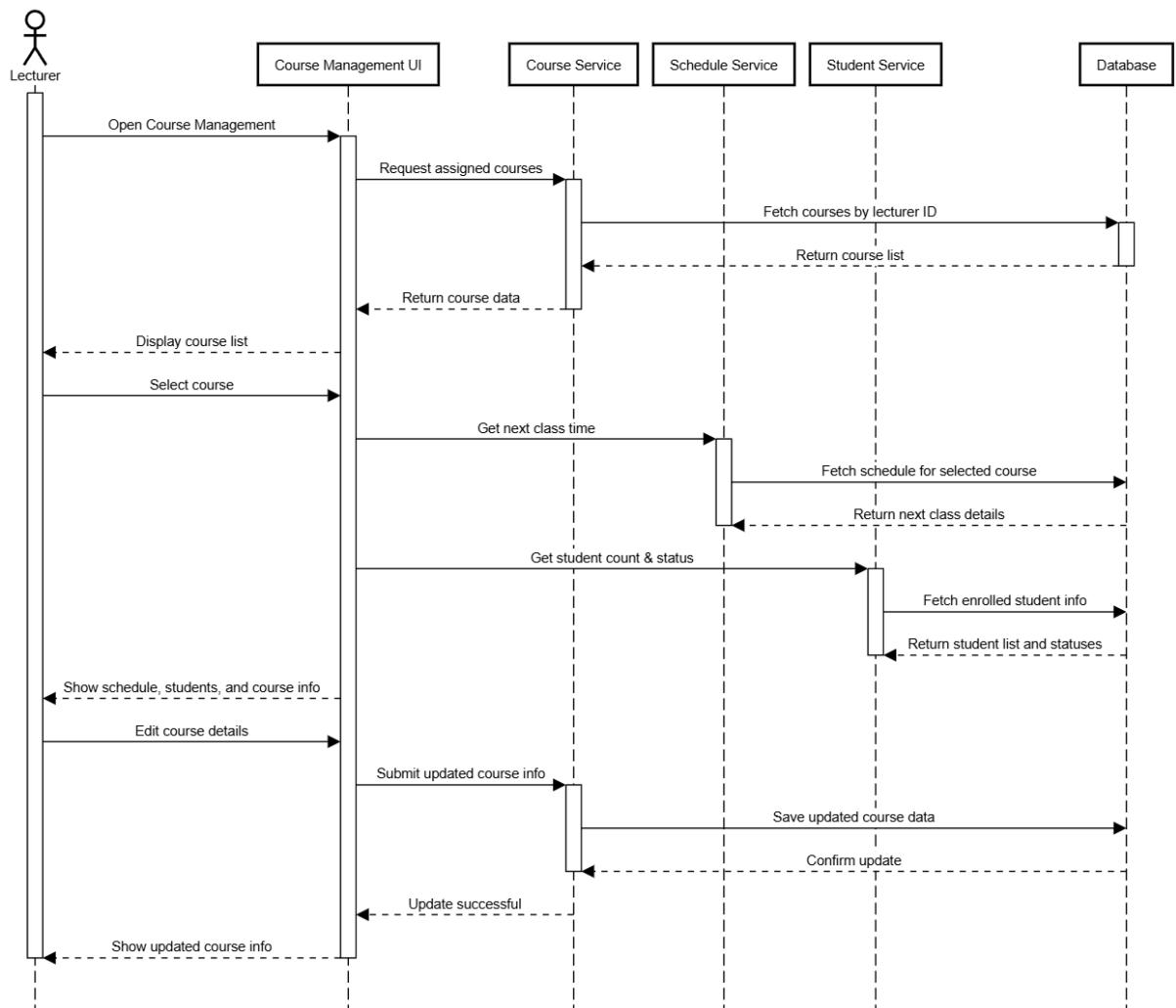
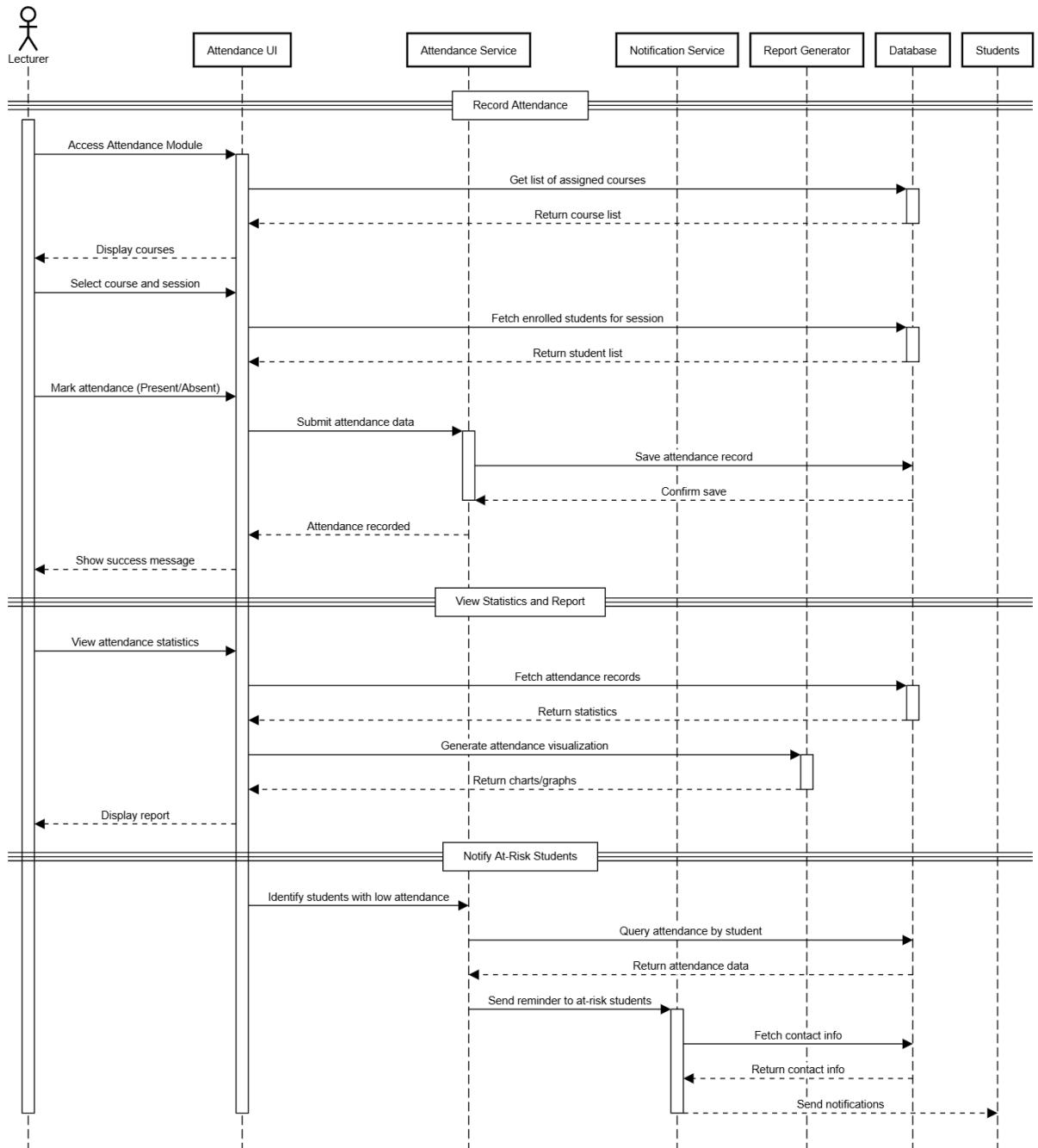


