WEB BASED CLASS SCHEDULE MANAGEMENT SYSTEM

This project is submitted to the Department of Computer Science and Engineering, Dhaka International University, in partial fulfillment to the requirements of Bachelor of Science (B. Sc.) in Computer Science and Engineering (CSE).

Submitted by

NAME	REG. NO	ROLL NO
DIPTA CHANDRA BANIK	CS-D-55-19-110583	14
ADNAN KARIM	CS-D-55-19-110583	17
FABIHA MAHMUD	CS-D-55-19-110592	18
MST. ISRAT JAHAN ILA	CS-D-55-19-110855	35
APU PAL	CS-D-55-19-115695-CT	51

Batch: 55th (1st shift), Session (2019-2020)

Supervised by

Md. Nur-A-Alam
Lecturer
Dept. of Computer Science and Engineering
Dhaka International University

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
FACULTY OF SCIENCE AND ENGINEERING
DHAKA INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH

OCTOBER-2023

SUPERVISOR'S STATEMENT

This is to certify that the project paper entitled as **"WEB BASED CLASS SCHEDULE MANAGEMENT SYSTEM"** submitted by Dipta Chandra Banik, Roll No:- 14; Adnan Karim, Roll No:- 17; Fabiha Mahmud, Roll No:- 18; Mst. Israt Jahan Ila, Roll No:- 35; Apu Pal, Roll No:-51; has been carried out under my supervision. This project has been prepared in partial fulfillment of the requirement for the Degree of B.Sc. in Computer Science & Engineering, Department of Computer Science & Engineering, Dhaka International University, Dhaka, Bangladesh.

Supervisor's Signature

Date: ……

.....…...……

Md. Nur-A-Alam
Lecturer
Dept. of Computer Science & Engineering
Dhaka International University

APPROVAL

The project report as **"WEB BASED CLASS SCHEDULE MANAGEMENT SYSTEM"** submitted by Dipta Chandra Banik, Adnan Karim, Fabiha Mahmud, Mst. Israt Jahan Ila, Apu Pal to the Department of Computer Science & Engineering, Dhaka International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents.

Board of Honorable Examiners

1. Chairman .....

Prof. Dr. A. T. M. Mahbubur Rahman
Chairman,
Dept. of Computer Science and Engineering
Dean,
Faculty of Science and Engineering,
Dhaka International University

2. Internal Member .....

Associate Prof. Md. Abdul Based
Chairman,
Dept. of Electrical, Electronics & Telecommunication Engineering,
Dhaka International University

.....

Mst. Jahanara Akhter
Associate Professor,
Dept. of Computer Science and Engineering
Faculty of Science and Engineering,
Dhaka International University

3. Supervisor and Member .....

Md. Nur-A-Alam
Lecturer,
Dept. of Computer Science and Engineering,
Dhaka International University

4. External Member ……

Dr. Md. Manowarul Islam
Associate Professor,
Dept. of Computer Science and Engineering,
Jagannath University

DECLARATION

We hereby declare that; this project has been carried out by us and it has been submitted for the award of the B.Sc. degree. We also certify that this project was prepared by us for the purpose of fulfillment of the requirements for the Bachelor of Science (B.Sc.) in Computer Science & Engineering.

Authors Signature

.....

Dipta Chandra Banik
B.Sc. In CSE, Roll No: 14
Reg. No: 110583
Batch: D-55th, Session: 19-20
Dhaka International University

.....

Adnan Karim
B.Sc. In CSE, Roll No: 17
Reg. No: 110589
Batch: D-55th, Session: 19-20
Dhaka International University

.....

Fabiha Mahmud
B.Sc. In CSE, Roll No: 18
Reg. No: 110592
Batch: 55th, Session: 19-20
Dhaka International University

.....

Mst. Israt Jahan Ila
B.Sc. In CSE, Roll No: 35
Reg. No: 110855
Batch: 55th, Session: 19-20
Dhaka International University

.....

Apu Pal
B.Sc. In CSE, Roll No: 51
Reg. No: 115695-CT
Batch: 55th, Session: 19-20
Dhaka International University

Supervisor's Signature

Date: ……

.....

Md. Nur-A-Alam
Lecturer
Dept. of Computer Science and Engineering
Dhaka International University

ABSTRACT

This document proposes a Class scheduling system which is a web-based application that manages all daily activities of an educational institution. The system can keep track of administrators adding, editing, searching, viewing and deleting of class schedule. This is the class scheduling process for administrators, teachers, and students within our educational institutions. This comprehensive system offers tailored and user-friendly interfaces for three distinct user roles, ensuring seamless access and optimized user experiences throughout the scheduling journey. Administrators stand to gain the most from this innovative platform as it simplifies the traditionally complex task of creating and managing class schedules. This not only reduces the administrative burden but also empowers administrators to make data-driven decisions, ultimately leading to more efficient academic operations. For teachers, the Online Class Schedule Generator provides a user-friendly environment. Students, too, benefit significantly from this system. They can effortlessly access their class schedules and relevant information, making it easier to plan their academic activities and extracurricular commitments. Technologically, the system is built using a robust stack that includes HTML, Tailwind CSS (CSS Framework) advanced PHP, PHPMailer and dompdf (library for generating PDF schedules), JavaScript, ajax and jQuery (library for enhanced user interaction). The backend leverages the power of MySQL for efficient data storage and retrieval, ensuring seamless performance even under heavy user loads. These developments will further elevate the scheduling process, catering to the evolving needs of educational institutions and ensuring that the Web based Class Scheduling System remains a valuable solution for efficient scheduling, optimal resource utilization, and enhanced time management within the educational sector.

ACKNOWLEDGEMENTS

We would like to pay our gratitude to the Almighty Allah who created us with not only the ability to design and program this system but also the power of practice.

We would also like to express our sincere thanks to our respected supervisor **Nur-A-Alam, Lecturer**, Department of CSE, Dhaka International University for his continuous encouragement, motivation and professional guidance during the work of this project.

We would like to thank all the faculty members for their valuable time spent on requirements analysis and evaluation of the project work.

We would like to express our sincere and cordial gratitude to the people those who have supported us directly, provided mental encouragement. We are also thankful to our family and friends who have contributed directly or indirectly the development word and its associated activities.

We warmly thank **Prof. Dr. A.T.M. Mahbubur Rahman,** Chairman, Department of Computer Science and Engineering and Dean (Acting), Faculty of Science and Engineering, Dhaka International University for their valuable advice and moral support. Their extensive discussion around work and interesting exploration in operations has been very helpful for this study.

Finally, we would like to dedicate this project to our teachers for their love, encouragement and professional guidance throughout the project.

DEDICATION

Dedicated to

Our Parents

&

Teachers

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 INTRODUCTION

1.1 INTRODUCTION

In today's rapidly evolving educational landscape, effective academic management hinges significantly on the process of scheduling classes. Educational institutions face the daunting task of efficiently arranging class schedules, a crucial aspect of ensuring a smooth academic operation. In response to these challenges, we introduce the Web Based Class Schedule Management System for Educational Institutions—a web-based platform designed to redefine and optimize the class scheduling process, primarily leveraging the power of database management through MySQL.

Unlike automation systems that rely on complex algorithms and artificial intelligence, our approach centers on harnessing the capabilities of a robust database system. This system has been meticulously crafted to cater to the unique needs of administrators, teachers, and students within educational institutions. It offers tailored interfaces that enhance their scheduling experiences and contribute to more efficient academic operations, all while maintaining a database-driven foundation.

Administrators bear the weight of orchestrating class schedules that align with faculty availability, classroom resources, and student demands. The Web Based Class Schedule Management System simplifies this complex task by offering a streamlined solution, rooted in efficient data management. It reduces the administrative load without resorting to automated algorithms. Instead, it empowers administrators with the ability to efficiently store, manage, and retrieve data from MySQL, providing them with accurate and up-to-date information. This allows administrators to make informed decisions, ultimately leading to more efficient academic operations, resource allocation, and institutional excellence.

For teachers, the system still provides a user-friendly environment for accessing class schedules and related information. While automation may not be the driving force behind the scheduling process, the database-centric approach ensures that the information presented is accurate and reliable.

Students also benefit significantly from this system. They can effortlessly access their class schedules and relevant information, enabling them to plan their academic activities and extracurricular commitments with confidence. The emphasis here is on data accuracy and real-time updates, ensuring students have the information they need when they need it.

Technologically, the system is built upon a foundation that prominently features MySQL—a robust and proven database management system. The use of MySQL ensures that the platform remains stable, efficient, and reliable, even when handling large volumes of data.

Our Web Based Class Schedule Management System for Educational Institutions is a database-driven solution that simplifies the class scheduling process. It stands as a testament to the power of efficient data management in reshaping education. As we delve deeper into the subsequent chapters of this paper, we will explore the intricacies of this transformative platform, examining its architecture, functionality, and its tangible impact on the various

stakeholders within the educational sector.

1.2 OBJECTIVES

The primary objectives of the Web Based Class Schedule Management System for Educational Institutions are as follows:

- **Streamline Class Scheduling:** The core aim of this project is to simplify and streamline the class scheduling process within educational institutions. By leveraging database management with MySQL, we seek to provide a user-friendly platform that allows administrators to create, manage, and maintain class schedules efficiently.

- **Enhance Administrative Efficiency:** To alleviate the administrative burden associated with class scheduling, our objective is to empower administrators with a reliable system that facilitates data entry, retrieval, and updates. This efficiency boost enables administrators to focus on higher-level strategic tasks and decision-making.

- **Improve Data Accuracy:** Through meticulous data management within the MySQL database, our project aims to improve data accuracy in class schedules. By maintaining up-to-date and error-free information, we ensure that scheduling conflicts are minimized and that all stakeholders have access to reliable scheduling data.

- **Facilitate Teacher Engagement:** While not automated, the system aims to provide teachers with easy access to their class schedules and related information, thus encouraging greater engagement with their assigned classes and responsibilities.

- **Empower Student Planning:** The project seeks to empower students by providing them with effortless access to their class schedules and pertinent information. This empowers students to effectively plan their academic activities and extracurricular commitments.

- **Strengthen Resource Allocation:** By maintaining accurate data in the MySQL database, the system contributes to improved resource allocation within educational institutions. This includes optimizing the allocation of classrooms, faculty, and other resources, thereby enhancing institutional efficiency and cost-effectiveness.

- **Adaptability and Long-term Value:** The system is designed with adaptability in mind to cater to the evolving needs of educational institutions. It aims to remain a valuable solution over the long term, accommodating changes in curriculum, faculty, and student enrollment, and adapting to the ever-evolving educational landscape.

1.3 BENEFITS

The deployment of a Web Based Class Schedule Management System unveils an array of tangible benefits, each contributing to a more efficient and user-centric educational

environment.

1.3.1 Efficient Class Scheduling:

The Web Based Class Schedule Management System is a game-changer in terms of scheduling efficiency. It automates the traditionally complex process of creating class schedules, allowing administrators to create comprehensive schedules with ease. This translates into significant time savings, freeing up administrative staff to focus on other critical tasks.

1.3.2 Time and Resource Savings:

By reducing the manual effort required for class scheduling, the system saves valuable time and resources for administrators, enabling them to focus on other critical tasks.

1.3.3 Reduced Scheduling Conflicts:

Scheduling conflicts are a common headache in educational institutions. By centralizing scheduling data in the MySQL database and providing real-time updates, the system helps minimize conflicts. Teachers can confidently plan their classes without worrying about overlaps or changes.

1.3.4 Enhanced Data Accuracy:

Data accuracy is paramount in the educational sector. The system ensures that class schedules are kept accurate and up-to-date, reducing errors and confusion. Teachers and students can rely on the system to provide reliable information, ultimately leading to smoother academic operations.

1.3.5 Improved Resource Allocation:

Efficient scheduling directly impacts resource allocation. With the Web Based Class Schedule Management System, institutions can optimize classroom usage, allocate faculty more effectively, and distribute other resources based on real-time demand. This leads to cost savings and a more efficient use of available resources.

1.3.6 Empowerment of Administrators:

The system empowers administrators by providing them with a robust database-driven platform. They can efficiently manage scheduling data, monitor resource utilization, and make informed decisions based on accurate information. This newfound power enhances their role as strategic leaders within the institution.

1.3.7 User-Friendly Interfaces:

Tailored user interfaces for administrators, teachers, and students ensure that each stakeholder has an intuitive and accessible experience. The interfaces are designed to be user-friendly, requiring minimal training and ensuring a smooth interaction with the system.

1.3.8 Student Planning:

For students, having easy access to their class schedules and related information is a significant advantage. It enables them to plan their academic activities, study sessions, and extracurricular commitments more effectively. This proactive approach to time management contributes to academic success.

1.3.9 Real-Time Updates:

The system's real-time updates ensure that everyone has access to the most current scheduling information. This dynamic nature of the system helps in adapting to unexpected changes, such as last-minute class cancellations or room reassignments.

1.3.10 Adaptability:

Educational institutions are dynamic environments where change is constant. The Web Based Class Schedule Management System is designed to adapt to these changes. It remains a valuable tool as institutions grow, modify curricula, and experience shifts in enrollment, ensuring long-term relevance and value.

1.3.11 Data-Driven Decision-Making:

The system promotes data-driven decision-making by providing accurate and comprehensive scheduling data. This empowers institutions to analyze trends, optimize scheduling strategies, and align resources with educational goals.

1.3.12 Enhanced Academic Experience:

Ultimately, the benefits of the system contribute to an enhanced academic experience for both teachers and students. Teachers can focus on delivering quality education, and students can engage more actively in their studies without the distraction of scheduling uncertainties.

In conclusion, the Web Based Class Schedule Management System for Educational Institutions offers a holistic set of benefits that span efficiency, accuracy, empowerment, and adaptability. Its user-centric design and data-driven approach are pivotal in transforming the scheduling landscape in educational institutions, ultimately resulting in improved academic operations and outcomes.

1.4 AIM

The aim of the Web Based Class Schedule Management System for Educational Institutions is to provide a comprehensive and user-centric solution that addresses the challenges and complexities associated with class scheduling in educational institutions. This project aims to:

- ✓ **Simplify Class Scheduling:** The primary aim is to simplify the class scheduling process for administrators, ensuring that they can create and manage schedules efficiently without the need for complex manual calculations or labor-intensive processes.

- ✓ **Enhance Administrative Efficiency**: This project seeks to enhance administrative efficiency by reducing the time and effort required for class scheduling. By providing administrators with a user-friendly platform, it empowers them to allocate resources, assign faculty, and create schedules more effectively.

- ✓ **Improve Data Accuracy:** An important aim is to improve the accuracy of class schedules. By maintaining a centralized and up-to-date database of scheduling information, the project minimizes errors, scheduling conflicts, and confusion among stakeholders.

- ✓ **Empower Stakeholders:** The project aims to empower all stakeholders within educational institutions, including administrators, teachers, and students. It provides them with easy access to scheduling data, enabling them to make informed decisions and plan their activities effectively.

- ✓ **Optimize Resource Allocation:** Efficient scheduling directly impacts resource allocation. The aim is to optimize the allocation of classrooms, faculty, and other resources based on real-time demand, leading to cost savings and improved resource utilization.

- ✓ **Promote Data-Driven Decision-Making:** By offering accurate and comprehensive scheduling data, the project encourages data-driven decision-making within educational institutions. It empowers administrators to analyze trends, identify scheduling inefficiencies, and align resources with academic goals.

- ✓ **Facilitate Adaptability:** Educational institutions are dynamic environments that experience changes in curriculum, faculty availability, and student enrollment. The project aims to remain adaptable, ensuring that it can accommodate these changes and continue to provide value over the long term.

- ✓ **Enhance the Academic Experience**: Ultimately, the aim is to enhance the academic experience for both teachers and students. By reducing scheduling conflicts, providing real-time updates, and offering user-friendly interfaces, the

project creates an environment conducive to effective teaching and learning.

1.5 FEATURES

The main objective of the proposed Web Based Class Schedule Management System is to computerize the existing system and reduce manpower and time consumption. It provides the following features-

- **Batch and Semester Selection:**

  Users can select a specific batch (group of students) and semester (academic term) for which they want to generate a class schedule.

- **Schedule Generation:**

  The system process data of creating a class schedule based on predefined criteria and rules. This includes factors such as course availability, teacher preferences, and room capacity.

- **Teacher and Room Availability:**

  The system ensures that teachers and rooms are available during the specified time slots to avoid scheduling conflicts. It takes into account the availability of both resources when generating schedules.

- **Customization:**

  Admin have the ability to customize the schedule generation process to accommodate various factors. This may include adjusting course credits, specifying preferred time slots for certain courses, or incorporating other constraints or preferences.

- **Interactive Interface:**

  The system provides a user-friendly web interface that allows users to input batch and semester details easily. Users can also view the generated schedules and make adjustments if needed through this interface.

These features collectively create an efficient and user-friendly system for generating class schedules in an educational institution. Such a system can help optimize resource utilization, reduce conflicts, and improve the overall scheduling process for both students and faculty.

1.6 SOME ADVANTAGES OF WEB BASED ClASS SCHEDULE

- Accessibility and Convenience
- User-Friendly Interfaces

- Real-Time Updates
- Paperless Operation
- Centralized Data Management
- Customization and Flexibility
- Data Security
- Cost-Efficiency
- Scalability
- Remote Learning Support
- Simplicity

1.7 SOME DISADVANTAGES OF WEB BASED CLASS SCHEDULE

- Dependence on Internet Connectivity
- Data Security Concerns
- Accessibility Challenges
- Limited Offline Access
- Data Privacy and Compliance
- Maintenance and Updates

1.8 OVERVIEW OF THE PROJECT PAPER

The subsequent chapters of this project will provide a comprehensive exploration of the Web Based Class Schedule Management System. In Chapter 2, we will delve into the project's design, outlining its architectural framework and technological underpinnings. Chapter 3 will illuminate the workflow and methodology employed in the development process. Moving forward, Chapter 4 will dive into the implementation of the database, shedding light on its structure and functionality. Chapter 5 will offer an encompassing overview of the entire project, providing a holistic understanding of its components and how they work together. As we approach the project's conclusion, Chapter 6 will summarize the key findings and insights gathered throughout the project's development. It will also outline potential future directions and enhancements for this innovative scheduling solution. Lastly, in Chapter 7, we will provide references and citations to acknowledge the sources and contributions that have informed and guided this project's development.

Chapter 2 PROJECT DESIGN

2.1 SYSTEM ANALYSIS AND DESIGN

In this pivotal section of the project, we embark on a comprehensive journey of system analysis and design, crucial steps in the development of the Web Based Class Schedule Management System. This section serves as the bedrock for understanding the intricate details of our project.

❖ **System Analysis: Understanding the Requirements and Objectives**

In the realm of system analysis, we initiate our exploration by meticulously dissecting the existing educational landscape. The primary objective here is to gain a profound understanding of the unique requirements, objectives, and constraints that govern class scheduling within educational institutions.

- **Requirements Elicitation:**

  We engage in a systematic process of requirements elicitation, actively involving stakeholders, including administrators, teachers, and students. Through interviews, surveys, and data collection, we extract valuable insights into their needs and expectations.

- **Objective Alignment:**

  We align the project's objectives with the overarching goals of educational institutions. This alignment ensures that our scheduling solution contributes directly to the efficiency and excellence of academic operations.

- **Constraint Identification**:

  With a keen eye, we identify and document any constraints, be they technological, budgetary, or logistical. Acknowledging these constraints allows us to design a solution that is not only effective but also feasible within the defined parameters.

❖ **System Design: Blueprinting the High-Level Architecture**

As we transition to the design phase, we embark on the creative process of blueprinting the high-level architecture of the Web Based Class Schedule Management System. This phase is where our vision takes shape, and the foundations of the system come to life.

- **Architectural Framework:**

  We present a robust architectural framework that outlines the structure of the system. This includes defining the key components, modules, and their interconnections.

- **Data Flow:**

An integral aspect of our design is the visualization of data flow within the system. We illustrate how scheduling data moves through various stages, ensuring accuracy, reliability, and accessibility.

- **System Components:**

  In this section, we delve into the specific components that constitute the Web Based Class Scheduling System. We elucidate their functionalities and roles within the system's ecosystem.

By engaging in this meticulous process of system analysis and design, we set the stage for the development of a scheduling solution that not only meets the needs of educational institutions but also enhances their academic operations. The subsequent chapters will build upon these foundations, taking you deeper into the intricacies of our project.

2.2 ANALYSIS OF THE EXSISTING SYSTEM

In this critical section, we embark on a thorough analysis of the existing class scheduling methods and processes that have long been in use within educational institutions. It serves as a foundational step in understanding the motivations and necessity for the Web Based Class Scheduling System project.

❖ **Existing System Overview: Evaluating Limitations and Shortcomings**

We commence our exploration by casting a discerning eye upon the traditional class scheduling methods and processes that have historically prevailed in educational institutions. This section aims to provide a comprehensive overview of the prevailing system while critically assessing its limitations and shortcomings.

- **Legacy Systems:**

  We delve into the historical methods and systems that have been relied upon for class scheduling. This includes manual scheduling, spreadsheet-based approaches, or rudimentary software solutions.

- **Identifying Gaps**:

  We meticulously identify the gaps in the existing system, shedding light on areas where inefficiencies, errors, and challenges have become prevalent. These gaps may encompass data inaccuracies, scheduling conflicts, and administrative burdens.

- **Operational Inefficiencies:**

  We illuminate the operational inefficiencies that stem from the limitations of the traditional system. These inefficiencies often lead to suboptimal resource allocation,

reduced adaptability, and increased administrative workload.

- **Challenges and Issues: Highlighting Administrative, Teacher, and Student Concerns**

Building upon our understanding of the existing system's limitations, we proceed to unveil the specific challenges and issues faced by key stakeholders within educational institutions, including administrators, teachers, and students.

- **Administrative Burden:**

  We delineate the administrative burden placed upon institutions when managing class schedules manually or through outdated systems. This burden extends to data entry, conflict resolution, and resource allocation.

- **Teacher Frustrations:**

  We delve into the frustrations experienced by teachers who contend with scheduling conflicts, ambiguities, and frequent changes that disrupt their academic planning and teaching activities.

