INTEGRATED TIMETABLE SYSTEM FOR CLASS SCHEDULING AND RESOURCE MANAGEMENT

BY

YAP ZHONG YEW

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONOURS) INFORMATION SYSTEMS ENGINEERING

Faculty of Information and Communication Technology

(Kampar Campus)

JUNE 2025

COPYRIGHT STATEMENT

This Final Year Project report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Information Systems (Honours) Information Systems Engineering at Universiti Tunku Abdul Rahman (UTAR). This Final Year Project report represents the work of the author, except where due acknowledgment has been made in the text. No part of this Final Year Project report may be reproduced, stored, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author or UTAR, in accordance with UTAR's Intellectual Property Policy.

ACKNOWLEDGEMENTS

I would like to express thanks and appreciation to my supervisor, Ts Dr Chan Lee Kwun and my moderator, Cik Ana Nabilah Binti Sa'uadi who have given me a golden opportunity to involve in the Internet of Things field study. Besides that, they have given me a lot of guidance in order to complete this project. When I was facing problems in this project, the advice from them always assists me in overcoming the problems. Again, a million thanks to my supervisor and moderator.

Other than that, I would like to thank my family who has provided a lot of assistance to me when completing this project. Although both of us are having different project and task scope, he is still willing to support me when I faced difficulties in developing this project.

ABSTRACT

Due to the growing complexity of scheduling courses at the university, the necessity for a quality and user comfortable scheduling system has emerged. It is very difficult to take advantage of surplus resources fully and allocate them effectively without affecting normal activities and constraints [13]. For students to be able to easily book courses, check timetables, and manage their resources better, the project proposes developing an integrated timetable system. The system allows students to arrange for classes, check the availability of teachers, view their schedules, and track changes or cancellations of lessons in real time. It gives students also complete information on the rooms where the classes and courses are taking place, including maps of the campus for classes. Thanks to the real-time information available about the schedules of lecturers and the changes for classes, students are alerted immediately whenever there are changes in the planned meeting times. Moreover, the application includes features which empower the management to control class schedules, teachers and classrooms combination to form a complete solution for students and staff members. The central aim of the system being proposed is that it will automate and centralize the designing process to reduce the incidence of scheduling conflicts, enhance resource allocation and improve learning experience for the educators and the learners.

Area of Study (Minimum 1 and Maximum 2): Timetable System, Website Application Development

Keywords (Minimum 5 and Maximum 10): Integrated Timetable System, Real-time Updates, Resource Allocation, Classroom Management, Course Scheduling System

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

OTP	One-time password
API	Application Programming Interface
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
UI	User Interface
IP	Integer Programming
RAM	Random Access Memory
DBMS	Database Management System
CRUD	Create Read Update Delete
HTML	Hypertext Markup Language

Chapter 1
Introduction

In this era of rapid technological advancement, efficient scheduling, automated timetable generators, efficient resource allocation, and class control are what are just needed by the education institutions. Most of the students face the same trouble, which is manually arranging the timetable for their courses. In a world with an increasing number of students, teachers, and courses available, there is a severe scarcity of effective and user-friendly tools for most of the common needs. The aim of developing an integrated timetable system is to address some of these issues. It is possible to resolve problems of scheduling and resource allocation in the context of academic classes at schools, colleges, and even universities. This system allows students to register for courses according to the existing course capacity, view their own timetables, and save them as PDFs if necessary. Since drag-and-drop is applied here, the user is able to drag the class to another available class for convenience in doing comparisons with both times chosen. Furthermore, it will ensure that all students can get the most recent information regarding their course details by indicating where the courses are being held. The system contains many features, such as course booking, dynamic timetable views, and notifications received for class cancellation. Besides that, the class time will display on the student's personal calendar, while the class cancellation is updating on time. Students can manage their schedules in an efficient and convenient way. At the same time, administrators have permission to easily manage course information, lecturer assignments, and room allocations. Administrators can see the resource utilization rate in graph view for better allocating the resources. By automating many of the scheduling tasks and offering real-time updates, the system aims to improve resource utilization, reduce conflicts, and enhance the overall academic experience for both students and educators. In conclusion, the project aims to provide a comprehensive solution for the modern educational institution's scheduling demands, ensuring that the process is as efficient, transparent, and student-centered as possible.

1.1 Problem Statement and Motivation

To start a project, we should define the problem statements. The first problem statement is **inefficient manual scheduling**. Managing the timetable manually is happening to most of the students. Traditional timetable creation is a technique that easily gets wrong on scheduling. [5]. It's because most of the parts of the timetable completion are done by hand, which brings a frequent complaint with inefficiencies, mistakes, and difficulties in educational institutions. [5] Such inefficiencies highlight the pressing need for an automated system to streamline scheduling and address these challenges. Besides that, the second problem statement is **difficulty in obje**. Confirming a class status is a difficult challenge for administrators. Soft constraints are usually based on teacher or student preference, while hard constraints are usually based on student or teacher availability [6]. Teachers often have overlapping tasks or accidental situations that make it hard to create a schedule that fulfills their availability while meeting the needs of students. Students cannot know the situation on time since some teachers do not have free time to announce to students. Without a centralized system to track and optimize teacher schedules, administrators struggle to avoid conflicts, which negatively impacts both teaching quality and the student experience. Then, the third problem statement is **lack of real-time information for students**. A system that incorporates up-to-date train status information can provide a user with valid timetable information in the presence of disturbances. [7]. Students always face difficulties in accessing real-time updates about their schedules, such as class cancellations or teacher unavailability. This lack of immediate updates leads to confusion and wasted time, affecting students' ability to plan their academic activities effectively. Furthermore, the fourth problem statement is **inconvenient course scheduling**. Scheduling courses in higher education is one of the challenges in their educational life. In today's climate, reducing operational expenses is essential, making manual timetable development seem inefficient when it can be automated and optimized [8]. The absent features for course registration make challenges for students. Students need to do many surveys or check information before registering for the courses. This inconvenience obstructs students' ability to engage with their academic responsibilities in an efficient way. After this, the problem statement is **limited administrative control and resource optimization**. The institution assigns the courses taken by students and delivered by tutors to a defined finite set of resources, which includes time slots and classrooms. This process presents many challenges [9]. Administrators work hard to manage the allocation of resources such as classrooms, lecturers,

and time slots. Furthermore, the absence of an efficient platform makes it difficult to adapt quickly to changes, further aggravating the challenges of managing a dynamic academic environment. For motivation, the integrated timetable system is for all the users to bid their timetable for every semester. This system offers several functions that help to make users more convenient. First, one of the functions involved in this system is having the latest notifications or news, like holidays. When the user registers and logs in, the system will navigate the user to the dashboard page, which displays the latest news or notifications. Besides that, the web application allows users to schedule his/her classes and ensures no conflicts in time. It also assigns teachers to the classes at appropriate times. Then, resource management is a function provided in the web application that was used to manage room availability. Furthermore, web development provides a chat system to allow users to communicate with each other. This function is used to change the culture of students. Each of them can exchange their insight or comment on courses or teachers. In this project system, the XAMPP software is utilized to access the database layer easily. Before running a project system, there are 2 services called Apache and MySQL that are required to be started before running the project system. Then, phpMyAdmin is a useful website to create multiple tables and match them by adding foreign keys to the database. With this function, developers can more easily manage the data in the database through the CRUD operation.

1.2 Objectives

There are four objectives in this project. The main goal of this project is to develop an integrated timetable system for efficiency in the timetable. The first objective needed to be achieved is to **define the distinct functions for each role**. The system design has different functions for each role: admin, student, and lecturer. In order to increase the smooth experiences for users when using the web application.

The second objective of the project is to **define the dynamic timetabling**. The system aims to improve the timetabling from the traditional method. It allows users to register for the course by dragging and dropping the timetable to make it convenient to see the comparison of different time slots. Automated course scheduling, including the constraint rules, can be done to improve the user experience.

The third objective of the project is to **design a personal calendar**. The system aims to improve the user experience. A calendar with auto-written class display on the student dashboard to make it easier for the user to check whether they have class or not. There is real-time updates with a red mark when class is cancelled on a class date.

The last objective of the project is to **develop a transparency resource utilization view**. The system aims to improve the resource allocation. The admin dashboard displays information on room utilization and course utilization directly. To make sure the resource can be allocated in a better manner.

1.3 Project Scope and Direction

The Integrated Timetable System for Class Scheduling and Resource Management seeks to develop a comprehensive solution that solves the challenges faced by educational institutions in scheduling courses, class status, and resource management. This system will provide a seamless experience for students, lecturers, and administrators by automating the entire scheduling process. The scope of this project contains the design, development, and implementation of a system that optimizes the scheduling of courses, improves resource allocation, and facilitates real time updates.

One of the points of the system will be its user interface for students; it allows students to manage their courses easily. Students are able to check course availability based on their specific program and choose the time slot to suit their needs. The system will inform students on time about any class changes, ensuring they will not miss important announcements. Additionally, there is an auto-emailing to students if there are any class cancellations. The status of the class will display on the student calendar and in announcements.

The system will provide a user-friendly interface for lecturers where they can set their availability, view their courses, and stay informed about any schedule changes or cancellations. Teachers will be notified on time about any adjustments to their teaching schedules, including updates to class timings or venue assignments. This feature aims to reduce scheduling conflicts and ensure that lecturers are fully aware of their teaching commitments.

A significant aspect of this system is its resource management capabilities. The software will ensure that classrooms and other resources are utilized efficiently, helping to prevent overbooking or underutilization. The system will consider factors such as room size, required equipment, and teacher availability to make sure resources are allocated optimally. Furthermore, it will provide reports and analytics that allow administrators to monitor room usage and make data-driven decisions for future scheduling.

In conclusion, the scope of this project is to design and develop an integrated timetable system that addresses the key scheduling challenges faced by educational institutions. By automating the process of course booking, timetable management, and resource allocation, the system will enhance the efficiency of scheduling operations, improve communication between students,

teachers, and administrators, and optimize the use of resources. Ultimately, the project seeks to create a more efficient and streamlined academic environment that benefits all stakeholders involved.

1.4 Contributions

The Integrated Timetable System for Class Scheduling and Resource Management gives contributions to educational institutions by addressing several challenges related to course scheduling and resource allocation. By automating the creation and management process of the timetable, this project will improve the efficiency of scheduling and improve student experience.

One of the primary contributions of the project is its ability to improve resource utilization. Educational institutions often headache with scheduling conflicts, overbooked classrooms, and underutilized teaching resources. With this integrated timetable system, administrators can effectively manage classrooms, ensuring that rooms are allocated based on course requirements such as size, equipment, and accessibility. Additionally, the system will help reduce scheduling conflicts for both teachers and students by automatically checking for clashes in course timings, class status, and class assignment. This optimization brings better utilization of resources and a seamless academic experience for all users.

The project will also significantly improve the user experience for students, making course booking and timetable management more transparent. The system's real-time notifications about course updates and class cancellations will make students more convenient. By providing an interactive timetable view and easy access to course information, students will be empowered to manage their academic schedules more effectively, with drag-and-drop applied to do comparisons with each available class timeslot. There is a personal calendar for students to track on. It displays the class and class cancellations with a red mark.

Moreover, there is an important contribution on enhancing administrative control on the system. Administrators can easily track room availability, course availability, classroom assignment, and user information. This smooth process improves the administrative workflow, saving time and reducing the potential for human errors that might occur in manual scheduling methods.

Thus, admin can improve decision making on utilizing resources on viewing the resources utilization rates in graph view.

In summary, the Integrated Timetable System will bring important contributions to the teacher and student. It gives a big help in scheduling a timetable with base-rule filtering. There is an error message when detecting any conflicts on resources. The project will not only bring good feelings to the lecturer and student but also bring a surprise for the educational institution.

The prototype consists of the following 5 modules:

Module 1: Student Module

1. Display enrolled course
2. Check announcement
3. Real-time update schedule on calendar

Module 2: Course Module

1. Register course without conflict
2. Recommendation on class
3. Apply based-rule filtering

Module 3: Timetable Module

1. Save as PDF
2. Drag and drop feature implemented
3. Change available class directly

Module 4: Lecturer Module

1. Check student list
2. Publish announcements
3. Class cancelling
4. Check schedule

Module 5: Admin Module

1. Manage user, room, course and classroom assignment
2. View resource utilization rate in graph view

1.5 Report Organization

This report is written in 7 chapters. Each chapter records every important part for the project. In chapter 1, the project is introduced based on the demands of the roles of students and teachers. It included the problem statement, motivation, objectives, project scope and direction, and the contributions to develop the project. Besides that, problem statements and objectives are defined in the project. This chapter will explain the project scope of the project that included functions offered by the systems. Chapter 2 is a literature review. Comparing the existing software or websites related to my project is included in this chapter for reference purposes. In Chapter 3, the system methodology applied while doing this project is recorded, and a Gantt chart is drawn to estimate the time of completion of every stage. Drawing system architecture diagrams, use case diagrams with descriptions, and activity diagrams. For Chapter 4, a system block diagram has been drawn based on the modules in my system, and an ERD diagram and System Components Interaction Operation are shown in the next section. Moreover, hardware and software setup, setting and configuration, and system operation will be included in Chapter 5. This chapter teaches how to set up the environment before deploying the system. The step is very important if the user is a newbie at developing a system. The project issues and challenges are also recorded in this chapter for revision purposes. In Chapter 6, system evaluation and discussion will be included. The test case is processed to output the testing result. This chapter plays an important role in testing the integrated timetable system that was developed over a long time. Lastly, the conclusion and recommendation for this project will be written in Chapter 7. This emphasizes the future work to improve the project.

Chapter 2

Literature Review

2.1 Existing System of Timetable

In this era, educational institutions offered a variety of courses and got complicated when scheduling timetables. The increasing number of programs and students and the availability of resources like lecture halls or lecturers will make students suffer. Thus, timetable management is an important function for students, especially those undergraduate students. I researched several timetable systems available that aim to make scheduling courses easier. Gizmoa [15], The Timetable Factory [16], Skooly [18], PrimeTable [17], and OpenEducat [19] offer web versions, while the few articles are "A genetic algorithm-based university timetabling system" [14], "Automatic timetable generation using genetic algorithm" [4], "Implementation of a university course and examination timetabling system" [3], and "Timetable generation system" [2]. They need to pay attention to choosing a time slot for each course and consider the possibility of overlapping time slots. Over the years, humans have investigated some approaches to address or reduce these problems; they can be separated into manual methods and semi-automated systems with their unique strengths and limitations. [1] I will do the review for the existing system to compare with my system. In order to increase the quality of the system and improve the functions of the system.

2.1.1 Gizmoa

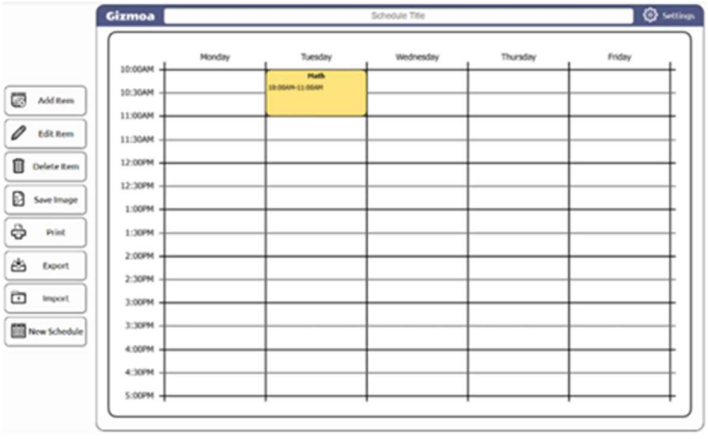


Figure 2.1: College Schedule Maker [15]

Gizmoa contains 2 main functions: College Schedule Maker and Grade Calculator. College schedule makers are built to help users create a weekly schedule for their school or college in a few minutes. The schedule can be manipulated at any time. The schedule is flexible so that the user can set Sunday as the first day of the week, or the system will resize to fit those times if the user has classes on the weekend. College Schedule Maker contains many functions, like adding items, editing items, deleting items, saving them as an image, printing, exporting and importing a CSMO file, and creating a new schedule if the user wants to reschedule.

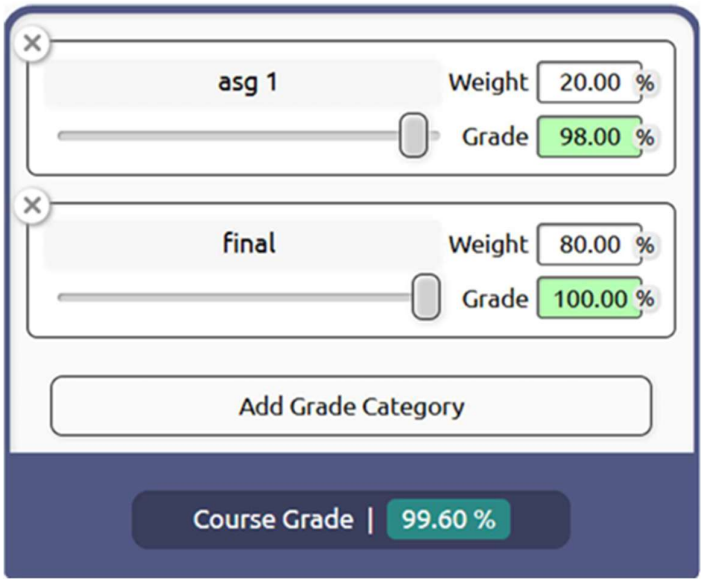


Figure 2.2: Grade Calculator [15]

Furthermore, Grade Calculator is built to compute a user's weighted course grade; it helps users to have mental preparation for their final grade. Also, users can use it to calculate how many marks are needed to score in the final examination. The system is computing the user's weighted course grade using the following formula:

*Course Grade = (Weight1 * Grade1) + (Weight2 * Grade2) + ... (Weightn * Graden)*

The limitations of Gizmoa's College Schedule Maker are the advanced features. The tool focuses on simple features only and can lack many functions like detecting conflict, recurring events, or integrating with a calendar system.

2.1.2 The Timetable Factory

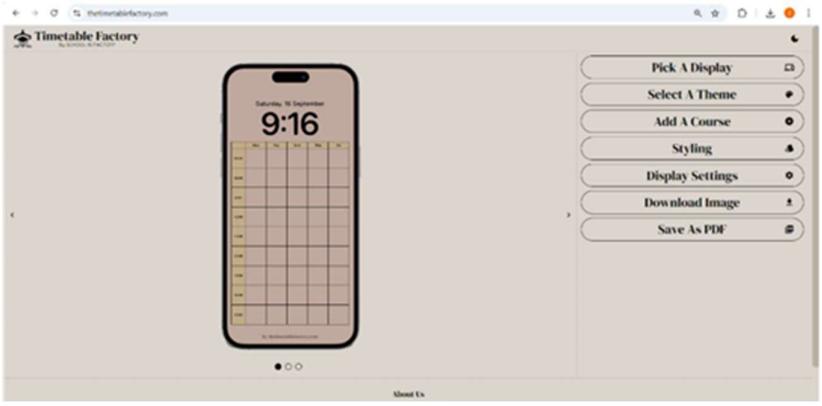


Figure 2.3: Timetable Factory [16]

The Timetable Factory is a web-based timetable design that lets users edit a timetable preference and save it as an image to fit either a phone, an iPad, or an A4 size. There are several settings, like selecting themes, styling, adding courses, and display settings**.**

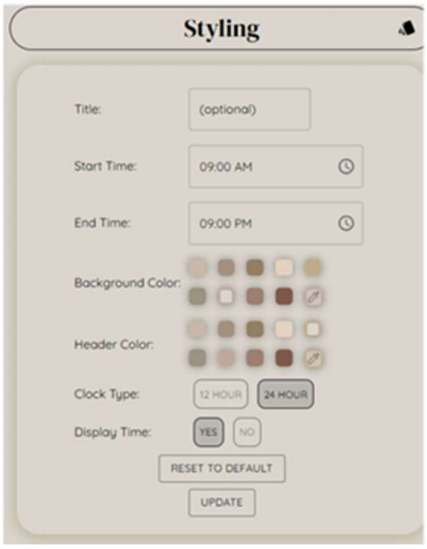


Figure 2.4: Styling [16]

For styling functions, it is able to let the user change the background color, header color, clock type, etc. The designed timetable is attractive to decorate the user's device. The limitation of the system is limited functionality; the tool only focuses on creating visual timetables and lacks advanced features like automatic scheduling. Device compatibility is one of the limitations that work well on desktop browsers only, affecting user experience on other devices like smartphones.