Figure 3.1.2.15 Sequence Diagram for **Manage Course**



*Figure 3.1.2.16 Sequence Diagram for **Track Attendance***

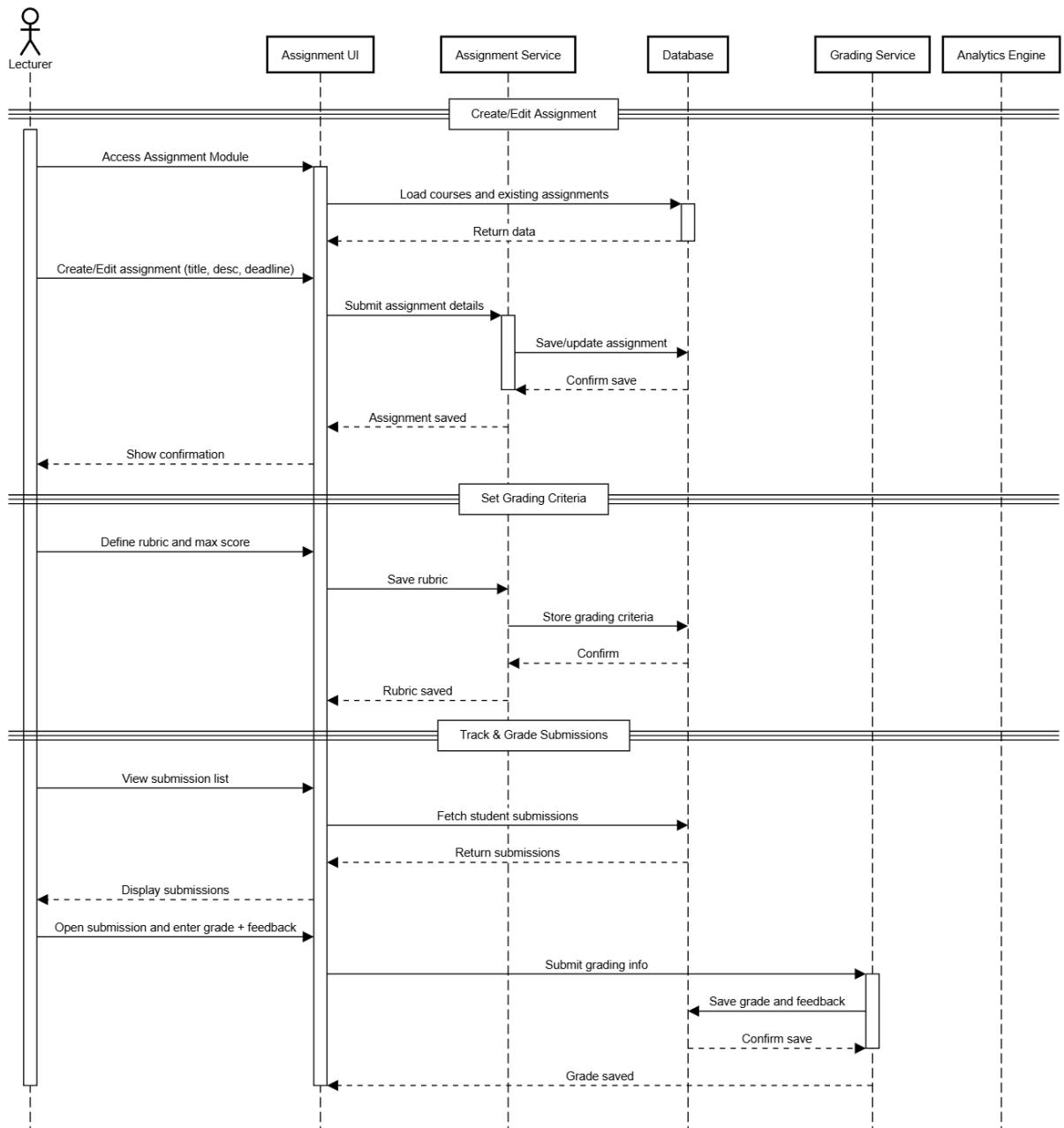


Figure 3.1.2.17 Sequence Diagram for **Manage Assignment**

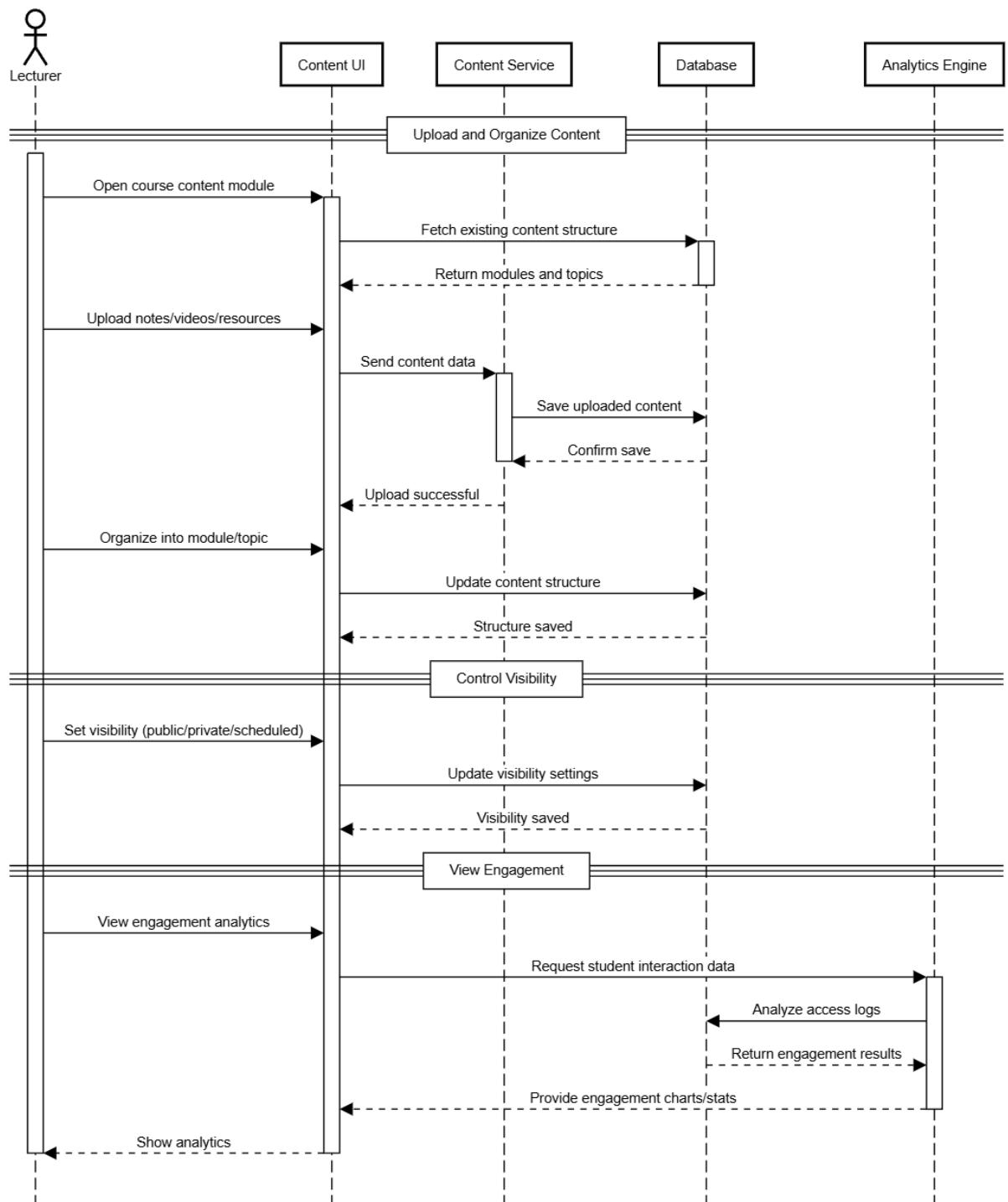


Figure 3.1.2.18 Sequence Diagram for **Create Content**

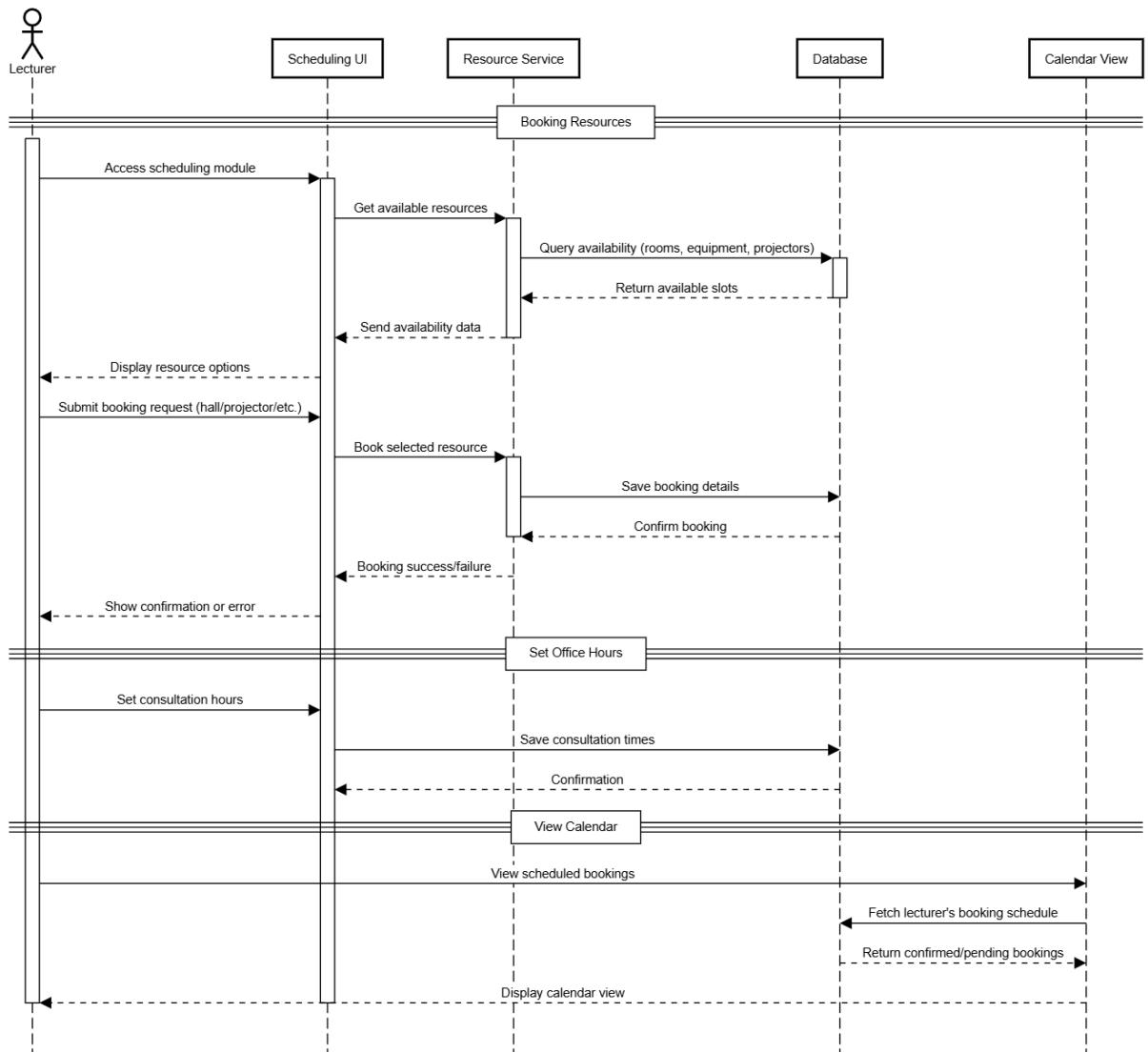


Figure 3.1.2.19 Sequence Diagram for Resource Scheduling

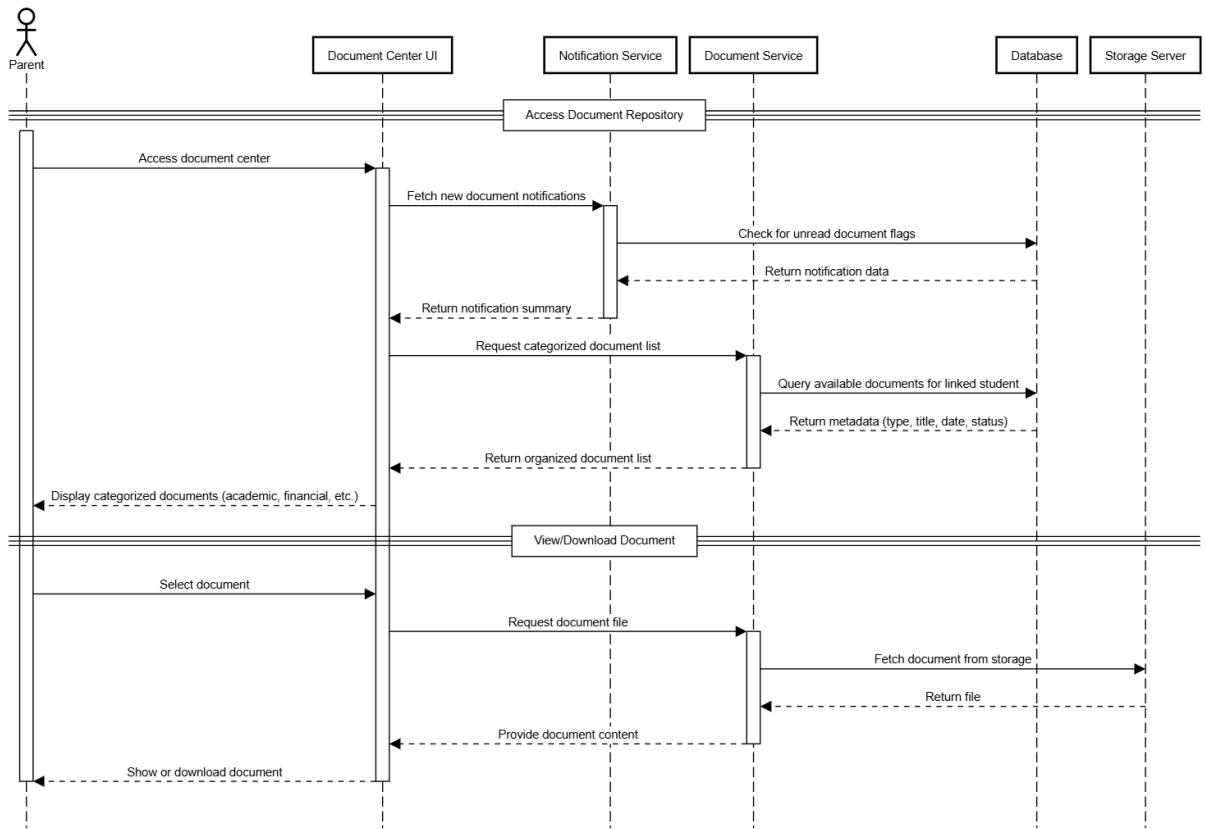


Figure 3.1.2.20 Sequence Diagram for Access Document

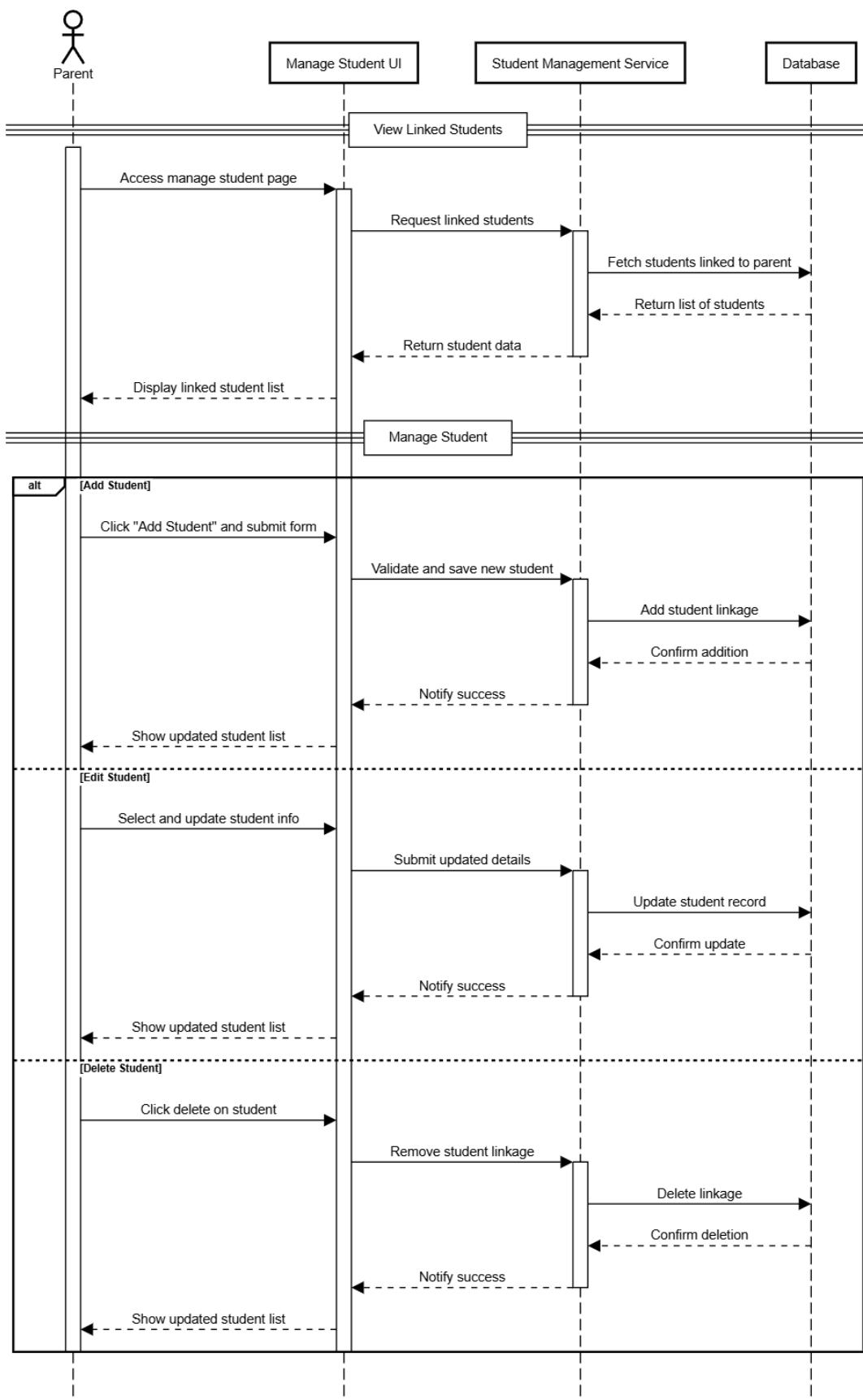


Figure 3.1.2.21 Sequence Diagram for Manage Student

3.2. Performance Requirements

Static Requirements:

Maximum Response Time:

- The SIS/LMS system should respond to user interactions within 5 seconds for operations such as logging in, accessing course materials and submitting assignments

System Downtime:

- The system should have a maximum allowable downtime of 4 hours per month for scheduled maintenance activities.
- Unscheduled downtime should not exceed 1 hour per month

Minimum Hardware and Software Specifications:

- The system should be able to run on hardware with the following minimum specifications:

Desktop:

Processor: Dual-core 2.0 GHz CPU or better

RAM: 4GB

Mobile:

OS: IOS 14 and Android 9

RAM: 2GB

Storage: 100MB

- The system should be able to run on any web browser and search engine such as Google Chrome, Opera, Bing and more

Dynamic Requirements:

Maximum Concurrent Users:

- The system should support a minimum of 1000 concurrent users during peak usage times (e.g., registration periods, exam weeks).

Throughput Requirements:

- The system should manage a minimum of 1000 course registrations per hour during peak registration periods.

Scalability:

- The system should be capable of scaling up to support a 10% increase in concurrent users within a 6-month period without significant degradation in performance

3.3. Usability Requirements

Usability Requirements:

Effectiveness:

- The system shall allow students to easily access their academic information, including grades, schedules, and course materials.
- The system shall enable instructors to efficiently manage course content, assignments, and assessments.

Efficiency:

- The interface shall support auto-fill, dropdown menus, and default values to minimize the time and effort required for data entry.
- The system shall load pages and display data within 3 seconds under normal load conditions.

Satisfaction:

- The portal shall achieve a System Usability Scale (SUS) score of at least 75/100 in usability surveys post-deployment.
- The system shall have an intuitive and user-friendly interface, contributing to user satisfaction.
- The system shall include an optional feedback form or quick rating tool for users to evaluate individual features and gather suggestions for improvement.

Quality in Use Requirements:

Effectiveness:

- The system shall provide accurate and real-time updates for key data such as grades, attendance, and billing, retrieved directly from integrated systems.
- Users shall receive confirmation notifications (via screen and/or email/SMS) for submitted requests or actions.

Efficiency:

- The system shall be accessible on mobile and desktop devices with responsive design, allowing users to access services efficiently in various contexts.
- The system shall respond to user requests within 2 seconds on average, ensuring swift access to information.

Satisfaction:

- User satisfaction surveys shall yield an average score of at least 4 out of 5, showing an elevated level of user satisfaction.

- Multilingual interface options shall be provided to increase satisfaction among non-native English speakers.

Avoidance of Harm:

- The system shall have a documented error rate of less than 1%, minimizing the risk of data loss or corruption
- All irreversible or critical actions (e.g., deleting requests, logging out, submitting forms) shall require explicit user confirmation.
- Sensitive personal and academic data shall be protected using encryption and secure login mechanisms, with access limited based on user roles.
- To prevent session hijacking, automatic logout after 30 minutes of inactivity shall be enforced.

3.4. Interface Requirements

3.4.1. System Interfaces

- 3.4.1.1. Student Information System (SIS)
- 3.4.1.2. Learning Management System (LMS)
- 3.4.1.3. University Authentication System
- 3.4.1.4. Financial Management System
- 3.4.1.5. SMS Gateway

3.4.2. User Interfaces

3.4.2.1. Student

- Login pages

Figure 3.4.2.1.1 displays the Login Pages, it is designed to save login credentials in cookies, so that students do not need to enter their username and password each time they log in.

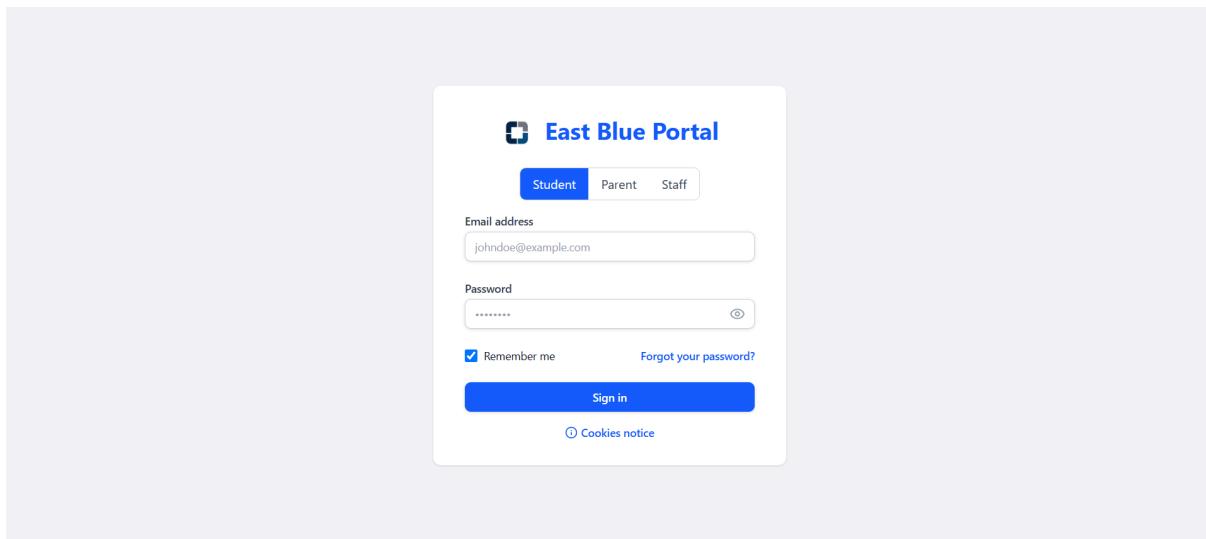


Figure 3.4.2.1.1 Login Pages

- Desktop Dashboard Page

After logging in, students are directed to a personalized dashboard Figures 3.4.2.1.2 and 3.4.2.1.3. The dashboard has navigation tabs and relevant student information. It also integrates an AI assistant that allows students to ask questions via chat. For instance, students can ask, "When is the SRE assignment due?" and the assistant will respond with the appropriate deadline or information. The customizable layout is similar to the iPhone's widget system, where users can resize components and personalize their dashboard by adding elements such as the timetable, user profile, and today's schedule, as shown in Figures 3.4.2.1.2 and 3.4.2.1.3.

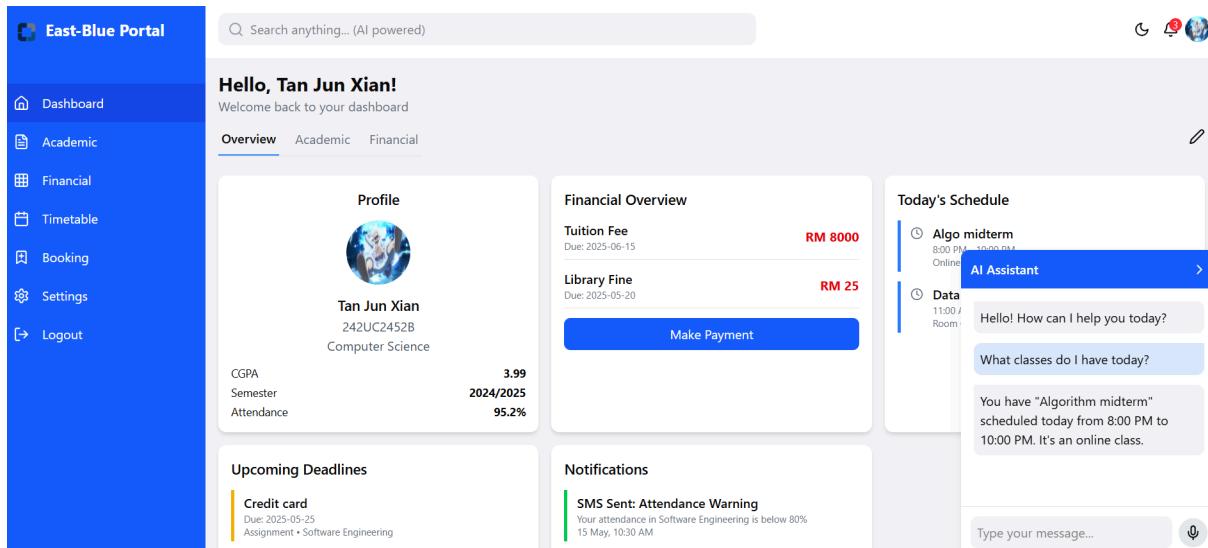


Figure 3.4.2.1.2 Desktop Dashboard Page

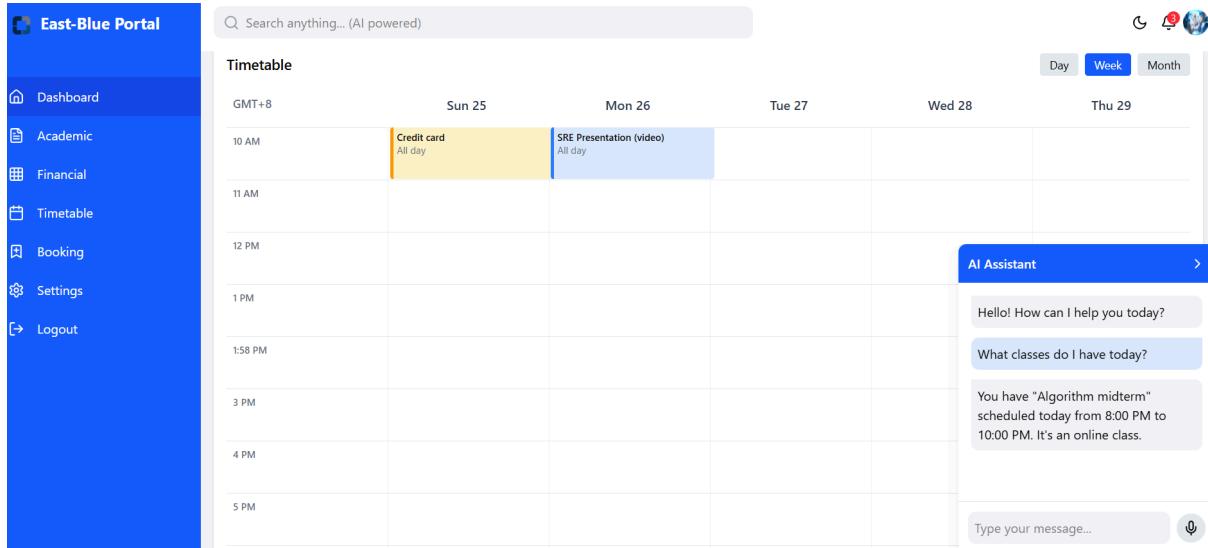
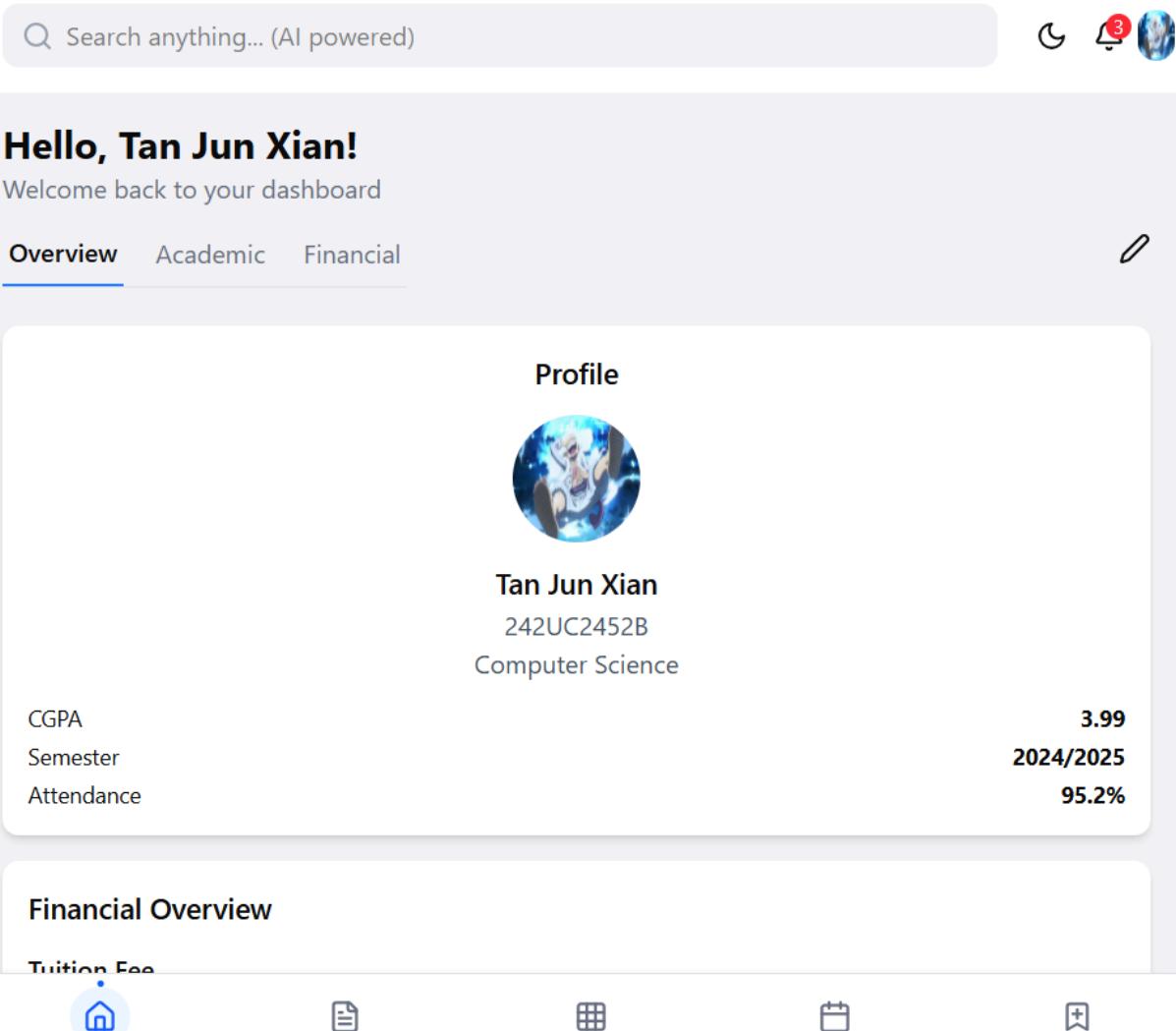


Figure 3.4.2.1.3 Desktop Dashboard Page

- Mobile Dashboard Page

Figure 3.4.2.1.4 displays the mobile dashboard page, which has the same layout and features as Figures 3.4.2.1.2 and 3.4.2.1.3 but is optimized for viewing within the mobile application.