- **Student Concerns:**

  We empathize with the concerns of students who encounter scheduling challenges that hinder their ability to plan their academic activities and extracurricular commitments effectively.

Through this comprehensive analysis of the existing system, we aim to shed light on the compelling need for a modern, efficient, and user-friendly class scheduling solution. The subsequent chapters will unveil the blueprint for the Web Based Class Schedule Management System, addressing these challenges and providing innovative solutions for educational institutions.

2.3 SDLC (SOFTWARE DEVELOPMENT LILE-CYCLE)

In the dynamic field of software development, the choice of a suitable Software Development Life Cycle (SDLC) model is paramount to the success and efficiency of any project. This section provides an insightful overview of the SDLC model carefully chosen for the development of our Web Based Class Schedule Management System. We explore the model's inherent phases and methodologies that guide our project through its transformative journey. There are following six phases in every Software development life cycle model and showing The Figure **‘Figure 2.3’** represent SDLC model.

- Planning
- Requirement gathering and analysis

- Design
- Implementation or coding
- Testing
- Deployment
- Maintenance

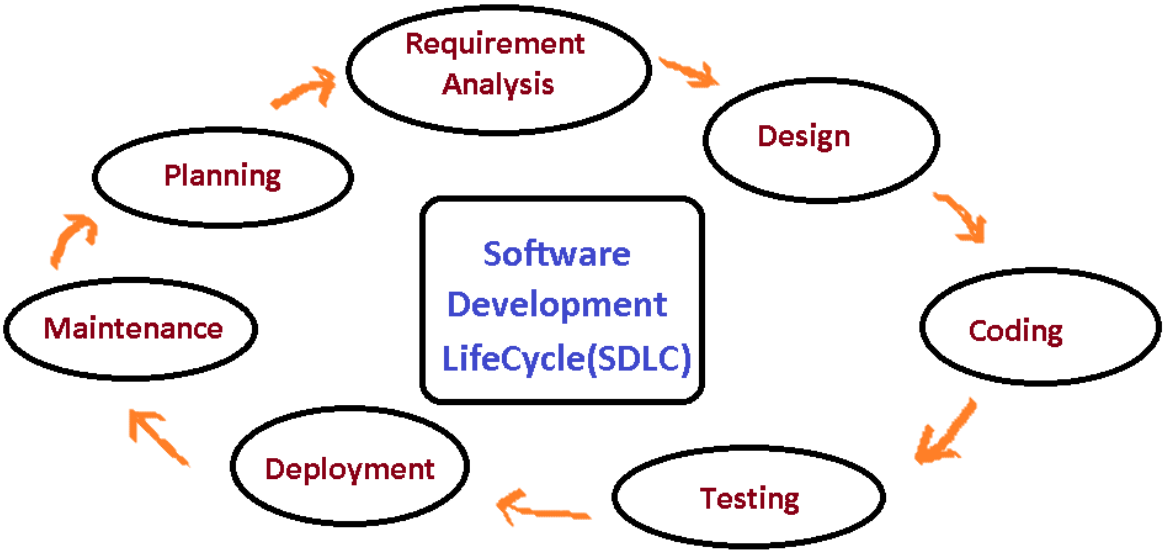

Figure 2.3: Software Development Life-Cycle

- **Planning:**

  We highlight how the Planning phase was meticulously executed to set project objectives, define scope, allocate resources, and establish a roadmap for development.

- **Requirement Gathering and Analysis:**

  During this phase, all the relevant information is collected from the customer to develop a product as per their expectation. Any ambiguities must be resolved in this phase only.
  Once the requirement gathering is done, an analysis is done to check the feasibility of the development of a product. In case of any ambiguity, a call is set up for further discussion.

- **Design:**

  In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

- **Implementation or Coding:**

  Implementation/Coding starts once the developer gets the Design document.

The Software design is translated into source code. All the components of the software are implemented in this phase.

- **Testing:**

  Testing starts once the coding is complete and the modules are released for testing. In this phase, the developed software is tested thoroughly and any defects found are assigned to developers to get them fixed.
  Retesting, regression testing is done until the point at which the software is as per the customer's expectation. Testers refer SRS document to make sure that the software is as per the customer's standard.

- **Deployment:**

  Once the product is tested, it is deployed in the production environment or first UAT (User Acceptance testing) is done depending on the customer expectation.

- **Maintenance:**

  After the deployment of a product on the production environment, maintenance of the product i.e. if any issue comes up and needs to be fixed or any enhancement is to be done is take care by the developers.

2.4 JUSTIFICATIONS OF THE NEW SYSTEM

The journey begins with a comprehensive exploration of the critical reasons and justifications that have led educational institutions to seek a transition from the existing class scheduling methods to the proposed Web Based Class Schedule Management System.

- Operational Inefficiencies: We highlight the operational inefficiencies that stem from the limitations of the existing system. These inefficiencies often lead to suboptimal resource allocation, reduced adaptability, and increased administrative workload.

- Increasing Complexity: Educational institutions face a dynamic landscape with evolving academic programs, diverse student needs, and extracurricular activities. The existing systems may struggle to keep pace with this complexity.

- Limitations of the Existing System: We meticulously dissect the limitations and shortcomings of traditional class scheduling methods, which often involve manual processes or outdated software. These limitations may encompass data inaccuracies, scheduling conflicts, and administrative burdens.

2.5 SYSTEM MODEL

Our system model is designed around three distinct user roles: Admin, Teacher, and Student, each with their specific privileges and functionalities. Let's delve into the

details of what each role can access and accomplish within our Web Based Class Schedule Management System with flowchart:

Admin Role:

- **Login:** Admins can log in to the system.

- **Admin Dashboard:** After logging in, admins are directed to their dashboard, where they can access the following functionalities:

- **Overview:** Admins can access an informative summary of essential data, including details about teachers, departments, time slots, batches, rooms, regular rooms, lab rooms, and per-semester statistics. This overview provides a comprehensive snapshot of the system's key components.

- **Schedule:** Admins can manage schedules, including creating routines, viewing routines, and viewing batches.

- **Add:** Admins have the authority to add various entities, such as teachers, batches, departments, semesters, courses, rooms, and time slots.

- **View:** Admins can view detailed information related to teachers, batches, departments, semesters, courses, rooms, and time slots.

- **Logout:** Admins can log out of the system, returning to the home page.

ADMIN:

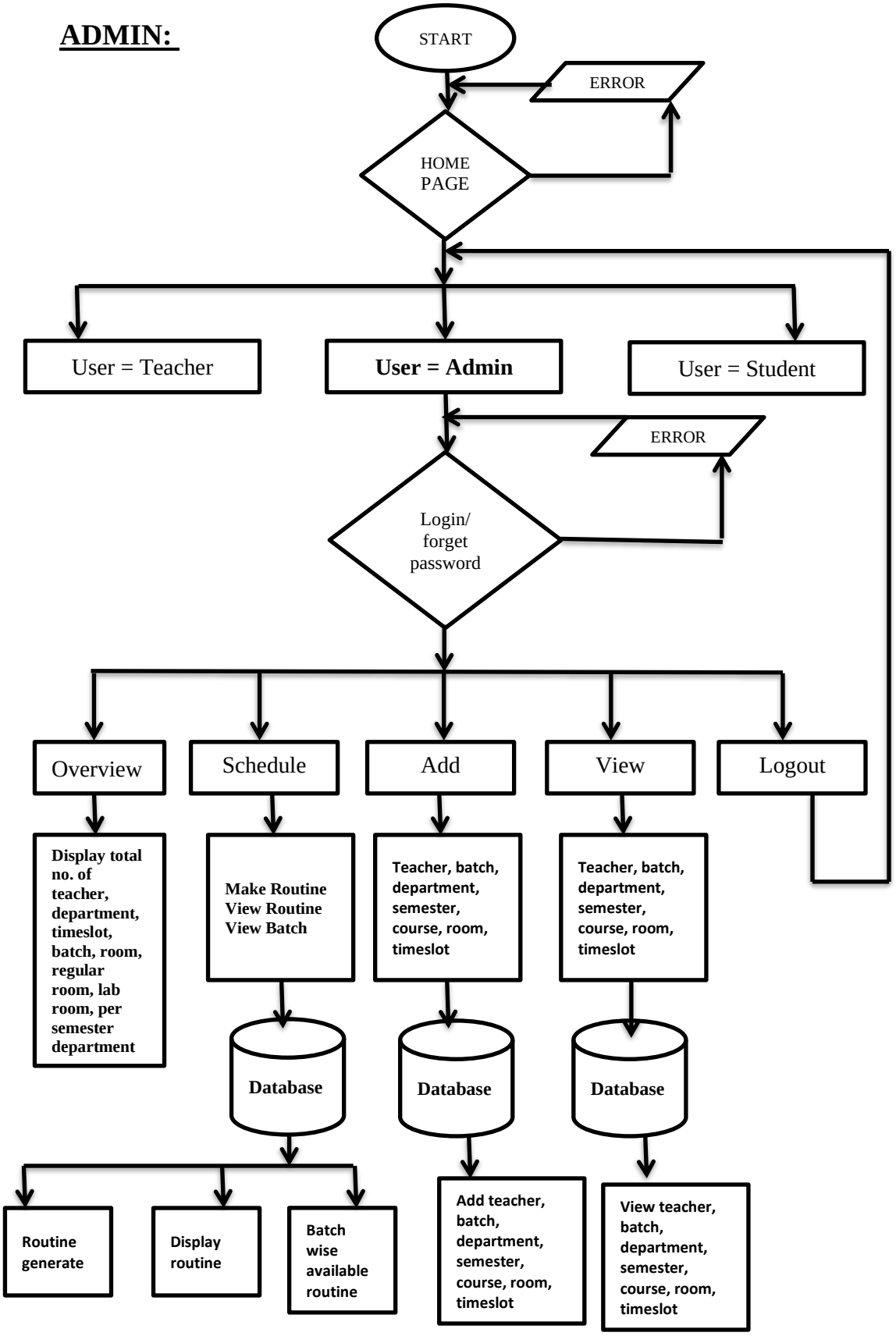

Figure 2.5.1: Admin Workflow Diagram

Teacher Role:

- **Login:** Teachers can securely log in to the system using their unique credentials, ensuring access to their personalized features.

- **Teacher Dashboard:** Upon successful login, teachers are seamlessly directed to their dedicated dashboard, offering a tailored set of functionalities that cater to their specific needs and responsibilities.

- **Profile:** Within the Teacher Dashboard, educators can conveniently access and review their profile information. This feature allows them to keep their personal and professional details up to date.

- **Routine:** Routine feature empowers teachers to effortlessly access and manage their class schedules. Teachers can easily view their schedules, ensuring they are well-prepared for their teaching commitments. Furthermore, they have the option to download their schedules as PDF files, facilitating offline reference.

- **Current Class:** The Current Class functionality provides teachers with real-time information about their ongoing classes. This valuable feature helps educators stay on top of their daily teaching commitments, ensuring a smooth classroom experience.

- **Logout:** At any point, teachers can opt to log out from the system, which gracefully returns them to the home page. This option ensures security and privacy when accessing the system from shared or public computers.

The Teacher role is designed to streamline teachers' access to essential information and tools, promoting efficient class management and an enhanced teaching experience within the educational institution.

TEACHER:

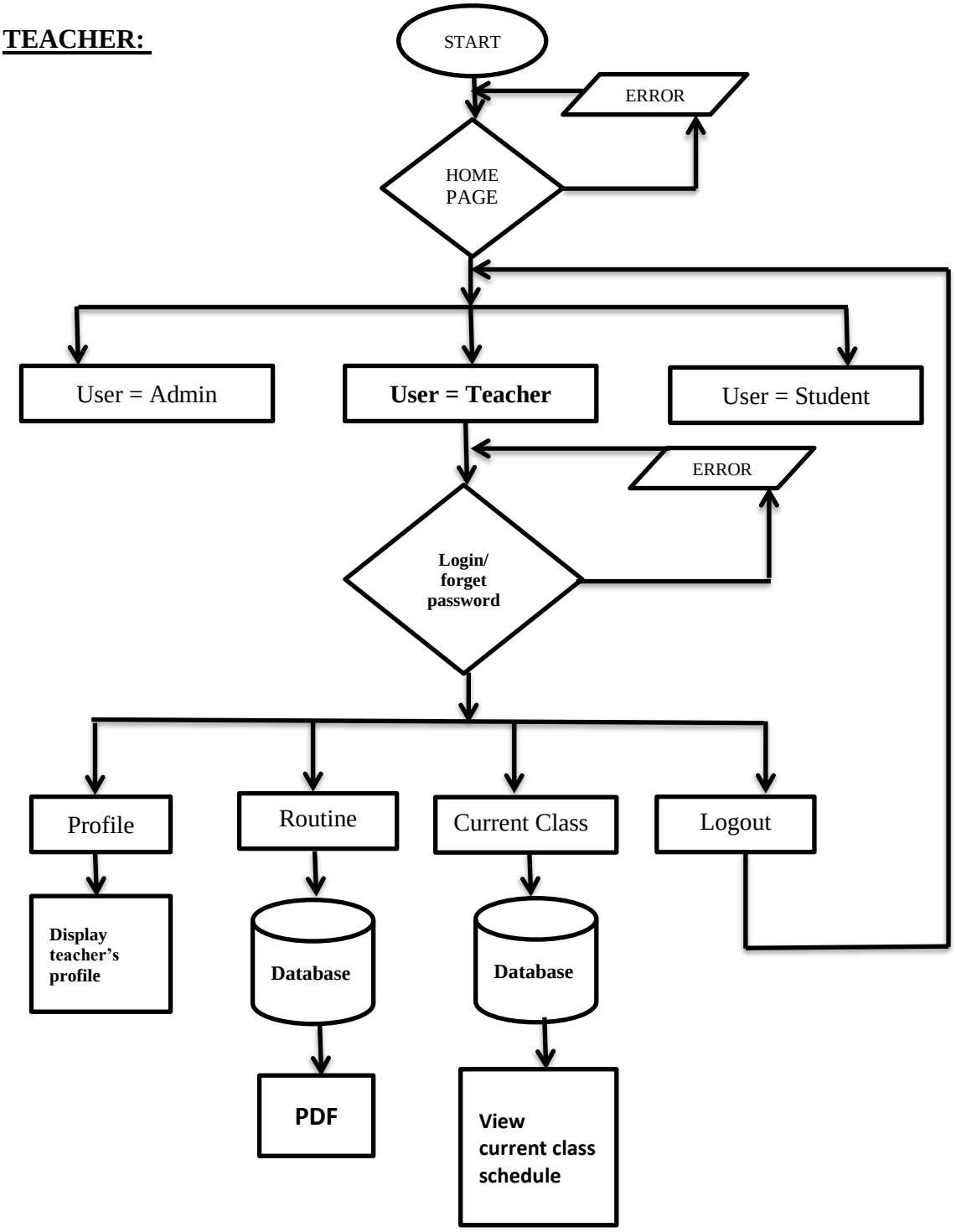

Figure 2.2: Teacher Workflow Diagram

Student Role:

- **Home Page:** Students, upon visiting the home page, can immediately access specific features without the need for a login.
- **Student Dashboard:** They can directly navigate to their dashboard, where they can utilize the following functionalities:
- **Routine:** Students can view their class schedule by selecting batch, semester, session parameters and also to download their schedules as PDF files.
- **Find Teacher:** Students can check the availability of a specific teacher by entering day, time, and selecting the teacher.
- **Check Room:** Students can find available rooms for a particular day and time slot.
- **Back Option:** Students have the option to return to the home page.

STUDENT:

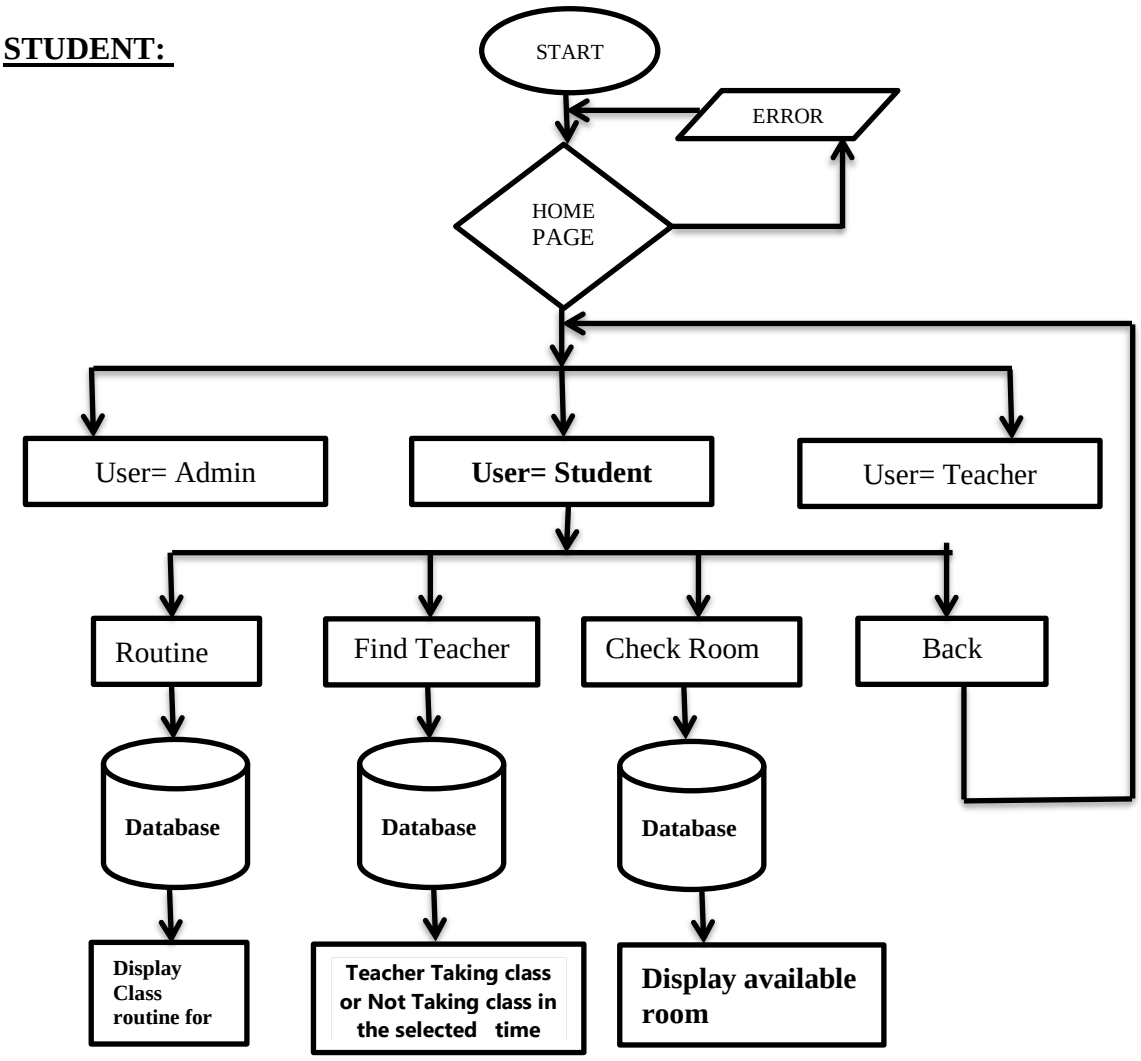

Figure 2.3: Student Workflow Diagram

Chapter 3 WORK FLOW AND METHODOLOGY

3.1 Requirements Analysis

For each system development, two types of requirements are needed, which are mentioned below:

- Hardware Requirement
- Software Requirement

Hardware requirements: These are the physical components that the system will need to run, such as the CPU, memory, storage, and network.

Software requirements: These are the features and functions that the system must have, such as the ability to store data, process transactions, and generate reports.

3.1.1 Hardware requirement

In the development and deployment of the Web Based Class Schedule Management System project, careful consideration of hardware requirements is paramount. This section delves into the specifics of the hardware infrastructure necessary to ensure the smooth operation of the application.

Server: The core component of the hardware requirement is the web server. The project relies on a standard web server to host and serve the application to users. This server should possess several key attributes to support the demands of the Web Based Class Scheduling System.

Processing Power: A web server with adequate processing power ensures that the application can handle concurrent user requests efficiently. This is especially important during peak usage times, such as class registration periods.

Memory (RAM): Sufficient RAM is crucial for the application to store and manipulate data efficiently. It allows the server to handle multiple database queries and perform calculations without slowing down.

Storage: Adequate storage space is necessary to house the application's code, database files, and any additional assets such as images or documents. This ensures that the server has the capacity to manage all aspects of the Class Scheduling System.

Network Connectivity: A stable and high-speed internet connection is essential for the server to respond promptly to user requests. It ensures that data can be transferred between the server and client devices without delays.

Redundancy and Backup: To minimize downtime and data loss, hardware redundancy and regular backups should be considered. This helps ensure that the Class Scheduling System remains accessible and data remains secure even in the event of hardware failures.

Scalability Considerations: Additionally, it's important to consider scalability. As the Class Schedule Generator gains users and data, the hardware infrastructure should be scalable to accommodate increased demand. Scalability may involve upgrading server resources or even deploying multiple servers to distribute the workload.

Cloud-Based Solutions: In modern web application development, many institutions opt for cloud-based hosting solutions. Cloud platforms like Amazon Web Services (AWS), Microsoft Azure, or Google Cloud offer scalability, reliability, and ease of management. The choice of a cloud or on-premises server depends on institutional preferences and resources.

Budget Considerations: Lastly, budget constraints may influence the selection of hardware. While it's important to ensure that hardware meets the project's requirements, cost-effective solutions should also be explored to maximize resource utilization.

The hardware requirement section serves as a foundation for the Class Schedule Generator's technical infrastructure. It highlights the critical need for a robust web server with adequate processing power, memory, storage, and network connectivity to support the application's efficient operation. Additionally, considerations for scalability and redundancy are pivotal for long-term sustainability and uninterrupted access to the Class Schedule Generator. Hardware requirements depending on the machine and operating system.

Below given the hardware requirements for the PHP application. **'Table 3.1.1'** Hardware requirement.