2.1.3 Skooly

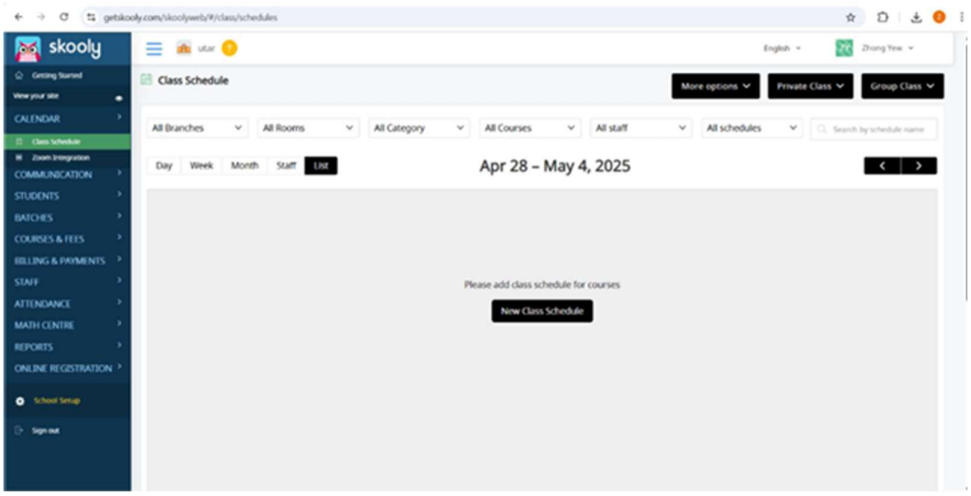


Figure 2.5 Skooly [18]

Skooly is an education platform to help users manage class schedules, check the courses and fees, and send notifications in communication functions. There is an important function that is the attendance feature that lets users check in or apply for leave. That site can check student reports if the user's role is an admin or teacher.

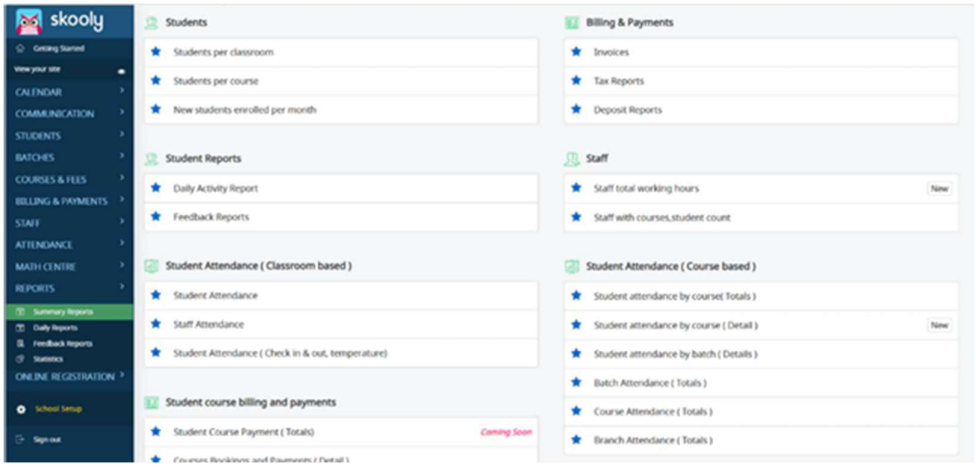


Figure 2.6 Skooly const. [18]

Furthermore, users can check the reports in the student category, like daily activity reports, feedback reports, new students enrolled per month, student attendance, student course billing, and payments. Also, users can check tax and deposit reports. While in the staff category, users can check total working hours and what courses are assigned to staff. Then, users can check the student attendance by course or batch in detail. Users can set the training plans, assessments, course lessons, and tasks online. To make the student easier to access.

The limitations of Skooly are its pricing structure, as the pricing plan is set at $99 per month. It is not so affordable to manage schools or campuses.

2.1.4 PrimeTimetable

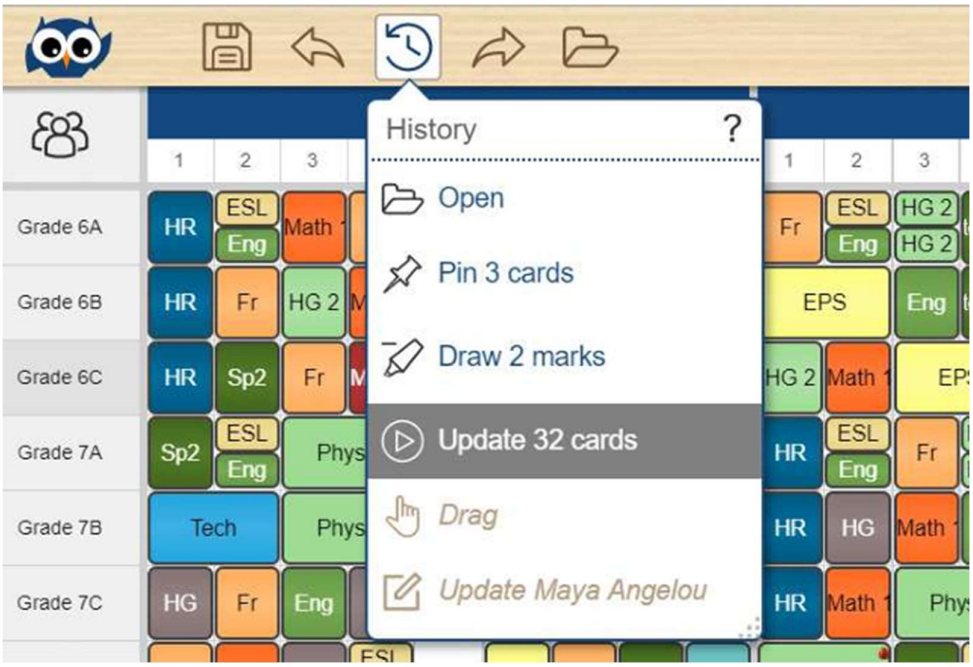


Figure 2.7 PrimeTimetable [17]

PrimeTimetable is comprehensive school scheduling software designed for users to do automatic or manual scheduling processes. First, creating and managing timetables is one of the core functions to initiate new timetables, edit, and manage them. Users can import data like days, periods, subjects, rooms, teachers, classes, and groups with CSV files. It is able to combine and remove timetables. To make the system more attractive and easier to use, the user can drag and drop the cards. The cards can be customized and personalized. Also, the system contains conflict management to avoid scheduling conflict by utilizing a sophisticated algorithm.

The limitations of PrimeTimetable are conflict management; users need to make manual adjustments for conflicts in very complex situations. Then, training is required for new users, although the system is designed to be user-friendly. Users still need to go through the training to fully understand using all the features.

2.1.5 OpenEducat

Figure 2.8 OpenEduCat [19]

OpenEduCat is an open-source ERP solution designed to streamline the management of institutions. There are several modules, including student, faculty, course, examination, and financial. Each module oversees their own role, like storing student records, including health information, academic history, and so on, in student modules. Besides, users can manage faculty profiles, courses, schedule exams, track expenses, and so on.

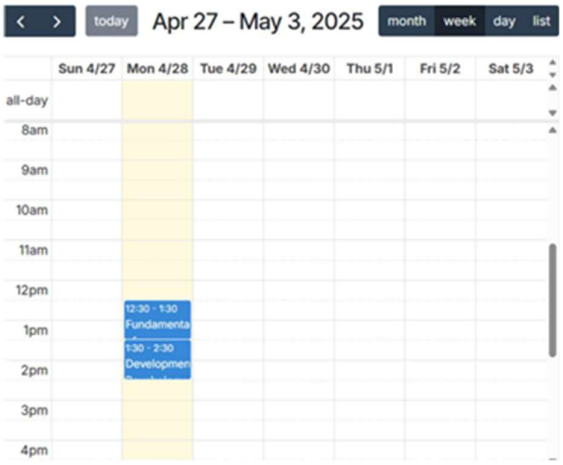


Figure 2.9 Timetable management [19]

User can do many things in the modules like monitoring student and faculty attendance, collecting assignments, creating and managing class schedules, handling leave requests for staff and students. The system automatically checks for conflicts in the timetable, like

overlapping classrooms. Automatic timetabling is also a feature of OpenEduCat, system considers available rooms, courses and other constraints to create an optimized timetable.
The limitations of OpenEduCat are complex setup and maintenance, setting up configuration before starting to use is hard and time consuming. Timetable views are one of the limitations in customization, it only provides basic timetable views. Then, manual adjustments should be made by the user even after automatic timetabling for special cases.

2.1.6 University Course and Examination Timetabling System

This paper discusses an integer programming (IP) that is used to design an examination timetable system with integrated university course. The system can resolve conflicts between courses, teacher availability, and room constraints when generating a schedule. It uses both IP models and heuristic algorithms to create timetables. IP is a mathematical optimization method used to represent binary conditions. Also, the model is used to minimize or maximize a specific goal. After that, a heuristic algorithm is applied to further improve the timetable.

2.1.7 Automatic Timetable Generation using Genetic Algorithm

This paper discusses automatic timetabling processes using genetic algorithms (GA). Genetic Algorithm is used to search for problems and find solutions to address. It involves some operations like selection, crossover, and mutation. In this paper, it describes each of the timetable solutions called a chromosome, and it encodes the time slots and rooms. Furthermore, hard and soft constraints also applied. Operation crossovers are undergone by selected chromosomes, which combine two chromosomes to become a new chromosome (timetable). Operation Mutation prevents premature convergence by altering some parts of the timetable.

2.1.8 Complex University Course Timetabling

This paper discusses the system to address complicated university-wide timetabling problems, which contain multiple departments, rooms, and large numbers of students and lecturers. The algorithm used by the system to iterate forward search to explore and find timetables. It can reduce conflicts and adjust the timetable dynamics when constraints change.

2.1.9 Timetable Generation System

This paper proposes a practical timetable generation system that integrates both hard and soft constraints to generate timetables automatically. All the information, including teacher, course,

room, day, and timeslot information, is provided via forms and stored in a knowledge base. Hard constraints are applied in this system to ensure that no overlapping classes are for the same student, teacher, or room. Soft constraints are applied in minimizing gaps between classes and faculty.

2.1.10 Manual Scheduling

Manual scheduling was the norm for a lot of institutions. This system depends on administrators manually assigning courses, teachers, and resources in spreadsheets or paper systems. It's easy and cheap, but manual scheduling is very time-consuming and prone to human error. There is often double booking, underutilization, and scheduling conflict, which interrupts the academic workflow. [1]

2.1.11 Automated Scheduling

Automatic scheduling is an innovative approach to replace the wasted time and headache of manual and semi-automatic timetable management. These algorithms take advantage of computational strategies like genetic algorithms, integer programming, and heuristic models to create optimal schedules that obey the stringent constraints. Through automating functions such as resource allocation, conflict recognition, and schedule creation, automated systems decrease the administrative burden and human error rate dramatically. They can handle a variety of needs, including the need to work with teacher time, room space, and course requisites, while still maintaining schedules that are both realistic and feasible. Also, automated calendars can often be updated in real-time, so students and staff are notified of shift changes or cancellations on the spot. These systems also enable dynamic flexibility for adjusting at short notice to unexpected events, like sudden room or lecturer disruption. With their capability to process a large volume of data and various constraints in the same time, automated scheduling is a must-have technology in today's schools, which allows for an efficient and streamlined scheduling experience for everyone involved. [2] Table 2.1 provides a comparison of functionalities among existing systems.

2.2 Summary Review

Table 2.1 Summary Review

Feature	1: University Course Timetabling	2: Genetic Algorithm based System	3: Skooly	4: OpenEduCat	5: Prime Timetable	6: Gizmoa	7: The Timetable Factory	8: Timetable Generation System	9: Integrated Timetable System
Strong Constraint Handling	√	√	√	√	√	√	√	√	√
Soft Constraint Handling		√	√	√		√	√	√	√
Optimization Techniques	IP and heuristic	Genetic	Built-in tool	Open-source ERP System	Built-in tool	Built-in tool	Aesthetic timetable design	Automated system with XML knowledge base	Rule-based Filtering, Interactive User Optimization
Real-Time Scheduling Updates				√	√			√	√
Resource Management				√			√	√	√
Flexibility in User Input	√	√	√	√	√	√	√	√	√

2.2.1 Limitation of Existing Application

Based on Table 2.1, I reviewed 9 existing systems that related to my project title, which is timetable management. The system with a genetic algorithm handles strong and soft constraints, which ensure essential rules are followed to meet preferences. There is some importance to soft constraints. Honors and general courses need to be scheduled in non-overlapping timeslots [10]. The number of students taking a course must fit into the assigned room [11]. To open the course, a minimum number of students should be registered [12]. The second system is similar to the first system; it can address both strong and soft constraints. For the third system, it focuses on strong constraints but limited flexibility for preferences. For example, as a hard constraint, a room cannot be assigned to more than one lecture in each period [11]. The timetable generation system will be like a genetic algorithm-based system that handles both strong and soft constraints. Besides that, one of the important features that is real-time scheduling updates is only implemented in Timetable Generation on the system and the

web tool of Skooly and OpenEduCat; the others did not have the feature to inform students and lecturers on time. Furthermore, the first and second systems can manage a very limited resource, which means that they may have errors when interacting with complex situations. For the OpenEduCat web tool, it offered comprehensive resource management, which is better than the 2 systems I mentioned just now. For the last system, it handles classroom and student flexibility constraints. Then, for flexibility in user input, the first 2 systems offer moderate flexibility, which allows users to input basic constraints only. For the university course timetabling, they utilize IP with heuristic methods, which allow users to customize constraints. Lastly, the first 2 systems with genetic algorithms can optimize timetables iteratively by simulating the natural selection process; however, this approach is effective for complex situations. The last system is my integrated timetable system; it applies strong and soft constraints. Using optimization techniques like rule-based filtering and interactive user optimization. For example, rule-based filtering helps to prevent time overlap, class overlap, and so on. Interactive user optimization can help to improve user experience, like the drag-and-drop feature implemented in the timetable.

2.3 Fact Findings

For retrieving information, I will do document reviewing, brainstorming based on the previous timetable, research allocation records, and reports on scheduling. Besides that, I will also interview lecturers and students to get feedback on the previous scheduling method and the responsibility of administrators. I try to know more about available constraints and preferences. I will observe how the administrators create a timetable to identify which manual step can be automated. Lastly, I will prepare a list of questions and distribute the surveys to the targeted audience, which are students and lecturers, for gathering feedback purposes.

2.4 Critical Remarks of Previous Work

After reviewing the system with genetic algorithms, IP-based models, and other web tools, I am trying to evaluate their strengths and limitations for better development in my own system. Systems with genetic algorithms focus on constraint but lack real-time adaptability. I will try to do the update on time for the students and lecturers. The IP-based model provides real-time scheduling updates but is soft in resource management. Then, there are many features I can take as reference from the web tools, like styling functions, enabling users to change

background color, font size, etc. I will combine the strengths of both existing systems and address their limitations.

Chapter 3

System Methodology/Approach

Agile methodology is the method to apply in this project; it separates into a few stages, having a planning phase, a design phase, a development phase, a testing phase, a deployment phase, and a review phase. Agile methodology is used to let me more clearly understand step-by-step to complete a project. Each phase is important to lead the success of the project, including identifying what and how to do.

3.1.1 System Architecture Diagram

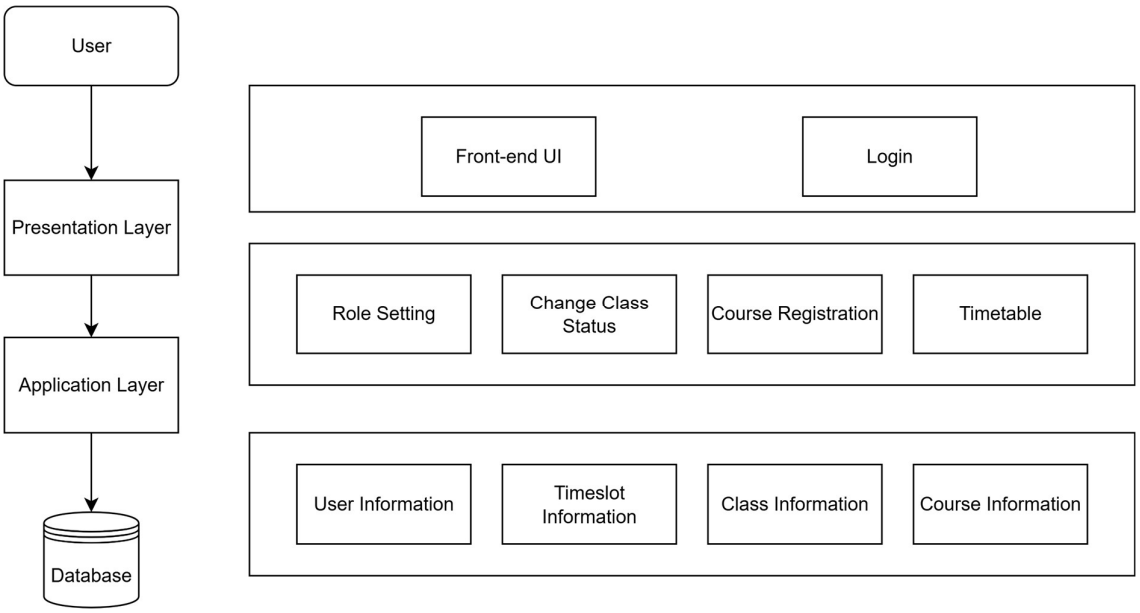


Figure 3.1: Architecture Diagram

This project system involved the 3-tier architecture pattern, which included the presentation layer, the application logic layer, and the data layer. The presentation layer is showing the user interface. Hence, the user will interact with the system through the user interface. The user interface is created using PHP, HTML, CSS, and JavaScript languages, and it will display on the website after connecting the database. After that, the application logic layer is used to respond to the request by the interaction between the user and the system. This layer will handle the interaction and communication between the presentation layer and application logic layer. The bottom layer, which is the data layer, is used to store data in a database. The data can be managed through the CRUD operation. This allows the system to respond to the user's request

quickly by retrieving data. In the timetable system website development, there is a dashboard page displayed to the user in the presentation layer. The application logic layer will process the validation for the data of the system, such as registering the course, timeslot list, and other functions. For the data layer, we will store all the user, course, and timeslot information.