The screenshot shows a mobile dashboard for a user named Tan Jun Xian. At the top, there is a search bar with the placeholder "Search anything... (AI powered)" and a notification badge with the number "3". Below the search bar, the greeting "Hello, Tan Jun Xian!" is displayed, followed by the message "Welcome back to your dashboard". A navigation menu includes tabs for "Overview" (which is underlined in blue), "Academic", and "Financial", along with a pencil icon for editing.

Profile

Tan Jun Xian
242UC2452B
Computer Science

CGPA	3.99
Semester	2024/2025
Attendance	95.2%

Financial Overview

Tuition Fee

Book Academic Finance Schedule Book

Figure 3.4.2.1.4 Mobile Dashboard Page

- Desktop Academic Page

Figures 3.4.2.1.5 and 3.4.2.1.6 display the desktop view of the Academic page. The interface includes a complete academic record showing both current and past grades, along with a CGPA progression chart that visualizes performance across semesters. It also features real-time course enrollment status, a degree progress tracker, a detailed breakdown of attendance for all enrolled courses, and a section highlighting academic achievements and recognitions.

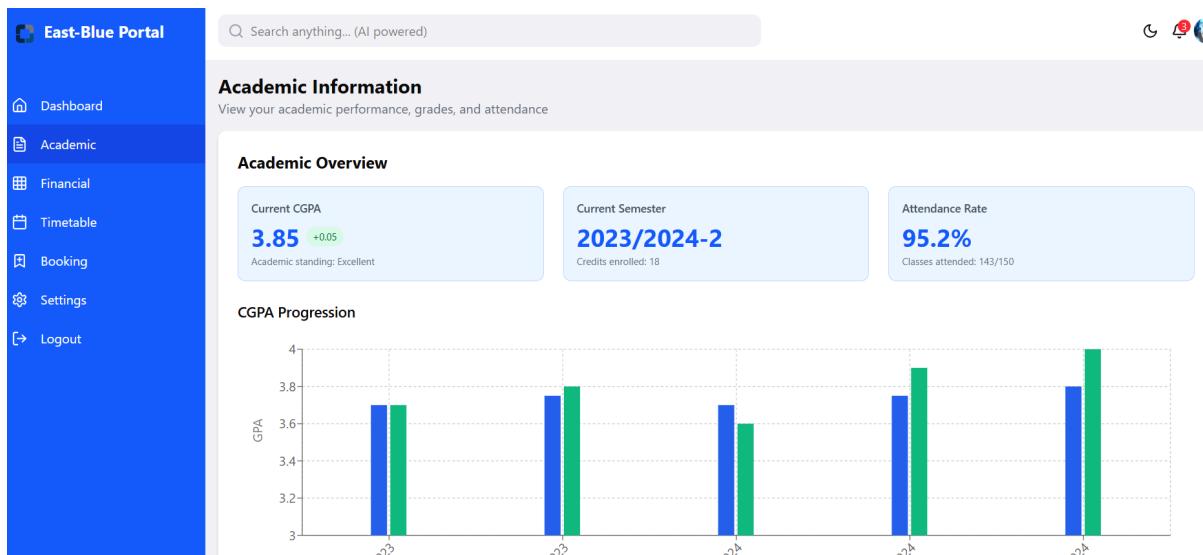


Figure 3.4.2.1.5 Desktop Academic Page

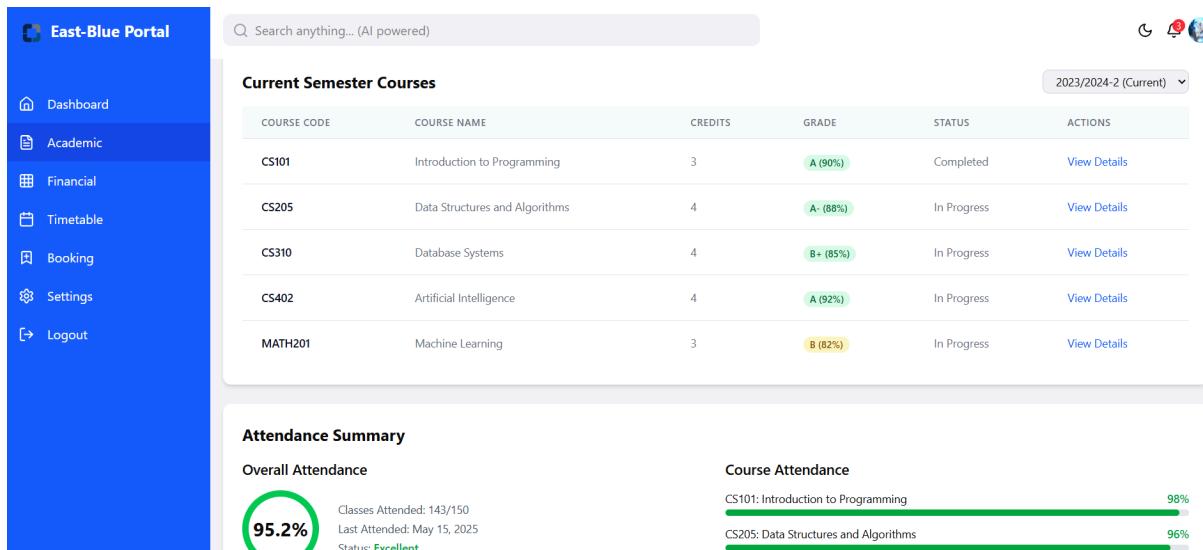


Figure 3.4.2.1.6 Desktop Academic Page

- Mobile Academic Page

Figure 3.4.2.1.7 displays the mobile academic page, which provides the same features as shown in Figures 3.4.2.1.5 and 3.4.2.1.6. However, it is optimized for viewing within the mobile application.

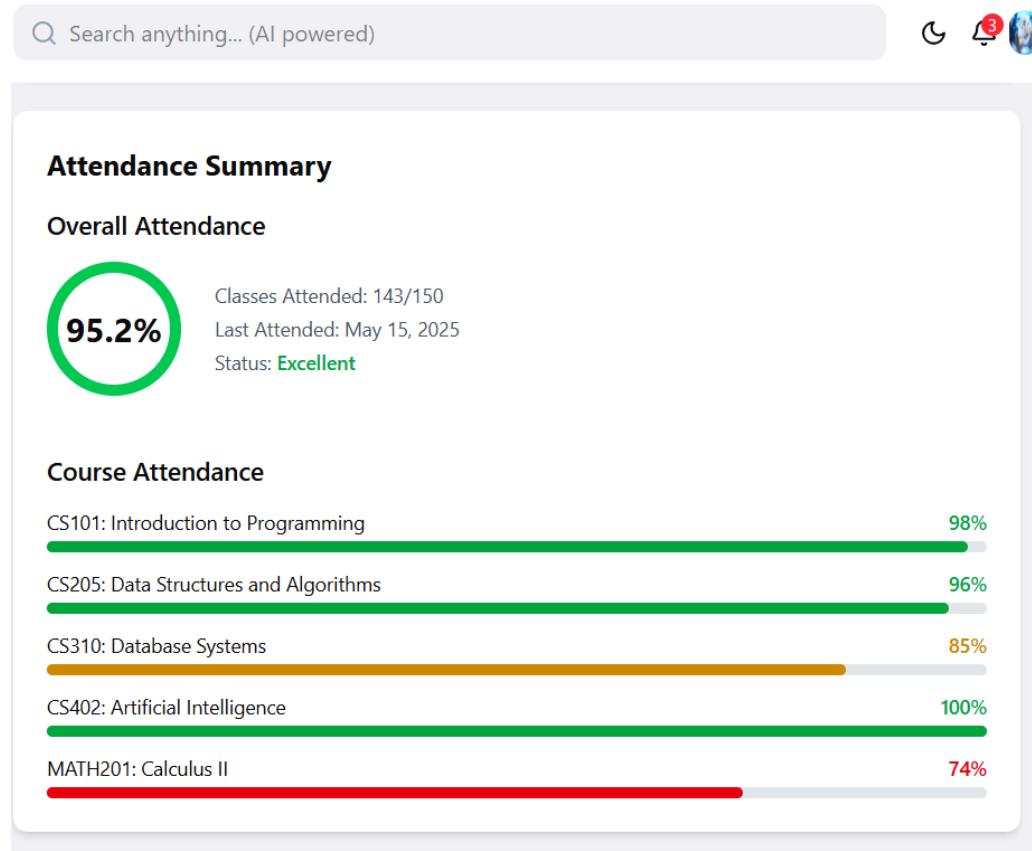


Figure 3.4.2.1.7 Mobile Academic Page

- Desktop Financial Page

Figures 3.4.2.1.8 and 3.4.2.1.9 display the Desktop Financial Page, which offers comprehensive money management features including balance overview with pending payments, downloadable payment history receipts, multiple online payment options, financial aid information, and payment deadline notifications.

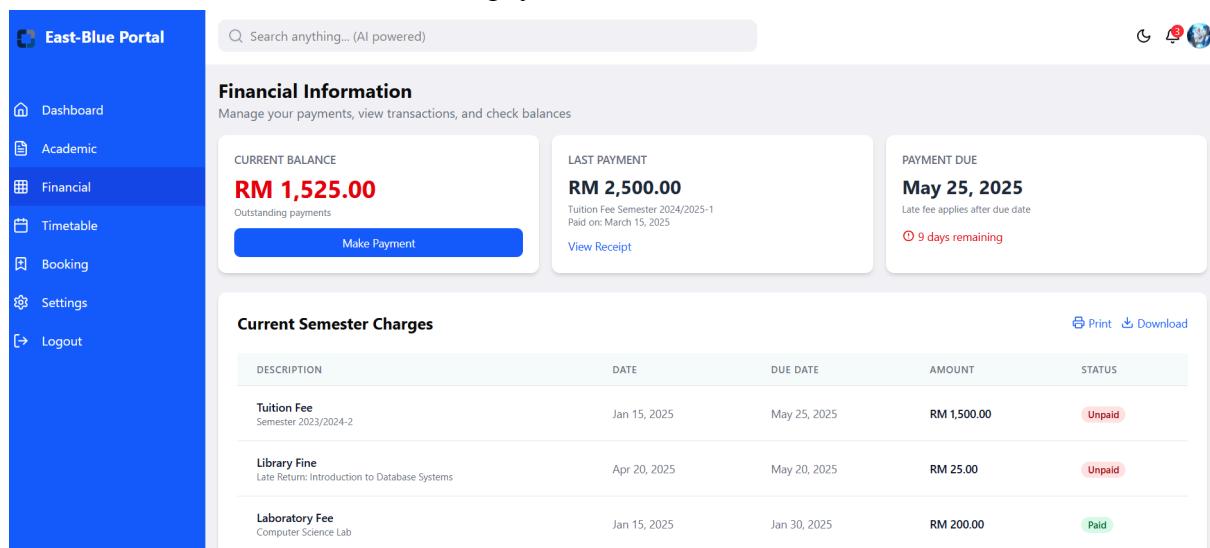


Figure 3.4.2.1.8 Desktop Financial Page

The screenshot shows the East-Blue Portal desktop interface. On the left is a blue sidebar with navigation links: Dashboard, Academic, Financial (which is selected), Timetable, Booking, Settings, and Logout. The main area has a search bar at the top. Below it are two sections: 'Payment History' and 'Invoice'. Both sections have a header with a dropdown for 'All Semesters' and an 'Export' button. The 'Payment History' section lists three transactions: TRX-25789 (Tuition Fee, Mar 15, 2025, RM 2,500.00, Credit Card), TRX-23451 (Laboratory Fee, Jan 28, 2025, RM 200.00, Online Banking), and TRX-21987 (Tuition Fee, Sep 10, 2024, RM 2,500.00, Bank Transfer). The 'Invoice' section lists two invoices: INV-56173 (Tuition Fee, July 12, 2024, RM 7,532.00, Credit Card) and INV-61534 (Laboratory Fee, Jan 04, 2024, RM 6,532.00, Online Banking).

Figure 3.4.2.1.9 Desktop Financial Page

- **Mobile Financial Page**

Figure 3.4.2.1.10 displays the mobile Financial Page, which provides the same features as shown in Figures 3.4.2.1.8 and 3.4.2.1.9. However, it is optimized for viewing within the mobile application.

The screenshot shows the East-Blue Portal mobile interface. At the bottom are five navigation icons: Home, Academic, Financial (selected), Schedule, and Book. The main area has a search bar at the top. Below it are two sections: 'Payment History' and 'Invoice'. Both sections have a header with a dropdown for 'All Semesters' and an 'Export' button. The 'Payment History' section lists the same three transactions as the desktop version. The 'Invoice' section lists the same two invoices as the desktop version. A horizontal scrollbar is visible between the two sections.

Figure 3.4.2.1.10 Mobile Financial Page

- Desktop Timetable Page

Figures 3.4.2.1.11 and 3.4.2.1.12 display the Desktop Timetable Page, featuring an interactive calendar with daily, weekly, or monthly views, color-coded events for classes, exams, and deadlines, schedule export capabilities, and visual indicators for upcoming events. Figures 3.4.2.1.13 and 3.4.2.1.14 display the Desktop Timetable Page, showcasing the auto-enrollment feature that recommends optimal course schedules. Users can set preferences such as minimum and maximum credits, required and recommended courses, and optional schedules.

The screenshot shows the East-Blue Portal's desktop timetable interface. On the left is a blue sidebar with navigation links: Dashboard, Academic, Financial, Timetable (which is selected and highlighted in blue), Booking, Settings, and Logout. At the top right are user icons for profile, notifications, and help. A search bar at the top center contains the placeholder "Search anything... (AI powered)". Below the search bar is a "Timetable" section header with the sub-instruction "View your class schedule and manage academic events". A date range selector shows "May 16 - May 22, 2025" with arrows to change the range. To the right are buttons for "Day" (selected), "Week", "Month", a dropdown for "All Courses", and an "Export" button. The main area displays a daily timeline from 8:00 AM to 4:00 PM. At 10:00 AM, there is a blue box for "CS205: Data Structures and Algorithms" with details: "10:00 AM - 12:00 PM", "Room CNMX 0001", and "Professor Michael Chen". At 2:00 PM, there is a green box for "MATH201: Machine Learning" with details: "2:00 PM - 4:00 PM", "Room M-105", and "Dr. Lisa Wong". The "Lecture" and "Tutorial" labels are placed next to their respective boxes.

Figure 3.4.2.1.11 Desktop Timetable Page

This screenshot shows the same East-Blue Portal interface as Figure 3.4.2.1.11, but with a different focus. The sidebar and top navigation are identical. The main content area now features an "Upcoming Events" section with three items: "Machine Learning assignment deadline" (Software Engineering, May 25, 2025, Assignment), "SRE Presentation submission" (Requirements Engineering, May 26, 2025, Presentation), and "Algorithm Project Submission" (CS310: Database Systems, May 30, 2025, Project). Below this is a "Weekly Schedule Preview" table:

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00 AM					
10:00 AM	CS205 10:00 - 12:00		CS205 10:00 - 12:00		
11:00 AM		ENG102 11:00 - 12:30		ENG102 11:00 - 12:30	
12:00 PM					

Figure 3.4.2.1.12 Desktop Timetable Page

The screenshot shows the East-Blue Portal's desktop interface for creating a timetable. On the left is a blue sidebar with navigation links: Dashboard, Academic, Financial, Timetable, Booking, Settings, and Logout. The main content area has a search bar at the top. Below it is a section titled "Your Enrollment Preferences" with fields for Academic Term (Fall 2025), Preferred Time Slots (Morning, Afternoon checked, Evening unchecked), Minimum Credits (12), Maximum Credits (18), and Additional Preferences (Compact Schedule checked, Free Day Preference and Same Instructor When Possible unchecked). A second section titled "Required & Recommended Courses" lists three courses: CS205: Data Structures and Algorithms (selected, required, 2 available), MATH201: Machine Learning (not selected, recommended, 2 available), and ENG102: Technical Writing (selected, required, 1 available).

Figure 3.4.2.1.13 Desktop Timetable Page

This screenshot shows the final optimized schedule from the desktop portal. It lists three courses: CS205: Data Structures and Algorithms (selected, required, 12 seats available), MATH201: Machine Learning (selected, 15 seats available), and ENG102: Technical Writing (selected, 20 seats available). Below the schedule, a message states: "This schedule was optimized based on your preferences and degree requirements." A blue "Enroll Now" button is visible.

Figure 3.4.2.1.14 Desktop Timetable Page

- Mobile Timetable Page

Figure 3.4.2.1.15 displays the mobile Timetable Page, which provides the same features as shown in Figures 3.4.2.1.11 until Figures 3.4.2.1.14. However, it is optimized for viewing within the mobile application.

Algorithm Project Submission
CS310: Database Systems
May 30, 2025

Weekly Schedule Preview

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00 AM					
10:00 AM	CS205 10:00 - 12:00		CS205 10:00 - 12:00		
11:00 AM		ENG102 11:00 - 12:30		ENG102 11:00 - 12:30	
12:00 PM					
1:00 PM					
2:00 PM					MATH201 2:00 - 3:30
3:00 PM					
4:00 PM					

Home Academic Finance **Schedule** Book

Figure 3.4.2.1.15 Mobile Timetable Page

- Booking Page
- Figures 3.4.2.1.16 and 3.4.2.1.17 display the Desktop Booking Page, featuring lecturer consultation scheduling with available time slots, classroom and study space reservations, resource booking capabilities, appointment confirmations with reminders, and comprehensive booking history tracking.

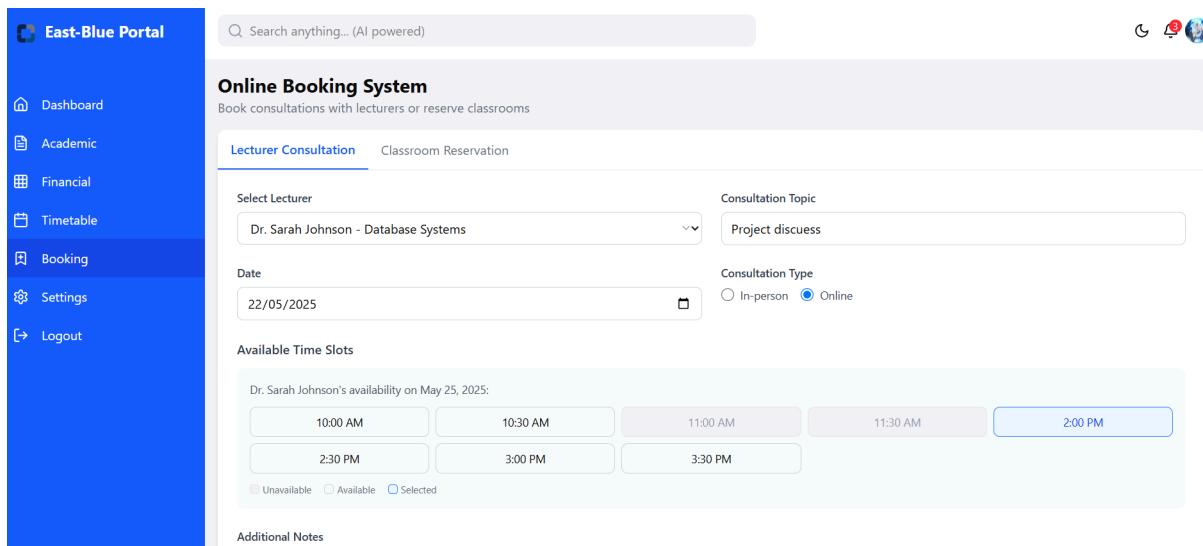


Figure 3.4.2.1.16 Desktop Booking Page

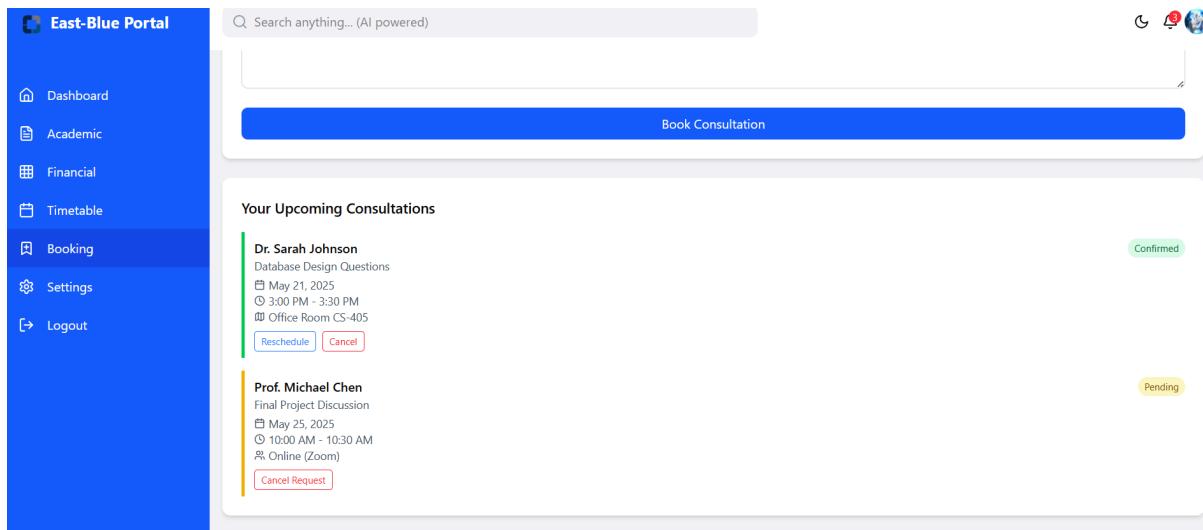


Figure 3.4.2.1.17 Desktop Booking Page

- **Mobile Booking Page**

Figure 3.4.2.1.18 displays the mobile Booking Page, which provides the same features as shown in Figures 3.4.2.1.16 and 3.4.2.1.17. However, it is optimized for viewing within the mobile application.

