

Project 3 Report

**Semi-Structure Data Analytics**



Saw Zwe Wai Yan 6318013

CSX 4205 Big Data Analytics (1/2023)

|  |  |
| --- | --- |
|  |  |

ABSTRACT

In this report, we embark on a data analytics journey, leveraging MongoDB and a set of real-world data from the "sample\_supplies" collection. Our mission is to extract valuable insights, answer specific queries, and present our findings to illuminate the underlying trends and patterns. Our data pipeline is meticulous, consisting of five distinct stages: Ingestion, Transformation, Preparation, Processing, and Result. We employ a Python-based notebook tool for this analysis, ensuring our readiness for future data-driven challenges.

The culmination of our work is presented in a Jupyter Notebook, accessible on GitHub. This notebook serves as a comprehensive article, meticulously outlining our methodologies, code, and visualizations. It stands as a testament to our dedication to the art of data analysis.

Keywords: Data Analytics, MongoDB, Jupyter Notebook, GitHub, Ingestion, Transformation, Insights, Data Pipeline, Query Analysis.

Table of Contents

1. Introduction
2. Data Pipeline
   * Ingestion
   * Transformation
   * Preparation
   * Processing
   * Result
3. Query Analysis
   * Query 1: Top 10 Products by Sales
   * Query 2: Top 3 Products by Sales for Each Store
   * Query 3: Store Rankings
   * Query 4: Purchased Method by Gender
   * Query 5: Monthly Total Sales
4. Presentation Video
5. Notebook as an Article
   * Methodologies
   * Code and Visualizations
6. Conclusion
7. References

# 1 INTRODUCTION

In this report, we're diving into the world of data analysis. We're using a dataset called "sample\_supplies" from MongoDB to find interesting insights and answer some important questions.

We'll follow a step-by-step process to analyze the data, and we're using a handy tool called a Jupyter Notebook to do this. The results will be shared in a notebook on GitHub, where we'll explain our findings

# 2 DATA PIPELINE

We get the Data from MongoDB sample Data, we use the MongoDB tools to get the data. We can choose between 2 format. CSV format and Json format. I have chosen the Json format since the original data is also in Json format. I use pandas to read the json format. I exported the file using mongo export. The command is

mongoexport --uri "<yourDatabaseString>/sample\_supplies" --collection sales --out sales.json

# 3 QUERY ANALYSIS

**3.1 Show top 10 products (name) sales (quantity x price).**

A graph with blue squares

Description automatically generated

This query is to show the top 10 sales of product. There is no sale column thus we have to calculate for it. We can do that by multiplying quantity and price. We can see the visualization from the graph that, Laptop has the most significant sales compared to other items.

**3.2 Show top 3 products (name) sales by store (location).**

A group of blue squares

Description automatically generated

The graph represents each location and what their sales are. We can conclude that all locations perform well with the laptop, and next we have backpack and lastly we have pens. These are the top 3 products.

**3.3 Show rankings of each store (location).**

**A graph of blue rectangular bars

Description automatically generated with medium confidence**

For the rankings there are 2 ways you can approach, either by the sales or by the satisfaction of the customer. I have chosen customer satisfaction, we can conclude that the New York, London and Denver are tided for the first place with an average rating of 3.8/5 and the lowest rating is from Austin with a rating of 3.77/5.

**3.3 Show purchased method by gender table**

A graph of purchase by gender

Description automatically generated

This shows the demographic of people who purchases the items online or in store. We can conclude that for both genders, both of them prefer buying items in store rather than online.

**3.5 Show monthly total sales**

**A graph with colorful lines and arrows

Description automatically generated**

The graph represents how much each month made sales with the range of 5 years, 2013 – 2017. From the graph we can see that the sales performance generally rises during April or July. Meaning we can have better deals during that time.

# 4 VIDEO PRESENTATION

# 5 NOTEBOOK

<https://github.com/sawzwe/bigdata_project3/blob/6254f14f75897bcb080320a53322c1c2f827c9bf/project_3.ipynb>

# 1 INTRODUCTION

This chapter introduces the project and sets the stage for the rest of the report. It includes three main sections: Rationale and Motivation, Overview of the Project, and Goals and Objectives.

The Rationale and Motivation section outlines why this project is important and necessary. It explains the reasons behind the decision to develop a web application for House of Griffin (HOG) and how it will benefit the organization and its stakeholders. This section will provide an understanding of the context and background of the project.

The Overview of the Project section provides a summary of what the project is about, including the main objectives and deliverables. This section will provide an overview of the scope of the project, the target users, and the main functionalities of the web applications.

The Goals and Objectives section outlines the specific goals and objectives of the project. It describes what the project aims to achieve and the expected outcomes. This section will provide a clear understanding of the intended results of the project and how they align with HOG's business goals and objectives.

## 1.1 RATIONALE AND MOTIVATION

House of Griffin (HOG) is a tutoring school based in Bangkok, Thailand. They specialized in tutoring specific course such as GED and SAT. They recognize the increasing importance of web applications in education and the need for digital tools that can enhance collaboration, streamline administrative tasks, and improve communication between teachers, students, and staff. Therefore, the project's purpose is to design and develop two comprehensive and userfriendly web applications - one for students and teachers and another for staff - that will assist HOG students in managing their schedules, tracking their progress, and accessing learning resources more efficiently. By providing a comprehensive solution that meets these needs, the project aims to improve the academic experience of HOG students and support their academic success, as well as enhance communication and organization within the educational institution.

## 1.2 OVERVIEW OF THE PROJECT

The project consists of two web applications: a student and teacher application, and a staff application. The student and teacher application will provide an interactive platform where students can access course materials, submit assignments, and communicate with their teachers. Teachers, on the other hand, will be able to manage their courses, upload materials, grade assignments, and communicate with their students. The staff application, on the other hand, will provide a centralized platform where administrators can manage school-wide activities, such as enrollment, scheduling, and resource allocation.

## 1.3 GOALS AND OBJECTIVES

The primary goal of the GriffinEd web application project is to develop a comprehensive and user-friendly web-based platform that will enhance the academic experience of HOG students and staff. The project objectives include:

* Designing and developing a web application that is intuitive, user-friendly, and accessible to all students.
* Implementing a secure and scalable database that ensures the safety and accessibility of all student data.
* Providing a platform for students to manage their schedules, track their academic progress, and access learning resources more efficiently.
* Enhancing the academic experience of HOG students and supporting their academic success through the development of an effective and innovative web application.

By achieving these goals and objectives, the GrffinEd web application project will provide a significant contribution to the education sector, demonstrating the importance of technology in improving academic outcomes and supporting student success.

# 2 BUSINESS REVIEW

The business review of House of Griffin (HOG) provides a foundation for the GriffinEd web application project, which aims to improve the course registration process for students and staff. The chapter highlights the pain points and problems experienced by HOG in their current workflow and explores the business requirements of the project. By outlining the project's goals and objectives, the chapter provides insight into how the project aims to address these pain points and improve the course registration process. Furthermore, the chapter discusses the scope and limitations of the project, which outlines the boundaries and constraints within which the project will operate. By addressing these considerations, the GriffinEd web application project can make a significant contribution to the education sector by utilizing technology to improve academic outcomes and support student success.

The problem statements section provides a comprehensive overview of the challenges and issues that HOG is experiencing. This section will identify the pain points of HOG's current workflow and highlight the areas where improvements are necessary. The business requirements section outlines the specific needs and expectations of HOG from the project. It provides an understanding of the features and functionalities that HOG requires from the web applications.

The scope and limitations section outlines the boundaries and constraints of the project. It identifies the features and functionalities that will be included in the web applications as well as those that will be excluded. The section will also highlight any technological or operational limitations that will impact the project's implementation and delivery.

## 2.1 PROBLEM STATEMENTS

House of Griffin (HOG) has been using paper-based system for quite a while. This can be a significant pain point for HOG and its stakeholders. A paper-based system is often slow, inefficient, and prone to errors. It requires manual data entry and handling, which can be timeconsuming and tedious for staff members. Paper-based systems can also be difficult to manage and track, as documents can be easily lost or misplaced.

Moreover, paper-based systems are not easily scalable and can become unwieldy as the organization grows. This can lead to additional costs and resource allocation for storage and management of physical documents. Additionally, paper-based systems can be vulnerable to data breaches or unauthorized access, as it can be difficult to control who has access to sensitive information. Lastly, it can leads to delays, errors, and inefficiencies, which can have a negative impact on both the organization and its stakeholders.

## 2.2 BUSINESS REQUIREMENTS

The GriffinEd system comprises two distinct web applications that have been specifically designed to facilitate and streamline the course registration process for multiple stakeholders, including students, teachers, education planners, education administrators, and office administrators. These web applications have been separated based on user roles and are intended to provide an optimized user experience. One application is a management platform specifically designed for staff members, while the other application is mobile-friendly and targeted toward students and teachers.

The primary objective of the GriffinEd system is to enhance the user experience and optimize the course registration process. The system achieves this by offering a user-friendly interface, real-time updates, and easy access to relevant information. The system is designed to improve the overall efficiency of the course registration process by reducing the amount of time and effort required to complete the registration process.



Figure 2.2.1: Business Requirements Gathering 1

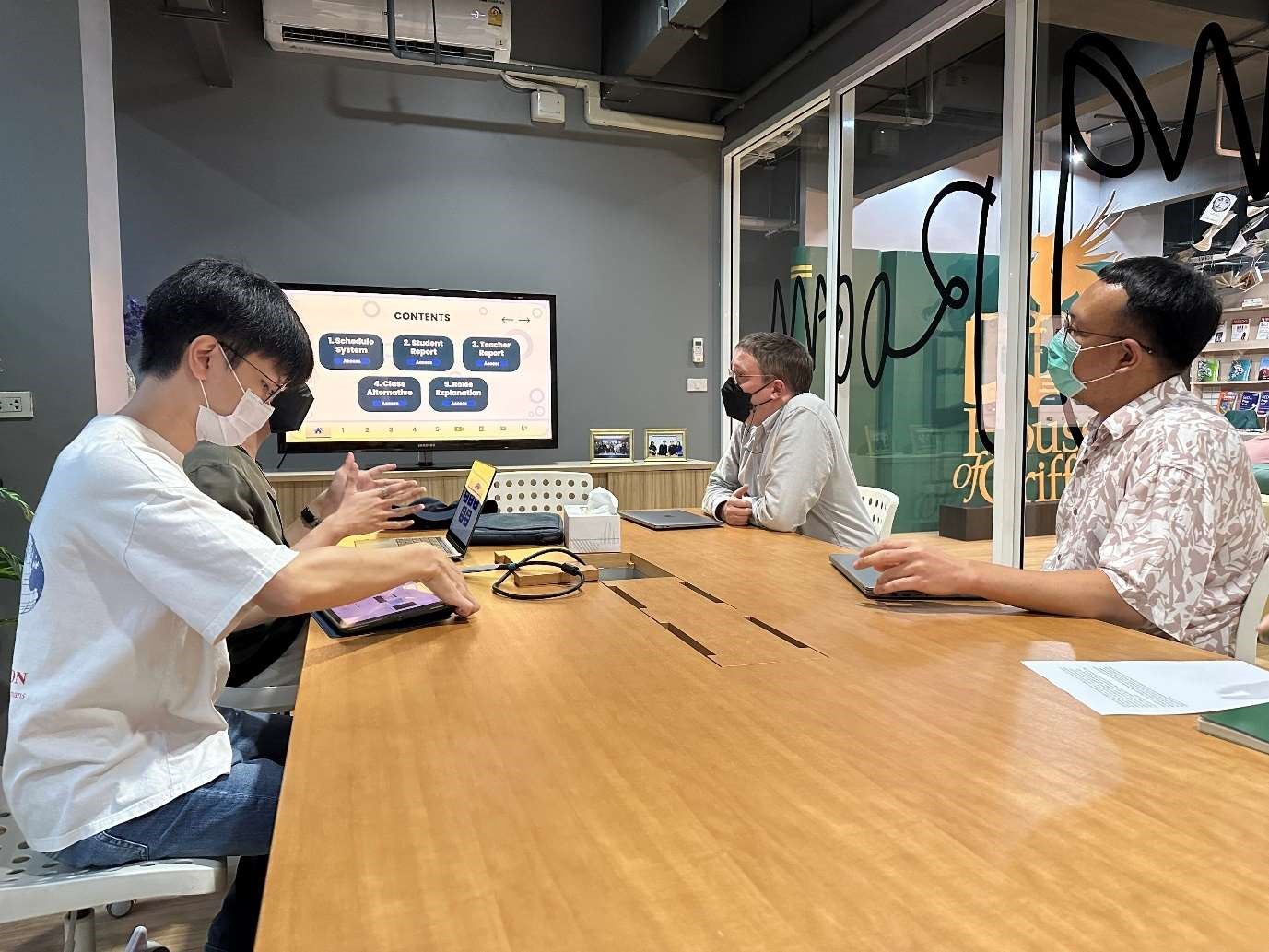


Figure 2.2.2: Requirements Gathering 2

The figure 2.2.1 and figure 2.2.2 depict the process of the requirements gathering during the first phase of the project.