Hardware	Details
Processor	3th gen or above and core i3or above
Speed	2.80GHz
RAM	6GB
Hard Disk/SSD	500GB/256GB
CD/DVD Drive	750MB
Keyboard	Standard
Mouse	Two or Three Button Mouse
Monitor	SVGA/LED/LCD

Table 3.1.1: Hardware requirement

3.1.2 Software requirement

The "Software Requirement" section is the compass that guides the Class Schedule System project through the intricate realm of web-based application development. It lays out the essential software components and tools, akin to a master blueprint that architects the digital infrastructure.

Web Server: The selection of Apache as the web server is akin to choosing the foundation for a skyscraper. Apache, renowned for its reliability and scalability, acts as the bedrock on

which the Class Schedule System's web-based interface thrives. It's the gateway that ensures seamless user access, serving web pages and executing the intricate web of scripts that power the application.

Database Management System: MySQL, the chosen Database Management System, is the digital repository that safeguards the project's lifeblood – data. Think of it as the meticulously organized library where schedules, course details, teacher allocations, and room availability records are diligently cataloged. Its open-source nature ensures accessibility, reliability, and performance for efficient data storage and retrieval.

Programming Languages: PHP and JavaScript, the dynamic duo of web development, serve as the project's linguistic backbone. PHP is the scripting maestro that orchestrates the server-side magic. It processes data, generates dynamic content, and communicates with the MySQL database, ensuring schedules are generated with precision. On the other hand, JavaScript, the client-side virtuoso, adds a layer of interactivity. It breathes life into the user interface, responding to user actions, and providing a seamless, responsive experience.

Development Tools: Visual Studio Code (VS Code) is the virtual workspace where the project's code symphony is composed. Much like a conductor's podium, VS Code offers an array of tools and features that enhance the development process. Its syntax highlighting, debugging capabilities, and version control integration ensure that the development orchestra performs in harmony.

Libraries or Frameworks: While not specified, this category recognizes that the project may call for additional tools to expand its capabilities. It's akin to a treasure chest of possibilities. Depending on the project's needs, libraries or frameworks like Bootstrap and jQuery can be introduced to weave responsive design elements and enhance user interactions seamlessly.

Software requirements depending on the Application. Below given the Software requirements for the PHP application. **'Table 3.1.1'** Software requirement

Software	Details
Operating System	Windows/Linux/Ubuntu
Technology	PHP, FPDF
Web Technologies	HTML, CSS, Tailwind, JavaScript, jQuery
IDE	VS Code
Text Editor	VS Code/Sublime Text
Web Server	XAMPP/WAMP
Database	MySQL
Browser	Chrome/Firefox

Table 3.1.2: Software requirement

3.2 Work Flow and Methodology

This section acts as a well-structured blueprint that navigates the entire development journey of the Web Based Class Scheduling System, from its inception to its triumphant completion. It serves as a roadmap, outlining the steps, processes, and methodologies that will guide the project through its lifecycle. Let's take a closer look at how this section sets the foundation for the development process:

Requirements Gathering: The Compass of Insight

Our journey commences with a pivotal phase known as requirements gathering. During this phase, the project team engages in close collaboration with key stakeholders, including educational administrators, teachers, and potentially even students. This engagement allows us to gain profound insights into their unique needs and expectations. By actively listening and meticulously documenting these requirements, we lay the cornerstone of our project's roadmap.

Design and Planning: Crafting the Vision

With a comprehensive understanding of the project's requirements in hand, we move seamlessly into the design and planning phase. Here, we bring our vision to life through the creation of wireframes, mockups, and the establishment of a robust system architecture. This phase shapes the visual and structural framework of the Class Scheduling System, ensuring its alignment with the project's overarching objectives.

Development: Breathing Life into Ideas

As the design begins to take form, we transition into the development phase where the real magic happens. This is where the coding and scripting take center stage. Utilizing languages such as HTML, CSS, JavaScript, and PHP, we craft the user interface, implement intricate functionality, and skillfully manage the flow of data.

Testing: Ensuring Excellence

Quality assurance is our steadfast commitment. In the testing phase, we establish and execute rigorous testing protocols. This encompassing approach includes unit testing to validate individual components, integration testing to ensure seamless interaction between various modules, and user acceptance testing to guarantee that the Class Scheduling System functions seamlessly and aligns perfectly with the initial requirements.

Deployment: The Moment of Triumph

Finally, the moment of culmination arrives as the Class Scheduling System is gracefully deployed onto a web server. It becomes readily accessible to administrators and teachers, signaling the dawn of a new era in efficient class scheduling.

In essence, this workflow and methodology serve as the North Star for our project. They ensure that every step we take is purposeful and meticulously aligned with our overarching objectives. This section lays out the path from the inception of an idea to its triumphant realization—a transformative solution known as the Class Scheduling System.

3.3 Technical Skills Acquired

During the development of our Web Based Class Schedule Management System project, we, the development team, acquired a diverse set of technical skills and knowledge that were instrumental in the successful execution of the project. These skills spanned various domains, including web development, database management, and programming languages, such as PHP, JavaScript, and SQL. Below, we delve into the specific technical skills that we honed during this journey:

Web Development Expertise:

- **HTML Mastery:** We delved deep into HTML (Hyper Text Markup Language), mastering the art of structuring web content. We became adept at creating semantically meaningful web pages and understanding the significance of proper HTML markup.

- **Tailwind CSS Proficiency:** Tailwind CSS emerged as a cornerstone skill. We honed our abilities to style web pages, creating visually appealing and responsive designs. We learned to leverage Tailwind CSS to streamline the styling process.

- **JavaScript Prowess:** JavaScript emerged as a vital tool for client-side interactivity. We gained proficiency in JavaScript, enabling us to create dynamic and user-friendly web interfaces. We also explored JavaScript libraries like jQuery to simplify DOM manipulation.

Database Management Know-How:

- **MySQL Mastery:** The project's database management revolved around MySQL, a robust and widely used relational database management system. We acquired skills in designing database schemas, writing efficient SQL queries, and ensuring data integrity.

Server-Side Scripting Proficiency:

- **PHP Expertise**: PHP (Hypertext Preprocessor) was the chosen server-side scripting language. We became proficient in PHP, using it to build the backend logic of the Class Scheduling System. We crafted PHP scripts for tasks such as user authentication, data processing, and schedule generation.

Development Tools Utilization:

- **Visual Studio Code (VS Code):** As our primary development environment, VS Code offers a powerful set of tools for code editing, debugging, and version control integration. We became adept at utilizing VS Code's features to streamline the development process.

Libraries and Frameworks Familiarity:

- **PHPMailer:** PHPMailer is a popular PHP library for sending email messages via SMTP, allowing developers to programmatically generate and send emails from their web applications. It simplifies the process of email composition and delivery, offering features**.**

- **dompdf (PHP library):** dompdf is a PHP library that enables the generation of PDF documents from HTML and CSS. It allows developers to convert web-based content into PDF files, making it useful for creating printable reports, invoices, and other document types directly from web applications.

Database Management System Expertise:

- **Relational Database Management:** Understanding the principles of relational databases was crucial for managing complex scheduling data. We mastered concepts like tables, relationships, and normalization.

- **Open-Source Database Software:** The project capitalized on the advantages of open-source software, with MySQL being an open-source RDBMS. We appreciated the cost-effectiveness and community support that open-source software offers.

These technical skills and knowledge formed the backbone of our Class Schedule System project. Our ability to proficiently utilize these skills was instrumental in the successful development and deployment of the application, ensuring its efficiency, accuracy, and adaptability within educational institutions. This project has provided us a wonderful opportunity to acquire various technological skills. Though it may take some more time to master those skills, at this moment, we are confident to apply them in any of our projects. Some of the significant technical aspects that we have and led during the project are listed below.

3.4 SOME DESIGN AND DEVELOPMENT TOOLS ARE GIVEN BELOW

We are using some open-source design and development tools for developed our software. Bellow some open-source tools description.

3.4.1 HTML (Hyper Text Markup Language)

Foundation of Web Development:

HTML, which stands for Hyper Text Markup Language, serves as the cornerstone of web development. It is the language used to structure and format the content of web pages. Every web page you encounter on the internet is constructed using HTML.

Structured Content:

At its core, HTML is all about organizing content. It provides a structured way to define various elements on a web page, such as headings, paragraphs, lists, images, links, forms, and more. Each of these elements is represented by HTML tags, which are enclosed in angle brackets (< >). For example:

```
<h1>This is a Heading</h1>
<p>This is a paragraph of text.</p>
<img src="image.jpg" alt="An example image">
<a href="https://www.example.com">Visit Example Website</a>
```

Hierarchy and Nesting:

HTML employs a hierarchical structure. Elements can be nested within other elements, creating a tree-like structure. For instance, a paragraph (<p>) can contain a link (<a>) or a list (<ul>) can contain list items (<li>).

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
</ul>
```

Attributes and Values:

HTML elements often have attributes that provide additional information about the element. These attributes are specified within the opening tag and have values. For example, the src attribute in an image tag specifies the image file's source.

```
<img src="image.jpg" alt="An example image">
```

Semantic HTML:

HTML also incorporates semantic elements that carry meaning about the structure of the content. For instance, <header>, <nav>, <main>, <section>, and <footer> are semantic tags that help define the layout and organization of a web page. Using semantic HTML enhances accessibility and search engine optimization (SEO).

```
<header>
  <h1>Website Header</h1>
</header>
<main>
  <section>
    <h2>Section Title</h2>
```

```
      <p>Content within the section.</p>
    </section>
</main>
<footer>
    <p>Copyright © 20XX Website</p>
</footer>
```

Responsive Design:

HTML plays a crucial role in creating responsive web pages that adapt to different screen sizes and devices. It allows developers to structure content in a way that makes it readable and visually appealing on various platforms, from desktops to smartphones.

Interactivity and JavaScript:

While HTML provides the structure and content of web pages, interactivity is often achieved through JavaScript. HTML elements can be manipulated and animated using JavaScript to create dynamic and engaging user experiences.

So, HTML is the foundational language of the web, used to structure and organize content on web pages. It provides a structured and hierarchical approach to web development, allowing developers to create visually appealing and interactive websites that are accessible across different devices and screen sizes.

3.4.2 Tailwind CSS (Cascading Style Sheets)

Tailwind CSS is a cutting-edge technology revolutionizing web development by offering precise control over the visual appearance and layout of web pages. It has emerged as the design language of choice for the web, empowering developers to craft pixel-perfect designs and responsive layouts effortlessly. Tailwind CSS has quickly become a cornerstone of contemporary web design, enabling websites to achieve diverse visual styles and adaptable designs with remarkable ease.

Key Functions of Tailwind CSS

Styling Elements: Tailwind CSS empowers web developers to effortlessly style HTML elements. It provides an extensive utility-first framework, allowing developers to apply styles with attributes such as font size, color, background, spacing, borders, and more. With Tailwind CSS, defining how elements should look is a breeze, making it a pivotal tool for creating visually stunning web pages.

Layout Control: Tailwind CSS is instrumental in precisely controlling the layout of web content. It offers a rich set of classes to specify the positioning and dimensions of elements, ensuring content is organized and visually pleasing. Tailwind CSS's layout capabilities shine when crafting responsive and adaptable designs that seamlessly fit various screen sizes and

devices.

Consistency: One of Tailwind CSS's core strengths lies in promoting design consistency. Developers can define styles centrally, ensuring that changes to styling rules are instantly applied across the entire website. This maintains a unified visual identity throughout the web project.

Separation of Concerns: Tailwind CSS adheres to the principle of separation of concerns, keeping the structure (HTML), presentation (CSS), and behavior (JavaScript) of a web page neatly separated. This clean separation enhances code maintainability and facilitates collaborative development efforts.

How Tailwind CSS Works

Tailwind CSS operates by applying utility classes directly to HTML elements. Each utility class corresponds to a specific property-value pair that defines styles. Here's an example:

```html
<!-- HTML Element with Tailwind CSS Classes -->
<div class="bg-blue-500 text-white p-4 rounded-lg">
  This is a styled div.
</div>
```

In this example, Tailwind CSS classes are applied to a `<div>` element, defining its background color, text color, padding, and rounded corners.

Benefits of Tailwind CSS

Efficiency: Tailwind CSS streamlines the process of creating and managing consistent styles across a website efficiently. Changes can be made in a centralized configuration file and instantly affect multiple elements.

Flexibility: Tailwind CSS offers immense design flexibility. Designers can easily experiment with different layouts, colors, and typography to achieve the desired look and feel.

Responsiveness: Tailwind CSS excels in supporting responsive design. It seamlessly adapts web pages to various screen sizes and devices using responsive utility classes and media queries.

Accessibility: Tailwind CSS can be used to enhance web accessibility by providing semantic structure and ensuring content is perceivable and usable by all users, including those with disabilities.

Compatibility: Tailwind CSS enjoys broad support from all major web browsers, making it a reliable and future-proof choice for web development.

In summary, Tailwind CSS stands as a powerful and versatile tool in modern web

development. It empowers developers and designers to effortlessly create visually stunning, well-structured, and responsive websites while maintaining code efficiency and flexibility. Tailwind CSS plays a central role in defining the user experience and presentation of web content, making it a go-to choose for web developers seeking precision and speed in their projects.

3.4.3 Brief description on PHP program

PHP, which stands for Hypertext Preprocessor, plays a pivotal role in the Class Scheduling System project. It is a versatile and widely used server-side scripting language that is integral to the web development process. Here's an extended explanation of PHP and its significance in this context:

Understanding PHP: PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose language. It is particularly well-suited for creating dynamic web pages and web applications. PHP scripts are executed on the web server, generating HTML, JavaScript, or other output that is then sent to the client's web browser.

Role in the Class Schedule Generator: In the context of the Class Schedule Generator, PHP serves several crucial purposes:

Server-Side Logic: PHP handles the server-side logic of the application. This includes processing user requests, interacting with the database, and generating dynamic web pages in response.

Database Interaction: PHP connects to the MySQL database to retrieve and store data. It facilitates the creation of routines, such as generating class schedules based on user inputs.
User Authentication: PHP manages user authentication, ensuring that only authorized administrators and teachers can access the system.

Data Validation: PHP performs data validation and sanitization to prevent security vulnerabilities, such as SQL injection or cross-site scripting (XSS) attacks.
Dynamic Content: PHP allows for the generation of dynamic content on web pages. For instance, it populates web forms with data from the database and generates schedules in real-time based on user selections

Sample PHP Code:

Here's a simplified example of PHP code to illustrate its usage in a web application:
php
Copy code

```
<?php
// Define a variable
$name = "John";
// Use PHP to generate HTML content
```