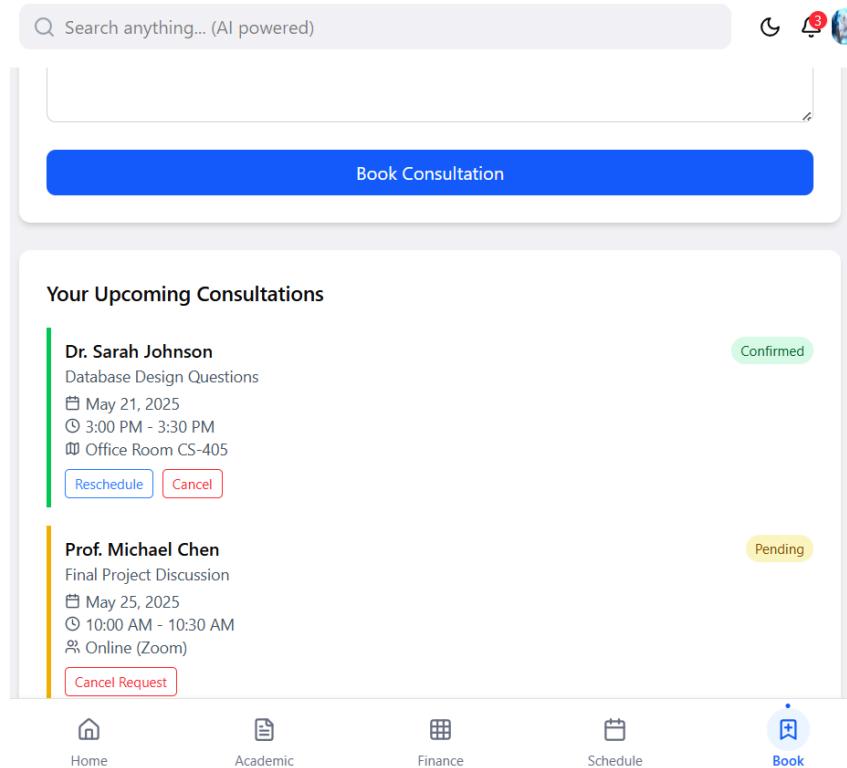


Figure 3.4.2.1.18 Mobile Booking Page

- Desktop Settings

Figures 3.4.2.1.19 through 3.4.2.1.23 display the Desktop Settings Pages, offering comprehensive profile management capabilities, customizable notification preferences, appearance settings including theme options, language and regional preferences, privacy and security controls, and account access management features.

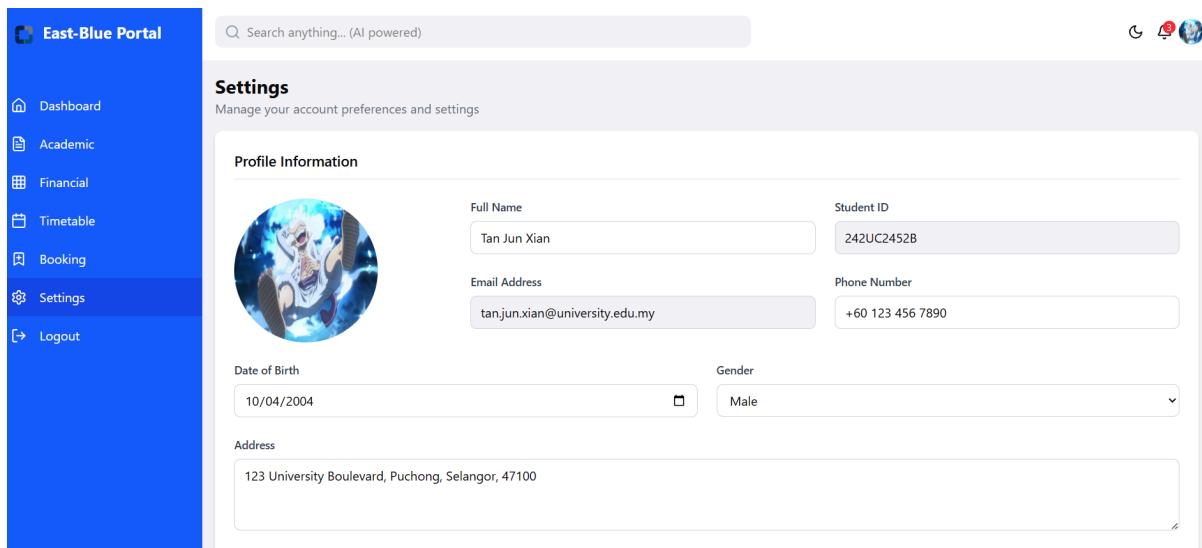


Figure 3.4.2.1.19 Desktop Settings

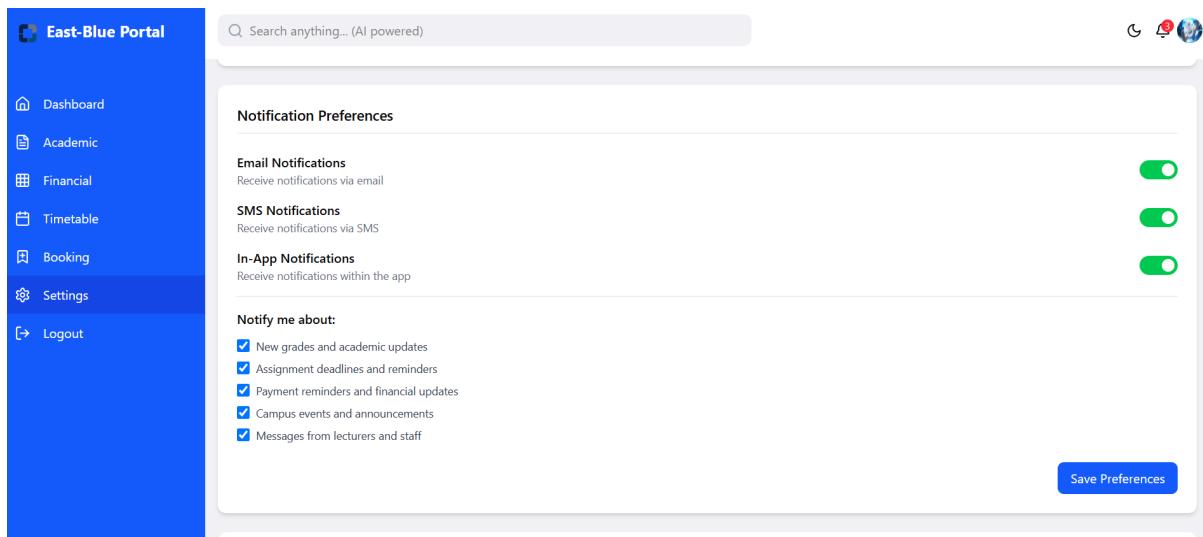


Figure 3.4.2.1.20 Desktop Settings

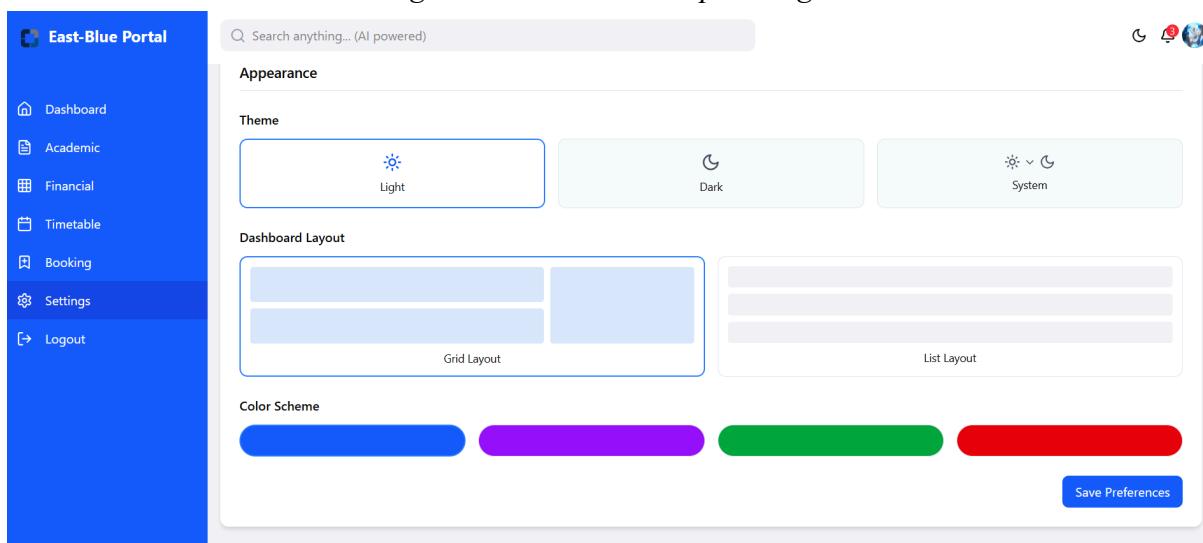


Figure 3.4.2.1.21 Desktop Settings

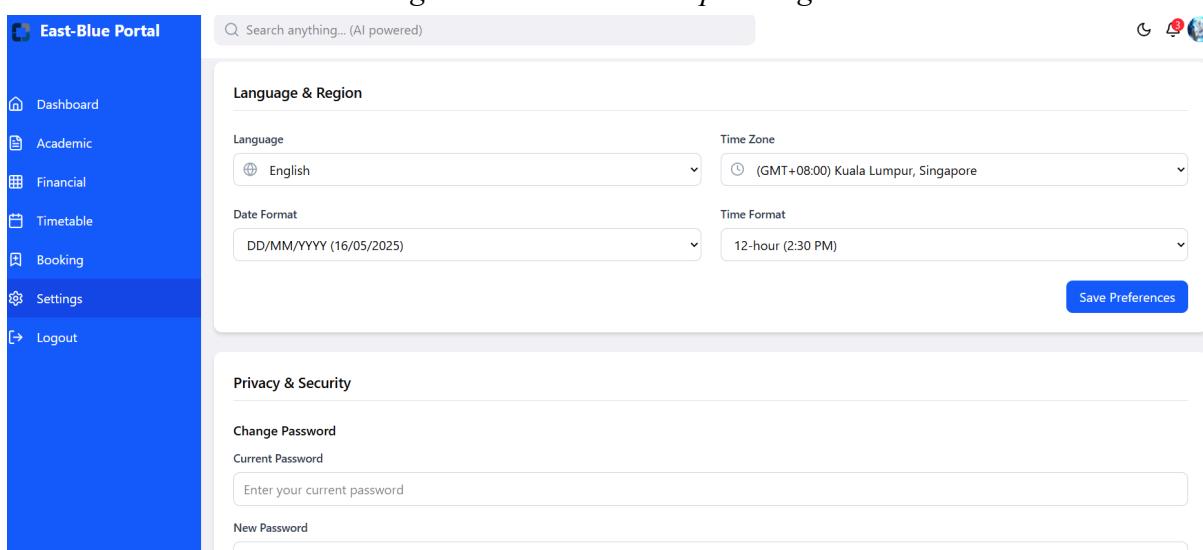


Figure 3.4.2.1.22 Desktop Settings

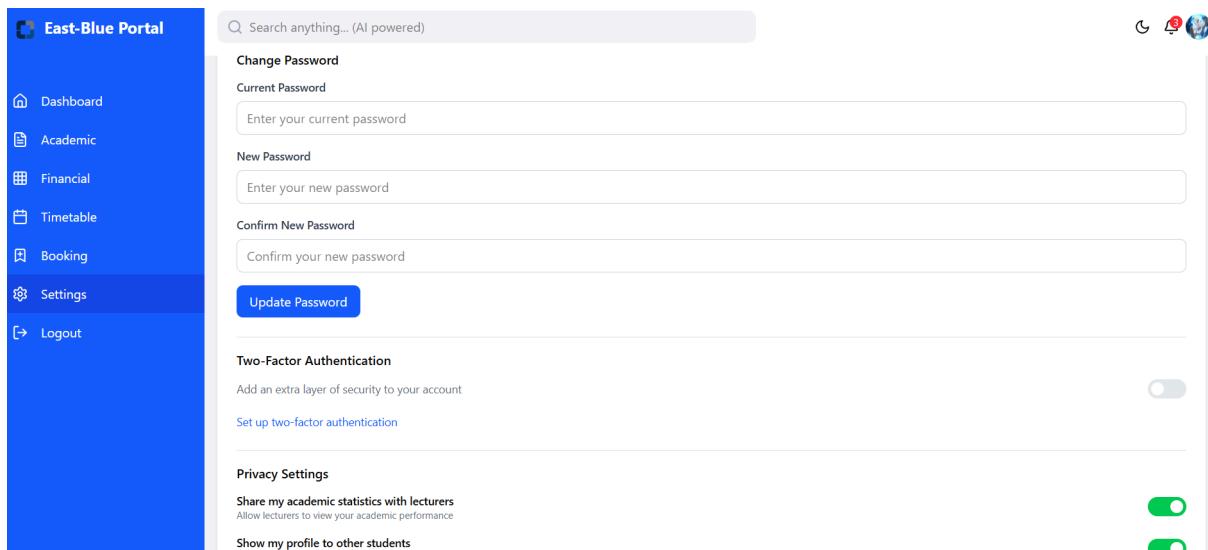


Figure 3.4.2.1.23 Desktop Settings

- **Mobile Settings**

Figure 3.4.2.1.24 displays the mobile Settings Page, which provides the same features as shown in Figures 3.4.2.1.19 through 3.4.2.1.23. However, it is optimized for viewing within the mobile application.

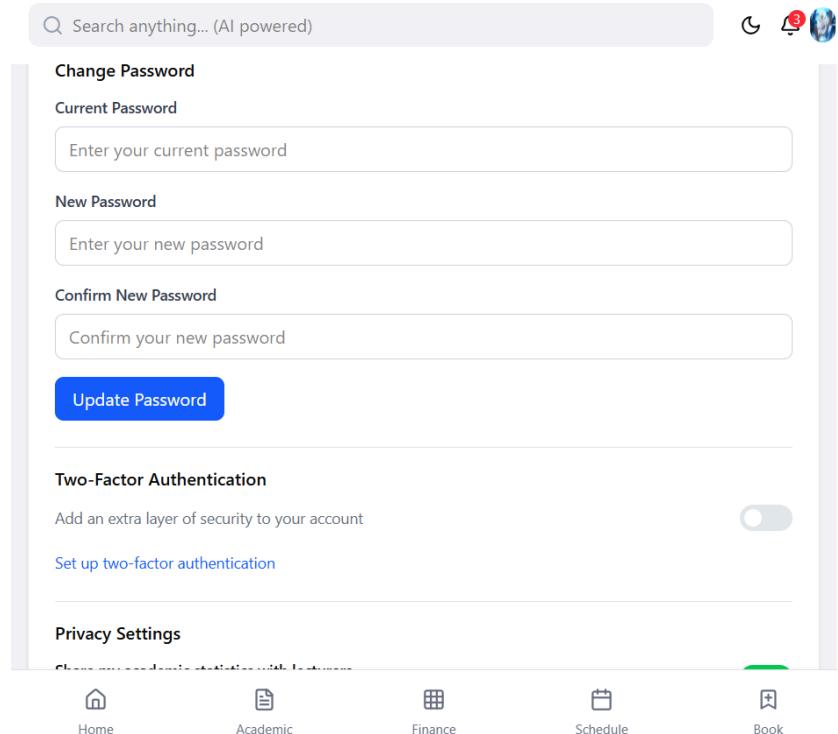


Figure 3.4.2.1.24 Mobile Settings

3.4.2.2. Lecturer

- **Dashboard**

Figures 3.4.2.2.1 and 3.4.2.2.2 display the Dashboard Pages, providing quick access to key metrics including courses, students, attendance rates, and pending assignments, along with upcoming events, schedules, and comprehensive course attendance and assignment submission statistics.

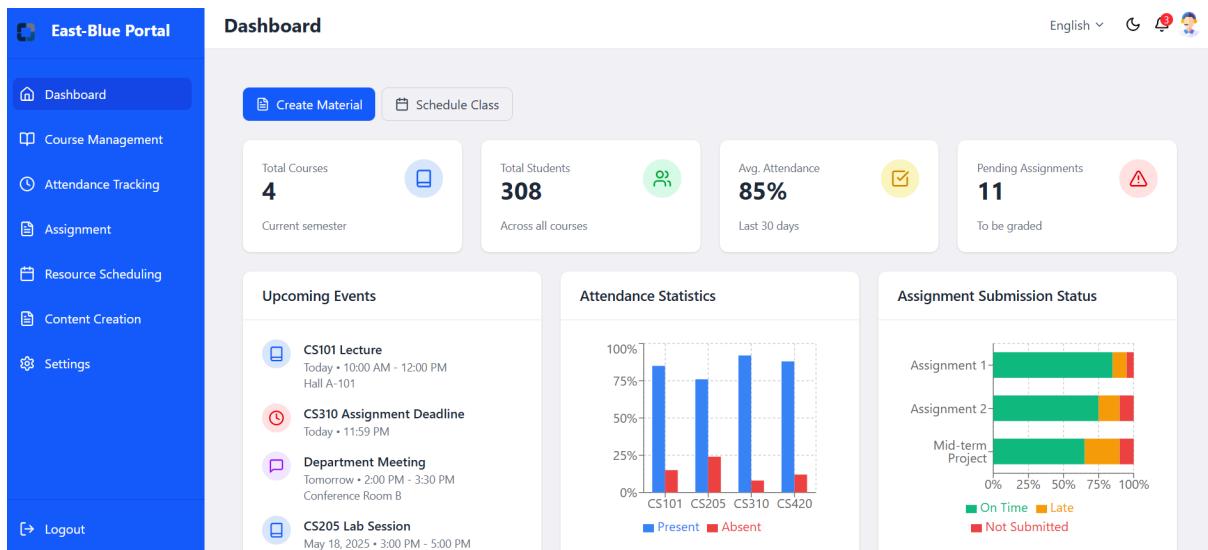


Figure 3.4.2.2.1 Dashboard

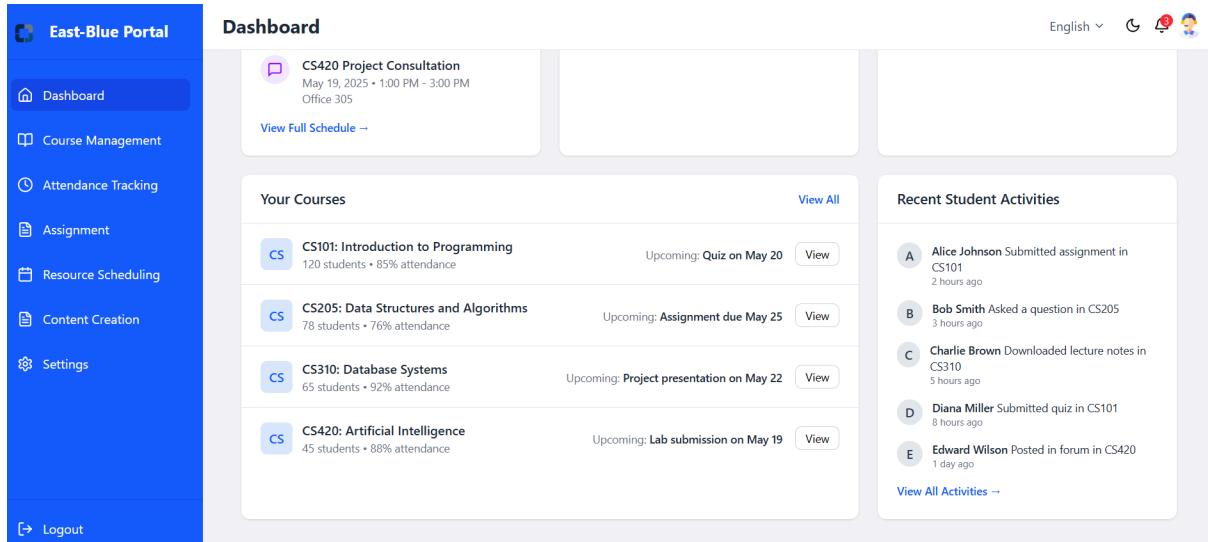


Figure 3.4.2.2.2 Dashboard

● Course Management

Figure 3.4.2.2.3 displays the Course Management Page, featuring tools to manage courses, view and edit upcoming class schedules, and monitor student enrollment with status information.

