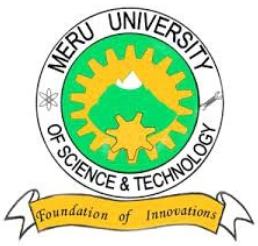


MERU UNIVERSITY OF SCIENCE AND TECHNOLOGY

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

COMPUTER LABORATORY SCHEDULING SYSTEM

STUDENT NAME	REGISTRATION NUMBER
1. Evans Loktari Chaun	CT201/100133/19
2. Ashley Mary Apondi	CT201/100106/19
3. Dinah Anyango	CT201/100139/19
4. Simon Ngaruiya	CT201/100125/19
5. Wambua Ndavi	CT201/100107/19
6. Joshua T Wanaina	CT201/100084/19

February, 2022.

DECLARATION

We declare that this research is our original work except for the quotations and citations that have been there and acknowledged before, and has not been previously submitted for any other degree at Meru University of Science and Technology.

STUDENT NAME	REGISTRATION NUMBER	SIGNATURE	DATE
1. Evans Loktari Chaun	CT201/100133/19	……..	…….
2. Ashley Mary Apondi	CT201/100106/19	……..	…….
3. Dinah Anyango	CT201/100139/19	……..	…….
4. Simon Ngaruiya	CT201/100125/19	……..	…….
5. Wambua Ndavi	CT201/100107/19	……..	…….
6. Joshua T Wanaina	CT201/100084/19	……..	…….

ACKNOWLEDGMENTS

We wish to thank Dr. Mary Mwadulo for taking us through this interesting unit and giving us her dedicated time. We would also like to thank Mr. Irungu for his unconditional support and his mentorship through our project work.

Contents

Figures

Tables

CHAPTER ONE

INTRODUCTION

1.1 Background of study

The timetable of an institution is one of its most essential features. That is why all institutions, large and small, adhere to a schedule. It is a schedule that divides the school day into different times and assigns different subjects to each session. A time chart lists the activities that must be completed in a school on a specific day. As a result, it is critical in institutions.

Many industries require scheduling in order to run efficiently and effectively. It can be found in public transportation, hospitals, and educational institutions, among other places. There are many factors to consider in educational settings, particularly at higher education institutions, which makes scheduling a difficult task[1]. Some of the frequent considerations in the case of educational institutions are the availability of lecturers, the number of classes and courses, and budgeting.

Planning a schedule manually is effort and time-consuming compared to automated scheduling. Several requirements must be met during scheduling. The following are some common limits to consider: only one instructor can teach one class at a time, a room can only hold one class at a time, and students should not have more than one class every session. Hard and soft constraints are frequently used to categorize these limitations[2].

The hard restrictions cannot be broken, while the soft constraints can be broken with the drawback of less optimum scheduling. Due to the significant amount of time and money spent manually scheduling, various attempts to automate this activity using computers have been made. Some lecturers, such as part-timers or visiting lecturers, can contribute to these limits. The timetable tries to satisfy other constraints which are not critical e.g. soft constraints that should be satisfied but may be violated. Lecturers can be assigned more than one course and furthermore, some constraints can be taken into consideration to improve the quality of education e.g. No lectures should be scheduled in the last slot any day or a certain day should have an activity slot for students, and a slot where all university academics can meet. For those slots, no sessions should be scheduled[3]

This, of course, is dependent on the number and complexity of constraints. There is no generic method that will provide the best answer for any schedule problem due to the difficulty of the problem and the numerous limitations. Several optimization algorithms have been implemented to solve this problem. Local search strategies like Tabu search and simulated annealing [1] to evolutionary algorithms like particle swarm optimization and genetic algorithms are examples of metaheuristic algorithms. The complexity of the problem, according to Ademiluyi and Ukaoha[2], explains why so many distinct methods have been implemented. Almost every school has its own set of limits and prerequisites that must be met. In most cases, evolutionary algorithms outperform local search algorithms in the early phases of the process. As a result, numerous hybrid algorithms have been developed, which combine evolutionary algorithms to restrict the search space with local search algorithms to discover the optimal answer in that space.

Developing an automated system to address both simple and complex timetabling problems is critical to providing an effective timetable that influences the lives of students and professors in university education. It produces far better timetables than those created manually, using a basic

technique that causes timetabling collision difficulties, and it dramatically reduces the administrative workload at the university.

In general, a university course timetabling challenge entails determining the exact amount of time allotted within a certain time frame. For example, during a week, a number of events (courses-lectures) are scheduled and assigned to a number of resources (lecturers-rooms) in order to satisfy a number of restrictions. According to Chen [1], on "Developing two heuristic algorithms with metaheuristic algorithms to improve solutions of optimization problems with soft and hard constraints", as cited by [2] has defined a demanding and challenging administrative task of any academic institution, the process which varies in difficulty according to the problem size and demanding constraints depending on the academic institution. The placement of a collection of subjects into a classroom across a limited number of time periods to avoid conflicts of interests between two subjects or lecturers is known as timetabling. A good scheduling strategy that leads to optimization is necessary to ensure that all timetables for students and lecturers are produced. The key challenge in creating university timetables is to supply lecturers and lecture activities by matching all lectures with authorized time, as well as the person in charge. Room availability, time slots, and the number of students per program are all needed information for the course schedule. For example, room availability information can be tied to room capacity for specific events. It is frequently used in the context of university timetables to refer to the creation of a schedule (with time slots) using a computer system while taking into account a large number of constraints [4]. The proposed approach is designed to create a course schedule. Any Nigerian university with identical limits and characteristics can use the system. The heuristics technique is used to assign courses, period, and room resources in phases to arrive at a realistic solution that meets the needs of the users (lecturers and students).

With respect to the above considerations, the process of coming up with a timetable for any learning institution is considered as a time consuming and interactive process that involves a lot of compromising on both the human and technical resources offered by the university. The technical resources would be the class rooms and the human resources would be the lecturers and the students. As a result, university course timetabling difficulties are combinatorial problems involving the scheduling of a number of courses in a given number of classrooms and time periods.

In order to come up with a feasible timetable, it is important to compromise between all lecturers' time preferences and all also keeping in mind availability of classrooms and students, number of students and curricula[5].

However, completing our system's algorithm construction phase was the most important factor. Because none of the logics listed above suited our needs. Finally, we used the Genetic Algorithm to lead our research.

ChayaAndradi and SamindaPremaratne used genetic algorithm in their research[6]. They've given a model, a set of methods, and a computer program for the schedule problem, with a focus on a real-world application (University of Moratuwa). They also compared their method to several forms of simulated annealing. Finally, they find that Genetic Algorithms produced better timetables than simulated annealing but slightly worse timetables than tabu search in their studies. A benefit of the Genetic Algorithm is that it allows the user to choose from a variety of possible timelines. Finally, they discovered that their method is a useful generalization of Genetic Algorithms and that it can be used to solve other highly constrained combinatorial optimization problems. Moreira could, in a different scenario, use Genetic Algorithms to solve the challenge of

establishing exam timetables. According to ChayaAndradi and SamindaPremaratne, Genetic Algorithms were employed in a unique way since algorithm performance was greatly improved by modifying basic genetic operators, which prevent the emergence of new conflicts among individuals.

As a result, there is a need to give a soft computing solution to the challenges connected with the manual approach to constructing timetables; a genetic algorithm approach to build an optimal timetable with constraint fulfilment and no clashes. One of the soft computing strategies for solving optimization problems is genetic algorithms (GA). GA is a type of adaptive heuristic search that is based on Darwin's theory of natural selection and genetics. It refers to the clever application of a random search inside a defined search space to find a solution to a problem[7].

1.2 Motivation of study

The following factors motivated this research: bias in computer laboratory allocations, idleness of computer laboratories in the afternoon hours and lack of proper student allocation programs to computer laboratories in the Meru University of Science and Technology timetabling system.

1.3 Problem Statement

Currently, the Meru University of Science and Technology timetable system is set up to generally allocate lecture rooms and laboratories to specific programs at specific times. This leads to lecturers being assigned two programs at the same time or more than one program being assigned to one lecture room at the same time, and as a result, a lot of time is wasted when students shift to look for an available lecture room. Moreover, an issue arises when practical units are not assigned to the laboratories, whereas theory units are assigned to the computer laboratories.

The proposed Computer Laboratory Scheduling System is primarily concerned with allocations of computer laboratories, by optimizing the algorithm used in today's timetable systems to generate a feasible timetable data with fewer or no clashes. This is achieved by assigning exactly one program to a laboratory at a specific time to a specific lecturer. It also aims at making sure that the practical units are given priority over the theory units in the allocation of computer laboratories.

1.4 Research objectives

1. To enable the Timetabler allocate a computer laboratory.
2. To enable the Timetabler send a notification to a lecturer and students when the timetable is out.
3. To enable a lecturer to book a free slot.
4. To enable the Timetabler reserve a booked slot.
5. To enable Timetabler to send an email notification to the lecturer that the slot is reserved.
6. To enable the System generate a weekly report on the day the laboratory is optimally used.

1.5 Significance of study

This will produce conflict-free timetable with less commitment and greater performance, thus meeting all the challenging constraints concerned with far less duration. Lecturers, students and other users can work and view the timetable on various devices simultaneously due to the portable nature of the system. The simple web-based interface will facilitate ease of use.

1.6 Scope of the study

The project is mainly focused on resolving the issue of s computer laboratory allocation and therefore will not address scheduling of exams and allocation of other rooms. It will be assessed at Meru University of Science and Technology in the school of computing and will also consider different criteria based on the form of lecturers, rooms and laboratories affiliated with the relevant department.

1.7 Assumptions of study

In conducting the study the following assumptions were made. It is assumed that: Data on the available computer laboratories and their capacity is available, there is no preferences (No biasness) in allocation of the computer laboratory and the computer hardware is able to compute the allocation algorithm to be used.

1.8 Limitations of the study

There is no full time dedication to the project as there are classes to attend to and there are no funding to the project as there is no one to fund the whole project alongside the group member

CHAPTER TWO

LITERATURE REVIEW

2.0 Overview

This chapter will provide a general understanding of the literature review of a computer laboratory scheduling system. The chapter expounds on the functionalities, components, classification Constraints of a scheduling system. The chapter further unveils the approaches used in developing timetable systems and the discussion on related studies. In the long run, this chapter will conclude by summarizing the chapter.

2.1 Functionalities of the computer scheduling system.

A timetabling system of a school is a web, mobile, or desktop application that consists of an administrative module, a student module, and a lecturer module. The system performs a variety of functions, including the authentication of all authorized users[8]. A Timetabler can schedule and upload a timetable using the system. The scheduled timetable includes a sub-menu for scheduling lectures and programs. Additionally, Timetablers has the option of allocating a room to different programs.

The lecturer is granted access to the Timetabling system to view the most recent schedule updates. The lecturer can also reserve a room and see which slots have been reserved. A

lecturer can also view the course code, lecture day, lecture duration, lecture venue, and lecturer (s) facilitating the course[5].

Accounts must be created by students. Students must use a login to gain access to the timetabled scheduling system. Students' functional navigation options include the most recent schedule updates after successful authentication[9].

2.2 Components of computer scheduling system.

2.2.0 Timetabler module

The Timetable scheduling system as a system consists of the authentication that provides the interface for authenticating the user of the system[5]. Upon successful authentication, the dashboard page loads the Timetabler schedule module by default which consists of schedule timetable, upload timetable. The schedule timetable provides the sub-menu for course-wise scheduling of lectures and lecture rooms[9].

Additionally, the functional navigation option available to timetabler includes allocating a computer laboratory to a class, and send notification to class representative when a computer laboratory slot is reserved.

2.2.1 Lecturer Module

The lecturer is given the privilege to logon to the timetabling schedule system and view the latest schedule updates[10]. The lecturer can also view the slots which have been reserved. In addition, a lecturer can also view the course code, day of lecture, duration of lecture, venue of lecture and also the lecturer(s) facilitating the course.

2.2.2 Students Module

The students are required to create accounts. Students are to have authentication into the Computer laboratory scheduling system through logon[5]. Upon successful authentication, the functional navigation option available to students includes viewing of the latest schedule updates.