### 2.2.1 STAKEHOLDERS

Stakeholders refer to individuals or groups who have a vested interest in or are directly affected by a project or system. Within House of Griffin, there are a total of five stakeholders who play critical roles in the GriffinEd system. These stakeholders include:

1. Students: These individuals are responsible for registering for courses, managing their schedules, and accessing course-related information.
2. Teachers: These stakeholders share the same web application as the students and are responsible for managing their course schedules, accessing student registration information, and providing course-related information.
3. Education Planners (EP): The Education Planner utilizes a separate web application from the student and teacher. They are responsible for handling requests such as student account creation and course registration. Furthermore, they act as intermediaries between the student and House of Griffin. They also collect payments for course registrations.
4. Education Administrators (EA): This group shares the same web application as the EP and is primarily responsible for schedule customization during the course registration process. They handle conflicts between class schedules and availability.
5. Office Administrators (OA): The highest-ranking members of the organization chart, they are responsible for approving course registrations and creating new staff accounts.

Like the EP and EA, the OA shares the same web application.

### 2.2.2 REQUIREMENTS

The GriffinEd system just satisfy the unique requirements of each stakeholder group to ensure the system's successful implementation and adoption. The requirements are as follows:

1. The system must allow students to register for courses, manage their schedules, and access relevant course-related information. Additionally, it must provide a user-friendly interface that is easy to navigate.
2. The system must enable teachers to manage their course schedules, access student registration information, and provide course-related information. Moreover, it should have features that allow them to communicate with their students and receive feedback on course content.
3. The system must provide an interface for Education Planners to handle student requests for account creation and course registration. It should also enable them to collect payments for course registrations and monitor the progress of student registrations.
4. The system must enable Education Administrators to customize schedules during the course registration process, handle conflicts between class schedules and availability, and monitor course enrollments.
5. The system must provide the Office Administrator with the ability to approve course registrations, create new staff accounts, and monitor the progress of course enrollments.

## 2.3 SCOPE AND LIMITATIONS

The scope and limitations of any system are critical to understanding its capabilities and potential shortcomings. The GriffinEd system, which comprises two web applications designed to facilitate the course registration process, is no exception. Understanding the scope of the system will help stakeholders appreciate its capabilities, while an understanding of its limitations will help identify areas for improvement or refinement.

### 2.3.1 SCOPE

The scope of any project defines the features, functionalities, and capabilities that are included in the system. In the context of the GriffinEd system, it is essential to have a clear understanding of the scope to ensure that all necessary features are included and that the system meets the needs of its stakeholders.

The following is a list of the scope of the GriffinEd system, which includes the essential features necessary for the system to function effectively. The list includes the features and functionalities of the two web applications that comprise the system, designed for students, teachers, education planners, education administrators, and office administrators.

#### I Course Registration

The process of course registration is an essential function of the House of Griffin's system. It is designed to accommodate students in enrolling for courses through a walk-in-based system. This approach leverages the use of schedule generation to aid in finding the appropriate schedule for the students and minimizing the workload for the staff.

House of Griffin offers two primary types of courses, namely, group courses and private courses. The schedules for group courses are predetermined and unalterable. Private courses, on the other hand, offer students the opportunity to request for customized schedules. Private courses are further divided into two subcategories: private and semi-private. Private courses involve one-to-one sessions, while semi-private courses are tailored for 2-25 students.

The Schedule Checking System is a vital component of the course registration process, which plays a crucial role in facilitating the customization of classes within private and semi-private courses. This system enables students to personalize their learning experience and ensures that they can attend all classes without encountering any schedule conflicts. By utilizing the Schedule Checking System during course registration, students can customize their course schedules according to their individual preferences and requirements, which in turn helps to optimize their academic outcomes.

For group courses provided by House of Griffin, there are three stakeholders involved in the course registration process. These stakeholders include the students, the Education Planner (EP), and the Office Administrator (OA). On the other hand, private courses involve four stakeholders in the registration process, including the students, the Education Planner (EP), the Education Administrator (EA), and the Office Administrator (OA).

#### II Schedule System

The scheduling system within the GriffinEd web application serves as a request-based platform for students to modify their schedules. This system is comprised of three distinct functionalities, each of which serves a specific purpose. These functionalities can be further categorized into three subcategories, encompassing the following:

1. Course Transfer

Course transfer refers to the process of transferring course hours from one course to another. It enables students to redistribute their hours among courses according to their preferences and needs. For instance, if a student enrolls in SAT Mathematics for 20 hours (10 hours) and SAT Verbal for 20 hours (10 hours), they can transfer the remaining hours from SAT Mathematics to SAT Verbal if they decide not to continue with SAT Mathematics after completing some hours. This process involves three stakeholders, including the student, the EP, and the EA. The student initiates the course transfer request, while the EP and the EA manage the transfer process by verifying the transfer request and updating the student's course schedule accordingly.

1. Hour Adjustment

Hour adjustments is a feature that enables students to request a change in class hours. For instance, if a student is scheduled to attend a 2-hour class on Monday but is only available for 1 hour, they can request to move the remaining hour to another day. In case the student is unavailable for the entire 2-hour class, they can request to reschedule the class to another day, which will count as a cancellation and a make-up class scheduled. This functionality involves two stakeholders, namely the students and the Education Administrator (EA).

1. Teacher’s Leave Request

This functionality is designed specifically for the teacher who needs to cancel a class. In order to ensure a substitute teacher can be found and assigned before the scheduled class time, the teacher must provide a valid reason for the cancellation. The involvement of three stakeholders is necessary for this functionality, including the teachers themselves, the Office Administrator (OA), and the Education Administrator (EA).

#### III Room Adding

"Room Adding" is a crucial functionality designed to provide the EA with the ability to update the teaching rooms that will be utilized the following day. This feature eliminates the need to manually update the schedule in the lobby for students to view. With this functionality, updated room information will be promptly displayed on the application, and the EA can efficiently manage it from the web application interface.

#### IV Today’s Class

The "Today's Classes" function serves to display the daily schedule of classes for both students and teachers, allowing them to easily identify the courses they need to attend on a given day. This function is updated daily to ensure that the displayed information remains accurate and relevant.

#### V Student’s Attendance and Teacher’s Teaching Hours Summary

The attendance of students can be recorded using the mobile application, enabling the teacher to receive compensation for the particular class of the day. The accumulation of teaching hours occurs throughout the month, summarizing the total number of hours for each teacher. These teaching hours play a pivotal role in calculating the remuneration of the teacher, including parttime personnel.

### 2.3.2 LIMITATIONS

As with any project, the House of Griffin's web application project has certain limitations that must be taken into consideration. These limitations define the boundaries within which the project will operate and can have an impact on the project's ultimate success. Understanding and acknowledging these limitations is crucial in ensuring that the project remains feasible, realistic, and achievable within the given constraints.

The limitations of this project are primarily related to time constraints and a lack of foundation. Due to the project's timeline, there may be limitations on the scope of the project and the features that can be implemented. Additionally, the project may face challenges due to a lack of a well-established foundation, which may result in unforeseen issues and complications during the development process. These limitations will need to be carefully considered and managed to ensure that the project stays on track and delivers the desired outcomes within the given timeframe. Moreover, the development team consists of individuals who have not had any formal experience in developing an education management web application. While the team members possess programming skills and have completed relevant courses, they have not previously worked on a project of this scale and complexity. This lack of formal experience could result in potential delays, errors, and inefficiencies in the development process. Additionally, the team members may need to spend extra time researching and learning new technologies and concepts related to the project, which could further slowdown the development timeline. To address this limitation, the team plans to allocate extra time for research, testing, and development to ensure the successful delivery of the project within the given timeline. The followings are the scope that will not be implemented:

1. Schedule System
2. Teacher’s Teaching Hour Summary

# 3 METHODOLOGIES

This chapter provides an overview of the methodology used in the development of the web applications for House of Griffin (HOG). The chapter is divided into two main sections: Diagrams and Database Designs.

The Diagrams section presents the various diagrams that were used in the development process, including sequence diagrams. These diagrams were used to visualize the flow of data and information through the system, as well as the interactions between different components of the web applications. The section will also explain the purpose and function of each diagram and how they were used in the development process.

The Database Designs section provides an overview of the database designs used in the development process. It includes a detailed description of the database schema and the various tables and fields that were included. This section also discusses the decision-making process behind the choice of database management system and the specific tools and technologies used in the development process.

## 3.1 USER JOURNEY

The user journey is a critical component of any product or service. It outlines the various stages that a user goes through while interacting with the system, from the initial stage of browsing to the final stage of completing an action or task. Understanding the user journey is essential for designing a user-centric experience that meets the needs and expectations of the user.

In this chapter, the user journey of House of Griffin will be explored in the context of a specific functionalities. The various stages of the user journey will be examined, the touchpoints where users interact with the UI/UX, and the emotions and motivations that drive user behavior. In addition, techniques for mapping, analyzing, and optimizing the user journey to enhance the overall user experience will be further looked at. These methods aim to provide a better understanding of how users interact with the product or service, identify pain points and areas for improvement, and implement changes that optimize the user journey. Ultimately, these strategies aim to create a more efficient, intuitive, and satisfying user experience.