COURSE	SEMESTER	STATUS	STUDENTS	PROGRESS	NEXT CLASS	ACTIONS
CS101 Introduction to Programming	Semester 1, 2025	Active	120 12 modules	65% Complete	Today, 10:00 AM	Edit ...
CS205 Data Structures and Algorithms	Semester 1, 2025	Active	78 15 modules	45% Complete	Tomorrow, 2:00 PM	Edit ...
CS310 Database Systems	Semester 1, 2025	Active	65 10 modules	70% Complete	May 19, 11:00 AM	Edit ...
CS420 Artificial Intelligence	Semester 1, 2025	Active	45 14 modules	50% Complete	May 20, 3:00 PM	Edit ...
CS250 Computer Graphics	Semester 2, 2025	Upcoming	0 8 modules	0% Complete	Starts Aug 15, 2025	Edit ...
CS115 Web Development Fundamentals	Semester 3, 2024	Archived	95 12 modules	100% Complete	Completed	Edit ...

Figure 3.4.2.2.3 Course Management

- Attendance Tracking

Figures 3.4.2.2.4 and 3.4.2.2.5 display the Attendance Tracking Pages, featuring tools to record and monitor student attendance, view course-specific statistics, identify at-risk students with attendance issues, send automated reminders, and generate comprehensive attendance reports with visualizations.

Course	Date	Present	Absent	Rate
CS101	May 16, 2025	68	7	91%
CS205	May 15, 2025	55	7	89%
CS310	May 15, 2025	79	4	95%
CS420	May 14, 2025	82	6	93%

Figure 3.4.2.2.4 Attendance Tracking Pages

The screenshot shows the 'Attendance' section of the East-Blue Portal. At the top, it displays a message: 'Tomorrow • 9:00 AM - 11:00 AM Hall B-305'. Below this is a table titled 'Attendance by Course' for 'Introduction to Computer Science'. The table includes columns for Student ID, Name, Attendance Rate (represented by a progress bar), Last Attended, and Status. The status column uses color-coded circles: green for Present, yellow for Absent, and red for Late. The table shows five students with their respective attendance rates and last attended dates. At the bottom, it says 'Showing 5 of 75 students' with navigation links for 'Previous' and 'Next'.

Figure 3.4.2.2.5 Attendance Tracking Pages

- Assignment Pages

Figures 3.4.2.2.6 and 3.4.2.2.7 display the Assignment Pages, featuring tools to create and manage assignments with deadlines, establish grading criteria and rubrics, track submission status, provide student feedback, and analyze performance across different assignments.

The screenshot shows the 'Assessment' section of the East-Blue Portal. It features a table titled 'Assignments' with columns for Title, Course, Due Date, Status, Type, Submission, and Actions. The table lists four assignments: 'Database Normalization' (ASGN1001, CS310, May 25, 2025, Open, Individual, 65/83 graded), 'Sorting Algorithms Implementation' (ASGN1002, CS205, May 28, 2025, Open, Individual, 45/62 graded), 'Mid-term Project' (ASGN1003, CS101, May 20, 2025, Closed, Group, 72/75 graded), and 'Neural Network Design' (ASGN1004, CS420, June 05, 2025, Draft, Individual, 0/88 graded). Below the table are two tabs: 'Assignment Details' and 'Submissions'. The 'Assignment Details' tab is active, showing 'Database Normalization'.

Figure 3.4.2.2.6 Assignment Pages

Assignment Details

Database Normalization
ASGN1001

Course: Database Systems (CS310)

Due Date: May 25, 2025

Status: Open

Type: Individual

Total Points: 100

Description:
This assignment focuses on database normalization concepts. Students will analyze a given database schema and normalize it to Third Normal Form (3NF). They should identify functional dependencies, apply normalization rules, and justify their design decisions.

Submissions

STUDENT	SUBMITTED	STATUS	GRADE	ACTIONS
Ahmed Khan STU1001	May 16, 2025	Submitted	-	<button>Grade</button>
Lisa Wong STU1042	May 15, 2025	Graded	86/100	<button>Review</button>
Michael Chen STU1078	May 17, 2025	Graded	92/100	<button>Review</button>
Sarah Johnson STU1105	May 14, 2025	Graded	78/100	<button>Review</button>
Raj Patel STU1142	Not submitted	Not Submitted	-	No submission

Showing 5 of 83 students

[Edit Assignment](#)

Figure 3.4.2.2.7 Assignment Pages

- Resource Scheduling Pages

Figures 3.4.2.2.8 through 3.4.2.2.10 display the Resource Scheduling Pages, featuring tools to book lecture halls, labs, and spaces with equipment like projectors, manage teaching equipment requests, set up student consultation office hours, view resource scheduling calendars, and track booking status.

Resources

[+ Book Resource](#) [Office Hours](#)

Calendar

May 17-23, 2025

Sun	Mon	Tue	Wed	Thu	Fri	Sat
17	18	19	20	21	22	23
9:00						
10:00	CS101 Lecture 10:00 - 12:00 Hall A-101					
11:00						
12:00						
1:00						
2:00						
3:00						

Figure 3.4.2.2.8 Resource Scheduling Pages

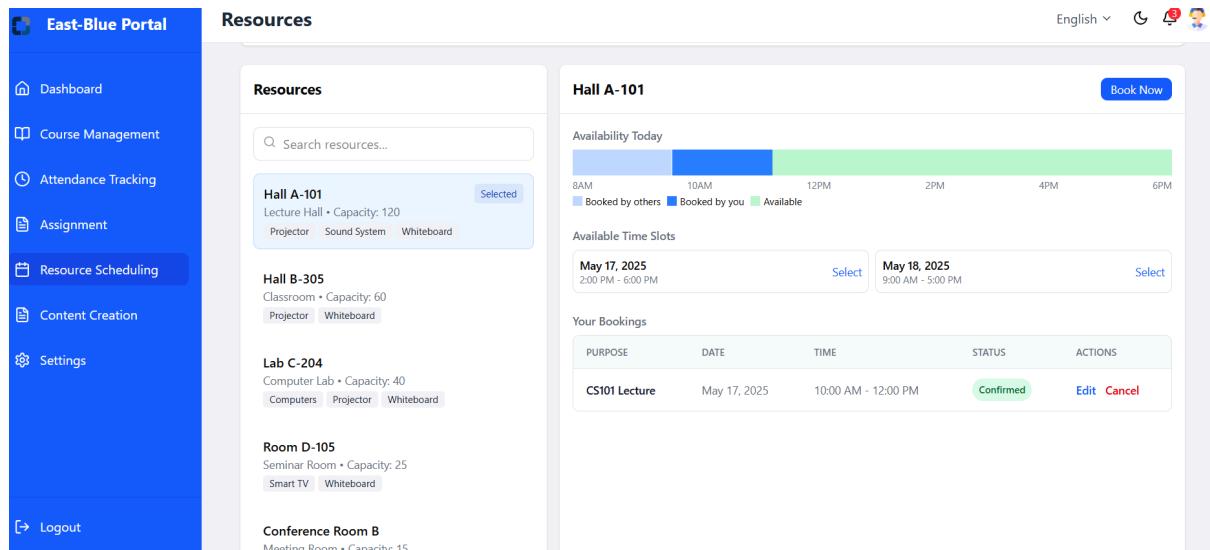


Figure 3.4.2.2.9 Resource Scheduling Pages

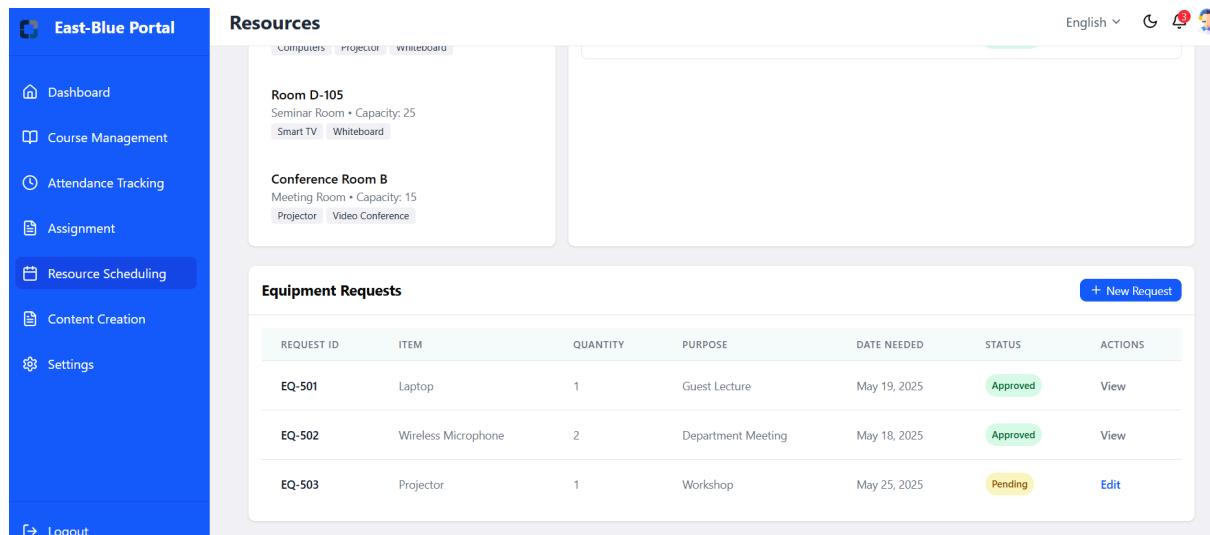


Figure 3.4.2.2.10 Resource Scheduling Pages

- Content creation

Figures 3.4.2.2.10 through 3.4.2.2.12 display the Content Creation Pages, featuring tools to develop and organize course materials, upload various learning resources, structure content into modules, control material visibility, and track student engagement through analytics.

The screenshot shows the 'Content' section of the East-Blue Portal. At the top, there are buttons for '+ Create Content' and 'Bulk Upload'. Below this is a 'Recent Content' table with columns: TITLE, TYPE, COURSE, CREATED, VISIBILITY, ANALYTICS, and ACTIONS. The table lists four items: 'Database Normalization Guide' (Document, CS310, May 15, 2025, Published, 83 views, 67 downloads), 'Sorting Algorithms Walkthrough' (Video, CS205, May 12, 2025, Published, 94 views, 58 downloads), 'Mid-term Project Guidelines' (Document, CS101, May 10, 2025, Published, 112 views, 89 downloads), and 'Neural Network Design Concepts' (Video, CS420, May 16, 2025, Draft, -). Below the table are sections for 'Course Content Structure' and 'Content Analytics'.

Figure 3.4.2.11 Content Creation Pages

The screenshot shows the 'Content' section of the East-Blue Portal. On the left, there is a sidebar with links: Dashboard, Course Management, Attendance Tracking, Assignment, Resource Scheduling, Content Creation (highlighted), Settings, and Logout. The main area has tabs for 'Course Content Structure' and 'Content Analytics'. The 'Course Content Structure' tab shows a tree view of modules: 'Introduction to Computer Science' (4 modules, 12 topics, 28 resources), 'Introduction to Programming' (Variables and Data Types, Control Structures), and Functions and Methods (Understanding Functions). There is a '+ Add New Module' button at the bottom. The 'Content Analytics' tab displays a 'Performance Overview' with a total of 1,590 views, a 12% increase from last semester, and a bar chart titled 'Monthly Views & Downloads' for Jan, Feb, Mar, Apr, and May. It also shows the 'Most Popular Content' with three items: 'Mid-term Project Guidelines' (112 views, CS101), 'Sorting Algorithms Walkthrough' (94 views, CS205), and 'Database Normalization Guide' (83 views, CS310).

Figure 3.4.2.12 Content Creation Pages

- Setting pages

Figure 3.4.2.13 displays the Setting Pages, featuring tools to configure user preferences and system options.

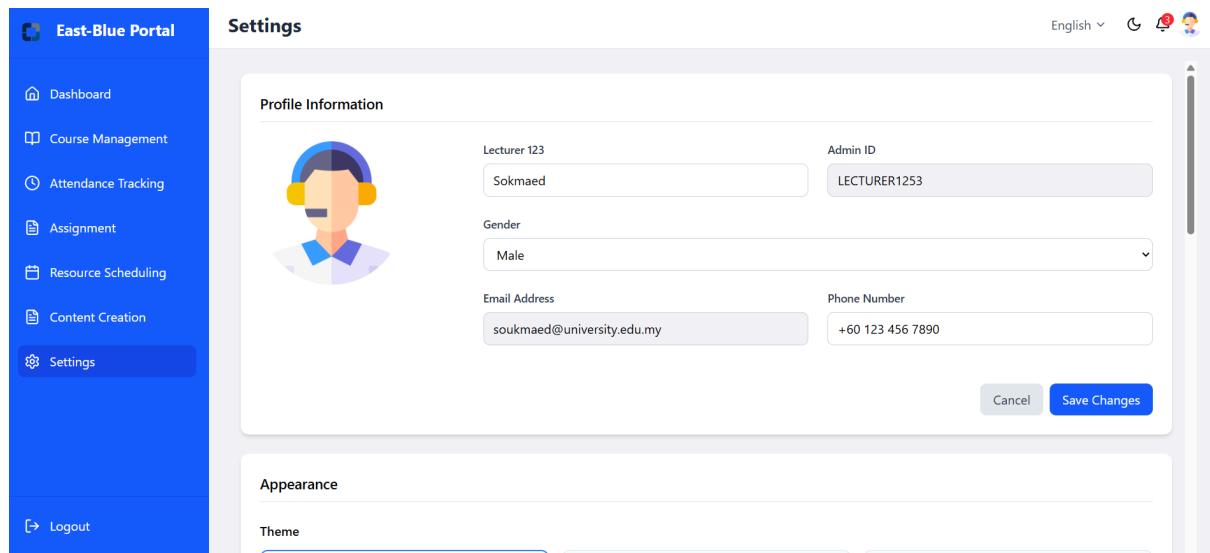


Figure 3.4.2.2.13 Setting Pages

- Figure 3.4.2.2.14 displays the Login Pages, it is designed to save login credentials in cookies, so that lecturers do not need to enter their username and password each time they log in.

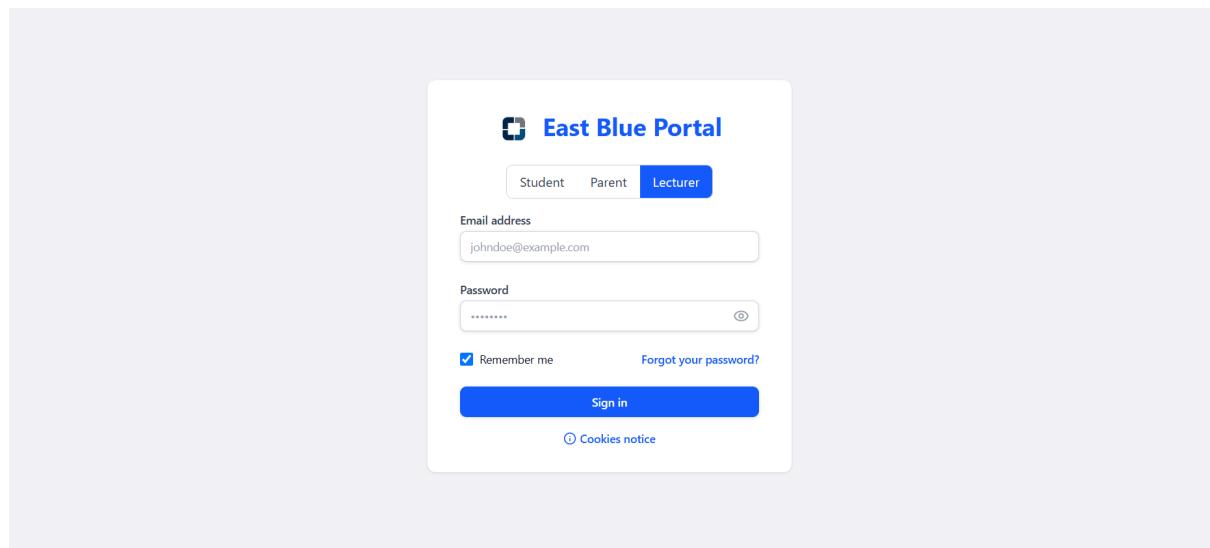


Figure 3.4.2.2.14 Login pages

3.4.2.3. Administrators

- **Login pages**
Figure 3.4.2.3.1 displays the Login Page, which requires the admin to enter their username and password each time they log in.

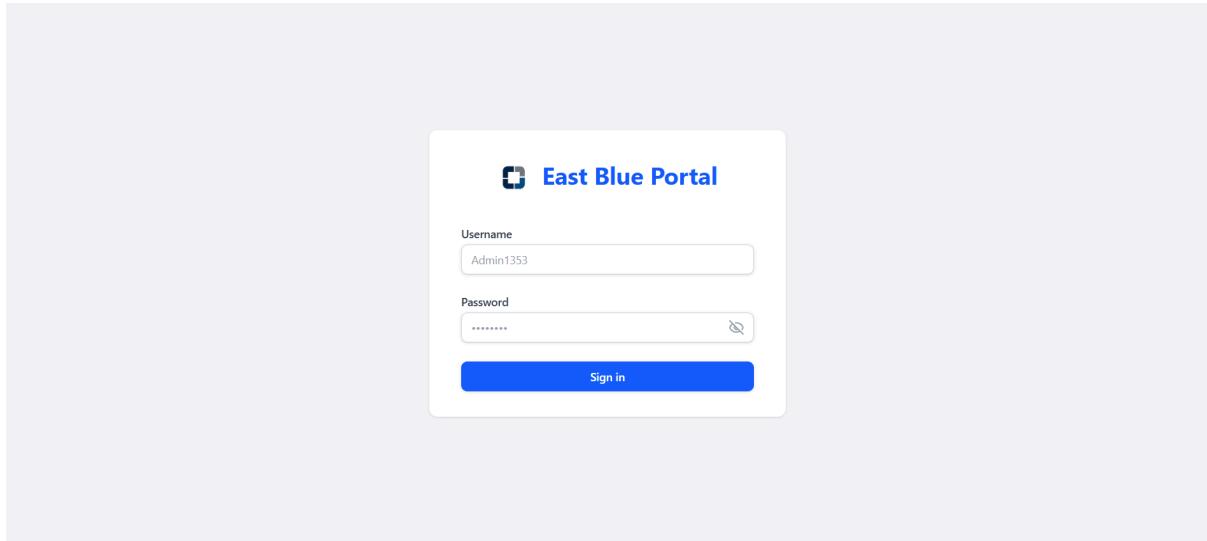


Figure 3.4.2.3.1 Login pages

- Dashboard Pages

Figures 3.4.2.3.2 and 3.4.2.3.3 display the Dashboard Pages, featuring system analytics with total user counts, active courses and system status, fee collection metrics, user activity charts, system monitoring graphs, and recent system logs.