3.1.2 Use Case Diagram and Description

Figure 3.1.2: Use Case Diagram

The use case diagram above illustrates the functionalities of website development. This system includes basic security functions; there are login, register, and reset password functions. The

reset function is developed using a PHPMailer API. It helps to send an OTP code to a real email and require the OTP to be inputted in 5 minutes. When registering an account as a student, the user should choose the intake semester, while the option is hidden when the role as lecturer is selected. After that, the data will be recorded in databases, since the roles "student" and "lecturer" have separate tables. Thus, the system will navigate users to their dashboard based on each role. In the dashboard, students can check the enrolled courses, latest announcement, and status of class in a calendar. Students can write notes on the calendar freely. Once there is any class cancellation, the student can see it very clearly on the calendar. Besides that, users can go to the course registration page to register for the course. When registering for courses, users can choose the courses, and with applying based-rule filtering, the system will prevent users from registering when conflicts occur. Then, students can go to the timetable page to view and edit their own timetable by drag-and-drop features. Students can drag the class to another available class without any conflict. Students are allowed to save the timetable as a PDF. It makes it easier for students to do comparisons with the available timeslots. It will directly change the register information, so there is no need to go back to the course registration page. For the lecturers' site, the student list can be checked, and announcements can be published to a specific class. The lecturer is able to cancel the class on the correct date and update it on time in the student dashboard. Students can check back on the calendar; the class cancellation will show with a red mark and in the announcement. On the other hand, all the users, courses, rooms, and lecturers' information will be managed by an admin. To optimize the allocation of resources, the admin dashboard displays the resource utilization rate in graph view in order to make good use of each resource. The admin account is special because the user can only log in with the username and password of "admin."

3.1.3 Activity Diagram

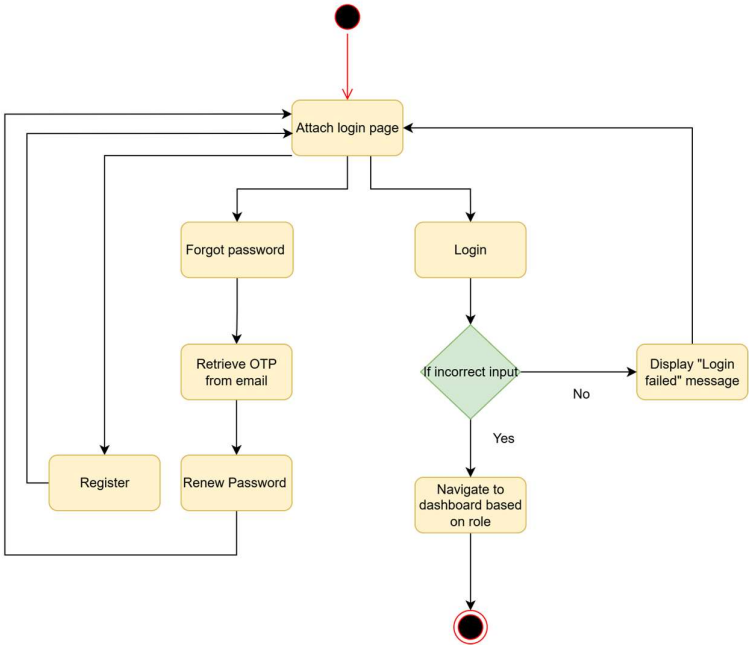


Figure 3.1.3: The Process of Logging Page

Based on the figure 3.3 activity diagram, it showed that when a user attaches to the login page, if either the username or password is wrong, it will display a “login failed” message and let the user try again; if the username and password are correct, it will navigate the user to the dashboard based on the role. Then, when users forget passwords or need to register an account, the system will send an OTP to the user’s email. There is only 5 minutes to validate the OTP, and the user is required to type it in. Then the user can renew the password or register successfully.

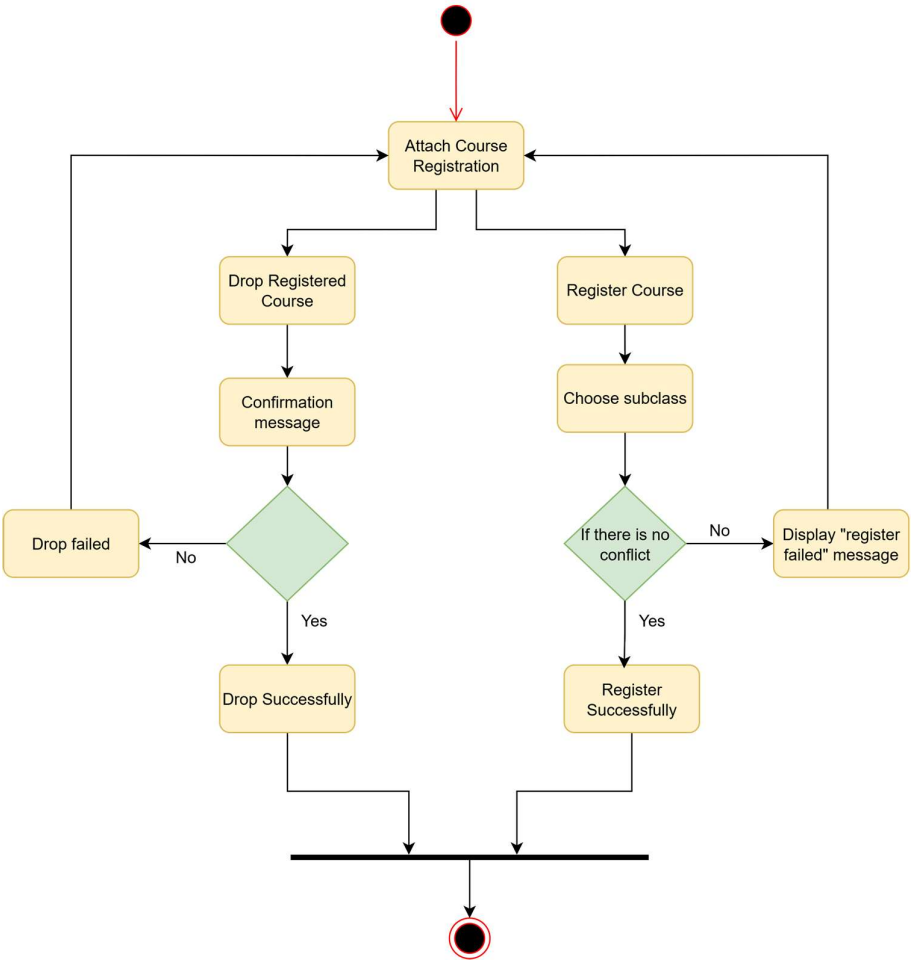


Figure 3.1.4: The Process of Course Registration

Figure 3.1.4 above showed the activity diagram when users registered courses. There are some courses in specific majors, so users can only register for courses that are in their major. Then, if the course with a specific time slot is already registered, the system will display a “register failed” message. Users also can drop registered courses; it confirms the message and drops successfully; otherwise, the drop action fails.

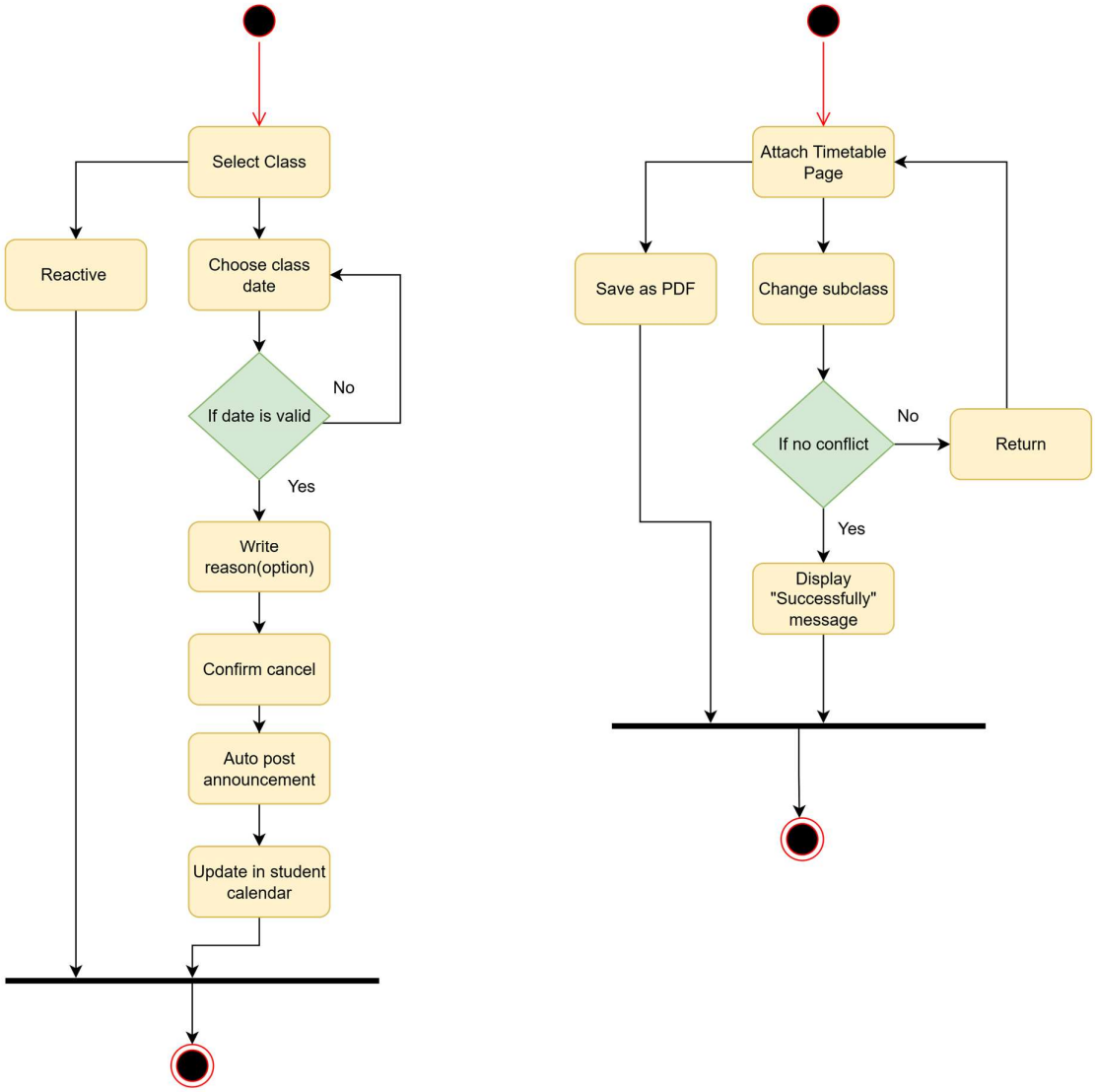


Figure 3.1.5: The Process of Changing status and timetable

Figure 3.1.5 above displays the activity diagram of changing status if class and viewing the timetable. For the class status, it allow lecturer to cancel a class on correct date, the system prevent the class cancellation if the date is not correct, once the action successfully, there is real-time updating in student dashboard, student can check on announcement and personal calendar. Lastly, the timetable page can be save as PDF if necessary. A drag-and-drop feature applied here to allow student drag the class to another available class directly.

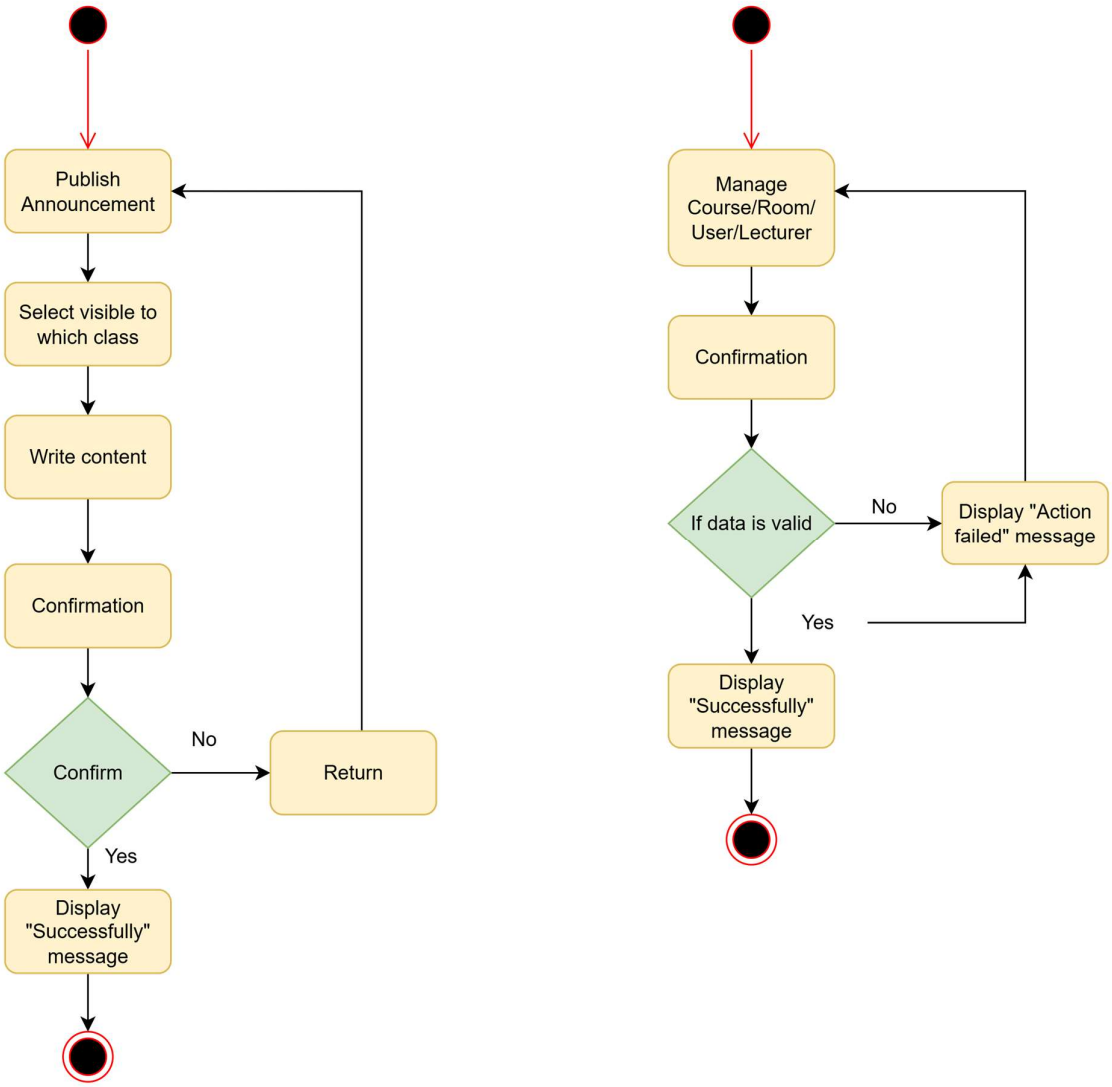


Figure 3.1.6: The Process of Publishing Announcement and Data management

Figure 3.1.6 above shows the activity diagram of notification and data management. For notification, lecturers can publish announcements for students if there is any emergency situation like class cancellation. Then, the admin can manage the data of course, room, user, and lecturer to make sure they are providing smooth experiences for users.

3.2 Project Verification Plan

The project verification plan is made to ensure the project meets the requirements. The verification plan involves the completion of coding and the database structure. Firstly, the expected output is when users want to reset a password or register an account. Users will get an OTP code from email in time and successfully reset their password or register an account. I will keep trying to type a real email for the system so that it can really send an OTP code to the email. After that, normal users will be separated into 2 roles: one is student, and the other is lecturer. And I will ensure each role navigates to their own dashboard. Then, for student roles, users can register the courses with specific time slots. There is some strong constraint applied to prevent illogical registration. For example, there is no overlap in class time and no exceeding the credit hour of the registered course. The available course will only show a specific semester. After that, the timetable will be kept checking to see whether there is a wrong timetable. Since the drag-and-drop feature is applied here, dragging a class to another available class is convenient for the user. For the lecturer part, the student list can be checked under the lecturer's courses; hence, I will try to compare the database table with the student list. The lecturer can publish announcements to specific class students. I will do verification by taking the lecturer course and trying the function. Class cancellation is one of the lecturer's functions; I try to cancel a class on an invalid date, and it exactly fails to cancel. After cancelling a class, the status will update on the student dashboard and calendar. I will test it by inserting some data into the table, ensuring there is a foreign key match to other tables, and making changes to the data.

3.3 Timeline

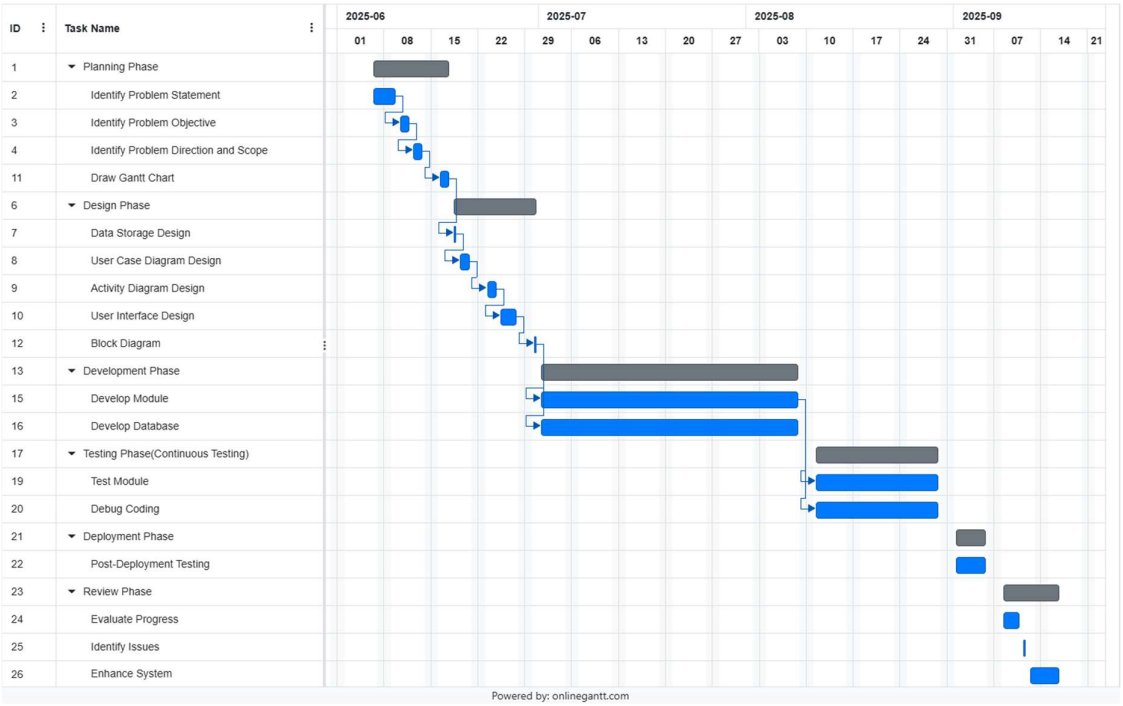


Figure 3.3 Gantt Chart

Figure 3.3 above displays a Gantt chart that illustrates the schedule of the development project. The beginning phase of this project using Agile Methodology is the planning phase. I do research and conduct a few existing timetable systems first. Then I start identifying the problem statement, project objective, project direction, and scope. I draw a Gantt chart to record and estimate the completion time of the project after finishing the basic work before starting a project. After that, I will be moving to the next phase, the design phase. In this phase, design the data storage, use case diagram, system architecture, and user interface design. I will draw the block diagram as well. Furthermore, the next stage is the development stage. The stage is starting by developing an expected user interface as designed before and even a module. The stage is important, so I will use most of the time to do development. The module is developing functionally while developing a database to ensure the module can work with the database. Besides that, the next 2 stages are working simultaneously; the module testing and debug coding are being done at the same time. It helps to fix the errors and bugs of my system. Then, there is a deployment phase. A post-deployment testing process will be done. Last but not least, there is a review phase; I will do an evaluation on my project progress and identify the existing issues. I will do the last enhancement to make it close to perfect.

