AUTOMATION OF DEPARTMENTAL COURSE ALLOCATION AND TIME TABLE

BY

MOLAKE BENEDICT AGAJI
(ST/CS/ND/21/072)

DEPARTMENT OF COMPUTER SCIENCE,
SCHOOL OF SCIENCE AND TECHNOLOGY,
FEDERAL POLYTECHNIC, MUBI, ADAMAWA STATE.

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF NATIONAL DIPLOMA (ND) IN COMPUTER SCIENCE.

SEPTEMBER, 2023

DECLARATION

I hereby declare that the work in this project titled “**Automation of Departmental Course Allocation and Time Table”** was performed by me under the supervision of Mr. Simon Galadima. The information derived from literatures has been duly acknowledged in the text and a list of references provided. The work embodied in this project is original and had not been submitted in part or in full for any other diploma or certificate of this or any other institution.

MOLAKE BENEDICT AGAJI ________________ ________________
(ST/CS/ND/21/072) Signature Date

CERTIFICATION

This project titled “**Automation of Departmental Course Allocation and Time Table”** meets the regulations governing the award of National Diploma (ND) in Computer Science, Federal Polytechnic Mubi, Adamawa State

Mr. Simon Galadima
(Project Supervisor) Sign/Date

Mr. Mustapha Kassim.
(Head of Department) Sign/Date

Mal. Abdulrahman Saidu
(External Examiner) Sign/Date

DEDICATION

This project is dedicated to my beloved parents for their advice, encouragement and financial support towards my academic pursuit.

ACKNOWLEDGEMENTS

I want to acknowledge Almighty God for his infinite mercy and protection throughout my academic activities. And for the understanding in achieving our academic success.

I also recognize my Supervisor Mr. Simon Galadima who took time, despite his busy schedule to direct and guide me throughout this research work.

I also acknowledge the Head of Department Computer Science Mr. Mustapha Kassim for his moral encouragement throughout my period of study. I also acknowledge all Staff of Computer Science Department for their support and encouragement and the knowledge they've impacted on me throughout my studies.

I also want to appreciate my lovely parents for their love and care and for giving me the opportunity to be trained and achieve my dreams.

Finally, I appreciate the efforts of my Uncles and aunties, for their encouragement and support throughout the course of my study and also my friends and relatives, course mates and all well-wishers. I love you all, may the Almighty God bless you abundantly, Amen.

TABLE OF CONTENTS

TITLE PAGE .... **i**

DECLARATION .... **ii**

CERTIFICATION .... **iii**

DEDICATION .... **iv**

ACKNOWLEDGEMENTS .... **v**

TABLE OF CONTENTS .... **vi**

LIST OF FIGURES .... **viii**

LIST OF TABLES .... **ix**

ABSTRACT .... **x**

CHAPTER THREE **11**

SYSTEM DESIGN AND ANALYSIS **11**

3.1 Introduction 11

3.2 Disadvantages of the existing system 11

3.3 Advantages of the proposed system 11

3.4 The Proposed Method 11

3.5 Method of Data Collection 13

3.6 System Design 13

3.6.1 Algorithm diagram 13

3.6.2 System Architecture 14

3.6.3 Database Tables/Queries Structures 14

3.6.4 Entity Relationship Modelling 15

3.6.5 Input and Output Design 16

3.7 System Requirement Specification 18

3.7.1 Hardware Requirements 18

3.7.2 Software Requirements 18

3.7.3 Personnel Requirement 18

CHAPTER FOUR **19**

RESULTS AND DISCUSSION **19**

4.1 Introduction 19

4.2 Results 19

4.2.1 Welcome Interface 19

4.2.2 Login Interface 19

4.2.3 Add Lecturer Interface 20

4.2.4 Add Course Interface 20

4.2.5 Generate Time Table Interface 21

4.2.7 List of Lecturers 22

4.3 Discussion 22

4.4 User manual 23

CHAPTER FIVE 24

SUMMARY, CONCLUSION AND RECOMMENDATIONS 24

5.1 Summary 24

5.2 Conclusion 24

5.3 Recommendations 24

5.4 Contribution to Knowledge 25

5.5 Area for Further Work 25

REFERENCES 26

APPENDICES 28

LIST OF FIGURES

Figure 3.1: Waterfall model - - - - - - - - 12

Figure 3.2: Use Case Diagram- - - - - - - - 13

Figure 3.3: System Architecture- - - - - - - - 14

Figure 3.4: Entity Relationship Modelling- - - - - - - 15

Figure 3.5: Add Lecturer- - - - - - - - -- 16

Figure 3.6: Login form- - - - - - - - - 16

Figure 3.7: Add Lecturer- - - - - - - - - 17

Figure 3.8: Add Lecture Hall - - - - - - - - 17

Figure 4.1: Welcome Interface- - - - - - - - 19

Figure 4.2: Login page interface- - - - - - - - 19

Figure 4.3: Add Lecturer Interface- - - - - - - - 20

Figure 4.4: Add Course Interface- - - - - - - - 20

Figure 4.5: Generate Time Table Interface- - - - - - - 21

Figure 4.6: View Time Table Interface- - - - - - - 21

Figure 4.7: List of Lecturers Interface- - - - - - - 22

LIST OF TABLES

Table 1: Admin Details - - - - - - - - 14

Table 2: Lecture Halls - - - - - - - - 14

Table 3: Lecturer’s Details - - - - - - - - 15

Table 4: Courses Details - - - - - - - - 15

ABSTRACT

The "Automation of Departmental Course Allocation and Time Table" represents a pivotal advancement in the realm of academic administration and resource optimization within educational institutions. This transformative system is designed to streamline the complex and often labor-intensive processes of course allocation and timetable generation. By leveraging cutting-edge technology, automation, and data-driven decision-making, this solution enhances efficiency, reduces errors, and fosters a more productive learning environment. This innovative system offers a comprehensive approach to course allocation, considering various parameters such as course offerings, faculty availability, room assignments, and student preferences. Through automated algorithms and intelligent scheduling, it optimizes resource utilization and minimizes scheduling conflicts, resulting in a well-organized and efficient academic calendar. Key features of the system include user-friendly interfaces for administrators, faculty, and students, as well as real-time access to schedules and communication tools. The system also enhances transparency and accountability in the course allocation process, promoting fairness and equity among faculty members. The successful implementation of this automation system not only simplifies administrative tasks but also empowers educational institutions to adapt to dynamic scheduling challenges and changes. It offers a model for other academic departments and institutions seeking to modernize and optimize their course allocation and timetable management processes. In conclusion, the "Automation of Departmental Course Allocation and Time Table" embodies the synergy of technology and academia, demonstrating how automation can revolutionize administrative operations and positively impact the educational experience. It serves as a beacon of innovation in educational technology, offering a pathway toward greater efficiency, resource optimization, and academic excellence within academic institutions.

.

CHAPTER ONE

INTRODUCTION

1.1 Background to the Study

Departmental course allocation and timetable creation is an essential administrative task in educational institutions. It involves assigning courses to faculty members, scheduling classes, and creating a comprehensive timetable that meets the requirements of both students and faculty. Traditionally, this process has been carried out manually, consuming significant time and effort from administrators and resulting in potential errors and inconsistencies. However, with recent advancements in technology and the rise of automation, there is an opportunity to streamline and optimize this process through the automation of departmental course allocation and timetable creation. Automation in educational institutions has gained significant attention in recent years, with various studies focusing on the development of intelligent systems and algorithms to automate administrative tasks. These automation efforts aim to reduce the burden on administrators, improve efficiency, and enhance the overall quality of the educational process. One area where automation has shown great potential is the departmental course allocation and timetable creation process (Zhang *et al*., 2020).

Recent studies have explored different approaches to automate the course allocation and timetable creation process. For example, *Li et al.* (2021), proposed a multi-objective optimization model for faculty course assignments, considering factors such as expertise, workload, and availability. The model used genetic algorithms to generate optimal course assignments, resulting in improved resource utilization and faculty satisfaction.

Another study by Zhang *et al.* (2020) introduced a hybrid algorithm combining genetic algorithms and simulated annealing to optimize the timetable creation process. Their approach considered various constraints such as classroom capacities, time slot availability, and student preferences. The results demonstrated that the automated system outperformed manual methods in terms of time efficiency and timetable quality.

In addition to optimizing the allocation and timetable creation process, recent studies have also focused on incorporating flexibility and adaptability into automated systems Gao *et al.* (2022) proposed an intelligent course allocation system that can dynamically adjust faculty assignments and timetables in real-time. The system utilized machine learning techniques to analyze historical data and make proactive adjustments based on changes in faculty assignments or course offerings. This dynamic approach ensures that the timetable remains up-to-date and can accommodate unforeseen changes effectively.

Overall, the background research indicates a clear need for the automation of departmental course allocation and timetable creation processes in educational institutions. By addressing the limitations of manual methods and leveraging recent advancements in technology, automated systems have the potential to revolutionize these administrative tasks and contribute to the overall efficiency and effectiveness of educational institutions.

1.2 Problem Statement

The problem statement for the study on the automation of departmental course allocation and timetable creation in educational institutions can be summarized as follows:

i. Manual and Labor-Intensive Process: The traditional process of departmental course allocation and timetable creation is time-consuming and labor-intensive.

ii. Potential Errors and Inconsistencies: The manual nature of the process increases the likelihood of errors and inconsistencies in course allocations and timetable creation.

iii. Lack of Optimization: The absence of optimization techniques can lead to inefficient allocations, which can impact the quality and effectiveness of the educational experience.

iv. Inflexibility to Changes: Modifications require extensive manual adjustments, which can be time-consuming and prone to errors, making it difficult to maintain an up-to-date and accurate timetable.

v. Time-Consuming Revisions: This elongates the course allocation and timetable creation process, delaying the availability of the final timetable and potentially causing inconvenience to faculty members and students.

vi. Administrative Burden: The manual process places a significant burden on administrators responsible for course allocation and timetable creation.

1.3 Aim and Objectives

The aim of this project is to develop an automated system for departmental course allocation and timetable. Specific objectives include:

i. Develop a system that can assign courses to faculty members based on their expertise, workload, and availability.

ii. Design an algorithm that can generate an optimized timetable considering constraints such as classroom capacities, time slot availability, and faculty preferences.

iii. Implement a user-friendly interface that allows administrators to input and modify course information, faculty preferences, and other relevant data.

iv. Incorporate mechanisms for handling changes and updates in real-time, ensuring the system can adapt to modifications in faculty assignments or course offerings.

1.4 Significance of the Study

The automation of departmental course allocation and timetable creation offers several benefits to educational institutions. First, it reduces the time and effort required by administrators, allowing them to focus on more strategic tasks. Secondly, it minimizes the occurrence of errors and inconsistencies that can arise from manual processes, improving the overall quality of the timetable. Thirdly, it enhances the utilization of resources, such as faculty expertise and classroom capacities, leading to an optimized allocation. Additionally, the automated system enables flexibility and adaptability to changes, ensuring timely updates to the timetable when required. Overall, this study contributes to the advancement of administrative processes in educational institutions and provides a foundation for future research in the field of automation.