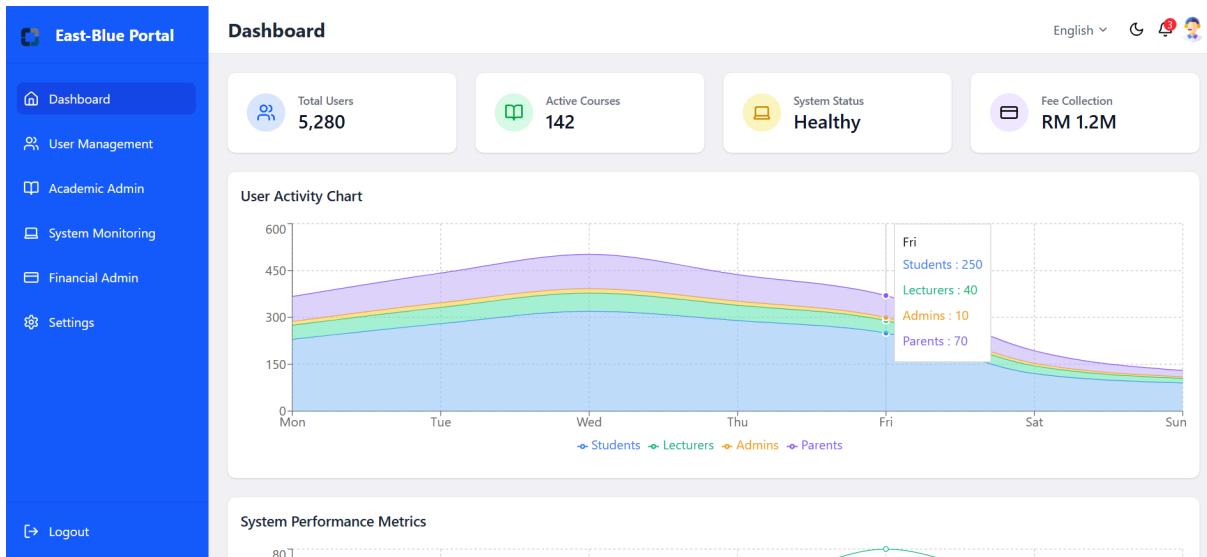


Figure 3.4.2.3.2 Dashboard Pages

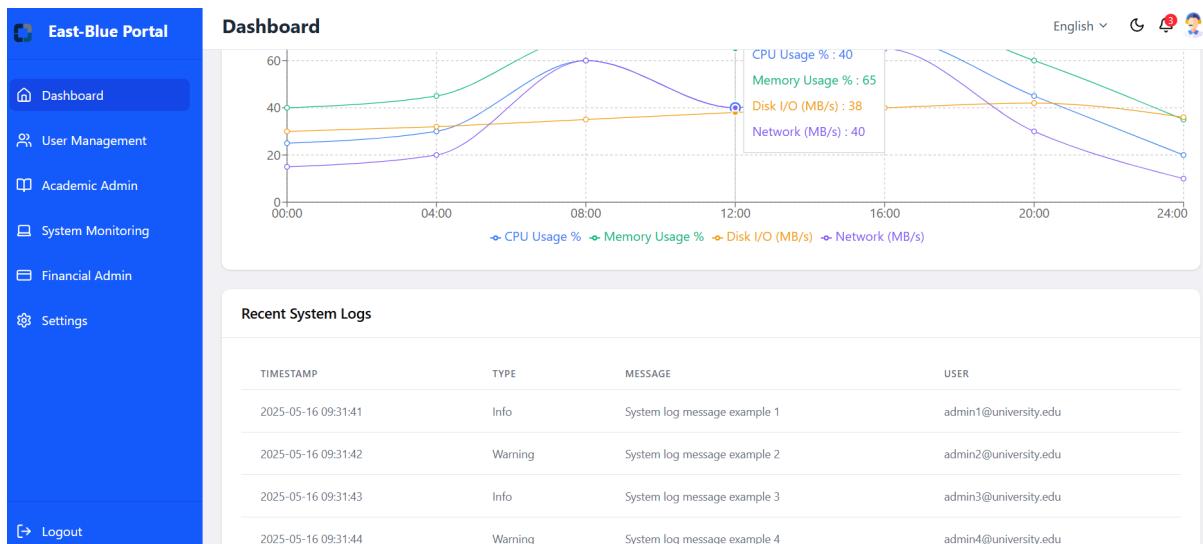


Figure 3.4.2.3.3 Dashboard Pages

- **User Management Pages**

Figure 3.4.2.3.4 displays the User Management Pages, featuring comprehensive tools to create, edit, and deactivate user accounts across all roles, assign and modify permissions, perform bulk user operations, and manage account security including password resets.

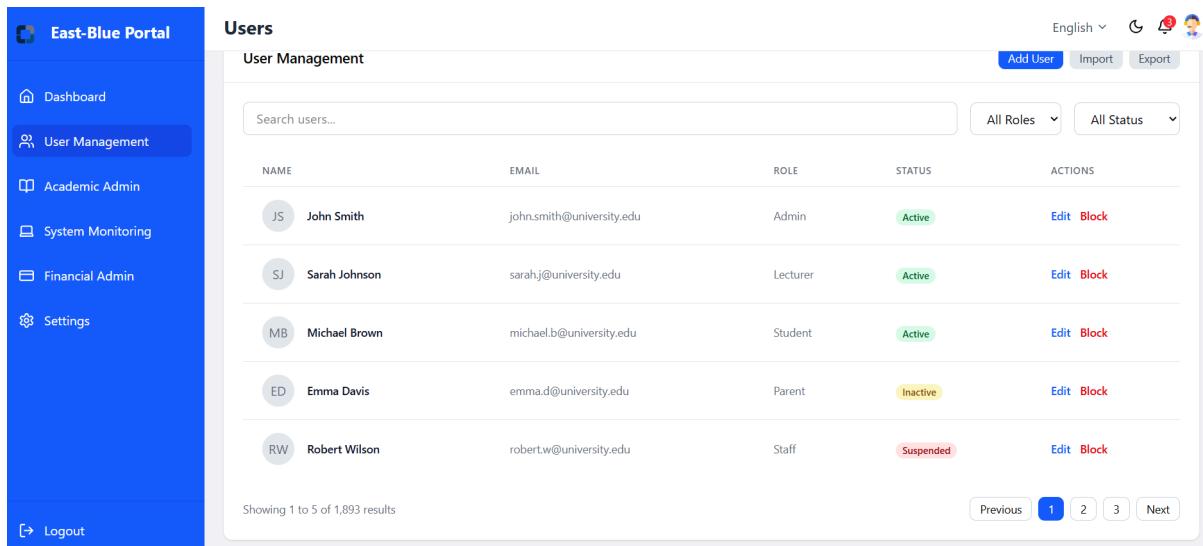


Figure 3.4.2.3.4 User Management Pages

- **Academic Admin Pages**

Figure 3.4.2.3.5 and Figure 3.4.2.3.6 displays the Academic Admin Pages, featuring tools to configure academic periods, establish organizational structures, manage course creation and scheduling, override enrollment parameters, and set up grading scales with academic policies.

The screenshot shows the 'Academic' section of the East-Blue Portal. On the left sidebar, 'Academic Admin' is selected. The main area displays three cards: 'Academic Periods' (Semester 1, 2025, Active; Semester 2, 2025, Upcoming; Summer Term, 2025, Upcoming), 'Faculties & Departments' (Faculty of Science, Faculty of Arts, Faculty of Engineering, Faculty of Business), and 'Course Statistics' (Total Active Courses: 142, Courses at Capacity: 38, Low Enrollment Courses: 15, New Courses This Semester: 24). Below these is a 'Course Management' table with columns: COURSE CODE, COURSE NAME, FACULTY, ENROLLMENT, STATUS, and ACTIONS. A search bar and filters for All Faculties and All Periods are also present.

Figure 3.4.2.3.5 Academic Admin Pages

This screenshot shows the 'Course Management' table from Figure 3.4.2.3.5. The table lists six courses with the following details:

COURSE CODE	COURSE NAME	FACULTY	ENROLLMENT	STATUS	ACTIONS
CS101	Introduction to Programming	Science	45/50	Active	Edit View
MATH201	Calculus II	Science	38/40	Active	Edit View
ENG150	Academic Writing	Arts	32/35	Active	Edit View
BUS305	Marketing Principles	Business	28/30	Pending	Edit View
PHYS220	Electromagnetism	Science	22/30	Active	Edit View

Figure 3.4.2.3.6 Academic Admin Pages

- System Monitoring Pages

Figures 3.4.2.3.7 and 3.4.2.3.8 display the System Monitoring Pages, featuring comprehensive system health statistics, server and database performance metrics, user activity tracking, usage analytics with reporting capabilities, and error log review for troubleshooting.

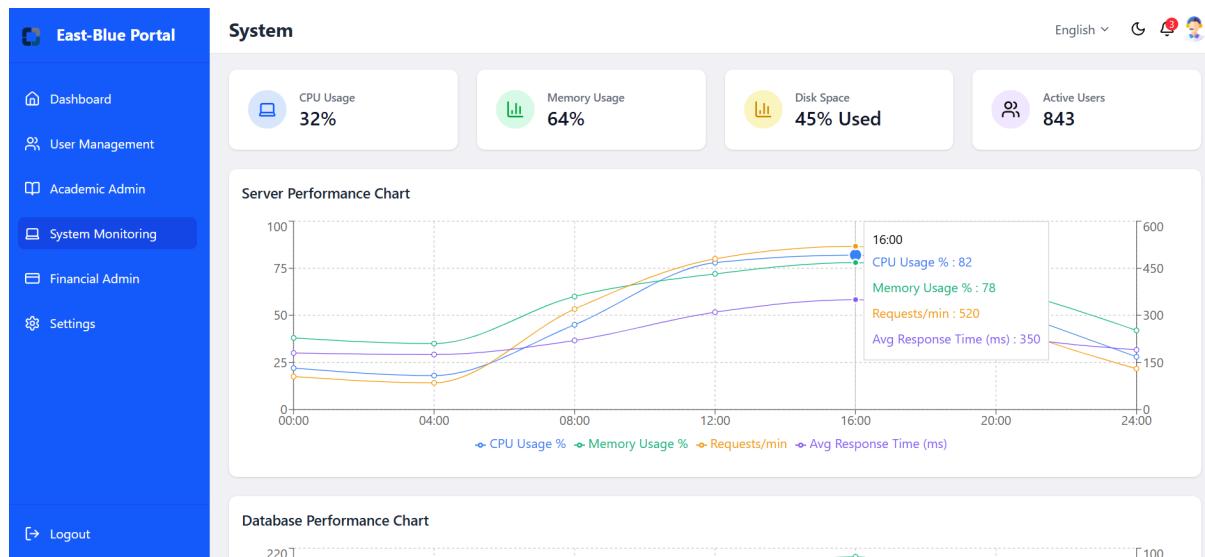


Figure 3.4.2.3.7 System Monitoring Pages

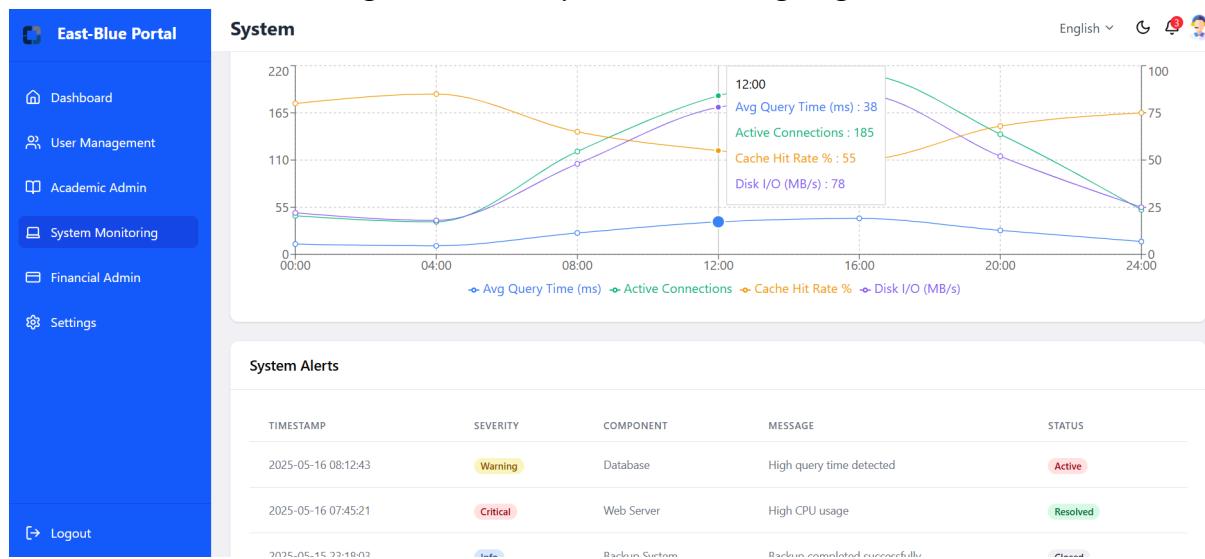


Figure 3.4.2.3.8 System Monitoring Pages

- Financial Admin Pages

Figures 3.4.2.3.9 and 3.4.2.3.10 display the Financial Admin Pages, featuring tools to configure payment systems and gateways, establish fee structures with payment schedules, generate financial reports with audit logs, and process refunds and financial adjustments.

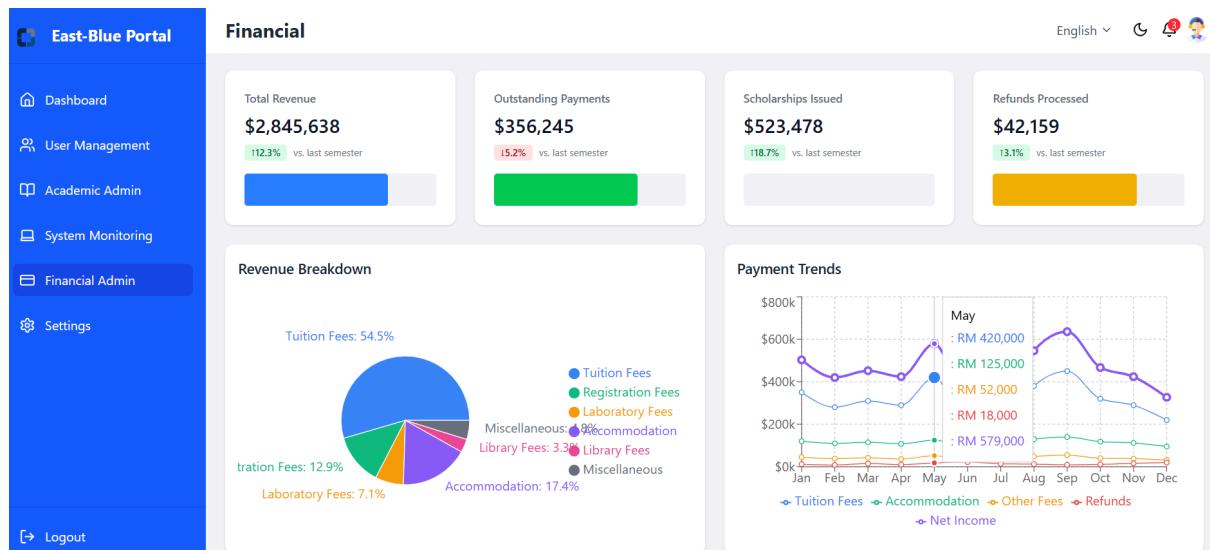


Figure 3.4.2.3.9 Financial Admin Pages

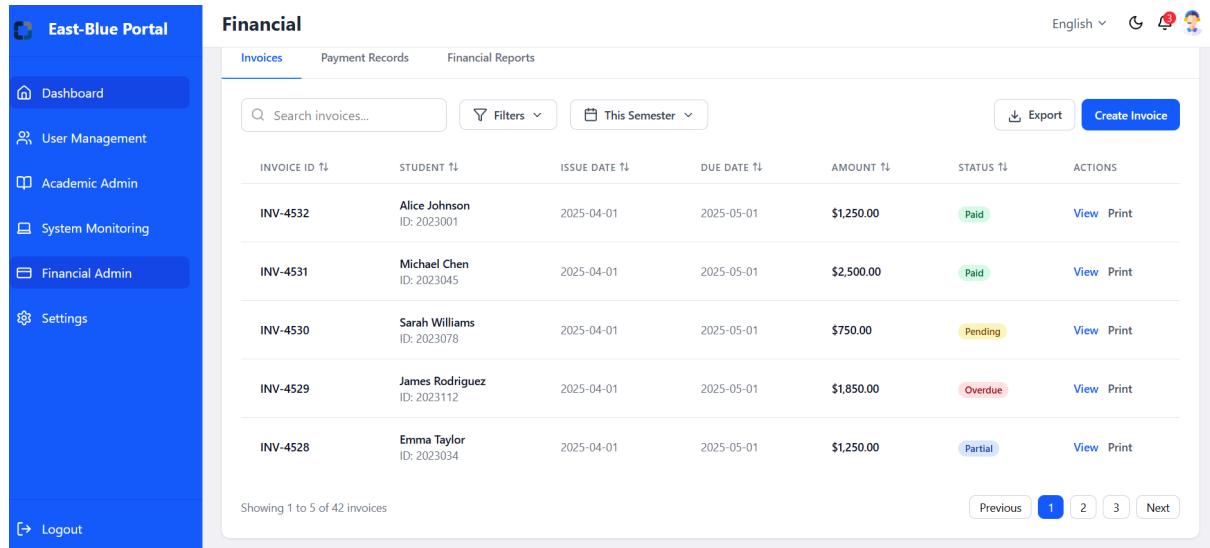


Figure 3.4.2.3.10 Financial Admin Pages

- Setting Pages

Figure 3.4.2.3.11 displays the Setting Pages, featuring tools to configure user preferences and system options.

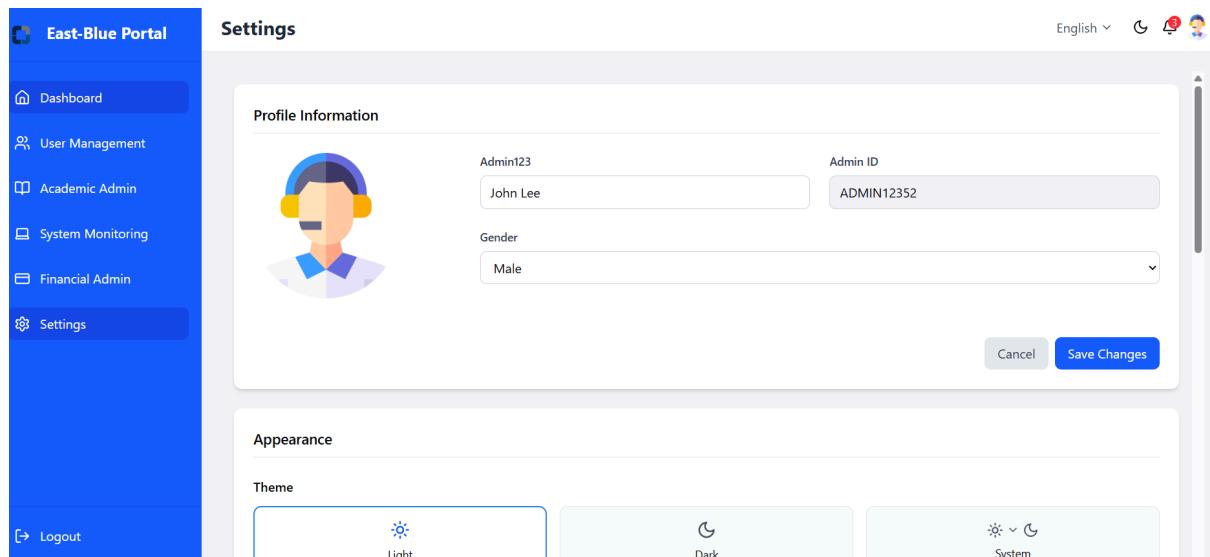


Figure 3.4.2.3.11 Setting Pages

3.4.2.4. Parent

- Login Pages

Figure 3.4.2.4.1 displays the Login Pages, it is designed to save login credentials in cookies, so that parents do not need to enter their username and password each time they log in.