Chapter 4

System Design

4.1 System Block Diagram

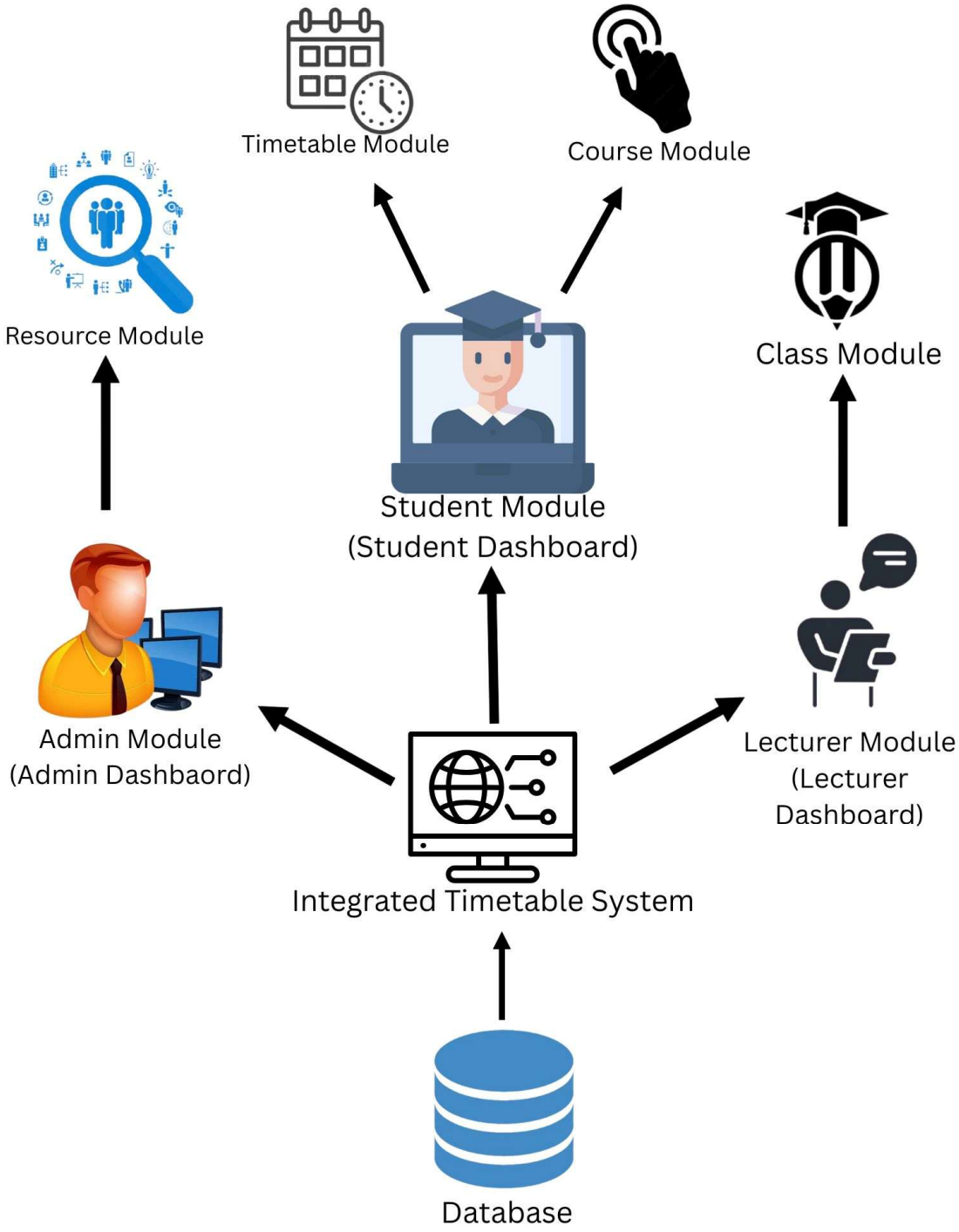


Figure 4.1 System Block Diagram

The integrated timetable system is structured into several components. Presenting in a system block diagram. There are a total of 6 modules I developed working aligned with the integrated timetable system while interacting with a database.

Students can log in to their own accounts and navigate to **the student dashboard**. On the dashboard, students can check the announcement and enrolled courses; below is the personal calendar through using a calendar API. The student is able to register for a course in the course module; there is a good setting that when a student needs to drop a course, the system will auto-drop the same course with the subclass. There is a timetable with drag-and-drop; it helps to do comparisons with other available timeslots of classes more easily. In the student dashboard, it is able to check class cancellation and add any notes on a personal calendar, while it will update on the announcement as well. The frontend UI is built using the languages HTML, CSS, and JavaScript. It interacts with the database with a PHP script. The database is needed to retrieve data and do CRUD actions by communicating with it.

Lecturers can log in to their own accounts and navigate to **the lecturer dashboard**. The dashboard allows lecturers to track student names, publish announcements, and cancel class while sending emails to students and updating the class status on the student dashboard. The announcement posted by the lecturer is only visible to the specific class of students. The action is done by communicating with the database using a PHP script. A PHP script can handle the data transmission. The lecturer frontend UI is developed using the languages HTML, CSS, and JavaScript.

Administrators access the **admin dashboard.** The system is able to manage the user account, editing, and tracking. Managing resources is necessary, like room, classroom assignment, and course information. When managing resources, the system is strong in logic to prevent any conflict like room capacity, timeslot overlap, and so on. There is a resource utilization rate in the graph view displayed in the admin dashboard in order to maintain and improve the efficiency and quality of allocating resources. All the logic is based on coding without an AI mechanism. The admin dashboard frontend UI is designed using the languages HTML, CSS, and Javascript. The frontend UI is interacting with the database through PHP script to do CRUD operations.

4.2 System Component Specification

The section is describing the system component that separates into frontend and backend components. This part will start with frontend components. There is a dashboard page after logging in with a student or lecturer account. The student dashboard page provides the number of enrolled courses, a personal calendar, announcements, and a course registration page. For the lecturer dashboard, the system allows them to notify students of class cancellation, track student names, and publish announcements. Also, the lecturer can check their own class schedule. For the admin page, it enables the admin to manage resources like courses, rooms, and class assignment information. It can manage user accounts as well. The important thing is admin can view the resource utilization rate in graph view to allocate the resources in a good manner. It helps to improve the efficiency of working.

For the backend component, there is an authentication module; it handles the validated login credentials by interacting with the database. It helps manage the role-based access and navigation dashboard. There is a scheduling module to handle course registration requests. It will detect the conflict and prevent registration from the student. It simulates the real university registration system. It helps to allocate classroom resources dynamically. For the notification module, it helps to send real-time updates to students when classes are cancelled.

4.3 Database Schema Design

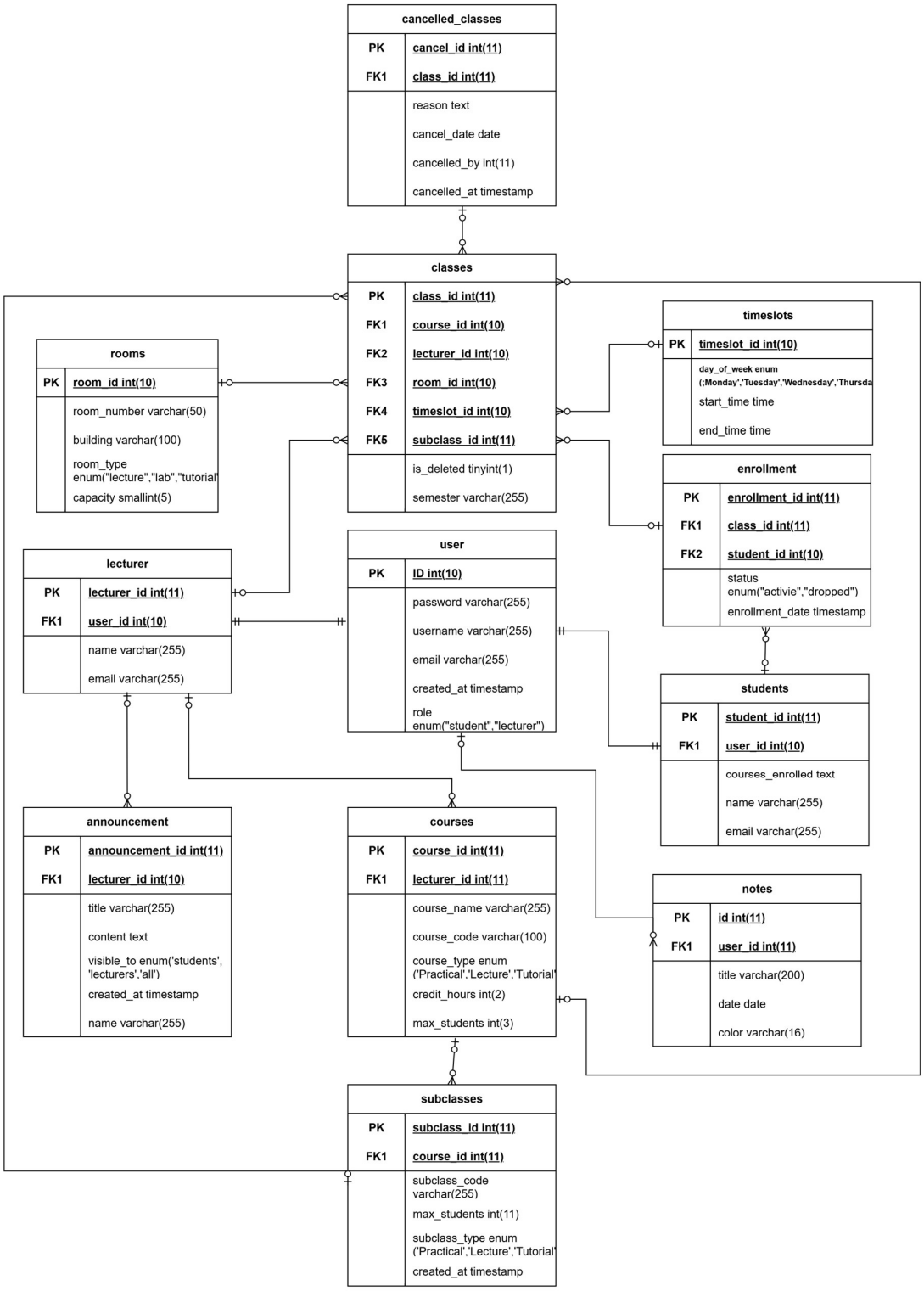


Figure 4.3: ERD Diagram

Figure 4.3 above shows the database of the timetable system website development. There are 11 tables created in the database. The first table is the user table for storing the user data and assigning roles to users. The second table is a student table for storing student information like name, courses enrolled, and email. It is used to retrieve data from users when registering as a student. The third table is the lecturers' table for storing the information of the lecturers with name, email, and user_id as a foreign key. It retrieves data from the users table when registering as a lecturer. The room table is used to store the room information, like room number and capacity, to ensure no more than the allowed number of users use the room. Besides that, the classes table records the class assignment by Admin. Then, the table of courses and timeslots stores information of courses and all the available timeslots from 8am to 6pm. The table notes are created to record the note in the student calendar for convenient usage. Furthermore, the table subclasses store the class information with subclasses. For example, one lecture with few practical or tutorial classes. The announcement table records the announcement published by the lecturer. It is used to store the slots available for specific courses. The last table is the enrollment, which stores data on the courses that are taken by students.

4.4 System Components Interaction Operation

This section explains the role interactions in integrated timetable system. The relationship between these 3 roles. How one of these roles functionalities influence another role.

- **Student**
    1. Register course without conflict
    2. View calendar
    3. Check Announcement
    4. View Timetable
- **Lecturer**
    1. Cancel a class
    2. Publish Announcement
    3. View schedule
- **Admin**
    1. Manage user
    2. Manage course

3. Manage room
4. Manage class assignment
5. Check resource utilization rate in graph view

Table 4.4

Role	**Action**	**System Component Involved**	**Output**
Student	Register for a course	Frontend (Course Registration Page) + Database	Course added to timetable and calander
Lecturer	Cancel a class	Frontend (Dashboard) + Database	Real time updates in student calendar and announcement section.
Admin	Allocate classroom resources	Frontend (Admin Module) + Database	Resource allocation updated successfully or failed if conflict occur

Table 4.4 shows interaction between the roles. When students register for a course, the courses are added to the timetable and personal calendar. The database will be updated. When a student drags and drops the class to another available class in the timetable, the database will be updated also. Students should make sure that the semester intake registered has courses to choose from; otherwise, there is no course to register for. The classes' semester settings are all based on the admin dashboard; for example, if admin assigns a class to the semester of June 2025. The student account that registers for the semester for June 2025 only can register for the courses. Besides that, when a lecturer cancels a class, he/she needs to make sure that there is a class on the date; it will fail to cancel if there is no class. The database will be updated, and data willto student announcements and the calendar. Students can see the real-time updates in their own dashboard. Also, students will get emails for cancellations. Lastly, when admin manages a room, make sure there is no duplicate room number. When managing the user, the system is not allowed to change a username or password for the admin account.

Chapter 5

System Implementation

5.1 Hardware Setup

The hardware used in this project is a computer. The whole system is being done on this computer, including testing and coding.

Table 5.1: Specifications of laptop

Description	Specifications
Model	HP Pavilion Laptop 15-eg2xxx
Processor	12th Gen Intel® Core™ i5-1240P
Operating System	Microsoft Windows 11 Home Single Language
Memory	32GB RAM
Storage	952GB Disk Drive

5.2 Software Setup

Table 5.2 : Software requirements

Integrated Development Environment (IDE)	Visual Studio Code
Front-end language	Html, CSS, JavaScript
Back-end language	PHP
Database Management System (DBMS)	XAMPP

Based on table 4.1 above, there are some requirements I need to set up first before starting the project. Visual Studio Code initialized as a text editor at the beginning and then became the IDE of this project by installing extensions like debugging and code completion. It is useful for this project to do manipulation for the code. Then, the front-end languages that can decorate and design the front end of the website are HTML, CSS, and JavaScript. At the same time, PHP is used as a back-end language to access the XAMPP database; it is able to control and manipulate the data in the database. Lastly, XAMPP becomes the DBMS in this project, aiming to retrieve data from the database to let the project have more possibilities to do specific tasks.

5.3 Setting and Configuration

In this system, the setting and configuration are quite important. For the DBMS, XAMPP was used to set up a local server. Both Apache and MySQL services must be started before running the system. PhpMyAdmin is used to create the project database and the needed table. To connect the database with the front end, the backend PHP files must be set with the correct host, username, password, and database name by using the PHP language.

The frontend languages used to develop the integrated timetable system were HTML, CSS, and JavaScript. To design and implement more dynamic functions, JavaScript and jQuery were used to interact with the database through PHP scripts. It allows retrieving or updating the database. During development, make sure every file is connecting to the database.

The backend PHP scripts are responsible for data validation, handling, and CRUD in the database. During development, make sure the configuration with the API is correct. Finally, necessary settings are a must to start this project. Frontend and backend are both important things to develop the integrated timetable system.

5.4 System Operation

This part will describe the system operation from configuration to functioning well.

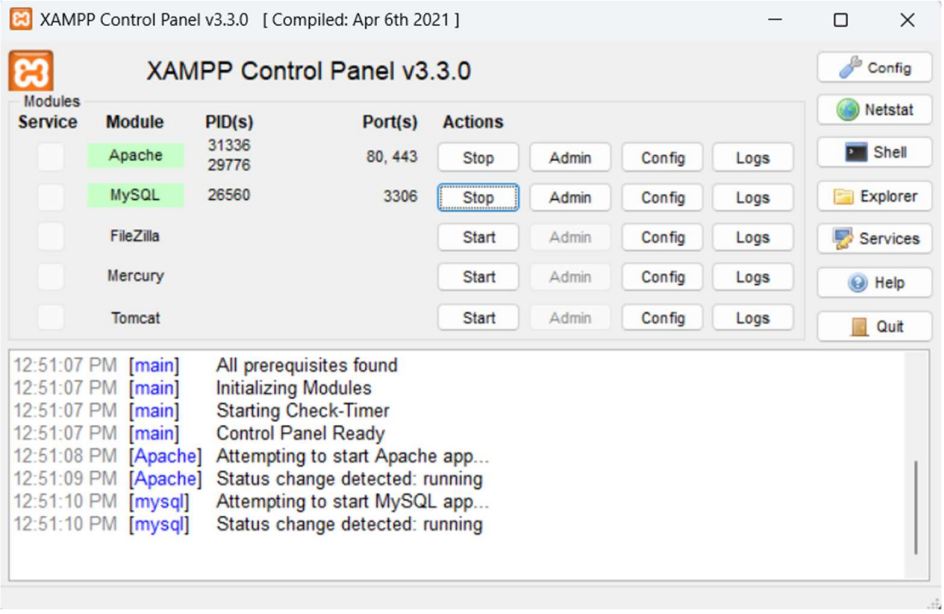


Figure 5.4.1 XAMPP Launcher

Start the Apache service and MySQL service.

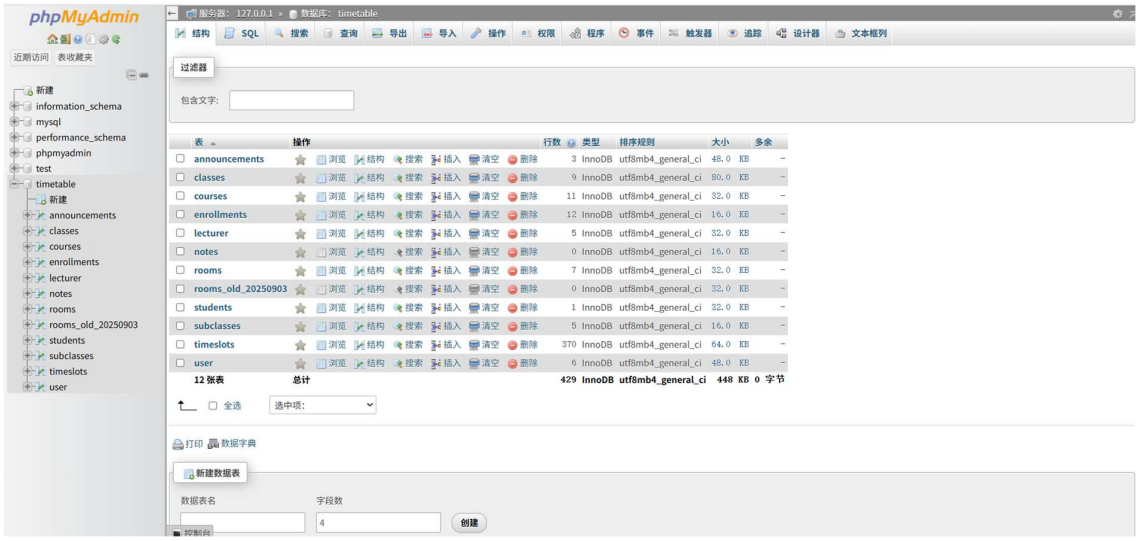


Figure 5.4.2 Backend Setup

Build database and tables, required to work together with front-end

DB.php

C: > xampp > htdocs > FYP1 > DB.php

```php
<?php
// Start the session (if needed in other pages)


// Database connection variables
$servername = "localhost"; // or your database server IP/hostname
$username = "root"; // Your database username
$password = ""; // Your database password
$dbname = "timetable"; // Name of your database

// Create connection
$con = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($con->connect_error) {
    die("Connection failed: " . $con->connect_error);
}

// Optionally: For better error handling, you can set this:
// mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);

?>
```

Figure 5.4.3 Database Connection

Set up the configuration in PHP files before running the system on website.