1.5 Scope of the Study

The scope of this study encompasses the automation of departmental course allocation and timetable creation in educational institutions. The research project focuses on developing an automated system that improves the efficiency, accuracy, and effectiveness of the allocation process, considering the various constraints and preferences involved. The system will primarily cater to the needs of administrators responsible for these tasks, while also taking into account the preferences and constraints of faculty members and students. The study will consider the following aspects; Course Allocation, Timetable Creation, User Interface and real time updates.

1.6 Definition of Some Operational Terms

Allocation: Allocation refers to the process of distributing or assigning resources, such as funds, time, personnel, or assets, to different individuals, groups, or activities based on predetermined criteria (Parry *et al.*, 2021).

Automation: In the context of departmental course allocation and timetable creation, automation involves utilizing software and algorithms to streamline and optimize the allocation process (Gao *et al.*, 2022).

Course: A course refers to a specific subject or topic of study that is offered within an educational institution (Parry *et al.*, 2021).

Department: A department is a distinct organizational unit within an educational institution or an organization that specializes in a specific field of study, discipline, or area of expertise (Parry *et al.*, 2021).

Departmental Course Allocation: Departmental course allocation refers to the process of assigning specific courses to lecturers within a department or academic unit of an educational institution (Li *et al.*, 2021).

Optimization: Optimization, in the context of course allocation and timetable creation, refers to the process of finding the best possible solution that maximizes resource utilization, minimizes conflicts, and satisfies various constraints and preferences (Zhang *et al.*, 2020).

Timetable Creation: Timetable creation involves designing and scheduling classes, lectures, and other academic activities within a specific timeframe (Zhang *et al*., 2020).

User Interface: The user interface (UI) is the graphical or visual component of the automated system that allows administrators to interact with and input data into the system (Li *et al.*, 2021).

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter presents a comprehensive review of the relevant literatures on the automation of departmental course allocation and timetable creation in educational institutions. The literature review aims to provide an overview of the current state of research, identify key concepts, methodologies, and technologies employed, and highlight gaps or areas for further investigation. By examining existing studies, this chapter lays the foundation for the development and implementation of the automated system in the subsequent chapters.

2.2 Automation in Educational Institutions

Automation has emerged as a transformative technology in educational institutions, revolutionizing administrative processes and enhancing the overall efficiency and effectiveness of operations. Recent research and developments have explored the application of automation in various areas within educational institutions, such as course management, grading, student support services, and resource allocation (Rahman *et al.*, 2022).

2.2.1 Course Management Automation

Course management encompasses tasks such as course registration, scheduling, and assignment of faculty. Automation has been employed to streamline these processes, reducing administrative burden and improving accuracy. Intelligent systems and algorithms have been developed to facilitate online course registration, optimize course scheduling, and automate faculty assignments.

Ma *et al.* (2022), developed an automated course registration system that utilized machine learning algorithms to predict student course preferences based on historical data. The system provided personalized course recommendations to students, resulting in improved course enrollment efficiency and student satisfaction.

2.2.2 Grading Automation

Automating the grading process can significantly reduce the time and effort required by instructors. Machine learning techniques have been employed to automate grading tasks, such as multiple-choice assessments, essays, and programming assignments. These systems analyze student responses and provide accurate and timely feedback.

A recent study by Parry *et al.* (2021), focused on automated essay scoring using natural language processing and machine learning algorithms. The system assessed essays based on predefined criteria, providing consistent and reliable grading, while also saving instructor time.

2.2.3 Student Support Services Automation

Automation has been applied to various student support services, enhancing accessibility and efficiency. Chatbots and virtual assistants have been implemented to provide automated responses to frequently asked questions, offer guidance on course selection, and provide academic advising support.

Kizilcec *et al.* (2021), conducted a study on the use of chatbots in academic advising. They developed an intelligent chatbot that utilized natural language processing and machine learning to provide personalized advice and assistance to students. The results showed that the chatbot was effective in reducing advising workload and improving student satisfaction.

2.2.4 Resource Allocation Automation

Automation has also been applied to optimize resource allocation within educational institutions. This includes the allocation of classrooms, faculty workload, and budgetary resources. Intelligent algorithms and optimization models have been developed to ensure efficient utilization of resources and maximize cost-effectiveness.

Liu *et al.* (2020), proposed an automated system for classroom allocation in universities. The system employed an optimization algorithm to allocate classrooms based on course requirements, classroom capacities, and time availability. The study demonstrated improved efficiency in classroom allocation, minimizing conflicts and maximizing resource utilization.

2.2.5 Administration and Workflow Automation

Automation has been employed to streamline administrative processes and workflows in educational institutions. This includes automating document management, data entry, and administrative tasks, reducing manual effort and enhancing efficiency.

A recent study by Rahman *et al.* (2022), focused on the automation of administrative workflows in higher education institutions. They developed an intelligent system that automated document processing and workflow management, resulting in significant time savings and improved accuracy in administrative tasks.

2.3 Automation of Departmental Course Allocation

The automation of departmental course allocation in educational institutions has gained significant attention due to its potential to streamline the allocation process, optimize resource utilization, and enhance faculty satisfaction. Recent research has explored various methodologies and techniques to automate the course allocation process, considering factors such as faculty expertise, workload, availability, and institutional constraints.

Li *et al.* (2021), proposed a multi-objective optimization model for faculty course assignments. The model considered factors such as faculty expertise, workload, and availability to generate optimal course allocations. The study demonstrated that the automated system achieved better resource utilization and increased faculty satisfaction compared to manual methods.

Machine learning techniques have been applied to automate course allocation by analyzing historical data and making predictions based on patterns and trends. These approaches leverage algorithms to learn from past course allocations and make informed decisions regarding future assignments. For instance, Al-Mogren *et al.* (2021), developed a machine learning-based approach for course allocation that considered faculty expertise, preferences, and institutional constraints. The system analyzed historical data to identify patterns and generate course allocations that matched faculty expertise and reduced conflicts. The results showed improved efficiency and accuracy compared to manual allocation processes.

Genetic algorithms have been employed to automate the course allocation process by simulating the process of natural selection and evolution. These algorithms iteratively generate and refine course allocations based on fitness functions and genetic operators, such as crossover and mutation. Liu *et al.* (2020), proposed an automated course allocation system that utilized a genetic algorithm to optimize faculty assignments. The system considered factors such as faculty expertise, preferences, and workload constraints. The study demonstrated that the genetic algorithm approach effectively generated course allocations that satisfied constraints and preferences while maximizing resource utilization.

Wang *et al.* (2022), developed an automated course allocation system based on constraint satisfaction programming. The system considered faculty preferences, course requirements, and classroom capacities to generate course allocations that met all constraints. The study showed that the automated system achieved better constraint satisfaction compared to manual allocation methods.

Gao *et al.* (2022), proposed an intelligent course allocation system that utilized machine learning techniques to analyze historical data and proactively adjust faculty assignments and timetables. The system dynamically adapted to changes, ensuring up-to-date and optimal course allocations. The study demonstrated the system's effectiveness in handling changes and reducing administrative burden.

2.4 Timetable Creation Automation

Automating the process of timetable creation in educational institutions has the potential to optimize resource utilization, reduce scheduling conflicts, and enhance overall efficiency. Recent research has focused on developing automated approaches for timetable creation, employing various algorithms and optimization techniques to generate optimized and conflict-free timetables.

Zhang *et al.* (2020), proposed a hybrid algorithm combining genetic algorithms and simulated annealing to optimize the timetable creation process. The algorithm considered constraints such as classroom capacities, time slot availability, and faculty preferences. The study demonstrated improved time efficiency and timetable quality compared to manual methods.

Jin and Chen (2022), developed an automated timetable creation system based on constraint satisfaction programming. The system considered constraints such as classroom capacities, faculty availability, and course requirements. The study showed that the automated system effectively generated timetables that satisfied all constraints, minimizing scheduling conflicts.

Chen *et al.* (2021), proposed a hybrid approach combining simulated annealing, ant colony optimization, and genetic algorithms for timetable creation. The approach considered constraints such as classroom capacities, course requirements, and faculty preferences. The study demonstrated that the hybrid approach produced timetables with higher quality and reduced conflicts compared to individual algorithms.

Ma *et al.* (2021), developed an automated timetable creation system that utilized machine learning algorithms to analyze historical scheduling data. The system learned from previous timetables and patterns, allowing it to generate optimized timetables that considered constraints such as classroom availability, faculty preferences, and course requirements. The study showed that the machine learning-based approach improved the efficiency and quality of timetable creation.

2.5 User Interface and System Implementation

The development and implementation of an automated system for course allocation and timetable creation require a user-friendly interface that allows administrators to input and modify data easily. Studies have focused on designing intuitive interfaces that facilitate efficient data entry and provide real-time feedback on course allocations and timetables.

Jin and Li (2021), proposed a user interface design framework for a course scheduling system. The framework focused on the layout, color scheme, and organization of the interface to optimize usability. The study found that a well-designed interface significantly improved user satisfaction and task completion time.

An example of real-time feedback implementation is demonstrated in the study by Wu *et al.* (2021), they developed an automated course scheduling system with a user interface that provided instant notifications to administrators about course conflicts, overlapping schedules, and resource utilization. The real-time feedback enabled administrators to quickly address scheduling issues, resulting in improved efficiency and accuracy.

In their research, Shahin *et al.* (2021), focused on the integration of an automated course allocation system with an existing student information system. The integration allowed for the transfer of student enrollment data, course prerequisites, and other relevant information. The study found that integrating the automated system with existing systems improved data accuracy and reduced duplication of efforts.

A study by *Lin et al.* (2022), investigated the impact of training on user acceptance of an automated course scheduling system. The research emphasized the importance of comprehensive training programs and user support in facilitating user acceptance and system utilization. The study found that well-trained users exhibited higher satisfaction and efficiency in utilizing the automated system.

Yang *et al*. (2021), conducted usability testing on an automated course registration system. The study involved collecting feedback from administrators and faculty members regarding the user interface, functionalities, and overall usability. The findings informed system improvements and adjustments to better align with user needs and preferences.

2.6 Evaluation of Automated Systems

The evaluation of automated systems for departmental course allocation and timetable creation is crucial to assess their effectiveness and performance. Studies have employed various metrics to evaluate the systems, including time efficiency, accuracy of course allocations, and user satisfaction. Comparative studies between automated systems and manual processes have highlighted the benefits

Zhou *et al*. (2021), conducted a study comparing the time efficiency of an automated course scheduling system with a traditional manual process. The study found that the automated system significantly reduced the time required for course scheduling tasks, leading to substantial time savings for administrators.