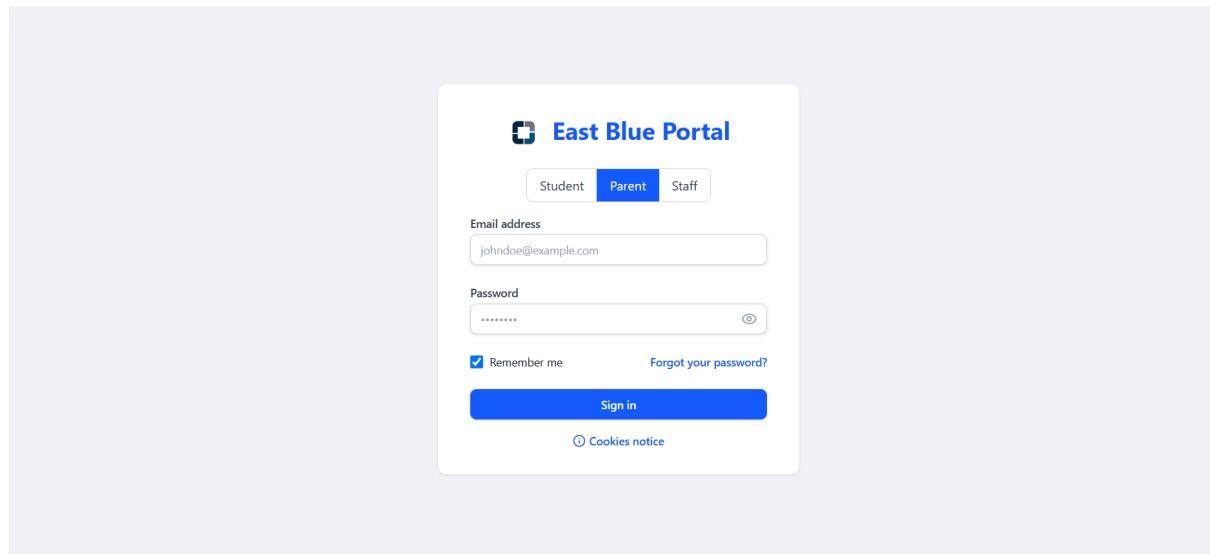


Figure 3.4.2.4.1 Login Pages

- Dashboard Pages

Figure 3.4.2.4.2 displays the Dashboard Pages, featuring a comprehensive parent overview of their child's academic standing with key metrics, financial obligations, recent grades, attendance records, academic alerts, and upcoming university events with notifications.

Dashboard

Luffy
Bachelor of Computer Science • Year 2
Student ID: STU1042 • Spring 2025

Current GPA **3.72** Attendance **92%**

Current Courses **5** Academic Standing **Good** Outstanding Balance **RM 2,800** Alerts **1**

Recent Grades

Course	Assessment	Grade	Date
CS205 Data Structures and Algorithms	Mid-term Exam	A- (87%)	May 15, 2025
CS310 Database Systems	Assignment	B+ (78%)	May 12, 2025
MATH201 Calculus II	Quiz	A (94%)	May 10, 2025

Upcoming Payments

Description	Due Date	Amount	Action
Tuition Fee - Spring 2025 (2nd Installment)	May 25, 2025	RM 2500.00	Pay Now
Laboratory Fee	May 30, 2025	RM 300.00	Pay Now

Figure 3.4.2.4.2 Dashboard Pages

- Academic Monitoring Pages

Figures 3.4.2.4.3 and 3.4.2.4.4 display the Academic Monitoring Pages, featuring comprehensive views of student academic performance, current courses with status indicators, detailed assessment results, attendance tracking, academic history, and GPA trend analysis with academic standing information.

Academic

Luffy
Bachelor of Computer Science • Year 2
Student ID: STU1042 • Spring 2025

Current GPA **3.72** Cumulative GPA **3.68** Credits Earned **32**

Current Courses

COURSE	INSTRUCTOR	CREDITS	CURRENT GRADE	ATTENDANCE	STATUS	ACTIONS
Data Structures and Algorithms CS205	Dr. Rebecca Chen	4	A-	92%	Good Standing	View Details
Database Systems CS310	Prof. Michael Rodriguez	3	B+	88%	Good Standing	View Details
Calculus II MATH201	Dr. Sarah Williams	3	A	95%	Good Standing	View Details
Technical Writing ENG102	Prof. James Thompson	2	C+	76%	At Risk	View Details
Physics for Computing PHY150	Dr. Alan Morris	3	B	89%	Good Standing	View Details

Figure 3.4.2.4.3 Academic Monitoring Pages

Recent Assessments

Course	Assessment	Due Date	Score	Grade
CS205	Mid-term Exam	May 15, 2025	87/100 (87%)	A-
CS310	Database Design Assignment	May 12, 2025	39/50 (78%)	B+
MATH201	Quiz 3 - Integration	May 10, 2025	19/20 (95%)	A
ENG102	Technical Report Draft	May 8, 2025	21/30 (70%)	C+
PHY150	Lab Exercise 5	May 5, 2025	21/25 (84%)	B

Upcoming Assessments

Final Exam	Due: June 12, 2025
CS205 • Exam Weight: 40%	
Final Project	Due: June 5, 2025
CS310 • Assignment Weight: 30%	
Final Exam	Due: June 10, 2025
MATH201 • Exam Weight: 40%	

Recent Attendance

May 17, 2025	Full History
CS205	Present
MATH201	Present

Figure 3.4.2.4.4 Academic Monitoring Pages

- **Financial Management Pages**

Figures 3.4.2.4.5 and 3.4.2.4.6 display the Financial Management Pages, featuring comprehensive financial overview with fee breakdowns, payment schedules with status indicators, transaction history with receipts, payment method management, full fee history, and options for downloading statements and making payments.

Spring 2025 Financial Summary
Bachelor of Computer Science • Year 2

Total Due	Paid	Balance
RM 4150.00	RM 1650.00	RM 2500.00

Fee Breakdown - Spring 2025

Tuition Fee	RM 5000.00
Laboratory Fee	RM 300.00
Library Fee	RM 150.00
Activity Fee	RM 200.00
Scholarship (Academic Merit)	RM -1500.00
Subtotal	RM 4150.00
Amount Paid	RM 1650.00

Outstanding Balance **RM 2500.00**

Payment Schedule

Tuition Fee - Spring 2025 (1st Installment) Due: January 15, 2025 ID: PAY-1001	RM 1500.00
(Paid on January 12, 2025)	
Library & Activity Fee Due: January 20, 2025 ID: PAY-1002	RM 350.00
(Paid on January 18, 2025)	
Tuition Fee - Spring 2025 (2nd Installment) Due: May 25, 2025 ID: PAY-1003	RM 2500.00
(Pay Now)	
Laboratory Fee Due: May 30, 2025 ID: PAY-1004	RM 300.00
(Pay Now)	

Figure 3.4.2.4.5 Financial Management Pages

The screenshot shows the financial management section of the East-Blue Portal. On the left sidebar, 'Financial Management' is selected. The main area displays:

- Recent Transactions:**

DATE	DESCRIPTION	PAYMENT METHOD	RECEIPT	AMOUNT	ACTIONS
January 12, 2025	Tuition Fee Payment - 1st Installment TRX-2001	Credit Card (XXXX-XXXX-XXXX-4589)	RCP-10058	-\$1500.00	Receipt
January 14, 2025	Spring 2025 Scholarship Credit TRX-2002	Institutional Credit	SCH-20025	+\$1500.00	Receipt
January 18, 2025	Library & Activity Fee Payment TRX-2003	Bank Transfer	RCP-10062	-\$350.00	Receipt
- Payment Methods:**

Credit Card Default ***** 4589 • Expires 05/27 Ahmed Rahman	Edit Remove
Bank Account National City Bank **** 7532	Edit Set Default Remove
- Fee History:**

Semester	Total Fee	Scholarship	Net Fee	Status
Fall 2024	RM 4950.00	RM 1500.00	RM 3450.00	Paid Statement
Spring 2025	RM 5650.00	RM 1500.00	RM 4150.00	Partially Paid Statement

Figure 3.4.2.4.6 Financial Management Pages

- Document Access Pages

Figures 3.4.2.4.7 and 3.4.2.4.8 display the Document Access Pages, featuring a central repository for all student-related documents including academic reports, transcripts, financial statements, administrative notices, document notifications, and parent resources.

The screenshot shows the document access section of the East-Blue Portal. On the left sidebar, 'Document Access' is selected. The main area displays:

- Recent Documents:**

DOCUMENT TITLE	TYPE	DATE	SIZE	STATUS	ACTIONS
Spring 2025 Academic Report DOC-1001	Academic	May 15, 2025	2.1 MB	New	View Download
Tuition Fee Statement - Spring 2025 DOC-1002	Financial	January 10, 2025	1.5 MB	Viewed	View Download
Mid-Semester Progress Report DOC-1003	Academic	April 5, 2025	3.2 MB	Viewed	View Download
Attendance Warning Notice - ENG102 DOC-1004	Administrative	May 10, 2025	0.8 MB	New	View Download
- Academic** **Financial** **Administrative** **University**
- Fall 2024 Transcript**
Official transcript for Fall 2024 semester
December 20, 2024 • 1.2 MB
[View](#) [Download](#)
- Spring 2025 Course Registration**

Figure 3.4.2.4.7 Document Access Pages

The screenshot shows the 'Document Access' section of a portal. On the left, under 'Notifications', there are three items: 'New Document Available' (Spring 2025 Academic Report uploaded on May 15, 2025), 'Administrative Notice' (Attendance Warning Notice for ENG102 course on May 10, 2025), and 'Document Access Reminder' (Mid-Semester Progress Report available on April 5, 2025). Each notification has 'Mark as Read' and 'View Document' links. Below these is a 'View All Notifications' link. On the right, under 'Help & Resources', there are three sections: 'Parent's Guide to Academic Reports' (with a download guide), 'Understanding Financial Statements' (with a download guide), and 'University Policies & Procedures' (with a download handbook). A 'Need Help?' box contains a message about contacting Parent Support, with a 'Contact Support' button.

Figure 3.4.2.4.8 Document Access Pages

- **Setting Pages**

Figure 3.4.2.4.9 displays the Setting Pages, featuring tools to configure user preferences and system options.

The screenshot shows the 'Settings' page of the East-Blue Portal. The left sidebar includes links for Dashboard, Academic Monitoring, Financial Management, Document Access, Settings (which is selected), and Logout. The main content area is titled 'Settings'. It contains two main sections: 'Profile Information' and 'Appearance'. In 'Profile Information', there is a profile picture of a family, fields for 'Parent' (Shanks) and 'Parent ID' (PARENT52318), a 'Gender' dropdown set to 'Male', an 'Email Address' field (shanks@gmail.com), and a 'Phone Number' field (+60 123 456 7890). At the bottom are 'Cancel' and 'Save Changes' buttons. Below this is the 'Appearance' section, which includes a 'Theme' dropdown.

Figure 3.4.2.4.9 Setting Pages

3.4.3. Hardware Interfaces

3.4.3.1. Server Requirements

- Application Servers: AWS EC2 instances with minimum 8 CPU cores, 16GB RAM
- Database Servers: AWS RDS instances with 16GB RAM, 100GB storage
- File Storage: AWS S3 buckets for document storage with appropriate encryption

3.4.3.2. Client Device Requirements

- Desktop Access

- CPU: Dual-core 2.0GHz or better
- RAM: 4GB minimum (8GB recommended)
- Storage: 1GB free space for desktop application
- Resolution: Minimum 1280×720 display
- Mobile Access
 - Supported OS: iOS 14+ and Android 9+
 - RAM: 2GB minimum
 - Storage: 100MB free space for mobile application
 - Camera: Required for document scanning and identity verification

3.4.4. Software Interfaces

3.4.4.1. Development Stack

- Frontend:
 - Web: React 18+ with TypeScript, Next JS shadcn/ui component library
 - Desktop: Electron with React and Redux for state management
 - Mobile: React Native with Redux
- Backend:
 - Node.js with Express.js framework
 - RESTful API architecture
 - GraphQL for complex data queries
- Database:
 - AWS PostgreSQL RDS for relational data
 - Redis for caching and session management

3.4.4.2. Third-Party Integrations

- Payment Processing: Stripe
- SMS Service: Twilio API for text message notifications
- Cloud Storage: AWS S3 API for document storage
- Analytics: Google Analytics and custom event tracking

3.4.5. Communications Interfaces

3.4.5.1. Network Requirements

- HTTPS with TLS 1.3 for all data transmission
- WebSockets for real-time notifications and AI assistant chat

3.4.5.2. Data Exchange

- JSON as primary data exchange format
- JWT (JSON Web Tokens) for authentication

3.4.5.3. Security Protocols

- AES-256 encryption for sensitive data
- CORS implementation for web security

3.4.5.4. Messaging System

- Push notifications using Firebase Cloud Messaging
- In-app notification system using WebSockets

3.4.5.5. AI Assistant Communication

- WebSocket connection for real-time chat functionality
- Natural Language Processing via cloud-based AI services
- Voice recognition integration for accessibility

3.5. Logical Database Requirements

3.5.1. Entity Relationship Diagram (ERD)

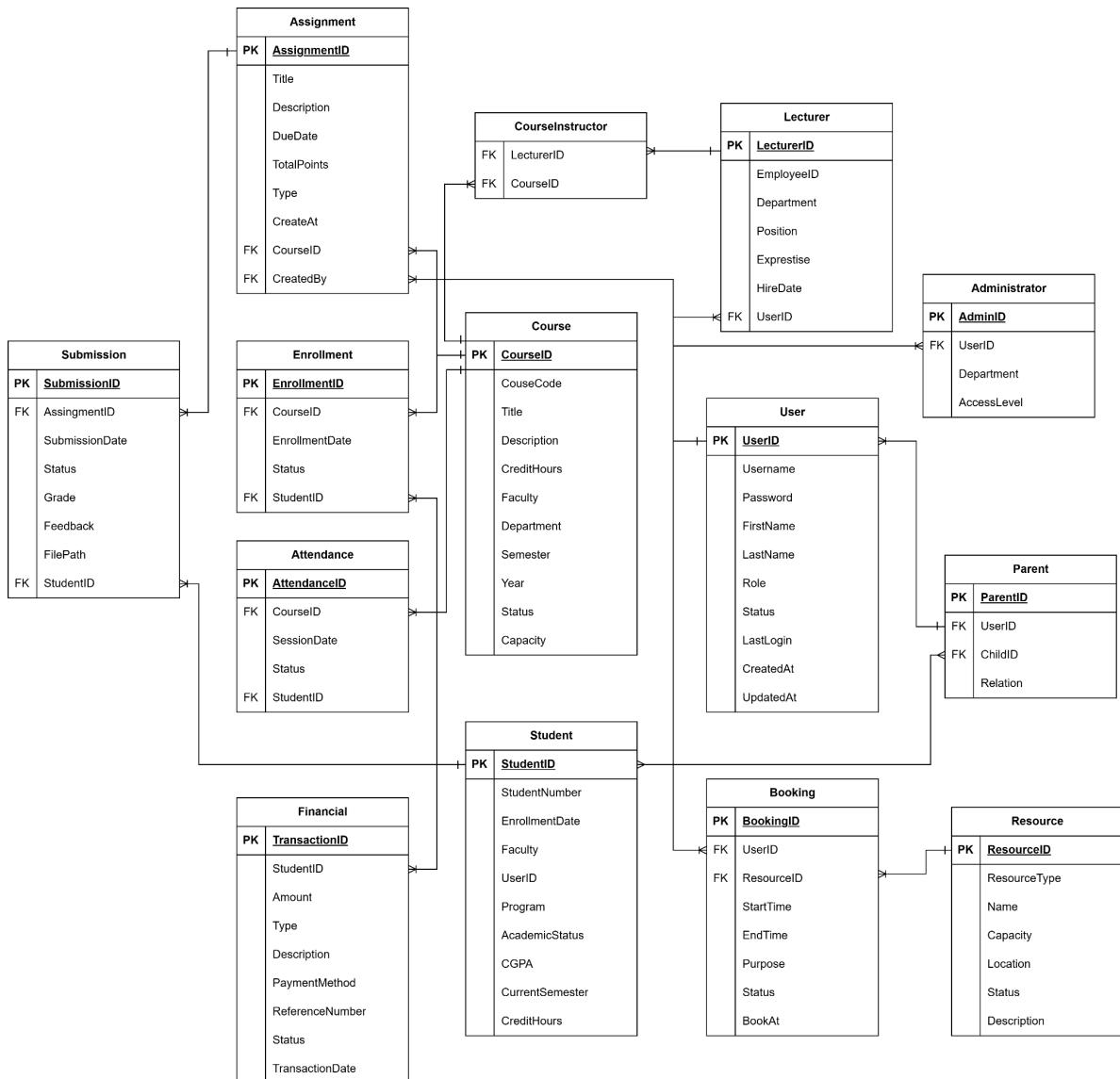


Figure 3.5.1.1 ERD

<https://drive.google.com/file/d/1JnCAi-yHJR8-E8RWUKVryEZBMOQKvRA2/view?usp=sharing>

3.5.2. Core Entities and Relationships

3.5.2.1. User

- Attributes
 - UserID (PK), Username, Password, FirstName, LastName, Role, Status, LastLogin, CreatedAt, UpdatedAt
- Relationships
 - One-to-many with Student, Lecturer, Parent, and Administrator
- Constraints
 - Unique username
 - password complexity requirements

3.5.2.2. Student

- Attributes
 - StudentID (PK), UserID (FK), StudentNumber, EnrollmentDate, Faculty, Program, AcademicStatus, CGPA, CurrentSemester, CreditHours
- Relationships
 - One-to-many with Attendance, Assignment, Payment
- Constraints
 - StudentNumber must be unique

3.5.2.3. Lecturer

- Attributes
 - StudentID (PK), UserID (FK), StudentNumber, EnrollmentDate, Faculty, Program, AcademicStatus, CGPA, CurrentSemester, CreditHours
- Relationships
 - One-to-many with CourseInstructor
- Constraints
 - EmployeeID must be unique

3.5.2.4. Parent

- Attributes
 - ParentID(PK), UserID(FK), CholdID(FK), Relation
- Relationships
 - Many-to-many with Student
- Constraints
 - Each parent record must be linked to a valid User account

3.5.2.5. Administrator

- Attributes
 - AdminID(PK), UserID(FK), Department(FK), AccessLevel
- Constraints
 - Each administrator record must be linked to a valid User account

3.5.2.6. CouseInstructor

- Attributes
 - LecturerID(FK), CouseID(FK)
- Relationships
 - Many-to-one with Lecturer and Course
- Constraints
 - The combination of LecturerID and CourseID must be unique.

3.5.2.7. Enrollment

- Attributes
 - EnrollmentID (PK), StudentID (FK), CourseID (FK), EnrollmentDate, Status, Grade
- Relationships
 - Many-to-one with Student and Course
- Constraints
 - A student can only enroll once in a course

3.5.2.8. Course

- Attributes
 - CourseID (PK), CourseCode, Title, Description, Credits, Faculty, Department, Semester, Year, Status, Capacity
- Relationships
 - One-to-many with Student
 - One-to-many with Lecturer
- Constraints
 - CourseCode must be unique within a semester

3.5.2.9. Assignment

- Attributes
 - AssignmentID (PK), CourseID (FK), Title, Description, DueDate, TotalPoints, Type, CreatedBy, CreatedAt
- Relationships
 - Many-to-one with Course
 - One-to-many with Submission
- Constraints
 - Assignment due dates must be within semester dates

3.5.2.10. Submission

- Attributes
 - SubmissionID (PK), AssignmentID (FK), StudentID (FK), SubmissionDate, Status, Grade, Feedback, FilePath
- Relationships
 - Many-to-one with Assignment and Student
- Constraints
 - A student can only submit once per assignment

3.5.2.11. Attendance

- Attributes
 - AttendanceID (PK), CourseID (FK), StudentID (FK), SessionDate, Status
- Relationships
 - Many-to-one with Course and Student
- Constraints
 - Only one attendance record per student per course session

3.5.2.12. Financia

- Attributes
 - TransactionID (PK), StudentID (FK), Amount, Type, Description, PaymentMethod, ReferenceNumber, Status, TransactionDate
- Relationships
 - Many-to-one with Student
- Constraints
 - Transaction amounts must be non-negative

3.5.2.13. Resource

- Attributes
 - ResourceID (PK), ResourceType, Name, Capacity, Location, Status, Description
- Relationships
 - One-to-many with Booking
- Constraints
 - Resource must have a valid type

3.5.2.14. Booking

- Attributes
 - BookingID (PK), ResourceID (FK), UserID (FK), StartTime, EndTime, Purpose, Status, BookedAt
- Relationships
 - Many-to-one with Resource and User
- Constraints

- No overlapping bookings for the same resource

3.6. Design Constraints

User-Friendly Interface

- This system should have an intuitive and user-friendly interface accessible to students, faculty, administrators and parents. This includes easy navigation, clear instructions and responsive design for mobile compatibility.