```
echo "<html>";
echo "<head><title>Welcome</title></head>";
echo "<body>";
echo "<h1>Hello, $name!</h1>";
echo "<p>This is a PHP-generated message.</p>";
echo "</body>";
echo "</html>";
?>
```

In the Class Schedule Generator, PHP code is employed extensively to create dynamic web pages, interact with the database to fetch schedule data, and manage user authentication. Its versatility and ability to seamlessly integrate with other web technologies make it a fundamental component of the project's success.

3.4.4 PHP library

PHPMailer:

PHPMailer is a widely used open-source PHP library that allows to send email messages from PHP application. It simplifies the process of sending emails through PHP by providing a convenient and flexible set of functions and features. Here are some key aspects of PHPMailer:

1. **Email Composition:** PHPMailer enables to compose emails with ease. You can set the sender's address, recipient's address, subject, message body, and attachments.

2. **Support for Multiple Protocols:** It supports various email-sending protocols, including SMTP, PHP's mail() function, and sendmail, allowing one to choose the most suitable method for sending emails.

3. **Security Features:** PHPMailer includes features like SMTP authentication, SSL and TLS encryption, and secure password handling, ensuring the security of email transmissions.

4. **Attachment Handling:** Can attach files to emails effortlessly, making it suitable for sending documents, images, and other files.

5. **HTML and Plain Text Emails:** PHPMailer supports both HTML and plain text email formats, giving flexibility in creating visually rich or straightforward email messages.

6. **Error Handling:** It provides detailed error messages and debugging information, making it easier to diagnose and troubleshoot email-sending issues.

dompdf (PHP library):

dompdf is a popular PHP library that allows to generation of PDF documents from HTML and CSS. It's particularly useful when we need to convert web content into printable or downloadable PDFs. Here are the key features of dompdf:

1. **HTML to PDF Conversion:** dompdf can take HTML and CSS as input and convert it into a PDF document. This is useful for generating reports, invoices, certificates, and other printable content dynamically.

2. **Support for CSS Styles:** It supports a wide range of CSS styles and properties, allowing to creation of visually appealing and formatted PDFs.

3. **UTF-8 and Unicode Support:** dompdf supports UTF-8 encoding and Unicode characters, ensuring that PDFs display text in multiple languages and character sets.

4. **Image Embedding:** This can include images in PDFs, making it possible to include logos, graphics, or other visual elements in generated documents.

5. **Custom Fonts:** It allows to include custom fonts in PDFs, ensuring that your documents match your desired typography.

6. **Pagination Control:** Can control page breaks and pagination within PDFs, ensuring that content flows correctly across pages.

7. **Open Source:** dompdf is an open-source library, making it freely available for use and modification in your PHP projects.

Both PHPMailer and dompdf are valuable PHP libraries that simplify email sending and PDF generation, respectively, making them essential tools for web developers when dealing with email communication and document generation in web applications.

3.4.5 JavaScript

JavaScript is a fundamental scripting language that plays a pivotal role in the development of the Class Scheduling System. Its presence in the project is multifaceted, contributing to various aspects that enhance user experience, interactivity, and functionality. Below, we delve into the extended explanation of JavaScript's significance:

1. Client-Side Interactivity:

JavaScript is primarily utilized for client-side interactivity. In the context of the Class Scheduling System, it empowers users, including administrators and teachers, to interact with the application seamlessly within their web browsers. For instance:
User Input Validation: JavaScript ensures that users provide valid input when configuring schedules or making adjustments, preventing erroneous data from being processed.
Dynamic User Interfaces: JavaScript facilitates the creation of dynamic user interfaces that respond in real time to user actions, providing an intuitive and responsive user experience.

Interactive Forms: User forms, such as those for schedule adjustments or profile updates, leverage JavaScript to validate input, provide instant feedback, and streamline the submission process.

2. Asynchronous Data Retrieval (AJAX):

In an educational institution's dynamic environment, the ability to retrieve and update data without refreshing the entire page is invaluable. JavaScript, in conjunction with AJAX (Asynchronous JavaScript and XML), allows for the seamless retrieval and display of schedules, teacher profiles, and other relevant information. This not only enhances user experience but also minimizes page load times, optimizing efficiency.

3. Client-Side Data Manipulation:

JavaScript is instrumental in client-side data manipulation. For example:
Sorting and Filtering: In the context of class schedules, JavaScript enables users to sort and filter schedules based on various criteria, facilitating quick access to specific information.
Schedule Adjustments: Teachers can make adjustments to their schedules, such as marking class cancellations or requesting room changes, directly within the web interface, thanks to JavaScript's data manipulation capabilities.

4. Event Handling:

The Class Schedule Generator's user interface is enriched with event-driven functionalities, allowing users to respond to specific triggers. JavaScript handles event handling, ensuring that actions like submitting forms or clicking buttons are captured and processed accurately.

5. Integration with External Libraries and Frameworks:

JavaScript seamlessly integrates with external libraries and frameworks, enhancing the project's capabilities. For instance, it can work in tandem with jQuery, a popular JavaScript library, to simplify complex tasks such as DOM manipulation or making AJAX requests.

6. Responsiveness and User Experience:

JavaScript plays a pivotal role in ensuring that the Class Schedule Generator is responsive across various devices and screen sizes. It adapts the user interface to offer an optimal experience, whether accessed from a desktop computer or a mobile device.

3.4.6 Ajax (Asynchronous JavaScript and XML)

Ajax is a fundamental technology in web development that allows for the asynchronous exchange of data between a web browser and a web server without requiring a full page reload. It stands for "Asynchronous JavaScript and XML," although XML is not mandatory, and JSON (JavaScript Object Notation) is often used instead. Ajax enables web applications to fetch and send data in the background, providing a more responsive and dynamic user

experience. Here are key components and concepts related to Ajax:

1. **Asynchronous Communication:** Ajax allows for asynchronous communication, which means that a web page can send a request to the server and continue to work without waiting for the response. When the response is ready, it can be processed by JavaScript, typically through callback functions.

2. **XMLHttpRequest (XHR) Object:** XHR is a built-in JavaScript object that is used to create HTTP requests to a server. It can send GET or POST requests and handle responses. Modern web development often uses the `fetch` API as an alternative to XHR.

3. **Data Formats:** While XML was originally the preferred data format for Ajax, JSON has become the standard due to its simplicity and ease of use. JSON is a lightweight, human-readable format for structuring data.

4. **Callback Functions:** Ajax relies heavily on callback functions. You can define functions that execute when certain events occur, such as when a request is successfully completed (success callback) or when an error occurs (error callback). This allows developers to update the web page or perform specific actions in response to the server's response.