A study by Li *et al*. (2021), evaluated the accuracy of an automated faculty course assignment system. The system's accuracy was assessed by comparing the generated course allocations with predefined criteria, such as faculty expertise and workload. The study demonstrated that the automated system produced more accurate course assignments compared to manual methods.

Zhang et al. (2020) conducted a user satisfaction evaluation of an automated timetable creation system. The study surveyed administrators and faculty members regarding their perceptions and satisfaction with the system's functionalities, ease of use, and overall user experience. The findings indicated high levels of user satisfaction and acceptance of the automated system.

A comparative study by Sun *et al.* (2022) evaluated an automated course allocation system by comparing its performance with a manual allocation process. The study examined factors such as accuracy, efficiency, and user satisfaction. The results indicated that the automated system outperformed the manual process in terms of efficiency, accuracy, and user satisfaction.

A recent study by Nguyen *et al.* (2022) evaluated the long-term impact of an automated course allocation system in a university setting. The study analyzed resource utilization, faculty satisfaction, and student enrollment outcomes over multiple academic terms. The findings demonstrated significant improvements in resource optimization, faculty workload management, and student satisfaction over time.

2.7 Summary

This chapter has provided an overview of the literature on the automation of departmental course allocation and timetable creation. The studies reviewed highlight the benefits of automation in improving efficiency, accuracy, and resource utilization. They also emphasize the importance of user interfaces, system implementation, and the evaluation of automated systems. While the existing literature provides valuable insights into the automation of departmental course allocation and timetable creation, there are still gaps and areas for further investigation. Future research should explore the integration of advanced technologies such as artificial intelligence, machine learning, and data analytics to enhance the automation process. Additionally, the scalability and adaptability of automated systems to different educational institutions and the impact of automation on student outcomes and satisfaction deserve further exploration. The identified gaps and future research directions pave the way for the development and implementation of an automated system in the subsequent chapters.

CHAPTER THREE

SYSTEM DESIGN AND ANALYSIS

3.1 Introduction

This chapter contains the system design, the disadvantages of the existing system, the advantages of the proposed system over the existing system, the system requirements (Hardware and Software), the design and the system architecture.

3.2 Disadvantages of the existing system

While the existing manual system for departmental course allocation and timetable generation has served educational institutions for many years, it is not without its shortcomings. This chapter delves into the disadvantages of the current manual system, highlighting the pain points that an automated system aims to address.

i. One of the primary drawbacks of the existing manual system is its inefficiency in resource allocation.

ii. Manual data entry and manipulation are susceptible to human errors. Mistakes in recording faculty preferences, student enrollments, or room availability can lead to conflicts in course schedules and lecture halls assignments.

iii. The manual system often lacks transparency in the allocation process.

iv. The manual system struggles to accommodate last-minute changes or adjustments to the course offerings and schedules.

3.3 Advantages of the proposed system

The following are the advantages of the automated system for departmental course allocation and timetable. They include the following:

i. One of the key advantages of the proposed automated system is to enhanced efficiency in resource allocation.

ii. Reduces the occurrence of human errors that are common in manual method.

iii. It enhances transparency in course allocation process.

iv. Offers improved flexibility and adaptability in responding to changes.

v. Saves time compared to the manual process.

3.4 The Proposed Method

The Waterfall model is a traditional software development methodology that follows a linear and sequential approach to project management and product development. Here, I'll outline the Waterfall model as it applies to the development of the proposed automated system for departmental course allocation and timetable generation.

Requirements Gathering and Analysis: In this initial phase, the project team collaborates with stakeholders, including administrators, faculty members, and students, to gather and document detailed requirements for the automated system. This involves understanding the current allocation process, identifying pain points, and defining specific functionalities and features of the system.

System Design: Based on the requirements gathered, the system design phase involves creating a detailed design of the automated system. This includes designing the user interface, database structure, allocation algorithms, timetable generation module, and integration points with existing systems. The design phase also defines the technical architecture, software components, and data flow within the system.

Implementation: During the implementation phase, the development team begins coding the system based on the design specifications. The user interface is developed, database tables and relationships are implemented, and allocation algorithms and timetable generation modules are coded. This phase focuses on translating the design into actual software components while adhering to coding standards and best practices.

Testing: Once the implementation is complete, the system undergoes rigorous testing to ensure its functionality, accuracy, and reliability. Various testing techniques are employed, including unit testing, integration testing, and system testing. The allocation engine and timetable generation module are thoroughly tested using both sample and real-world data to identify and rectify any bugs or errors.

Maintenance: The maintenance phase involves ongoing monitoring, support, and updates to the system. Feedback from users is collected and incorporated into system improvements. Bug fixes, enhancements, and optimizations are implemented as necessary to ensure the system's continued effectiveness. This phase ensures that the system remains aligned with the evolving needs of the institution.

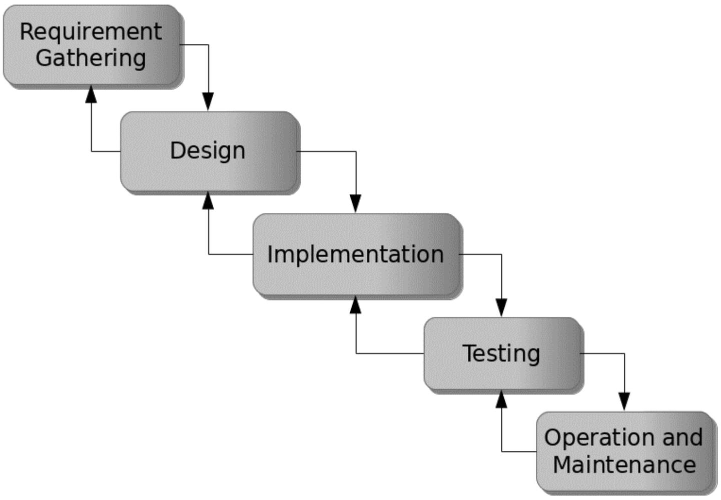


Figure 3.1: Waterfall model

3.5 Method of Data Collection

There are two main sources of data collection in carrying out this study, information was basically obtained from the two sources which are the primary and secondary source.

3.6 System Design

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements.