### 3.1.1 COURSE REGISTRATION

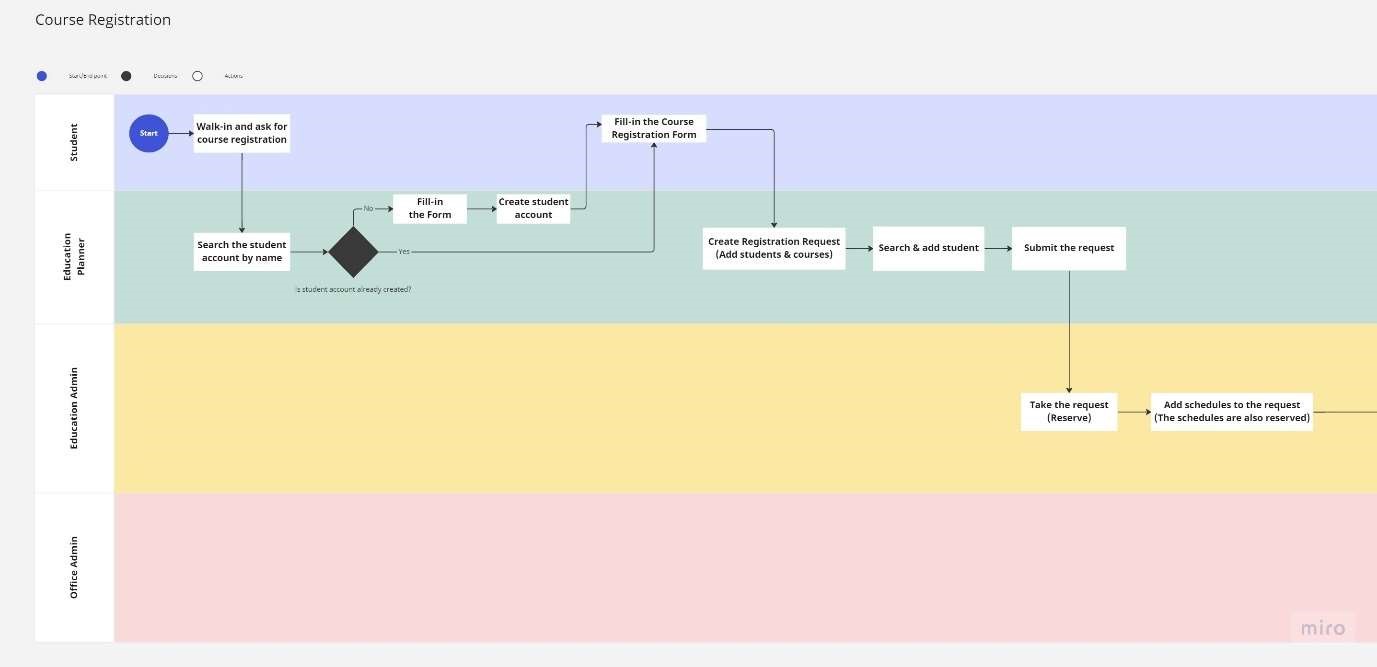


Figure 3.1.1.1: Course Registration User Journey 1

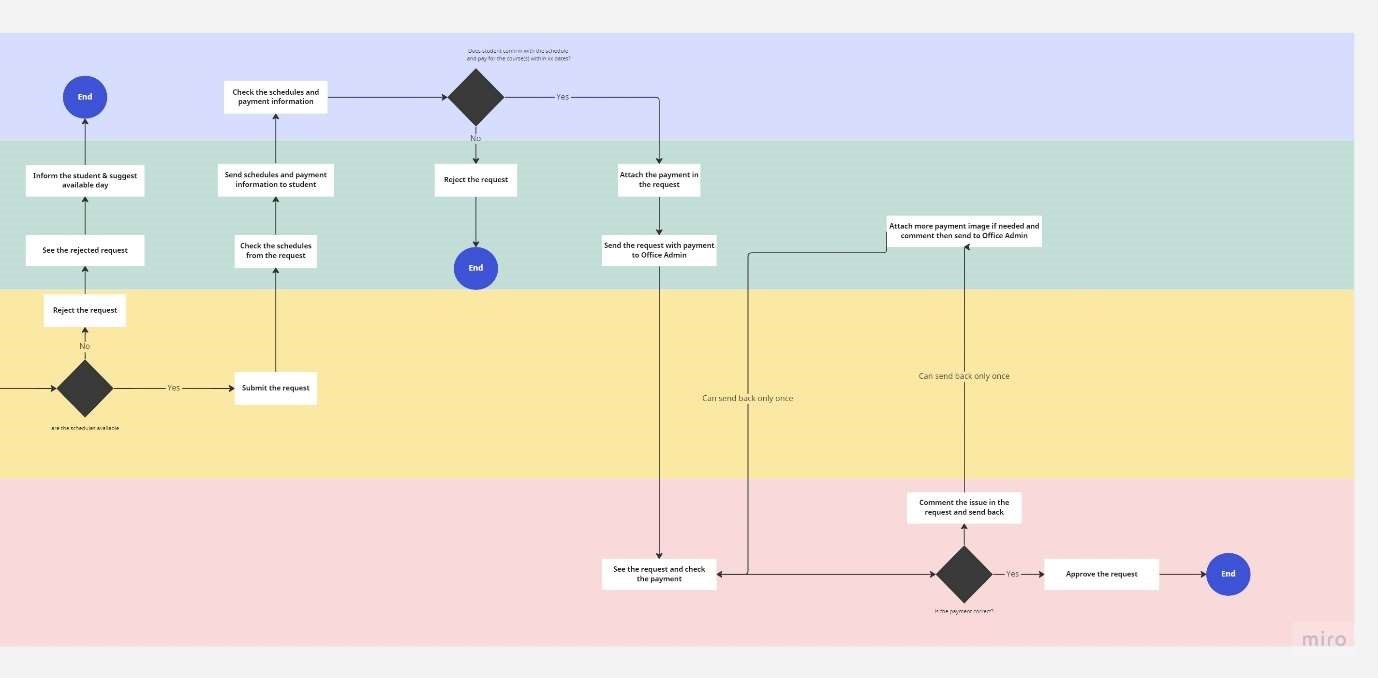


Figure 3.1.1.2: Course Registration User Journey 2

The figures 3.1.1.1 and 3.1.1.2 depict the user journey of the course registration process. These figures outline the various stages that a user goes through while registering for a course, from the initial stage of creating the request to the final stage of confirming enrollment. The user journey is presented in a sequential and systematic manner, highlighting the specific steps involved in the registration process. These figures serve as a formal representation of the course registration process, providing a visual aid that can be used to guide users through the process and ensure a smooth and efficient experience.

The user journey begins when the student talks to the EP (Education Planner) about course registration. If the student does not have an account, the EP creates one and fills in the necessary information to create a registration request. The EP then submits the request to the EA (Education Administrator) for processing.

If the EA approves the request, they create a schedule for the requested course. If the EA rejects the request, the EP informs the student that the request has been denied.

Once the request is approved, the EP collects payment from the student. The OA (Office Administrator) checks if the payment is valid, and if not, the registration request is rejected. If the payment is valid, the EP informs the student that the course has been successfully registered.

Overall, the figure 3.1.1.1 and 3.1.1.2 user journey involve multiple steps, each of which is crucial for ensuring a smooth and efficient registration process. By following this journey, the student can register for their desired course with minimal hassle, while the EP, EA, and OA can work together to ensure that the registration process is streamlined and effective.

### 3.1.2 EDIT SCHEDULE

There are two distinct methods by which the schedule can be modified: one involves a student, while the other involves a teacher. It should be noted that the responsibility for performing the action in question is limited solely to the EA role. Specifically, the EA is the only authorized individual who possesses the requisite privileges to carry out this task. It is important to recognize that this is a restricted function and that no other roles or individuals are authorized to perform it. As such, it is imperative that any action related to this function is only undertaken by individuals who hold the EA role and have been duly authorized to do so.

#### I Edit Schedule via Student

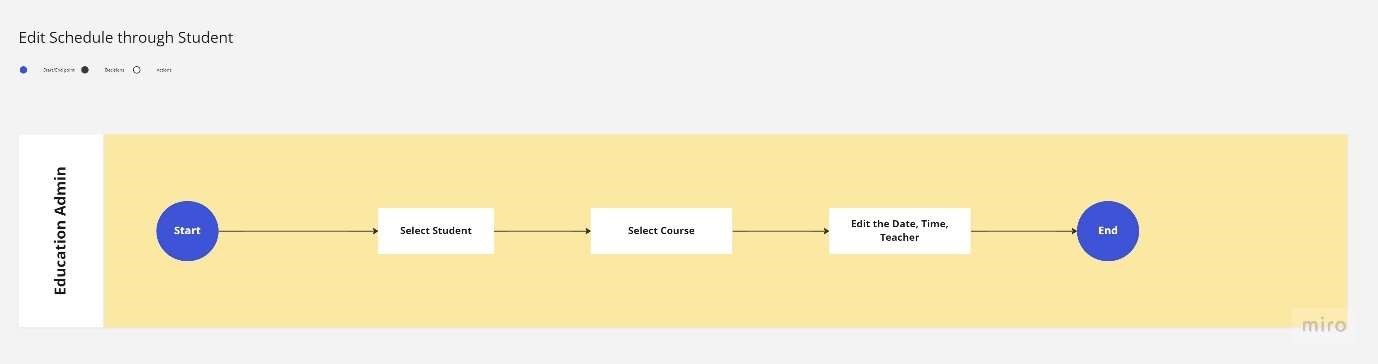


Figure 3.1.2.1: Edit Schedule via Student User Journey

The figure 3.1.2.1 illustrated a method of schedule modification that revolve around the student. Firstly, the EA must select the student for whom they want to modify the schedule. Then, they must choose the course that they wish to modify. Once the course is chosen, the user can edit the schedule by modifying the date, time, and/or teacher assigned to the course. Finally, the EA must confirm the changes they have made to the schedule and save the updated schedule. If desired, the EA may also choose to cancel the changes or make further modifications before ending the user journey.

#### II Edit Schedule via Teacher

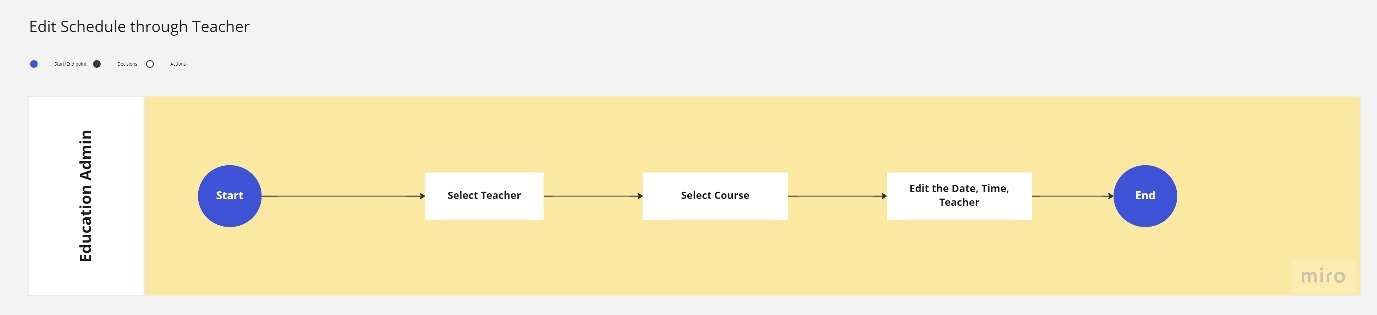


Figure 3.1.2.2: Edit Schedule via Teacher User Journey

It is notable that the figure 3.1.2.2 bears a strong resemblance to the figure 3.1.2.1, with the sole exception being that the user's selection has shifted from a student to a teacher. Specifically, the process by which the user modifies a schedule for a selected teacher is essentially the same as the process described in figure 3.1.2.1 for modifying a schedule for a selected student. The EA must still access the system, then choose the teacher they wish to modify the schedule for. They must then select the course they wish to modify, and proceed to edit the date, time, and/or teacher assigned to the course. Finally, the user must confirm the changes made and save the updated schedule. The primary difference between the two figures, therefore, is the user's initial selection of a student versus a teacher.

#### IV Quick Edit Schedule

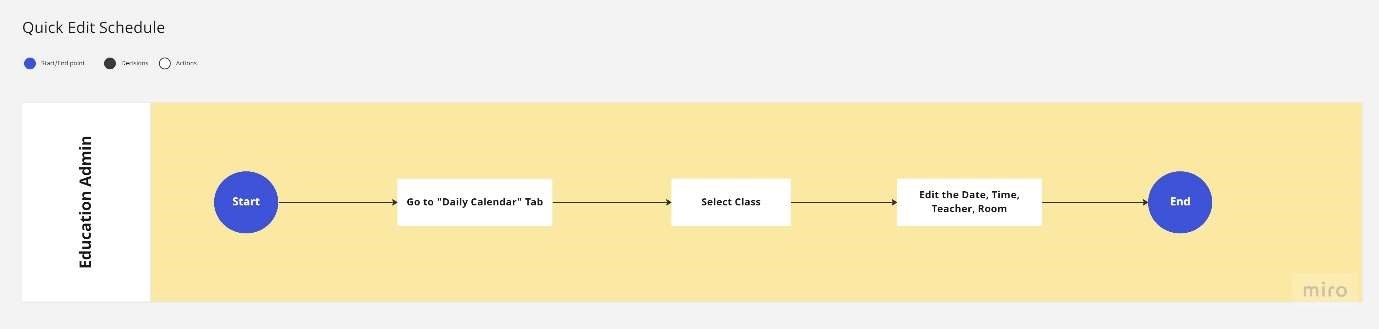


Figure 3.1.2.3: Quick Edit Schedule User Journey

The figure 3.1.2.3 depicts the user journey for a quick edit schedule, which is a distinct process from the previously described methods of modifying a schedule. In this case, the EA is presented with the option to access the "Daily Calendar" tab, which displays a list of classes. From this list, the EA can select the desired class they wish to modify by choosing the appropriate options for date, time, teacher, and room. After the modifications have been made, the EA must confirm the changes and save the updated schedule. It is worth noting that the EA also has the option to cancel the changes or make further modifications before concluding the user journey. It is important to emphasize that this process is only available to users with the EA role, as they are the only individuals authorized to perform this task. As such, it is essential that the EA exercise due caution and care when carrying out this task, as any changes made to the schedule will have significant consequences for the affected individuals and departments.

### 3.1.3 VIEW SCHEDULE

The following user journey outlines the steps for viewing the schedule via the system. It is important to note that this process is exclusively available to individuals who hold either the student or teacher role.

