**ONLINE TIMETABLE SCHEDULING SYSTEM**

**A CASE STUDY OF SULE LAMIDO UNIVERSITY (SLU),**

**KAFIN HAUSA, JIGAWA STATE**

**BY**

**ABUBAKAR ALHASSAN**

**HND/CS/19/012**

**A PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE, COLLEGE OF SCIENCE AND TECHNOLOGY, HUSSANI ADAMU FEDERAL POLYTECHNIC KAZAURE, JIGAWA STATE.**

**IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF HIGHER NATIONAL DIPLOMA IN COMPUTER SCIENCE.**

**JUNE, 2022**

# **ABSTRACT**

Scheduling course timetables for a large array of courses is a very complex problem which often has to be solved manually by the center staff even though results are not always fully optimal. Timetabling being a highly constrained combinatorial problem, this work attempts to put into play the effectiveness of evolutionary techniques based on Darwin’s theories to solve the timetabling problem if not fully optimal but near optimal.

Genetic Algorithm is a popular meta-heuristic that has been successfully applied to many hard combinatorial optimization problems which includes timetabling and scheduling problems. In this work, the course sets, halls and time allocations are represented by a multidimensional array on which a local search is performed and a combination of the direct representation of the timetable with heuristic crossover is made to ensure that fundamental constraints are not violated.

Finally, the genetic algorithm was applied in the development of a viable timetabling system which was tested to demonstrate the variety of possible timetables that can be generated based on user specified constraint and requirements. This system has been developed using PHP, jQuery, Apache and MySQL. The front-end is designed using PHP with excerpts of code written using jQuery and back-end is designed and managed through MySQL. This system software is more secured, user-friendly and less time-consuming.

**Keywords: Timetable, Scheduling, Genetic Algorithm, PHP, MySQL**

# **APPROVAL/CERTIFICATION PAGE**

I certify that I have supervised and read this study and that in my opinion, it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as having met the requirement for the award of Higher National Diploma in computer science.

Mr. Amaonwu Onyebuchi .…...……………......

Project Supervisor Sign & Date

I certify that I have co-ordinated and read this study and that in my opinion; it conforms to acceptable standard of scholarly presentation and is fully adequate, in scope and quality, as having met the requirement for the award of Higher National Diploma in Computer Science.

Mal. Adamu Isah .……………………..

Project Coordinator Sign & Date

This project was submitted to the Department of Computer Science, College of Science and Technology, Hussaini Adamu Federal Polytechnic Kazaure and is accepted as a fulfillment of the requirement for Higher National Diploma.

Dr. Muktar Hussaini .…………………………..

Head of Department Sign & Date

# **DECLARATION**

I hereby declare that this dissertation is the result of my own investigation, expect otherwise stated. I also declare that it has not been previously or concurrently submitted as a whole for any other Higher Diploma at HAFEDPOLY.

ABUBAKAR ALHASSAN

HND/CS/19/012

Signature………………………... Date………………………...

# **DEDICATION**

This project is dedicated to Almighty Allah (SWT) for given me the opportunity of compilation of this project and to my lovely families and friends for their prayers and financial support. I am strongly grateful for that. I also dedicated this project to the entire staff of this department.

# **ACKNOWLEDGEMENT**

All praise is due to Him (Allah) for giving me the opportunity and courage to pursue this project to its logical and certainly to him is all praise.

My special thanks and gratitude goes to my parents who help us towards my education financially and morally. I also wish to express my gratitude to my noble and unique supervisor in person of MR. AMAONWU ONYEBUCHI for his Constructive criticism and advise towards to success of this project. I equally wish to express my profound gratitude and special thanks to our parents, brothers and sisters for their advice, prayers and support towards our education.

And finally my special thanks also goes to our best friends that gave us their special contributions towards to the completion of this project.

# **TABLE OF CONTENTS**

[**ABSTRACT** ii](#_Toc106253871)

[**APPROVAL/CERTIFICATION PAGE** iii](#_Toc106253872)

[**DECLARATION** iv](#_Toc106253873)

[**DEDICATION** v](#_Toc106253874)

[**ACKNOWLEDGEMENT** vi](#_Toc106253875)

[**TABLE OF CONTENTS** vii](#_Toc106253876)

[**LIST OF TABLES** ix](#_Toc106253877)

[**LIST OF FIGURES** x](#_Toc106253878)

[**CHAPTER ONE** 1](#_Toc106253879)

[**1.0** **OVERVIEW** 1](#_Toc106253880)

[**1.1** **HISTORICAL BACKGROUND** 2](#_Toc106253881)

[**1.2** **STATEMENT OF THE PROBLEM** 2](#_Toc106253882)

[**1.3** **AIM AND OBJECTIVES OF THE STUDY** 3](#_Toc106253883)

[**1.4** **SIGNIFICANCE OF THE STUDY** 3](#_Toc106253884)

[**1.5** **SCOPE AND LIMITATION** 3](#_Toc106253885)

[**1.6** **DEFINITION OF TERMS** 4](#_Toc106253886)

[**1.7** **PROJECT ORGANIZATION** 5](#_Toc106253887)

[**CHAPTER TWO** 6](#_Toc106253888)

[**2.0** **OVERVIEW** 6](#_Toc106253889)

[**2.1** **MOST CURRENT WORK WITH REFERENCES** 6](#_Toc106253890)

[**2.2** **CONSTRAINTS IN SCHOOL TIMETABLING** 8](#_Toc106253891)

[**2.2.1** **HARD CONSTRAINTS** 8](#_Toc106253892)

[**2.2.2** **SOFT CONSTRAINTS** 9](#_Toc106253893)

[**2.3** **ALGORITHMS USED IN SCHOOL TIMETABLING** 9](#_Toc106253894)

[**2.3.1** **GRAPH COLORING (GRAPH HEURISTICS)** 9](#_Toc106253895)

[**2.3.2** **HILL CLIMBING (HC)** 9](#_Toc106253896)

[**2.3.3** **TABU SEARCH (TS)** 10](#_Toc106253897)

[**2.3.4** **SIMULATED ANNEALING (SA)** 10](#_Toc106253898)

[**2.3.5** **GENETIC ALGORITHMS (GA)** 10](#_Toc106253899)

[**2.4** **FEASIBILITY STUDY** 11](#_Toc106253900)

[**2.4.1** **TECHNICAL FEASIBILITY** 12](#_Toc106253901)

[**2.4.2** **OPERATIONAL FEASIBILITY** 12](#_Toc106253902)

[**2.4.3** **ECONOMIC FEASIBILITY** 13](#_Toc106253903)

[**2.4.4** **SCHEDULE FEASIBILITY** 13](#_Toc106253904)

[**CHAPTER THREE** 14](#_Toc106253905)

[**3.0** **OVERVIEW OF THE CHAPTER** 14](#_Toc106253906)

[**3.1** **ANALYSIS OF THE CURRENT SYSTEM** 14](#_Toc106253907)

[**3.1.1** **EXISTING SYSTEM CHARACTERIZATION** 15](#_Toc106253908)

[**3.2** **ANALYSIS OF THE NEW SYSTEM** 15](#_Toc106253909)

[**3.2.1** **NEW SYSTEM CHARACTERIZATION** 15](#_Toc106253910)

[**3.2.2** **NEW SYSTEM FLOWCHART** 16](#_Toc106253911)

[**3.2.3** **NEW SYSTEM DATA FLOW CHART** 17](#_Toc106253912)

[**3.3** **FEASIBILITY STUDY** 17](#_Toc106253913)

[**3.4** **NEW SYSTEM REQUIREMENT** 18](#_Toc106253914)

[**3.4.1** **HARDWARE REQUIREMENT** 18](#_Toc106253915)

[**3.4.2** **SOFTWARE REQUIREMENT** 19](#_Toc106253916)

[**CHAPTER FOUR** 20](#_Toc106253917)

[**4.0** **OVERVIEW OF DESIGN** 20](#_Toc106253918)

[**4.1** **SOFTWARE DESIGN** 20](#_Toc106253919)

[**4.1.1** **UML DIAGRAMS. THIS MUST INCLUDE THE FOLLOWING:** 21](#_Toc106253920)

[**4.2** **DATABASE DESIGN** 23](#_Toc106253921)

[**4.2.1** **E – R DIAGRAMS** 23](#_Toc106253922)

[**4.2.2** **TABLES – EXPLAINING ALL FIELDS AND THEIR DATA TYPES** 24](#_Toc106253923)

[**4.2.3** **SQL STATEMENTS** 24](#_Toc106253924)

[**4.3** **CODING FLOW** 25](#_Toc106253925)

[**4.4** **USER INTERFACE SPECIFICATIONS** 26](#_Toc106253926)

[**4.5** **SYSTEM PACKAGING AND DEPLOYMENT** 26](#_Toc106253927)

[**CHAPTER FIVE** 27](#_Toc106253928)

[**5.0** **SYSTEM TESTING** 27](#_Toc106253929)

[**5.1** **CHANGE OVER PLAN** 28](#_Toc106253930)

[**5.2** **IMPLEMENTATION ENVIRONMENT** 29](#_Toc106253931)

[**CHAPTER SIX** 30](#_Toc106253932)

[**6.0** **PROJECT SUMMARY** 30](#_Toc106253933)

[**6.1** **PROJECT OUTCOMES AND REQUIREMENT FULFILMENTS** 30](#_Toc106253934)

[**6.2** **RECOMMENDATIONS** 31](#_Toc106253935)

[**6.3** **CONCLUSION** 31](#_Toc106253936)

[**REFERENCES** 32](#_Toc106253937)

[**APPENDIX A: SOURCE CODES** 35](#_Toc106253938)

[**APPENDIX B: SNAP SHOT** 35](#_Toc106253939)

# **LIST OF TABLES**

Table 1: Hardware requirement specifications……………………………………..…….... 18

Table 2: Software requirement specifications…………………………………..…………. 19

Table 3: Admin Table…………………………………………………………………..…...24

Table 4: Classroom Table…………………………………………………………………...24

Table 5: Semester Table………………………………………………………………..…...25

Table 6: Subject Table………………………………………………………………….…...25

Table 7: Teachers Table………………………………………………………………..…...25

# **LIST OF FIGURES**

Figure 1: New system flowchart…………….……………………………………....….... 16

Figure 2: New system Data flow diagram……………………….……………………….. 17

Figure 3: Use Case Diagram………………………………….……………….…………...22

Figure 4: E R Diagram……………………………………………………..………………24

Figure 5: Homepage 1……………………………………………….……..………………27

Figure 6: Homepage 2……………………………………….…………..…………………28

Figure 7: Admin Login Page………………………………...…………………..…………28

Figure 8: View Teacher’s Record…………………………...………………………..…….29

Figure 9: View All Subject’s Records………………………………………………….… 29

Figure 10: View Classroom………………………………..…………………..……...……30

Figure 11: Generate Timetable page……………………………………………………….31

Figure 12: View Teacher’s Timetable……………………………………………..…..…...32

Figure 13: Levels of testing…………………………………………………..…..………...34

# **CHAPTER ONE**

**INTRODUCTION**

## **OVERVIEW**

This project examines the general and well-established research into the timetable scheduling problem. Timetabling concerns all activities with regard to making a timetable that must be subjective to different constraints. According to Collins Concise Dictionary (4th Edition) “A timetable is a table of events arranged according to the time when they take place.”

Academic resource planning is a difficult administrative practice that is based on broad analysis of data and proper allocation of resources associated with the educational structure. These resources which include people, space and time require proper allocation through workable and attractive schedules formed from decisions informed by accurate information. One of such resource allocation problems faced in academic administration is the University Timetabling Problem.

The University Timetabling Problem (UTP) can be defined as the task of assigning a number of events, such as lectures, exams, meetings, and so on, to a limited set of timeslots and venues such as class rooms, exam rooms and laboratories, in accordance with a set of constraints. The University Timetabling Problem (UTP) can be divided into Course and Examination Timetabling.

In similarity to course timetabling, the Examination Timetable Scheduling is basically a way of finding an exact time allocation within a slated time period to assign to some other resources in such a way as to satisfy some constraints, i.e. students, Exam-Hall and exams, to timeslots. The decisions involved are: Exam-hall assignment and timeslot assignment. The focus of this study is on the Course and Exam Timetable Scheduling system.

## **HISTORICAL BACKGROUND**

Historical Background of the University dates back to 22nd June, 2010 when the State Executive Council constituted a 9–Member Technical Committee chaired by Prof. Munzali Jibril, a former Executive Secretary of the National Universities Commission (NUC). The Committee submitted its Report on 31st December, 2010 which included an Academic Brief and the Draft University Law. The State Executive Council, in February 2013, considered the Report and approved the establishment of a university to be called “Jigawa State University, Kafin Hausa”. The University SULE LAMIDO UNIVERSITY, KAFIN HAUSA, was assented to by the Governor on 23rd December, 2014. It was also licensed by the NUC on 17th July, 2013 as the 39th State–Owned University and 129th in the Nigerian University System (NUS).