3.6.1 Algorithm diagram

Use Case Diagram

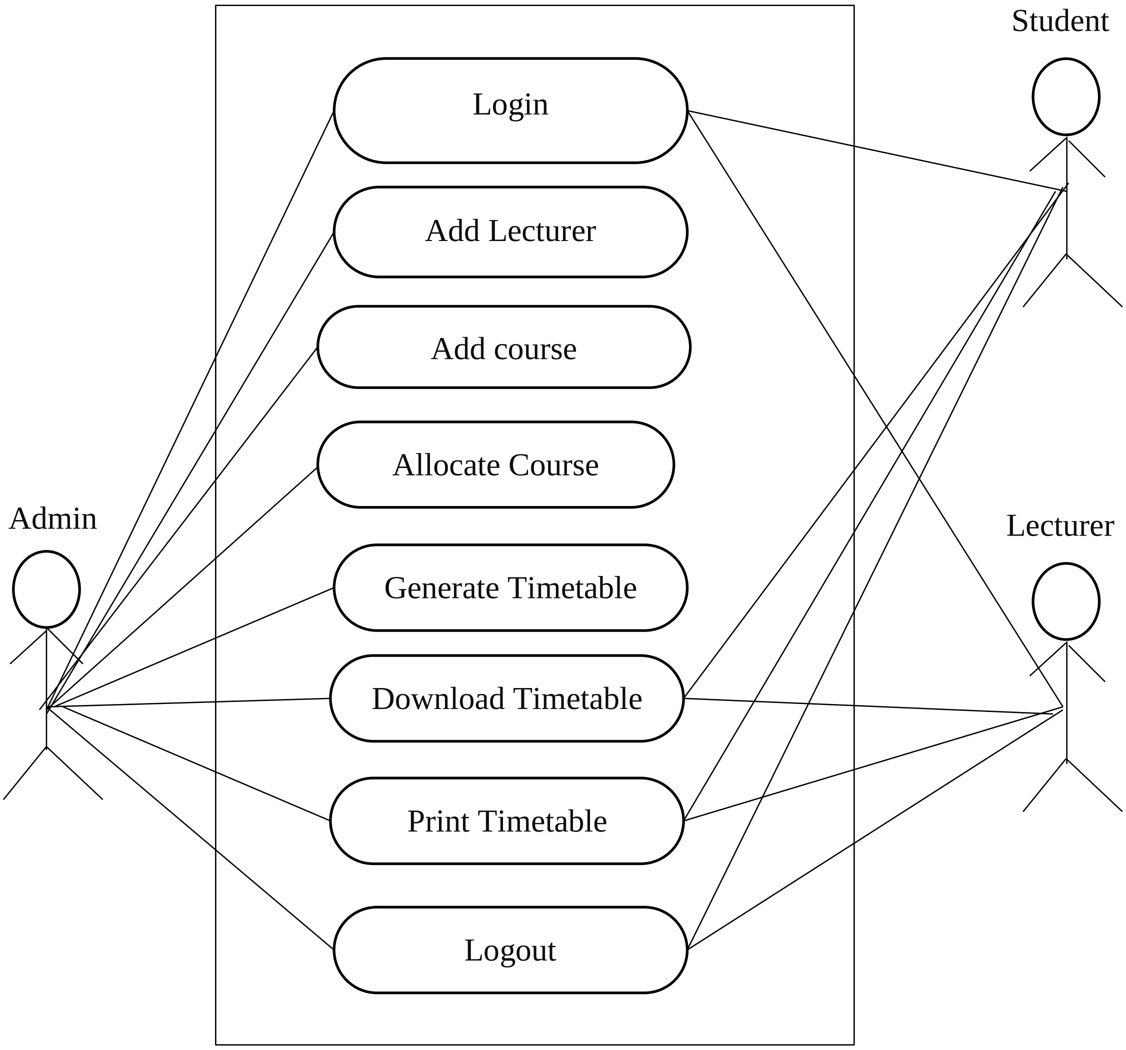


Figure 3.2: Use Case Diagram

3.6.2 System Architecture

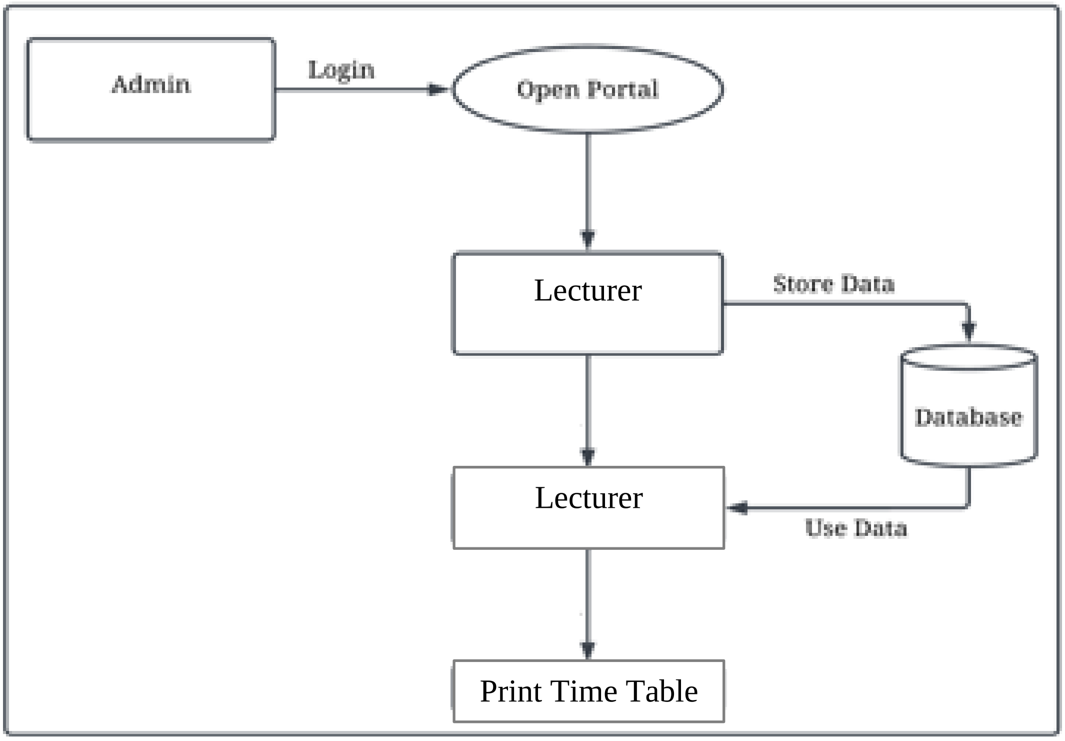


Figure 3.3: System Architecture

3.6.3 Database Tables/Queries Structures

The database is used to store all information that pertain the course allocation and timetable generation records. Below are the database table for the new system.

Table 1: Admin Details

Name	**Type**	**Extra**
id	int(11	AUTO_INCREMENT
Name	varchar(50)	
Password	varchar(50)	

Table 2: Lecture Hall Records

Name	**Type**	**Extra**
id	int(11)	AUTO_INCREMENT
Hall title	varchar(250)	
Hall code	varchar(250)	

Table 3: Lecturers' Details

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
File No	varchar(250)	
Name	varchar(250)	
Department	varchar(250)	
Phone number	Varchar(250)	
Email id	varchar(255)	

Table 4: Courses Details

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
Course Title	varchar(250)	
Course code	varchar(250)	
Semester	varchar(250)	
Department	Varchar(250)	
Level	Varchar (250)	

3.6.4 Entity Relationship Modelling

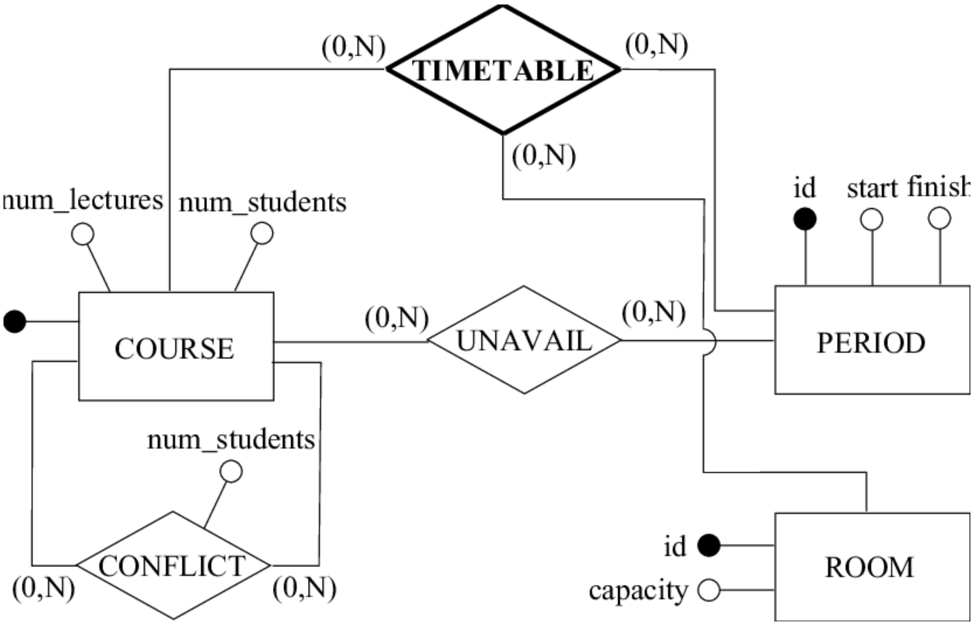


Figure 3.4: Entity Relationship Modelling

3.6.5 Input and Output Design

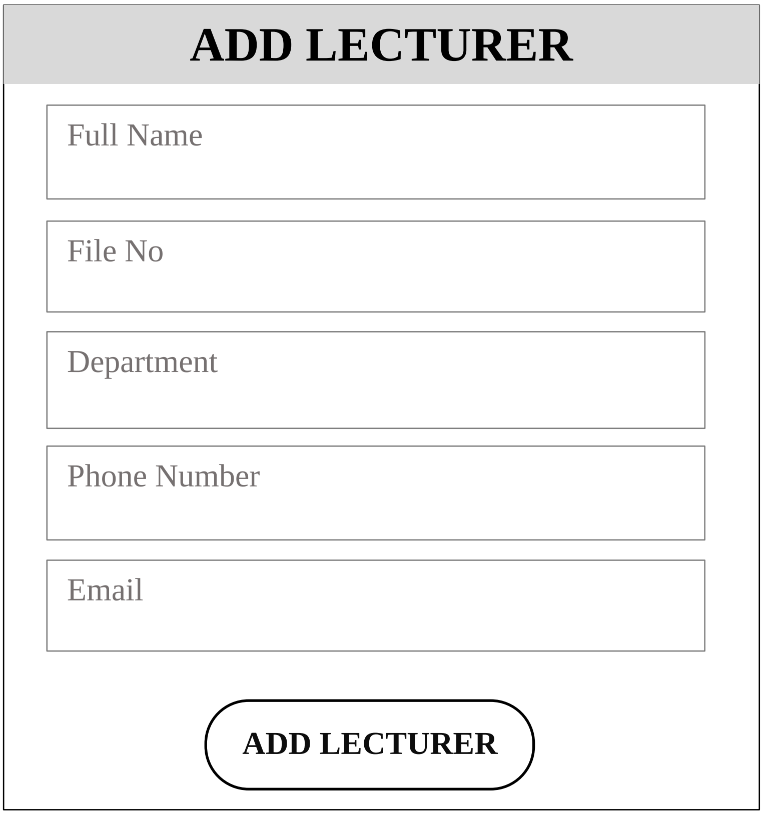


Figure 3.5: Add Lecturer

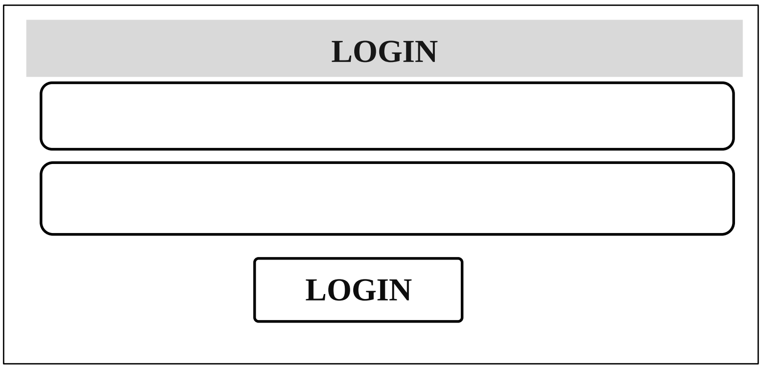


Figure 3.6: Login form

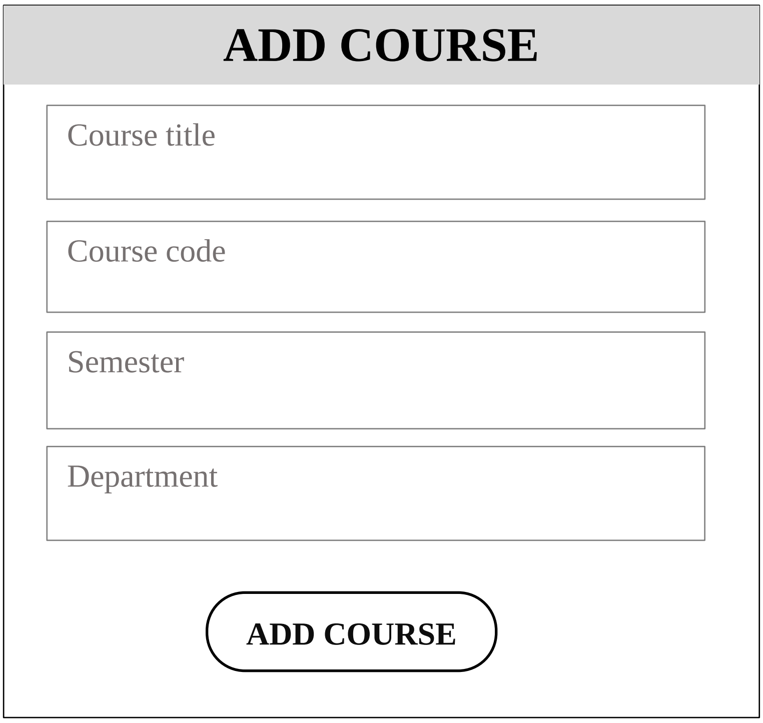


Figure 3.7: Add Lecturer

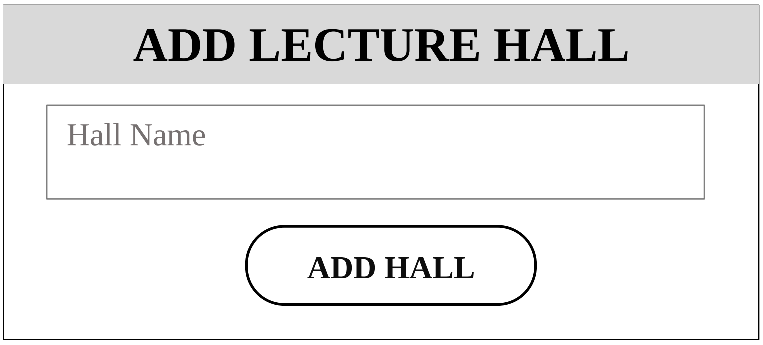


Figure 3.8: Add Lecture Hall

3.7 System Requirement Specification

3.7.1 Hardware Requirements

The software to be design needs the following hardware for an effective operation of the newly designed system.

- **i.** A system running on intel, P(R) duo core with higher processor
- **ii.** The-Random Access Memory (RAM) should be at least 512MB.
- **iii.** At least 20-GB hard disk.
- **iv.** A monitor.