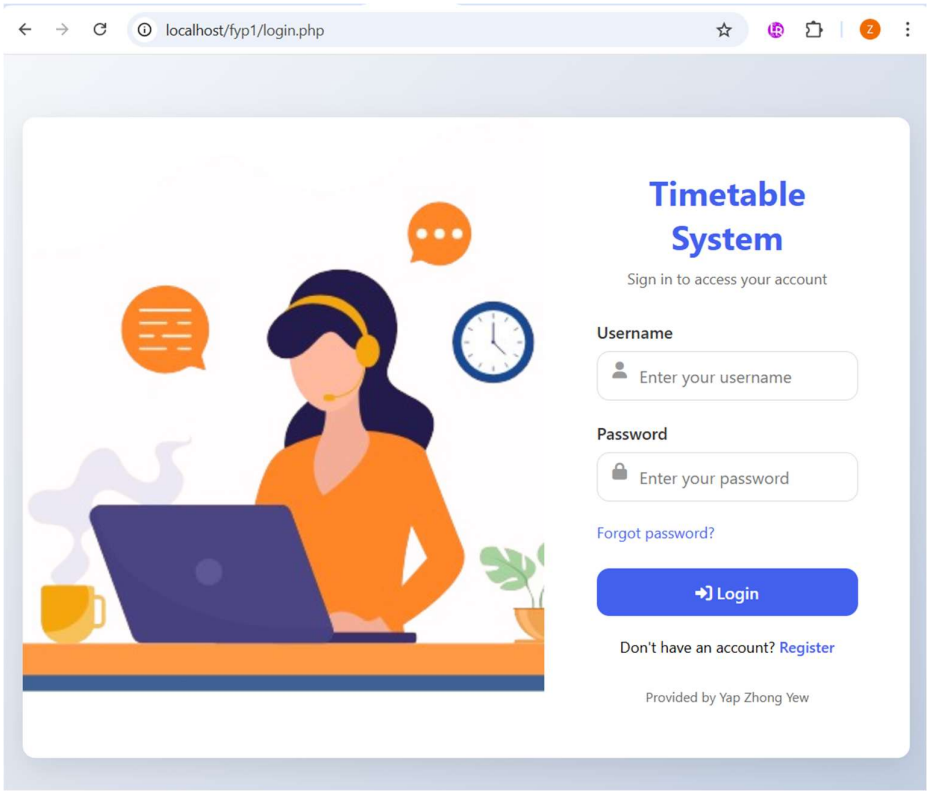


Figure 5.4.4 Login Page

Try to open the localhost website to check whether connected or not.

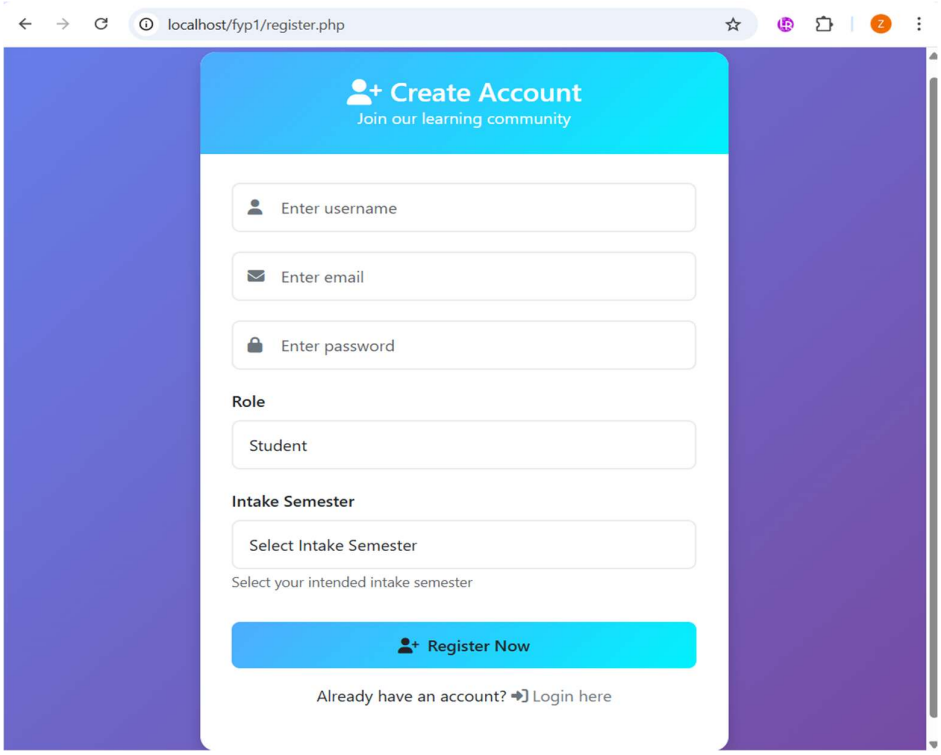


Figure 5.4.5 Register Page

Register a user account with a role.

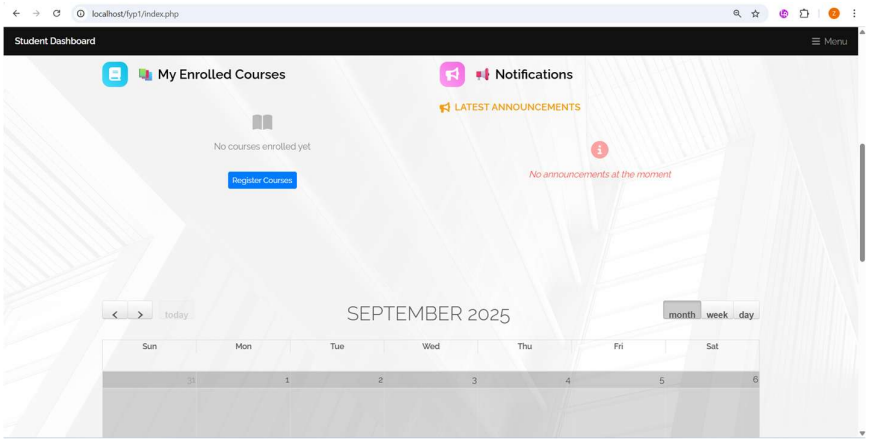


Figure 5.4.6 Student Dashboard

Student can login to student dashboard, able to view enrolled courses, notifications and calendar.

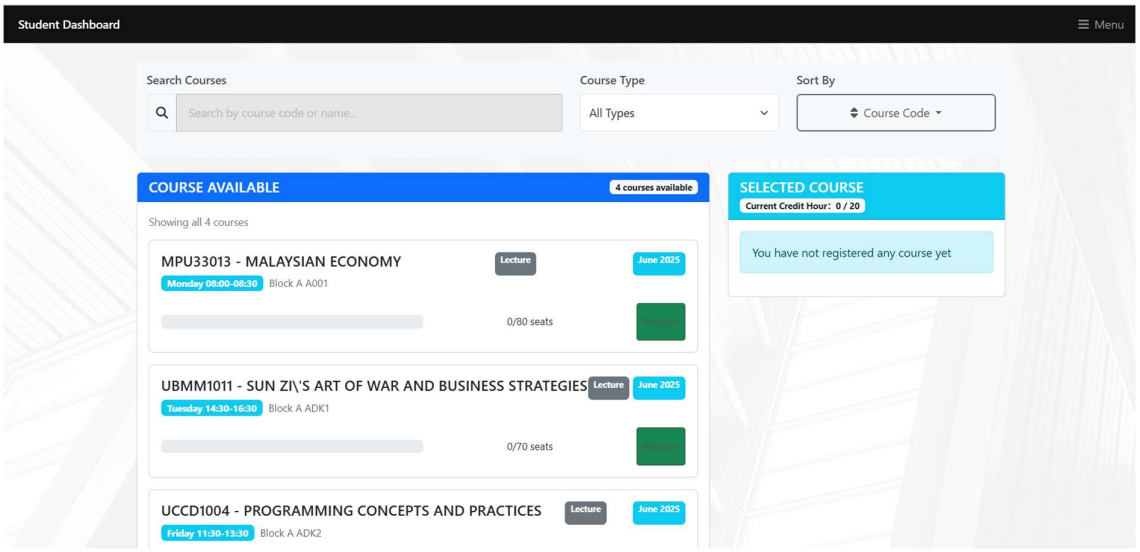


Figure 5.4.7 Course Registration

User can register courses without conflict.

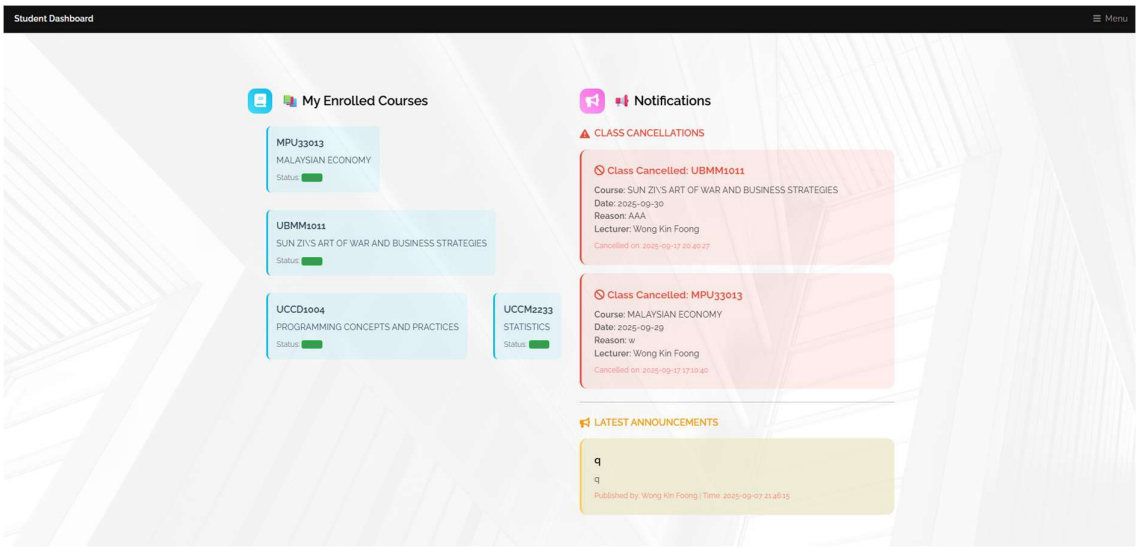


Figure 5.4.8 Information on dashboard

User is able to see enrolled courses and announcements.

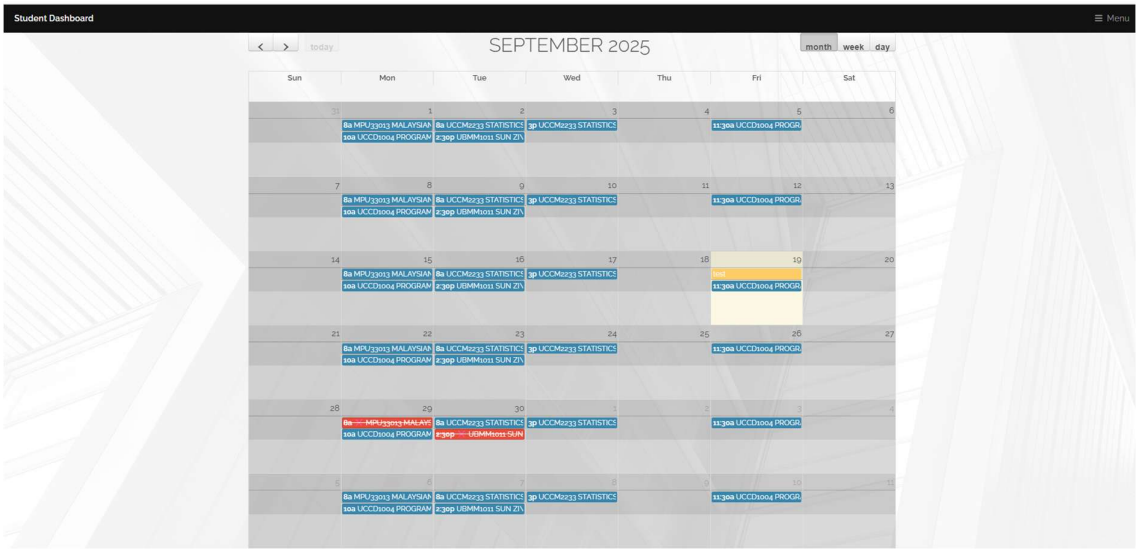


Figure 5.4.9 Calendar

User is able to view personal calendar, add note on calendar, view the real-time updates of class cancellations.

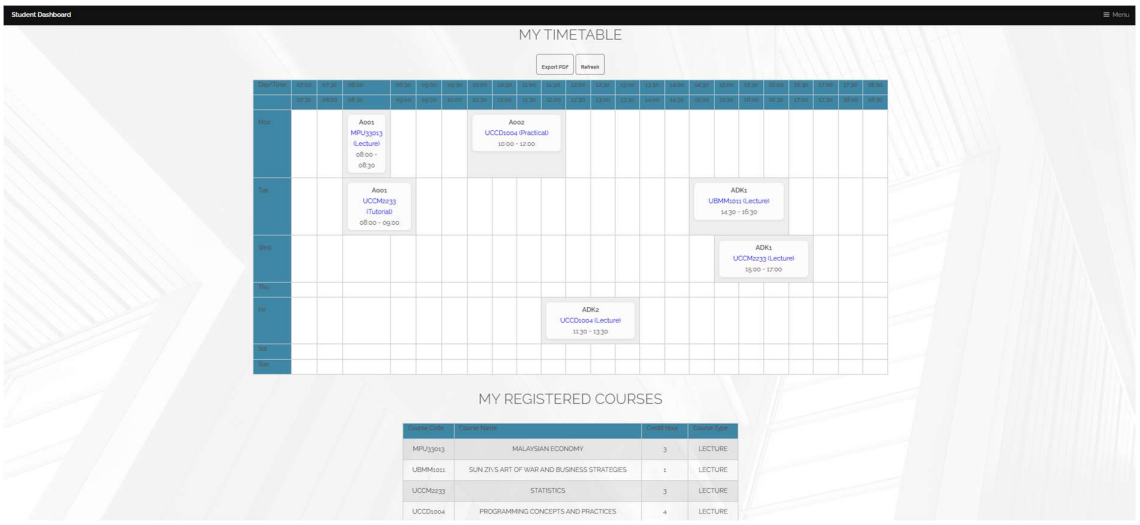


Figure 5.4.10 Timetable

User can see own timetable, drag and drop feature is implemented to drag class to another available class.

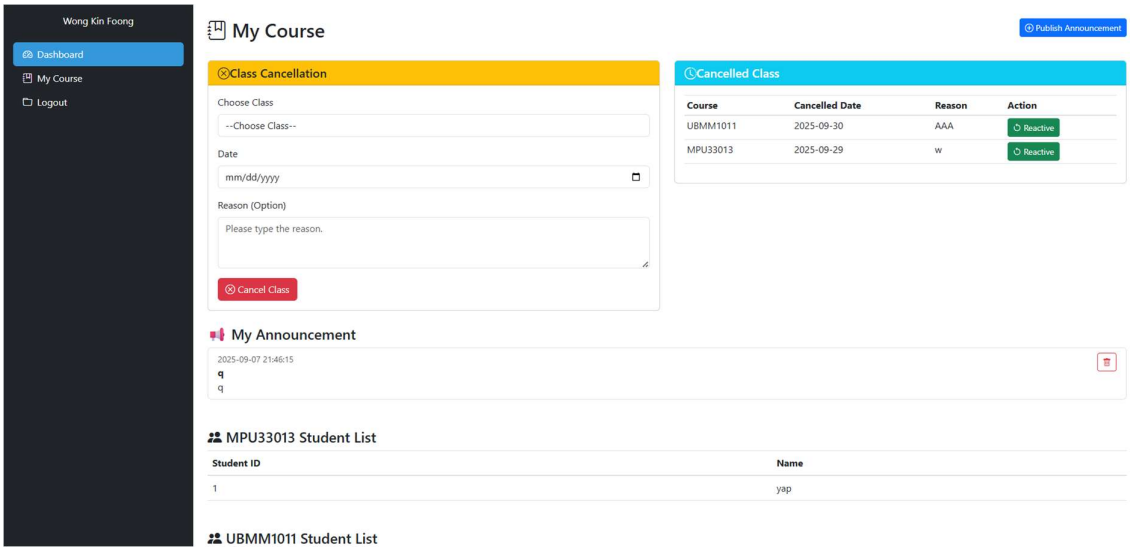


Figure 5.4.11 Lecturer Dashboard

Lecturer can cancel the class on correct date, publish announcements for own student can view of their name.