The Department of Mathematics and Computer Science of Sule Lamido University Kafin Hausa was established as one of the founding Department under Faculty of Natural and Applied Sciences of the University, on the approval of National Universities Commission (NUC). This Department has two units, Mathematics Unit & Computer Unit, and offers Bachelor of Science in Mathematics and Bachelor in Computer Science degrees and is headed by Dr. Muhammad Jamilu Ibrahim.

## **STATEMENT OF THE PROBLEM**

The systems available currently in the department of computer science can build or generate a set of timetables, but still, have issues with generating a clash-free and complete timetable and also takes time. Some of the problems related with the existing system are stated below:

1. Course and Exam Timetable Scheduling in the existing system is laborious and requires a lot of effort and time.
2. In the existing system, scheduling each class courses and exam on paper and creating proper seating arrangement for it requires lot of man hours and much planning.

## **AIM AND OBJECTIVES OF THE STUDY**

The main aim of this work is to design a unified, scalable and flexible Lecture and Exam Timetabling System for the department of computer science in such a way that they are optimal and complete with little or no redundancy through the development of the viable timetabling software.

The objectives of this project are:

1. To Design a reliable software that will manage lecture and exam timings and define different types of exams like first term, second term.
2. To be able to optimize the algorithm used in today’s timetable systems to generate the best of timetabling data with fewer or no clashes.
3. To create a reliable database for Courses and lectures, with unique features like Course code and course name.
4. The system will minimize time required to complete the entire process for timetable scheduling.

## **SIGNIFICANCE OF THE STUDY**

This project is expected to ease the process of lecture and Exam Scheduling. Ensure fast and reliable Timetable generation and retrieval if missing. The knowledge of the new system designed would make it very convenient for students and staffs/lectures to view their lecture timetable and also for the Exams at their comfort zone or anywhere they are in the world. Finally, the research projects would serve as reference materials to other scholars interested for further research on the related topics.

## **SCOPE AND LIMITATION**

The scope of this project is to take and display the record of courses, lecturers and Exam halls, and then use those records to generate Automated Timetable. And the project is based on inputting data, outputting Lecture and Exam timings on the timetable processing, using web design tools such as HTML, CSS for designing the platform and PHP for server site and MySQL as a database. And is only limited to Department of Computer Science, Sule Lamido University, Kafin Hausa, Jigawa State and not otherwise.

## **DEFINITION OF TERMS**

**Timetable:** a list showing the time at which a particular event will happen.

**Schedule:** a plan that lists all work that you have to do when you must do each of them.

**Computerization:** This is the process of using a computer system to carry out work which should have been done by man.

**Resources:** something that can be used to achieve an aim.

**Courses:** a series of lessons or lectures on a particular subject.

**Algorithm:** a set of rule that must be followed when solving a particular problem.

**Information System:** It is a collection of procedures, people instructions and equipment to produce information in a useful form.

**Technology:** It is a study of techniques or process of mobilizing resource (such as information) for accomplishing objectives that benefit man and his environment.

**Information:** Information can be defined as the process of gathering, transmitting, receiving, storing and retrieving data or several items put together to convey the desired message.

**Computer Network:** Computer Network is a system that connects two or more computers together using a communication link.

**Databases:** A systematically arranged collection of computer data, structured so that it can be automatically retrieved or manipulated. It is also called databank.

**File Transfer:** Any kind of computer file can be sent via the Internet from one Internet user to another. Table of accounts on spreadsheets, design by graphic artists, music sound files etc, can all be exchanged in this way.

## **PROJECT ORGANIZATION**

This project work is organized into six chapters which include chapter one, introduction of the study, overview, historical background, statement of the problem, objective of the study significance of the study, scope and limitation, definition of terms and project organization. Chapter two give the literature review of the research and feasibility study, chapter three describe the system analysis of the current and proposed system and justification of the analyzed system, chapter four consist of system design, flowchart of the design system and algorithm of the design system, chapter five contains testing implementation suggestion, and lastly chapter six consist of summary, conclusion and recommendation.

# **CHAPTER TWO**

**LITERATURE REVIEW**

## **OVERVIEW**

This chapter contain general understanding about the literature reviews of Online Timetable Scheduling System, Sule Lamido University as a case study. It will also focus on some relevant issue concerning to the topic of related research and the feasibility study is carried out to test the ability of the proposed system and whether is worth being implemented.

## **MOST CURRENT WORK WITH REFERENCES**

Some previous works that has been carried out in this area to address the problem of manual Timetable Scheduling System are:

Obit, J.H. (2010) stated that the university courses programming is part of a wide group of programming problems, commonly known as "Timetabling". These problems can be defined in a general way as the allocation of a set of events to a space and time fulfilling a series of restrictions that involve a limited set of resources (Obit, 2010).

Babaei et al. (2015) introduced a new approach to solving the timetabling problem in universities, a multiple agent technique based on multiple hybrid meta-heuristics with Local Search graph coloring. Tests were performed to compare different techniques in instances from the literature with diverse characteristics. The authors conclude that exact approaches do not have good efficiency in solving this problem because of the growing complexity with a higher number of students. Furthermore, they highlight the advantages of using multi-agent approaches, such as greater independence in the allocation of classes between different departments of the university. (Babaei et al.,2015)

Lewis & Thompson (2015) proposed a two-stage meta-heuristic for the university timetabling problem. The first stage is intended to achieve a feasible solution without considering soft constraints. The second stage, considering hard and soft constraints, seeks to improve the solution by increasing solution space connectivity using a neighborhood operator. The approach was tested in instances of the International Timetabling Competition (ITC) 2007, and the authors found that the quality of the solution is generally better with increased solution space connectivity, as the technique generated better results than the methods in the literature for most of the instances that were analyzed. (Lewis et al., 2015).

Veenstra & Vis (2016) analyzed the performance of three proposed techniques to reprogram a school timetable: an exact method of Integer Linear Programming (ILP) method, a simple general rule and a heuristic technique. The techniques were tested with the data of five schools in the Netherlands in two scenarios (minor or major alterations to the timetable). The results show that, for minor alterations, the heuristic achieved the optimal solution in 50% of the tests. When major alterations were required, the ILP method could not find the optimal solution in the established computational time (12 hours).

Jardim et al. (2016) solved and optimized the timetabling problem of a department of the Federal Fluminense University (FFU) using the Iterated Local Search (ILS) heuristic, which had better results than those of the tool used by the department at the time when the study took place. In the article written by Bucco et al. (2017), a Mixed Integer Linear programming (MILP) model was applied to the timetabling problem of university courses, focusing on optimizing the use of classrooms. To enable the processing of the proposed approach, the problem was divided into two sub-problems. The first addressed the generation of the timetable, while the second focused on classroom distribution. Tests were conducted with real data from a Brazilian university, demonstrating that the proposed approach could reduce the number of classrooms required. (Bucco et al., 2017)

Again, there are hard and soft constraints which can be easily inserted or removed while the specifications are maintainable. However, there were some drawbacks as necessity of generalized methodology, specialists’ skills while the problems are varying by concerning type of institute. Further, Anuja Chowdhary and his colleagues also introduced an automated timetabling system of handling soft and hard constraints wisely with the limitations of mentioning the logic of the system (Chowdhary, 2014). In a different research, Nelishia Pillay says even though there are number of researches found in timetabling few of them only developed as software. (Pillay, 2014).

## **CONSTRAINTS IN SCHOOL TIMETABLING**

Constraints are – as mentioned in above – divided into hard constraints (requirements) and soft constraints (preferences). The following constraints are some of the constraints that can exist in school timetabling. Not all of these are used in every problem and in some problems there can exist even more constraints.

### **HARD CONSTRAINTS**

1. Every event in every course is assigned a time slot.
2. All events should be in the right kind of room.
3. No student group has two events at the same time.
4. No lecturer has two events at the same time.
5. No two events are scheduled in the same room at the same time.
6. No event is in a room with less capacity than the number of students at the event.
7. Some classrooms may be too small for some subjects with lots of students.
8. Certain classes with special needs, e.g., computers or lab equipment must be taught in a room assigned for this.

These constraints are referred to as hard constraints, which mean that they are absolutely necessary for the solution to be valid. (Fredrikson et al., 2016)

### **SOFT CONSTRAINTS**

1. There can be preferences for certain classes of when the class should be held, e.g., during the morning or afternoon.
2. Some teacher may prefer to teach their subject during certain periods of the day.
3. It may be preferable to fill up smaller rooms first if possible.
4. It may be preferable to have courses distributed during the week so that the same course is not scheduled two times the same day.
5. It may be preferable to avoid having to much gaps in the schedule.

## **ALGORITHMS USED IN SCHOOL TIMETABLING**

This section will provide information about some popular algorithms that have been used for school timetabling problems. Fitness will be used to describe the quality of the solutions where “more fit” or “better fitness” is when a solution has less broken constraints i.e., a lower number of total penalty points.

### **GRAPH COLORING (GRAPH HEURISTICS)**

The use of graph coloring which has been identified as an efficient and fastest algorithm where Colors are assigned to vertices, such that no neighboring vertices attains exact or similar color (Tan et al., 2018) Other scholars have concluded that graph heuristics, according to their own, are not really suitable techniques for solving global timetabling problems and they have not even created feasible solutions for some of the problem cases. (Nelishia, P., & Ender, O., 2019). Latest evidence, however, has shown that they are successful in generating an actual meta-heuristics remedy. (Ahmed, K., & Keedwell, E., 2017).

### **HILL CLIMBING (HC)**

Hill Climbing (HC) or simple climbing is a classical local quest technique. Hill climbing is fast and quick to execute. The downside, however, is that the localized Optima is quickly attained. Recently (Hamed et al., 2019), suggested a late acceptance technique for scaling the HC. The mechanism postpones the step of comparison between the candidate solution and the actual (best) response.

### **TABU SEARCH (TS)**

Tabu search suggested by Glover, works in a manner comparable to HC but adding a memory to enable analysis of the differentiation of search space. Tabu search has been enforced as a search algorithm that generates course schedules by minimizing losses heuristically over an utterly impossible solution (Kheiri et al., 2017).

### **SIMULATED ANNEALING (SA)**

Simulated annealing (SA) was inspired by the physical annealing mechanism to heat up a solid to an extreme temperatures and cool it off slowly until everything crystallizes and there are no more changes. SA begins with an initial solution built using a positive heuristic and an optimal solution are always approved, whereas a much worse solution is only acknowledged with certain probabilities (Suliadi et al., 2018).

### **GENETIC ALGORITHMS (GA)**

Genetic Algorithms is a population-based search that produces appropriate results from one generation to another through the theory of biological evolution (Burke et al., 2009). In order to avoid infeasible solutions (Esraa & Ghada, 2016), advised using only the mutation operator to produce solutions for the offspring. Experimental results found that the genetic algorithm used by a generic crossover operator was less effective than their approach. Because of the direct chromosome representation that created infeasible offspring solutions, they introduced a repair mechanism to solve the infeasibility. There are two main stages in the genetic algorithm; the selection and the crossover.

* 1. **Selection**

There are few ways in selecting which chromosomes to be crossed. Some of these are elitism selection, roulette-wheel selection and tournament selection (Yamazaki, 2014).

1. **Crossover**

It may vary which genes are carried over when two chromosomes are being crossed. To decide this, there are few different methods: Some of them are single point crossover, two-point crossover and uniform crossover (Yamazaki, 2014)

Conclusively, in this Timetable Scheduling System, we are going to adopt Genetic Algorithm in generating the automatic timetable for Sule Lamido University (SLU) kafin Hausa, and the constraints explained above are going to be used in this project.

## **FEASIBILITY STUDY**

The feasibility study is carried out to test whether the proposed system is worth being implemented. Feasibility study is a test of system proposed regarding its work ability, its impact on the organization ability to meet user needs and effective use of resources. It is usually carried out by a small number of people who are familiar with the information system techniques, understand the part of the business or organization that will be involved or effected by the project and are skilled in the system analysis and design process. The reason is to also determine whether the development of the proposed system will bring positive or negative impact to the organization. If the positives dominate the negatives, then we can conclude that the system is feasible. A feasibility study can be performed in three different ways. These ways are:

* + Technical Feasibility
  + Economic Feasibility
  + Operational Feasibility
  + Schedule Feasibility

### **TECHNICAL FEASIBILITY**

Technical feasibility has to cope with the technical requirements of the proposed system. The technical requirements are then compared to the technical capability of the organization, in this case he hospital. By taking all possible consideration in developing the proposed system, the resources available in the organization are also studied. Some of the questions being answered are;

* Does the technology have the capacity to handle the solution?
* Manpower – Programmers, testers and debuggers?
* Do we currently possess such technology?
* Are the current technical resources sufficient for the new system?