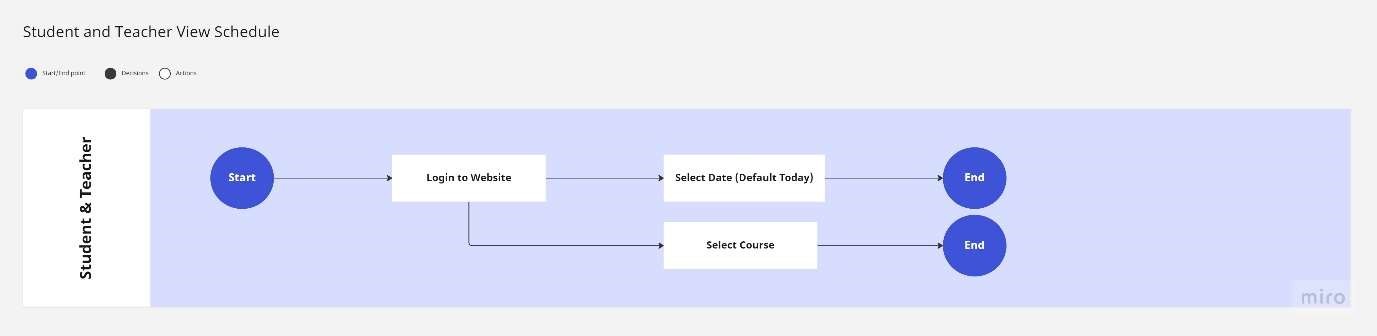


Figure 3.1.3.1: View Schedule User Journey

The formal process for accessing the scheduling platform and viewing the schedule involves several steps. First, the user must gain access to the platform by logging in with valid credentials to ensure secure access. After the authentication process, the user will be assigned the appropriate user role, either student or teacher. Then, the user must select the specific course or courses for which the schedule needs to be viewed. The scheduling platform will present the schedule information, including the date, time, teacher, and room, in accordance with the default settings of the system. It should be noted that the default settings of the system are configured to return the schedule for the current date. Therefore, if the user wishes to view the schedule for a different date, they must adjust the settings accordingly to retrieve the relevant information. It is important to emphasize that this process is only available to individuals who hold either the student or teacher role, and any attempt by unauthorized individuals to access or modify the schedule is strictly prohibited and will be subject to disciplinary action. Additionally, it is the responsibility of the users to maintain the integrity of the schedule information and ensure that it remains accurate and up to date.

### 3.1.4 CHECK ATTENDANCE

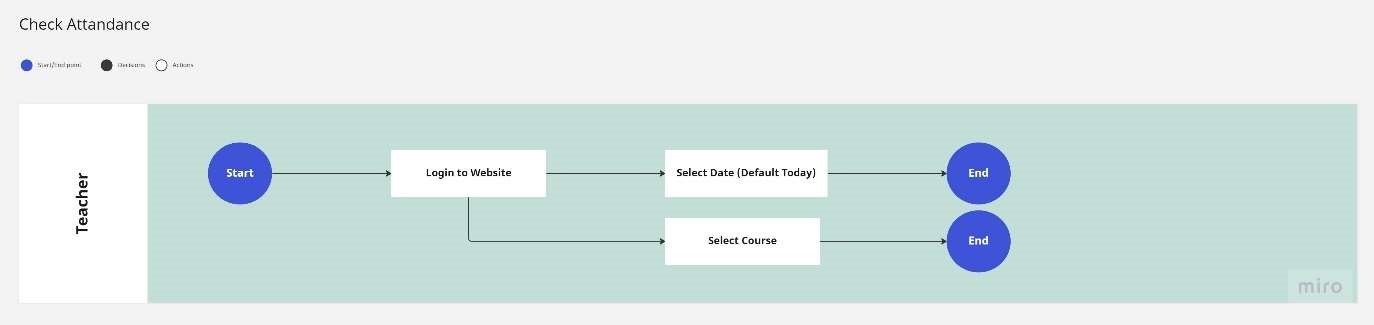


Figure 3.1.4.1: Attendance Check User Journey

The process of checking attendance is similar to viewing the schedule, with the distinction that only teachers are authorized to perform this function. To initiate the attendance check, the teacher must first access and log in using valid credentials to ensure secure access. Once authenticated, the teacher must select the appropriate class and student to check their attendance. The system will present the necessary attendance information for the selected class and student. The teacher must then review the attendance information and mark the student as present or absent, as per the relevant policies and procedures. It is important to note that this function is strictly limited to teachers, and any attempt by unauthorized individuals to access or modify attendance information is strictly prohibited and will be subject to disciplinary action. It is also the teacher's responsibility to maintain the accuracy and integrity of attendance records and ensure that they are always up to date.

## 3.2 USABILITY TESTING

A usability assessment was conducted on the web application of the House of Griffin tutoring school, with the aim of evaluating user satisfaction. The assessment was performed in-person and involved four members of the House of Griffin back-office staff, who utilized their individual computing devices to test the website. The purpose of this evaluation was to gauge the ease of use, efficiency, and overall user experience of the House of Griffin tutoring school's web application. It should be noted that the assessment was conducted by authorized personnel only, and all results were handled in accordance with established data protection protocols.

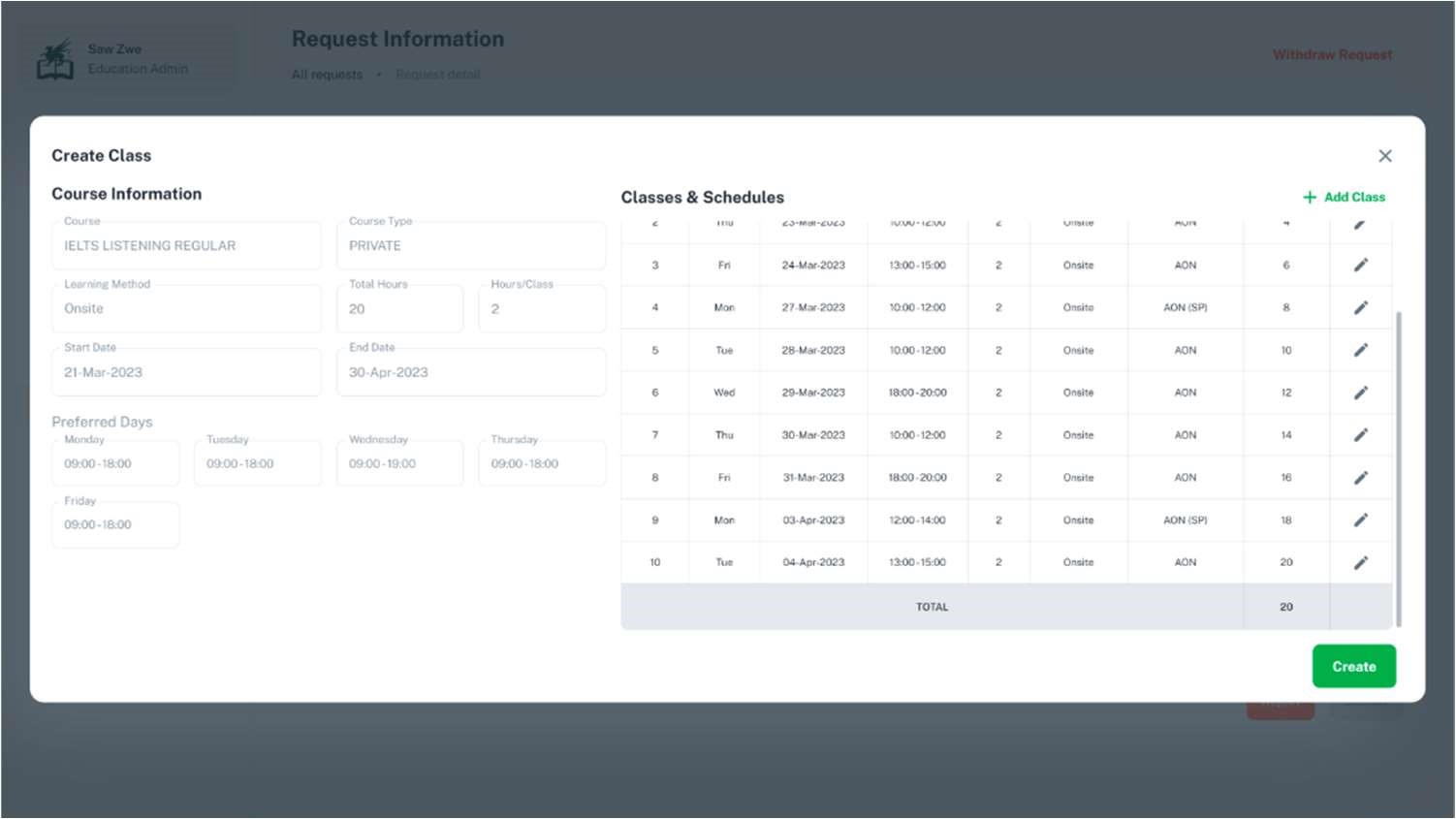


Figure 3.2.1: Create Schedule function in Education Admin Page

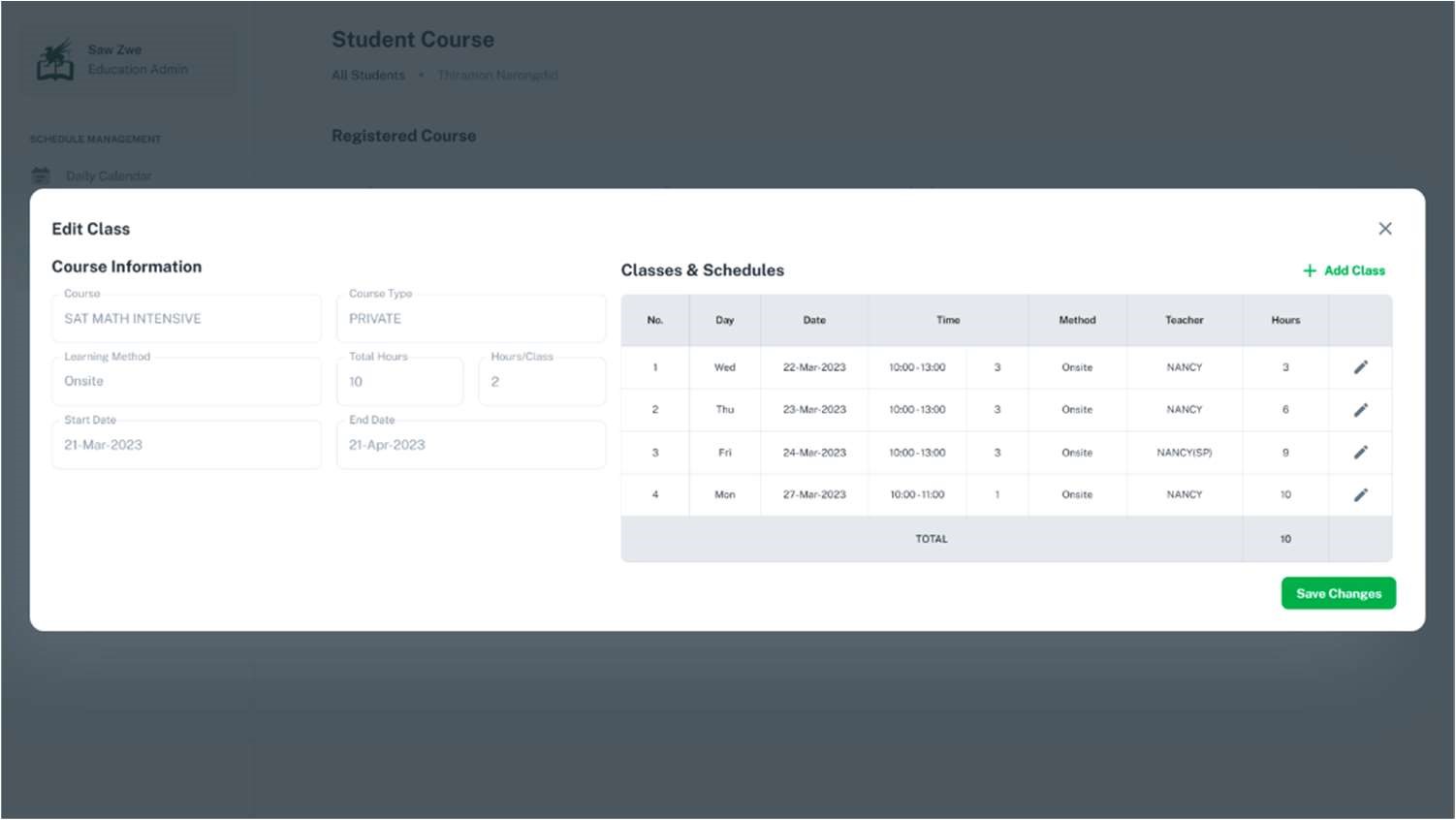


Figure 3.2.2: Customize Schedule function in Education Admin Page

The results of the usability assessment were largely positive, with a few areas of concern identified in relation to certain website functions. Specifically, the "Create schedule" function illustrated in Figure 3.2.1 and the "Customize schedule" function depicted in Figure 3.2.2 were found to be time-consuming in their current form. Additionally, upon clicking the "add class" button, a modal window appeared, obstructing the view of the remaining schedule and making it difficult to identify available time slots. This, in turn, made it challenging to determine why a particular time slot had already been booked. These identified issues highlight potential areas for improvement in order to optimize the user experience of the House of Griffin tutoring school's web application.