2.3 Classification of computer laboratory scheduling system.

There are quite a wide number of differences in the literature based on the nature of organizations and the kind of restrictions or constraints that are inherent in the timetabling problems. Problems of timetabling are divided into three key groups according to Ademiluyi[2]. A University timetable is a weekly calendar for all University students, prohibiting two classes from being attended at the same time by lecturers, and vice versa. Course timetabling on the other hand is a weekly calendar with all lectures for a variety of university classes. Examination timetabling is the scheduling of a series of university courses for examinations, preventing the duplication of examinations of courses and extending the examinations as far as practicable to the students[11].

2.4 Constraint Types in Automated Scheduling

2.4.1 Hard Constraints

Hard constraints must be satisfied completely so that the generated solutions become possible and without conflict and violation is not permitted. According to

- There is only one course presentation session for each particular course every day. To consider the features of class over the presented course[12].

- Each student and lecturer is present only in one class at the same time.

- More than one course which has been scheduled to the same timeslot cannot be assigned to the same classroom.
- Lecturers must be available at times when their courses have been scheduled.
- Some courses require constant and specified number of sessions every week
- Each course that has been prescheduled has a mark and the scheduling of those courses must be scheduled at a particular time (following the priority)[12].

2.4.2 Soft Constraints

In soft constraints which are related to the objective function and they must be satisfied as much as possible, but it is not necessary that soft constraints are being satisfied as hard constraints[12].

- The lecturer considers it a time priority to teach.
- Empty spaces must be eliminated within classrooms.
- A subject (part of a course) must not be taught for more than three successive hours. Attempt to uniform distribution of courses among timeslots.
- One or a group of students must not have only one timeslot for one class at a day.

2.5 Approaches to timetabling

2.5.1 Web based Application

A web-based system is an application that is accessed via HTTP. The term web-based is usually used to describe applications that run in a web browser[2]It can, though, also be used to describe applications that have a very small component of the solution loaded on the client PC. The host server for a web-based system could be a local server, or it could be accessed via the internet[5].

2.5.2 Android based

This class timetable for Android has been produced. This application will assume a fundamental part for organizations or universities[6] who need to advise their instructors of a unique time which is developed by either use Kotlin or Java using Android Studio IDE

2.5.3 Windows.net

Is an open source application for university and school course timetabling. It is written in C# and runs under.NET framework [14]on Windows (and will likely run under Mono on Linux or Mac OS X)

2.6 Related studies

According to Ademiluyi, O., Ukaoha, K.C., Ndunagu, J., and Osang, F.B,[2] the researchers developed a soft-computing Based course timetabling system which able to generate a timetable and it also saved on physical and mental stress that is usually undergone during manual drafting of the timetable. However, the system was not able to extend the input and output sets of the system. The system was not also implemented as a web based application.

According to A.M. Hambali, Y.A Olasupo and M. Dalhatu[4], the researchers developed an automated university lecture timetable using heuristic approach as a desktop application that was able to generate a zero clash timetable. However, the above system was not accessible to all students and lecturers within the institution and they suggested that it can still be implemented as online web-based application.

According to Stephanie Gaku[5], the researcher developed a web-enabled agent-Based university timetabling assistant that helped in the automated generation of feasible timetables for the university. However the system was not able to incorporate an emailing trigger whereby the timetables

could be emailed to the users with respect to their associative. This led to the timetable not to circulate to everybody.

According to ChayaAndradi, and semindaPremaratne[6], the researchers developed a medium scaled university timetable management system that was able to generate a near optimal timetables using the principles of genetic algorithm that was easily understood, it reduced the paper work and it was an automated system that was helpful to authorities of the faculty. However, the timetabling system was not fit for the medium scale universities since it was not implemented as a framework.

According to Nur lqtiyani llham, E.H. Mat Saat, N.H Abdul Rahman, Farah Yasmin Abdul Rahman, Nurhani Kasuan[15], the researchers developed an auto-generate scheduling system that was able to auto-generate timetable using AI-based expert system with the user customization option. However, the system was not able to have an automatic generation for the entire faculties available. The system still did not have an online access into the automatic timetable generation.

According to Ahmad Muklason, Nisa D Angresti, Putri C Bwananesia, VichaAztrhanty Supoyo, and Jasmi Hidayatu YT[16], the researchers developed an automated examination timetabling optimization using a greedy-late acceptance-hyperheuristic algorithm that was able to produce a feasible solution for all datasets and it was to generate an algorithm that was generating a better timetable than a manual generation. However, the system was not able to come up with an algorithm that can be tested over the benchmark datasets.

According to Ghazi Alnowaini and Amjad Abdullah Aljomai[13], the researchers developed a genetic algorithm for solving university timetabling problems using dynamic chromosomes that were able to reduce and prevent conflicts in times and halls. It also increased in the number of the

generation that reduced the number of conflicts for the courses. However, the system did not include the idea to all faculties in the institution and did not allow for the learners to request reservation of free times for the lecture halls.

According to Anuja Chowdhary , Priyanka Kakde , Shruti Dhoke , Sonali Ingle , Rupal Rushiya, and Dinesh Gawande[14], the researchers developed a timetable generation system that was able to generate a separate timetable for the individual class, faculty, and laboratories. It was also able to avoid slot clashes occurring. However, the system was not able to generate a master timetable for the department and the entire college.

According to Susana Limanto, Njoto Benarkah, and Tyrza Adelia[17], the researchers developed a Thesis examination timetabling using a genetic algorithm that was able to generate a thesis exam timetable proposal automatically. However, the system was not able to be implemented as a mobile application so that it may suit the current conditions, since almost everyone has devices.

According to Dian Kusumawardani, Ahmad Muklason, and Vicha Azthanty Supoyo[18], the researchers developed an examination timetabling automation and optimization using greedy-simulated annealing hyper-heuristic algorithm that was able to solve a real-world examination timetabling problem effectively. However, the system was not able to come up with a greater number of heuristics.

According to Paul Godwin Daniel, Bala Modi, and Maruf Alimi[19], the researchers developed a paperless master timetable scheduling system that was able to completely eliminate the manual process of generating timetables by making use of collision avoidance technique and process in scheduling an automated timetable system. However, the system was not able to have a real-time

generation of timetables with content-based analysis and reports that were to be generated through an embedded management system.

2.6 Summary

In summary, scheduling involves the arrangement, coordination, and planning of the utilization of resources to achieve an objective. School timetabling problems particularly address the optimal utilization of classrooms and timeslots in sequencing lectures between instructors and students, according to Ademiluyi, Ukaoha, Ndunagu, and Osang[2],"timetabling, either course timetabling or examination timetabling is one of the major factors that influences the academic performance of any institutions". During the construction of a timetable, one of the challenges is to ensure no constraints of faculty, courses, classrooms, timeslots, or students are violated[13]. These types of resource-constrained schedule optimization problems have been long term studied with a huge volume of work in both operational research and the artificial intelligence field.

CHAPTER THREE

METHODOLOGY

3.0 Overview

This chapter delves into the study's methodological choices and research design process in depth. The methodological choice has primarily been guided by the ideological attitude and the research challenge. To add more to this, it explains the different types of research designs that are popular and why Experimental research design is the most preferred approach. The chapter also lays out the protocols for gathering, analyzing, and reporting data. It has explained widely about the population study and the different techniques used for sampling. Furthermore, the methods used to improve the studies' validity and reliability are explained in depth. Finally, the chapter discusses research procedures, such as study time, weighting, and integration decisions, as well as ethical considerations.

3.1 Research design

Research design is intended to provide an appropriate framework for study. It supports research methods and techniques chosen by a researcher. It is used to cut expenses, maintain a tight grip on the consistency of the outcomes, and create a solid foundation for the entire research. Furthermore, it is utilized to collect relevant data and techniques in order to assist the seamless scaling of various research operations, resulting in the maximum amount of information. A poor groundwork of research design displeases the entire project[20].

3.1.1 Types of Research Design

A researcher must be familiar with several research designs in order to determine which one should be used for the study. There are several sorts of research designs, each of which is described below.

3.1.1.0 Explanatory Research Design

The research design is dubbed explanatory when the goal of the investigation is to explore a new world that has never been investigated before. The study is primarily concerned with the causes or "why" component of an event. It does not include comparisons or change factors[10].

3.1.1.1 Descriptive Research

It's also known as statistical research, and it's used to describe real-world occurrences. It is used to identify and collect data on characteristics of a certain issue, such as a community, group, or people. What, who, where, how, and when are all questions that descriptive research addresses. It is employed in the analysis of the current situation[20].

3.1.1.2 Experimental Design

When a particular cause leads to the same consequence, the cause will progress to the effect, therefore the degree of association is significant. The main procedure is in charge of all aspects of the experiment. More measurements and groups are used in the experimental design for longer periods of time[21]. We employed experimental research designs in this study because they are one of the most powerful ways for determining the causal relationship between variables. These studies produce the best evidence possible about the impact of certain interventions. Because the study was conducted in a controlled atmosphere, there is a higher degree of purity in observation. Conditions

that would take years to occur spontaneously in a natural setting can be established in an experimental setting in a short amount of time (therefore very useful in genetic studies). The challenges of real-life situations and the researcher's personal problems were eliminated because the study was conducted in an experimental context. The environment in which research is conducted can frequently be carefully regulated. As a result, determining the genuine effect of the variable of interest on the outcome of interest got easier[20].

3.2 Population, Sample and Sampling

Population is the set or group of all units on which findings of the research are to be applied[22] while population of interest is the study's target population that intends to study or treat[9]. The population of interest in this research will comprise of 1500 registered students from the school of computing and informatics at Meru University of Science and Technology. The choice of the population of interest was based on the fact that it was convenient for researcher to gather data.

Sample is a reprehensive part of a population of research[22] while sample size is the number of completed responses or the number of participants in a sample in our survey receives.

Sampling is the process of selecting sample from population[23]. There are different sampling methods which includes the following:

Simple Random Sampling**:** In this method is that each member of the population has an equal chance of being selected.

3.2.1 Stratified Sampling

The whole population is divided into homogeneous strata or subgroups according a demographic factor (e.g. gender, age, religion, socio-economic level, education, or diagnosis etc.). Then, the researchers select draw a random sample from the different strata.

3.2.2 Cluster sampling

The smallest units into which a population can be divided are called the elements of the population, and groups of elements of the clusters.

3.2.3 Quota sampling

In quota sampling, the researcher selects people non-randomly according to some fixed quota.

3.2.4 Purposive Sampling

In purposive sampling, sampling is done with a purpose in mind. Purposive sampling can be very useful for situations where one needs to reach a targeted sample quickly and where sampling for proportionality is not the primary concern. In this study, the researchers will use stratified sampling[23].

In this experiment we used Stratified sampling where the whole population is divided into homogeneous strata or subgroups according a demographic factor[24]. It allows the researcher to obtain an effect size from each strata separately as if it was a different study and allows obtaining samples from under presented population. The reason for using stratified sampling was that all the students will be put into consideration by dividing them into groups corresponding to their year of study that will ensure all students were properly presented.

The sample was calculated using the formula:

$n=N/(1+N(e)^2)$

where: n is the sample size, N is the population size and e is the margin of error of 5% (0.05).

Based on the sampling technique and sampling formula applied, table 3.1 shows the population and the sample size. Table 1 below shows the population and sample size

TABLE 1 POPULATION AND SAMPLE SIZE

Year of study	Population size of registered students(N)	Sample size(n)
One	300	171
Two	500	222
Three	450	211
Four	250	153

3.3 Data collection, instrumentation and procedures

Data collection is a methodical process of gathering and analyzing specific information to proffer solutions to relevant questions and evaluate the results[20]. It focuses on finding out all there is to a particular subject matter. Data is collected to be further subjected to hypothesis testing which seeks to explain a phenomenon.

Data collection tools refer to the devices/instruments used to collect data, such as a paper questionnaire or computer-assisted interviewing system. Case Studies, Checklists, Interviews,

3.3.1 Interview

An interview is a face-to-face conversation between two individuals with the sole purpose of collecting relevant information to satisfy a research purpose[25].

3.3.2 Questionnaire

This is the process of collecting data through an instrument consisting of a series of questions and prompts to receive a response from individuals[25] it is administered to. Questionnaires are designed to collect data from a group.

3.3.3 Reporting

By definition, data reporting is the process of gathering and submitting data to be further subjected to analysis. The key aspect of data reporting is reporting accurate data because of inaccurate data reporting leads to uninformed decision making[25].