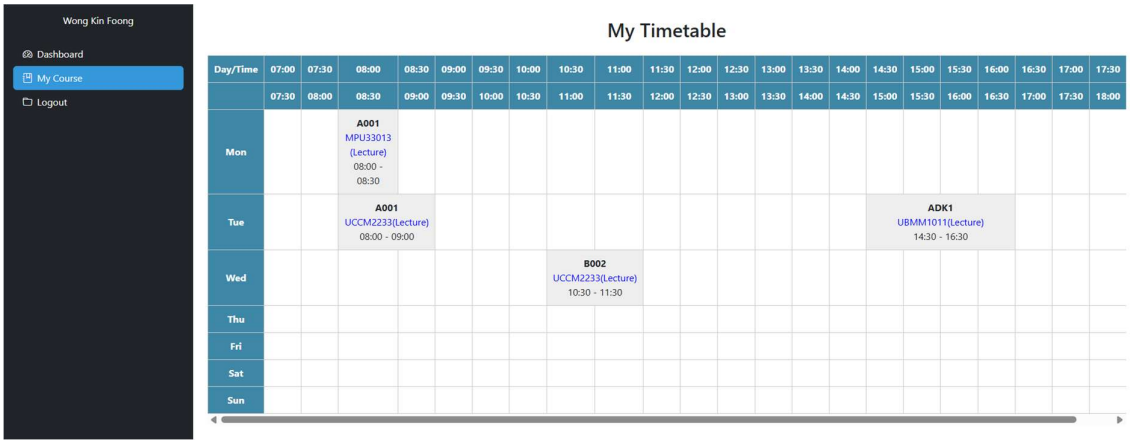


Figure 5.4.12 Lecture Schedule

Lectures can see own schedule easily

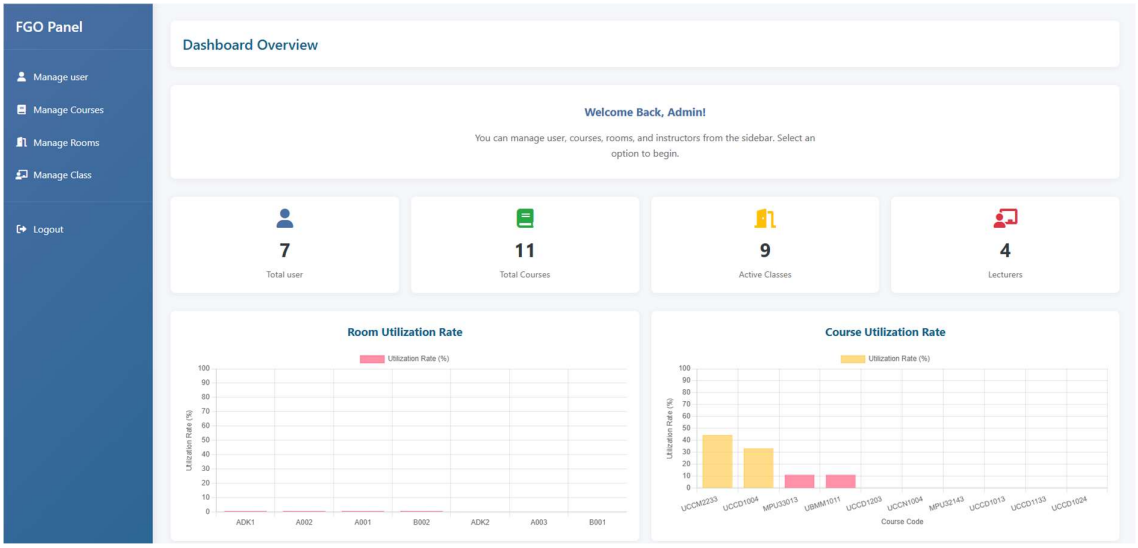
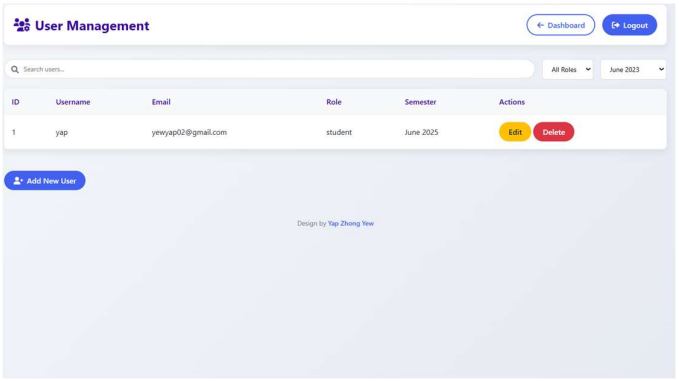


Figure 5.4.13 Admin Dashboard

Admin can view the resource utilization rate in graph.



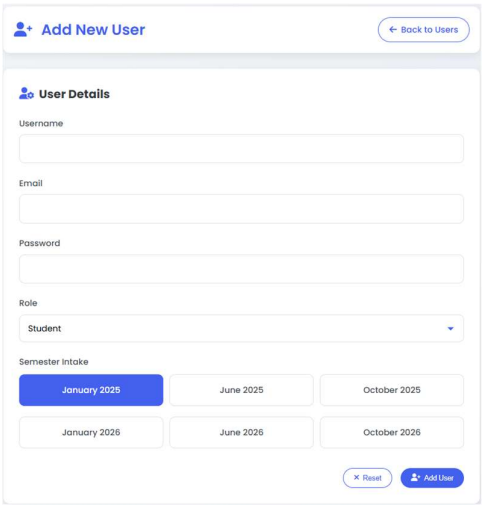
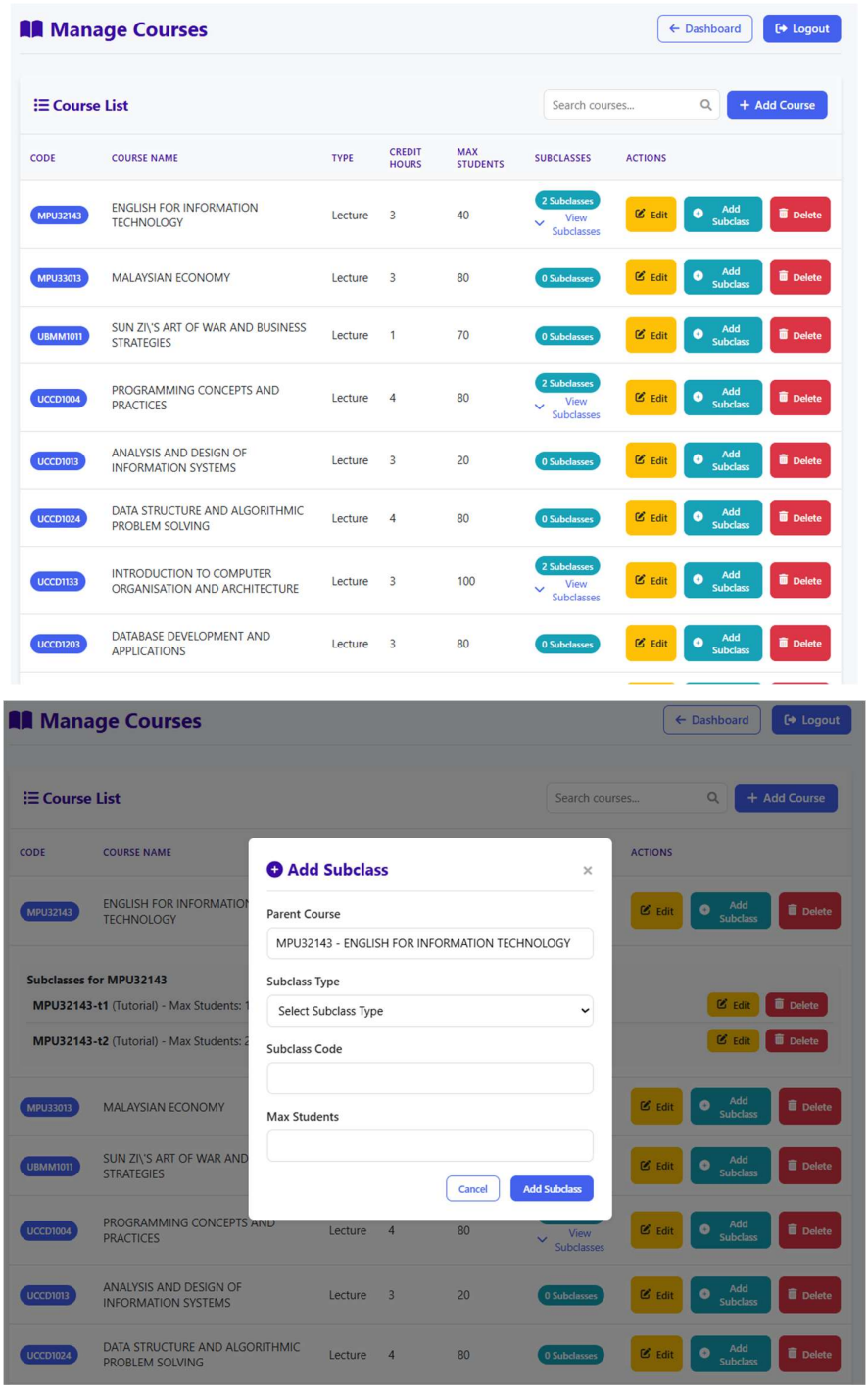


Figure 5.4.15 Manage user

Admin can search for user account by roles or semester. CRUD can be done here



5.4.15 Manage Course

Admin can manage the course and add subclass for course.

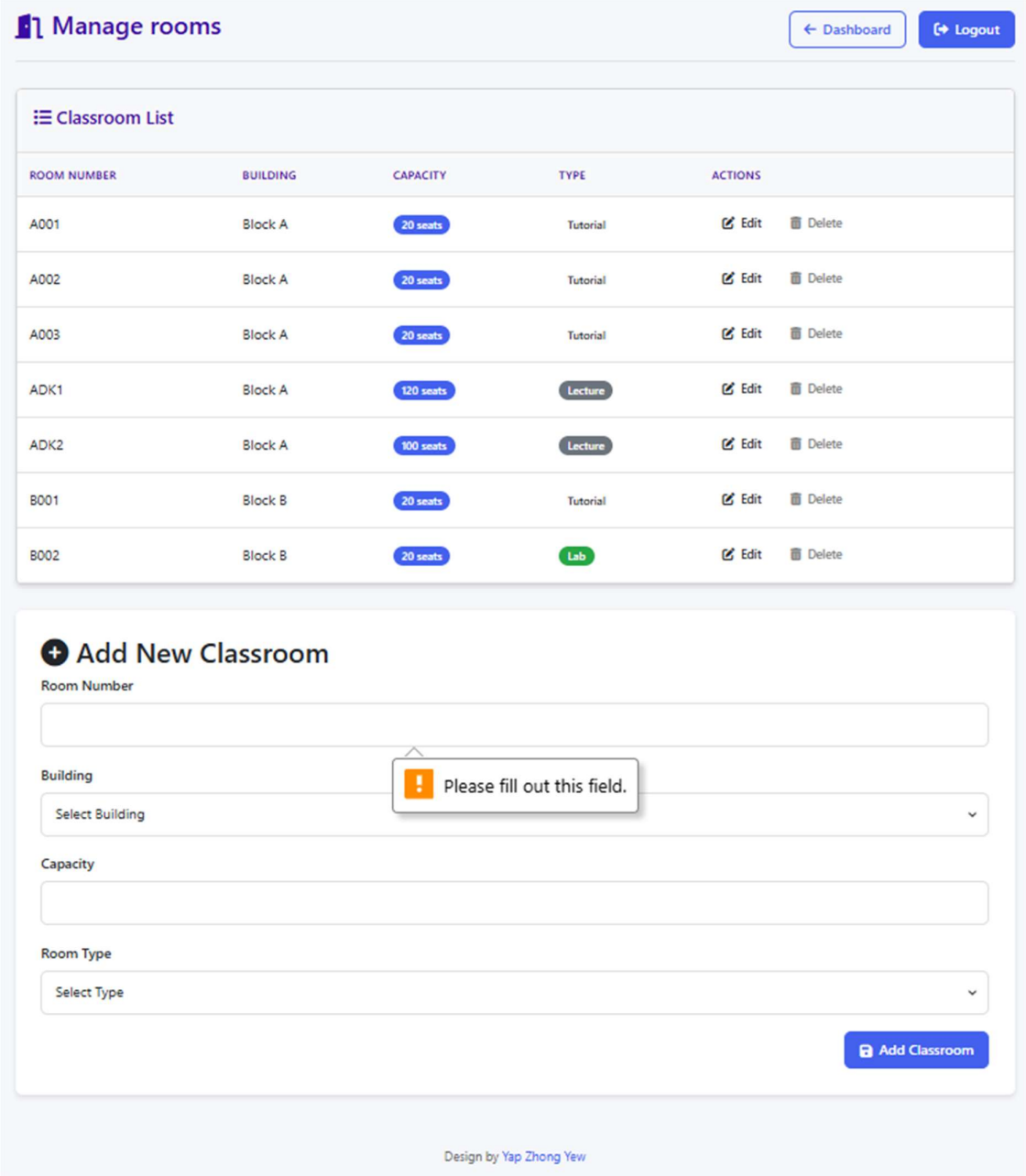


Figure 5.4.16 Manage room

Admin can manage room

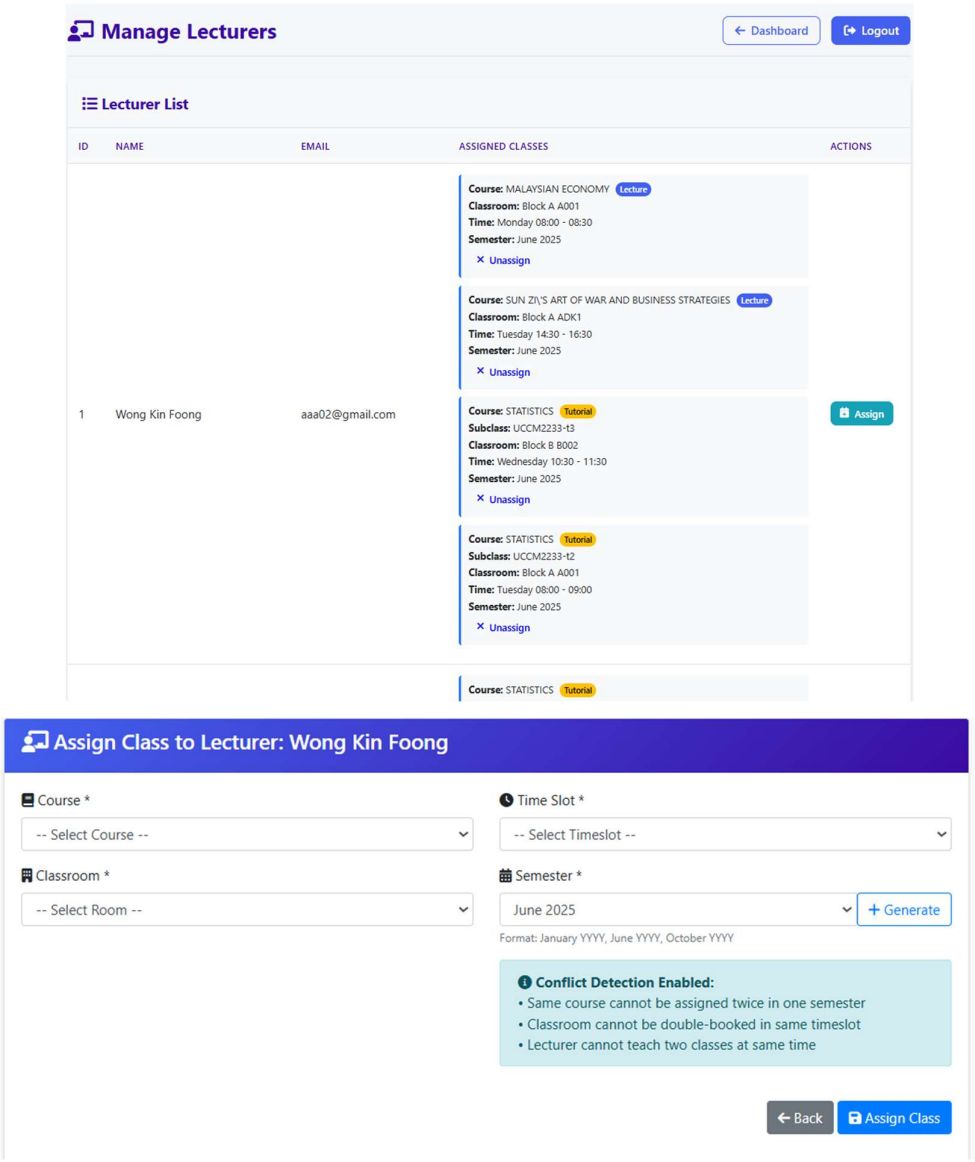


Figure 5.4.17 Manage Class

Admin can assign class to specific lecturer without conflict.

5.5 Implementation Issues and Challenges

During the development and deployment of the integrated timetable system, there are several issues and challenges appearing.

For the student site, displaying timetable is the hardest part among all features in student. The system needs to retrieve data from multiple tables to get the information. For example, course information, time information, class information, and enrollment records need to cooperate with each other. Thus, the timetable can only be displayed well. During the debugging, there were many bugs and errors occurring. This is the most challenging part in the student site.

On the lecturer site, publishing announcements to specific class students is also one of the challenges. To ensure other students will not see the unrelated announcements, the SQL query must be written correctly. Besides that, the class cancellation is also one of the challenges, cancelling class on a specific date, while the real-time updates need to apply to the student calendar.

For the admin site, managing resources is the issue. It is because I have never been a school administrator before; I do not fully understand the school operation flow. I need to keep fixing the error until there is almost zero error. In this way, I have implemented the resource utilization rate in graph view. Lastly, design is also an issue. To make sure the design is clean and intuitive, I change the UI design a few times while maintaining the modern style.

5.6 Concluding Remark

In conclusion, the integrated timetable system is successfully developed with full functionality and a user-friendly UI design. By deploying this system, the initial parameter must be configured correctly. Start the Apache and MySQL services and make sure every file is connected with PHP files. Adding the dynamic function while maintaining data integrity and usability can enhance the integrated timetable system performance.

Chapter 6

System Evaluation And Discussion

6.1 System Testing

This chapter is focused on describing the testing setup and result. Trying to evaluate the system and discussing whether that archives the project objectives, concluding the project challenges as well.

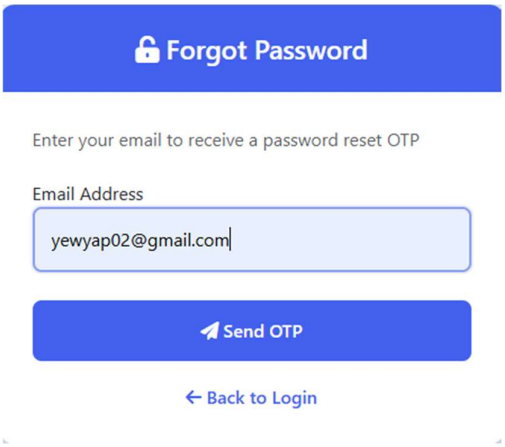


Figure 6.1.1 Test Case 1

When I need to renew a password due to a forgotten password, I have to input my email and receive OTP code. The code is successfully sent by email and only valid in 5 minutes. Thus, it is preferred to register an account with a real email address.