3.7.2 Software Requirements

The software requirements include:

- **i.** A window 7 or higher version of operating system.
- **ii.** XAMP or WAMP for Database
- **iii.** PHP
- **iv.** MySQL
- **v.** Browser

3.7.3 Personnel Requirement

Any computer literate who has a technical knowhow of internet surfing can use the system because it is user friendly.

CHAPTER FOUR

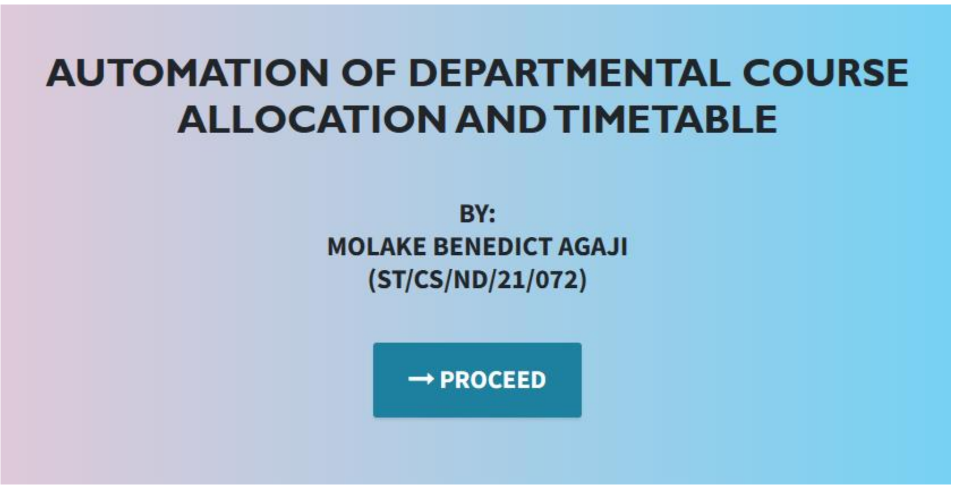
RESULTS AND DISCUSSION

4.1 Introduction

The new system is designed using PHP and MySQL programming language for easy records inserting and updating. This system will help in managing and easily retrieving of information from the system for management purposes. Automation of Departmental Course Allocation and Time Table.

4.2 Results

4.2.1 Welcome Interface



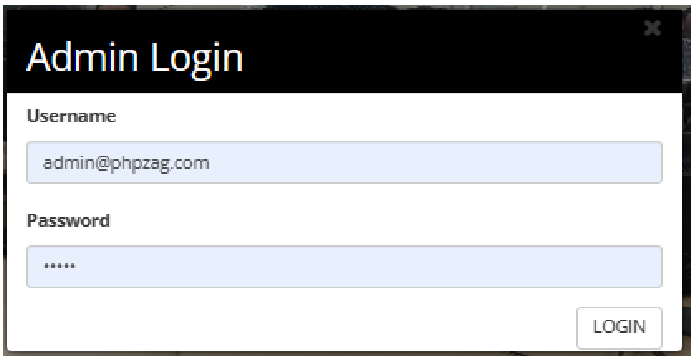
AUTOMATION OF DEPARTMENTAL COURSE ALLOCATION AND TIMETABLE

BY:
MOLAKE BENEDICT AGAJI
(ST/CS/ND/21/072)

Figure 4.1: Welcome Interface

The above figure 4.1 shows the welcome page of the Automation of Departmental Course Allocation and Time Table System, on the welcome page is the first page that displays the project topic.

4.2.2 Login Interface



Admin Login

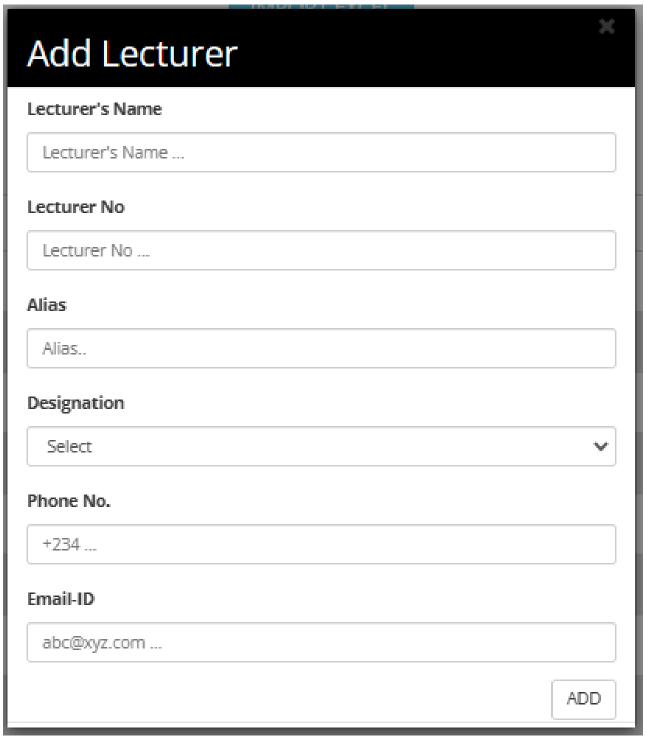
Username: admin@phpzag.com

Password: •••••

Figure 4.2: Login page interface

Figure 4.2 above shows the login interface which allows the admin to enter his login details to get access to the system in order to allocate courses and add lecturers.

4.2.3 Add Lecturer Interface



Add Lecturer

Lecturer's Name

Lecturer's Name ...

Lecturer No

Lecturer No ...

Alias

Alias..

Designation

Select

Phone No.

+234 ...

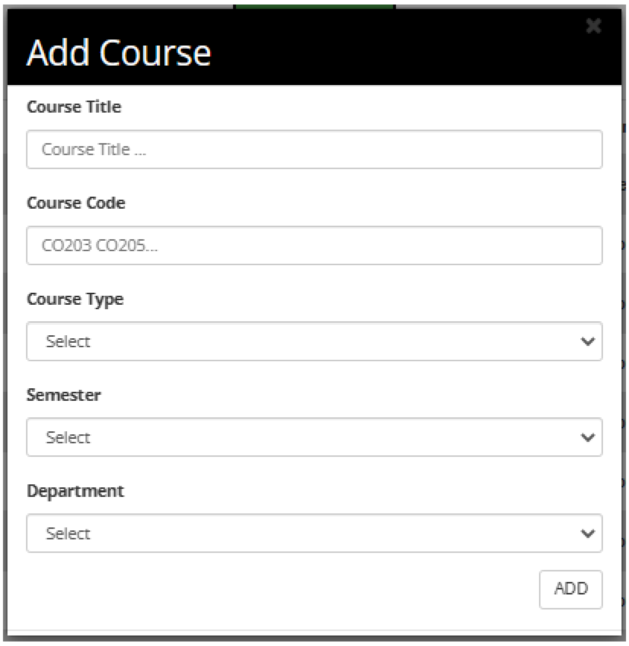
Email-ID

abc@xyz.com ...

Figure 4.3: Add Lecturer Interface

Figure 4.3 above shows the add lecturer interface which is used by the admin to register new lecturers into the system by filling in some lecturer’s details such as file no, name, phone.

4.2.4 Add Course Interface



Add Course

Course Title

Course Title ...

Course Code

CO203 CO205...

Course Type

Select

Semester

Select

Department

Select

Figure 4.4: Add Course Interface

Figure 4.4 above shows the add course interface which is used to insert new courses into the system for course allocation.

4.2.5 Generate Time Table Interface

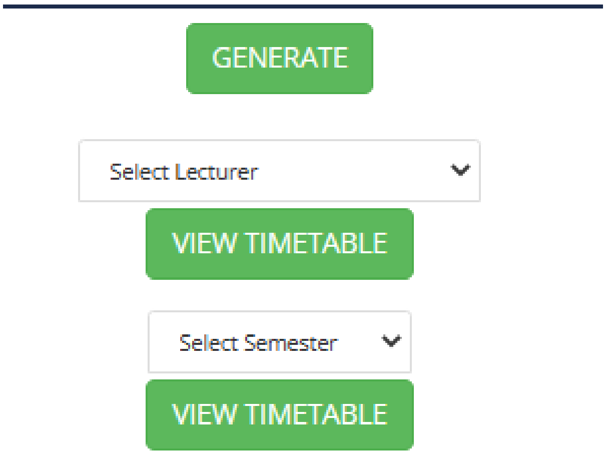
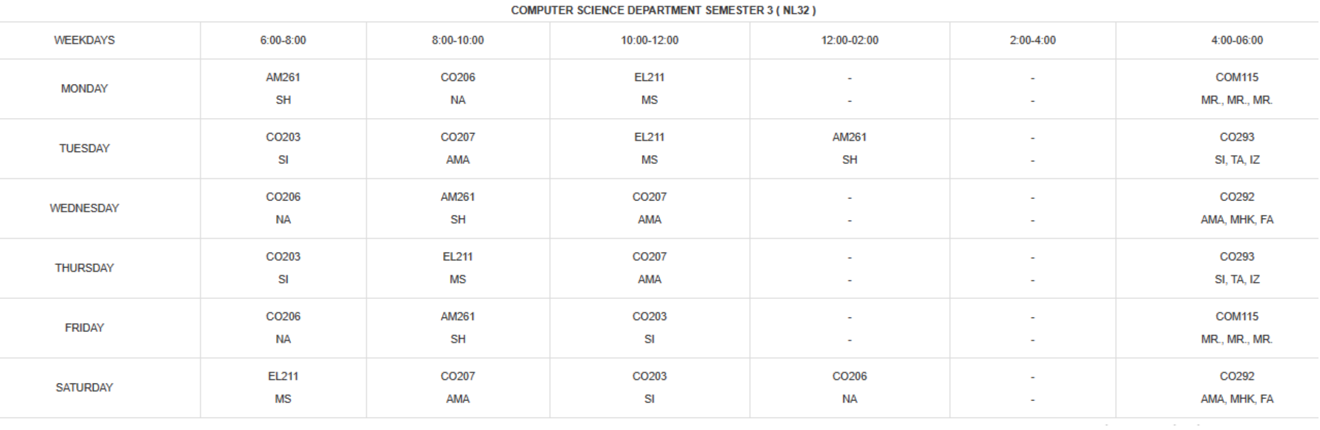


Figure 4.5: Generate Time Table Interface

Figure 4.5 above is used to generate time table for a particular lecturer or semester as selected by the user which will be displayed and printed.