### **OPERATIONAL FEASIBILITY**

This type of feasibility study involves human resources available for the proposed system. This actually involves protecting whether the proposed system will be used if it is developed and implemented. The operational feasibility is actually a measure of how well the proposed system will eventually solve the problem of the currently existing one,

Some of the questions that need to be answered under operational feasibility study include:

* + Does the current mode provide end users and management with timely, accurate and useful formatted information?
  + Does the current mode of operation provide cost-effective information services to the organization?
  + If the system is developed, will it be used?
  + Will the development of the system reduce time (operation) considerably?
  + Will the proposed system really benefit the organization?
  + How do the end users feel about their role in the old system?
  + Can or will the end-users and management adapt to the changes?

### **ECONOMIC FEASIBILITY**

This seems to be the most frequently used method for evaluating the effectiveness of a new system. In economic feasibility, the procedure is to determine the benefits and savings that are expected from the candidate system and compare them with the costs. If the benefits outweigh the costs, then it is time for developing and implementing the proposed system.

Possible questions raised in economic feasibility study are:

* + Do benefits outweigh costs?
  + Estimated cost of hardware
  + What are the savings that will result from the system?

### **SCHEDULE FEASIBILITY**

This is defined as the probability of a project to be completed within its scheduled time limits, by a planned due date. If a project has a high probability to be completed on-time, then its schedule feasibility is appraised as high. In many cases a project will be unsuccessful if it takes longer than it was estimated: some external environmental conditions may change, hence a project can lose its benefits, expediency and profitability. If a work to be accomplished at a project does not fit the timeframes demanded by its customers, then a schedule is unfeasible (amount of work should be reduced or other schedule compression methods applied).

# **CHAPTER THREE**

**SYSTEM ANALYSIS**

## **OVERVIEW OF THE CHAPTER**

System Analysis is the study of a business problem domain to recommend improvements and specify the business requirements and priorities for the solution. It involves the analyzing and understanding a problem, then identifying alternative solutions, choosing the best course of action and then designing the chosen solution.

It involves determining how existing systems work and the problems associated with existing systems. It is worthy to note that before a new system can be designed, it is necessary to study the system that is to be improved upon or replaced, if there is any.

## **ANALYSIS OF THE CURRENT SYSTEM**

Timetabling is the whole process concerned with making a timetable having events arranged according to a time when they take place which must be subject to the timing constraints of each entity placed in the table. University timetabling in this context refers to the rigorous task educational center staff in a Sule Lamido University (SLU) kafin Hausa undergo to draw up timetables that satisfies various courses that should compulsorily be inherent in the final timetable solution.

These courses are usually taught by varied lecturers in different departments who may also wish to specify some timing constraints on their courses. Given all the courses and course details, the educational center staff in Sule Lamido University (SLU) is charged with the responsibility of creating a near optimal timetable which would serve as a guide for academic activities in the university.

The traditional manual timetabling system is time-consuming, resource-intensive, involves many steps and requires re-processing the same data several times.

### **EXISTING SYSTEM CHARACTERIZATION**

The traditional manual generations of timetables has a number of characteristics which may include the following:

* It is subjective and can be made better through collaboration with the different entities involved.
* Repeated time allocations may be made for a particular course thereby leading to data redundancy.
* A lot of administrative error may occur as a result of confusing time requirements.
* Timetable generation by center staff may have a slow turnaround.
* Final generated timetable may not be near optimal as a result of clashing course requirements and allocations.
* It generates a lot of paperwork and is very tasking.
* It is not flexible as changes may not be easily made.

## **ANALYSIS OF THE NEW SYSTEM**

The proposed systems were developed to solve the timetabling problem being faced by universities every academic year and reduce high cost and slow turnaround involved in the generation of near-optimal timetables.

The system has capabilities for input of the various courses, halls of lectures, departments, programs, buildings, lecturers and the specification of a few constraints from which the timetable is constructed. The proposed timetabling system for this project seeks to generate near optimal timetables using the principles of genetic algorithm (selection and crossover).

### **NEW SYSTEM CHARACTERIZATION**

The timetable generation process by the education center staff is:

* Unlike the manual timetabling system, the system offers flexibility.
* It utilizes minimal processing/computing power.
* It greatly reduces the time needed to generate near-optimal timetables.
* It provides an easy means for data entry and revision through an intuitive interface.
* It increases productivity.
* Timetables generated are between to 60% - 80% optimum.
* It almost eliminates paperwork.
* It simplifies the timetabling process.

### **NEW SYSTEM FLOWCHART**

Processing the inputs

Generate Timetable

If required inputs are entered

Yes

No

Display Generated timetable

Add Lecturer

Add Courses

Add Lecture Room

Figure 1: New system flowchart

### **NEW SYSTEM DATA FLOW CHART**

**ADMIN**

**STUDENT**

**STAFF**

**DATABASE**

Figure 2: New system Data flow diagram

## **FEASIBILITY STUDY**

After the new system, whole system is required to be converted into computer understanding language. This means that coding the system into computer programming language. It is an important stage where the procedure is transformed into control specification by the help of a computer language. This is also called a programming phase in which the programmer converts the program specification into computer instructions, which we refer as program. The program coordinates the data movement and control the entire process in a system. It is generally applied that the program is modular in nature, this helps in fast development, maintenance, and future changes if required.

The method of designing this project is the use PHP, HTML, CSS and Java script, and the use of database application which is Mysql where the data will be stored. PHP and Apache are linked with Mysql database so that the manipulations will be done, it will be stored in the database.

## **NEW SYSTEM REQUIREMENT**

The requirement specification describes what are the hardware and software required by a system to run the online birth registration software.

### **HARDWARE REQUIREMENT**

The hardware is the physical component of the computer system that we can see, feel and touch. The hardware requirements of the software include:

Table 1: Hardware requirement specifications

|  |  |
| --- | --- |
| **HARDWARE** | **MINIMUM HARDWARE REQUIREMENT** |
| Laptop or Desktop | Any Model |
| Ram | 1GB and Above |
| Hard Disk Space | 20GB and Above |
| Processor Speed | 2GHZ |
| Display | 1024 x768 High Color Recommended |
| Uninterruptible Power Supply(UPS) | 650VA,100-240AC,Mercury |
| CD/DVD ROM Drive | 52x |
| Network Materials | LAN Switch,cat5e UTP Cable,RJ-45,Clip,Modem |
| Printer | HP laser jet printer or HP Desk Jet Printer |

### **SOFTWARE REQUIREMENT**

Software is a set of program or instruction written in computer understandable language for overall optimal and maximal utilization of the hardware. The software requirements used for this project are:

Table 2: Software requirement specifications

|  |  |
| --- | --- |
| **SOFTWARE** | **MINIMUM SOFTWARE REQUIREMENT** |
| Operating System | Any of windows XP,98,7, 8, 10 or 11 |
| Server | Any of WAMP or XAMPP server |
| Browser | Any of internet Explorer, Edge, Opera, Google chrome, Mozilla Firefox etc |
| Microsoft office | 2003/2007/2010/2013/2016 |

# **CHAPTER FOUR**

**SYSTEM DESIGN**

## **OVERVIEW OF DESIGN**

While designing the web application portal, a three tier architecture for application development was followed. The presentation tier occupies the front end design of the application. It relates to every entity with which the user interacts. It accepts user inputs and actions, and then sends this information to the data tier through the application tier for further processing. The online learning system portal accepts input in the form of student’s personal detail, student’s registration details, course details etc. The second tier is application tier, it serves as an intermediary to the presentation tier and the data tier. It applies business logic to the input received from the presentation tier. The business logic is applied combining the information collected from upper and lower layers of the three tier architecture. When a student asks for a report to be retrieved, on the basis of this command data is fetched from the database and the required business logic is applied and finally a report is displayed to the user. The third and final tier is the data tier which is concerned with the design of the database to be used with the web application. It is concerned with maintaining central servers and its information. The data tier is kept independent of application servers and business logic to promote data security. The database is developed using MySQL keeping in mind the authenticity concerns of different users. All the necessary primary keys and access controls are depicted with utmost care.

## **SOFTWARE DESIGN**

System design is the specification or construction of a technical, computer-based solution for the Institution requirements identified in a system analysis. It gives the overall plan or model of a system consisting of all specifications that give the system its form and structure i.e. the structural implementation of the system analysis.

System design is the most important phase in the development of any information, it is undertaken based on the user demands and analysis of the new system, in this phase the design proceeds into two stages, the preliminary or general design and structural or detail design.

### **UML DIAGRAMS. THIS MUST INCLUDE THE FOLLOWING:**

The Unified modeling language is an object-oriented system notation that provides a set of modeling conventions that is used to specify or describe a software system in terms of objects. The Unified Modeling Language (UML) has become an object modeling standard and adds a variety of techniques to the field of systems analysis and development hence its choice for this project.

* **USE CASE DIAGRAMS**

Use case diagrams describe what a system does from the standpoint of an external observer. The emphasis of use case diagrams is on what a system does rather than how. They are used to show the interactions between users of the system and the system. A use case represents the several users called actors and the different ways in which they interact with the system.

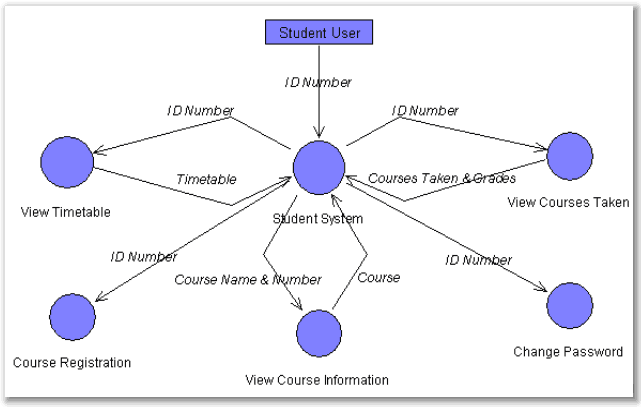
****

Figure 3: Use Case Diagram to show the interaction between the timetabling system and user

## **DATABASE DESIGN**

Database design is the process of producing a detail data model of a database. It is a collection of stored data organized in such a way that the data requirements are satisfied by the database. The general objective is to make information access easy, quick, inexpensive and flexible for the user. There are some specific objectives like controlled redundancy from failure, privacy, security and performance. A collection of relative records make up a table. To design and store data to the needed database tables are prepared. One essential setting for a database is:

**Primary Key:** The field that is unique for all the record occurrences.



### **E – R DIAGRAMS**

An entity relationship (ER) diagram is a database model that describes the attributes of entities and the relationships among them. An entity-relationship (ER) diagram is specialized graphic that illustrate the interrelationship between entities in database (Nembhard, 2013). To avoid insertion, updating, or deletion anomalies, entity relationship diagrams must be normalized (Peck, 2003). Hence, the database of this system is normalized to third normal form by eliminating fields that do not depend on the primary key.

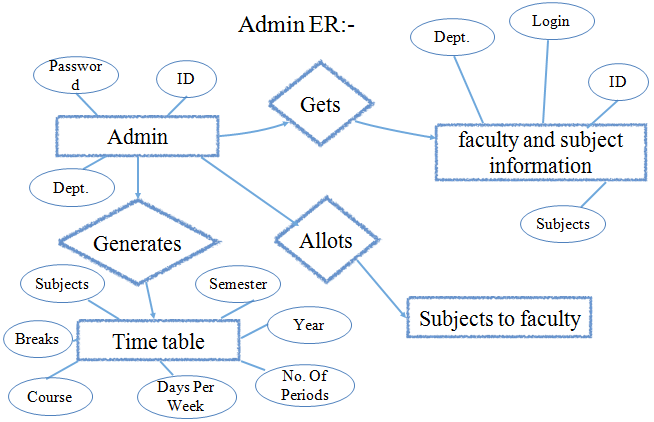


Figure 4: E R Diagram

### **TABLES – EXPLAINING ALL FIELDS AND THEIR DATA TYPES**

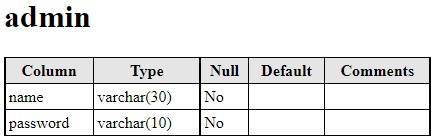


Table 3: Admin Table

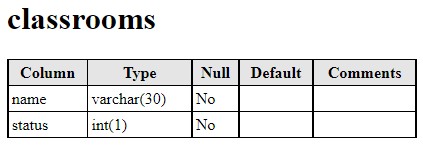


Table 4: Classroom Table

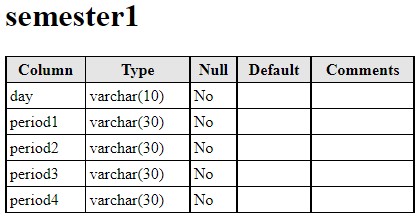


Table 5: Semester Table

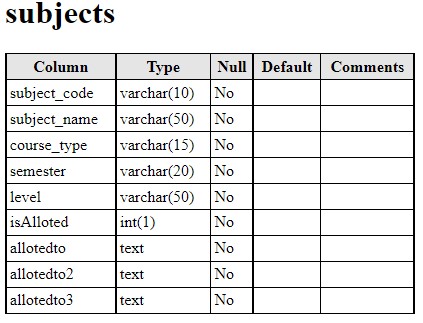


Table 6: Subject Table

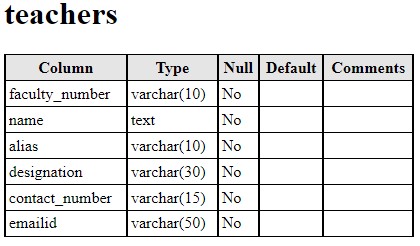


Table 7: Teachers Table

### **SQL STATEMENTS**

Structured Query Language is a special-purpose programming language designed for managing data in a relational database management, it is used to select data from a database, the data then returned s stored in a result table.