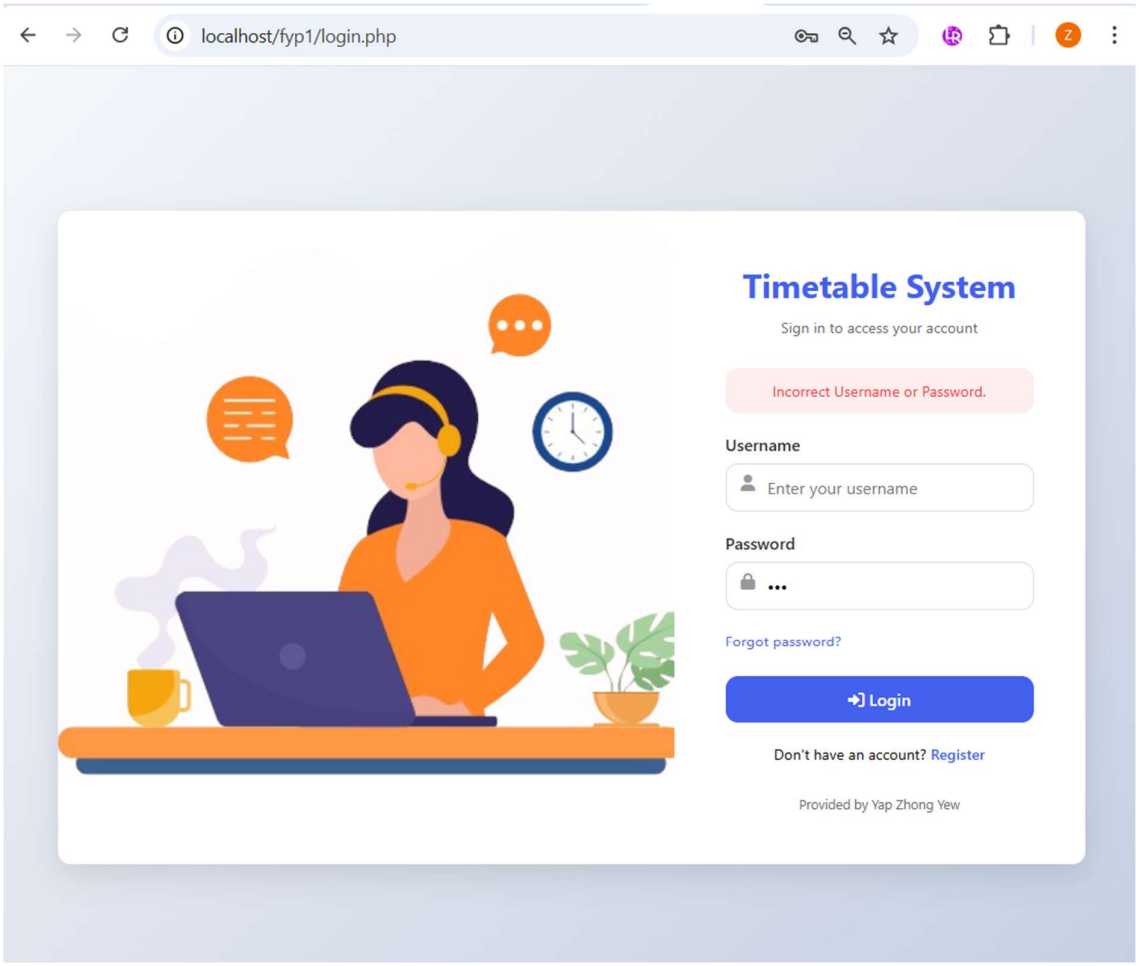


Figure 6.1.2 Test Case 2

When the username or password is incorrect, the error message shown is “Incorrect Username or Password.”

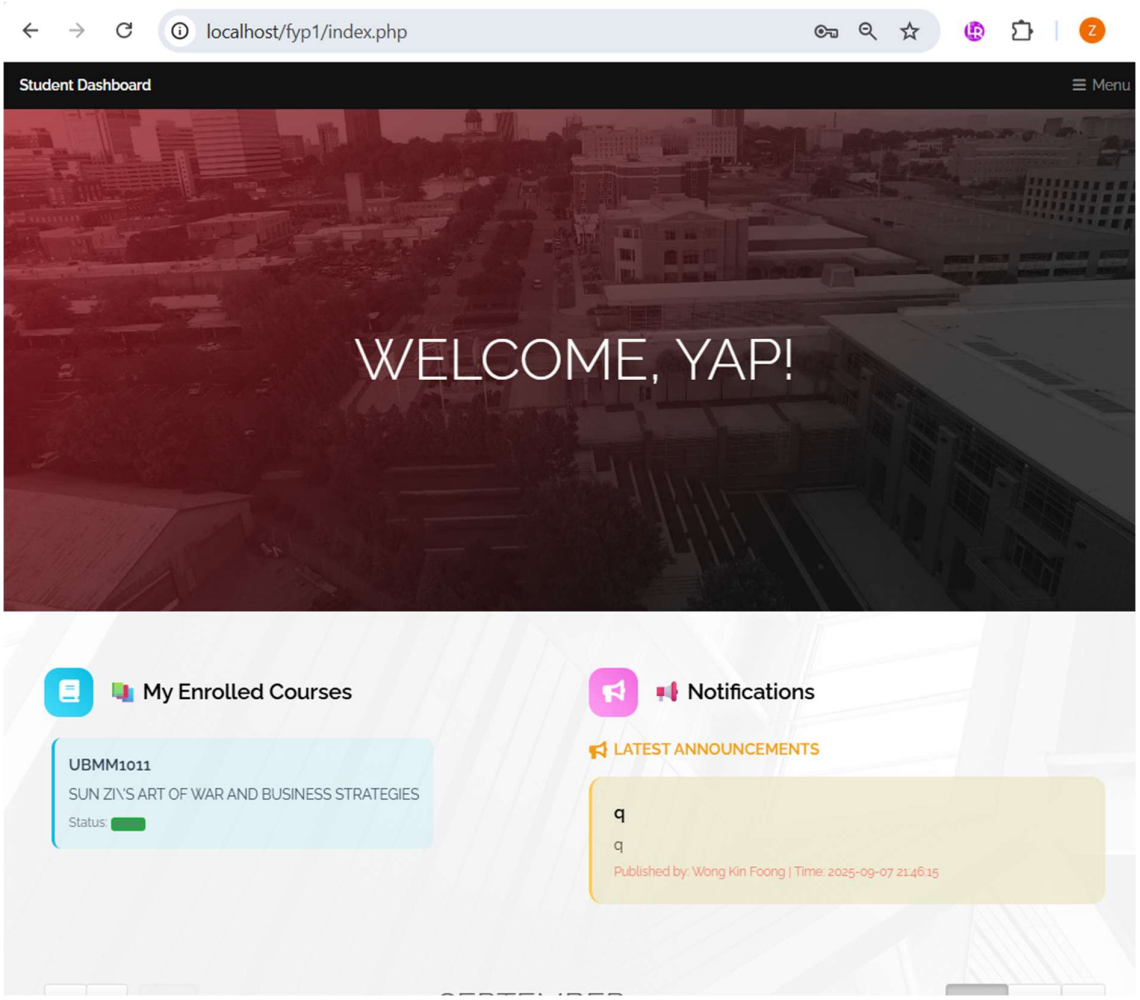


Figure 6.1.3 Test Case 3

User will be navigated to the related dashboard with a correct username and password.

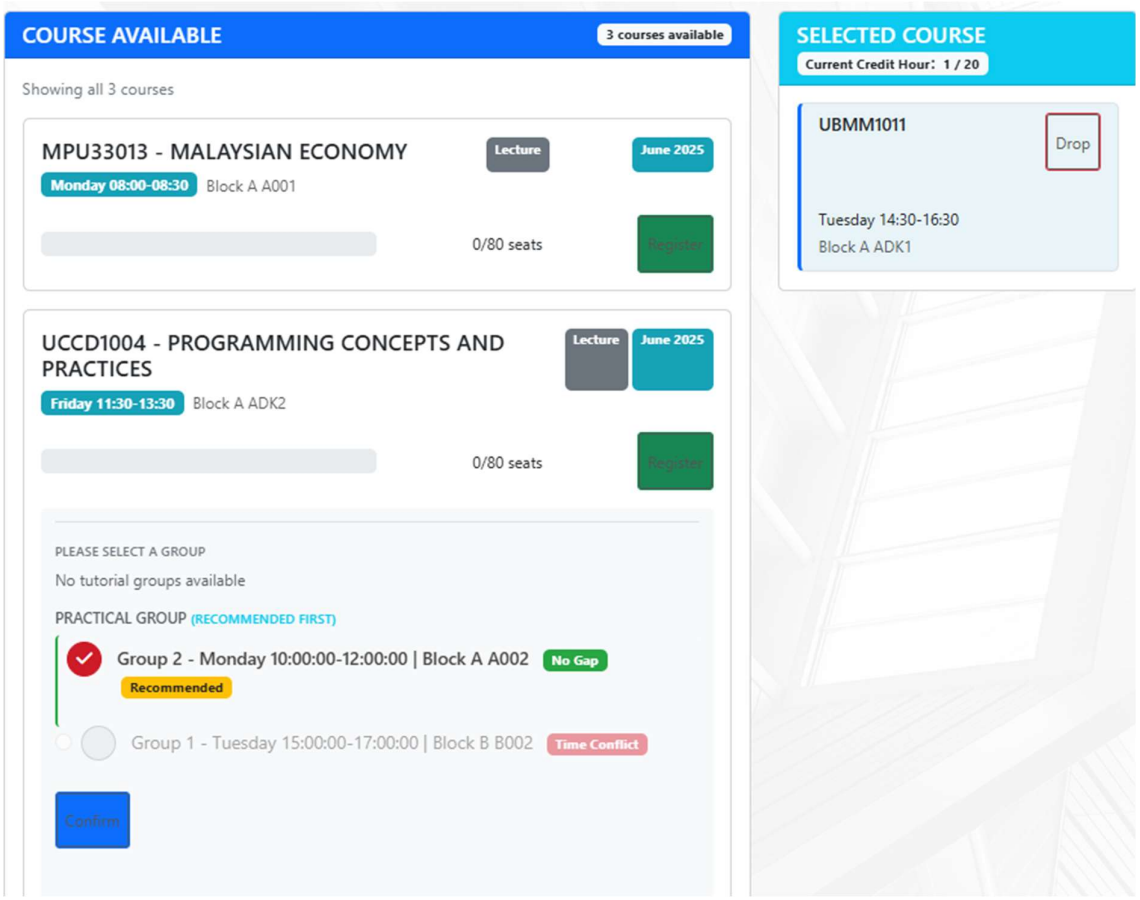


Figure 6.1.4 Test Case 4

The user is not able to register a class with overlapping time. Time conflicts will be displayed on every class that is unable to register.

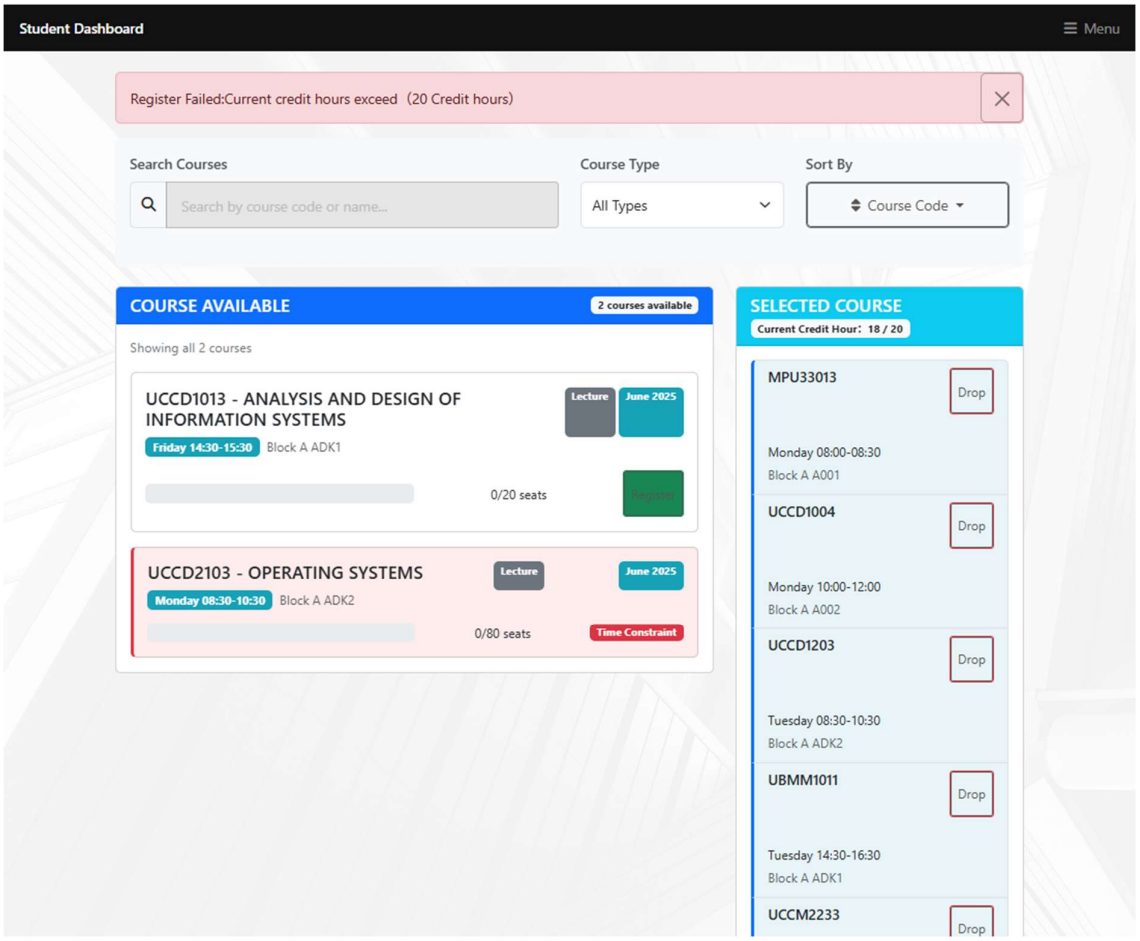


Figure 6.1.5 Test Case 5

System is not allow user to continue register course if the credit hours is exceed 20.

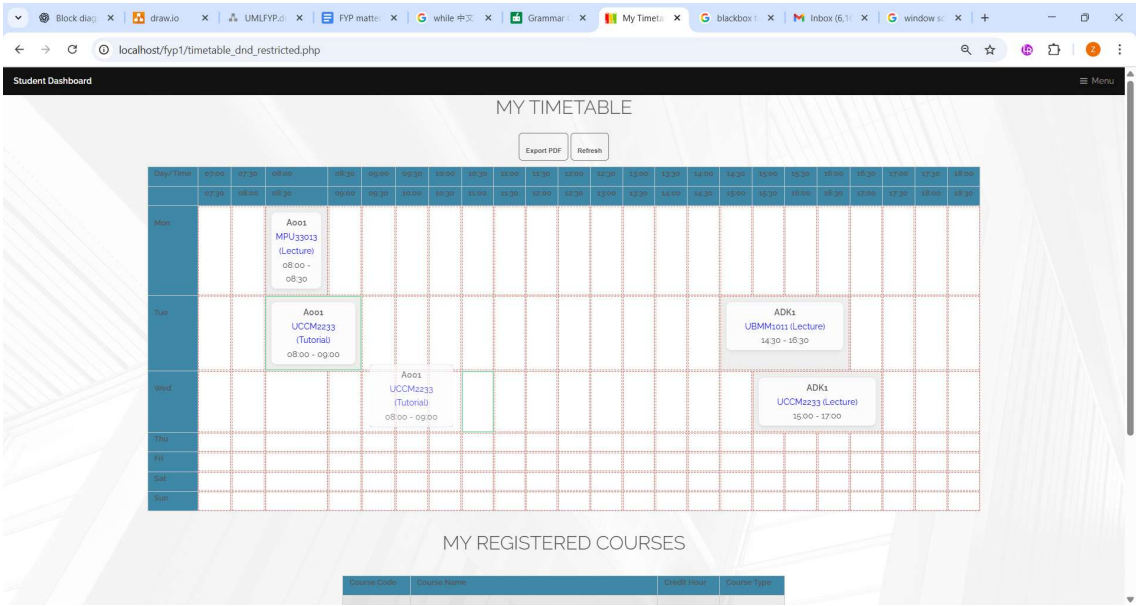


Figure 6.1.6 Test Case 6

The timetable can be viewed after registering for courses. When I drag the class, the slot shows a red color, meaning there is no such timeslot for the class, or a conflict happens. For the green color, it allows dropping there to change the timeslot of the registered course.

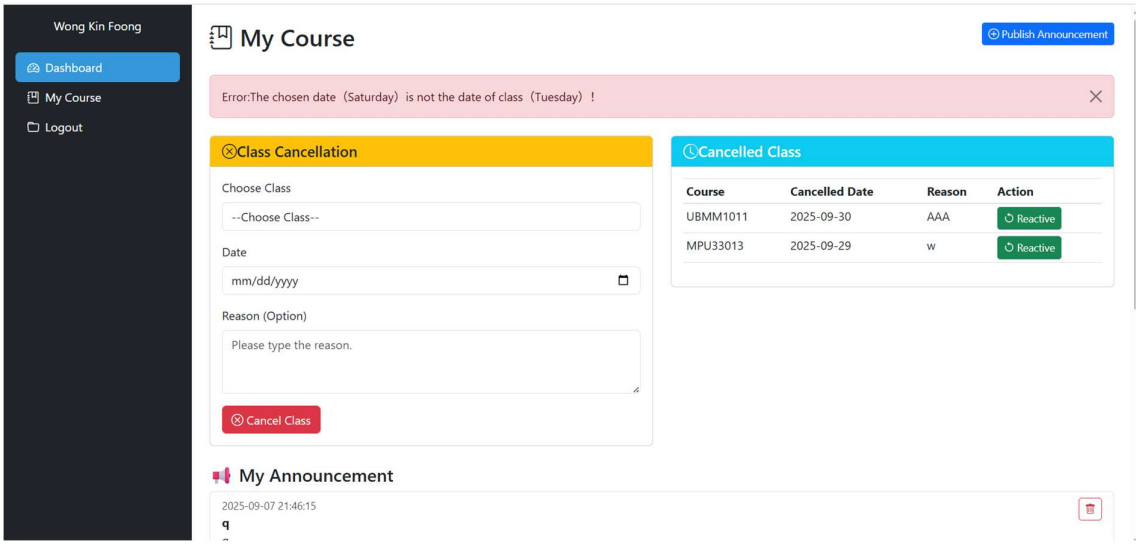


Figure 6.1.7 Test Case 7

The error message showed that the chosen date is not the date of class. The system will check whether the date is correct.

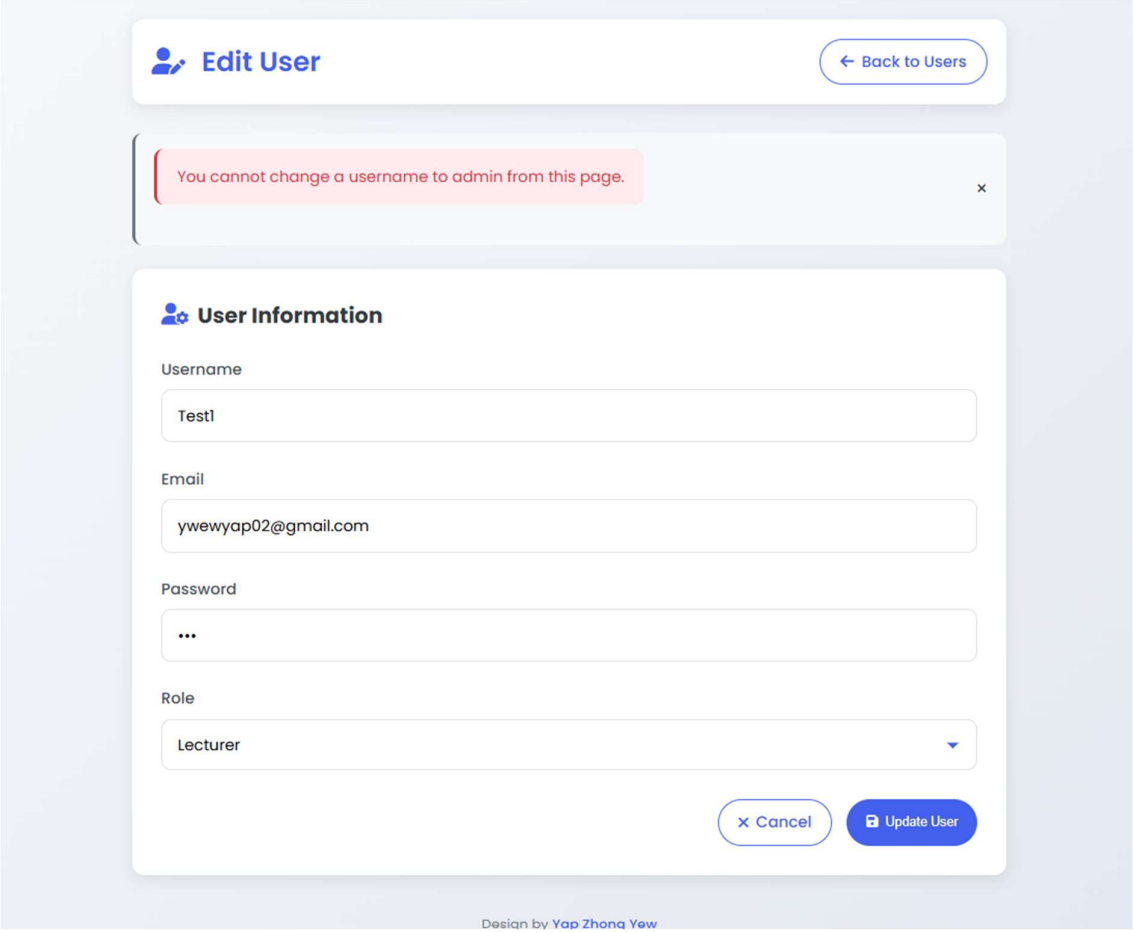


Figure 6.1.8 Test Case 8

System is not allowed to change an account username or password to admin.