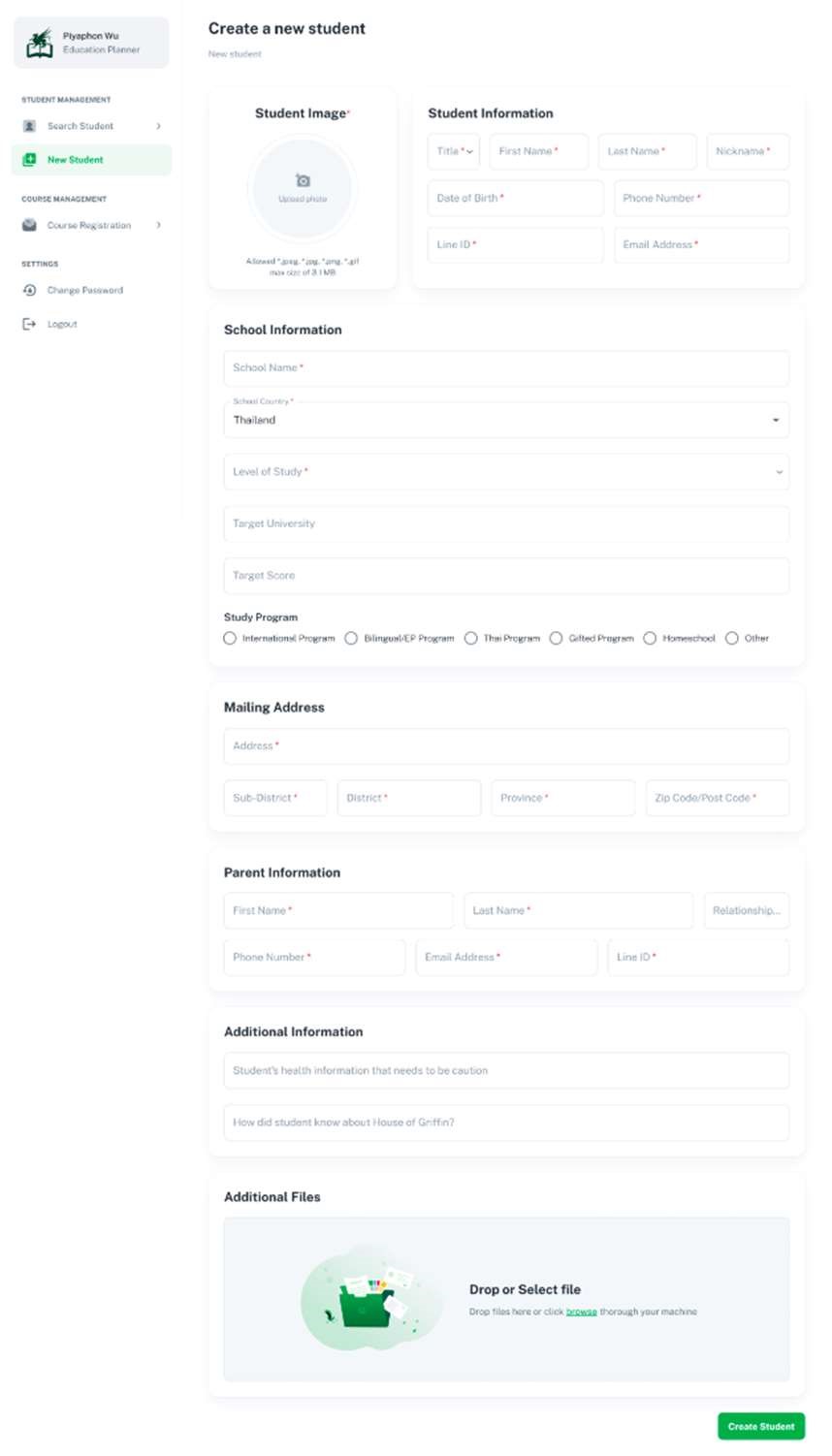


Figure 3.2.3: Create Student in Education Planner Page

Furthermore, the form utilized for creating student accounts, as depicted in Figure 3.2.3, was found to be excessively lengthy, which led to situations where some students were unable to provide all the necessary information to the staff. In order to address these identified pain points, adjustments could be made to the UI/UX of the website to enable users to add each class to the schedule without the need for a modal popup. Additionally, a function could be incorporated to streamline the registration form for students who are unable to provide all of the required information. The results of the usability assessment, therefore, provided valuable insights into areas where improvements could be made to enhance the overall user experience of the House of Griffin website, thus improving customer satisfaction.



Figure 3.2.4: Usability Test 1



Figure 3.2.5: Usability Test 2

The figure 3.2.4 and figure 3.2.5 shows the process of usability testing.

# 4 SYSTEM FRAMEWORK

This chapter provides an overview of the system framework that has been designed for the House of Griffin web applications. It includes two main sections: System Architecture and System Implementation.

The System Architecture section describes the overall architecture of the system, including the different components and how they interact with each other. It outlines the hardware and software requirements, as well as the technical infrastructure that will be used to support the web applications. This section provides a high-level view of the system, highlighting the main building blocks and how they fit together.

The System Implementation section focuses on the implementation details of the system. It provides a detailed description of how the system has been developed, including the programming languages, development tools, and methodologies that have been used. It also includes information on the different modules and components that make up the system, as well as the data structures and algorithms that have been employed. This section provides an in-depth view of the system implementation, detailing the technical aspects of the development process.

## 4.1 SYSTEM ARCHITECTURE

Client-side:

* The user interacts with the web applications through a web browser or a mobile app.
* The client-side code is written in React using the template from Material UI (MUI).
* The client-side code is responsible for rendering the user interface and handling user interactions.

Server-side:

* The server-side receives requests from the client-side and sends responses back.
* The API is implemented using a .NET CORE 7 framework and Entity Framework to manage the database.

Database server:

* The database server stores data for the web application.
* The Azure SQL database is used as a database server.

Web server:

* Azure web service is used as a web server for both the client side and the server side.

Authentication server:

* The authentication server is responsible for authenticating users and generating access tokens.
* Firebase and external API service are used as an authentication server.

### 4.1.1 AUTHORIZATION ARCHITECTURE

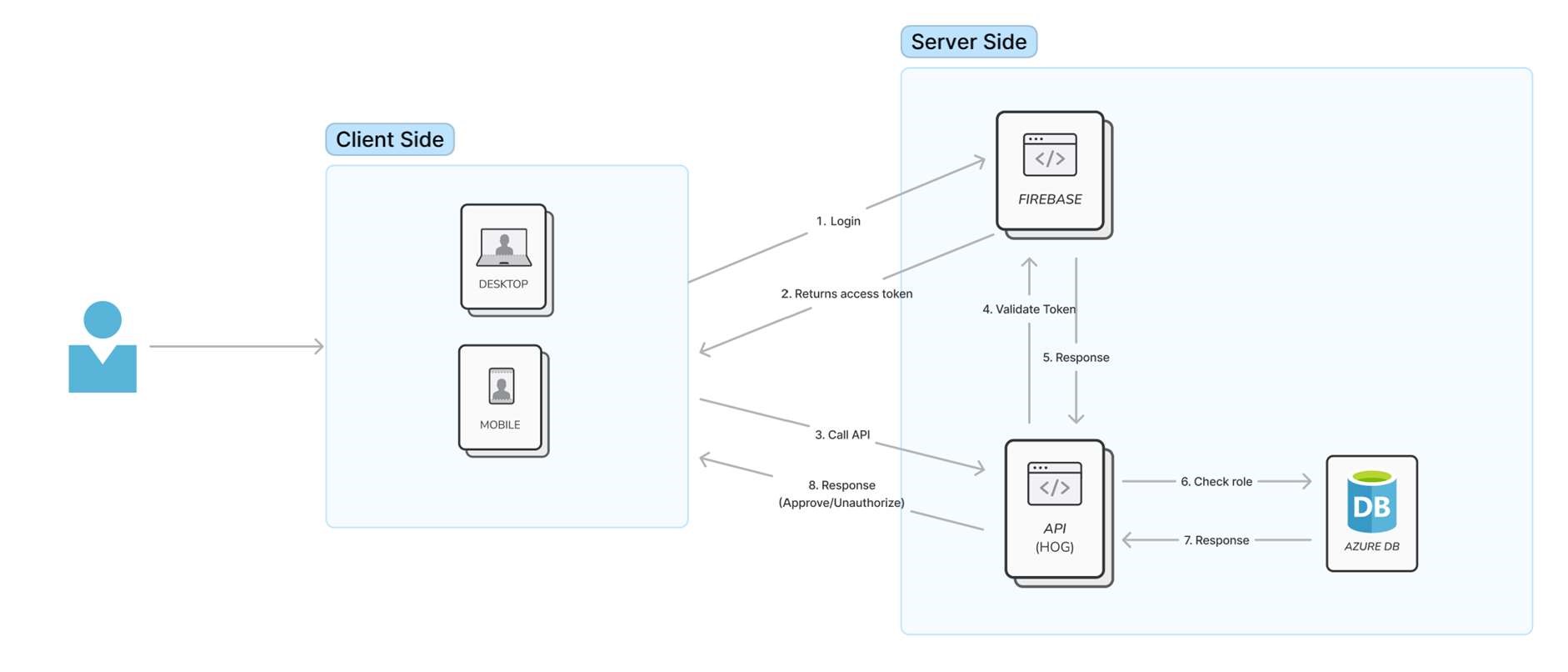


Figure 4.1.1.1: GriffinEd Authorization System Architecture

The authorization system architecture used by GriffinEd is depicted in Figure 4.1.1. The architecture allows users to interact with the system through the client-side. Upon sending a login request to Firebase, the system responds with an access token. This token is then utilized whenever a user attempts to access an API.

To authenticate the token, the API sends it to Firebase for validation. After receiving a response from Firebase, the API verifies the user's role through the database. The database subsequently returns a response to the API, either granting access to the function or denying access if the token is invalid.

This authorization system follows a formal procedure in which the user's identity is first authenticated by Firebase, and then verified through the database. The use of tokens adds an additional layer of security by ensuring that only authorized users are granted access to the system's functions.

## 4.2 FUNCTIONALITIES

This chapter explores the various functionalities of a system or software, focusing on the different features and tools that users can utilize to accomplish their tasks efficiently. The functionalities of a system are the essential components that make it useful and valuable to its users, and they can range from basic operations to complex processes. By understanding the functionalities of a system, users can leverage its capabilities and optimize their workflow to achieve their objectives. This chapter will provide an in-depth analysis of the functionalities of a system, highlighting their benefits, limitations, and best practices for effective usage. Whether you are a beginner or an experienced user, this chapter will equip you with the knowledge you need to leverage the functionalities of a system to achieve your goals.