Some of the Most Important SQL Commands are listed below:

**SELECT** - extracts data from a database

**UPDATE** - updates data in a database

**DELETE** - deletes data from a database

**INSERT INTO** - inserts new data into a database

**CREATE DATABASE** - creates a new database

**ALTER DATABASE** - modifies a database

**CREATE TABLE** - creates a new table

**ALTER TABLE** - modifies a table

**DROP TABLE** - deletes a table

**CREATE INDEX** - creates an index (search key)

**DROP INDEX** - deletes an index

## **CODING FLOW**

The coding flow show the step by step procedure in which a user must be followed to design a program, the following program structure show how a PHP and HTML coding flow look like, that contain a Procedure, Function and Method.

**Procedure**: A procedure is a group of sequential statements that have a name in common and can be executed by calling the group (by name, of course) from some other place in the program.

**Function**: A function is a sequence of statement that returns a value when executed in the program.

**Method**: The method simply refers to the logic part of the program that is assign by the software programmer.

## **USER INTERFACE SPECIFICATIONS**

The user interface of a system is the environments in which the user performs a specific task below are the snapshot of basic part of the software and their explanation.

**HOMEPAGE**

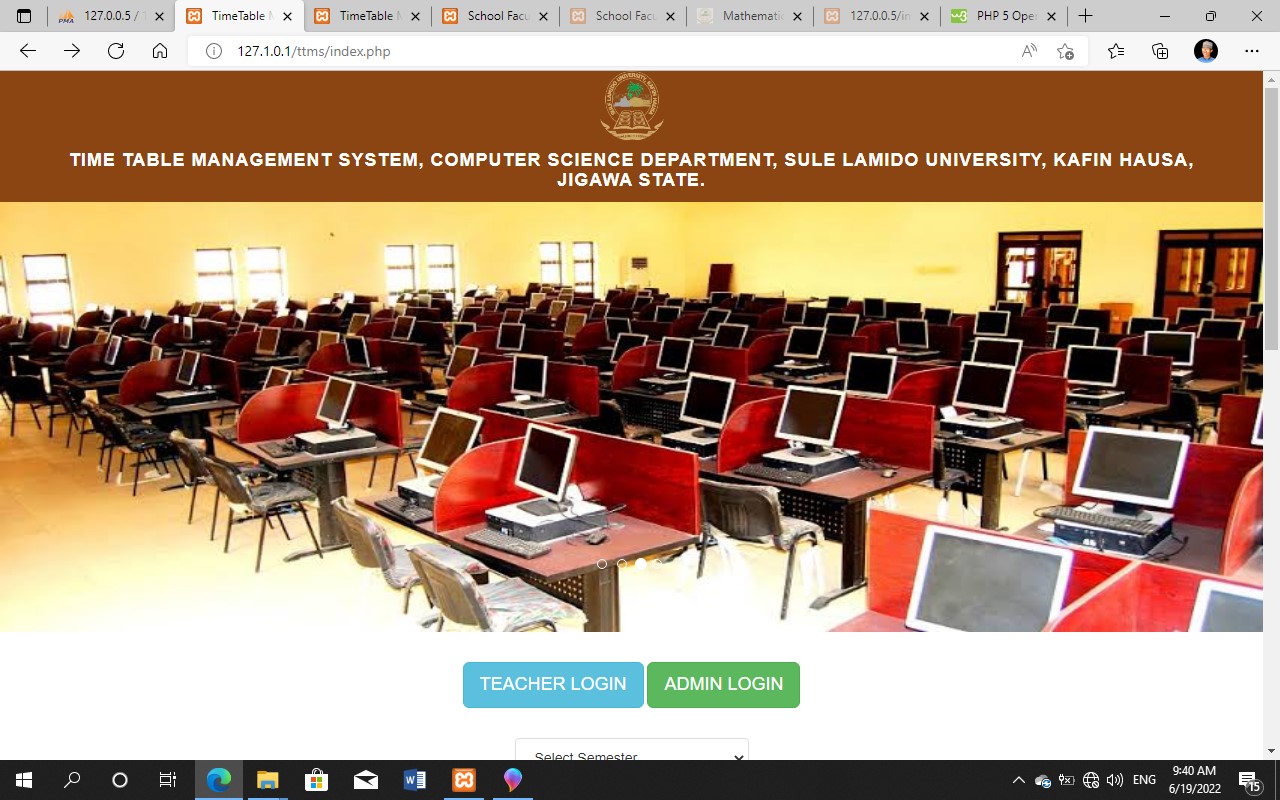


Figure 5: Homepage 1

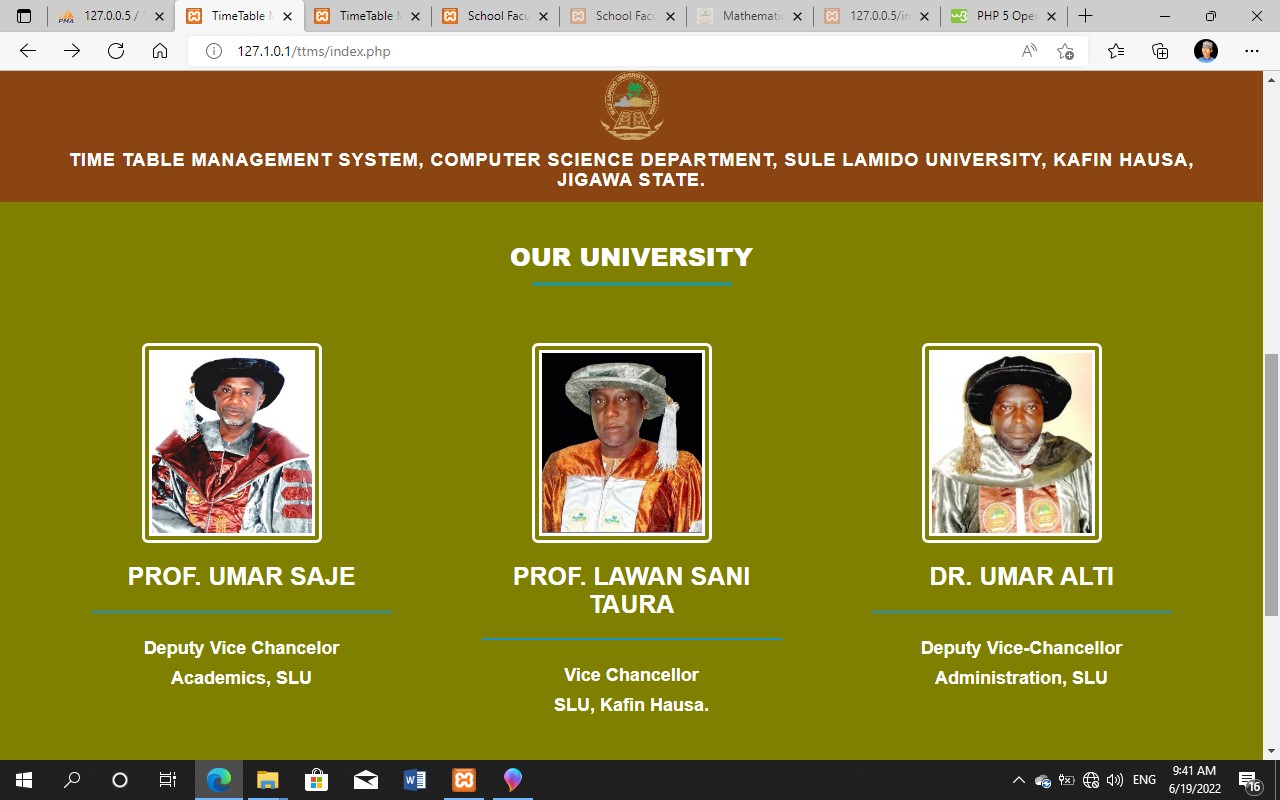


Figure 6: Homepage 2

**ADMIN LOGIN PAGE**

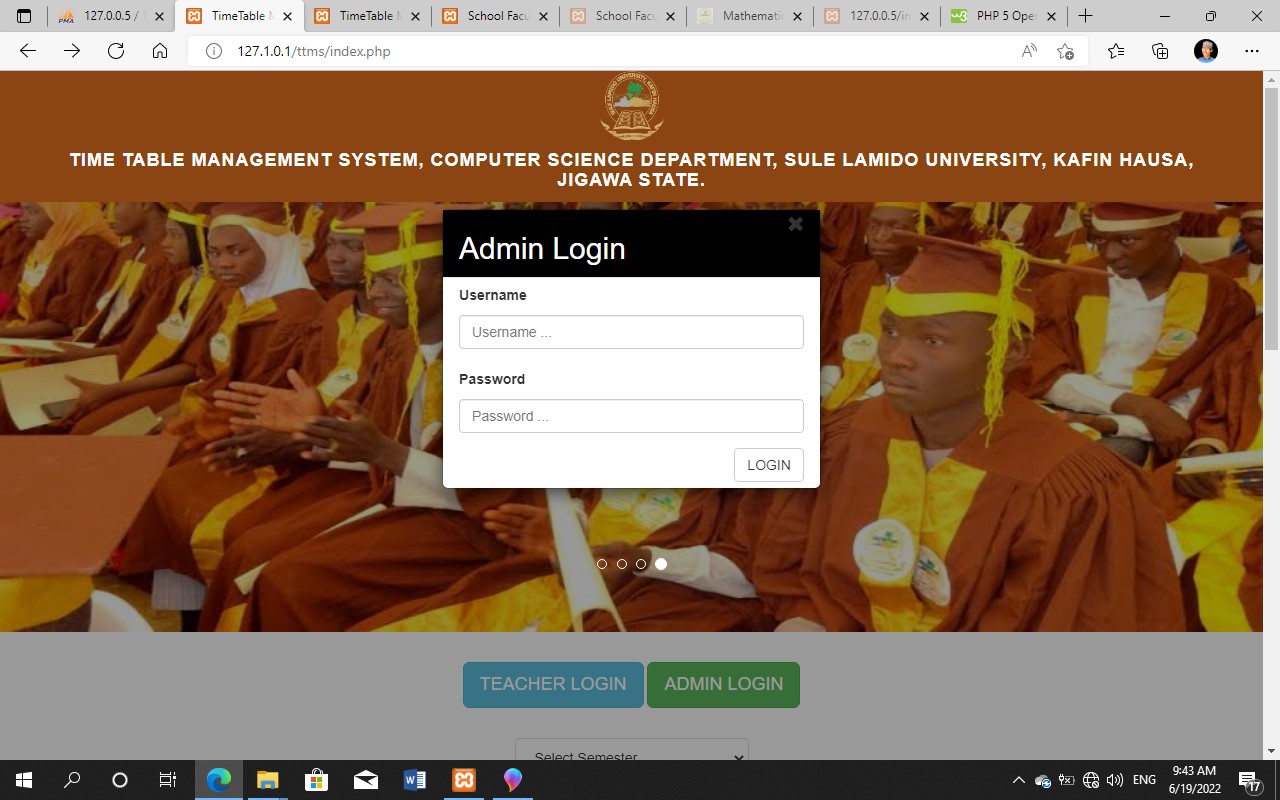


Figure 7: Admin Login Page

**VIEW TEACHER’S RECORD**

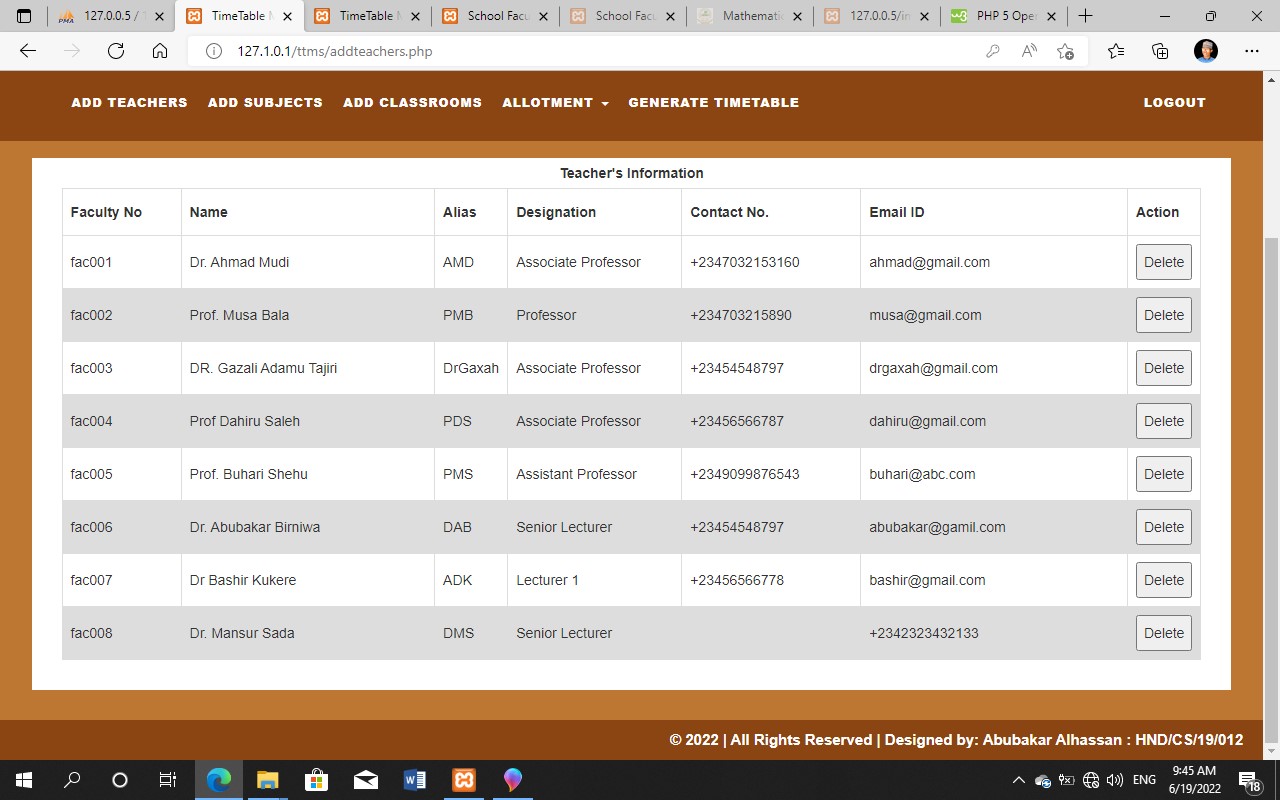


Figure 8: View Teacher’s Record

**VIEW ALL SUBJECT’S RECORDS**

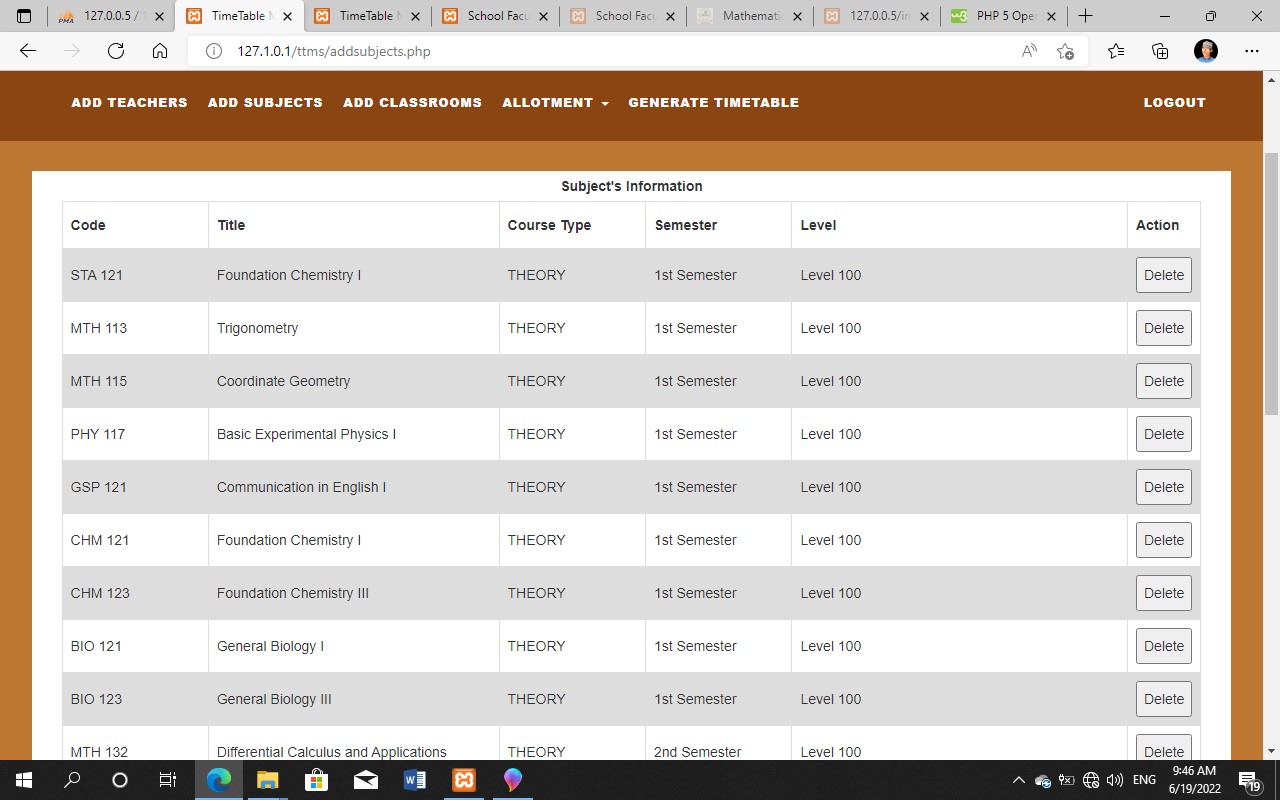


Figure 9: View All Subject’s Records

**VIEW CLASSROOMS**

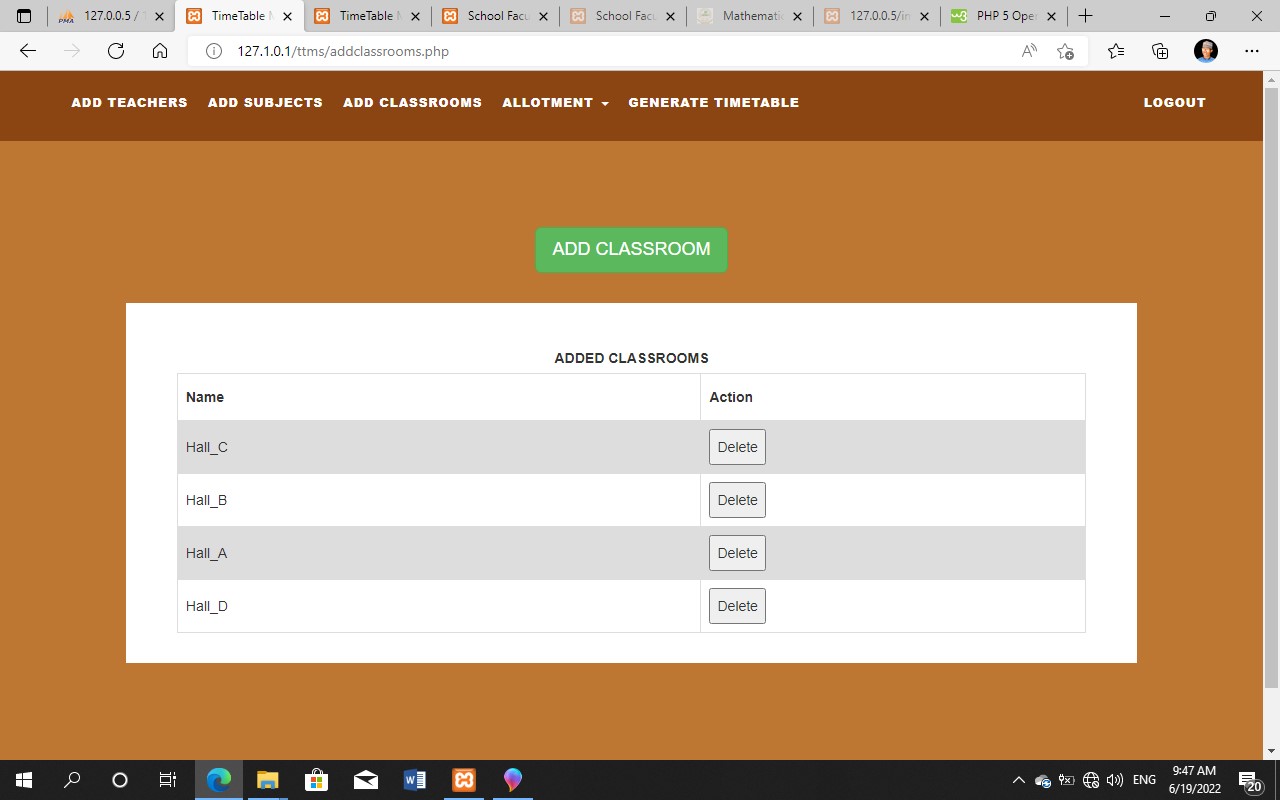


Figure 10: View Classroom

**GENERATE TIMETABLE PAGE**

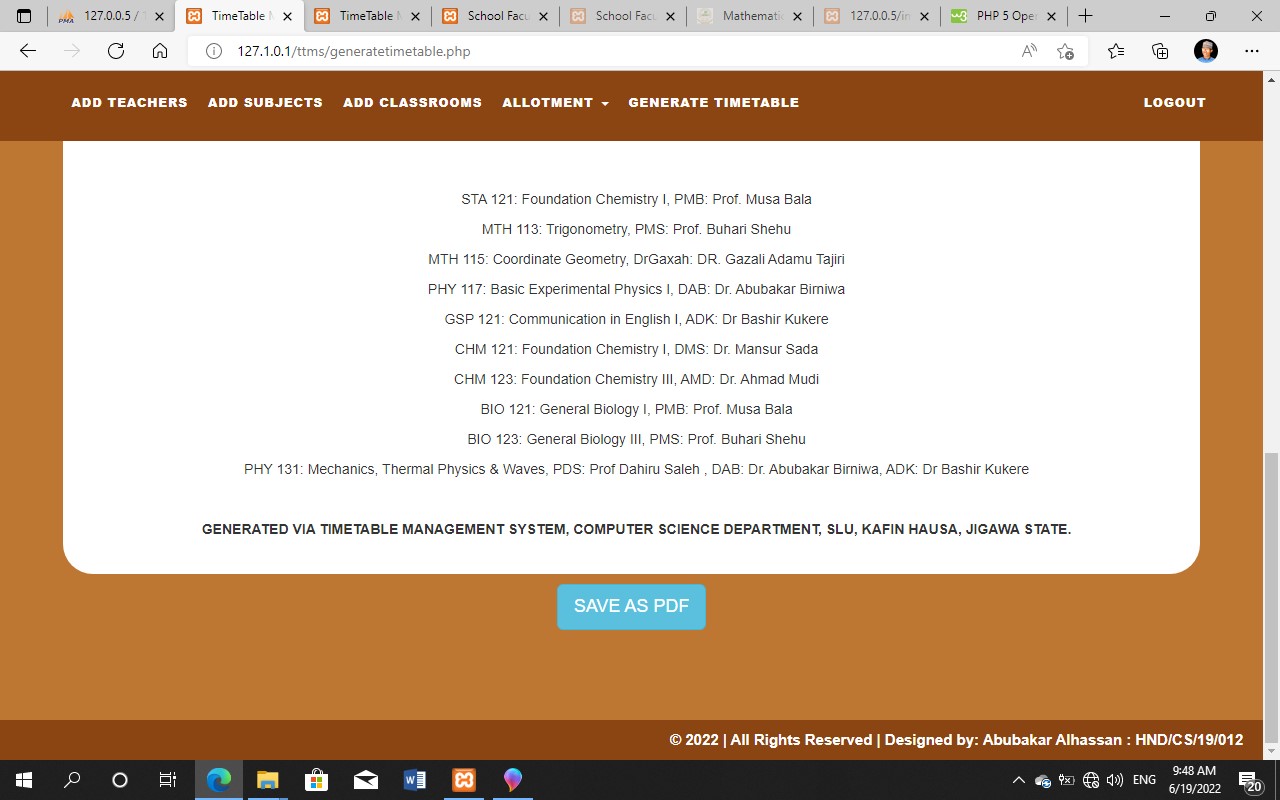
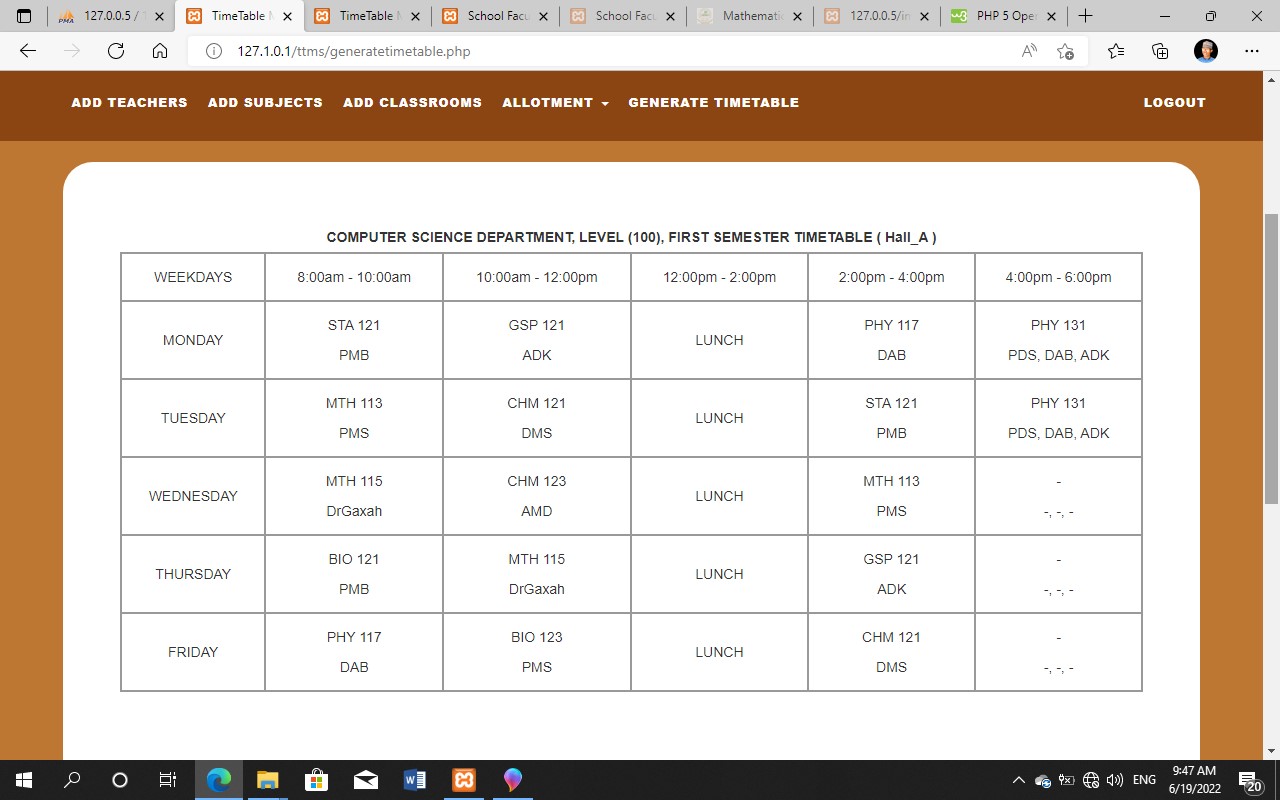


Figure 11: Generate Timetable page

**VIEW TEACHER’S TIMETABLE**

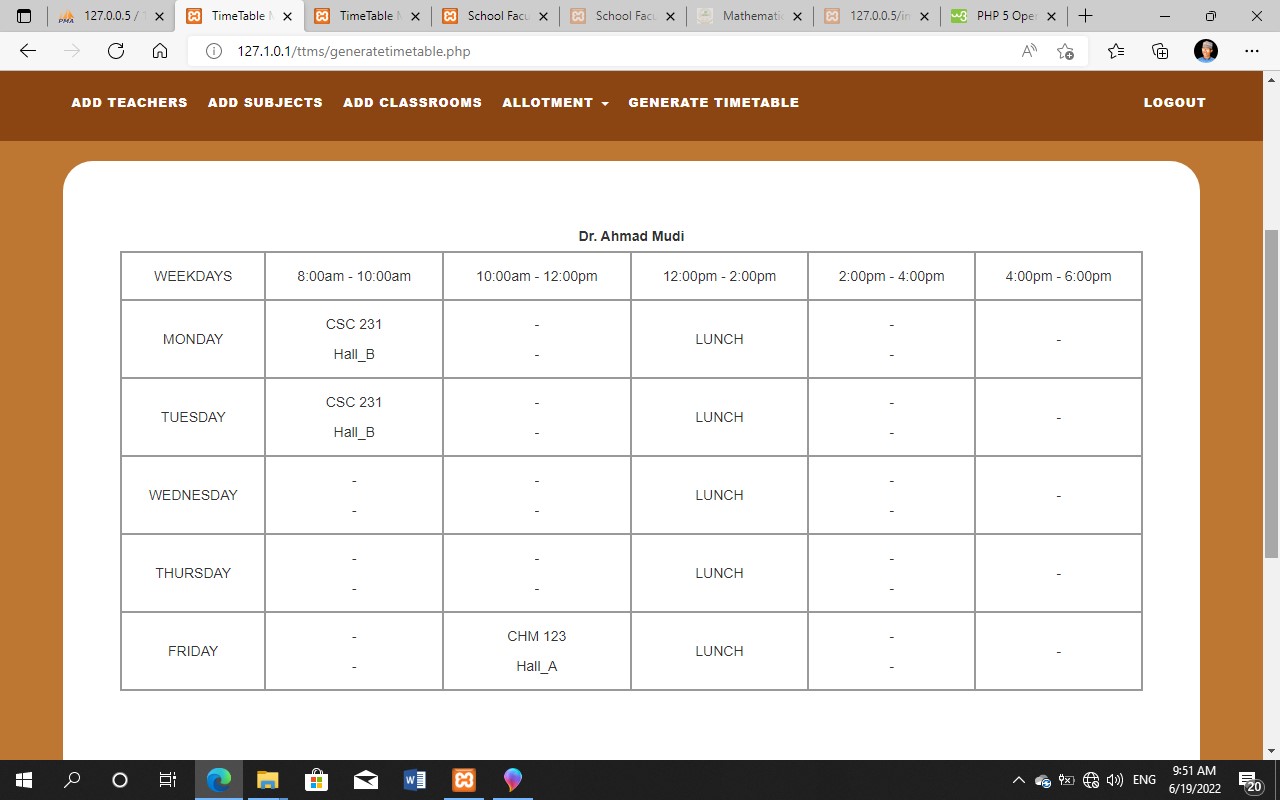


Figure 12: View Teacher’s Timetable

## **SYSTEM PACKAGING AND DEPLOYMENT**

After testing the system and it meets up with user’s requirement, the next phase is to implement the system, the process of implementation, involve the installation of the new system and also to enlightening peoples how to use the system, that is carrying about workshop or seminar in the organization were the system would be operated, so that they can be able to manage the data, records and information’s of the students.

The system can be implemented in the department using **DIRECT** or **PARALLEL METHOD**.

I. The **DIRECT METHOD**: Is when the previous method of Classroom Learning (that is the manual method) is automatically replaced with the new system, (which is Online Learning Management System)

II. The **PARALLEL METHOD**: Is when the new designed system replaces the previous system gradually, where the previous system is still in use until the new is practically understood and perfect to use.