4.2.6 View Time Table Interface



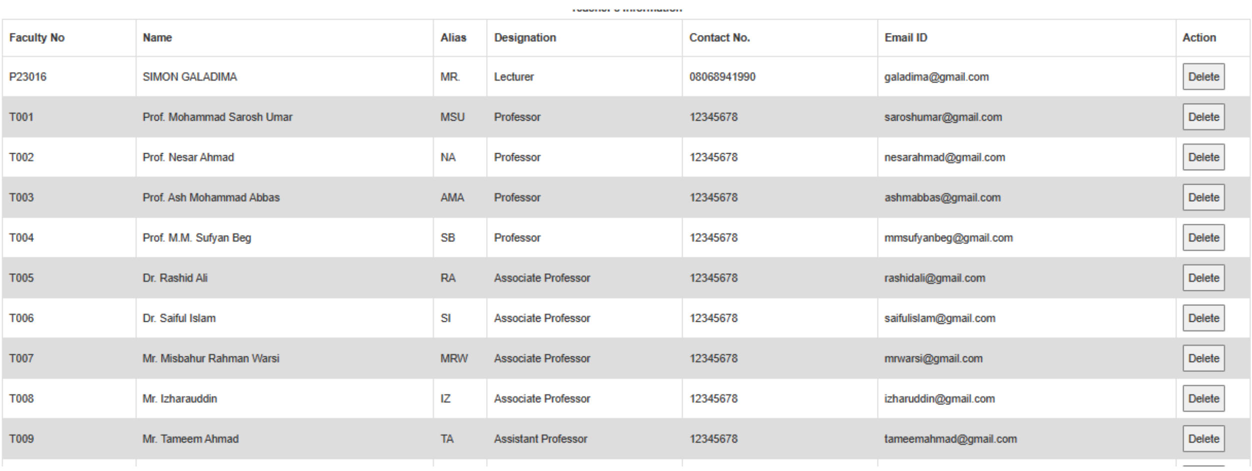
COMPUTER SCIENCE DEPARTMENT SEMESTER 3 (NL32)

WEEKDAYS	6:00-8:00	8:00-10:00	10:00-12:00	12:00-02:00	2:00-4:00	4:00-06:00
MONDAY	AM261 SH	CO206 NA	EL211 MS	- -	- -	COM115 MR., MR., MR.
TUESDAY	CO203 SI	CO207 AMA	EL211 MS	AM261 SH	- -	CO293 SI, TA, IZ
WEDNESDAY	CO206 NA	AM261 SH	CO207 AMA	- -	- -	CO292 AMA, MHK, FA
THURSDAY	CO203 SI	EL211 MS	CO207 AMA	- -	- -	CO293 SI, TA, IZ
FRIDAY	CO206 NA	AM261 SH	CO203 SI	- -	- -	COM115 MR., MR., MR.
SATURDAY	EL211 MS	CO207 AMA	CO203 SI	CO206 NA	- -	CO292 AMA, MHK, FA

Figure 4.6: View Time Table Interface

Figure 4.6 view time table interface is used to display a certificate that has been generated for a particular lecturer or semester.

4.2.7 List of Lecturers



Faculty No	Name	Alias	Designation	Contact No.	Email ID	Action
P23016	SIMON GALADIMA	MR.	Lecturer	08068941990	galadima@gmail.com	Delete
T001	Prof. Mohammad Sarosh Umar	MSU	Professor	12345678	saroshumar@gmail.com	Delete
T002	Prof. Nesar Ahmad	NA	Professor	12345678	nesarahmad@gmail.com	Delete
T003	Prof. Ash Mohammad Abbas	AMA	Professor	12345678	ashmabbas@gmail.com	Delete
T004	Prof. M.M. Sufyan Beg	SB	Professor	12345678	mmsufyanbeg@gmail.com	Delete
T005	Dr. Rashid Ali	RA	Associate Professor	12345678	rashidali@gmail.com	Delete
T006	Dr. Saiful Islam	SI	Associate Professor	12345678	saifulislam@gmail.com	Delete
T007	Mr. Misbahur Rahman Warsi	MRW	Associate Professor	12345678	mrwarsi@gmail.com	Delete
T008	Mr. Izharauddin	IZ	Associate Professor	12345678	izharuddin@gmail.com	Delete
T009	Mr. Tameem Ahmad	TA	Assistant Professor	12345678	tameemahmad@gmail.com	Delete

Figure 4.7: List of Lecturers Interface

Figure 4.7 above displays all the registered courses in the system which will be used for allocation of courses.

4.3 Discussion

Figure 4.1 Welcome Interface: The Welcome Interface serves as the entry point for users accessing the automation system. It typically includes a friendly greeting message and provides an overview of the system's functionalities. Users are usually presented with options to navigate further into the system, such as logging in, accessing course-related features, or generating timetables.

Figure 4.2 Login Interface: The Login Interface is where users provide their login credentials to access the system securely. This typically includes fields for entering usernames and passwords. Additionally, there may be options for password recovery or account creation if applicable.

Figure 4.3 Add Lecturer Interface: The Add Lecturer Interface allows authorized users to input information about new lecturers. It typically includes fields for entering details such as lecturer names, contact information, qualifications, and areas of expertise. Users can use this interface to register new teaching staff within the department.

Figure 4.4 Add Course Interface: The Add Course Interface is used to input information about new courses offered by the department. This interface usually includes fields for course names, descriptions, prerequisites, and the lecturer assigned to the course. It allows administrators to manage the course catalog.

Figure 4.5 Generate Time Table Interface: The Generate Time Table Interface is where the automated scheduling process takes place. Authorized users can specify parameters such as course offerings, lecturer availability, classroom assignments, and time constraints. Upon submission, the system generates a timetable that optimally allocates courses based on the provided criteria.

Figure 4.6 View Time Table Interface: The View Time Table Interface allows users to access and view the generated timetables. Users can select specific courses or time slots to see their respective schedules. This interface provides an overview of the department's class schedules and room assignments.

Figure 4.7 List of Lecturers: The List of Lecturers Interface presents a comprehensive list of lecturers within the department. It typically includes lecturer names, contact details, academic qualifications, and courses they are assigned to teach. This interface is useful for administrators and faculty members to track lecturer assignments and contact information.

These interfaces collectively form the user interface components of the Automation of Departmental Course Allocation and Time Table system. They are designed to streamline administrative tasks, enhance communication, and optimize the allocation of resources within the department for efficient course scheduling and management.

4.4 User manual

The following are the necessary steps to take in order to use the system efficiently and effectively.

i. Load the url of the system https://localhost/ttms / the welcome page will be displayed.

ii. Click on the **Proceed** button to proceed to the main system.

iii. If you created an account, provide your login details by entering your username and password.

iv. Depending on the login details provided you will be automatically directed to the dashboard.

v. The various task that you can perform on the portal will be displayed on the sidebar of the dashboard.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary

In summary, the "Automation of Departmental Course Allocation and Time Table" represents a transformative solution for academic institutions seeking to optimize their course scheduling and resource allocation processes. This system streamlines administrative tasks, enhances communication, and improves overall operational efficiency within the department. By automating the complex task of course allocation and timetable generation, it reduces errors and manual workload while ensuring optimal resource utilization. The successful implementation of this system offers a model for educational institutions looking to enhance their course allocation and scheduling procedures.

5.2 Conclusion

The design and implementation of the Course Allocation and Time Table Automation system have yielded significant benefits for the department. This project demonstrates the tangible advantages of leveraging technology to address complex scheduling challenges within an academic context. By automating these processes, the system has improved operational efficiency, reduced conflicts, and enhanced communication between faculty members and students. It stands as a testament to the transformative power of technology-driven solutions in the optimization of educational operations.

5.3 Recommendations

Based on the successful implementation of this system, several recommendations can be made for its continued improvement and adoption:

i. User Training: Provide comprehensive training and support for administrators, faculty members, and students to maximize the system's benefits and adoption.

ii. Data Integration: Explore opportunities for integrating the system with other institutional systems, such as student information and learning management systems, for a more seamless and holistic educational experience.

iii. Enhanced Reporting: Develop advanced reporting and analytics features to provide administrators with insights into scheduling trends and resource allocation.

iv. Scalability: Ensure that the system is scalable to accommodate the evolving needs of the department as it grows and expands its course offerings.

5.4 Contribution to Knowledge

This project contributes to knowledge in several key areas:

Educational Technology: It showcases how technology can significantly improve administrative processes within academic institutions, leading to enhanced operational efficiency.

Resource Optimization: The system's automated allocation processes offer insights into optimal resource utilization, which can benefit similar academic departments.

Scheduling Efficiency: The project demonstrates how automation can reduce scheduling conflicts and streamline complex scheduling tasks in educational settings.

5.5 Area for Further Work

Future work in this area could include:

Machine Learning Integration: Exploring the integration of machine learning algorithms for more intelligent and adaptive scheduling based on historical data and trends.

Mobile Accessibility: Developing a mobile application version of the system to cater to the mobile preferences of users.

Data Security Enhancements: Continuously improving data security measures to protect sensitive information and maintain trust.

Optimization Algorithms: Investigating advanced optimization algorithms to further enhance resource allocation and scheduling efficiency.

REFERENCES

Al-Mogren, A., Alshammari, R., & Almulhim, N. (2021). Machine learning techniques for faculty course allocation in educational institutions. International Journal of Computer Applications, 182(49), 13-20.

Chen, J., Xu, M., Yu, S., & Zhou, F. (2021). A hybrid optimization algorithm for course timetable scheduling problem in universities. Journal of Ambient Intelligence and Humanized Computing, 12(11), 14811-14822.

Gao, J., Wu, Z., Yang, Q., Cheng, B., & Xie, G. (2022). An intelligent course allocation system based on machine learning. Journal of Educational Technology Development and Exchange, 15(2), 1-15.

Jin, Z., & Chen, S. (2022). An automated timetable generation system based on constraint satisfaction programming. Journal of Ambient Intelligence and Humanized Computing, 13(1), 1539-1552.

Jin, Z., & Li, H. (2021). A study on the user interface design of the course scheduling system. Journal of Software, 32(3), 852-868.

Kizilcec, R. F., Marlow, J., & Bailenson, J. N. (2021). Supporting academic advising at scale with chatbots. Computers & Education, 166, 104222.

Li, Y., Zhang, J., Li, Z., Zhang, Y., & Zhang, S. (2021). A multi-objective optimization model for faculty course assignments. International Journal of Artificial Intelligence in Education, 31(2), 177-202.

Lin, C., Wang, X., & Wen, Q. (2022). The effect of training on user acceptance of automated scheduling systems in higher education. Educational Technology Research and Development, 70(1), 179-197.

Liu, Q., Zhang, H., & Wang, J. (2020). Automated classroom allocation in universities: A multi-objective optimization approach. Computers & Industrial Engineering, 140, 106306.

Ma, Y., Li, J., Yang, Y., Huang, Z., & Gu, Z. (2021). An intelligent recommendation system for online course registration based on collaborative filtering and deep learning. IEEE Access, 10, 79705-79716.

Nguyen, H. M., Tran, L. T., & Ngo, D. L. (2022). Long-term impact evaluation of an automated course allocation system in a university. Computers & Education, 182, 104712.

Parry, M., Richards, D., & Holmes, G. (2021). Automated essay scoring: A systematic review of the literature. Computers & Education, 165, 104170.