3.3.4 Observation

This is a data collection method by which information on a phenomenon is gathered through observation. The nature of the observation could be accomplished either as a complete observer, an observer as a participant, a participant as an observer, or as a complete participant. This method is a key base for formulating a hypothesis[20].

3.3.5 Combination research

This method of data collection encompasses the use of innovative methods to enhance participation to both individuals and groups[20].

This study used questionnaire to collect data. Questionnaire is the most appropriate data collection tool for this research because We used a questionnaire in our research in order to allow the participants to express their thoughts for the purposes of our system development. The questionnaire was distributed to three different personnel's that is students, lecturers and the timetable. By doing this, we were able gather the required information using less cost because was in form of online

forms[20]. The questionnaire saved on time because the forms were distributed online. The information gathered will help in this research to collect requirements from the user so as to develop a system to is going to suites their needs.

3.4 Development Tools and Materials

These are the software and hardware tools or system requirement and specification that was used in developing the Computer Laboratory Scheduling System. Such requirements helped in validating the system during testing[26]. The following are the specific software and hardware requirement. Used.

Software Requirements:

User Interface: HTML, CSS, JQUERY

Client-side Scripting: JavaScript, PHP Scripting

Programming Language: PHP.

IDE/Workbench/Tools: XAMPP.

Database: MySQL (MySQLite, Optional, Oracle 10g)

Serr Deployment: Apache/2.2.4, Tomcat Apache[26].

The PHP programming language was used to develop a user-friendly web interface to facilitate control and observation. The user interface languages (HTML, CSS and JQUERY) was used to design the front-end since enhanced ease in developing a simple, well-designed and intuitive user interface to use. The client-side scripting language (JavaScript) enabled users to select radiological technologist rosters by year, month, or number of radiological technologists, and be able users to scroll top or bottom of the page. MySQL was used to create the database that stored user data for easy retrieval when needed[26]. The server part of the software module (Services) was the intermediary between the database and client applications and enhance transparent access of clients to

the database, That ensured simultaneous operation of all schedule makers and synchronization and data exchange between clients[19].

Hardware Requirements:

Monitor: 17 inch LCD Screen (optional).

Processor: Pentium 3, 4, dual-core, Intel, Core i7.

Hard Disk: 500GB or 1, 2 or 4 Terabyte.

RAM: 4GB or more.

The monitor (17 inch LCD Screen) enabled display of timetable generated by the Computer Laboratory Scheduling system. The processor (Pentium 3, 4, dual-core, Intel, Core i7)[19] enhanced faster processing speed when implementing the Computer Laboratory Scheduling system. The hard disk (500GB or 1, 2 or 4 Terabyte) provided maximum storage of the data used and it provided maximum space for database development[19].

3.5 System development methodology

According to Roger S Pressman[27] in his book 'Software Engineering', a software development methodology/process, often known as the software development lifecycle, is a collection of activities, actions, and tasks that guide a development team through the successful completion of a project.

3.5.0 Waterfall model

This takes the fundamental process activities of specification, development, validation, and evolution and represents them as separate process phases. The waterfall model is a continuous software development model in which development is seen as flowing steadily downwards (like a waterfall) through the steps of requirements analysis, design, implementation, testing (validation), integration, and maintenance[27].

3.5.1 RAD Model

RAD or Rapid Application Development process is an adoption of the waterfall model; it targets developing software in a short period. The RAD model is based on the concept that a better system can be developed in lesser time by using focus groups to gather system requirements[27].

3.5.2 Spiral Model

The spiral model is a risk-driven process model. This SDLC model helps the group to adopt elements of one or more process models like a waterfall, incremental, waterfall, etc. The spiral technique is a combination of rapid prototyping and concurrency in design and development activities[28].

3.5.3 Incremental Model

The incremental model is not a separate model. It is necessarily a series of waterfall cycles. The requirements are divided into groups at the start of the project[27]. For each group, the SDLC model is followed to develop software. The SDLC process is repeated, with each release adding more functionality until all requirements are met. In this method, each cycle act as the maintenance

phase for the previous software release. Modification to the incremental model allows development cycles to overlap. After that subsequent cycle may begin before the previous cycle is complete.

3.5.4 Agile Model

Agile methodology is a practice which promotes continues interaction of development and testing during the SDLC process of any project. In the Agile method, the entire project is divided into small incremental builds. All of these builds are provided in iterations, and each iteration lasts from one to three weeks[28].

Any agile software phase is characterized in a manner that addresses several key assumptions about the bulk of software projects:

3.5.5 Iterative Model

It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces[27].

3.5.6 Big bang model

Big bang model is focusing on all types of resources in software development and coding, with no or very little planning. This model works best for small projects with smaller size development team which are working together. It is an ideal model where requirements are either unknown or final release date is not given[23].

3.5.7 Prototype Model

The prototyping model starts with the requirements gathering. The developer and the user meet and define the purpose of the software, identify the needs, etc. A '**quick design**' is then created. This design focuses on those aspects of the software that will be visible to the user[28]. It then leads to the development of a prototype. The customer then checks the prototype, and any modifications or changes that are needed are made to the prototype.

Looping takes place in this step, and better versions of the prototype are created. These are continuously shown to the user so that any new changes can be updated in the prototype. This process continue until the customer is satisfied with the system. Once a user is satisfied, the prototype is converted to the actual system with all considerations for quality and security.

The Incremental Model was used to develop the Laboratory Schedule System Because When compared to the waterfall paradigm, the quantity of analysis and documentation that needs to be redone is substantially less[27].Errors was easily identifiable during the development process[28].It was much easier to obtain client feedback on previously completed development work.Even if all of the functionality was not implemented, early delivery and deployment of valuable software to the customer is achievable[28].

3.6 Data processing and analysis

The data collected was checked and edited for clarity, legibility, relevance and adequacy[20]. This was going to involve checking for non-response and acceptance or rejection of answers, which was pre-coded by the researcher. The data was cleaned, tabulated frequencies and percentage it

was used to analyze the data This study splitted the received questionnaire into two groups including, group one(lecturers), group two(students) The purpose of tabulation is the process of summarizing data and displaying them in appropriate tables that further analysis is to be facilitated through. Then results of the analysis were presented in frequency tables and percentages

Once the data has been processed it was analyzed by Data analysis which was the process which is aimed at exploring and understanding data[29]. It helped the researcher get the insights of the behavior of raw information[29]. There are two statistical measures that a researcher can use to analyze data. i.e. descriptive and inferential statistic This study analyzed data using descriptive and inferential statistics. Descriptive statistic function such as standard deviation and frequency distribution used. Standard deviation was used to determine on average the number of students who registered by the end of the first day. While frequency distribution was used to determine fourth year students who have registered within the first two days of registration process.

Tabulation was in form of tables or frequency distribution table

3.7 Ethical consideration

The researcher understands that a University's Timetable contains some personal details of the lecturers, students and the timetable him/herself[12]. Therefore, the researcher practiced confidentiality of the highest form to make sure that their personal details like passwords during login are not visible to the outer world. In the password slot, there was a toggle to hide password, thus hiding their passwords and which created integrity.

3.8 Summary

This chapter describes the methodology used to collect and analyze the data required to carry out the study. The chapter begins with a discussion of the research design, followed by the population, sample and sampling where the size of the population and sample size are stated and the approach used in sample selection. Next the discussion centers on data collection, instrumentation and procedure stating the method used in data collection. Next topic that is put into consideration is development tools and materials which when broken down entails the software's and hardware's to be used. Also system development methodology is discussed on this chapter. Data processing and analysis was carried out stating the methods of analysis the data. Finally, ethical consideration was considered.

CHAPTER FOUR

SYSTEM ANALYSIS

4.0 Overview

Using the methodology mentioned in the previous chapter, we provide an approach to solve the problem as follows. This chapter will discuss the feasibility study, overall description of the current system, Using use case diagram, Class diagram, Data Flow Diagram, Activity Diagram, the Functional requirements Non-functional requirements and at the end there will be a summary of the aforementioned.

4.1 Feasibility study

Feasibility study is defined as a way of evaluating whether or not a project will be successful[30]. A feasibility study evaluates the practicality of your project plan in order to judge whether or not you're able to move forward with the project. Below are several types of feasibility study.

4.1.1 Technical Feasibility

Is the process of figuring out how you're going to produce your product or service to determine whether it's possible for your company[30].

4.1.2 Economic Feasibility

This assessment typically involves a cost/ benefits analysis of the project, which is going help organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated The proposed project will save on cost of acquiring a timetable ,the soft-wares being used are downloadable free from internet and the cost of a laptop is relatively low compared to purchasing a new timetabling system [30].

4.1.3 Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal re-quirements like zoning laws, data protection acts or social media laws[30].

4.1.4 Operational Feasibility

This assessment involves undertaking a study to analyze and determine whether—and how well—the organization's needs can be met by completing the project.

Scheduling Feasibility In scheduling feasibility, an organization estimates how much time the project will take to complete[30].

A short assessment of the current Timetabling system was carried out to determine whether developing a new system is a viable solution. An analysis on the worthiness to commit the resources to developing a Timetabling system was carried out.

To do these, a feasibility study was conducted. The study was aimed at how well can students use the rooms without any collision in terms of time that is more than one group having a session at the same time, lecturers having more than one group at the same time and students attending an already booked room.

During the feasibility study it was discovered that technically the institution has classes for students to be taught their courses.

Legal feasibility implies that a system should be able to meet all legal and contractual laws[30]. The current system meets all legal and contractual laws of a country.

Feasibility report is attached on appendix.

4.2 Overall description of the current system

The current timetabling systems that are being used, most of them are web based applications[5]. In that, there are three modules involved in the system, the Administrator, the Lecturer and the

Student. The Administrator logs into the system using his credentials and his functions are as follows; Manage programs, classes, faculty, department, sessions, timetable, lecturer and courses. The Lecturer and the Student are the direct users of the system. The Student and the Lecturer can log into the system, update their profile they can also change the password and log out. After doing so, they can both can view the timetable, add a comment and register into the system[4]. The overall view of timetabling systems is represented using use-case diagram, class diagram, sequence diagram, data flow diagram and activity diagram that are given in detail in section 4.2.1 to 4.2.5 below.

4.2.1 Use case diagram for timetable system

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform.[31]

The main actors of Timetable Management system in this use case Diagram are: Admin, Student and Lecturer, who perform different type of uses such as Timetable, manage class, view Timetable and Full Timetable Management Operations[5]. Major elements of the UML Use Case diagram of Timetabling Management system are shown in the figure 1