5. **DOM Manipulation:** One of the primary use cases of Ajax is to fetch data from the server and update the web page dynamically without requiring a full page reload. This can involve manipulating the Document Object Model (DOM) to display fetched data, such as updating a chat window, refreshing a news feed, or filtering and sorting a list of items.

6. **Security Considerations**: Ajax requests are subject to the same-origin policy, which means they can only be made to the same domain from which the web page originates. To overcome this limitation, techniques like Cross-Origin Resource Sharing (CORS) are used to allow requests to other domains.

7. **Progressive Enhancement:** Developers often implement Ajax as a form of progressive enhancement. This means that the basic functionality of a web page is maintained without JavaScript, and Ajax is used to enhance the user experience for those with JavaScript-enabled browsers.

8. **Frameworks and Libraries:** Many JavaScript frameworks and libraries, such as jQuery, Axios, and Fetch API, simplify the implementation of Ajax requests, making it easier for developers to work with asynchronous data.

Ajax is a critical technology that enables modern web applications to provide real-time updates, dynamic content loading, and a smoother user experience by asynchronously exchanging data with web servers. It plays a pivotal role in creating interactive and responsive web applications.

3.4.7 jQuery

jQuery, a widely adopted JavaScript library, serves as an indispensable tool for web developers seeking to enhance the functionality and interactivity of their websites. It is not only open source but also generously provided for free under the MIT license, making it accessible to developers worldwide. In recent years, jQuery has ascended to the pinnacle of popularity, establishing itself as the most widely used JavaScript library in the realm of web development.

At its core, jQuery simplifies the process of manipulating HTML documents and handling events within web pages. It offers a versatile and user-friendly interface for developers to seamlessly select and modify elements on a webpage, respond to user interactions, and create dynamic content. This simplification is particularly valuable because it abstracts the intricacies of JavaScript, making it more approachable for developers of varying skill levels.
One of jQuery's standout features is its ability to harmonize with a wide range of web development projects. Whether you're building a simple personal blog, a dynamic e-commerce platform, or a complex web application, jQuery's flexibility and extensive plugin ecosystem make it adaptable to diverse scenarios. Developers can effortlessly integrate jQuery into their projects, harnessing its capabilities to add features such as interactive forms, real-time content updates, and visually appealing animations.

Furthermore, jQuery's open-source nature fosters a vibrant community of developers and contributors. This community continuously expands the library's capabilities by developing and sharing plugins, extensions, and best practices. This collaborative environment ensures that jQuery remains a robust and evolving toolset for web development.

In essence, jQuery has earned its place as the go-to choose for web developers because of its user-friendly approach, adaptability to various project types, and the support of an active developer community. It empowers developers to create websites and web applications that are not only functional but also engaging and user-friendly, ultimately enhancing the overall web browsing experience for users across the globe.

3.4.8 MySQL

MySQL, the most popular Open-Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Web site (http://www.mysql.com/) provides the latest information about MySQL software. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free software open-source projects that require a full-featured database management system often use MySQL. TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook Twitter, Flickr, and YouTube. In below **‘Figure 3.4.8’** represent dashboard of database.

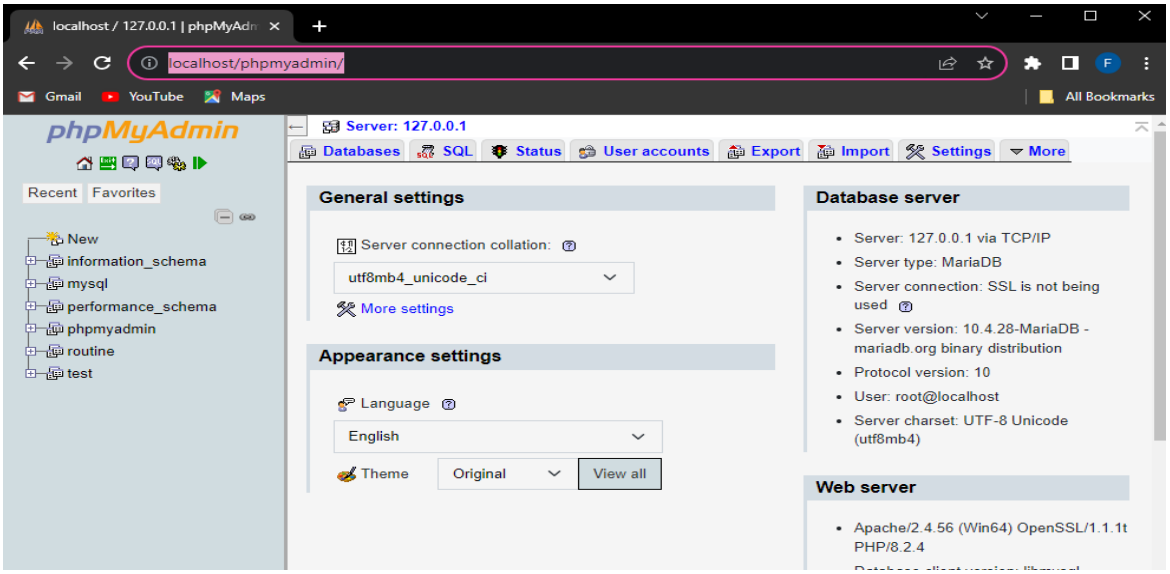

Figure 3.4.8: Dashboard of MySQL Database

3.4.8.1 MySQL DB is Relational

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and "pointers" between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

3.4.8.2 MySQL Software is Open Source

Open-Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs.

3.5 XAMPP SERVER

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works

equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well. '**Figure 3.5'** represent XAMPP Server Control Panel.

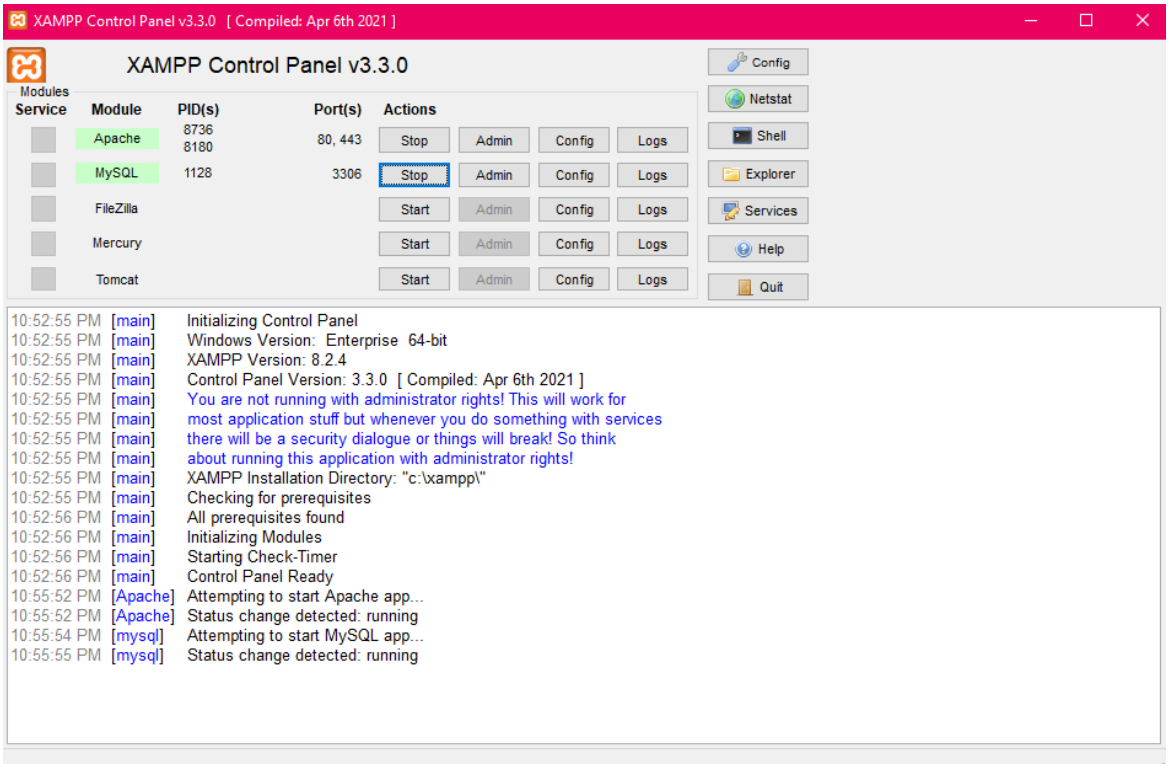

Figure 3.5: XAMPP Server Control Panel

3.6 VISUAL STUDIO CODE (VS CODE)

Visual Studio Code, commonly referred to as VS Code, stands as a powerful and widely adopted integrated development environment (IDE) that offers extensive support for a multitude of programming languages and technologies. Unlike NetBeans, which is often associated with Java and PHP development, VS Code is renowned for its versatility and adaptability, making it a preferred choice among developers across various domains.

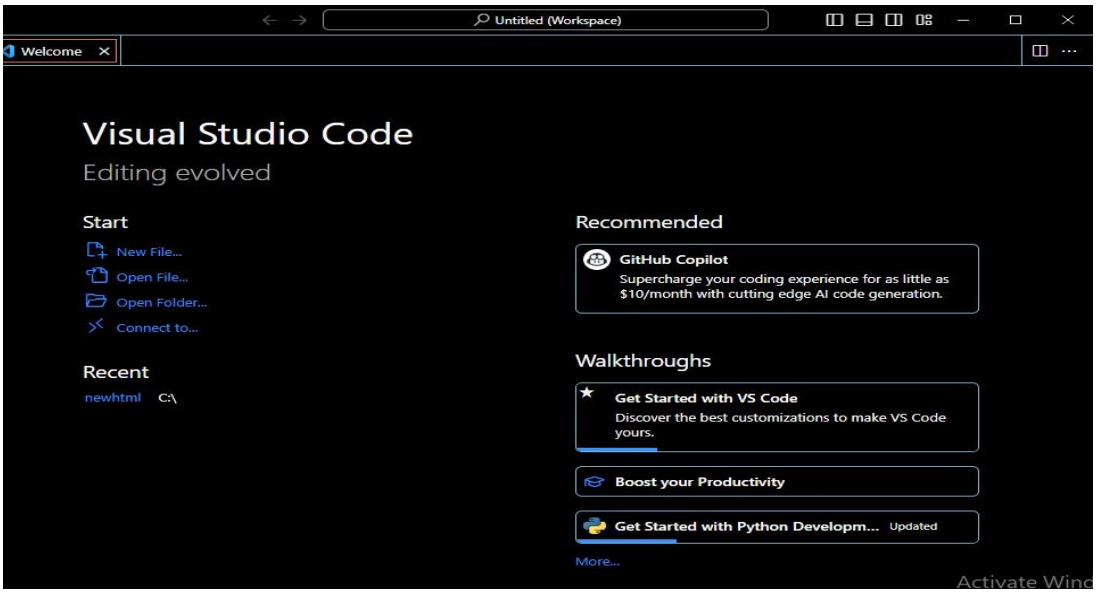

Figure 3.6: Visual Studio Code Interface

Key Characteristics of Visual Studio Code (VS Code):

- **Open-Source and Cross-Platform:** VS Code is open-source, just like NetBeans, but it goes a step further by being cross-platform, which means it can run on Windows, macOS, and Linux. This cross-platform compatibility allows developers to work seamlessly across different operating systems.

- **Lightweight and Fast:** VS Code is known for its speed and resource efficiency. It loads quickly and consumes minimal system resources, making it an ideal choice for both lightweight and resource-intensive projects.

- **Extensive Language Support:** While NetBeans specializes in Java, PHP, and C++, VS Code provides support for an extensive range of programming languages. It achieves this through a rich ecosystem of extensions that can be easily installed to add language-specific features and integrations.

- **Extensions Marketplace:** VS Code boasts a thriving marketplace of extensions, with thousands of plugins available for various purposes. These extensions can enhance code editing, provide debugging tools, and integrate with version control systems like Git, among many other functionalities.

- **Integrated Terminal:** VS Code comes with an integrated terminal, enabling developers to run command-line tools and scripts without leaving the IDE. This feature enhances productivity by reducing the need to switch between the IDE and external terminals.

- **Intelligent Code Editing:** VS Code provides intelligent code editing features such as auto-completion, code navigation, and real-time error checking. These features assist developers in writing clean and error-free code.

- **Version Control Integration:** VS Code seamlessly integrates with version control systems like Git. Developers can view and manage code repositories directly within the IDE.

- **Customization:** VS Code offers extensive customization options. Users can tailor the IDE to their preferences by choosing themes, configuring keyboard shortcuts, and adjusting settings to suit their workflow.

- **Community and Support:** VS Code has a vast and active community of users and contributors. This ensures a steady stream of updates, bug fixes, and support resources available through forums, documentation, and online communities.

In essence, Visual Studio Code (VS Code) distinguishes itself from NetBeans by its lightweight, cross-platform nature, extensive language support, and a rich ecosystem of extensions. It caters to a broad spectrum of developers and development scenarios, making it a highly adaptable and popular choice in the software development landscape. is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages.

Chapter 4 DATA BASE IMPLEMENTATION

4.1 DATA BASE CONNECTION

We delve into the vital aspect of our site's database system, which pertains to establishing a robust database connection and executing SQL queries to retrieve data for display on the front-end.

4.1.1 SQL Query