### 4.2.1 COURSE REGISTRATION

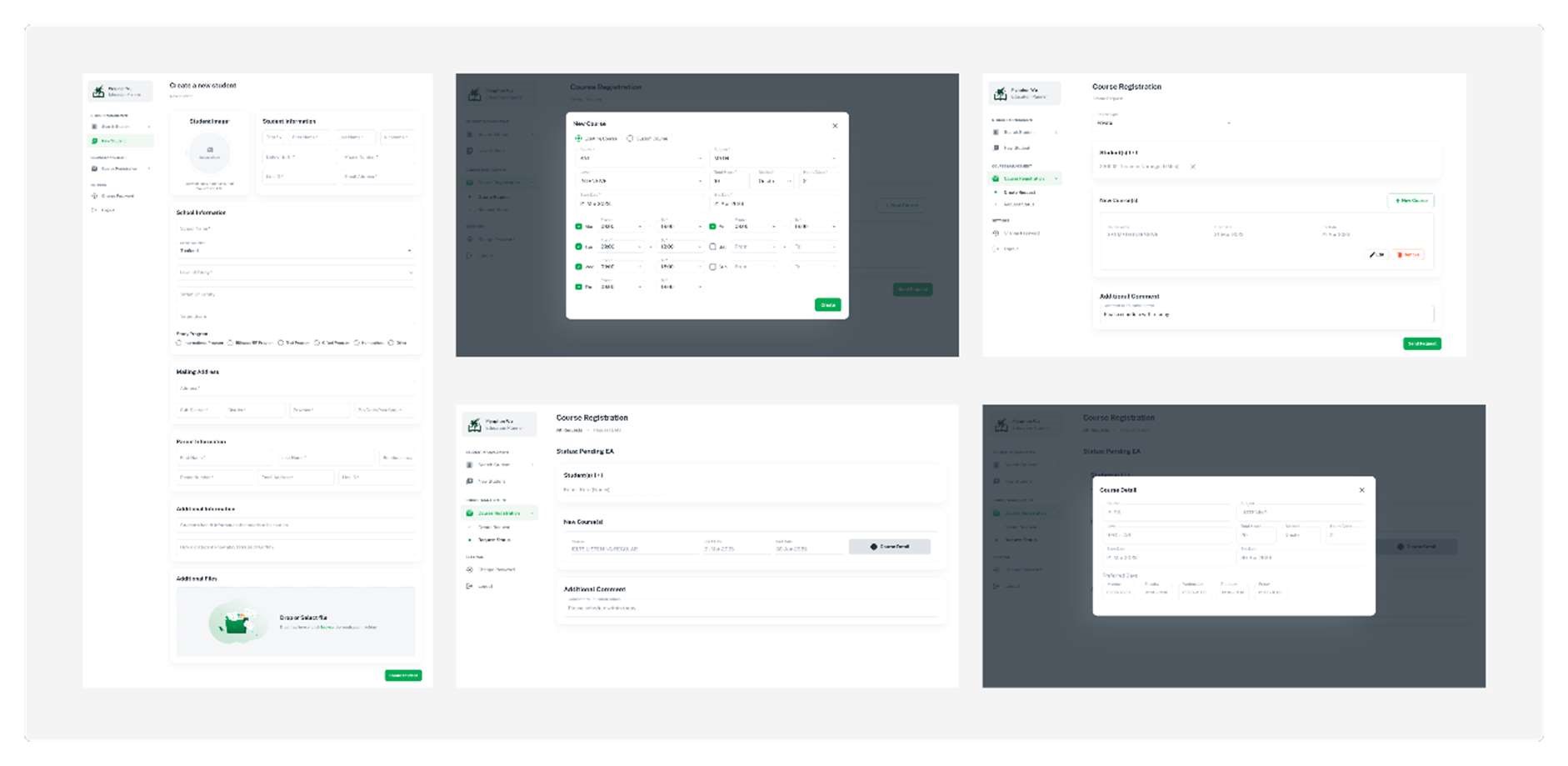


Figure 4.2.1.1: EP Create New Student and Request UI

The initial step in the course registration process involves the Educational Planner (EP) creating a new student account for individuals who do not possess one. Following the account creation, the EP will proceed to generate a request for course registration on behalf of the student. This procedure is crucial in facilitating the enrollment of students into courses and ensures that the necessary records are maintained accurately. The creation of a student account allows for easy identification and tracking of their course registrations, academic progress, and other related records. Therefore, it is imperative that this process is carried out meticulously to guarantee the accuracy and completeness of the records. The figure 4.2.1 depicts the pages of the application that the EP will be using.

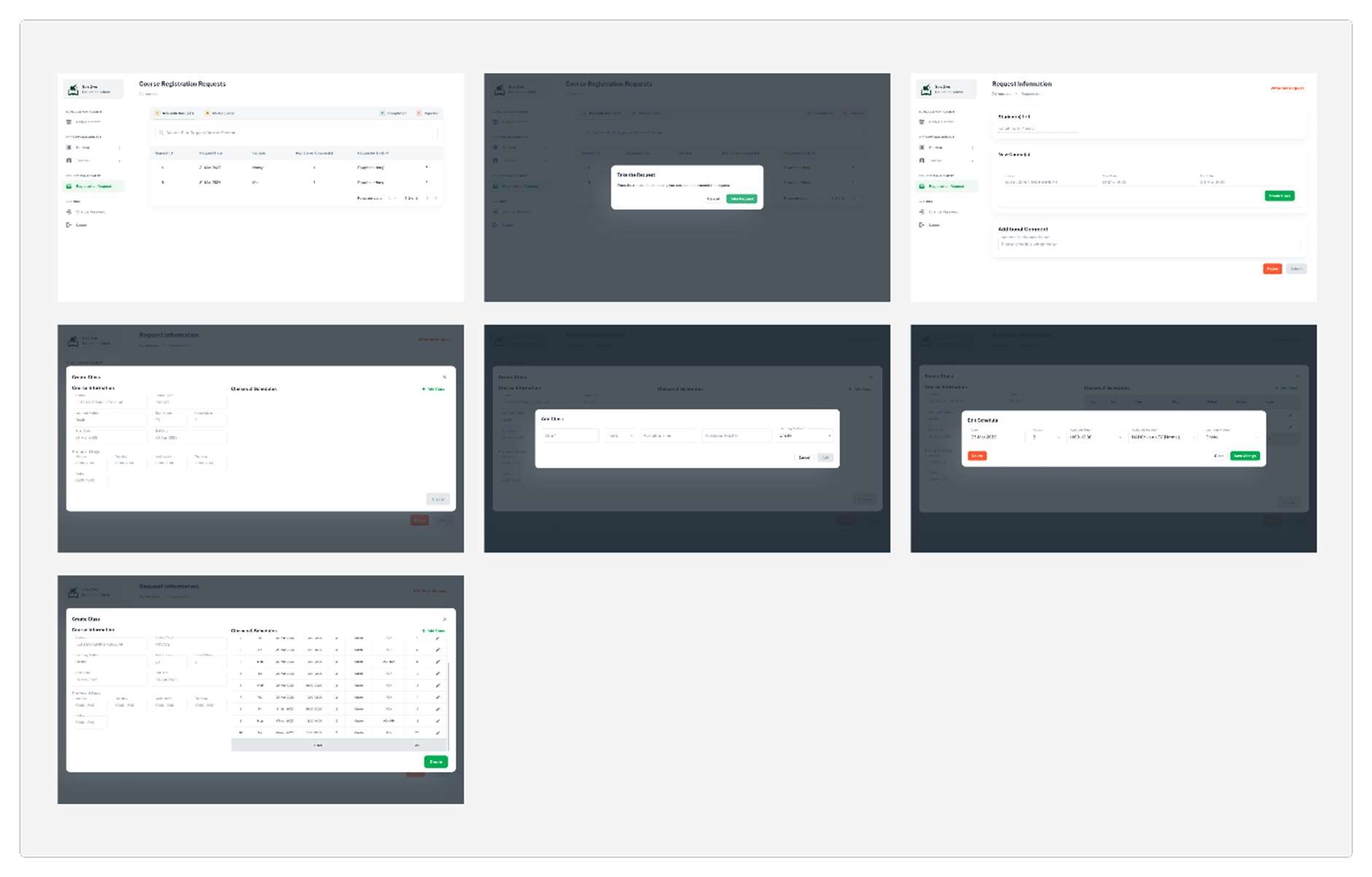


Figure 4.1.1.2: EA Create Request’s Schedules UI

Subsequently, once the request for course registration is generated by the EP, it is forwarded to the Education Administrator (EA) who will oversee the scheduling process. The EA will examine the request and create a schedule by adding multiple classes to the course as specified in the request submitted by the EP. This step is crucial in ensuring that the appropriate courses are made available to the students at suitable times and dates. The EA must ensure that the schedule is meticulously planned and designed to avoid scheduling conflicts, minimize overcrowding, and provide sufficient resources for each class. Additionally, the schedule must align with the academic requirements and guidelines of the institution to ensure that the educational needs of the students are met effectively. Thus, the scheduling process is a crucial component of the course registration process, and it requires careful planning and attention to detail to achieve successful outcomes.

The EA will utilize the pages depicted in Figure 4.1.2 during the course registration process. These pages serve as a tool for the EA to facilitate the scheduling of classes and ensure that the courses are made available to the students as per their academic requirements. The utilization of these pages is essential in streamlining the scheduling process, as they provide a visual representation of the courses and their respective timings. Additionally, the pages also assist in identifying scheduling conflicts and enable the EA to adjust the schedule accordingly to avoid overcrowding and ensure that sufficient resources are allocated to each class. Therefore, the usage of these pages is crucial in ensuring that the course registration process is carried out effectively and efficiently.

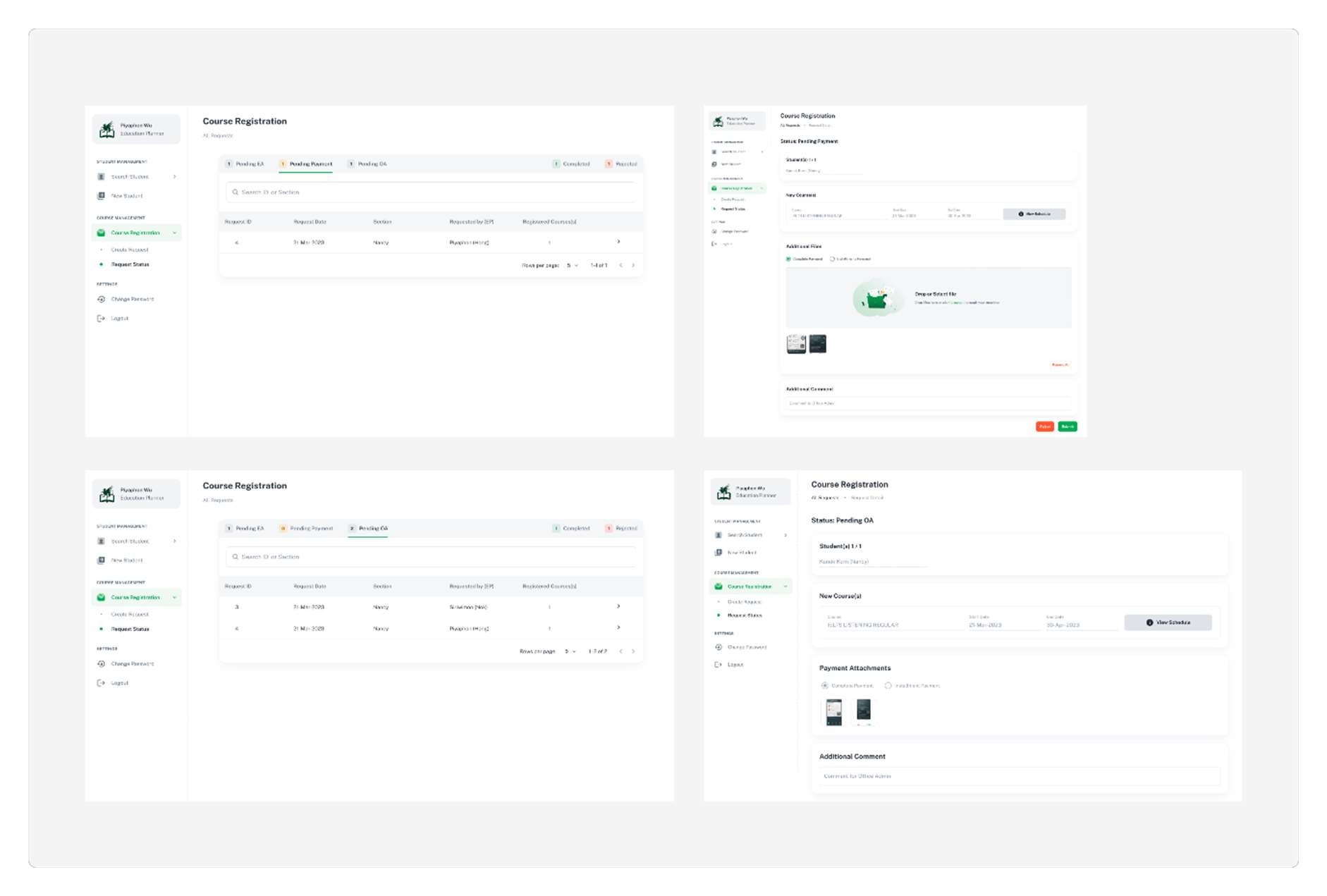


Figure 4.1.1.3: EP Requests Information with Payments UI

Once the EA has completed the scheduling process, the request for course registration will be returned to the EP. The EP will then proceed to collect the necessary payment from the student as specified in the request. This step is crucial in ensuring that the student's enrollment into the course is confirmed and that the necessary financial obligations are met. The EP must ensure that the payment collection process is carried out meticulously to avoid any discrepancies or errors that may cause delays or disruptions in the student's registration process. Additionally, the EP must provide the student with the relevant information regarding the payment procedure and any associated deadlines to ensure that the payment is made promptly. Therefore, the payment collection process is an essential component of the course registration process, and it requires careful planning and execution to ensure that the student's enrollment is confirmed successfully. The pages that the EP will utilize is based on the figure 4.1.3.