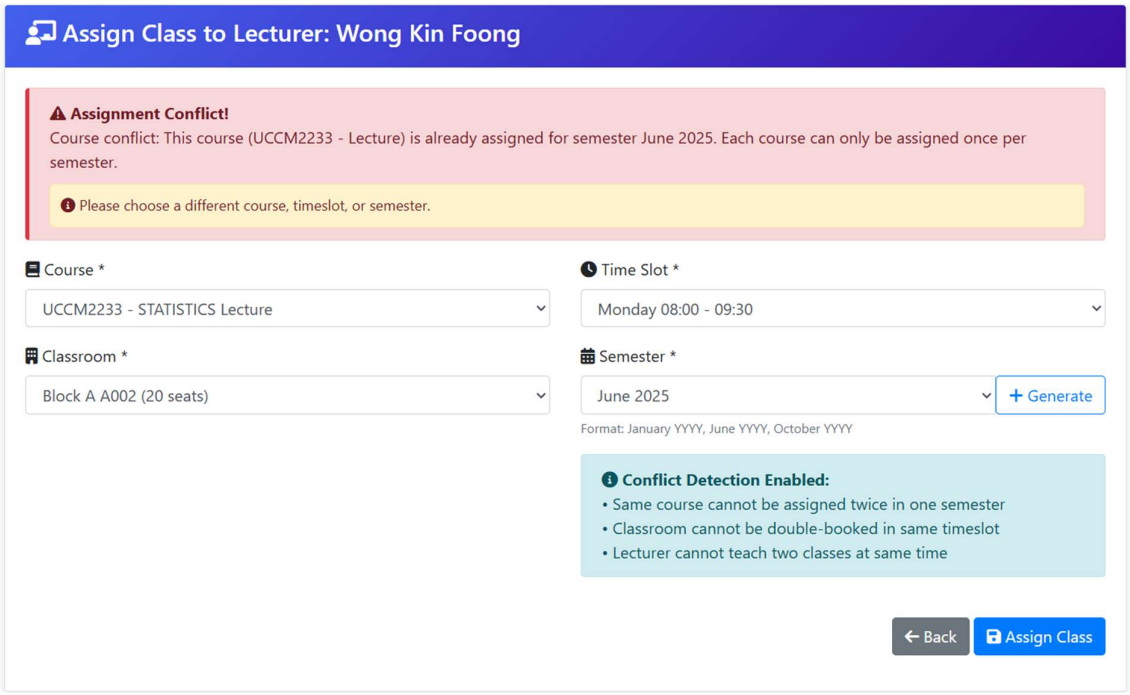


Figure 6.1.9 Test Case 9

Admin cannot assign a class to a lecturer who has a course, timeslot, or other conflict.

6.2 Testing Setup

The system testing was conducted on a laptop with 32 GB RAM using XAMPP, with Apache and MySQL services running on Windows 11. The integrated development environment to do coding was in Visual Studio Code. The backend language was PHP and was used in connecting from the database to the presentation layer. The frontend languages that were used included HTML, CSS, and JavaScript. The tests were performed on the Google Chrome and Mozilla Firefox web browsers to ensure that multiple web browsers can be used as well.

The testing stage is separated into three roles: admin, lecturer, and student. The different functions for each role are tested one by one to ensure reducing bugs and errors as much as possible. The three roles have mutual influence to work well. For students, there are several tests, such as course registration, drag-and-drop timetable, and real-time updates on personal calendars. For lecturers, the system allows them to cancel class, publish announcements, and view the class timetable and student names. Lastly, admin tests focused on allocating resources in a good manner and being able to change account usernames and passwords and manage courses, rooms, and assignment classrooms.

6.2.1 Testing Result

Table 6.2.1 Test Case

No	Test Case	Expected Output	Actual Output	Remark
1	Forgot password and hope to reset a new password with OTP code through email	User can reset password	User can get an OTP code in email, continue to reset the password	PASS
2	Login with invalid password or username	Login failed	Error message shown: "Incorrect Username or Password."	PASS
3	Login with correct username and password	Login successfully	Login successfully and navigate to each dashboard based on own role	PASS
4	Register courses with same timeslot	Register failed	There is message shown: "Time Conflict" and unable to choose the time	PASS
5	Register courses exceed credit hour	Register failed	There is error message and unable to register	PASS
6	Drag-and-drop timetable	Change class	Can drag the class to another timeslot and change the registered class	PASS
7	Cancel class on an invalid date	Cancel failed	Unable to cancel class	PASS
8	Cancel class on a valid date	Cancel successfully and recorded on both lecturer and student dashboard	The cancellation message display on student dashboard	PASS

			and show on the personal calendar	
9	Publish Announcement	Published successfully and recorded on both lecturer and student dashboard	The announcement posted by lecturer will display on the student dashboard	PASS
10	Change username or password to admin	Unable to change	Error message shown and unable to change username or password to admin	PASS
11	Assign a classroom with same timeslot	Unable to assign	Error message shown: "Assignment Conflict"	PASS

Based on table 6.1, I have done several test cases to make sure of the quality of my system. I am ensuring my system functions are well aligned with the project objectives. The first test case I did was reset a password. I tried to input my email, and then it sent an email to me. I input the OTP from the email in the form; it allowed me to reset the password. The second test case is about logging in with an incorrect username or password. It is unable to log in; at the same time, I input a correct username and password. It navigates me to the student dashboard, as my role setting is student. I have tried the lecturer role and the admin role; it navigates me to each dashboard.

Moreover, I try to register courses with the same timeslot. The time is conflicting with my previously registered class, and a display message showed that time conflict. Besides that, a test case with registering courses exceeding 20 credit hours will be unable to register. When the class is fully registered, the system does not allow me to register. After that, students can view their own timetable. There is a drag-and-drop function implemented in the timetable that allows students to drag the course to another available timeslot. It is convenient for students to do comparisons.

Furthermore, I have done some test cases in the lecturer role. The lecturer can cancel his/her class on the class date; if an invalid date is chosen, it is unable to cancel. After cancelling, the personal calendar in the student dashboard will update in real-time. Showing a red mark on the class. Besides that, a test case that publishes an announcement also can provide updates on the student dashboard. As an admin, to prevent unauthorized access, the account username and password cannot update to "admin." Assigning a classroom also will show an error when assigning a classroom to the same timeslot.

6.3 Project Challenges

During the development of the integrated timetable system, I faced several challenges. These challenges help me to improve a lot in decision-making and problem-solving skills.

Firstly, the challenge I faced in this project was the UI design. I change many times of my UI coding to ensure users can easily use initiative while maintaining the modern design. The button and others should be obvious to see. The function feedback also must respond in a second and in an obvious way, like changing or adding something to the presentation layer.

Besides that, one of the challenges was the dynamic interaction. The coding skill must be strong to do a dynamic timetable. To maintain the system run, I need to combine both backend and frontend coding to make sure they can be utilized together. To prevent many conflicts, there was much logical coding to make sure it is fine to use. For example, a classroom cannot be used at the same time.

Furthermore, another challenge was the API implementation. I use a Gmail API to send an OTP code to a real Gmail. During testing, I use a real Gmail to make sure it can receive the OTP. In order to reset the account password. Although the API looks easy, I spend a lot of time configuring the settings to make sure everything is normal and works.

Lastly, the challenge was role interactions in the system. Since the system has three user roles—administrators, students, and lecturers—it was very important in brainstorming related to the relationship between these three roles. For example, a lecturer can cancel class on a date,

and then back on the student dashboard, there will be an announcement related to the class cancellation, and the calendar will be displaying the cancelled class.

6.4 Objectives Evaluation

The integrated timetable system was developed aligned with the three objectives from the start. The objectives were created to complete the system and satisfy the needs of users. Each objective was evaluated based on the testing results.

1. **Define the distinct functions for each role**

   Evaluation: Fully achieved. There were three roles in my system; the functions were defined and developed for each role. Students are able to view announcements, the calendar, and the timetable and do course registration. The lecturer is able to cancel class and reactivate class, publish announcements, and view their own schedule and student names. Admin is able to manage resources.

2. **Develop the dynamic timetabling**

   Evaluation: Fully achieved. Students can view their own timetable, and it is able to be printed as a PDF. The drag-and-drop feature is implemented in the timetable. Students can drag the class to another available time of class without conflict.

3. **Design a personal calendar**

   Evaluation: Fully achieved. Students have their own calendars and are able to view the class every day. The calendar has the benefit that the class view is obvious. There are real-time updates if class is cancelled. Notes can be added on the calendar.

4. **Develop a transparency resource utilization view**

   Evaluation: Fully achieved. The system will show a dynamic graph on the admin dashboard. Admin can check the resource allocation directly. The data for room and course utilization is displayed.

Overall, the integrated timetable system successfully archived the primary objectives. Based on the testing results, the minor errors or bugs may appear, but they can be fixed in future enhancements.

6.5 Concluding Remark

In conclusion, the evaluation of the integrated timetable system showed that it successfully fulfills its primary objectives for students, lecturers, and administrators. The functional testing is done to ensure that all key functions were working well as expected. Performance metrics showing the information validation and error-handling mechanisms effectively maintain data integrity.

Overall, the system provides a stable, functional, and user-friendly website for users to do specific actions in a good manner.

Chapter 7

Conclusion and Recommendation

7.1 Conclusion

In the new era of globalization, traditional timetabling is not a convenient way to schedule; people may need a quick and convenient way to replace the old method. This project makes it possible to resolve problems of course scheduling and resource allocation in the context of academic classes at schools, colleges, and universities. This system is anticipated to enable students to register for courses subject to the existing course capacity, view their tailored timetables, and check if their class is cancelled on the calendar. Announcements and enrolled courses can be checked in the dashboard. This system also shows the resource utilization rates in a graph. Admin is able to view and allocate the resources well.

7.2 Recommendation

Based on my current project, I have done development, evaluation, and testing. I have successfully fulfilled the primary objectives in satisfying the needs of users. There are several recommendations to improve my work in the future.

For system improvement, mobile support can be added to the integrated timetable system. To allow the users to use it at any time after downloading. To archive it, making it mobile responsive is important to do. Besides that, I recommend applying AI to sort the class automatically. In the future, I believe that most of the software is applying AI to assist in functioning. AI significantly help human improve the user experience in many areas.

Furthermore, the system can apply a real-time chat system instead of only e-mailing. To allow lecturers, students, and admin to communicate directly, the system can improve the quality and effectiveness of important announcements. A real-time chat system should include an AI chatbot and 24-hour customer service; it benefits students if any emergency happens or a report is needed.

Lastly, the system could be further enhanced by integrating with existing university information systems, including the course information, grade management, and attendance

management. Thus, the system is functional and usable to an educational institution by including the academic system.

In conclusion, by implementing a chat system, mobile support, and the existing university information system, the integrated timetable system can be more efficient, robust, and user-centered.

REFERENCES

[1] P. G. Daniel, A. O. Maruf, and B. Modi, "Paperless master timetable scheduling system," International Journal of Applied, vol. 8, no. 2, 2018. [Online]. Available: https://www.researchgate.net/profile/Bala-Modi/publication/326595321_Paperless_Master_Timetable_Scheduling_System/links/ 5e21b362a6fdcc1015715590/Paperless-Master-Timetable-Scheduling-System.pdf

[2] A. Chowdhary, P. Kakde, S. Dhoke, S. Ingle, R. Rushiya, and D. Gawande, "Timetable generation system," International Journal of Computer Science and Mobile Computing, vol. 3, no. 2, pp. 410–414, 2014. [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/33058276/V3I2201474-libre.pdf?1393951167=&response-content-disposition=inline%3B+filename%3DTIMETABLE_GENERATION_SYSTEM.pdf& Expires=17336 01408&Signature=eV1WlB 1o0gaV9Ef2KnvnWeX3~WuPPZXxB6AX-AGdsvEzEHve4hdUNQiYSH2MpVIU8Z1xh7Ul-TsKy7zqOGEpFlGEdI8MG6BO0i6t0nzYwI5l0felkU32MOf3CxRij3206pGm6jCnbb ReJyd~MU5O 65JwRgbfRSnCs1JWbYQ2VnJwwTLaYoGKoGtISMHdqLDv9MgxAMXnjKMTupZ DvWk4Q7A5IFa XzowaRIidRTZBX02KzySInonI1U1S0pdCq9iM92qkslQXOkrFTNgRZYJne0I2ao7d Oq2TlUw1- AFKg2ifzdlwuBbYYZ~kcrymrT5cUePk7kJ7Xpq7mkM-fLz3w &Key Pair-Id=APKAJLOHF5GG SLRBV4ZA

[3] M. Dimopoulou and P. Miliotis, “Implementation of a university course and examination timetabling system,” European Journal of Operational Research, vol. 130, no. 1, pp. 202–213, Apr. 2001, doi: https://doi.org/10.1016/s0377-2217(00)00052-7.

[4] D. Mittal, H. Doshi, M. Sunasra, and R. Nagpure, "Automatic timetable generation using genetic algorithm," International Journal of Advanced Research in Computer and Communication Engineering, vol. 4, no. 2, pp. 245–248, 2015. [Online]. Available https://d1wqtxts1xzle7.cloudfront.net/55582644/IJARCCE4I-libre.pdf?1516374774=&response-content-disposition=inline%3B+filename%3DAutomatic_Timetable_Generation_using_Gen.pdf&Expi res=1733603236&Signature=YoyPrtDg8sEavaIQoHsEIsb-BeBeTsZmGEztlKXg7oC3vIjCjeLHAV2aCuPi8HiIk1TFKlIOancNutkgQgSNGbXnx

pR45GslNO1Zt7 GaY8fiENLBv6~fIat-X9lBS9mrMnY9azTNnlS1ftsHUkxMF7RYQt7Q3RexEo3smbIKJYlLnMHejPZ6kdS yaoYwKMHhxa 7wLVFrps9nZXGrqettpdfJKk5PVX~NJCe3gSEsCxbgwF-kiO~RRwDxtQgK9alI4aHAvJlevxa6ffYbqs4p1gYOhzTHUQRJSPxobzbqT4qE5pZs FYyaaAQjgG09 KWFultWes5jUO91u9eKs3CDKWQ &Key-Pair Id=APKAJLOHF5GGSLRBV4ZA

[5]"Schedule Management System | International Journal of Research in Engineering, Science and Management," journal.ijresm.com, Apr. 2024, Available: https://journal.ijresm.com/index.php/ijresm/article/view/2999

[6] S. R. P. van Hal and K. N. M. M. H. Osman, "Automated timetable generation for Egyptian schools," July. 12, 2018. https://repository.tudelft.nl/file/File_a603880f-2606-4c54-b691 7841f12967c7?preview=1

[7] M. Müller-Hannemann and M. Schnee, "Efficient Timetable Information in the Presence of Delays," Robust and Online Large-Scale Optimization, pp. 249–272, 2009, doi: https://doi.org/10.1007/978-3-642-05465-5_10.

[8] F. P. Diallo and C. Tudose, "Optimizing the Scheduling of Teaching Activities in a Faculty," Applied Sciences, vol. 14, no. 20, p. 9554, 2024, doi: https://doi.org/10.3390/app14209554.

[9] H. Alghamdi, T. Alsubait, H. Alhakami, and A. Baz, "A Review of Optimization Algorithms for University Timetable Scheduling," Engineering, Technology & Applied Science Research, vol. 10, no. 6, pp. 6410–6417, Dec. 2020, doi: https://doi.org/10.48084/etasr.3832.

[10] R. Ganguli and S. Roy, "A Study on Course Timetable Scheduling using Graph Coloring Approach," International Journal of Computational and Applied Mathematics, vol. 12, no. 2, pp. 469–485, 2017.

[11] P. Kenekayoro, "Incorporating Machine Learning to Evaluate Solutions to the University Course Timetabling Problem," Covenant Journal of Informatics and Communication Technology, vol. 7, no. 2, pp. 18–35, Dec. 2019.

[12] A. Malikov, V. Voronkin, and S. Pevchenko, "Software and hardware infrastructure for timetables scheduling in university," CEUR Workshop Proceedings, vol. 2254, pp. 191–198, 2018.

[13] H. Rudová, T. Müller, and K. Murray, "Complex university course timetabling," Journal of Scheduling, vol. 14, pp. 187–207, 2011.

[14] E. K. Burke, D. Elliman, and R. Weare, "A genetic algorithm based university timetabling system," in Proc. 2nd East-West Int. Conf. Computer Technol. Educ., vol. 1, Sep. 1994, pp. 35- 40.

[15] “Free College Schedule Maker | Gizmoa,” gizmoa.com. https://gizmoa.com/college-schedule maker/

[16] “The Timetable Factory | Free College Timetable Builder,” Thetimetablefactory.com, 2025. https://www.thetimetablefactory.com/

[17] “Powerful school scheduling software for Mac, PC, tablet, phone,” Primetimetable.com, 2025. https://primetimetable.com/#support&id=b918a5b4-7da5-4809-9e51-1722b3ea3207 (accessed Apr. 28, 2025).

[18] Skooly, “Skooly,” getskooly, 2025. https://getskooly.com/skoolyweb/#/academy/dashboard (accessed Apr. 28, 2025).

[19] “Display Demo Instance URL | OpenEduCat,” OpenEduCat Inc, 2025. https://openeducat.org/thank-you-for-trying (accessed Apr. 28, 2025).

POSTER

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY INFORMATION AND COMMUNICATION TECHNOLOGY

Integrated Timetable System

1. INTRODUCTION

To develop an Integrated Timetable System that automates course scheduling, resource allocation, and real-time updates to improve efficiency, transparency, and the academic experience for students and educators.

2. OBJECTIVE

- Define student, lecturer, and admin functions
- Design a personal calendar
- Define the dynamic timetabling
- Develop a transparency resource utilization view

3. METHOD

- Agile Methodology
- PHP, HTML, CSS, JavaScript

4. CONCLUSION

In the era of globalization, this project aims to replace traditional timetabling by providing a quick, convenient system for course registration, resource allocation, timetable viewing and editing, class cancellation and real-time updated announcements, and resource utilization rates in graph view.

Project Developer: Yap Zhong Yew
Project Supervisor: Ts Dr Chan Lee Kwun