# **CHAPTER FIVE**

**TESTING AND IMPLEMENTATION**

## **SYSTEM TESTING**

Testing is the process of executing the programs with the intention of finding out errors. During testing, the program to be tested is executed with a set of test cases and the output of the programs for the test case is evaluated to determine if the program is performing as it is expected to be.

As the software is created and added to the developing system, testing is performed to ensure that it is working correctly and efficiently. Testing is generally focused on two areas, internal efficiency and external effectiveness. The goal of external effectiveness testing is to verify that the software is functioning according to system design, and that it is performing all the required functions. The goal of internal testing is to make sure that the computer code is efficient, standardized, and well documented. Testing can be a labor-intensive process due to its iterative nature.

**Test Plan:**

We divided the testing procedure into these levels: -

* Unit Testing.
* Integration Testing.
* Validation Testing.
* System Testing.

These different levels of testing attempt to detect different types of faults. The relations of faults introduced in different levels of testing are as shown below

Requirements

System testing

Clients’ Needs

Design

Code

Validation Testing

Integration Testing

Unit Testing

Figure 13: Levels of testing

## **CHANGE OVER PLAN**

In a parallel changeover, the new system runs simultaneously with the old for a given period of time. For the changeover plan, the system cannot be change completely at once from the existing system to the new system, the old system must be remaining because of the purpose of some problems that can be happen at any time like corruption of the software or something else, new system running and old system kept for some problems against the new system.

## **IMPLEMENTATION ENVIRONMENT**

This phase is initiated after the system has been tested and accepted by the user. In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives established during the planning phase. Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes.

In system implementation, user training is crucial for minimizing resistance to change and giving the new system a chance to prove its worth. Training aids, such as user friendly, manuals, a data dictionary, job performance aids that communicate information about the new system and “help” screens provide the user with a good start on the new system.

There are three types of implementation:

* 1. Implementation of a computer system to replace a manual system: The problems encounter is converting files, training users, creating accurate files and verifying printouts for integrity.
  2. Implementing of a new computer system to replace an existing one: This is usually a difficult conversion. If not properly planned there can be many problems. Some large computer system has taken as long as a year to convert.
  3. Implementation of a modified application to replace an existing one using the same computer; This type of conversion is relatively easy to handle, provided there are no major changes in the files.