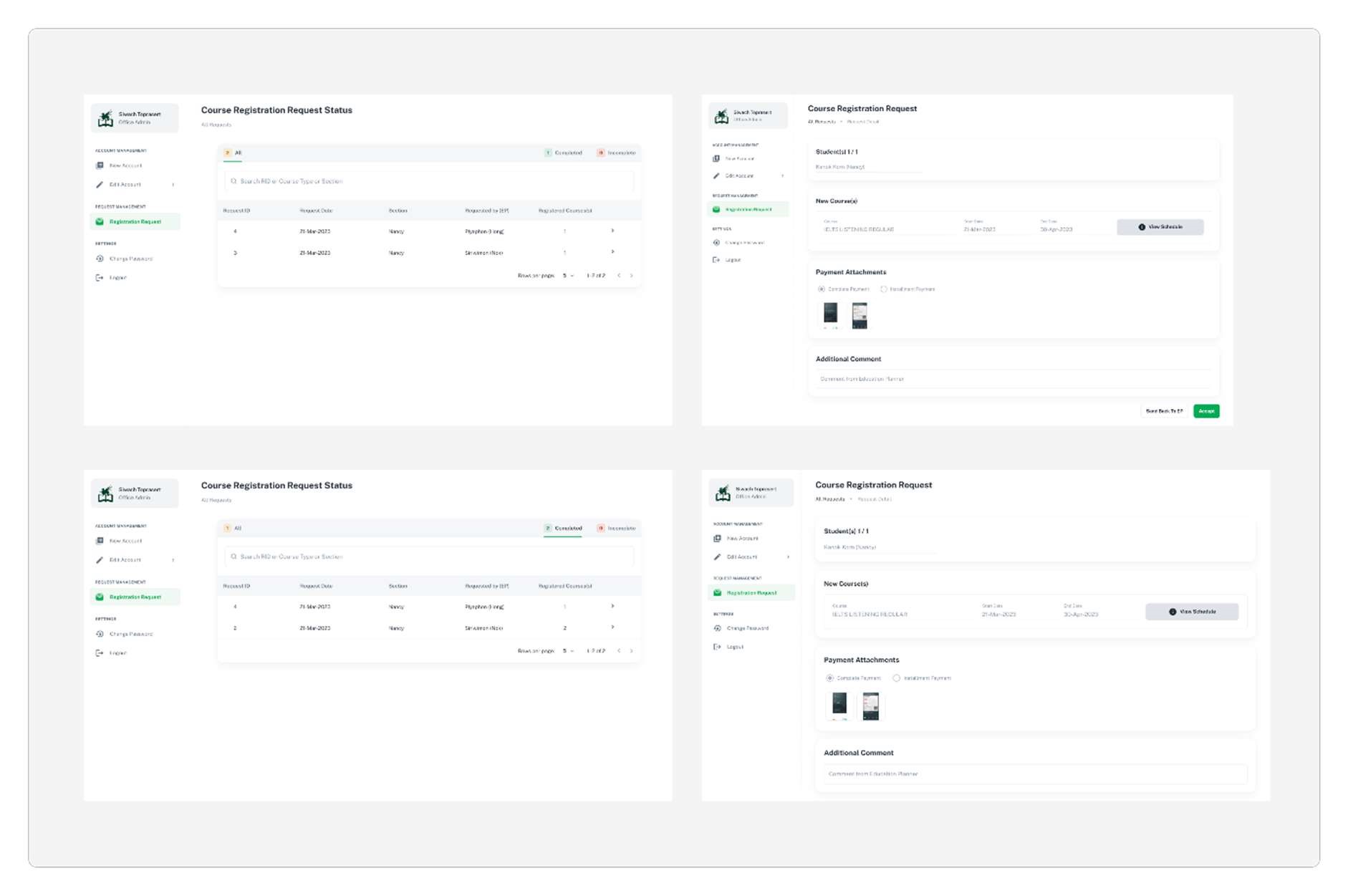


Figure 4.1.1.4: OA Approval UI

Upon successful payment collection by the Educational Provider (EP), the request for course registration will be forwarded to the Office of Administration (OA) for approval. The OA will examine the payment and verify its validity. If the payment is found to be valid, the OA will proceed to approve the course registration request. Following the approval, the EP will be notified that the course registration process has been successful, and the student's enrollment into the course has been confirmed. This step is crucial in ensuring that the course registration process is carried out successfully and that the student's academic requirements are met promptly. Additionally, the OA's approval serves as a confirmation of the accuracy and completeness of the records associated with the course registration process. Therefore, the OA's role is critical in ensuring that the course registration process is carried out effectively and efficiently, and that the necessary records are maintained accurately. The figure 4.1.4 illustrated the pages that the OA will be using.

### 4.2.2 EDIT COURSE

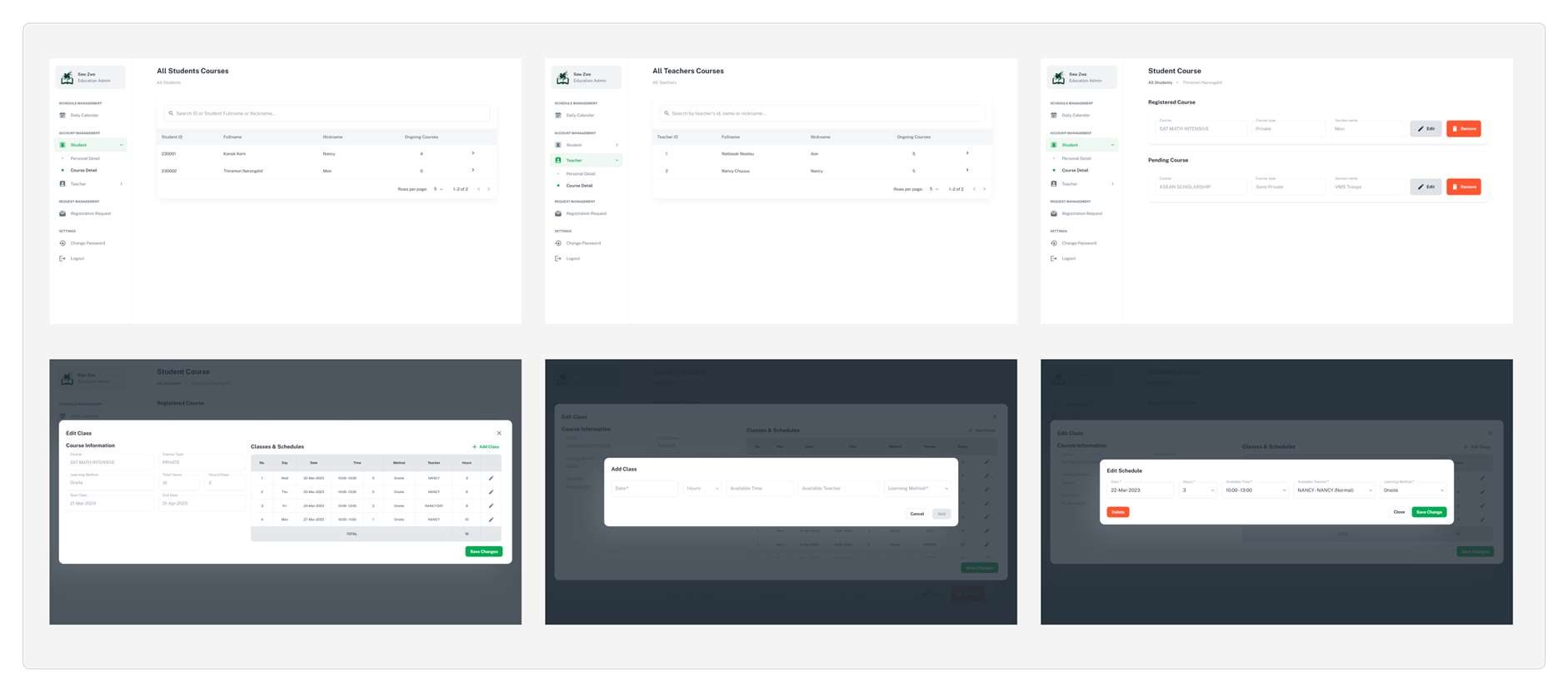


Figure 4.2.2.1: Edit Course UI

The user interface displayed in Figure 4.2.2.1 has been designed exclusively for the use of the EA. This particular function serves the purpose of allowing the EA to make alterations to the information pertaining to a particular course. Such modifications may include changes to class details such as date, time, and any other relevant information related to the course. It should be noted that this feature is not intended for use by any other user apart from the EA, as it requires a certain level of administrative privileges and access.

### 4.2.3 ROOM ADDING

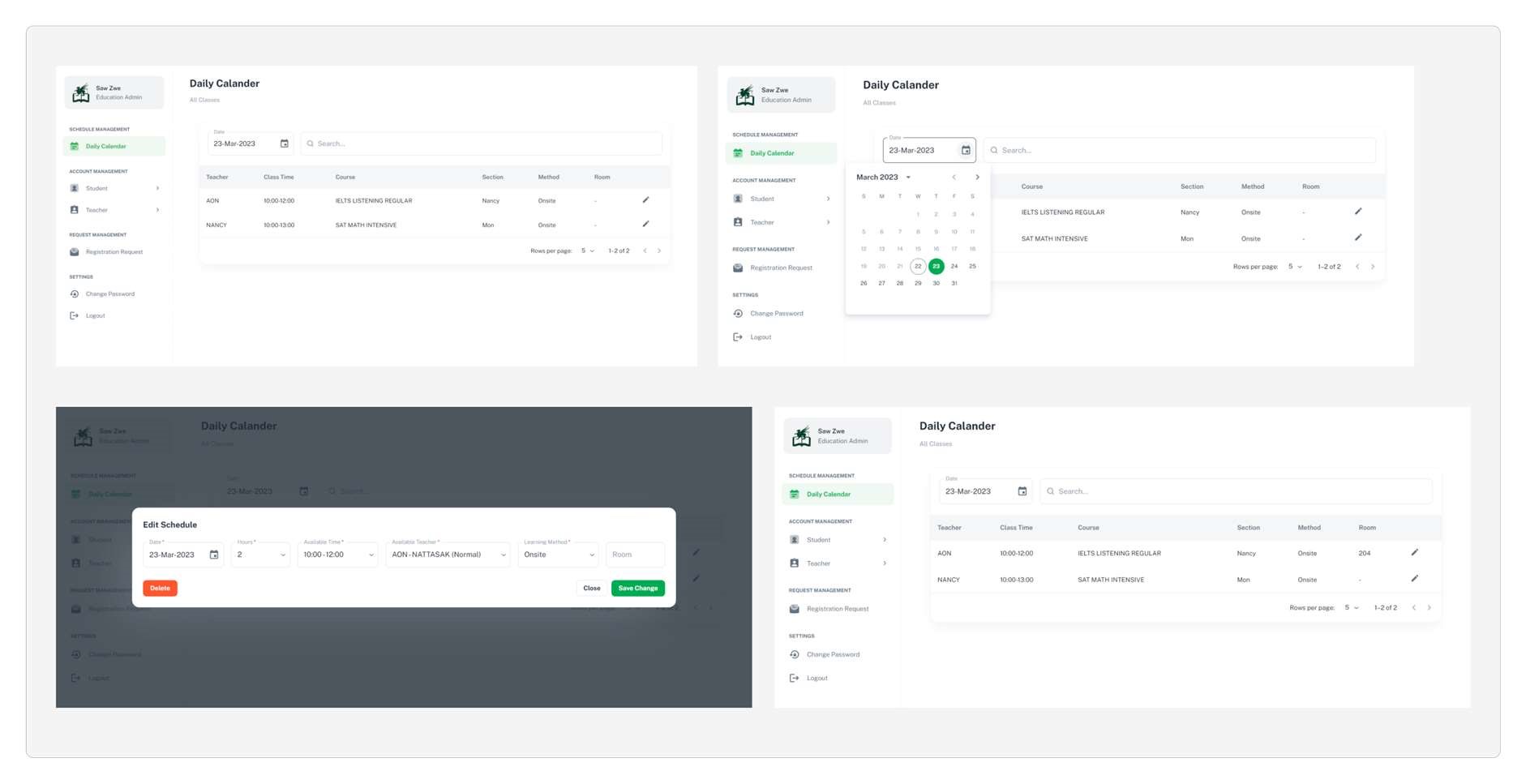


Figure 4.2.3.1: Room Adding UI

The functionality for adding rooms is intended for the purpose of updating classroom information. The House of Griffin has implemented a policy whereby their classrooms are updated on a daily basis, in the evening prior to the commencement of classes the following day. This policy has been put in place to ensure that students are aware of the location of their respective classrooms for upcoming classes. The aforementioned functionality has been designed to facilitate this policy and aid in the dissemination of updated classroom information to the students. It should be noted that this feature is exclusively intended for use by the EA.