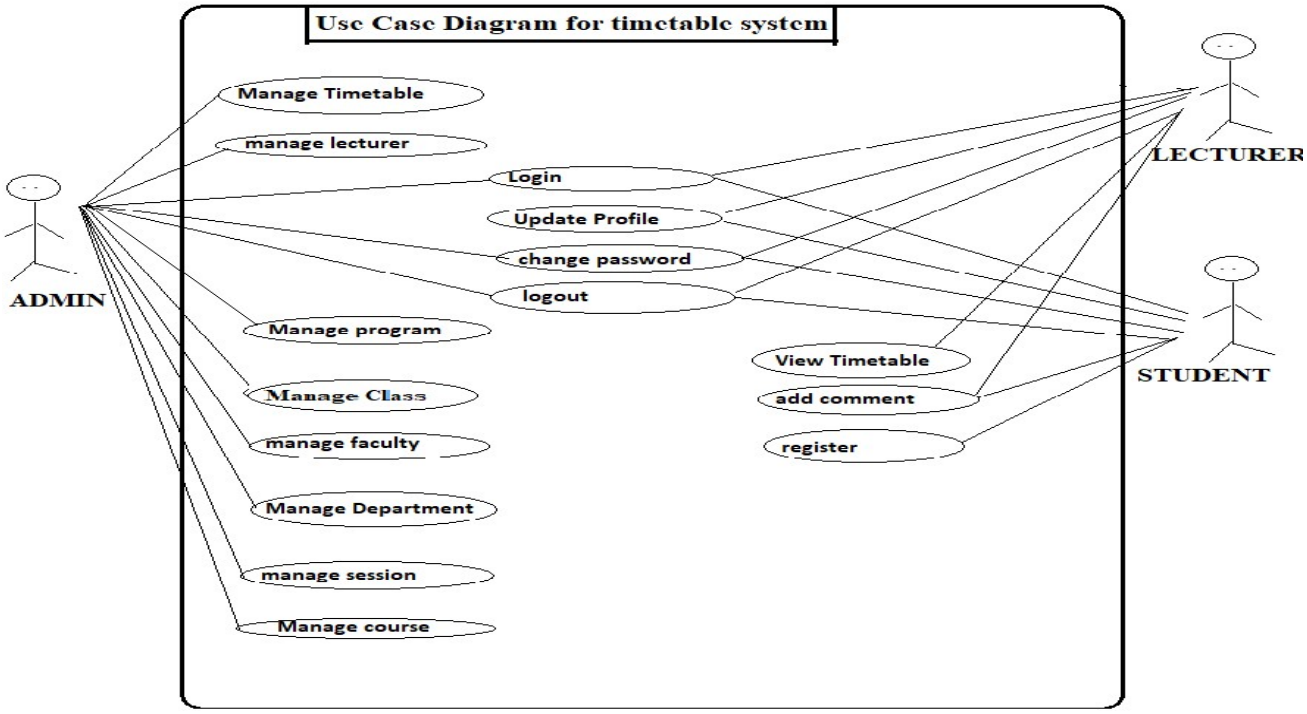


FIGURE 1 USE CASE DIAGRAM

4.2.2 Class Diagram

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML)[31]. The Unified Modeling Language (UML) is the de facto standard formalism for object-oriented modeling. A class defines the methods and variables in an object, which is an abstraction of entity in a program or the unit of code representing that entity[32]. Class diagrams are useful in all forms of object-oriented programming (OOP). The concept is several years old but has been refined as OOP modeling paradigms have evolved. In a class diagram, the classes are arranged in groups that share common characteristics. A class diagram resembles a flowchart in which classes are portrayed as boxes, each box having three rectangles inside. The top rectangle contains the name of the class; the middle rectangle contains the attributes of the class; the lower rectangle contains the methods, also called operations, of the class.

Lines, which may have arrows at one or both ends, connect the boxes. These lines define the relationships, also called associations, between the classes. Since class diagram has been developed widely as shown in figure 2 below

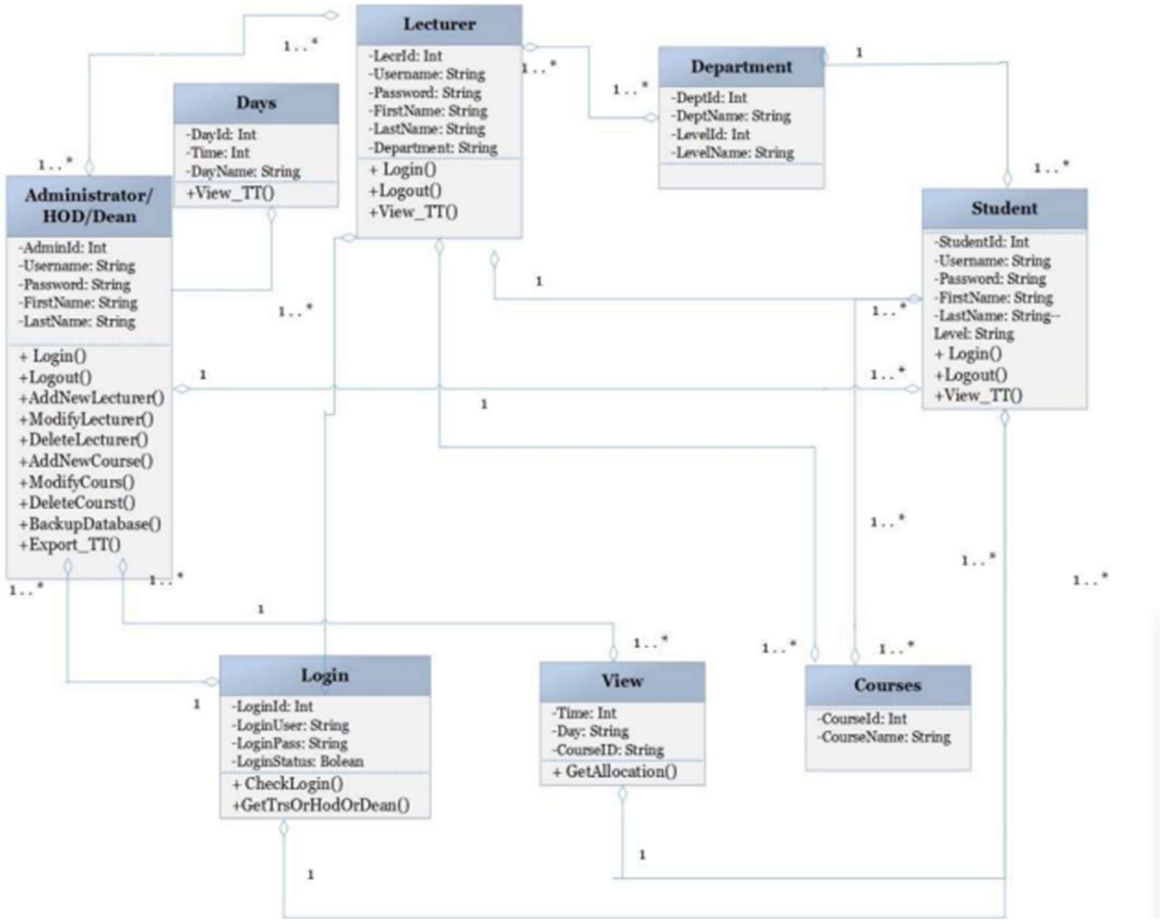


FIGURE 2 CLASS DIAGRAM

4.2.3 Sequence Diagrams for timetable system

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration[31]. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

This is the UML sequence diagram of Timetable Management System which shows the interaction between the objects of Student, Teacher, Timetable, Class.[5] The instance of class objects involved in this UML Sequence Diagram of Timetable Management System are as shown below in figure 3:

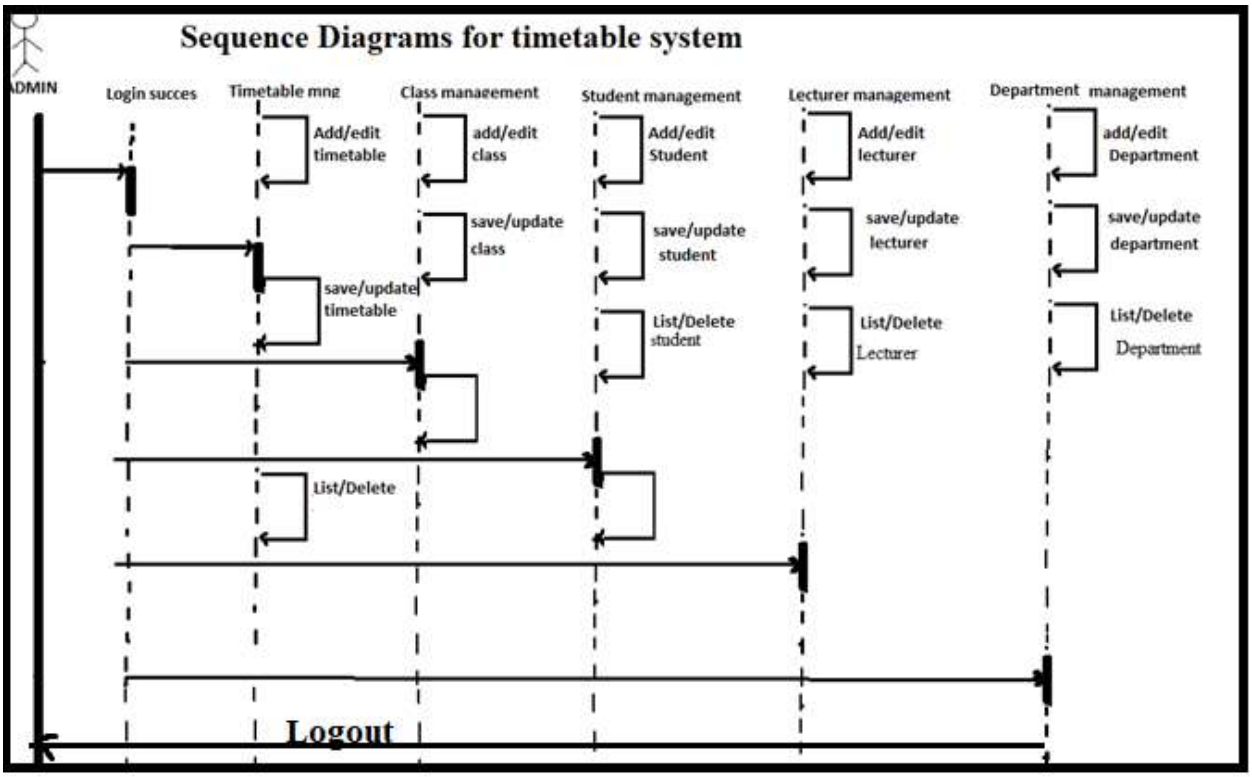


FIGURE 3 SEQUENCE DIAGRAMS

4.2.4 Data flow Diagram

Data flow Diagram is the graphical model that map out the flow of information for any process[31]. It shows the input, output, process and storage. It also focuses in the flow of information, where the data comes, where it goes and how it is stored. The importance of using Data Flow Diagram is that it provides more details about system components, describe all the system boundaries and easy to understand by audiences[32]. This is shown in figure 4 below

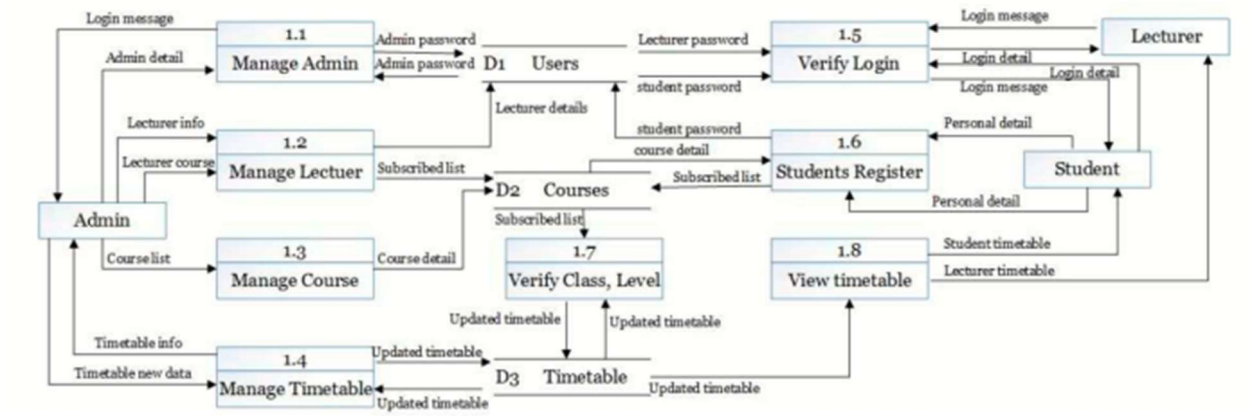


FIGURE 4 DATAFLOW DIAGRAM

4.2.5 Activity diagram

Unified Modeling Language like activity diagrams is purposely designed to model out both computational and organizational process[31]. The essence of Activity Diagram for Meru University timetabling system is to model the workflow of its Use Case to show paths within it. It will show the sequence of an activity from the other. The Activity Diagram will illustrate existing functionalities and how those functionalities coordinate with other pieces' functionalities of the system[32]. The general overview or description of the existing system is shown in figure 5

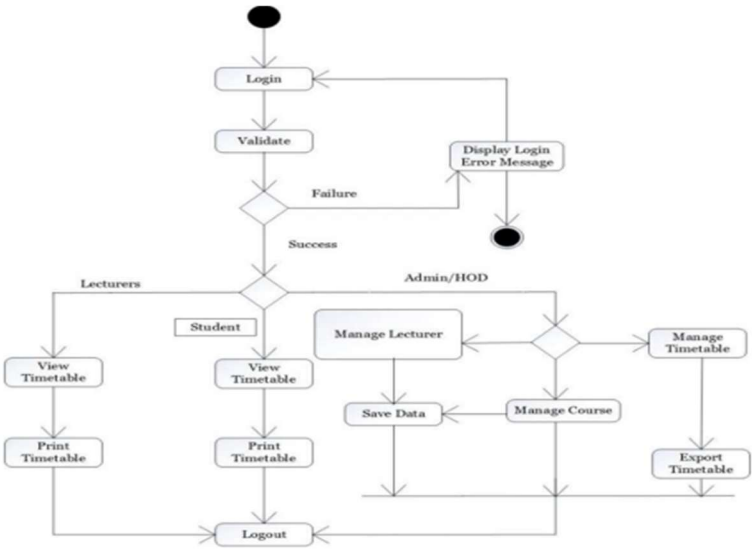


FIGURE 5 ACTIVITY DIAGRAM

4.3 Requirement gathering

Requirements gathering is the process of determining what your projects need to achieve and what needs to be created to make that happen[27].