Conversion means changing from one system to another. The objective is to put the tested system into operation while holding costs, risks, and personnel irritation to a minimum.

This phase continues until the system is operating in production in accordance with the defined

# **CHAPTER SIX**

**SUMMARY, CONCLUSION AND RECOMMENDATION**

## **PROJECT SUMMARY**

This study was carried out as is to design and develop an Online Timetable Scheduling System that will reduce the intense manual effort being put into creating and developing university timetables. It presents user-friendly features that will familiarize the Staffs and the Student on the application. In addition, it will purvey supplement material in their front desk operation course. The researchers used the HTML as a front end and MYSQL as the back end. It also provided offline security to protect privacy and financial information of students.

## **PROJECT OUTCOMES AND REQUIREMENT FULFILMENTS**

The project outcome was to design an Online Timetable Scheduling System which would be used in the proper record keeping, automatic timetable generation and effective usage of Learning Environment.

The requirement fulfilled by the project is as follow:

* Provision of a proper and better record keeping.
* Automatic Timetable Generation.
* Lecturers, Courses and Lecture-room registration.
* Easy means of individual lecturers and Courses record searching.

Creation of good communication means between administrators and students.

## **RECOMMENDATIONS**

In furtherance of this work, the following are recommended:

* 1. The University should implement the system to improve efficiency.
  2. The computer must be well maintained otherwise it will breakdown frequently this losing useful information.
  3. Management of the School must ensure that the Admin is properly trained on how to use the computer.
  4. The timetable system developed as the outcome of this project should be made open to avid students of computing who can collaborate and improve on the techniques and ideas inherent in this project.
  5. Further works on developing a timetabling system should be based on this research work so as to utilize the incremental model of software development.
  6. A collaborative model of timetabling system which utilizes a computer network can also be built which entails different departments and entities allocating courses and constraints concurrently while the system threads and reports clashes.

## **CONCLUSION**

Timetabling problem being the hard combinatorial problem that it is would take more than just the application of only one principle. The timetabling problem may only be solved when the constraints and allocations are clearly defined and simplified thoroughly and more than one principle is applied to it i.e. a hybrid solution (a combination of different solution techniques).

Lastly, the new system which this work entails was developed such that it is quite flexible and easily manipulated, modifiable and portable. It does not permit unauthorized use of system; users can make used of the system even in the absence of computer expert or a programmer because all necessary document concerning the system were provided.