SQL, or Structured Query Language, stands as a powerful programming language meticulously designed for managing data within relational databases. SQL operates through clear and declarative statements, ensuring data accuracy, security, and the preservation of database integrity, regardless of its size.
Here, we present a compendium of commonly used SQL commands and concepts essential for efficient database management:

- **SELECT Statements:** The cornerstone of SQL queries, SELECT statements are employed to retrieve data from a database. Every query commences with SELECT.
- **WHERE Clause:** The WHERE clause is a pivotal component that indicates the desire to filter the result set based on specific conditions, ensuring that only rows meeting these conditions are included.
- **UPDATE Statements:** These statements enable you to edit existing rows within a table, facilitating data modification.
- **INSERT Statements:** Used for adding new rows to a table, INSERT statements expand the database with fresh data.
- **DELETE Statements:** DELETE statements are crucial for removing unwanted rows from a table, maintaining data cleanliness.
- **CREATE TABLE:** This command is used to create a new table within the database, allowing you to define the table's name and its individual columns.
- **AND Operator:** The AND operator combines two conditions, requiring both to be true for a row to be included in the result set.
- **LIKE Operator:** Utilized with the WHERE clause, the LIKE operator facilitates the search for specific patterns within a column, enhancing data retrieval.
- **LIMIT Clause:** The LIMIT clause offers control over the maximum number of rows included in the result set, aiding in result set management.
- **Aggregate Functions:** SQL offers various aggregate functions for data analysis, including SUM(), MAX(), MIN(), COUNT(), and AVG(), each serving a unique purpose in data manipulation and analysis.
- **INNER JOIN:** This operation combines rows from different tables when the specified join condition is satisfied, allowing for data integration across tables.
- **OR Operator:** The OR operator offers flexibility by filtering the result set to include rows where either of the specified conditions is true, expanding query possibilities.

In essence, SQL is the backbone of database management, providing a robust and structured approach to handling data within relational databases. The commands and concepts outlined here represent a foundational understanding of SQL, enabling effective data retrieval,

modification, and analysis for our project's database system.

4.2 TABLE OF DATABASE

Here, we will provide an overview of the database structure, including all the tables and their respective data types. Our database is a vital component of the system, containing real-world data that is essential for the successful implementation of our Web Based Class Schedule Management System.

- **Table 1: Admin**

  In this table 1, we have structured it to collect crucial data, including id, email addresses, passwords, etc. **'Figure 4.2.1'** provides a visual representation of this table, showcasing how the admin data is organized within our database.

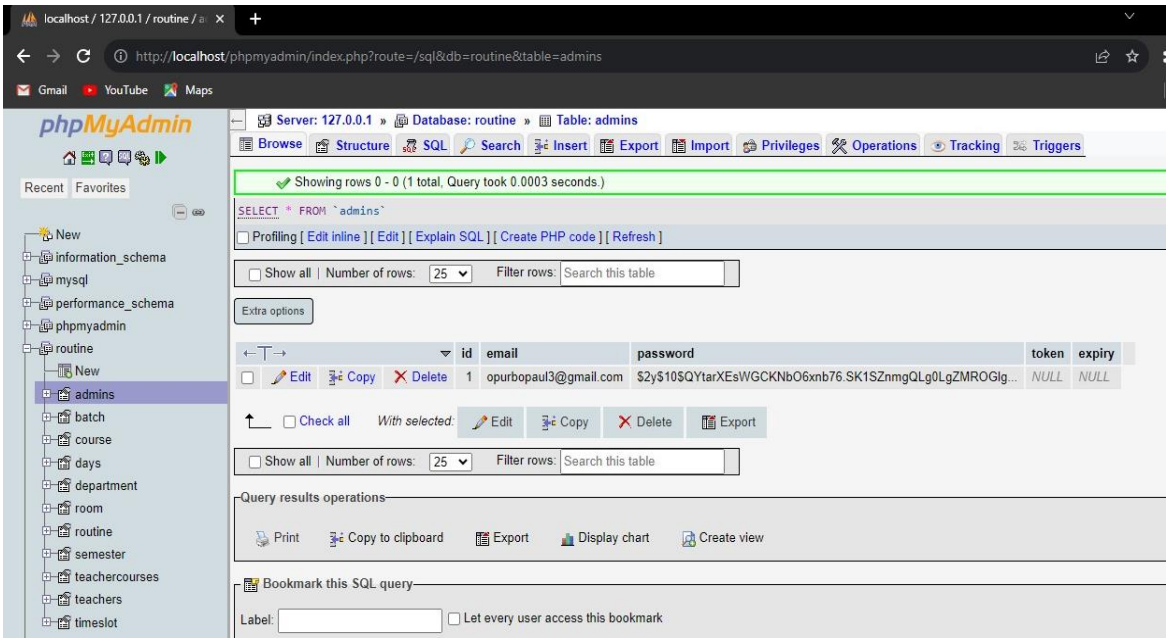

Figure 4.2.1: Admin Table Structure

- **Table 2: Batch**

  Table 2, representing the "Batch" table, is meticulously crafted to serve as a structured repository for collecting essential data related to batches within our database. We have designed this table to capture vital information, including batch_id, batch_number, department_id, batch_shift, etc. **'Figure 4.2.2'** provides a visual representation of this table, showcasing how the batch data is organized within our database.

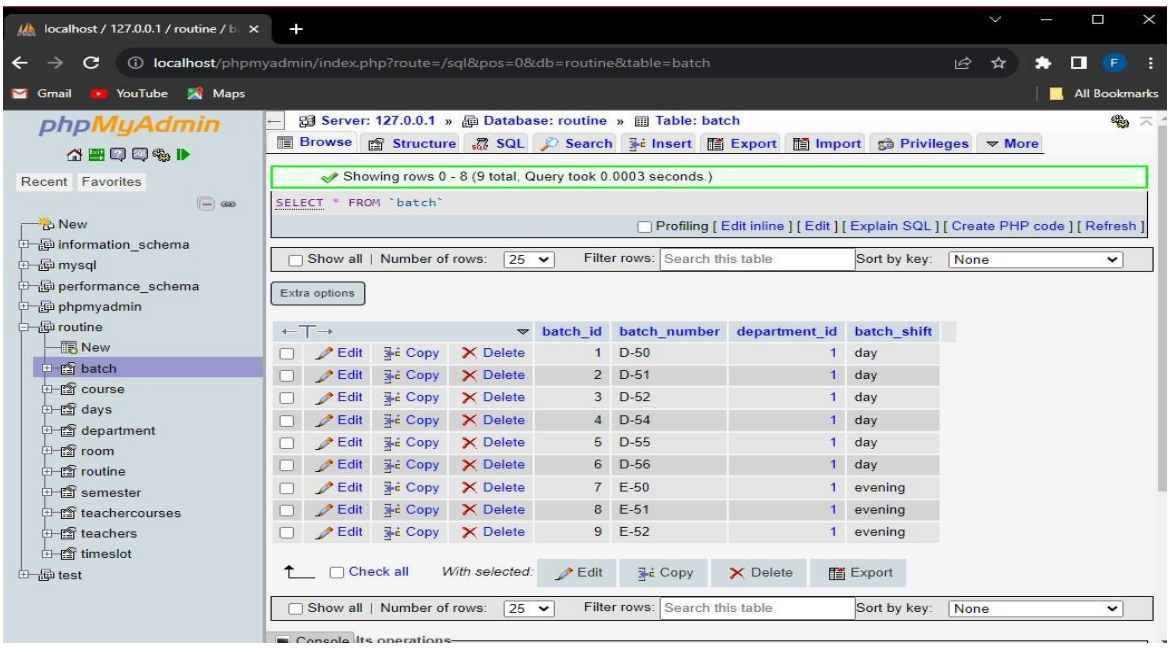

Figure 4.2.2: Batch Table Structure

- **Table 3: Course**

Table 2, representing the "course" table, is meticulously structured to serve as a comprehensive repository for collecting and organizing essential course-related data within our database. We have designed this table with the specific intention of capturing critical information, including course_id, course_name, semester_id, course_type, credits, course_code, department_id, and more. '**Figure 4.2.3**' provides a visual representation of this table, showcasing how the course data is organized within our database.

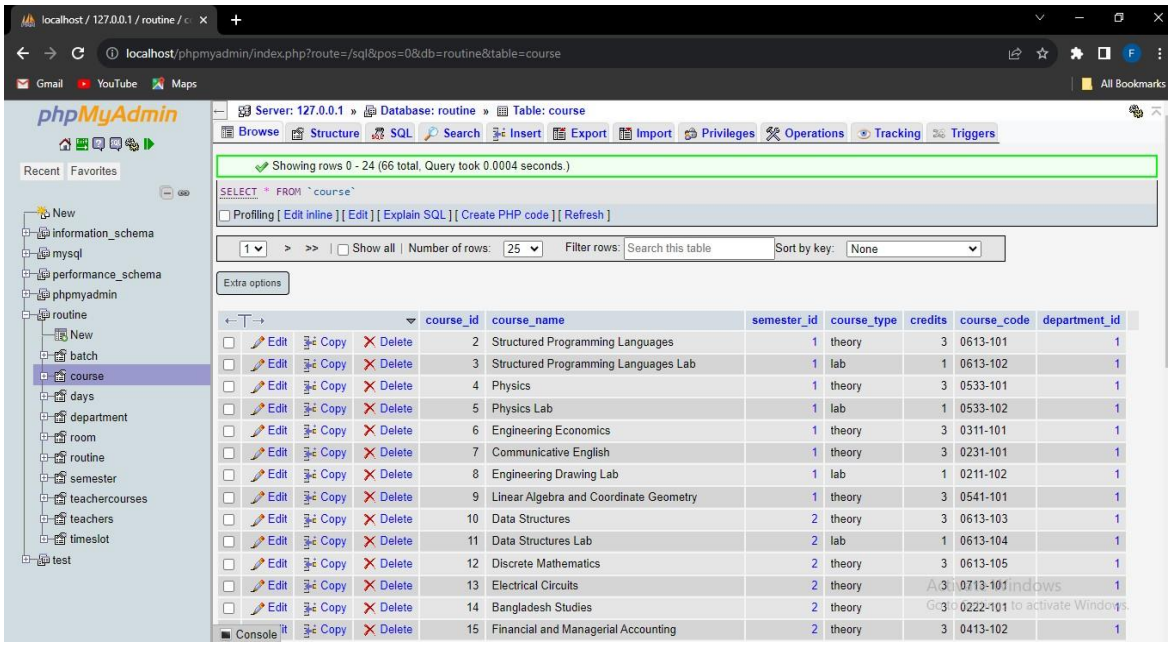

Figure 4.2.3: Course Table Structure

- **Table 4: Department**

  Table 4, representing the "department" table, is meticulously structured to serve as a comprehensive repository for collecting and organizing essential course-related data within our database. We have specifically designed this table to capture critical information, including department_id, department_name, etc. '**Figure 4.2.4**' provides a visual representation of this table, showcasing how the department data is organized within our database.

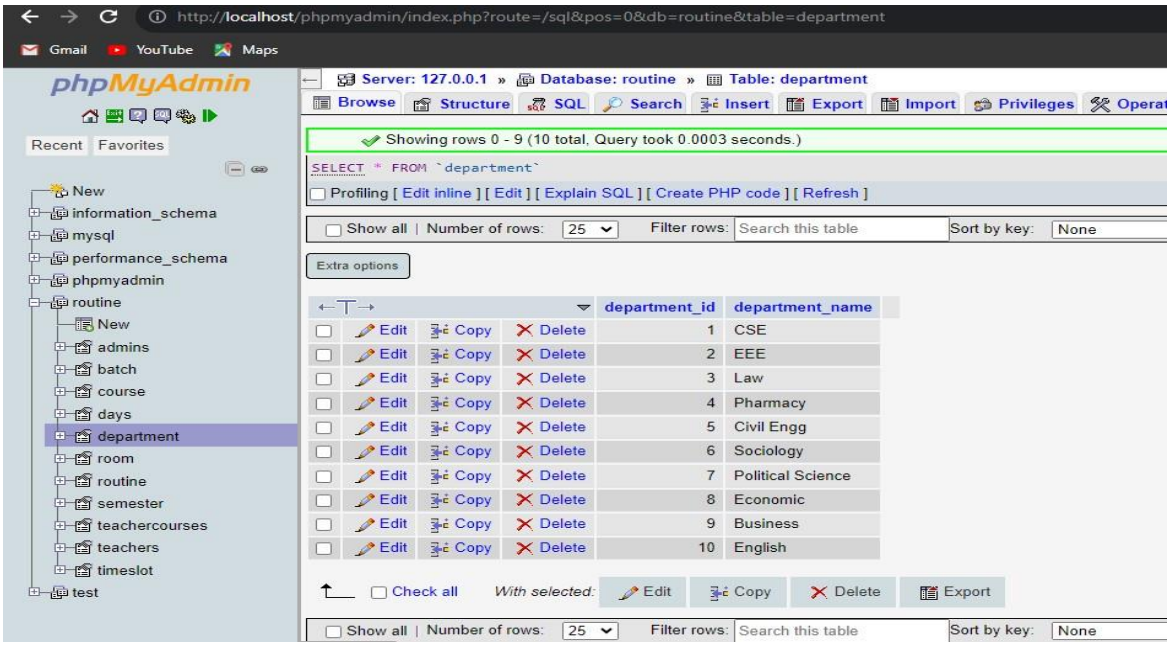

Figure 4.2.4: Department Table Structure

- **Table 5: Room**

  Table 5, representing the "room" table, is meticulously structured to function as a comprehensive repository for collecting and organizing crucial course-related data within our database. Our specific design of this table aims to capture vital information, including room_id, room_number, room_type, etc. '**Figure 4.2.5**' provides a visual representation of this table, showcasing how the room data is organized within our database.

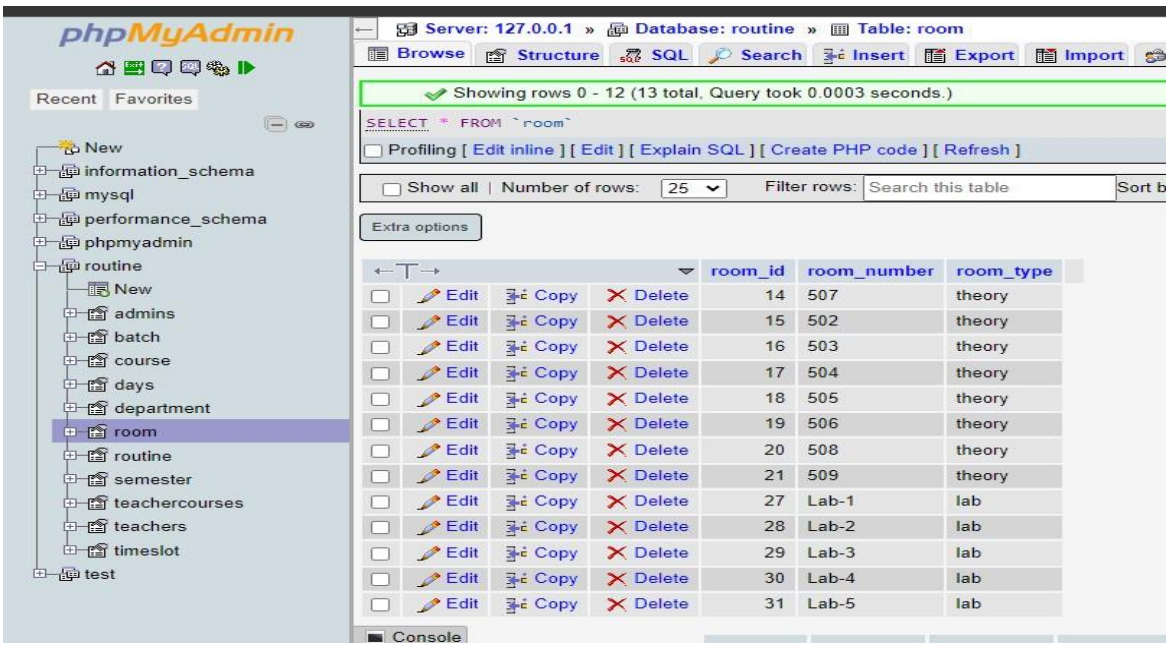

Figure 4.2.5: Room Table Structure

- **Table 6: Routine**

Table 6, representing the "Routine" table, has been meticulously structured to serve as a comprehensive repository for collecting and organizing crucial course-related data within our database. The unique design of this table is aimed at capturing vital information, including routine_id, course_number, day, start_time, room_id, batch, semester, end_time, session, etc. '**Figure 4.2.6**' provides a visual representation of this table, showcasing how the routine data is organized within our database.

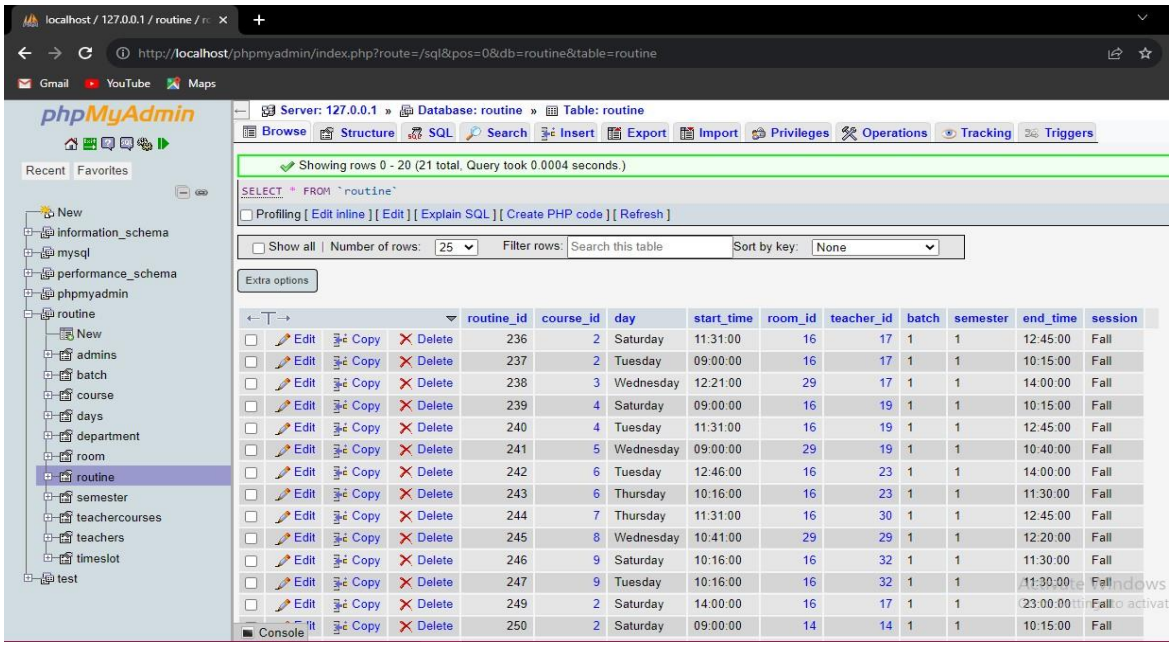

Figure 4.2.6: Routine Table Structure

- **Table 7: Semester**

Table 7, representing the "semester" table, has undergone meticulous structuring to function as a comprehensive repository for collecting and organizing essential course-related data within our database. Our distinctive design approach for this table is tailored to capture critical information, encompassing semester_id, semester_name, department_id, etc. '**Figure 4.2.7**' provides a visual representation of this table, showcasing how the semester data is organized within our database.

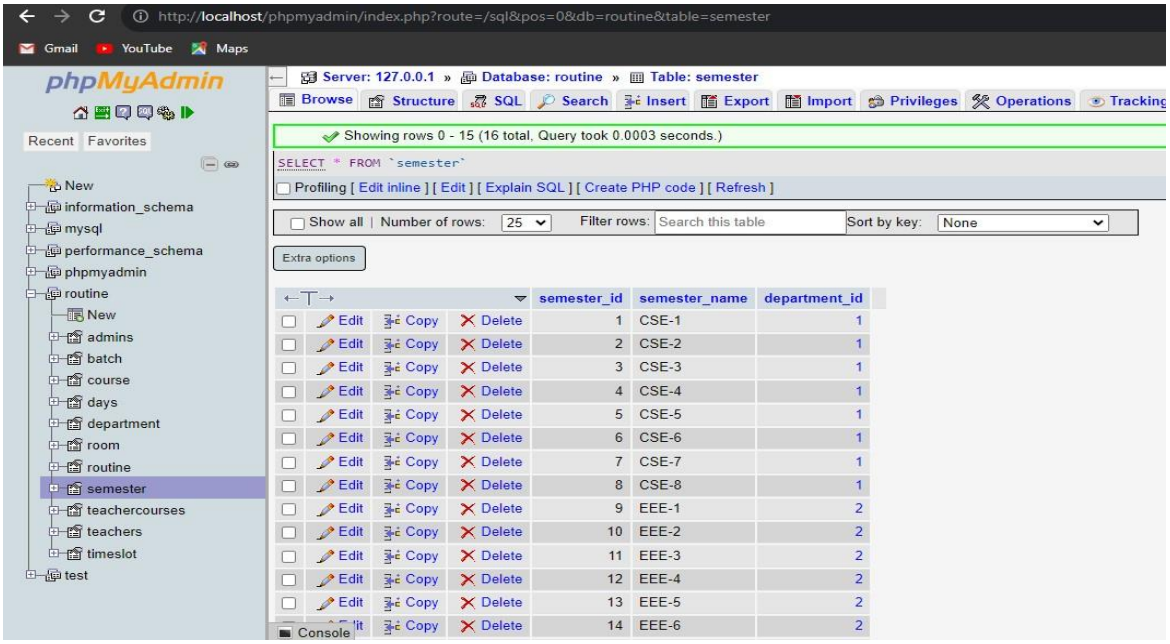

Figure 4.2.7: Semester Table Structure

- **Table 8: Teachercourses**

Table 8, representing the "teachercourses" table, has been meticulously structured to serve as a comprehensive repository for collecting and organizing vital course-related data within our database. The unique design of this table is tailored to capture critical information, including id, teacher_id, course_id, etc. '**Figure 4.2.8**' provides a visual representation of this table, showcasing how the teachercourses data is organized within our database.

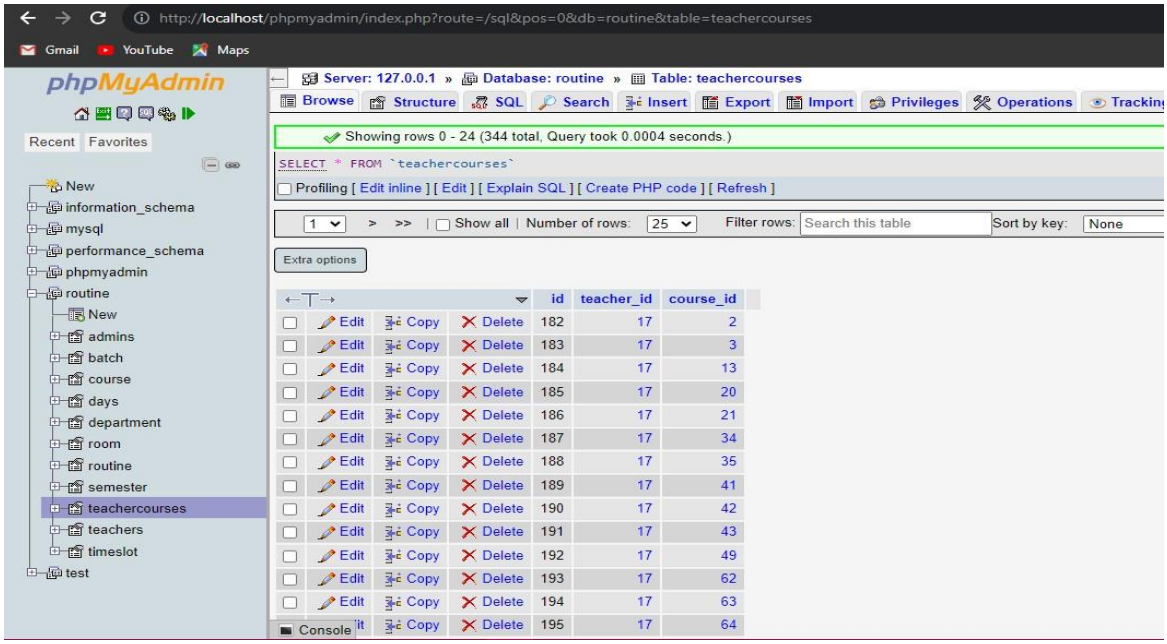

Figure 4.2.8: Teachercourses Table Structure

- **Table 9: Teachers**

  Table 9, representing the "teachers" table, has been intricately structured to function as a comprehensive repository for collecting and organizing vital course-related data within our database. The distinct design of this table is precisely crafted to capture essential information, encompassing id, teacher_id, name, mobile, department_id, position, email, picture, password, and more. **'Figure 4.2.9'** provides a visual representation of this table, showcasing how the teacher's data is organized within our database.

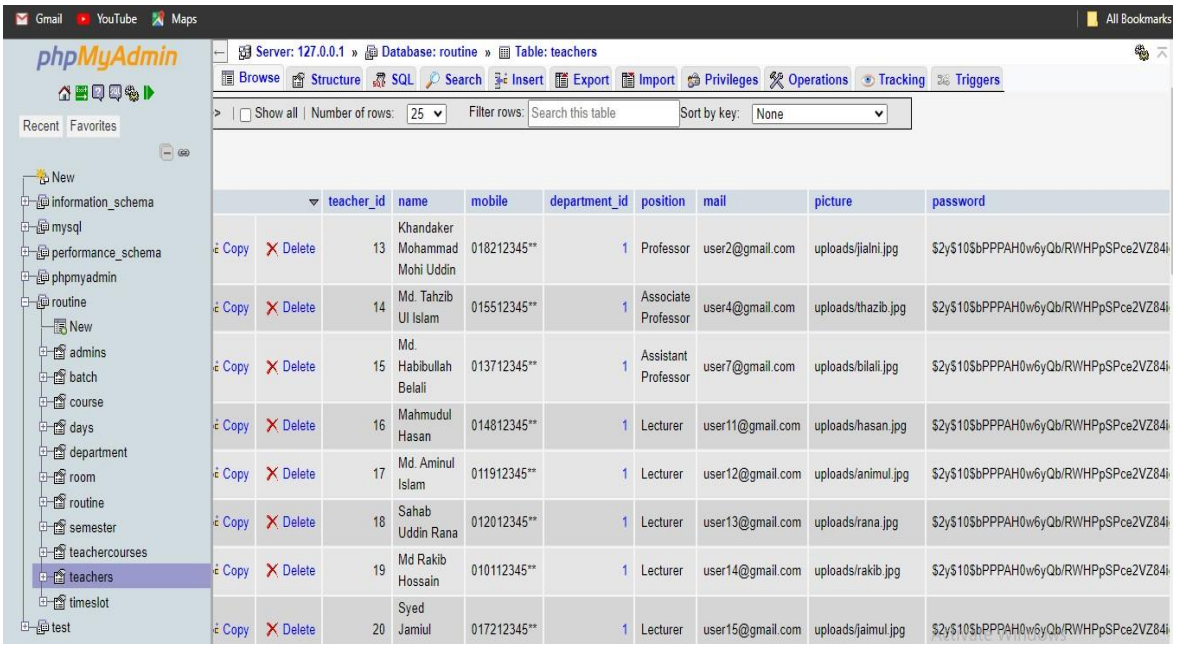

Figure 4.2.9: Teachers Table Structure

- **Table 10: Timeslot**

  Table 10, which represents the "timeslot" table, has been intricately structured to serve as a comprehensive repository for collecting and organizing vital course-related data within our database. The unique design of this table is meticulously crafted to capture essential information, including timetable_id, start_time, end_time, name, class_type, etc. **'Figure 4.2.9'** provides a visual representation of this table, showcasing how the timeslot data is organized within our database.