4.3.1 Functional Requirements

Functional requirements describe the services of the system, how the system should react to particular inputs and how the system should behave in definite situations[28]

4.3.1.1 Login

The administrator, lecturer and student are presented with a login dialogue box requiring entry of a valid username and password. The system authenticates the administrator, lecturer and student and provides access to the system according to the privileges of that specific user[5].

4.3.1.2 Change Password

To change a password is implementing a new password. The system provides the capability of the administrator, lecturer and student to change account's password. Changing the password requires the user to enter the old password and new password. The user is then expected to confirm the new password. The system then implements the new password.

4.3.1.3 Register:

The student is provided with a platform to register a user. The student is expected to enter his or her details which include: username, email address, year of study and password. The system is expected to save the information and show the confirmation.

4.3.1.4 View Timetable

The lecturer and student are provided with a platform to view the timetable that has been generated by the administrator.

4.3.1.5 Update profile

The lecturer and student are also provided with a platform to update their details in the profile.

4.3.1.6 Add comment

The lecturer and student are provided with a platform to add some comments about the timetable that has been generated by the administrator.

4.3.1.7 Manage Faculty

The administrator is provided with a platform to add faculty by inserting the faculty name

4.3.1.8 Manage Program

The administrator is provided with a platform to add program by inserting the program name, faculty and the department in it

4.3.1.9 Manage Department

The administrator is provided with a platform to add department by inserting the program name, faculty and the department name.

4.3.1.10 lecturer

The administrator is further provided with a platform to add lecturer by inserting the program name[5], faculty, department name and lecturer name.

4.3.1.11 Course

The administrator is provided with a platform to add a course by inserting the course name, course code.

4.3.1.12 Room

This provides information about the venue capacity and its location this information is used by administrator to know which room to allocate a certain capacity.

4.3.1.13 Session

The administrator is further provided with a platform to add a session and allocate it.

4.3.1.14 Timetable

The administrator is further provided with a platform to do allocation of timetable

4.3.1.15 Report generation

The System can generate reports based on specific details provided by the system administrator. The system can generate three reports namely: monthly, weekly, daily.

4.3.2 Non-functional Requirements

Describes the attributes of the system. They are not related to functional aspect of software[28]. They are implicit or expected characteristics of software, which users make assumption of.

4.3.2.1 Security

The Timetabling system requires a student, lecturer to identify himself by using a login Username and Password. Therefore, any student or lecturer who needs to use the system must provide a correct username and password[5]. The metrics for measuring the security of the system is the number of attempts made to access the system with the correct or wrong login details.

4.3.2.2 Availability

The system facility is available to the administrator, lecturer and student for working hours. The functionality of student registering will be available for the first month of semester opening days.

4.3.2.3 Usability

It offers the ability to use a particular product including navigation where users can easily navigate its user interface in addition quality of performance where the features of a system are functioning well based on what developer predicted.

4.3.2.4 Localization

It has features that matches the geographical location of the users including aspects such as Languages, currencies and Time zones which a true representation of the current environment

4.3.2.5 Compatibility

It typically functions well when other application is running on a device it allows people who have different operating system to use the same application

4.4 Summary

This chapter focuses on the approaches to analyze the system. Firstly, the chapter will explain feasibility study and give its importance on the system analysis. Overall description of Meru university timetable schedule system is being explained with the use of case diagram, class diagram, sequence diagram, data flow diagram and activity diagram where introduction and importance is given in each of the diagram. Lastly, the chapter discussed about requirement gathering which is either functional or non-functional requirements where functional requirements describes the services of the system, how the system should react to a particular inputs and how the system should behave in definite situations while non-functional requirements describes the attributes of the system.

CHAPTER FIVE

SYSTEM DESIGN

5.0 Overview

This chapter will provide a general understanding of the system design of a computer laboratory scheduling system. The chapter expounds on the logical design, physical design and database design for the computer laboratory scheduling system. On the physical design for computer laboratory scheduling system outlines how the system looks like by presenting the interfaces available in the system. In the long run, this chapter will conclude by summarizing whatever have been discussed in the whole chapter.

5.1 Logical design for computer laboratory scheduling system.

Entity Relationship Diagram shows that the real world consists of a collection of entities, the relationships between them and the attributes that describe them[33]. Its importance is that provides mechanism for quickly and easily modelling data structures required by a software system.

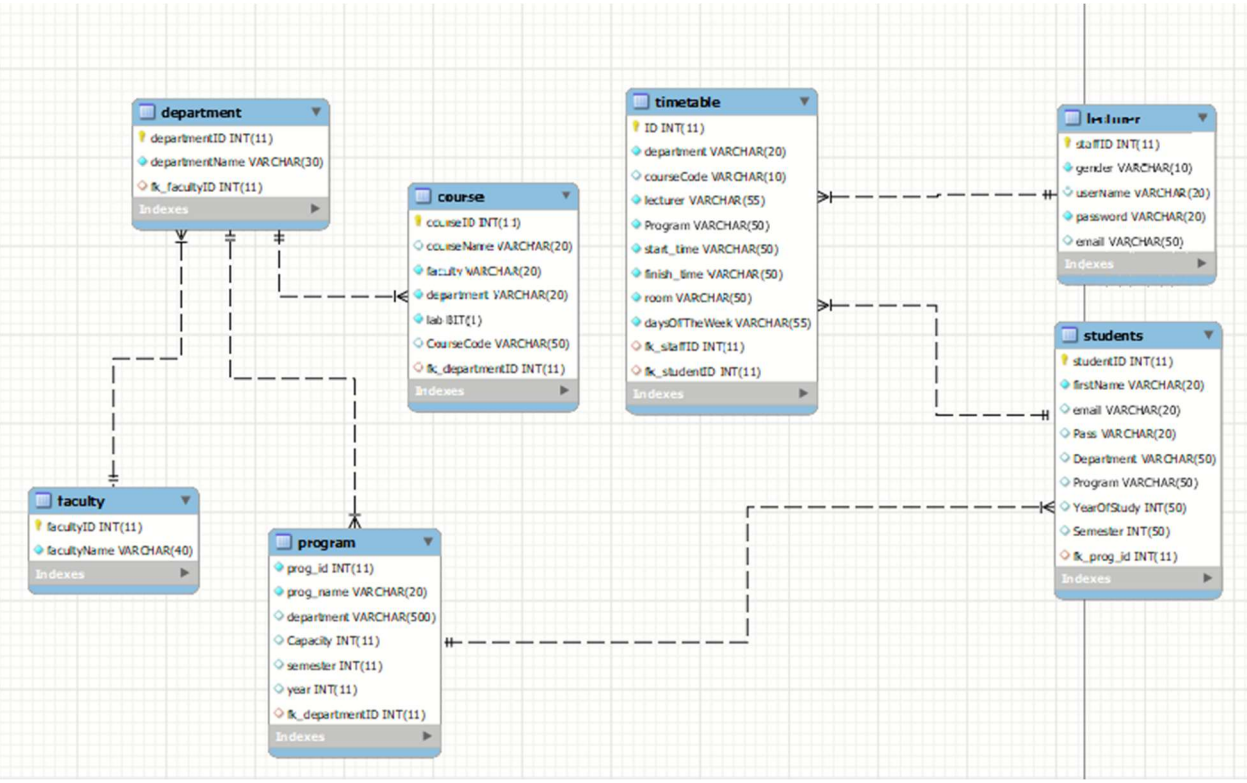


FIGURE 6 ENTITY RELATIONSHIP DIAGRAM

5.2 Physical design for computer laboratory scheduling system.

5.2.1 Home page

This is the Main Page of the project 'computer laboratory scheduling system'. On this page user will be able to register, contact and login once the user gets registered. Homepage interface is illustrated in the figure 7 below.

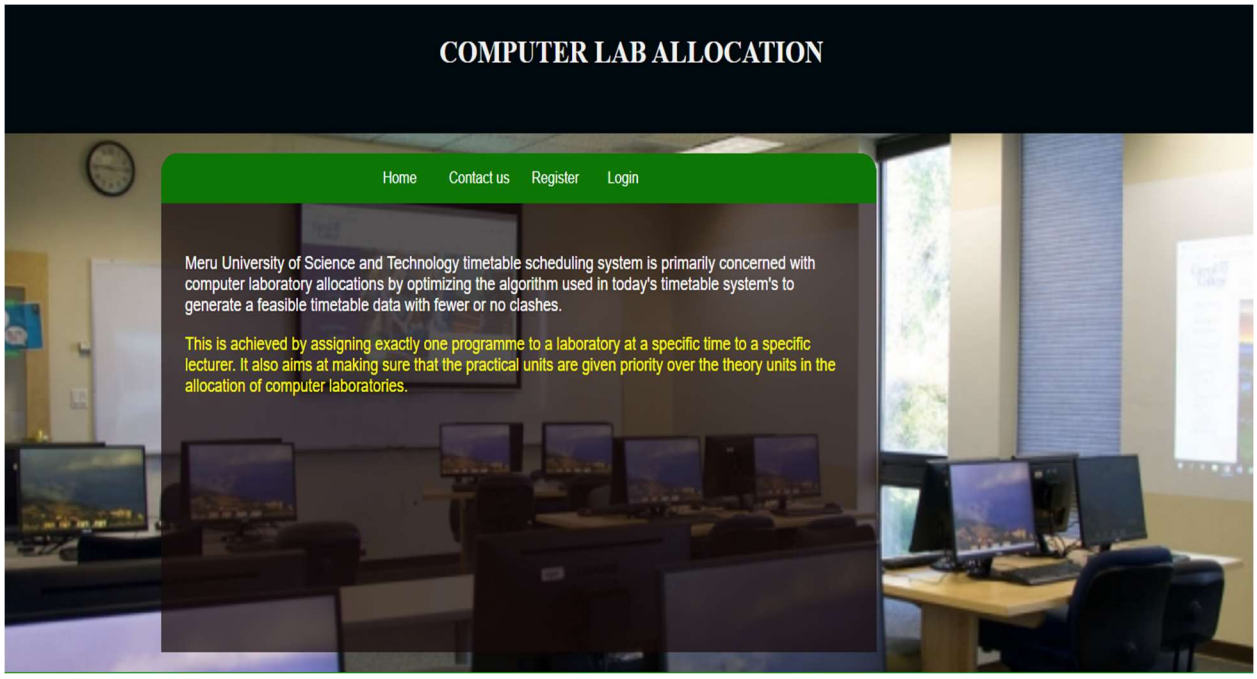


FIGURE 7 HOME PAGE INTERFACE

5.2.2 Registration form

On this registration form a user is required to fill in various fields in the form which gets stored into the database. This is illustrated in the figure 8 below.

FIGURE 8 REGISTRATION FORM INTERFACE

5.2.3 Login form

Once registered, a user only needs to login in every time they want to access the system. Login form is illustrated in the figure below.

FIGURE 9 LOGIN FORM INTERFACE

5.2.4 Admin panel

On this interface, once admin login is able to carry out all task required like adding or deleting programs, departments among others. Admin panel is illustrated in the figure below.

FIGURE 10 ADMIN PAGE INTERFACE

5.2.5 Student panel

On this interface, student login first thereafter is taken to this interface where students view the timetable, profile among others. The figure below gives student interface.