Rahman, A., Uddin, M. Z., Islam, S. U., Hossain, M. A., & Al-Fuqaha, A. (2022). Intelligent automation for administrative workflows in higher education institutions. IEEE Access, 10, 3152-3166.

Shahin, M. A., Al-Sharhan, S., Alarabeyyat, M., & Abualqumboz, M. (2021). Integrating a course allocation system with a student information system in higher education. International Journal of Computer Science Issues, 18(2), 1-14.

Sun, J., Ma, S., & Gao, J. (2022). Design and evaluation of an intelligent course allocation system. Journal of Educational Technology Development and Exchange, 15(1), 1-15.

Wang, X., Yang, W., Chen, Y., & Zhang, J. (2022). Course allocation system based on constraint satisfaction programming. International Journal of Computer Applications, 184(18), 27-35.

Wu, Z., Gao, J., Cheng, B., & Xie, G. (2021). An automated course scheduling system with intelligent scheduling and visualized feedback. International Journal of Emerging Technologies in Learning, 16(3), 46-63.

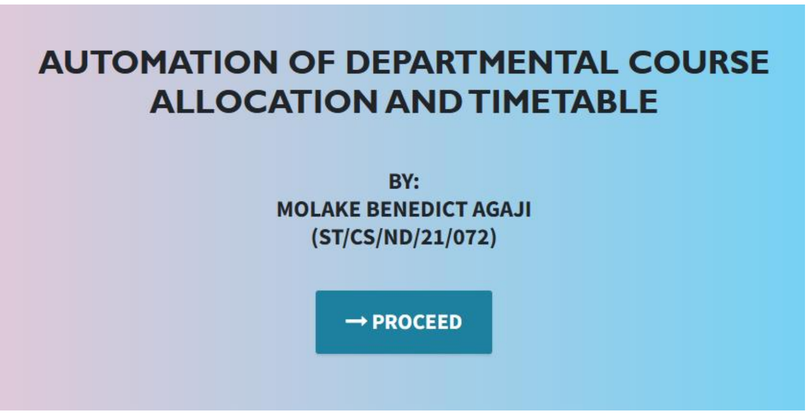
Yang, X., Xing, X., & Chen, M. (2021). Usability testing and evaluation for a college course registration system. Journal of Computational Science Education, 12(1), 46-55.

Zhang, Z., Wang, Q., & Zhang, J. (2020). Hybrid algorithm for course scheduling problem in universities. IEEE Access, 8, 204797-204806.

Zhou, J., Li, X., Zhang, Y., & Chen, Y. (2021). Design and application of the automatic college course scheduling system. In 2021 International Conference on Artificial Intelligence in Education and Smart Learning (ICAISL) (pp. 25-28). IEEE.

APPENDIX A

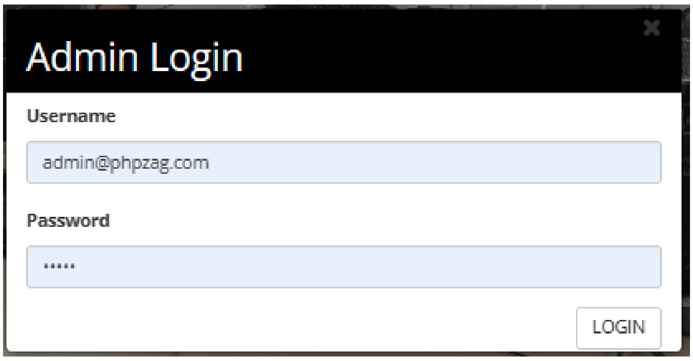
Welcome interface



AUTOMATION OF DEPARTMENTAL COURSE ALLOCATION AND TIMETABLE

BY:
MOLAKE BENEDICT AGAJI
(ST/CS/ND/21/072)

Login Interface

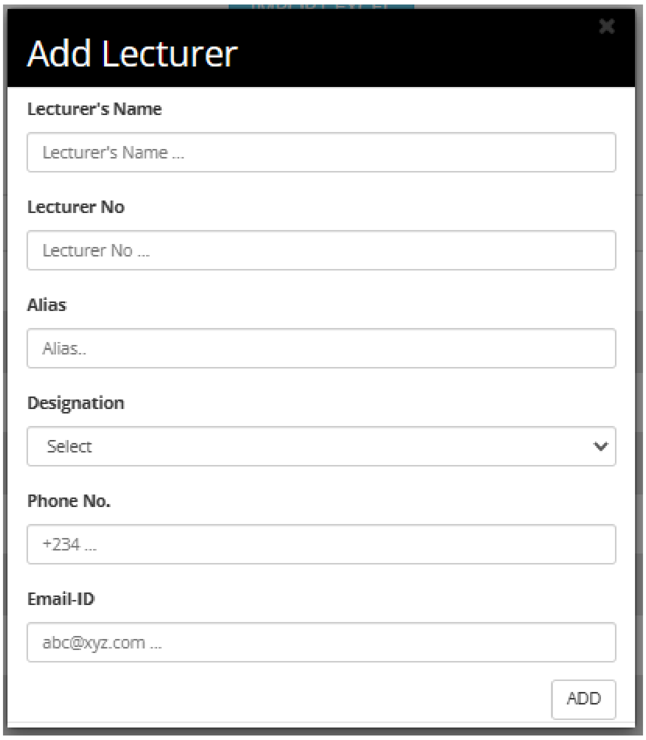


Admin Login

Username: admin@phpzag.com

Password: •••••

Add Lecturer Interface



Add Lecturer

Lecturer's Name: Lecturer's Name ...

Lecturer No: Lecturer No ...

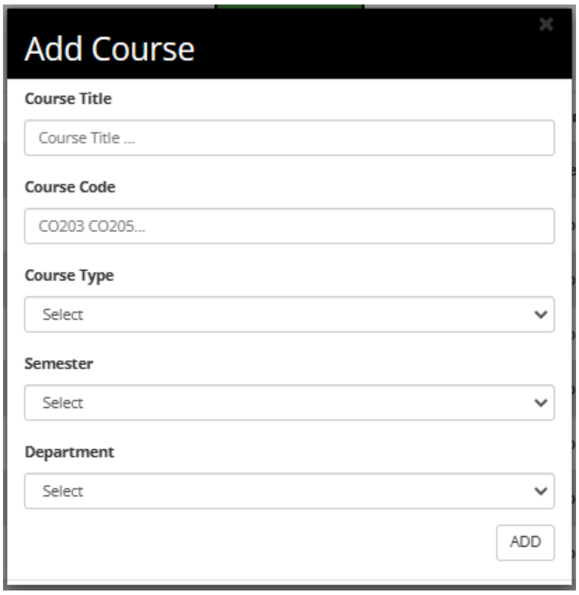
Alias: Alias..

Designation: Select

Phone No.: +234 ...

Email-ID: abc@xyz.com ...

Add Course Interface



Add Course

Course Title

Course Title ...

Course Code

CO203 CO205...

Course Type

Select

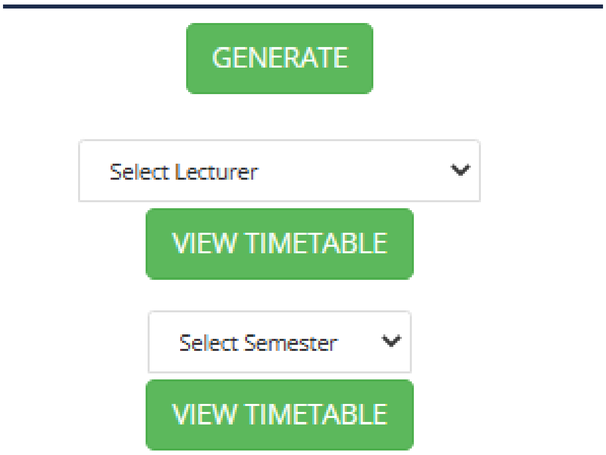
Semester

Select

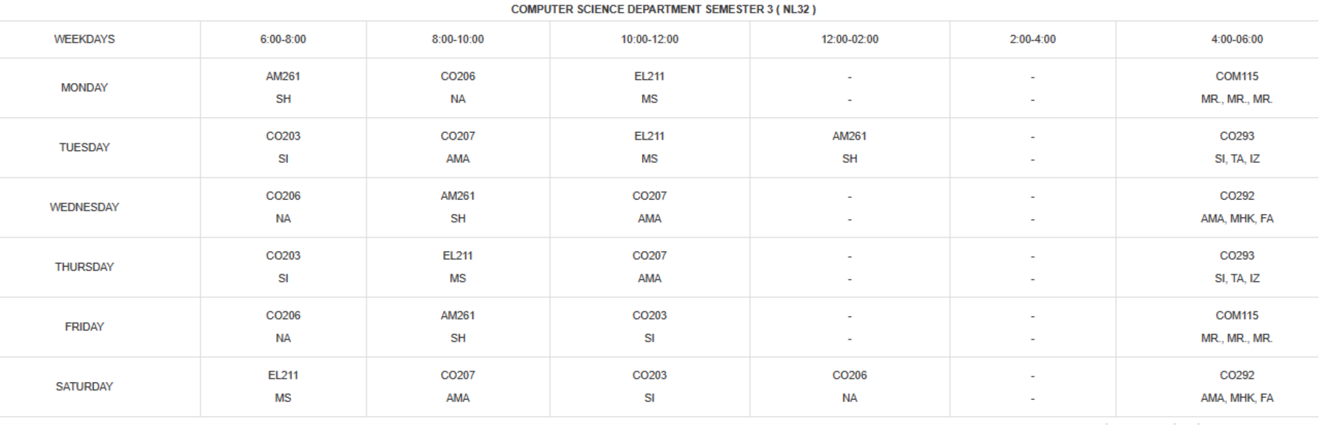
Department

Select

Generate Time Table Interface



View Time Table Interface



COMPUTER SCIENCE DEPARTMENT SEMESTER 3 (NL32)

WEEKDAYS	6:00-8:00	8:00-10:00	10:00-12:00	12:00-02:00	2:00-4:00	4:00-06:00
MONDAY	AM261 SH	CO206 NA	EL211 MS	- -	- -	COM115 MR., MR., MR.
TUESDAY	CO203 SI	CO207 AMA	EL211 MS	AM261 SH	- -	CO293 SI, TA, IZ
WEDNESDAY	CO206 NA	AM261 SH	CO207 AMA	- -	- -	CO292 AMA, MHK, FA
THURSDAY	CO203 SI	EL211 MS	CO207 AMA	- -	- -	CO293 SI, TA, IZ
FRIDAY	CO206 NA	AM261 SH	CO203 SI	- -	- -	COM115 MR., MR., MR.
SATURDAY	EL211 MS	CO207 AMA	CO203 SI	CO206 NA	- -	CO292 AMA, MHK, FA