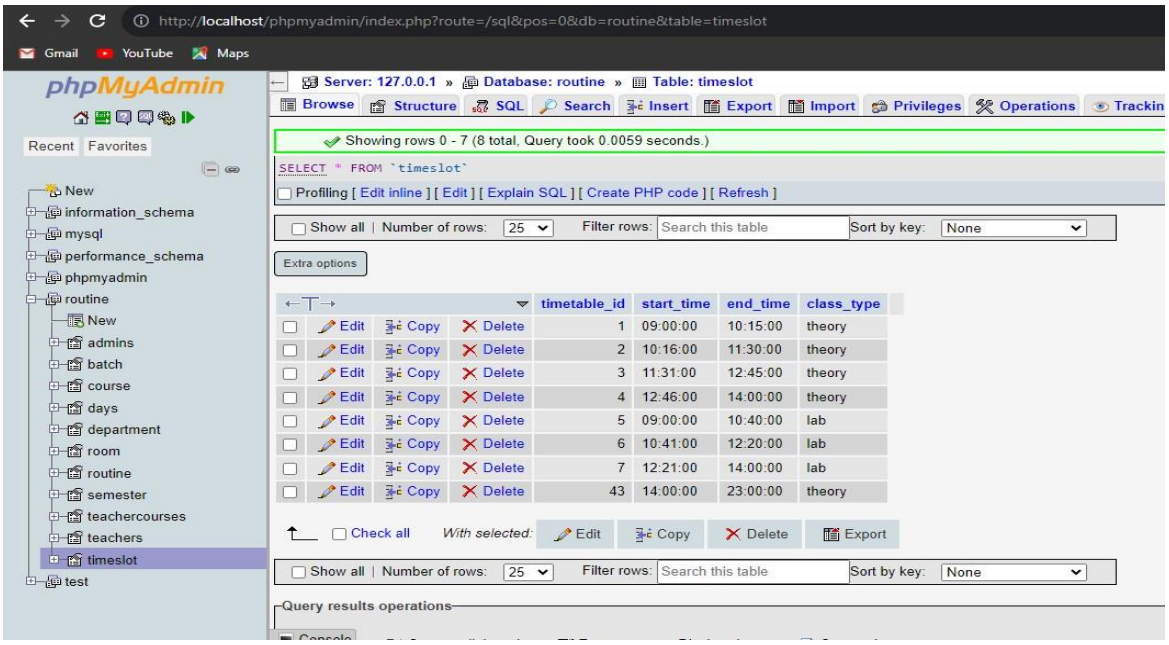

Figure 4.2.10: Timeslot Table Structure

4.3 RELATIONSHIPS OF DATABASE

The database underpinning our educational institution scheduling system comprises ten interconnected tables. These tables work together to efficiently manage and organize critical data for various aspects of academic scheduling. The Batch table, for instance, is closely related to other tables like Course, Routine, and Teachercourses, allowing us to associate courses with specific batches. Similarly, the Course table connects with the Batch and Teachercourses tables, enabling the assignment of courses to specific batches and teachers. Our Department table plays a pivotal role by linking courses to academic departments and assigning teachers to specific departments. Additionally, the Room table is tied to the Routine table, facilitating the scheduling of classes in available rooms. Furthermore, the Semester table associates courses with academic semesters, and the Teachercourses table establishes the connection between teachers and the courses they instruct. Our Teachers table serves as a central repository for teacher data, including department assignments, and connects with the Teachercourses and Department tables. Lastly, the Timeslot table manages class time slots and is linked to the Routine table to allocate specific time slots for classes. These intricate relationships among our database tables ensure the seamless management and scheduling of academic activities within our educational institution scheduling system.

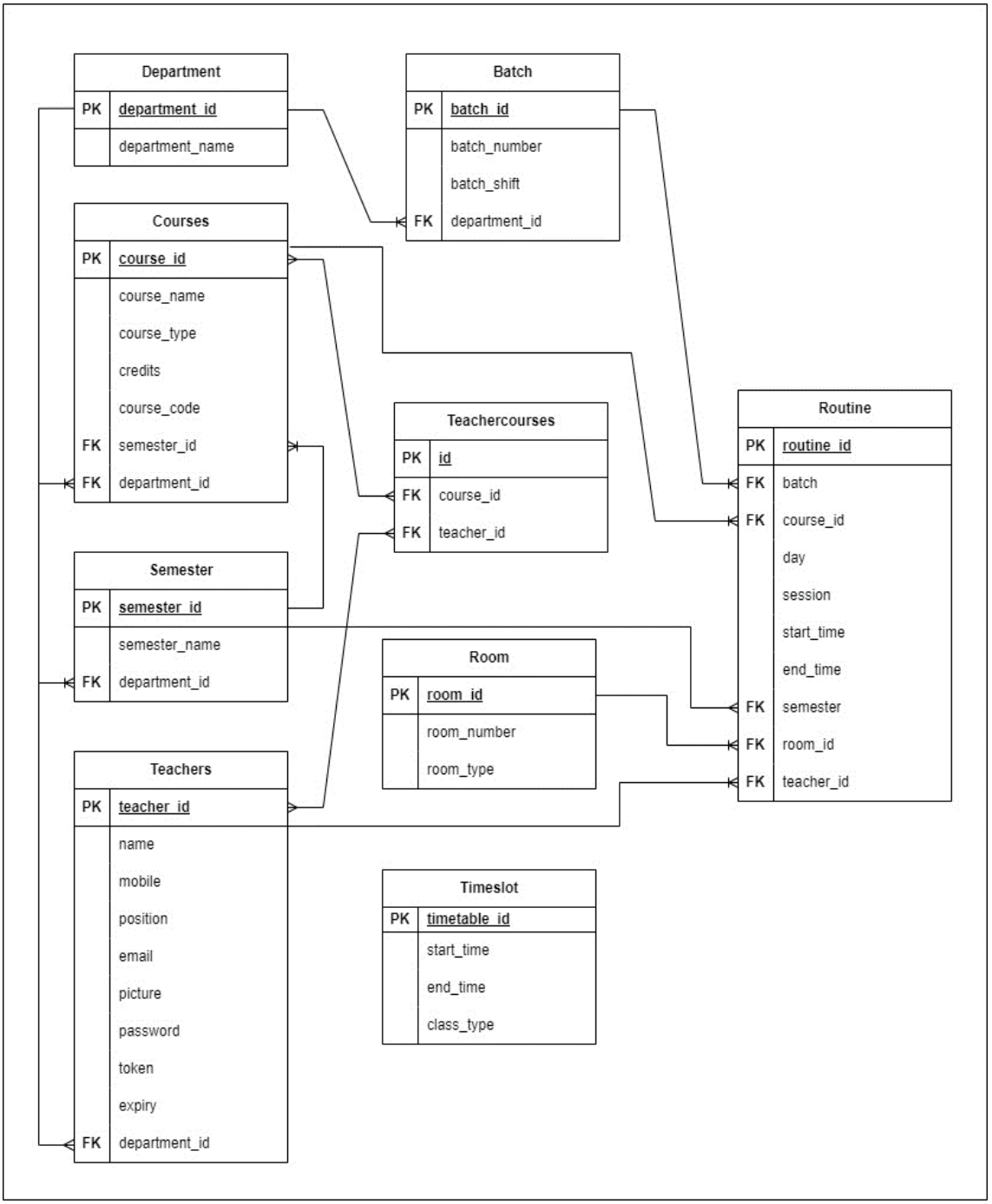

Figure 4.3: Database Table Relationships of Our System

Chapter 5 PROJECT OVERVIEW

5.1 FRONT-END USER INTERFACE DESIGN

Home page of our project '**Figure 5.1'** represents home page of our system.

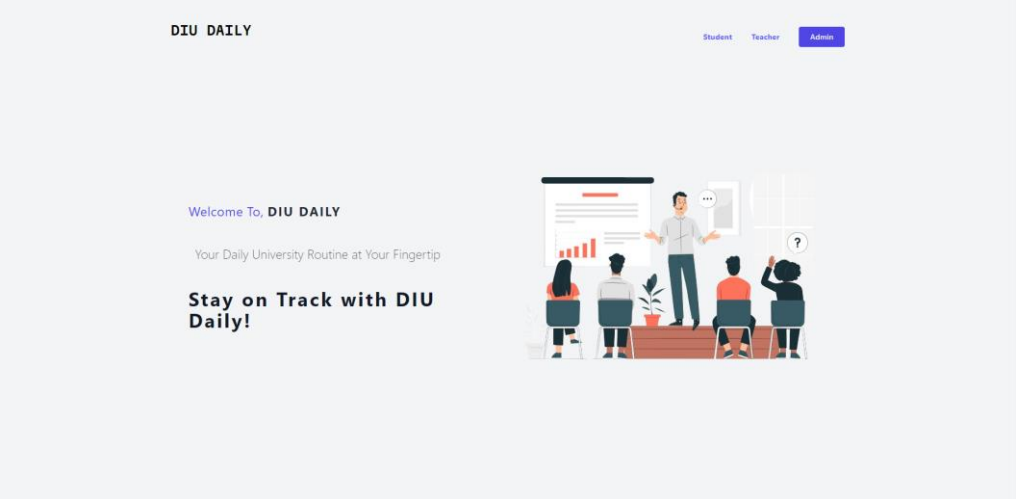

Figure 5.1: System Home Page

5.2 LOGIN PAGE FOR ADMIN

'**Figure 5.2'** illustrates the login page tailored for administrators within our project. This login page serves as the initial point of access for administrators to enter the system. With their unique credentials, administrators gain secure entry to the system, where they can then proceed to the comprehensive admin dashboard for managing various scheduling and academic tasks efficiently.

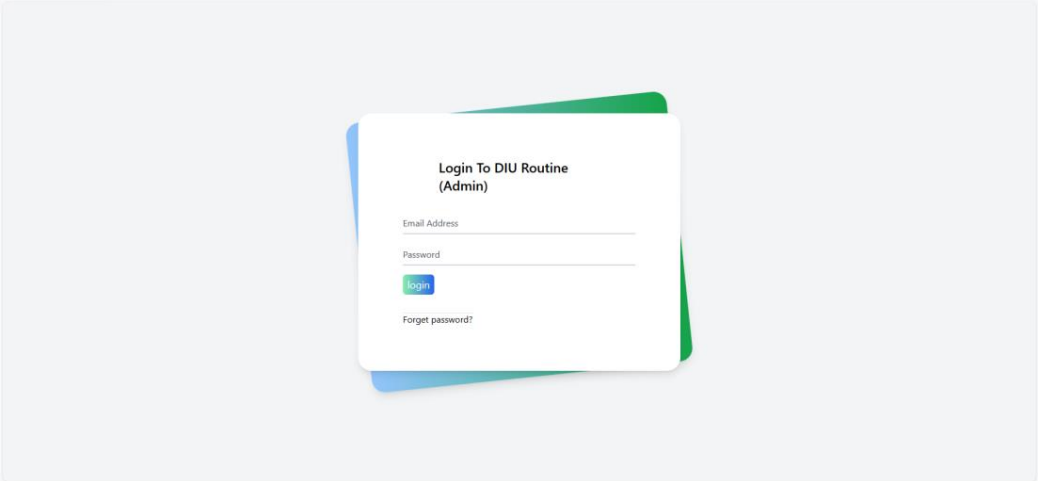

Figure 5.2: Login Page for Admin

5.3 ADMIN DASHBOARD

'**Figure 5.3'** serves as a visual representation of the comprehensive admin dashboard within our project. This dashboard is the control center for administrators, providing access to a range of

essential functionalities. It empowers administrators to efficiently oversee and optimize the scheduling process within the educational institution, ultimately contributing to more effective academic operations.

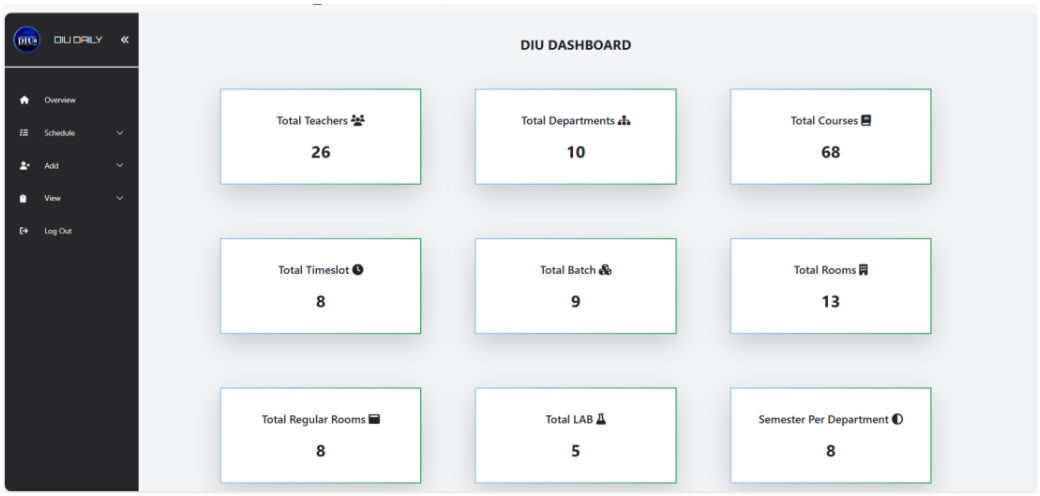

Figure 5.3: Dashboard for Admin of Our System

5.3.1 Schedule

‘**Figure 5.3.1’** visually represents the admin schedule dashboard within our system. This dashboard is a key component of the administrative interface, enabling administrators to manage schedules, including Make Routine, View Routine, View Batch, All Routine and Check Available Room. It offers a user-friendly environment for efficient schedule management, helping administrators streamline the academic operations of the educational institution.

5.3.1.1 Make Routine

‘Figure 5.3.1.1(a)’ illustrates the form for selecting batch, semester, and session for creating a schedule.

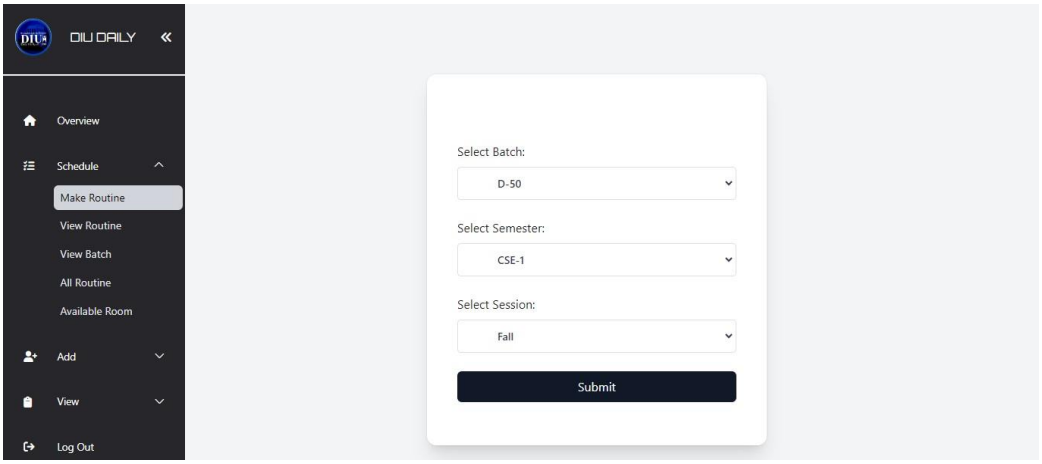

Figure 5.3.1.1(a): Routine Generation for Our System

‘Figure 5.3.1(b)’ also showcases the form used for selecting course, day, time, room, and teacher when creating a schedule.

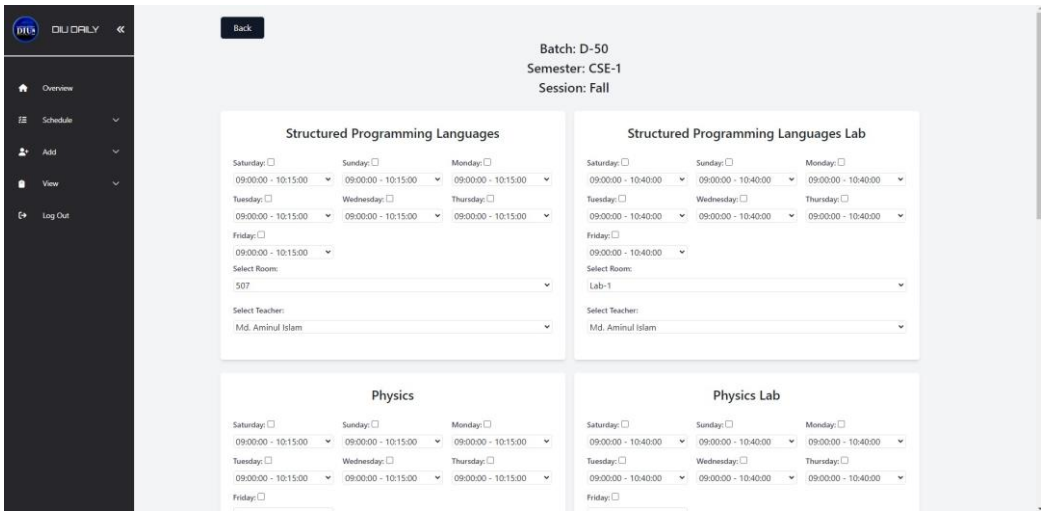

Figure 5.3.1.1(b): Routine Generation for Our System

5.3.1.2 View Routine

‘Figure 5.3.1.2’ depicts the view routine page within the admin interface, allowing administrators to update and delete specific functions for efficient management.

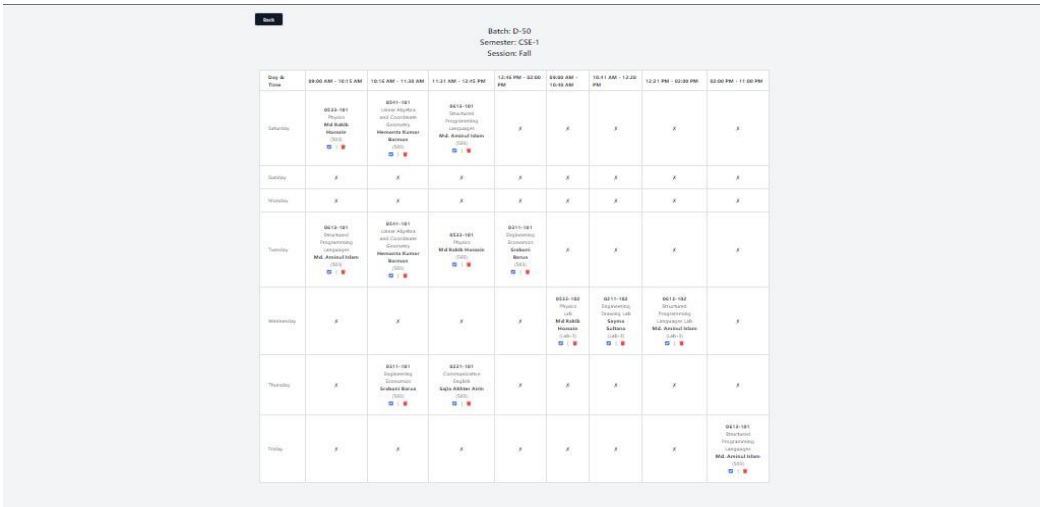

Figure 5.3.1.2: Admin Routine Management

5.3.1.3 All Routine

‘Figure 5.3.1.3’ showcases the comprehensive view of all schedules and routines within our system.

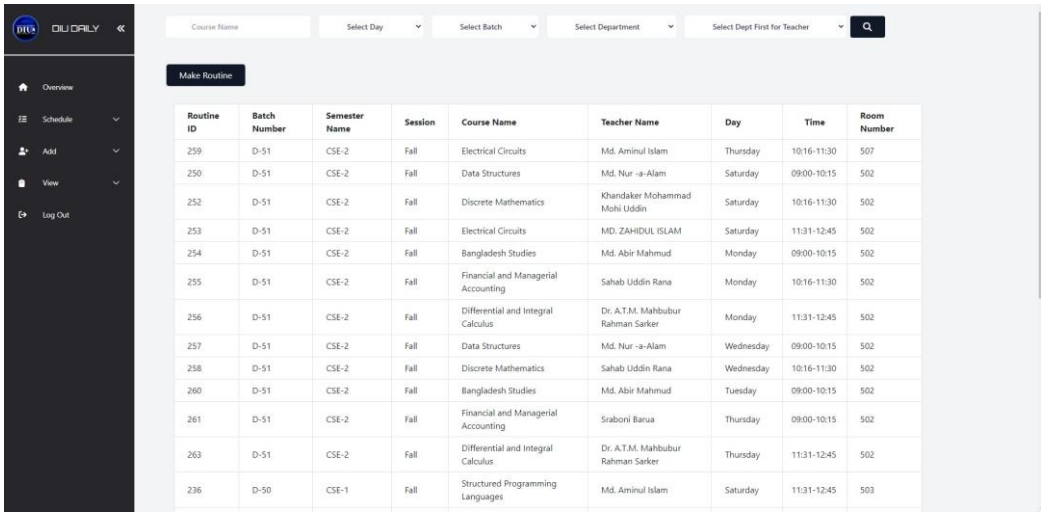

Figure 5.3.1.3: Comprehensive Routine Overview

5.3.2 Add

'**Figure 5.3.2'** visually represents the add dashboard within our system. This dashboard is a crucial part of the administrative interface, allowing administrators to add various entities and data relevant to the functioning of the educational institution. It provides a user-friendly and structured environment for inputting essential information, ensuring accuracy and efficiency in data management.

5.3.2.1 Add Teacher

'Figure 5.3.2.1' illustrates the user interface for adding a new teacher to the system, providing a form.

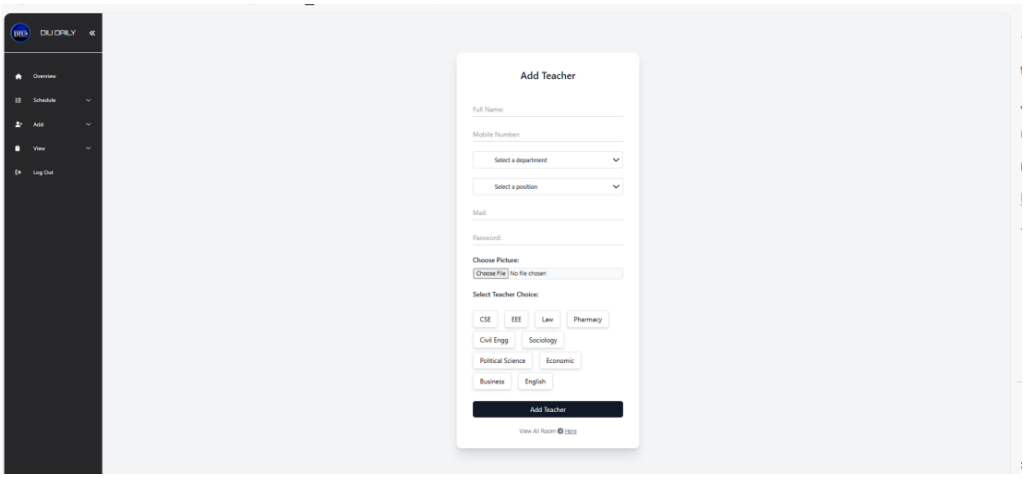

Figure 5.3.2.1: Add New Teacher Form

5.3.2.2 Add Batch

‘Figure 5.3.2.2’ displays the user interface for adding a new batch to the system, offering a form for entering batch-related details.

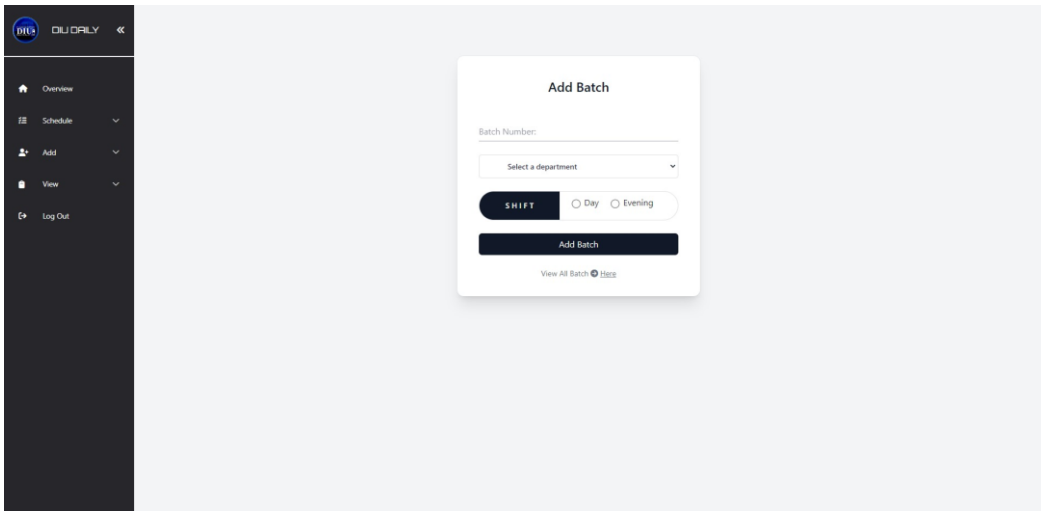

Figure 5.3.2.2: Add Batch Form

5.3.2.3 Add Department

‘Figure 5.3.2.3’ displays the user interface for adding a new department to the system.

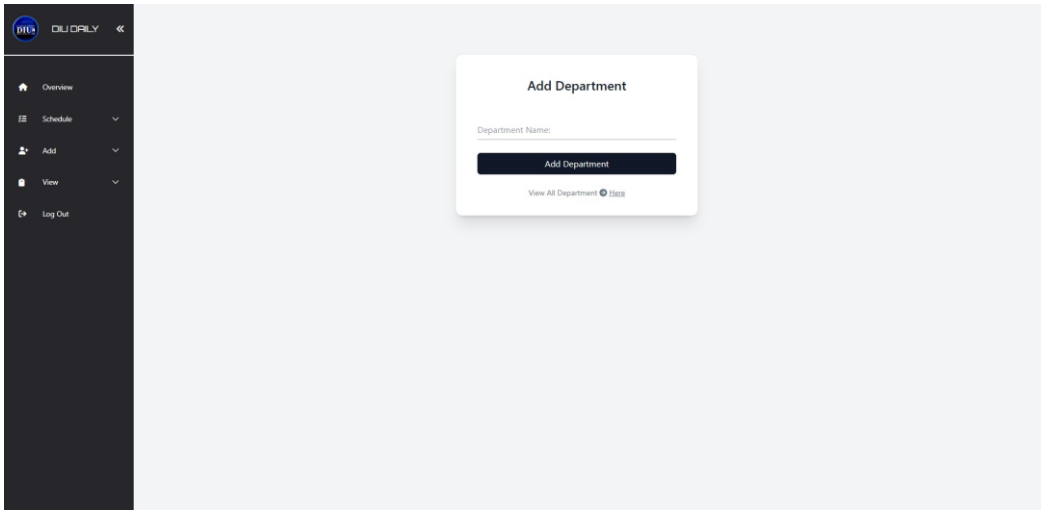

Figure 5.3.2.3: Add Department Form

5.3.2.4 Add Semester

‘Figure 5.3.2.4’ displays the user interface for adding a new semester to the system.

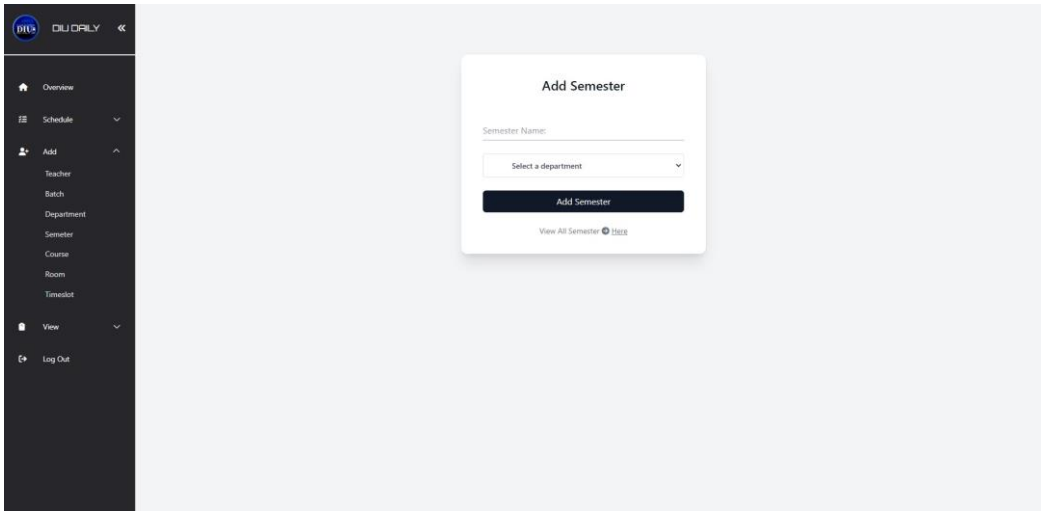

Figure 5.3.2.4: Add Semester Form

5.3.2.5 Add Course

'Figure 5.3.2.5' displays the user interface for adding a new course to the system.

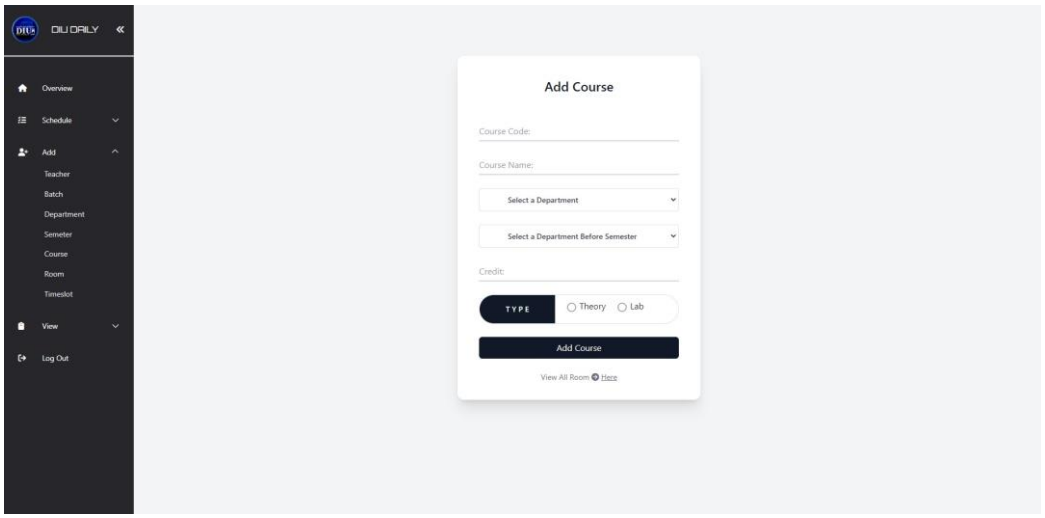

Figure 5.3.2.5: Add Course Form

5.3.2.6 Add Room

'Figure 5.3.2.6' displays the user interface for adding a new room to the system.

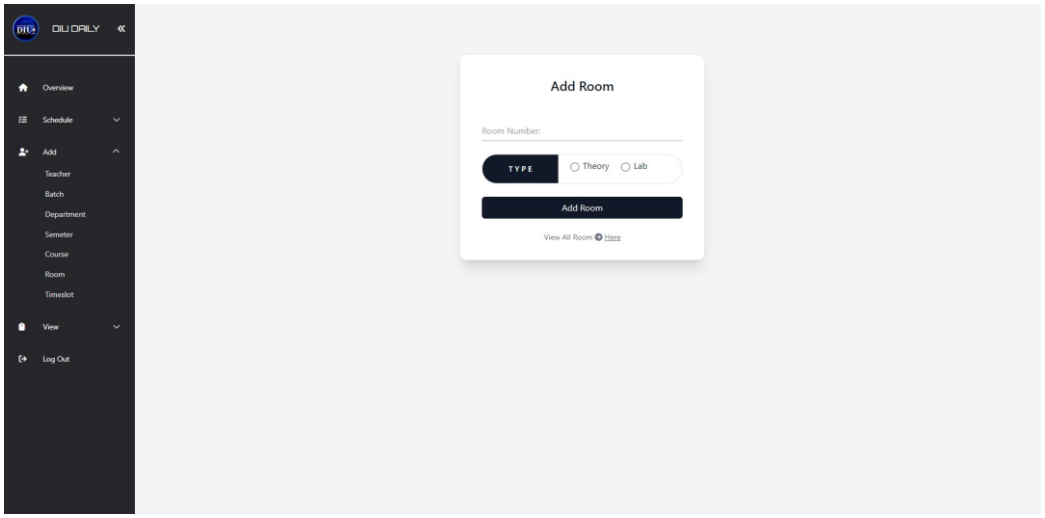

Figure 5.3.2.6: Add Room Form

5.3.2.7 Add Timeslot

'Figure 5.3.2.7' displays the user interface for adding a new timeslot to the system.

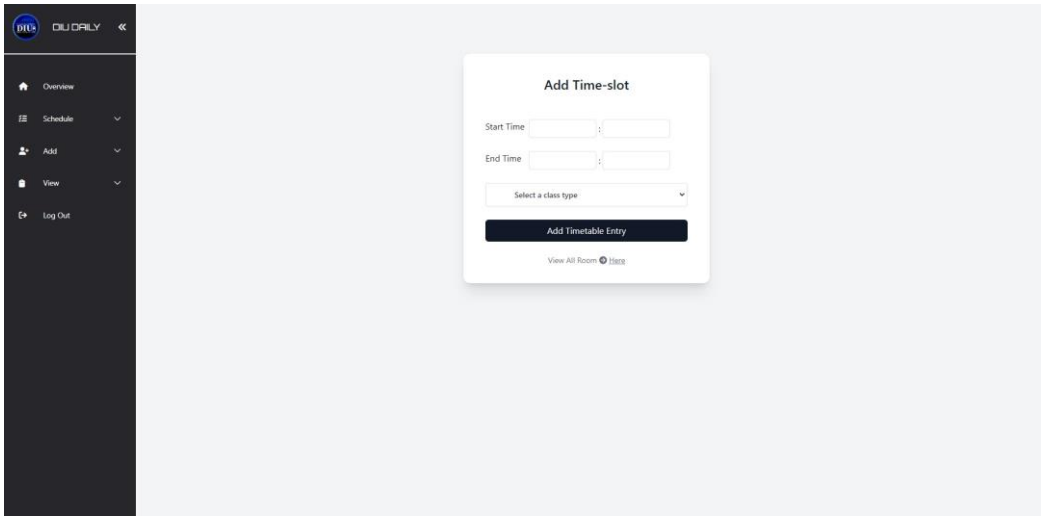

Figure 5.3.2.7: Add Timeslot Form

5.3.3 View

'**Figure 5.3.3'** visually represents the administrative view dashboard within our system. This dashboard provides administrators with a comprehensive overview of essential statistics and data related to the educational institution. It serves as a central hub for accessing various administrative functionalities and making informed decisions to optimize academic operations.

5.3.3.1 View Page for Adding New Teacher

'Figure 5.3.3.1' represents the user interface for adding new teachers to the system, providing a

view page for entering teacher-related information.

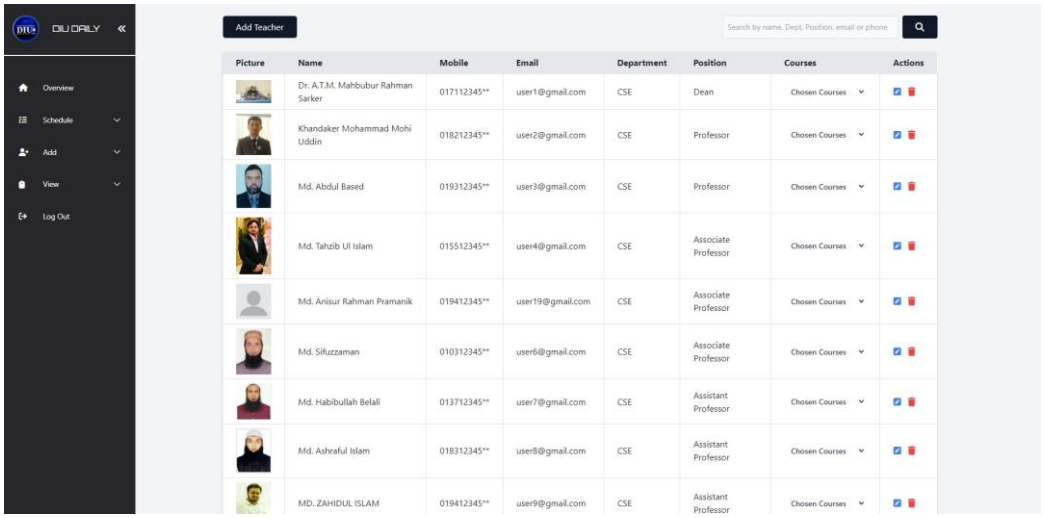

Figure 5.3.3.1: View Page for Adding New Teacher

5.3.3.2 View Page for Adding New Batch

'Figure 5.3.3.2' illustrates the view page for adding new batches to the system, allowing the input of batch-related details.

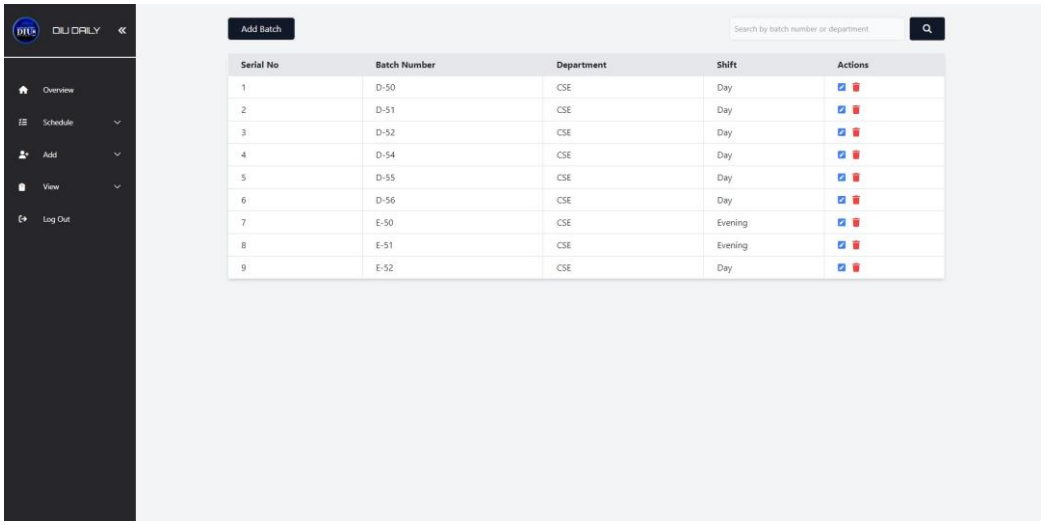

Figure 5.3.3.2: View Page for Adding New Batch

5.3.3.3 View Page for Adding New Department

'Figure 5.3.3.3' illustrates the view page designed for adding new departments to the system, enabling the entry of department-specific information.

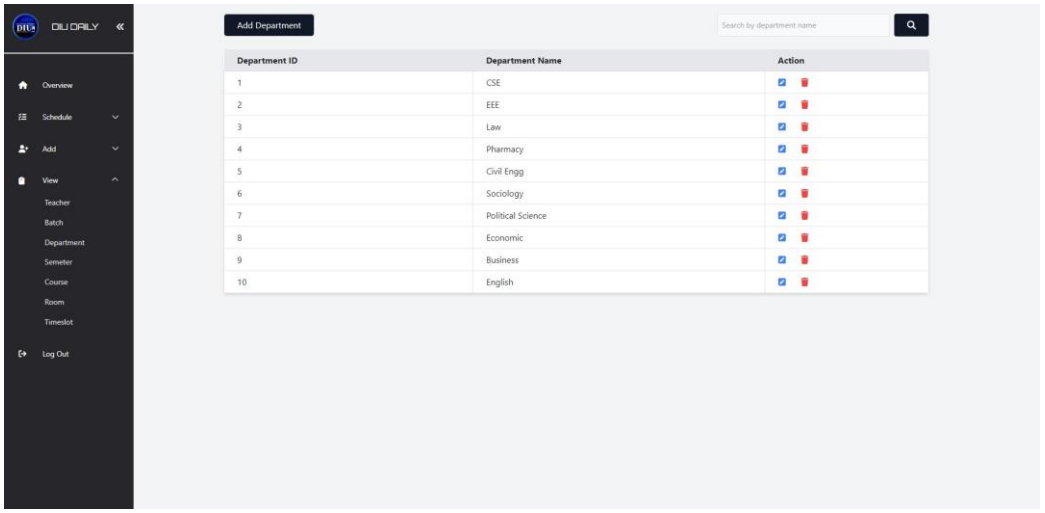

Figure 5.3.3.3: View Page for Adding New Department

5.3.3.4 View Page for Adding New Semester

'Figure 5.3.3.4' illustrates the view page designed for adding new semester to the system.

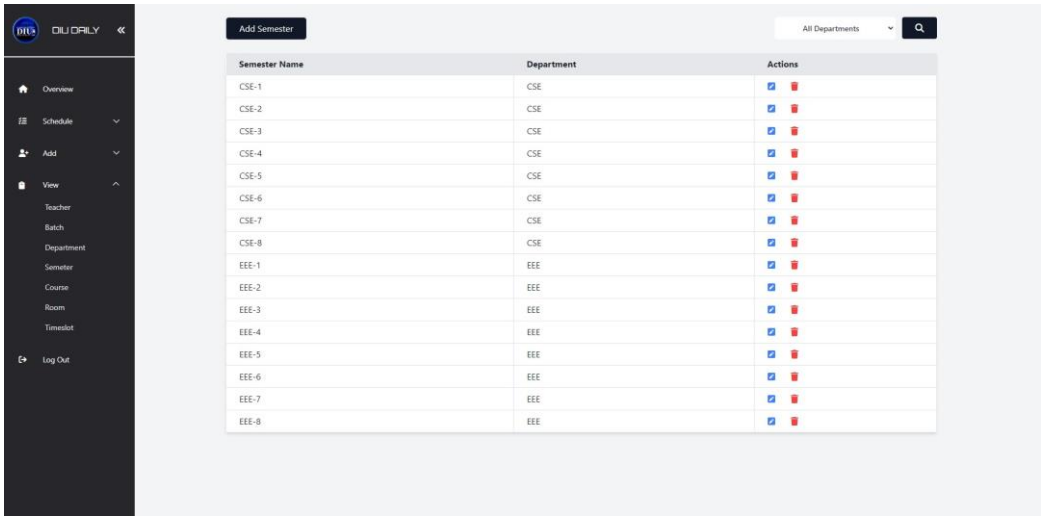

Figure 5.3.3.4: View Page for Adding New Semester

5.3.3.5 View Page for Adding New Course

'Figure 5.3.3.5' illustrates the view page designed for adding new course to the system.

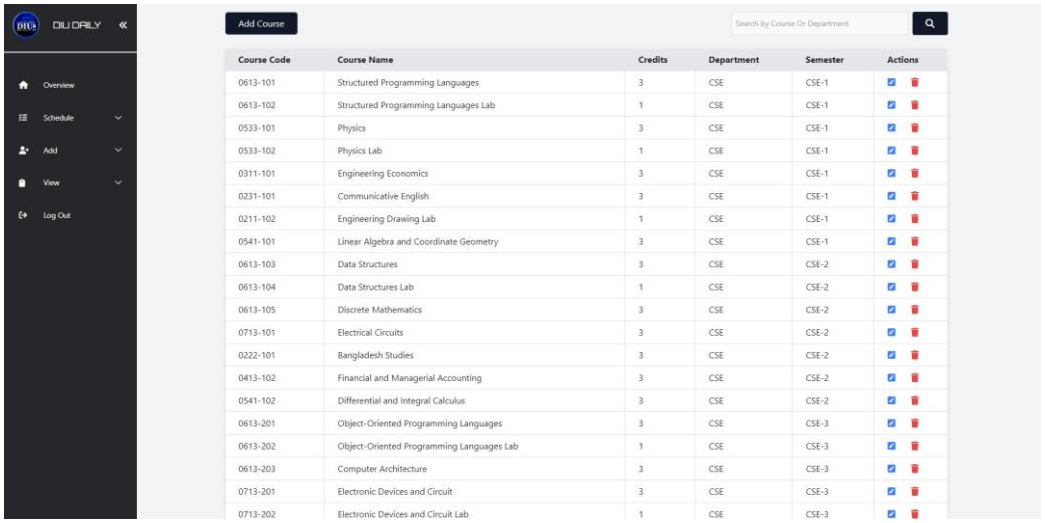

Figure 5.3.3.5: View Page for Adding New Course

5.3.3.6 View Page for Adding New Room

'Figure 5.3.3.6' illustrates the view page designed for adding new room to the system.

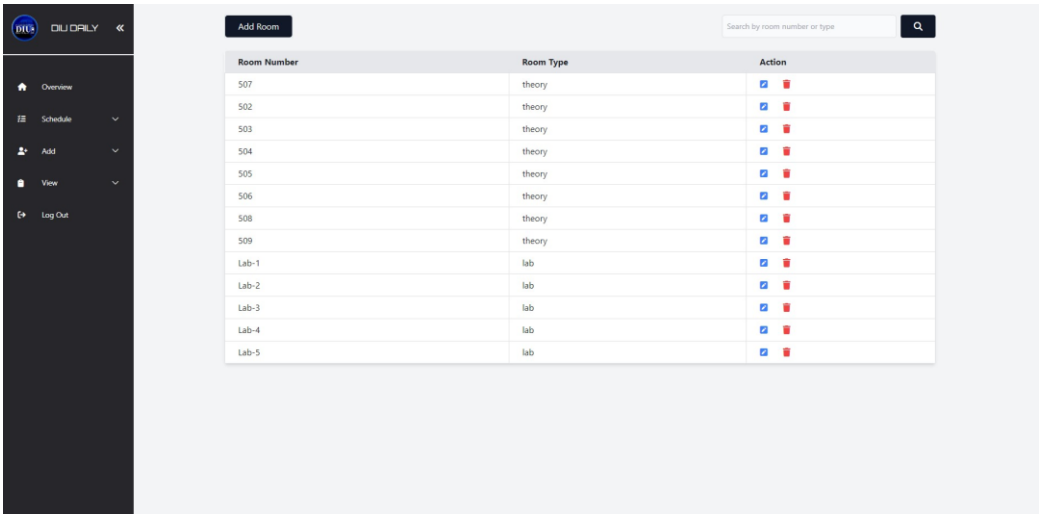

Figure 5.3.3.6: View Page for Adding New Room

5.3.3.7 View Page for Adding New Timeslot

'Figure 5.3.3.7' illustrates the view page designed for adding new timeslot to the system.