# **REFERENCES**

1. Ahmed, K., & Keedwell, E. (2017). "A Hidden Markov Model Approach to the Problem of Heuristic Selection in Hyper Heuristics with a Case Study in High School Timetabling Problems," Massachusetts Institute of Technology, 2017.
2. A. Chowdhary, P. Kakde, S. Dhoke, S. Ingle, R. Rushiya, and D. Gawande, “TIMETABLE GENERATION SYSTEM,” Int. J. Comput. Sci. Mob. Comput., vol. 3, no. 2, 2014.
3. Babaei, H., Karimpour, J., & Hadidi, A. (2015). A survey of approaches for university course timetabling problem. Computers & Industrial Engineering, 86, 43-59. http://dx.doi.org/10.1016/j.cie.2014.11.010.
4. Bucco, G. B., Bornia Poulsen, C. J., & Bandeira, D. L. (2017). Desenvolvimento de um modelo de programação linear para o problema da construção de grades horárias em universidades. Gestão & Produção, 24(1), 40-49. http://dx.doi.org/10.1590/0104-530x2133-15
5. E. K. Burke, M. Hyde, G. Kendall, G. Ochoa, E. Ozcan, and J. Woodward. Exploring hyperheuristic methodologies with genetic programming. In C. Mumford and L. Jain, editors, Collaborative Computational Intelligence. Springer, 2009.
6. Esraa, A., & Ghada, E., K. (2016). "A Utilization-based Genetic Algorithm for Solving the University Timetabling Problem (UGA)," Alexandria Engineering Journal, vol. 55, no. 2, pp. 1395-1409, 2016.
7. H. V. Yamazaki, & J. Pertoft. "Scalability of a Genetic Algorithm that Solves a University Course Scheduling Problem Inspired by KTH," 2014.
8. Hamed, B., Jaber, K. & Amin, H. (2019). "Generating an optimal timetabling for multi departments common lecturers using hybrid fuzzy and clustering algorithms," Methodologies and Application, vol. 23, p. 4735–4747.
9. Jardim, A. M., Semaan, G. S., & Penna, P. H. V. (2016). Uma heurística para o problema de programação de horários: um estudo de caso. In Anais do XLVIII Simpósio Brasileiro de Pesquisa Operacional (SBPO) Vitória: SOBRAPO.
10. Kheiri, A., & Ed, K. (2017) "A Hidden Markov Model Approach to the Problem of Heuristic Selection in Hyper Heuristics with a Case Study in High School Timetabling Problems," Evolutionary Computation, vol. 25, no. 3, pp. 473-501.
11. Lewis, R., & Thompson, J. (2015). Analyzing the effects of solution space connectivity with an effective metaheuristic for the course timetabling problem. European Journal of Operational Research, 240(3), 637-648. http://dx.doi.org/10.1016/j.ejor.2014.07.041
12. N. Pillay, “A survey of school timetabling research,” Ann. Oper. Res., vol. 218, no. 1, pp. 261 293, Jul. 2014.
13. Nelishia, P., & Ender, O. (2019). "Automated generation of constructive ordering heuristics for educational timetabling," Annals of Operations Research, vol. 275, no. 1, p. 181–208.
14. Obit, J.H., Developing novel meta-heuristic, hyper-heuristic and cooperative search for course timetabling problems, PhD Thesis, School of Computer Science, University of Nottingham, Nottingham, U.K., 2010.
15. R. Fredrikson, & D. Jonas, "A Comparative Study between a Simulated Annealing and a Genetic Algorithm for Solving a University Timetabling Problem," KTH Royal Institute of Technology, Degree Project in School of Computer Science and Communication, 2016.
16. Suliadi, S., Ahmad, I., Maselan, A., & Siti, R. (2018). "A Heuristics Approach for Classroom Scheduling Using Genetic Algorithm Technique," Journal of Physics: Conference Series, vol. 995.
17. Tan, L., Joe, O., Yu-Beng, L. and Jetol, B. (2018). "Implementation of Constraint Programming and Simulated.
18. Veenstra, M., & Vis, I. F. A. (2016). School timetabling problem under disturbances. Computers & Industrial Engineering, 95, 175-186. http://dx.doi.org/10.1016/j.cie.2016.02.011.

# **APPENDIX A: SOURCE CODES**

**INDEX/ADMIN LOGIN PAGE**

<?php

if (isset($\_GET['generated']) && $\_GET['generated'] == "false") {

unset($\_GET['generated']);

echo '<script>alert("Timetable not generated yet!!");</script>';}

?>

<!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'/>

<style type="text/css">

body {

background-color: #ffffff;}

</style>

</head>

<body>

<div class="navbar navbar-inverse navbar-fixed-top " id="menu">

<div class="container">

<div align="center">

<img src="assets/img/slulogo.png" style="height:70px; width:70px; border-radius:50%;">

<h4 align="center"><b>TIME TABLE MANAGEMENT SYSTEM, COMPUTER SCIENCE DEPARTMENT, SULE LAMIDO UNIVERSITY, KAFIN HAUSA, JIGAWA STATE.</b></h4>

</div>

</div>

</div>

<div id="myCarousel" class="carousel slide" data-ride="carousel">

<!-- Indicators -->

<ol class="carousel-indicators" style="margin-bottom: 160px">

<li data-target="#myCarousel" data-slide-to="0" class="active"></li>

<li data-target="#myCarousel" data-slide-to="1"></li>

<li data-target="#myCarousel" data-slide-to="2"></li>

<li data-target="#myCarousel" data-slide-to="3"></li>

</ol>

<!-- Wrapper for slides -->

<div class="carousel-inner" role="listbox">

<div class="item active">

<img src="assets/img/image1.jpg" alt="Chania">

</div>

<div class="item">

<img src="assets/img/image2.jpeg" alt="Chania">

</div>

<div class="item">

<img src="assets/img/image3.jpeg" alt="Flower">

</div>

<div class="item">

<img src="assets/img/image4.jpeg" alt="Flower">

</div>

</div>

</div>

<script type="text/javascript">

function genpdf() {

var doc = new jsPDF();

doc.addHTML(document.getElementById('TT'), function () {

doc.save('demo timetable.pdf'); });

window.alert("Downloaded!"); }

</script>

<div align="center" STYLE="margin-top: 30px">

<button data-scroll-reveal="enter from the bottom after 0.2s"

id="teacherLoginBtn" class="btn btn-info btn-lg">TEACHER LOGIN

</button>

<button data-scroll-reveal="enter from the bottom after 0.2s"

id="adminLoginBtn" class="btn btn-success btn-lg">ADMIN LOGIN

</button>

</div> <br>

<div align="center">

<form data-scroll-reveal="enter from the bottom after 0.2s" action="studentvalidation.php" method="post">

<select id="select\_semester" name="select\_semester" class="list-group-item">

<option selected disabled>Select Semester</option>

<option value="1">Level 100, First Semester</option>

<option value="2">Level 100, Second Semester</option>

<option value="3">Level 200, First Semester</option>

<option value="4">Level 200, Second Semester</option>

<option value="5">Level 300, First Semester</option>

<option value="6">Level 300, Second Semester</option>

<option value="7">Level 400, First Semester</option>

<option value="8">Level 400, Second Semester</option>

</select>

<button type="submit" class="btn btn-info btn-lg" style="margin-top: 10px">Download</button>

</form>

<br>

</div>

<!-- The Modal -->

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

<!-- Modal content -->

<div class="modal-content">

<div class="modal-header">

<span class="close">&times</span>

<h2 id="popupHead">Modal Header</h2>

</div>

<div class="modal-body" id="LoginType">

<!--Admin Login Form-->

<div style="display:none" id="adminForm">

<form action="adminFormvalidation.php" method="POST">

<div class="form-group">

<label for="adminname">Username</label>

<input type="text" class="form-control" id="adminname" name="UN" placeholder="Username ...">

</div>

<div class="form-group">

<label for="password">Password</label>

<input type="password" class="form-control" id="password" name="PASS"

placeholder="Password ...">

</div>

<div align="right">

<input type="submit" class="btn btn-default" name="LOGIN" value="LOGIN">

</div>

</form>

</div>

</div>

<!--Faculty Login Form-->

<div class="modal-body" id="LoginType">

<div style="display:none" id="facultyForm">

<form action="facultyformvalidation.php" method="POST" style="overflow: hidden">

<div class="form-group">

<label for="facultyno">Faculty No.</label>

<input type="text" class="form-control" id="facultyno" name="FN" placeholder="Faculty No. ...">

</div>

<div align="right">

<button type="submit" class="btn btn-default" name="LOGIN">LOGIN</button>

</div>

</form>

</div>

</div>

</div>

</div>

<script>

// Get the modal

var modal = document.getElementById('myModal');

// Get the button that opens the modal

var teacherLoginBtn = document.getElementById("teacherLoginBtn");

var adminLoginBtn = document.getElementById("adminLoginBtn");

var heading = document.getElementById("popupHead");

var facultyForm = document.getElementById("facultyForm");

var adminForm = document.getElementById("adminForm");

// Get the <span> element that closes the modal

var span = document.getElementsByClassName("close")[0];

// When the user clicks the button, open the modal

adminLoginBtn.onclick = function () {

modal.style.display = "block";

heading.innerHTML = "Admin Login";

adminForm.style.display = "block";

facultyForm.style.display = "none"; }

teacherLoginBtn.onclick = function () {

modal.style.display = "block";

heading.innerHTML = "Faculty Login";

facultyForm.style.display = "block";

adminForm.style.display = "none"; }

// When the user clicks on <span> (x), close the modal

span.onclick = function () {

modal.style.display = "none";

adminForm.style.display = "none";

facultyForm.style.display = "none"; }

// When the user clicks anywhere outside of the modal, close it

window.onclick = function (event) {

if (event.target == modal) {

modal.style.display = "none"; } }

</script>

<!--HOME SECTION END-->

<!--HOME SECTION TAG LINE END-->

<div id="faculty-sec">

<div class="container set-pad">

<div class="row text-center">

<div class="col-lg-8 col-lg-offset-2 col-md-8 col-sm-8 col-md-offset-2 col-sm-offset-2">

<h1 data-scroll-reveal="enter from the bottom after 0.1s" class="header-line">OUR UNIVERSITY </h1>

</div>

</div>

<!--/.HEADER LINE END-->

<div class="row">

<div class="col-lg-4 col-md-4 col-sm-4" data-scroll-reveal="enter from the bottom after 0.4s">

<div class="faculty-div">

<img src="assets/img/faculty/DVCacad.jpg" class="img-rounded"/>

<h3 align="center">Prof. UMAR SAJE</h3>

<hr/>

<h4 align="center">Deputy Vice Chancelor<br/>Academics, SLU</h4>

</div>

</div>

<div class="col-lg-4 col-md-4 col-sm-4" data-scroll-reveal="enter from the bottom after 0.5s">

<div class="faculty-div">

<img src="assets/img/faculty/VC.jpg" class="img-rounded"/>

<h3 align="center">PROF. LAWAN SANI TAURA</h3>

<hr/>

<h4 align="center">Vice Chancellor<br/> SLU, Kafin Hausa.</h4>

</div>

</div>

<div class="col-lg-4 col-md-4 col-sm-4" data-scroll-reveal="enter from the bottom after 0.6s">

<div class="faculty-div">

<img src="assets/img/faculty/DVCadmin.jpg" class="img-rounded"/>

<h3 align="center">DR. UMAR ALTI</h3>

<hr/>

<h4 align="center">Deputy Vice-Chancellor<br/>Administration, SLU</h4>

</div>

</div>

</div>

</div>

</div>

<div class="container">

<div class="row set-row-pad">

<div class="col-lg-4 col-md-4 col-sm-4 col-lg-offset-1 col-md-offset-1 col-sm-offset-1 "

data-scroll-reveal="enter from the bottom after 0.4s">

<h2><strong>Our Location </strong></h2> <hr/>

<div>

<h4>Sule Lamido University, P.M.B. 048, </h4>

<h4>Kafin Hausa, Jigawa State.</h4>

<h4><strong>Website:</strong> www.slu.edu.ng</h4>

<h4><strong>Email: </strong>admin@slu.edu.ng</h4>

</div>

</div>

<div class="col-lg-4 col-md-4 col-sm-4 col-lg-offset-1 col-md-offset-1 col-sm-offset-1"

data-scroll-reveal="enter from the bottom after 0.4s">

<h2><strong>Social Conectivity </strong></h2> <hr/>

<div>

<a href="#"> <img src="assets/img/Social/facebook.png" alt=""/> </a>

<a href="#"> <img src="assets/img/Social/google-plus.png" alt=""/></a>

<a href="#"> <img src="assets/img/Social/twitter.png" alt=""/></a>

</div>

</div>

</div>

</div>

<!-- CONTACT SECTION END-->

<?php include 'footer.php'; ?>

<!-- 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>

</div>

</body>

</html>

**TIMETABLE GENERATION PAGE**

<?php

// Start the session

session\_start();

if (isset($\_GET['success'])) {

echo "<script type='text/javascript'>alert('Time Table Generated');</script>";} ?>