APPENDIX B

PROGRAM CODE

```
<!DOCTYPE html>
<html lang="en">

  <head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>COURSE ALLOCATION AND TIME TABLE MANAGEMENT SYSTEM</title>

    <!-- Bootstrap Core CSS -->
    <link href="vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet"
type="text/css">
    <link
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,700,300itali
c,400italic,700italic" rel="stylesheet" type="text/css">
    <link href="vendor/simple-line-icons/css/simple-line-icons.css"
rel="stylesheet">

    <!-- Custom CSS -->
    <link href="css/stylish-portfolio.min.css" rel="stylesheet">

  </head>

  <body id="page-top" style=" background: linear-gradient(90deg, pink, rgb(67,
207, 250));">



    <!-- Header -->
    <header class="masthead d-flex">
      <div class="container text-center">

        <h1 class="mb-2" style="font-size: 45px; font-weight: bolder; font-family:
'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif; text-transform:
uppercase;"><span style=" margin-top:15px;"> AUTOMATION OF DEPARTMENTAL COURSE
ALLOCATION AND TIMETABLE</span> <br>  </h1>
        <h2 class="mb-3" style=""></h2><br><br>
        <h2>BY: <br> MOLAKE BENEDICT AGAJI<br> (ST/CS/ND/21/072)</h2> <br>


        </h3>

        <br>
        <strong><a class="btn btn-primary btn-xl js-scroll-trigger" href="files/" 
style="font-size: 30px;"><span class="fa fa-long-arrow-right"></span> 
PROCEED</a></strong>
        <br> <br> <br>
        <!-- <h4 class="alert alert-success"><a href="onlinefood-
order/admin/index.php">Admin Login Here!</a></h4> -->
              </div>
      <div class="overlay"></div>
    </header>



    <!-- Scroll to Top Button-->
    <a class="scroll-to-top rounded js-scroll-trigger" href="#page-top">
      <i class="fa fa-angle-up"></i>
    </a>

    <!-- Bootstrap core JavaScript -->
    <script src="vendor/jquery/jquery.min.js"></script>
    <script src="vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

    <!-- Plugin JavaScript -->
    <script src="vendor/jquery-easing/jquery.easing.min.js"></script>

    <!-- Custom scripts for this template -->
    <script src="js/stylish-portfolio.min.js"></script>

  </body>

</html>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="utf-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-
scale=1"/>
    <meta name="description" content=""/>
    <meta name="author" content=""/>
    <title>TimeTable Management System</title>
    <!-- BOOTSTRAP CORE STYLE CSS -->
    <link href="assets/css/bootstrap.css" rel="stylesheet"/>
    <!-- FONT AWESOME CSS -->
    <link href="assets/css/font-awesome.min.css" rel="stylesheet"/>
    <!-- FLEXSLIDER CSS -->
    <link href="assets/css/flexslider.css" rel="stylesheet"/>
    <!-- CUSTOM STYLE CSS -->
    <link href="assets/css/style.css" rel="stylesheet"/>
    <!-- Google Fonts -->
    <link href='http://fonts.googleapis.com/css?family=Open+Sans:400,700,300' 
rel='stylesheet' type='text/css'/>
</head>
<body>

<div class="navbar navbar-inverse navbar-fixed-top " id="menu">
    <div class="container">
        <div class="navbar-header">
            <button type="button" class="navbar-toggle" data-toggle="collapse" 
data-target=".navbar-collapse">
```

```
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>

        </div>
        <div class="navbar-collapse collapse move-me">
            <ul class="nav navbar-nav navbar-left">
                <li><a href="addteachers.php">ADD LECTURER</a></li>
                <li><a href="addsubjects.php">ADD COURSE</a></li>
                <li><a href="addclassrooms.php">ADD HALLS</a></li>
                <li class="dropdown"><a class="dropdown-toggle" data-
toggle="dropdown" aria-expanded="false">ALLOTMENT
                        <span class="caret"></span></a>
                    <ul class="dropdown-menu">
                        <li>
                            <a href=allotsubjects.php>THEORY COURSES</a>
                        </li>
                        <li>
                            <a href=allotpracticals.php>PRACTICAL COURSES</a>
                        </li>
                        <li>
                            <a href=allotclasses.php>HALLS</a>
                        </li>
                    </ul>
                </li>
                <li><a href="generatetimetable.php">GENERATE TIMETABLE</a></li>

            </ul>
            <ul class="nav navbar-nav navbar-right">
                <li><a href="index.php">LOGOUT</a></li>
            </ul>

        </div>
    </div>
</div>
<!--NAVBAR SECTION END-->
<br>

<div align="center" style="margin-top:80px">
    <form name="import" method="post" enctype="multipart/form-data">
        <input type="file" name="file"/>
        <input type="submit" name="teacherexcel" id="teacherexcel" class="btn btn-
info btn-lg" value="IMPORT EXCEL"/>
    </form>
    <?php
    if (isset($_POST['teacherexcel'])) {
        if (empty($_FILES['file']['tmp_name'])) {
            echo '<script>alert("Select a file first! ");</script>';
        } else {
            $file = $_FILES['file']['tmp_name'];
            $handle = fopen($file, 'r');
            $headings = true;
            while (!feof($handle)) {
                $filesop = fgetcsv($handle, 1000);

                $facno = $filesop[0];
                $name = $filesop[1];
                $alias = $filesop[2];
```

```
                    mysqli_query(mysqli_connect("localhost", "root", "", "ttms"),
$sql);
                    $days = array('monday', 'tuesday', 'wednesday', 'thursday',
'friday', 'saturday');
                    for ($i = 0; $i < 6; $i++) {
                        $day = $days[$i];
                        $sql = "INSERT into " . $facno . "
VALUES('$day','','','','','','')";
                        mysqli_query(mysqli_connect("localhost", "root", "",
"ttms"), $sql);
                    }
                }
            }
        }
    }
    ?>
</div>
<div align="center" style="margin-top:20px">
    <button id="teachermanual" class="btn btn-success btn-lg">ADD
LECTURER</button>
</div>

<div id="myModal" class="modal">

    <!-- Modal content -->
    <div class="modal-content" style="margin-top: -60px">
        <div class="modal-header">
            <span class="close">&times</span>
            <h2 id="popupHead">Add Lecturer</h2>
        </div>
        <div class="modal-body" id="EnterTeacher">
            <!--Admin Login Form-->
            <div style="display:none" id="addTeacherForm">
                <form action="addteacherFormValidation.php" method="POST">
                    <div class="form-group">
                        <label for="teachername">Lecturer's Name</label>
                        <input type="text" class="form-control" id="teachername"
name="TN"
                               placeholder="Lecturer's Name ...">
                    </div>
                    <div class="form-group">
                        <label for="TF">Lecturer No</label>
                        <input type="text" class="form-control" id="facultyno"
name="TF" placeholder="Lecturer No ...">
                    </div>
                    <div class="form-group">
                        <label for="TF">Alias</label>
                        <input type="text" class="form-control" id="alias_name"
name="AL" placeholder="Alias..">
                    </div>
                    <div class="form-group">
                        <label for="designation">Designation</label>

                        <select class="form-control" id="designation" name="TD">
                            <option selected disabled>Select</option>
                            <option value="Professor">Professor</option>
                            <option value="Doctor">Doctor</option>
                            <option value="Associate Professor">Associate
Professor</option>
```

```
                            <option value="Lecturer">Lecturer</option>
                            <option value="Assistant Lecturer">Assistant
Lecturer</option>
                        </select>
                    </div>
                    <div class="form-group">
                        <label for="teachercontactnumber">Phone No.</label>
                        <input type="text" class="form-control"
id="teachercontactnumber" name="TP"
                               placeholder="+234 ...">
                    </div>

                    <div class="form-group">
                        <label for="teacheremailid">Email-ID</label>
                        <input type="text" class="form-control"
id="teacheremailid" name="TE"
                               placeholder="abc@xyz.com ...">
                    </div>
                    <div align="right">
                        <input type="submit" class="btn btn-default" name="ADD"
value="ADD">
                    </div>
                </form>
            </div>
        </div>
        <div class="modal-footer">
        </div>
    </div>
</div>

<script>
    // Get the modal
    var modal = document.getElementById('myModal');

    // Get the button that opens the modal
    var addteacherBtn = document.getElementById("teachermanual");
    var heading = document.getElementById("popupHead");
    var facultyForm = document.getElementById("addTeacherForm");
    // Get the <span> element that closes the modal
    var span = document.getElementsByClassName("close")[0];
    <script>
        function deleteHandlers() {
            var table = document.getElementById("teacherstable");
            var rows = table.getElementsByTagName("tr");
            for (i = 0; i < rows.length; i++) {
                var currentRow = table.rows[i];
                //var b = currentRow.getElementsByTagName("td")[0];
                var createDeleteHandler =
                    function (row) {
                        return function () {
                            var cell = row.getElementsByTagName("td")[0];
                            var id = cell.innerHTML;
                            var x;
                            if (confirm("Are You Sure?") == true) {
                                window.location.href = "deleteteacher.php?name=" +
id;

                            }
                        };
                    };
                currentRow.cells[6].onclick = createDeleteHandler(currentRow);
            }
        }
    </script>

    <table id=teacherstable style="margin-left: 80px">
        <caption><strong>Teacher's Information </strong></caption>
        <tr>
            <th width="130">Faculty No</th>
            <th width=290>Name</th>
            <th width=50>Alias</th>
            <th width="190">Designation</th>
            <th width="190">Contact No.</th>
            <th width="290">Email ID</th>
            <th width="40">Action</th>
        </tr>
        <tbody>
        <?php
        include 'connection.php';
        $q = mysqli_query(mysqli_connect("localhost", "root", "", "ttms"),
            "SELECT * FROM teachers ORDER BY faculty_number ASC");

        while ($row = mysqli_fetch_assoc($q)) {
            echo "<tr><td>{$row['faculty_number']}</td>
                    <td>{$row['name']}</td>
                    <td>{$row['alias']}</td>
                    <td>{$row['designation']}</td>
                    <td>{$row['contact_number']}</td>
                    <td>{$row['emailid']}</td>
                   <td>
                    <button>Delete</button></td>
</div>
<!--HOME SECTION END-->

<!--<div id="footer">
    <!--  &copy 2014 yourdomain.com | All Rights Reserved |  <a
href="http://binarytheme.com" style="color: #fff" target="_blank">Design by :
binarytheme.com</a>
-->
<!-- FOOTER SECTION END-->

<!--  Jquery Core Script -->
<script src="assets/js/jquery-1.10.2.js"></script>
<!--  Core Bootstrap Script -->
<script src="assets/js/bootstrap.js"></script>
<!--  Flexslider Scripts -->
<script src="assets/js/jquery.flexslider.js"></script>
<!--  Scrolling Reveal Script -->
<script src="assets/js/scrollReveal.js"></script>
<!--  Scroll Scripts -->
<script src="assets/js/jquery.easing.min.js"></script>
<!--  Custom Scripts -->
<script src="assets/js/custom.js"></script>
</body>
</html>
```