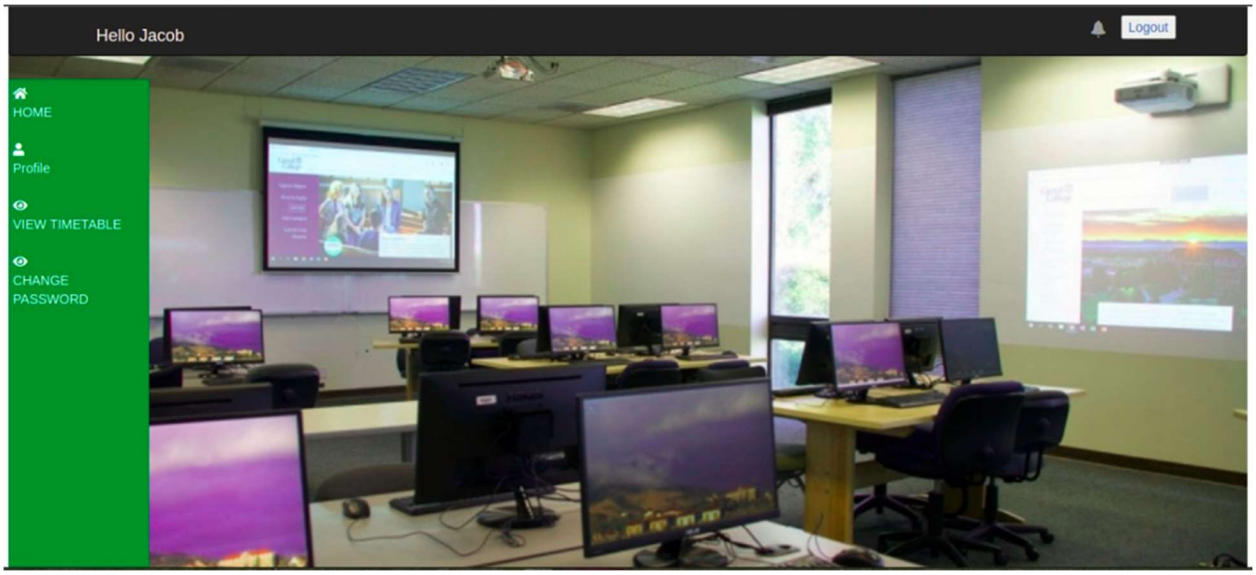


FIGURE 11 STUDENT DASHBOARD INTERFACE

5.3 Database design for computer laboratory scheduling system.

5.3.1 Students table

Students table include email, registration number, first name, last name, program, department and other attributes as shown in Table below. studentID is set as the primary key for the students table to identify the constraints. Table 2 below shows the student table

TABLE 2 STUDENT TABLE

Field	Type	Null	Key	Default	Extra
studentID	int(11)	NO	PRI	*NULL*	auto_increment
firstName	varchar(20)	NO		*NULL*	
email	varchar(20)	YES		*NULL*	
Pass	varchar(20)	YES		*NULL*	
Department	varchar(50)	YES		*NULL*	
Program	varchar(50)	YES		*NULL*	
YearOfStudy	int(50)	YES		*NULL*	
Semester	int(50)	YES		*NULL*	

5.3.2 Lecturer table

Lecturer table has the following attributes; first name, gender, last name, username, password, lectureID among others. LecturerID is the only primary key which is used to identify constraints and ensure that each data content is not exactly the same with another data content. Table below is the lecturer table. Table 3 below is a Lecturer table

TABLE 3 LECTURER TABLE

Field	Type	Null	Key	Default	Extra
staffID	int(11)	NO	PRI	*NULL*	auto_increment
gender	varchar(10)	NO		*NULL*	
userName	varchar(20)	YES		*NULL*	
password	varchar(20)	NO		*NULL*	
email	varchar(50)	YES		*NULL*	

5.3.3 Faculty table

Attributes for faculty table are facultyID and faculty Name. Primary key on faculty table facultyID which is used to identify the constraints and ensure data contents are not exactly the same. Figure 14 below illustrates faculty table.

TABLE 4 FACULTY TABLE

Field	Type	Null	Key	Default	Extra
facultyID	int(11)	NO	PRI	*NULL*	auto_increment
facultyName	varchar(40)	NO		*NULL*	

5.3.4 Department table

Attributes for department table are departmentID and department Name. Primary key on faculty table departmentID which is used to identify the constraints and ensure data contents are not exactly the same. Figure 15 below illustrates department table.

TABLE 5 DEPARTMENT TABLE

Field	Type	Null	Key	Default	Extra
departmentID	int(11)	NO	PRI	*NULL*	auto_increment
departmentName	varchar(30)	NO		*NULL*	
fk_facultyID	int(11)	YES	MUL	*NULL*	

5.3.5 Program table

Attributes for faculty table are prog_id, capacity, department, prog_name, semester and year. Primary key on faculty table prog_id which is used to identify the constraints and ensure data contents are not exactly the same. Figure 16 below illustrates program table.

TABLE 6 PROGRAM TABLE

Field	Type	Null	Key	Default	Extra
prog_id	int(11)	NO	PRI	*NULL*	auto_increment
prog_name	varchar(20)	NO		*NULL*	
department	varchar(500)	YES		*NULL*	
Capacity	int(11)	YES		*NULL*	
semester	int(11)	YES		*NULL*	
year	int(11)	YES		*NULL*	

5.4 Summary

The chapter covered logical design where entity-relationship diagram for computer laboratory scheduling system is given showing entities, relationship between them and the attributes for each entity. Physical design gives interfaces within computer laboratory scheduling system. Lastly, database design also has tables that have been discussed like student table, lecturer table among other tables in the system.

CHAPTER SIX

SYSTEM IMPLEMENTATION

6.0 Overview

The previous chapter has provided the design of the proposed solution. This chapter is about how the solution will be implemented in the context of proposed timetable management system. This chapter will further describe about equipment, tools, software and hardware required for implementing processes of the proposed system. Moreover, Algorithm implementation, Database implementation and Interface implementation are been thoroughly.

6.1 Implemented Features

6.1.1 Scheduling feature

The scheduling feature allows the timetable to make allocation of computer laboratory rooms that are available. Figure 12 below is code syntax of the scheduling feature.

```
$days=$_POST["days"];

$que=mysqli_query($conn,"select * from timetable where room='$room' and start_time='$startTime' and daysOfTheWeek='$days'");

    if($row=mysqli_num_rows($que))
    {
    print "<font color='red'>Room is occupied</font>";
    }
    else
    {
        $que=mysqli_query($conn,"select * from timetable where Program='$programName' and start_time='$startTime' and daysOfTheWeek='$days'

    if($row=mysqli_num_rows($que))
    {
    print "<font color='red'>This program is in another session at this time</font>";
    }
    else
    {
        $que=mysqli_query($conn,"select * from timetable where lecturer='$lecturer' and start_time='$startTime' and daysOfTheWeek='$days'");

        if($row=mysqli_num_rows($que))
        {
        print "<font color='red'>Lecturer as another class at this time</font>";
        }
        else
        {
$sql="INSERT INTO timetable (department,Program,lecturer,courseCode,daysOfTheWeek,start_time,finish_time,room) VALUES('$departmentName','$progr
```

FIGURE 12 SCHEDULING FEATURE

6.1.2 Notification Feature

The notification feature allows the timetabler to send a notification to both the lecture and the student whenever the timetable is out. On the other end the lecturer and the student will receive the notifications. Figure 13 below is a code syntax of the notification feature.

```
<?php

if(isset($_POST["subject"]))

{

include("../connect.php");

$subject = mysqli_real_escape_string($conn, $_POST["subject"]);

$comment = mysqli_real_escape_string($conn, $_POST["comment"]);

$query = "INSERT INTO comments(comment_subject, comment_text)VALUES ('$subject', '$comment')";

if(mysqli_query($conn, $query))
{

print "<h3 >Notification sent Successfuly</h3>";
}

}

?>
```

FIGURE 13 NOTIFICATION FEATURE

6.1.3 Booking Feature

The booking feature allows the lecturer to book a free slot that has not been assigned to any program.

Figure 14 below is a code snippet of the Booking feature.

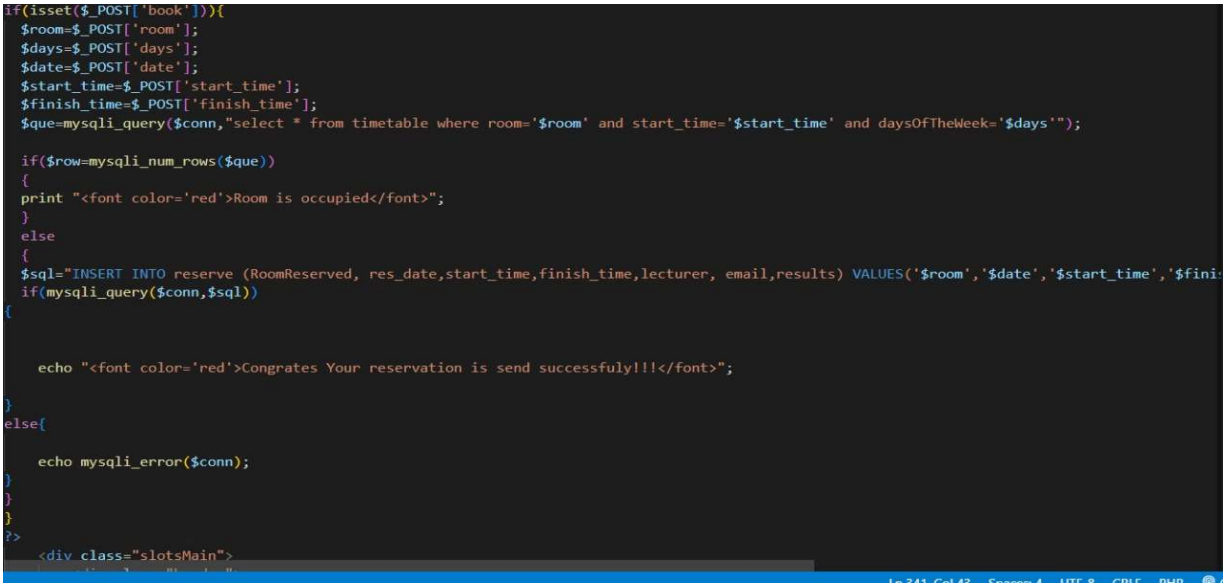


FIGURE 14 BOOKING FEATURE

6.1.4 Reservation Feature

This feature allows the timetabler to reserve a booked slot. Once the lecturer has a booked a slot, the timetabler is able to make the reservations. Figure 15 below is a code snippet of the Reservation

Feature.

```php
<?php
require_once('../class/Book.php');
if(isset($_POST['tracker'])){
    $tracker = $_POST['tracker'];
    // echo $tracker;

    $result = $book->updateBook($tracker);
    $return['valid'] = true;
    $return['msg'] = "Reserved Successfully!";
    echo json_encode($return);
}//isset

$book->Disconnect();
```

FIGURE 15 RESERVATION FEATURE

6.1.5 Email notification Feature

The email notification feature enables the timetabler to send an email notification to the specific lecturer who has reserved a computer laboratory. Figure 16 below is a code syntax for the email notification feature.

```php
//$tracker = $_POST['tracker'];
$sqlreminder = mysqli_query($conn,"SELECT * FROM  reserve WHERE results='pending';");
$lecturers = mysqli_fetch_all($sqlreminder,MYSQLI_ASSOC);
foreach($lecturers as $lecturers){
    // send email reminder==============================

    $email = $lecturers['email'];
    $name = $lecturers['lecturer'];
    //$time = $patient['tyme'];
    //$dname = $patient['dname'];


        $to = $email;
        $subject = "Laboratory reservation";

        $message = "$name, how are you? The laboratory space you requested has been reserved. ";
        // Always set content-type when sending HTML email
        $headers = "MIME-Version: 1.0" . "\r\n";
        $headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";

        // More headers
        $headers .= 'From: <evanschaun@gmail.com>' . "\r\n";


        if(mail($to,$subject,$message,$headers)){
            echo "sent successfully";
        }else{
            echo "Error";
        }
}
```

FIGURE 16 EMAIL NOTIFICATION

6.2 System Testing

System testing refers to process of evaluating the software with intention to find out error in it[5]. It is a technique aimed at evaluating an attribute or capability of a program or product and determining that it meets its quality[17].

Importance of System Testing

System testing is needed to verify and validate that the software being built, meets the set specifications. If not, we may probably lose the client[17]. So in order to make it sure, that we provide the client a proper software solution, we go for testing. Testing ensures that what you get in the end is what you wanted to build. We check out if there is any problem, any error in the system, which can make software unusable by the client. This helps in the prevention of errors in a system.