Compliance with University IT Policies

- The system must fully comply with all existing University IT governance frameworks, data security standards (including encryption protocols for sensitive data), and privacy policies (e.g., PDPA compliance if applicable in the Malaysian context). This includes any university-mandated authentication protocols.

Integration with Existing University Systems

- The East Blue Portal must integrate with the current versions of the University's Student Information System (SIS) and Learning Management System (LMS) using their officially provided APIs and data exchange formats. The design cannot necessitate major changes to these existing core systems.

Branding and Style Guidelines

- The user interface of the East Blue Portal must adhere to the official University branding guidelines, including the use of approved logos, color palettes, and typography to ensure a consistent user experience with other university platforms.

3.7. Software System Attributes

The quality and performance of the East Blue Portal depend on various non-functional attributes. These attributes define how well the system performs under different conditions, how secure and maintainable it is, and how users experience it. The following software system attributes are essential to ensure the portal meets the expectations of its stakeholders.

a) Reliability

The East Blue Portal must operate consistently and correctly over time, without unexpected failures or data loss. Reliability is crucial since the portal will be used by students, lecturers, administrators, and parents for accessing academic and administrative services.

- The system should be able to recover from a crash within 1 minute.
- In case of failure, the system should support automatic recovery and failover mechanisms.
- Reliability testing should be conducted to ensure all critical features (e.g., login, notifications, data retrieval) function correctly under normal and stress conditions.

b) Availability

High availability ensures that the system is accessible to users whenever needed. The East Blue Portal must be accessible 99.9% of the time, especially during peak periods like course registration, result releases, and payment deadlines and working hours (Monday through Friday, 8 AM to 6 PM).

- Scheduled maintenance must be announced at least 24 hours in advance.
- Real-time monitoring tools should be implemented to track server health.
- In the event of downtime, a user-friendly message and estimated restoration time should be shown.

c) Security

Security is paramount, especially as the system deals with sensitive academic and financial data.

- The system must use HTTPS encryption for all data transmissions.
- Role-based access control (RBAC) must be enforced to ensure users can only access data and features permitted for their role (Student, Lecturer, Parent, Admin).
- Login should be secured using strong password policies and optional two-factor authentication (2FA).
- Data must be encrypted in storage and comply with university data privacy policies and applicable regulations.

d) Maintainability

Maintainability refers to the ease with which the system can be updated, debugged, or modified. A modular and well-documented system is essential for long-term support.

- The codebase should follow clean coding standards and be accompanied by developer documentation.
- The system should be designed using a modular architecture so that updates to one component do not affect others.
- Source code should be version-controlled and update logs should be maintained.
- Logs for system errors, user actions, and security events should be stored for audit and debugging.

e) Portability

The system should be portable across different environments to support development, testing, and deployment.

- The system must run on Linux, macOS and Windows-based web servers.
- It should support containerization such as Docker for easier deployment across multiple environments.
- Configuration files should be environment-based (e.g., dev, test, prod) for flexible deployment.

3.8. Supporting Information

3.8.1. Summary of Key Elicitation Findings

3.8.1.1. Type of Elicitation Used

For the East Blue Portal project, several elicitation techniques were employed to gather comprehensive user requirements. The primary technique used was a questionnaire-based survey designed using the Kano model. This method allowed the team to capture user preferences and expectations regarding various portal features across stakeholder groups including students, lecturers, administrators, and parents.

In addition to the survey, a series of brainstorming sessions were conducted with selected participants. These sessions followed a structured approach aligned with the Kano model, categorizing requirements into basic (Must-Be), performance (One-Dimensional), and attractive (Delighter) features. The brainstorming discussions took place virtually via Discord and were moderated by a facilitator. Each session targeted a specific requirement type, promoting focused and relevant idea generation.

Following the brainstorming activities, the team developed a prioritization matrix to further refine the captured requirements. This matrix helped map each feature's perceived value against the estimated implementation effort, allowing the team to prioritize development based on impact and feasibility.

3.8.1.2. Explanation of Elicitation Use

The questionnaire served as a scalable method for gathering input from a large number of users. By applying the Kano model, the team could classify responses in terms of user delight, performance expectation, or basic necessity. Questions were intentionally designed to be simple and accessible to all target groups, including multiple-choice and Likert-scale formats. The data gathered provided quantifiable insights into what users truly valued, expected, or could live without.

The brainstorming sessions complemented the questionnaire by allowing participants to speak openly about their needs, frustrations, and ideas for improving the current system. These sessions created a collaborative environment where new, sometimes unexpected ideas emerged. Participants helped surface features that might not have been captured through closed-form questions. The discussions were transcribed and analyzed, leading to the identification of recurring themes and requirements.

The resulting prioritization matrix was developed using inputs from both the questionnaire and brainstorming sessions. It served as a strategic planning tool to help the team decide which requirements should be addressed first. Features

that were highly valued by users and easy to implement were marked as early-phase priorities, while more complex or low-priority features were deferred to later stages.

3.8.1.3. Benefits

Using a combination of structured surveys and open-ended brainstorming allowed the team to approach the elicitation process from both a quantitative and qualitative perspective. The questionnaire enabled the team to reach a wide pool of users efficiently, while the brainstorming sessions provided deeper insight into context, real-world scenarios, and user expectations.

The process ensured that feature prioritization was not arbitrary but instead rooted in actual user feedback. It also helped surface “delighter” features, which are enhancements that users hadn’t expected but appreciated, such as chatbots and personalized dashboards. The prioritization matrix ensured a practical balance between user desire and development effort, helping to guide implementation planning.

3.8.1.4. Challenges Faced

While the elicitation process yielded useful results, the team encountered several challenges. One issue was lower participation among some stakeholder groups, particularly parents and administrative staff. Despite targeted outreach, student responses dominated the dataset, potentially introducing bias toward student-centric features.

There were also some interpretation challenges with the Kano-style questions. Some respondents may not have fully understood the difference between “I like it that way” and “I expect it that way”, potentially impacting the clarity of feature classification. In future iterations, additional instruction or examples could be provided alongside the survey.

Another notable challenge was the time constraint. Conducting surveys, facilitating brainstorming sessions, and analyzing all collected data had to be done within a relatively short time frame, which limited the opportunity to conduct follow-up sessions or validations.

3.8.1.5. Recommendations

Based on the experience gained, it is recommended that future iterations of the project include continuous and iterative elicitation throughout the system’s lifecycle. As user needs evolve, maintaining a feedback loop will help the system remain relevant and user-friendly.

To address stakeholder diversity, future efforts should include more targeted recruitment strategies to improve participation among parents, lecturers, and administrators. This could involve working directly with faculty or department heads to distribute surveys or invite participants.

Combining elicitation techniques with usability testing and direct observation could also enhance understanding of user needs, especially regarding interface behavior and accessibility. Furthermore, the use of automated analytics tools may help streamline the analysis of large-scale questionnaire results and reduce manual workload.

Finally, it is recommended to maintain and update the prioritization matrix as new features are proposed or implemented. This ensures that system evolution continues to reflect real user value and practical constraints.

3.8.2. Sample Input/Output Formats

3.8.2.1. Student Semester Report Export (Output):

- Format: PDF (Portable Document Format)
- Content Example Snippet:
- Header: University Logo, Student Name, Student ID, Program, Semester/Year
- Body: Table with columns (Course Code, Course Name, Credits, Grade, Grade Points)
- Footer: Semester GPA, CGPA, Academic Standing, Official Remarks, Date of Issue.
- Reference: See Use Case verification scenario 1.

3.8.2.2. SMS Notification for Low Attendance (Output):

- Format: Plain Text SMS
- Content Example: "East Blue Portal Alert: Dear [Student Name], your attendance for [Course Code] [Course Name] has fallen to [XX]%. Please ensure regular attendance. Contact [Lecturer Name/Admin Office] if you need assistance."
- Reference: See Use Case verification scenario 5.

3.8.2.3. Course Registration Data (Input - conceptual for batch processing or API):

- Format: CSV (Comma Separated Values) or JSON (JavaScript Object Notation)
- CSV Example: StudentID,CourseCode,Action
"S1001","CS101","REGISTER"
"S1002","MA202","REGISTER"
- JSON Example:

```
[  
  { "studentID": "S1001", "courseCode": "CS101", "action": "REGISTER" },  
  { "studentID": "S1002", "courseCode": "MA202", "action": "REGISTER" }  
]
```

Figure 3.8.2.3.1 Sample JSON File

4. Verification

4.1. Verification Approach

Methods of Verification (How):

- **Functional Testing:** To verify that all features and functions specified in functions (section 3.1) and detailed in the use case specifications (section 3.1.1) operate correctly. This will involve executing test cases derived from these requirements.
 - *How:* Manual and automated test case execution. Test scenarios like those outlined in the initial draft of SRS (section 4.1) (e.g., One-Click Semester Report Export, Online Booking) will be formalized into test cases.
- **Performance Testing:** To assess the system's responsiveness, stability, and scalability under various load conditions, ensuring it meets the performance requirements outlined in performance requirements (section 3.2).
 - *How:* Automated load testing tools will simulate peak concurrent user loads and measure response times, throughput, and resource utilization.
- **Usability Testing:** To evaluate the ease of use, learnability, and user satisfaction of the portal, ensuring it meets the usability requirements specified in usability requirements (section 3.3).
 - *How:* Observation of representative users (students, lecturers, parents, administrators) performing predefined tasks using a think-aloud protocol. Feedback will be collected via questionnaires (e.g., SUS) and interviews.
- **Interface Testing:** To verify that all external and internal interfaces (System, User, Hardware, Software, Communications) defined in interface requirements (section 3.4) function correctly and integrate seamlessly.
- **Security Testing:** To identify and address potential vulnerabilities, ensuring the system meets the security attributes defined in software system attributes (section 3.7.c). This will include checks for data encryption, role-based access control, and protection against common web vulnerabilities.
- **Logical Database Testing:** To ensure data integrity, correct relationships, and adherence to constraints as defined in ERD (section 3.5).

Responsible Parties (who):

- **Development Team:** Responsible for unit testing and initial integration testing.
- **Quality Assurance (QA) Team:** Responsible for functional testing, performance testing, interface testing, security testing, and coordinating usability testing.
- **Selected End-Users (Students, Lecturers, Parents, Admin Staff):** Will participate in usability testing and User Acceptance Testing (UAT).

Timing of Verification (When):

- **Unit Testing:** Conducted by developers continuously during each development sprint.
- **Integration Testing:** Conducted after each sprint when new modules are integrated.
- **System Testing (Functional, Performance, Security, Interface):** Conducted by the QA team at the end of each major development phase or release cycle (e.g., Alpha, Beta releases).
- **Usability Testing:** Carried out at significant turning points, such as prior to UAT and following the creation of significant UI components.
- **User Acceptance Testing (UAT):** Conducted by end-users prior to final system deployment.

Locations/Environments for Verification (Where):

- **Development Environment:** Used for unit and initial integration testing.
- **Staging/QA Environment:** A dedicated environment mirroring the production setup, used for system testing (functional, performance, security, interface) and UAT. This environment will be used for tests simulating remote access and on-campus access.
- **University Computer Labs/Classrooms:** Utilized for controlled usability testing sessions with students and lecturers.
- **University Administrative Offices:** Used for specific UAT scenarios involving administrative staff and functions.
- **Parent Home environment:** Used for UAT scenarios involving parent portal access, student progress monitoring, and communication features from remote locations with varying internet connectivity and device types.

4.2. Verification Criteria

Functional Criteria

Table 4.2.1 Functional Criteria

VC_FUNC_001 (Login):	Users (Student, Lecturer, Parent) with valid credentials can successfully log in within 3 attempts. The system correctly saves credentials in cookies if selected. Admin users must manually enter credentials each time and successfully log in.
VC_FUNC_002 (Student Dashboard Customization):	A logged-in student can add, remove, and rearrange widgets on their dashboard, and the layout changes persist across sessions.
VC_FUNC_003 (View Academic Record):	A logged-in student/parent can view the student's complete academic record, including current and past grades, CGPA progression, course enrollment, and attendance, accurately reflecting data from the SIS.
VC_FUNC_004 (Online Booking):	A logged-in student can successfully search for available venue/consultation slots and complete a booking. The booking is recorded, and confirmation is provided.
VC_FUNC_005 (AI Chatbot):	The AI chatbot correctly interprets and responds to student queries regarding academic or administrative information contained within the system (e.g., assignment due dates, financial deadlines).
VC_FUNC_006 (Manage Finance - Student):	A logged-in student/parent can view current balances, payment history, and successfully

	initiate an online payment through the integrated payment gateway.
--	--

Performance Criteria

Table 4.2.2 Performance Criteria

VC_PERF_001 (Response Time):	All user interactions (e.g., logging in, accessing course materials, submitting assignments) shall be responded to within 5 seconds under normal load (up to 500 concurrent users).
VC_PERF_002 (Concurrent Users):	The system shall support a minimum of 1000 concurrent users during peak usage times without performance degradation below specified response times.
VC_PERF_003 (Downtime):	Scheduled system downtime shall not exceed 4 hours per month. Unscheduled downtime shall not exceed 1 hour per month.

Usability Criteria:*Table 4.2.3 Usability Criteria*

VC_USAB_001 (Task Completion - Effectiveness):	90% of student users shall be able to successfully complete the task of viewing their grades within 3 minutes without assistance.
VC_USAB_002 (Efficiency):	Key system pages (e.g., Dashboard, Academic Page, Timetable) shall load within 3 seconds under normal load conditions.
VC_USAB_003 (Satisfaction):	The system shall achieve a System Usability Scale (SUS) score of at least 75/100 in post-deployment usability surveys.
VC_USAB_004 (Error Prevention):	Critical actions (e.g., deleting records, submitting payments) shall require user confirmation before execution.

Interface Criteria

Table 4.2.4 Interface Criteria

VC_INT_001 (SIS Integration):	Data related to student enrollment, academic records, and course details shall be accurately and consistently synchronized between the East Blue Portal and the SIS.
VC_INT_002 (UI Responsiveness):	The user interface shall be responsive and display correctly on common desktop browsers (Chrome, Firefox, Edge latest versions) and mobile devices (iOS 14+, Android 9+).

Software System Attribute Criteria

Table 4.2.1 Software System Attribute Criteria

VC_ATTR_001 (Reliability):	The system shall recover from a crash within 1 minute.
VC_ATTR_002 (Availability):	The system shall be available 99.9% of the time during working hours (Monday through Friday, 8 AM to 6 PM).
VC_ATTR_003 (Security - RBAC):	Users shall only be able to access data and features permitted for their defined role (Student, Lecturer, Parent, Admin) as specified in the use cases.

4.3. Timing Verification

GANTT CHART

Verification Timeline

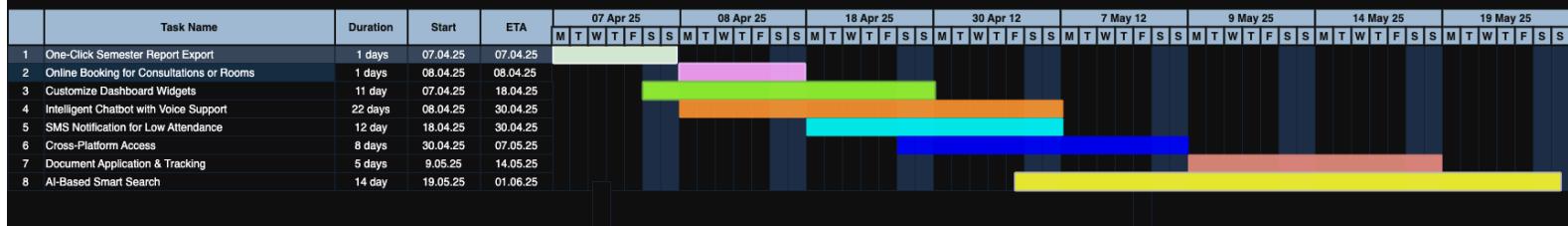


Figure 4.3.1 Gantt Chart

The method of production has been thoroughly tested in certain locations and conditions. These settings simulate real-world usage scenarios, ensuring the system functions consistently under different conditions.

Remote Access Testing:

Purpose:

The verification of the remote access testing is to ensure a smooth experience for users operating from off-campus locations.

Environment :

Users access the system using personal devices (laptops, tablets) from different places. To simulate normal remote usage, the remote testing environment should have stable internet access.

Computer Labs/Classrooms:

Purpose:

Its aim is to provide a regulated environment for students and instructors to interact with technology.

Environment:

Classrooms and labs are commonly used by students and lecturers, making them conveniently accessible.

University Administrative Offices

Purpose:

To test academic record management and student enrolment systems in a real-world administrative environment.

Environment:

Staff personnel access all of the systems (SIS, LMS, databases) through desktop PCs connected to the university network.

5. Appendices

5.1. Assumptions and Dependencies

Assumptions and Dependencies

User Availability

- Enough administrative staff and student participation are assumed for the testing and observation phases.

Technical Environment

- It is assumed that the required technical infrastructure—such as databases, email, messaging platforms, and authentication systems—is completely functional and compatible with East Blue Portal.

Consistent Usage

- In order to gather useful information on user interactions and system performance, it is assumed that users will interact with the software on a regular basis throughout the observation period.

Data Security

- All information transferred between East Blue Portal and the current university systems is presumed to be safe and impervious to unwanted access.

Dependencies

Integration with Existing Systems

- For East Blue Portal to operate as planned, it must successfully integrate with the university's Learning Management System (LMS) and Student Information System (SIS).

IT Support

- The university's IT department must provide ongoing assistance in order to resolve any technical problems that might come up during the testing and observation stages.

User Feedback

- Active and honest user feedback is important to the observation process' efficacy. Finding and fixing any usability or functionality problems requires this users feedback

Regulatory Compliance

- To guarantee that the implementation and use of East Blue Portal are morally and legally acceptable, loyalty to university policies and pertinent legal requirements (such as data protection laws) is required.

5.2. Acronyms and Abbreviations

- AI: Artificial Intelligence
- API: Application Programming Interface
- AWS: Amazon Web Services
- CGPA: Cumulative Grade Point Average
- CORS: Cross-Origin Resource Sharing
- CPU: Central Processing Unit
- CRUD: Create, Read, Update, Delete
- CSV: Comma-Separated Values
- DB: Database
- ERD: Entity Relationship Diagram
- GUI: Graphical User Interface
- HTML: HyperText Markup Language
- HTTPS: Hypertext Transfer Protocol Secure
- IEEE: Institute of Electrical and Electronics Engineers
- I/O: Input/Output
- ISO: International Organization for Standardization
- JSON: JavaScript Object Notation
- JWT: JSON Web Token
- LMS: Learning Management System
- NLP: Natural Language Processing
- OS: Operating System
- PDPA: Personal Data Protection Act (Malaysian context)
- PDF: Portable Document Format
- QA: Quality Assurance
- RAM: Random Access Memory
- RBAC: Role-Based Access Control
- RDS: Relational Database Service (AWS)
- S3: Simple Storage Service (AWS)
- SIS: Student Information System
- SMS: Short Message Service
- SRS: Software Requirements Specification
- SUS: System Usability Scale
- TLS: Transport Layer Security
- UAT: User Acceptance Testing
- UI: User Interface
- UX: User Experience