<!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>

<script type="text/javascript" src="assets/jsPDF/dist/jspdf.min.js"></script>

<script type="text/javascript" src="assets/js/html2canvas.js"></script>

<!-- 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 TEACHERS</a></li>

<li><a href="addsubjects.php">ADD SUBJECTS</a></li>

<li><a href="addclassrooms.php">ADD CLASSROOMS</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>CLASSROOMS</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>

<!--Algorithm Implementation-->

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

<!-- Modal content -->

<div class="modal-content">

<div class="modal-header">

<span class="close">&times</span>

<h2 id="popupHead">Assign Substitute</h2>

</div>

<div class="modal-body" id="AssignSubstitute">

<!--Admin Login Form-->

<div style="display:block" id="assignSubstituteForm">

<form method="post" action="assignSubstituteFormValidation.php">

<div class="form-group">

<label for="substitute">Substitute</label>

<select class="form-control" id="substitute" name="SB">

</select>

<input type="hidden" id="cell\_number" class="btn btn-default" name="CN">

</div>

<div align="right" class="form-group">

<input type="submit" id="submit" class="btn btn-default" name="ADD" value="CHECK">

</div>

</form>

</div>

</div>

</div>

</div>

<script>

var assignsubstitueForm = document.getElementById("assignSubstitueForm");

// Get the <span> element that closes the modal

var modal = document.getElementById('myModal');

var span = document.getElementsByClassName("close")[0];

span.onclick = function () {

modal.style.display = "none";

assignsubstitueForm.style.display = "none"; }

// When the user clicks anywhere outside of the modal, close it

window.onclick = function (event) {

if (event.target == modal) {

modal.style.display = "none";

assignsubstitueForm.style.display = "none"; } }

</script>

<form action="algo.php" method="post">

<div align="center" style="margin-top: 50px">

<button type="submit"

id="generatebutton" class="btn btn-success btn-lg">GENERATE

</button>

</div>

</form>

<form action="generatetimetable.php" method="post">

<div align="center" style="margin-top: 30px">

<select name="select\_teacher" class="list-group-item">

<option selected disabled>Select Teacher</option>

<?php

$q = mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"),

"SELECT \* FROM teachers ");

while ($row = mysqli\_fetch\_assoc($q)) {

echo " \"<option value=\"{$row['faculty\_number']}\">{$row['name']}</option>\""; } ?>

</select>

<button type="submit" id="viewteacher" class="btn btn-success btn-lg" style="margin-top: 5px">VIEW TIMETABLE

</button>

</div>

</form>

<form action="generatetimetable.php" method="post">

<div align="center" style="margin-top: 20px">

<select name="select\_semester" class="list-group-item">

<option selected disabled>Select Semester</option>

<option value="1">Level 100, First Semester</option>

<option value="2">Level 100, Second Semester</option>

<option value="3">Level 200, First Semester</option>

<option value="4">Level 200, Second Semester</option>

<option value="5">Level 300, First Semester</option>

<option value="6">Level 300, Second Semester</option>

<option value="7">Level 400, First Semester</option>

<option value="8">Level 400, Second Semester</option>

</select>

<button type="submit" id="viewsemester" class="btn btn-success btn-lg" style="margin-top: 5px">VIEW TIMETABLE

</button>

</div>

</form>

<script>

var index = -1;

function Substitute() {

var table = document.getElementById("timetable");

var cells = table.getElementsByTagName("td");

// window.alert(cells[3].innerHTML.toString());

for (i = 0; i < cells.length; i++) {

if (i % 8 == 6 || i % 8 == 7 || parseInt(i / 8) == 0 || i % 8 == 0) {

continue; }

var currentCell = cells[i];

//var b = currentRow.getElementsByTagName("td")[0];

var createSubstituteHandler =

function (cell, i) {

return function () {

document.getElementById('cell\_number').value = i;

index = i;

var xmlhttp = new XMLHttpRequest();

xmlhttp.onreadystatechange = function () {

if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {

var modal = document.getElementById('myModal');

modal.style.display = "block";

document.getElementById("substitute").innerHTML = this.responseText; } };

xmlhttp.open("GET", "getcellindex.php?i=" + i, false);

xmlhttp.send() } };

currentCell.onclick = createSubstituteHandler(currentCell, i); } }

</script>

<div> <br>

<style>

table {

margin-top: 5px;

font-family: arial, sans-serif;

border-collapse: collapse;

width: 90%; }

td, th {

border: 2px solid #999999;

text-align: left;

padding: 8px; }

tr:nth-child(even) {

background-color: #ffffff; }

tr:nth-child(odd) {

background-color: #ffffff; }

</style>

<center>

<div id="TT" style="background-color: #FFFFFF; width:90%; border-radius:30px;">

<table border="2" cellspacing="3" align="center" id="timetable">

<caption><strong><br><br>

<?php

if (isset($\_POST['select\_semester'])) {

if (($\_POST['select\_semester']) == 1) {

$sem = "LEVEL (100), FIRST SEMESTER TIMETABLE ";

}elseif (($\_POST['select\_semester']) == 2) {

$sem = "LEVEL (100), SECOND SEMESTER TIMETABLE ";

}elseif (($\_POST['select\_semester']) == 3) {

$sem = "LEVEL (200), FIRST SEMESTER TIMETABLE ";

}elseif (($\_POST['select\_semester']) == 4) {

$sem = "LEVEL (200), SECOND SEMESTER TIMETABLE ";

}elseif (($\_POST['select\_semester']) == 5) {

$sem = "LEVEL (300), FIRST SEMESTER TIMETABLE ";

}elseif (($\_POST['select\_semester']) == 6) {

$sem = "LEVEL (300), SECOND SEMESTER TIMETABLE ";

}elseif (($\_POST['select\_semester']) == 7) {

$sem = "LEVEL (400), FIRST SEMESTER TIMETABLE ";

}else{

$sem = "lEVEL (400), SECOND SEMESTER TIMETABLE ";

}

echo "COMPUTER SCIENCE DEPARTMENT, ".$sem . " ";

$year = (int)($\_POST['select\_semester'] / 2) + $\_POST['select\_semester'] % 2;

$r = mysqli\_fetch\_assoc(mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"), "SELECT \* from classrooms

WHERE status = '$year'"));

echo " ( " . $r['name'], " ) ";

} else if (isset($\_POST['select\_teacher'])) {

$id = $\_POST['select\_teacher'];

$r = mysqli\_fetch\_assoc(mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"), "SELECT \* from teachers

WHERE faculty\_number = '$id'"));

echo $r['name'];

} else if (isset($\_GET['display'])) {

$id = $\_GET['display'];

$r = mysqli\_fetch\_assoc(mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"), "SELECT \* from teachers

WHERE faculty\_number = '$id'"));

echo $r['name']; } ?>

</strong></caption>

<tr>

<td style="text-align:center">WEEKDAYS</td>

<td style="text-align:center">8:00am - 10:00am</td>

<td style="text-align:center">10:00am - 12:00pm</td>

<td style="text-align:center">12:00pm - 2:00pm</td>

<td style="text-align:center">2:00pm - 4:00pm</td>

<td style="text-align:center">4:00pm - 6:00pm</td>

</tr>

<tr>

<?php

$table = null;

if (isset($\_POST['select\_semester'])) {

$table = " semester" . $\_POST['select\_semester'] . " ";

} else if (isset($\_POST['select\_teacher'])) {

$table = " " . $\_POST['select\_teacher'] . " ";

} else if (isset($\_GET['display'])) {

$table = " " . $\_GET['display'] . " ";

} else

echo '</table>';

if (isset($\_POST['select\_semester']) || isset($\_POST['select\_teacher']) || isset($\_GET['display'])) {

$q = mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"),

"SELECT \* FROM" . $table);

$qq = mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"),

"SELECT \* FROM subjects");

$days = array('MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY');

$i = -1;

$str = "<br>";

$tid = "";

if (isset($\_POST['select\_semester'])) {

while ($r = mysqli\_fetch\_assoc($qq)) {

if ($r['isAlloted'] == 1 && $r['semester'] == $\_POST['select\_semester']) {

$str .= $r['subject\_code'] . ": " . $r['subject\_name'] . ", ";

if (isset($r['allotedto'])) {

$id = $r['allotedto'];

$qqq = mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"),

"SELECT \* FROM teachers WHERE faculty\_number = '$id'");

$rr = mysqli\_fetch\_assoc($qqq);

$str .= " " . $rr['alias'] . ": " . $rr['name'] . " ";

}

if ($r['course\_type'] !== "LAB") {

$str .= "<br>";

continue;

} else {

$str .= ", ";

}

if (isset($r['allotedto2'])) {

$id = $r['allotedto2'];

$qqq = mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"),

"SELECT \* FROM teachers WHERE faculty\_number = '$id'");

$rr = mysqli\_fetch\_assoc($qqq);

$str .= " " . $rr['alias'] . ": " . $rr['name'] . ", ";

}

if (isset($r['allotedto3'])) {

$id = $r['allotedto3'];

$qqq = mysqli\_query(mysqli\_connect("localhost", "root", "", "ttms"),

"SELECT \* FROM teachers WHERE faculty\_number = '$id'");

$rr = mysqli\_fetch\_assoc($qqq);

$str .= " " . $rr['alias'] . ": " . $rr['name'] . "<br>";

}

}

}

} else if (isset($\_POST['select\_teacher']) || isset($\_GET['display'])) {

if (isset($\_POST['select\_teacher'])) {

$tid = $\_POST['select\_teacher'];

} else if (isset($\_GET['display'])) {

$tid = $\_GET['display'];

$tid = strtoupper($tid);

}

while ($r = mysqli\_fetch\_assoc($qq)) {

if ($r['isAlloted'] == 1 && $r['allotedto'] == $tid) {

$str .= $r['subject\_code'] . ": " . $r['subject\_name'] . " <br>";

} else if ($r['isAlloted'] == 1 && isset($r['allotedto2']) && $r['allotedto2'] == $tid) {

$str .= $r['subject\_code'] . ": " . $r['subject\_name'] . " <br>";

} else if ($r['isAlloted'] == 1 && isset($r['allotedto3']) && $r['allotedto3'] == $tid) {

$str .= $r['subject\_code'] . ": " . $r['subject\_name'] . " <br>";

} } }

while ($row = mysqli\_fetch\_assoc($q)) {

$i++;

echo "

<tr><td style=\"text-align:center\">$days[$i]</td>

<td style=\"text-align:center\">{$row['period1']}</td>

<td style=\"text-align:center\">{$row['period2']}</td>

<td style=\"text-align:center\">LUNCH</td>

<td style=\"text-align:center\">{$row['period3']}</td>

<td style=\"text-align:center\">{$row['period4']}</td>

</tr>\n";

}

echo '</table>';

$sign = "GENERATED VIA TIMETABLE MANAGEMENT SYSTEM, COMPUTER SCIENCE DEPARTMENT, SLU, KAFIN HAUSA, JIGAWA STATE.";

echo "<div style='margin-left: 10px' align='center'>" . "<br>" . $str . "<br></div>" .

"<div style='margin-left: 10px' align='center'>" . "<strong>" . $sign . "<br></strong></div>";

}

if (isset($\_POST['select\_teacher'])) {

echo "<script>Substitute();</script>";

$\_SESSION['shown\_id'] = $\_POST['select\_teacher'];

}

if (isset($\_GET['display'])) {

echo "<script>Substitute();</script>";

$\_SESSION['shown\_id'] = $\_GET['display']; } ?>

<br>

</div>

</div>

</center>

<script type="text/javascript">

function gendf() {

var doc = new jsPDF();

doc.addHTML(document.getElementById('TT'), function () {

doc.save('<?php

if (isset($\_POST["select\_semester"])) {

echo "ttms semester " . $\_POST["select\_semester"];

} else if (isset($\_POST["select\_teacher"])) {

echo "ttms " . $\_POST["select\_teacher"];

} else if (isset($\_GET["display"])) {

echo "ttms " . $\_GET["display"];

}

?>' + '.pdf');

alert("Downloaded!"); }); }

</script>

<div align="center" style="margin-top: 10px">

<button id="saveaspdf" class="btn btn-info btn-lg" onclick="gendf()">SAVE AS PDF</button>

</div>

<!--HOME SECTION END--><br><br>

<?php include 'footer.php'; ?>

<!-- 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>

# **APPENDIX B: SNAP SHOT**