6.2.1 Scheduling Feature

A scheduler should be able to use the system to create class schedules. The module's functionality was tested, and data was acquired based on difficulties such as one or more lecturers being allocated to two programs at the same time, or more than one program is assigned to one laboratory. This is shown below in Table 7

TABLE 7 SCHEDULING FEATURE

Test Field	**Test data**		**Test type**	**Test Result**
Lecturer and Time	Ashley	08:00-11:00	Ability to check whether The Lecturer has another session at that time	pass

Program, room and Time	BCS 1.1	Lab 1	ability to determine if the laboratory that will be assigned to a program is currently occupied	pass
Program and Time	BCS 1.1	08:00-11:00	Ability to see whether The Program has a different schedule available at that time	pass

6.2.2 Notification Feature

The system should allow the timetabler send a notification to both the lecturer and student informing them that the timetable is out. This feature was tested on the ability of the module to allow input into the text area, submit the input data, validate the data, store the data in the database and send to the lecturer and student. Table 8 shows the results

TABLE 8 NOTIFICATION FEATURE

Test Field	**Test data**	**Test Result**	
		Data Validation	**Data Storage into the Database**
Subject	Timetable	Pass	Pass
Comment	Timetable is out	Pass	Pass

6.2.3 Booking Feature

The system allows lecturers to book open slots. The module's functionality was verified, and data was collected based on whether or not the room to be booked was occupied at the preferred time. Table 9 below shows the booking feature

TABLE 9 BOOKING FEATURE

Test data		**Test type**	**Test Result**
Lab 1	08:00-11:00	Ability to determine whether a lecturer will be able to book a slot.	Pass
Lab 2	11:00-13:00	Check if the system will prevent a lecturer from booking an occupied laboratory.	Pass

6.2.4 Reservation Feature

The system allows scheduler to be able to view, accept, or delete the booked laboratory. The functioning of the module was tested, and data was collected based on the success of each field. Table 10 below shows the reservation feature.

TABLE 10 RESERVATION FEATURE

Test data		**Test type**	**Test Result**
Lab 15		Accept	Pass
Lab 17		Delete	Pass

6.2.5 Email Notification Feature

When the lecturer's slot is reserved, the system should automatically send an email to the lecturer. The success of the email message to that specific lecturer was used to test the functionality of this module and the data was collected as shown in table 11 below

TABLE 11 EMAIL NOTIFICATION FEATURE

Test Field	Test data	Test Result	Email Delivery
Lecturer email	Slot is reserved	Pass	Pass

6.3 System Changeover

This is the process of putting the new information system online and retiring the old system. There are four system changeover approaches phase, parallel, pilot and direct[28].

6.3.1 Phase Changeover

The problems associated with existing system are determined in the previous steps, the analyst can initiate preparation how the proposed system will fix those difficulties and try to convert difficulties into opportunity[28].

6.3.2 Direct Changeover

Is a direct approach where any old running system is directly over write by the new one. This approach may cause the system changeover from the older system with a completely new system which starts operation immediately.

6.3.3 Parallel Changeover

Parallel word is use when two things run simultaneously, so here two operations run simultaneously[27]. All of the data and information which are used as input into the old system may be used as input into the new system. Results of the new system is verified with old system and also verify the check list which are prepared in the previously. Ultimately, the old system or module is stopped working only the new module work successfully. This has its advantages If anything goes wrong with the new system, the old system will act as a back-up.

6.3.4 Pilot Changeover

The pilot operation is the basic changeover technique, which involves the various modules implementing a complete new system at a specific sector of the organization[27]. The group or subgroup that uses the new modules are called the pilot batch or pilot site.

6.4 Documentation

Documentation can be defined as the process of recording key aspects of a project details and producing the documents that are required to implement it successfully. There are two types of documentation internal and external documentation.

6.4.1 Internal documentation

These entails details of the program that are explained by comments placed within the code.

Internal documentation is intended for programmers who will be working on the software. Figure 17 below shows how comments have been implemented

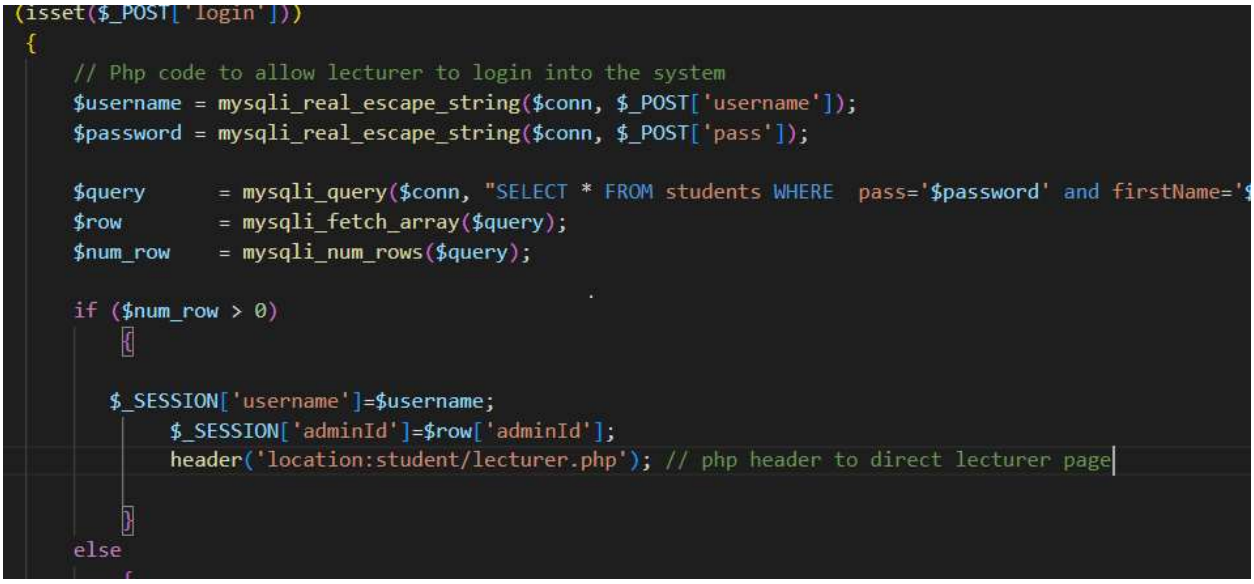


FIGURE 17 INTERNAL DOCUMENTATION

6.4.2 External documentation

This is written in a place where people who need to use the software can read about how to use the software. This type of documentation is intended for people who will use the software. The user manual is attached in the appendix.

6.5 Summary

This chapter presented a detailed system implementation on the computer laboratory scheduling system. A discussion on implemented features, system testing, system changeover and documentation of the computer laboratory scheduling system was presented. In the implemented features all features which are objectives of the system were as well explained while in system testing all testing was done and evaluated according to expected results and tables generated. The implemented system was developed through parallel changeover which is also discussed. The Internal and external documentation was done and a user manual was created.

CHAPTER SEVEN

FINDINGS, CONCLUSION AND RECOMMENDATION

7.1 Findings

Timetable scheduling problems have received tremendous attention from disciplines like Operations Research and Artificial Intelligence throughout the years. Several optimization techniques are used to solve them and produce optimal or near-optimal solutions instead of the exact solution. In universities, however, the semester timetable scheduling process has become very complex since there are four or five batches of students, and many student groups and some student groups take some courses together. There are some courses in a specific batch that are studied by all the students. You have to make sure that there are no conflicts between student groups, lecturers, and lecture halls when planning the timetable. Large-scale schedules such as university schedules may require many hours of work spent by a qualified person or team to produce high quality schedules with optimum restriction satisfaction.

Anuja Chowdhary[14] in this paper, through the analysis and the summarization of the existing problems, a mathematical model for the course timetable system is proposed. At the same time, through the use of the pattern recognition technology in artificial intelligence, aiming at this mathematical model a new university course timetable system design program is proposed and realized. This program not only can well solve the shortages of the existing course timetable system, but also is simple and easy to operate, has strong versatility.

K. Jilcha Sileyew [20] found that evolutionary techniques have been used to solve the timetable-scheduling problem. Methodologies like Genetic Algorithms (GAs), Evolutionary Algorithms (EAs) have been used with mixed success. In this paper, we have reviewed the problem of educational time table scheduling and solving it with genetic algorithm. We have further solved the problem with a mimetic hybrid algorithm, genetic artificial immune network (GAIN) and compare the result with that obtained from GA. Results show that GAIN is able to reach the optimal feasible solution faster than that of GA.

[21] Finding a feasible lecture/tutorial timetable in a large university department is a challenging problem faced continually in educational establishments. This paper presents an evolutionary algorithm (EA) based approach to solving a heavily constrained university timetabling problem. The approach uses a problem-specific chromosome representation. Heuristics and context-based reasoning have been used for obtaining feasible timetables in a reasonable computing time. An intelligent adaptive mutation scheme has been employed for speeding up the convergence. The comprehensive course timetabling system presented in this paper has been validated, tested and discussed using real world data from a large university.

ChayaAndradi, and semindaPremaratne[6], found out that generating timetable in a large university department is a challenging problem faced continually in educational institutions. The researchers developed a medium scaled university timetable management system that was able to generate a near optimal timetables using the principles of genetic algorithm that was easily understood, it reduced the paper work and it was an automated system that was helpful to authorities of the faculty. However, the timetabling system was not fit for the medium scale universities since it was not implemented as a framework.

According to Ghazi Alnowaini and Amjad Abdullah Aljomai[13], they found out that generating a semester timetable has become very complex in the universities. The researchers developed a genetic algorithm for solving university timetabling problems using dynamic chromosomes that were able to reduce and prevent conflicts in times and halls. It also increased in the number of the generation that reduced the number of conflicts for the courses. However, the system did not include the idea to all faculties in the institution and did not allow for the learners to request reservation of free times for the lecture halls.

7.2 Conclusion

The web application developed contains three modules; the Administrator, the Lecturer and the Student. The Administrator logs into the system using his or her credentials and his or her functions are as follows; Manage programs, classes, faculty, department, sessions, timetable, lecturer and courses. The Administrator also reserves the room that has been booked by the lecturer and sends notification to the lecturer and the student when the timetable is out. The Lecturer and the Student are the direct users of the system. The Student and the Lecturer can to register into the system, log into the system, update their profile they can also change the password and log out. After doing so, they can both view the timetable, add a comment and register into the system. The student and

Lecturer are also able to receive notification via their emails when the timetable is out. The Lecturer is also able to book a free room that is empty at his or her appropriate time. The overall view of timetabling system is represented using use-case diagram, class diagram, sequence diagram, data flow diagram and activity diagram that are given in detail in figure 1,figure 2,figure 3,figure 4 and figure 5 respectively.

Using the web application developed, the school is able to generate timetables with the greatest of ease. The automation of this system has greatly reduced cases of conflict and is able to create timetables from scratch according to user preference for the potential users of the system. Also, the system is able to come up with feasible timetables on time using the least amount of energy whereas significantly reducing the number of clashes. Implementing such a system in the university will prove to be an added advantage for it will save up a lot of time and significantly cut down on any expenses used during the generation of timetables. Since the system is web based, it will be accessible to everyone and also the expenses on web-hosting are significantly low hence more reason to implement the system.

7.3 Recommendations

The following are additions that can be made to this project:

- Development of the system as a mobile application.
- Incorporating the system to generate master timetable for the departments and to the entire college by making further modifications keeping the approach and techniques used for this project.
- Send alerts via sms to users when any changes are made on the timetable.

- Allow prioritization of users, such that high level users are given first priority in booking courses and timeslots.

APPENDIX

0.0 Questionnaire

TABLE 12 LECTURES QUESTIONNAIRE

	Strongly Agree	Agree	Neither agree nor disagree	Disagree	Strongly disagree
Are there any collisions that occurs in room allocations?					
Would you like the allocation system to be implemented?					
Do you think that the allocation system will overcome the collision challenge?					

	YES	NO
Is the time allocated for the computer laboratory minimum or enough?		
Is there conflicts in the allocation of units?		
Are the computer laboratories spacious enough to accommodate the students?		

TABLE 13 TIMETABLER QUESTIONNAIRE

What type of algorithm do you use for timetabling?	
What challenge do you get in allocation of computer laboratories to specific departments?	