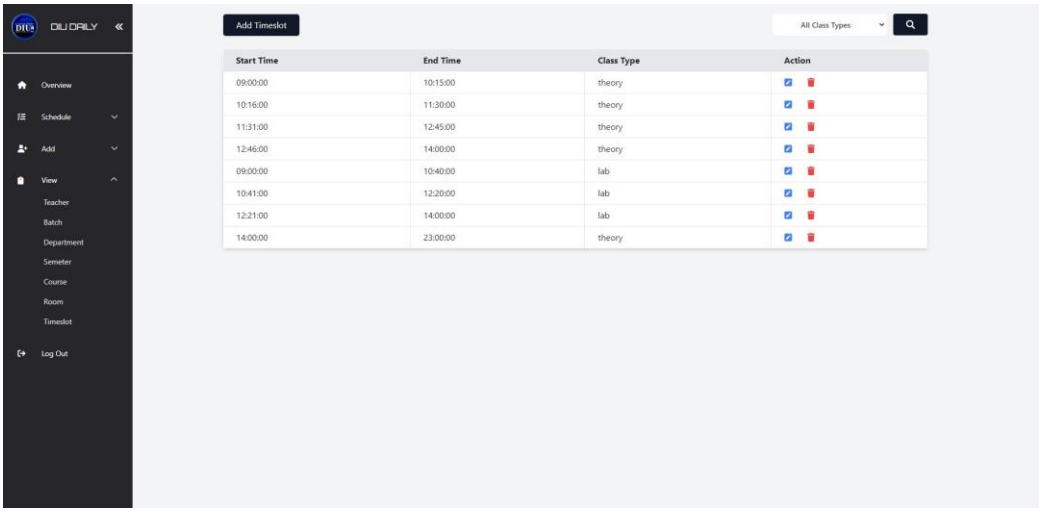

Figure 5.3.3.7: View Page for Adding New Timeslot

5.4 LOGIN PAGE FOR TEACHER

'**Figure 5.4'** visually represents the login page designed specifically for teachers within our system. This login page serves as the entry point for teachers, ensuring secure access to their respective accounts and the functionalities provided to them within the platform. Teacher Login page of our project.

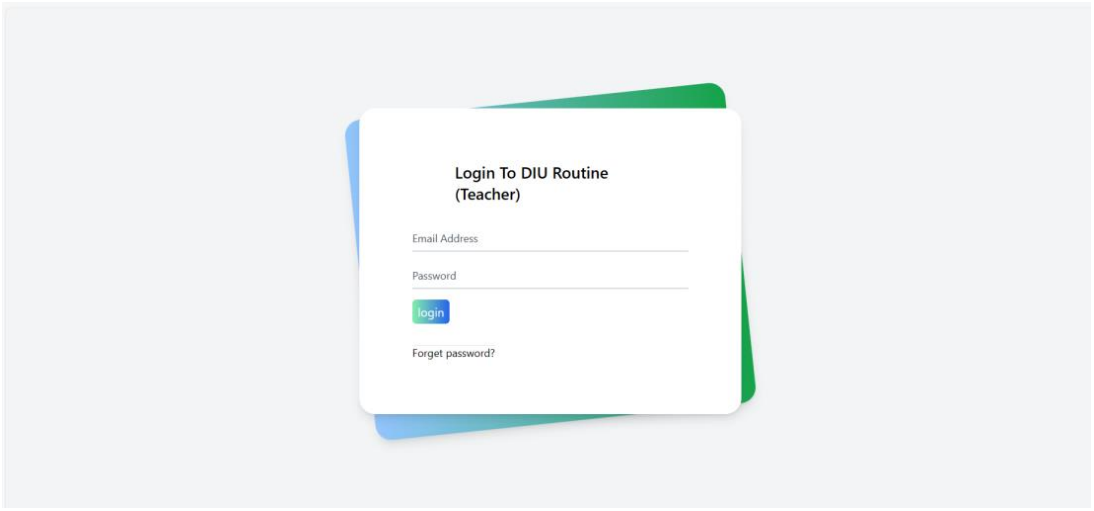

Figure 5.4: Login Page for Teacher

5.4.1 Profile

'Figure 5.4.1' provides a visual representation of the teacher's profile feature within our system. This feature allows teachers to access and view their profile information, enhancing their interaction with the platform and ensuring accurate record-keeping.

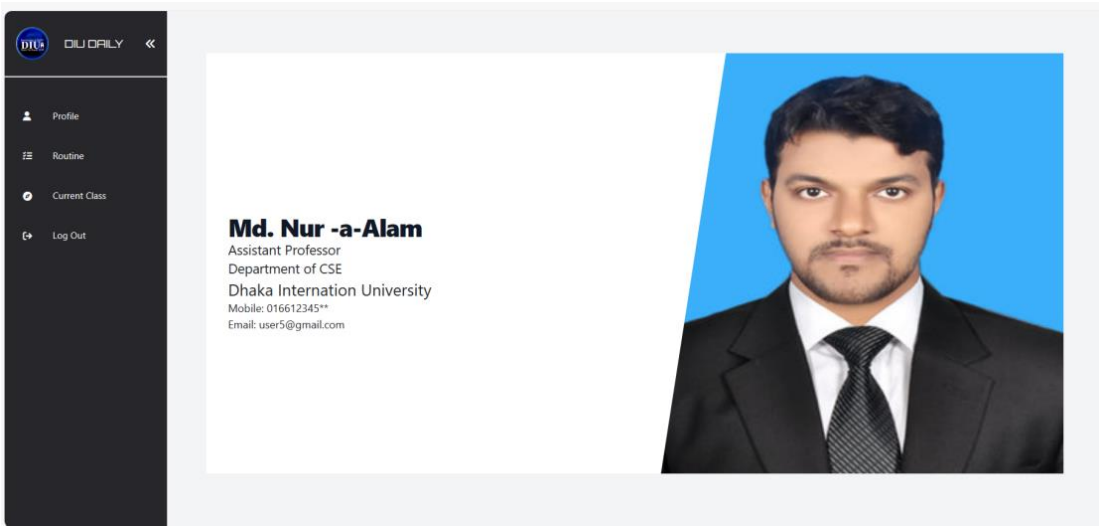

Figure 5.4.1: Profile Page for Teacher

5.4.2 Routine (Teacher)

'Figure 5.4.2' visually illustrates the teacher's routine feature within our system. This feature provides teachers with a convenient and organized view of their class schedules, enhancing their ability to manage their academic responsibilities efficiently.

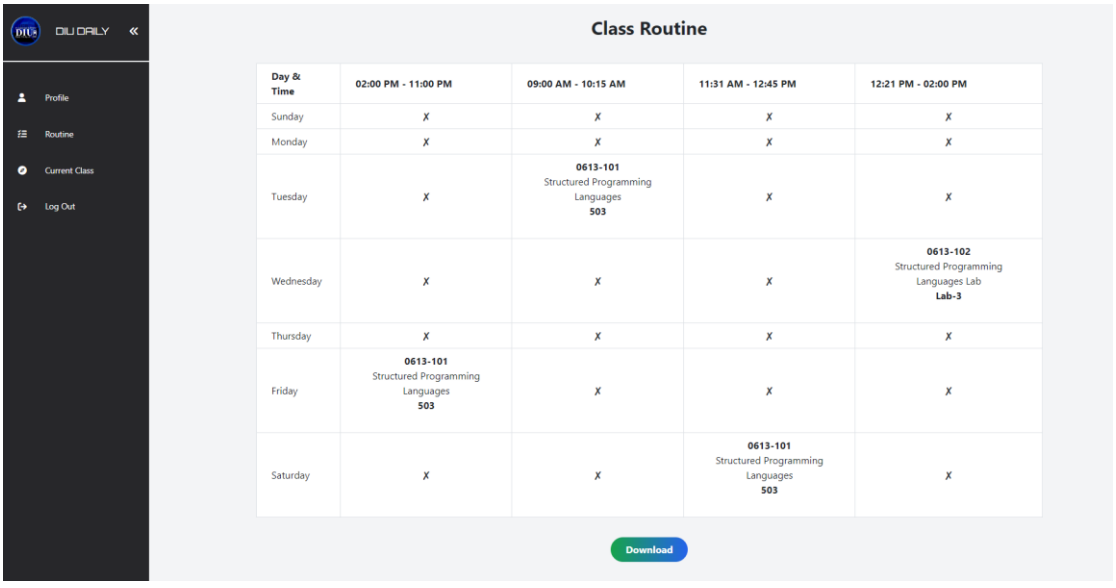

Figure 5.4.2: Teacher's Daily Routine

5.4.3 Current Class

'Figure 5.4.3' illustrates the current class feature tailored for teachers within our system. This functionality empowers teachers to efficiently manage their ongoing classes, ensuring a smooth teaching experience.

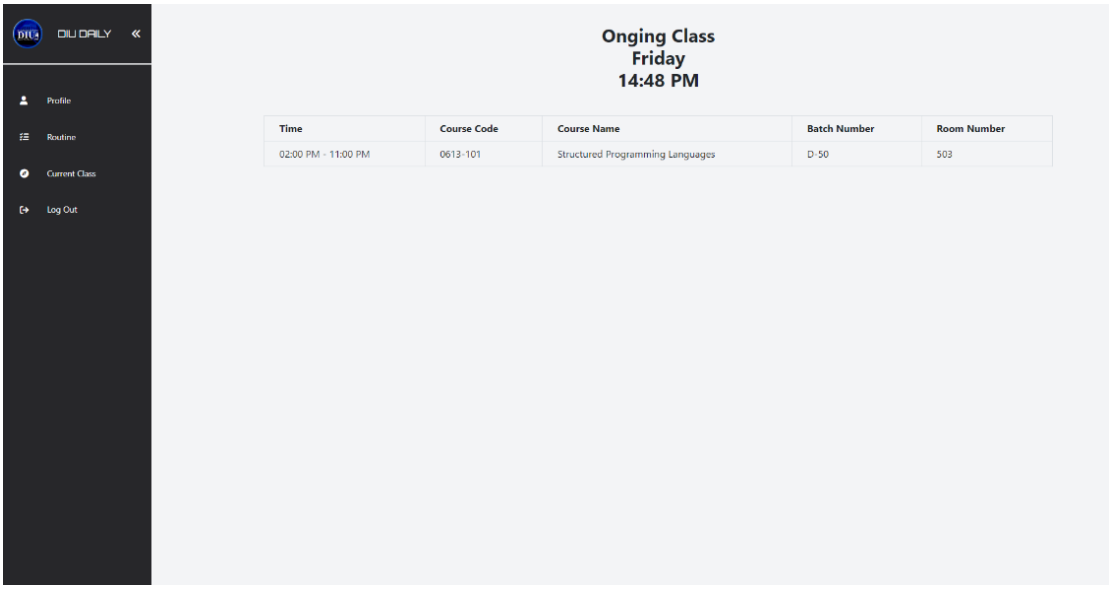

Figure 5.4.3: Current Class View for Teachers

5.5 STUDENT DASHBOARD

The student dashboard, as depicted in **'Figure 5.5'** is a central component of our project. It serves as the gateway for students to access their academic schedules, making their educational journey more streamlined and user-friendly.

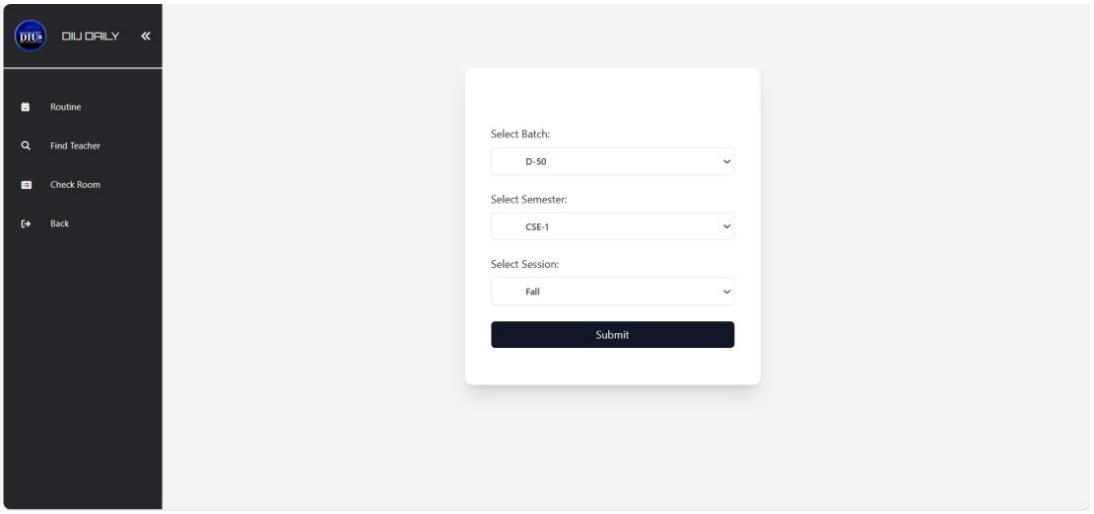

Figure 5.5: Student Dashboard of Our System

5.5.1 Routine (Student)

'**Figure 5.5.1'** provides a visual representation of the student routine system within our project.

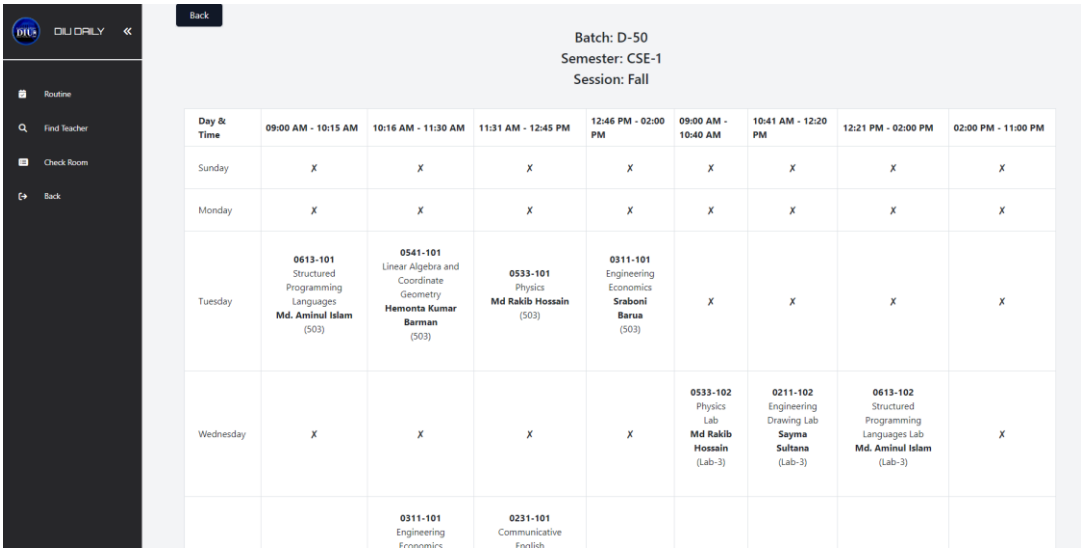

Figure 5.5.1: Student Class Schedule

5.5.2 Find Teacher

'Figure 5.5.2' visually presents the teacher finding system available to students within our project. This feature empowers students to search and locate their teachers efficiently, facilitating communication and interaction within the educational environment.

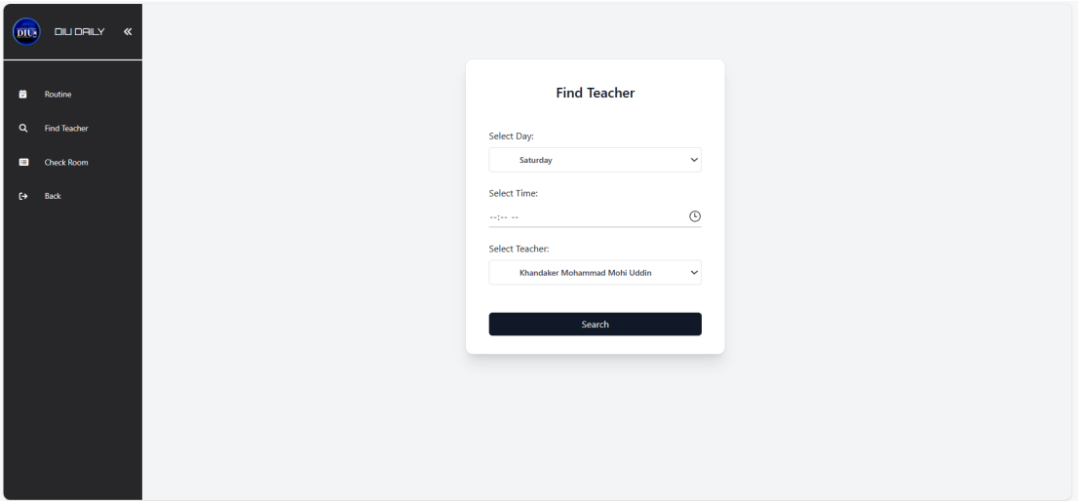

Figure 5.5.2: Find Teacher Availability

5.5.3 Check Room

'Figure 5.5.3' visually represents the room availability feature designed for students in our system. This feature enables students to check the availability of rooms for various activities, ensuring they can efficiently plan and manage their academic and extracurricular tasks within the educational institution.

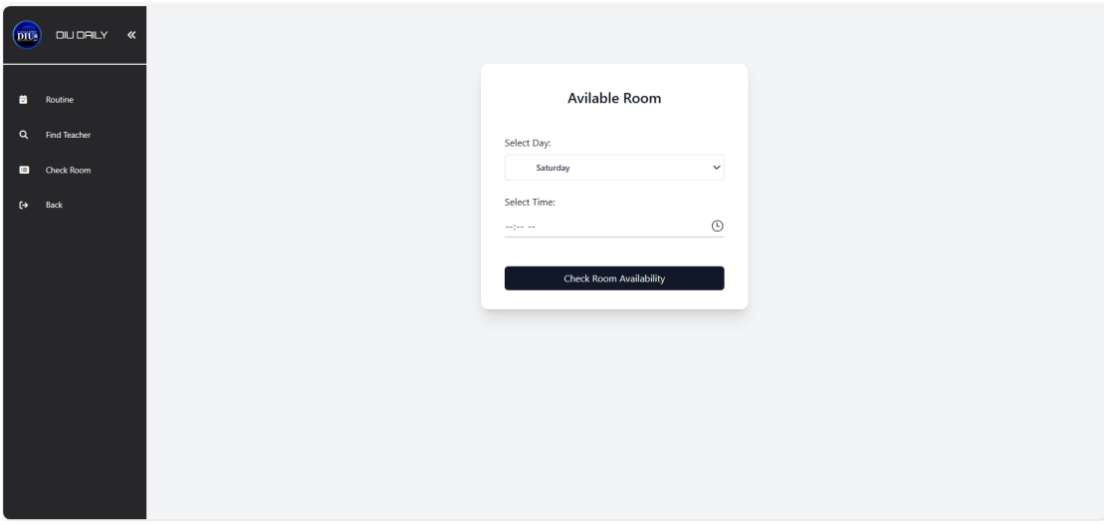

Figure 5.5.3: Room Availability

5.6 Non-Functional Requirements

All of the application data is stored in a MySQL database, and therefore a MYSQL Database must also be installed on the host computer.

The server hardware can be any computer capable of running both the web and database servers and handling the expected traffic. For a showroom that is not expecting to see much web traffic, an average personal computer may be appropriate. Once the site starts generating more hits, though, it will likely be necessary to upgrade to a dedicated host to ensure proper performance. The exact cutoffs will need to be determined through a more thorough stress testing of the system.

Chapter 6 CONCLUSION AND FUTURE DEVELOPMENT

6.1 Conclusion & Future Work

In conclusion, our Web Based Class Schedule Management System represents a significant leap forward in academic management within educational institutions. This web-based platform has successfully modernized the intricate process of class scheduling, offering customized interfaces tailored to administrators, teachers, and students. Its impact on educational institutions is profound, simplifying administrative tasks, enabling data-informed decision-making, and enhancing the overall academic experience.

Administrators have experienced a notable reduction in their administrative workload, allowing them to redirect their efforts toward strategic planning and decision-making. Teachers benefit from a user-friendly interface that simplifies scheduling and academic management tasks, while students can effortlessly access and organize their class schedules and academic commitments.

From a technological standpoint, the system is built on a robust stack comprising HTML, CSS, Bootstrap, advanced PHP, PHPMailer, dompdf, JavaScript, Ajax and jQuery. The utilization of a MySQL database on the backend ensures efficient data storage and retrieval, even under substantial user loads.

6.2 Future Development

The following section describes the work that will be implemented with future releases of the software.

- Mobile Optimization: Developing a mobile-responsive version of the Online Class Schedule Generator can improve accessibility, catering to users who prefer mobile devices.

- Advanced Reporting: Expanding reporting capabilities can offer administrators and teachers more comprehensive insights into scheduling patterns, resource utilization, and academic performance trends.

- Continuous Feedback Loop: Establishing a feedback mechanism that regularly collects input from administrators, teachers, and students will guide ongoing improvements to the system's interface and functionality.

- Integration Opportunities: Exploring integration options with existing educational software or systems can further streamline academic operations and data management.

- Security Enhancements: Enhancing security measures is crucial to safeguard user data and privacy, ensuring compliance with evolving regulations and standards.

6.3 Limitations

It is important to acknowledge the system's inherent limitations:

- Dependency on Internet Connectivity: The system relies on internet access, which may pose challenges in areas with limited connectivity.

- User Training: Users may require training to fully utilize the system's capabilities and features.

- Scalability Considerations: As user numbers increase, continuous monitoring and optimization are necessary to maintain system performance and scalability.

- Security Challenges: Ensuring data security and privacy remains an ongoing effort to protect sensitive user information. The following section describes the work that will be implemented with future releases of the software.

Web Based Class Schedule Management System Generator has already delivered substantial benefits to educational institutions. However, there is ample room for further enhancements and ongoing maintenance to ensure its continued relevance and effectiveness in meeting the dynamic needs of the educational sector.

REFERENCES

[1] https://www.academia.edu/41654011/DEPARTMENT_OF_COMPUTER_SCIENCE_Project_on_Web_Based_Class_Scheduling_System_for_ADU_Final_project_II

[2] https://www.academia.edu/34447178/Online_class_scheduling_system

[3] https://www.researchgate.net/publication/335685036_Class_Schedule_System

[4] https://www.scribd.com/document/532282985/Class-Scheduling-and-Management-System-for-SAC

[5] http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/2875/P12366%20%2818%25%29.pdf?sequence=1&isAllowed=y

[6] https://www.softwaretestinghelp.com/software-development-life-cycle-sdlc//

[7] https://www.theserverside.com/definition/HTML-Hypertext-Markup-Language/

[8] https://www.mecs-press.org/ijeme/ijeme-v12-n2/v12n2-5.html

[9] https://www.geeksforgeeks.org/tailwind-css/

[10] https://www.w3.org/standards/webdesign/htmlcss

[11] https://www.php.net/manual/en/history.php.php

[12] http://www.fpdf.org/

[13] https://techterms.com/definition/javascript

[14] https://www.w3schools.com/xml/ajax_intro.asp

[15] https://techterms.com/definition/jquery#:~:text=jQuery

[16] https://findwords.info/term/xampp

[17] https://code.visualstudio.com/docs/introvideos/basics

[18] https://www.apachefriends.org/index.html

[19] https://www.w3schools.com/php/default.asp

[20] https://getbootstrap.com/docs/4.0/getting-started/introduction/

END