The figure 4.2.3.1 illustrates the UI of the room adding function.

### 4.2.4 STUDENT VIEW

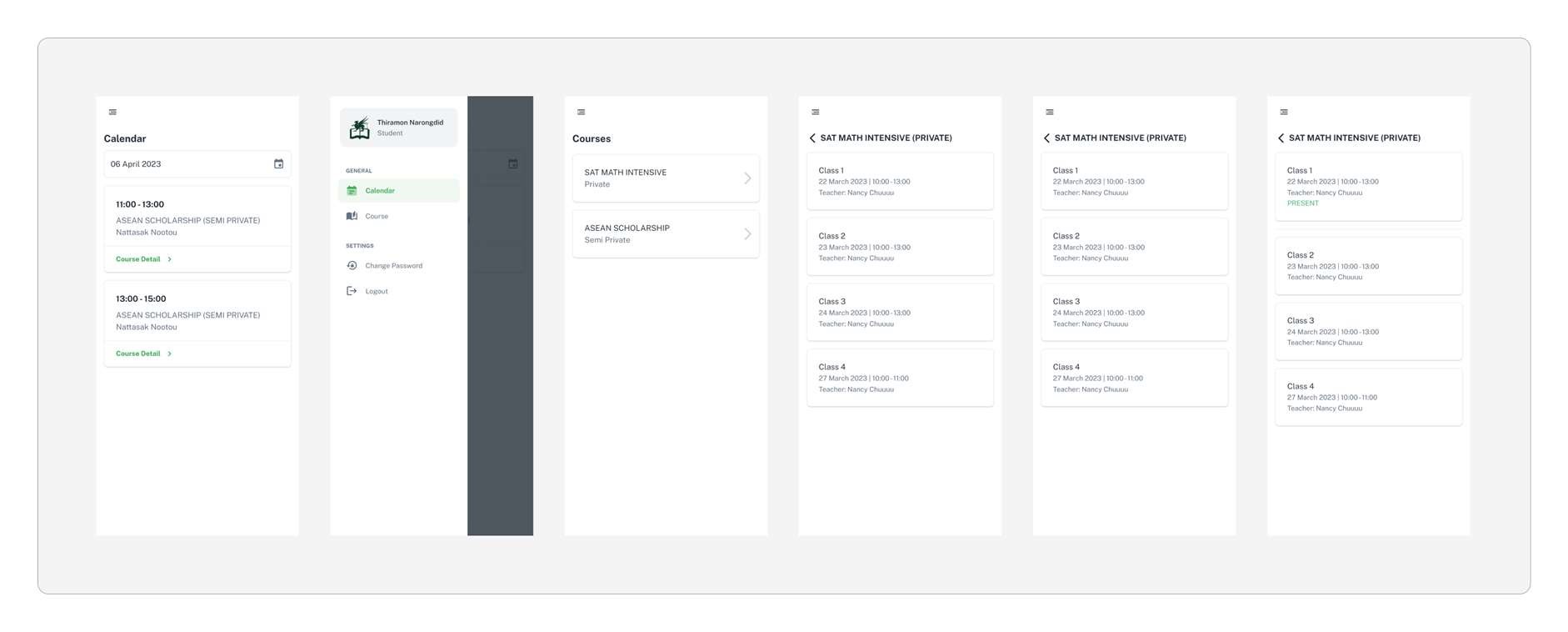


Figure 4.2.4.1: Student Course View UI

Figure 4.2.4.1 depicts the user interface of the student application, which enables students to view the courses they have enrolled in, as well as the associated classes within each respective course. This functionality has been designed to facilitate a seamless and efficient experience for students, allowing them to easily access and manage their course schedules through the House of Griffin tutoring school's web application.

### 4.2.5 TEACHER VIEW & ATTENDANCE CHECK

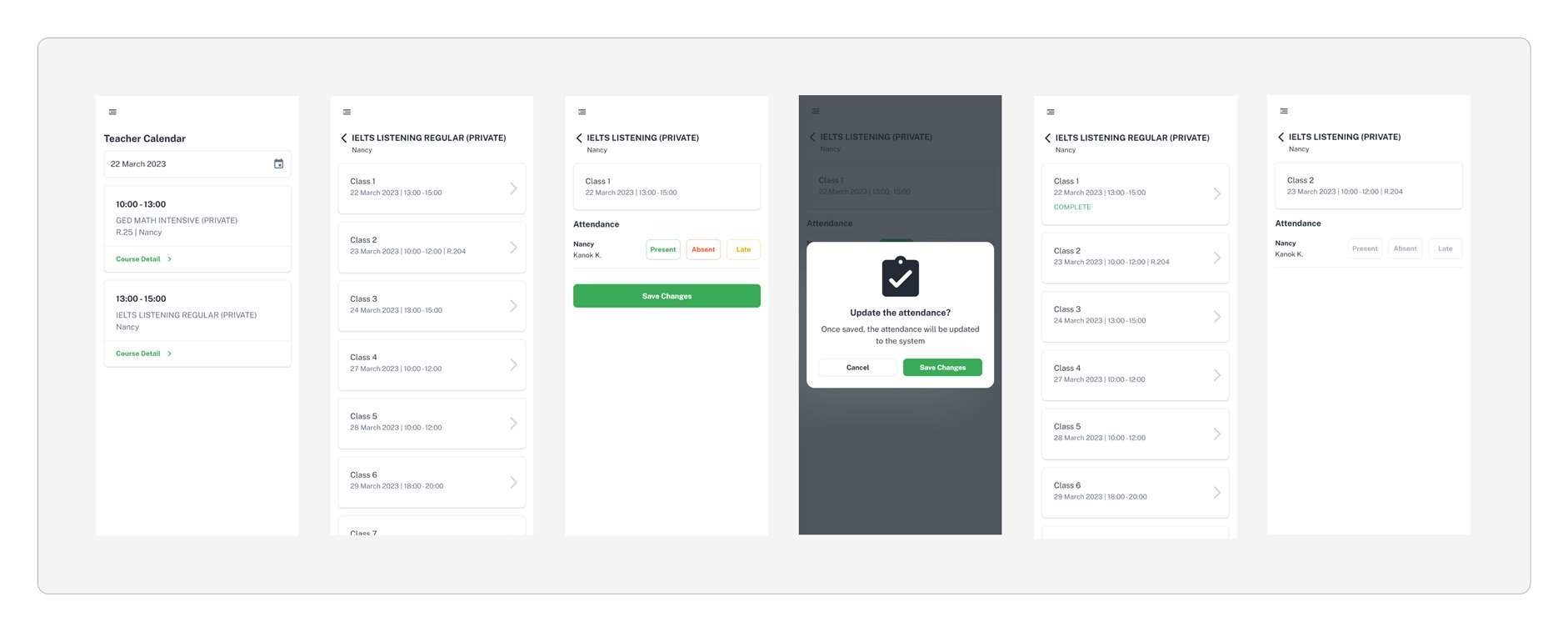


Figure 4.2.5.1: Teacher View and Attendance Check

Similar to the previously discussed functionality, the teacher view feature provides teachers with the ability to view the course they are currently teaching, along with the corresponding classes that are associated with their course. This functionality has been designed to enable teachers to efficiently manage their course schedules and monitor student attendance. Specifically, teachers are able to track attendance by selecting the relevant class for a given day and clicking on it within the web application interface. This feature has been developed with the aim of optimizing the user experience for teachers utilizing the House of Griffin tutoring school's web application.

# 5 SYSTEM COMPARISON

This chapter aims to compare the newly developed system with existing systems that are currently used by HOG or other similar organizations. The purpose of this comparison is to evaluate the strengths and weaknesses of the new system and identify areas of improvement.

|  |  |  |
| --- | --- | --- |
| Aspect | Paper-based System | GriffinEd |
| Data Entry | Manual, time-consuming, prone to errors | Automated, efficient, reduced errors |
| Data Storage | Physical storage, limited space, risk of damage or loss | Digital storage, scalable, secure |
| Access Control | Limited access, security concerns, difficulty in tracking access | Role-based access control, enhanced security, audit trail |
| Cost | High printing and storage costs, frequent paper, and ink refills | Lower printing and storage costs, reduced need for paper and ink |
| Environmental Impact | High carbon footprint due to paper usage and waste | Reduced carbon footprint, environmentally friendly solution |
| Overall Efficiency | Slow and inefficient, prone to errors and delays | Fast and efficient, reduced errors and delays, improved productivity |

Table 5.1: Paper-based System vs. GriffinEd System Comparison

The comparison between the paper-based system and the GriffinEd system is shown in the table 5.1. The developed system provides a significant improvement over the paper-based system in terms of efficiency, accuracy, security, and sustainability. It eliminates the drawbacks of the paper-based system and offers several benefits such as faster data processing, better data analysis, and enhanced collaboration. Additionally, it reduces the cost and environmental impact associated with paper usage and waste.

# 6 CONCLUSION

In conclusion, the development of GriffinEd presents an important solution to meet the needs of House of Griffin. These web applications offer a comprehensive platform that enhances communication and collaboration between students, teachers, and staff members, while streamlining administrative tasks and improving overall efficiency and productivity.

The student and teacher application provides an interactive platform that allows students to easily access their timetable virtually. Teachers, in turn, can efficiently manage their courses and check the student’s attendance. Meanwhile, the staff application provides a centralized platform that allows administrators to manage school-wide activities, such as enrollment and scheduling.

Through personalized learning tools and interactive course materials, these web applications also improve the overall learning experience for students. The scalability and customizability of the platform further ensure that it can meet the needs of different educational institutions, while its security and reliability guarantee that sensitive data is protected, and continuous operation is maintained.

Overall, the development of GriffinEd represents an important and valuable contribution to the education industry. By meeting the needs of students, teachers, and staff members, these web applications provide a comprehensive and effective solution to improve communication, streamline administrative tasks, and enhance overall efficiency and productivity within educational institutions.

# REFERENCES

1. Indig, K. (2021, April 15). Client-Side vs. Server-Side Rendering: What's Best for SEO? [Blog post]. Botify. Retrieved From https://www.botify.com/blog/client-side-server-siderendering-seo.
2. Kocherhin, O. (2020, December 7). Client-side vs. Server-side Pre-rendering. Toptal.

Retrieved from https://www.toptal.com/front-end/client-side-vs-server-side-pre-rendering.

1. UXPin. (2021, September 13). What is MUI? [Blog post]. UXPin. Retrieved from https://www.uxpin.com/studio/blog/what-is-mui/
2. Gaurav, A. (2021, October 29). Firebase Authentication in .NET Core Web API. LinkedIn. Retrieved from https://www.linkedin.com/pulse/firebase-authentication-net-coreweb-api-aseem-gaurav/
3. K., D. (2021, April 20). Using Firebase JWT Tokens in ASP.NET Core (.NET 5). Medium. Retrieved from https://dominique-k.medium.com/using-firebase-jwt-tokens-in-aspnet-core-net-5-834c43d4aa00
4. Coursera. (n.d.). AWS vs Azure vs Google Cloud. Retrieved from https://www.coursera.org/articles/aws-vs-azure-vs-google-cloud
5. Karki, S. (2022, January 7). Build Minimal APIs In .NET 7 Using Entity Framework Core 7. Retrieved from https://www.c-sharpcorner.com/article/build-minimal-apis-in-net-7-usingentity-framework-core-7/
6. Kogut, B., & Zander, U. (2004). Transforming knowledge for new forms of value creation: Conceptualizing the transformation process. Academy of Management Learning &

Education, 3(4), 401-417. https://doi.org/10.5465/amle.2004.12436815

1. Zhu, X., Li, Y., Li, Q., & Li, W. (2014). A brief review of text mining. Chinese Journal of

Library and Information Science, 7(1), 63-77. https://doi.org/10.1080/18770568.2013.878788

1. Intellipaat. (n.d.). Azure Web Apps. Retrieved from

https://intellipaat.com/blog/tutorial/microsoft-azure-tutorial/azure-web-apps/