Do you practice biasness when allocating the laboratories to certain depatments?	
Do you have the enough resources for creating a timetable?	

1.1 Feasibility report for Timetabling system

1.1.1 Purpose

The purpose of the feasibility report is to address the challenge on how students and lecturers struggle when there is a collision in room allocation and to explore recommendations to resolve the problems[30]. This report investigated two possible alternative solutions to correct this problem.

1.1.2 Alternative Solutions

1.1.2.1 First Solution

The first solution is to develop Timetabling system which allows a lecturer to reserve a room for students. The solution ensures that room collision will be less since when a lecturer books a room for the students at a particular time, when any other lecturer tries to occupy that room, they are notified that the room has already been booked for another group[30].

Economic feasibility: This is not an expensive solution[30], but still it can be expensive solution where you have a lot of rooms to do allocation. But in doing this exercise, it will help the institution to solve the issue of room and time collisions. This exercise will incur a total cost of KSH700,000 which will be expected to be paid by the institution.

Operational Feasibility: This solution is not expected to alter or replace the normal timetabling system but it is going to provide a better and quicker way of room allocation[30].

Schedule Feasibility: In order to implement this solution, the team of programmers will require a duration of seven months[30]. The first two will be used to design the solution, the next one month will be used for testing and debugging. One month for implementations another one month for training the different types of users and finally the last month for sorting user issues and challenges.

1.1.2.2 Second Solution

The second solution is upgrade the already existing software and hardware and into it incorporate the necessary modules and continue working with the system.

Economic feasibility: This solution will incur the cost of hiring and training new registration officers[30]. It will require hiring a group of programmers in order for them to come up with an algorithm and implement it. The programmers will have to spend a cost of KSH500,000 because the system will not be built from scratch.

Operational Feasibility: This solution is not expected to fully replace the normal operation of the computer laboratory scheduling system, however the challenges experienced by the students and lecturers will be less.

Technical Feasibility: Implementing this solution will require procuring new hardware and software[30]. However, the problem of room allocation will become less than before since a new algorithm will be implemented.

Schedule Feasibility: There will be less time required to implement this solution since it's the solution current in use.

1.2 Recommendation

After undertaking the feasibility study, based on the two solutions discussed above. I recommend the first solution. It may seem to be the most expensive solution but it is the most viable solution in terms of providing a solution for the current Timetabling system.

User Manual

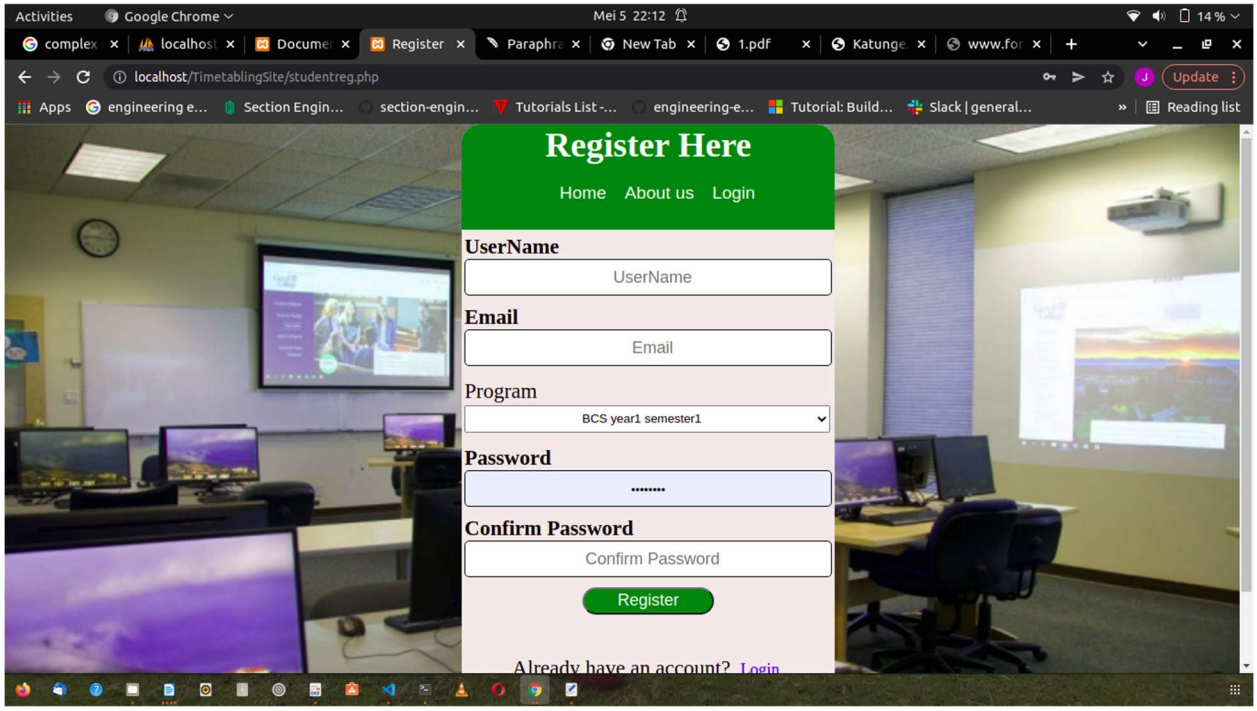
This section gives a description of the procedures to be followed when using the computer laboratory scheduling System.

One needs to have installed Xampp server on their computer. To launch the web-based application, first start the Xampp server. You may find the start-up icon on your desktop, or you can go to 'Programs' and start it up from there. Open the browser of your choice and specify the local host address which directs you to where your application is stored. A user has to present his/her identification details together with a password in order to be authorized to access and utilize the scheduling system. There are various links on the home page that allow a user to perform various tasks. Such tasks and access to information depends on the administration rights a user has. Information is limited to ordinary system users, while administrative user has rights to access all the information in the system.

There are three different user levels. The student, lecturer and Timetabler.

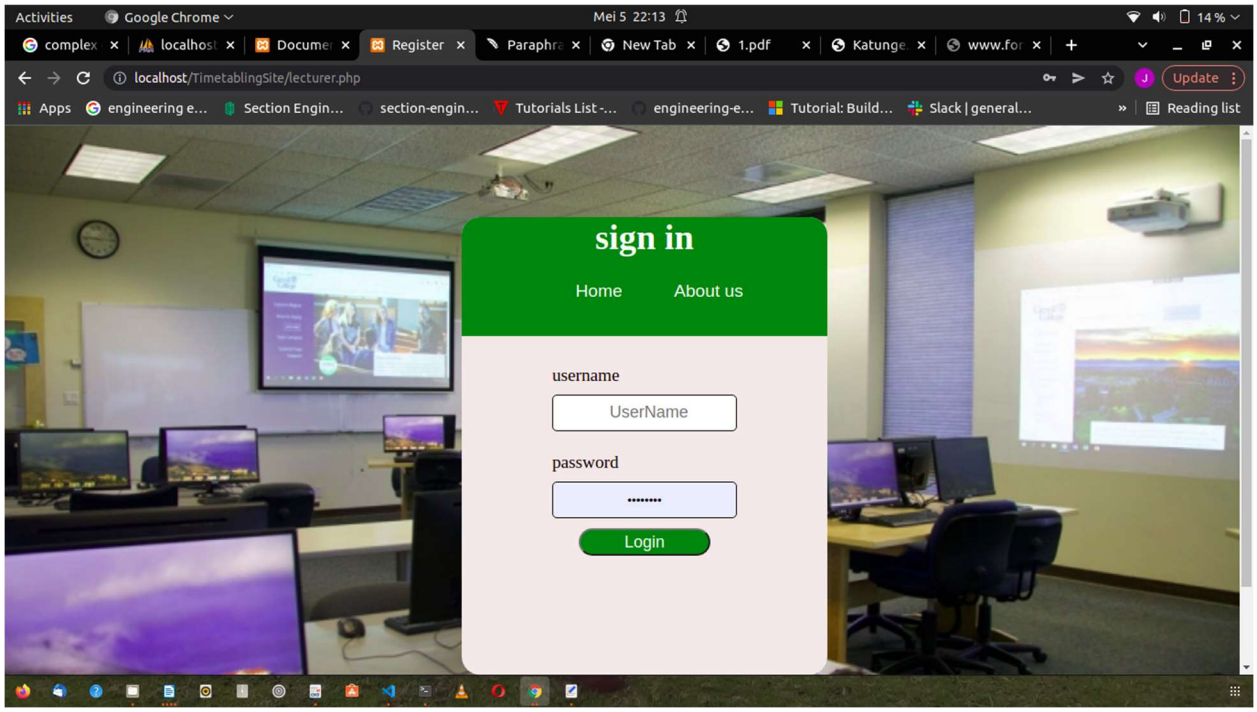
The student user login, leads to a group of pages that allow the user to view the timetables and profile.

2.1students' registration

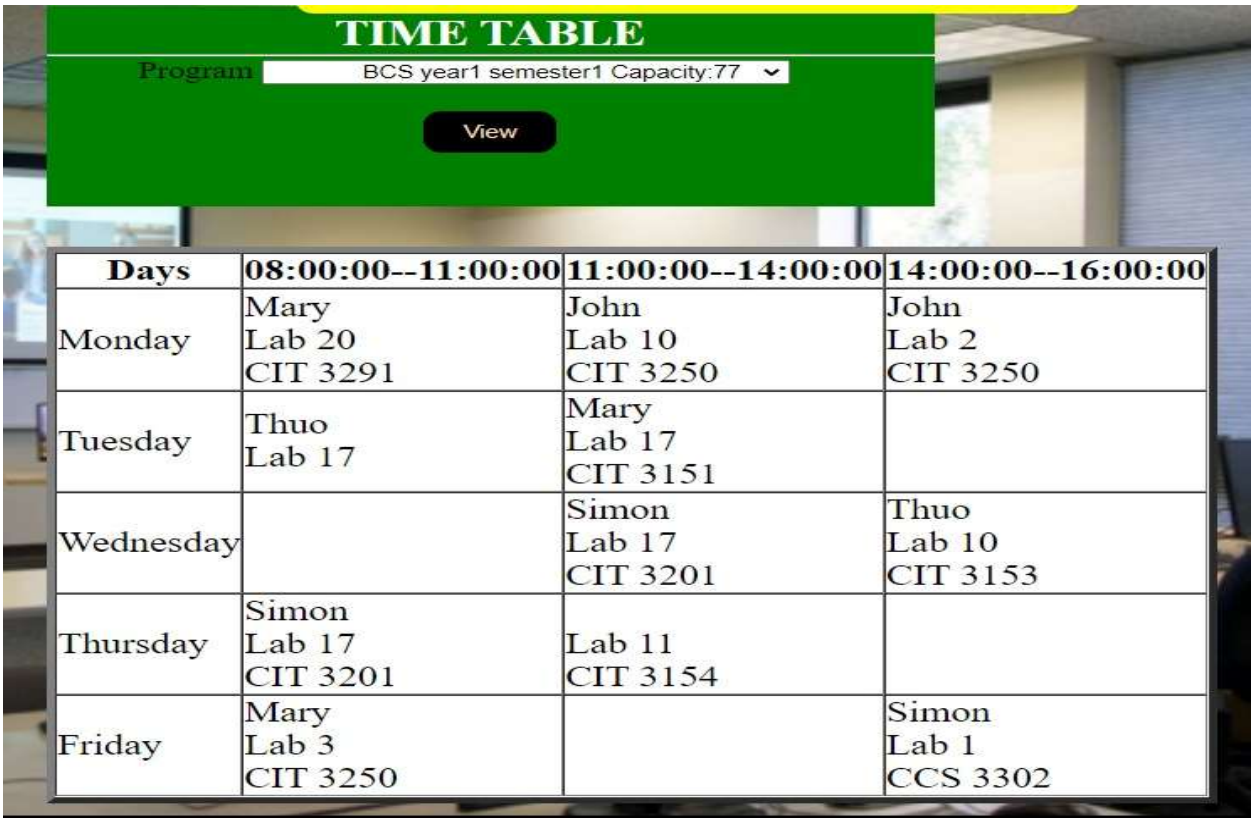


2.2 lecturer's and Students login

The lecturer login leads you to a group of pages that allow the lecturer to book free slots, view timetable and view profile.

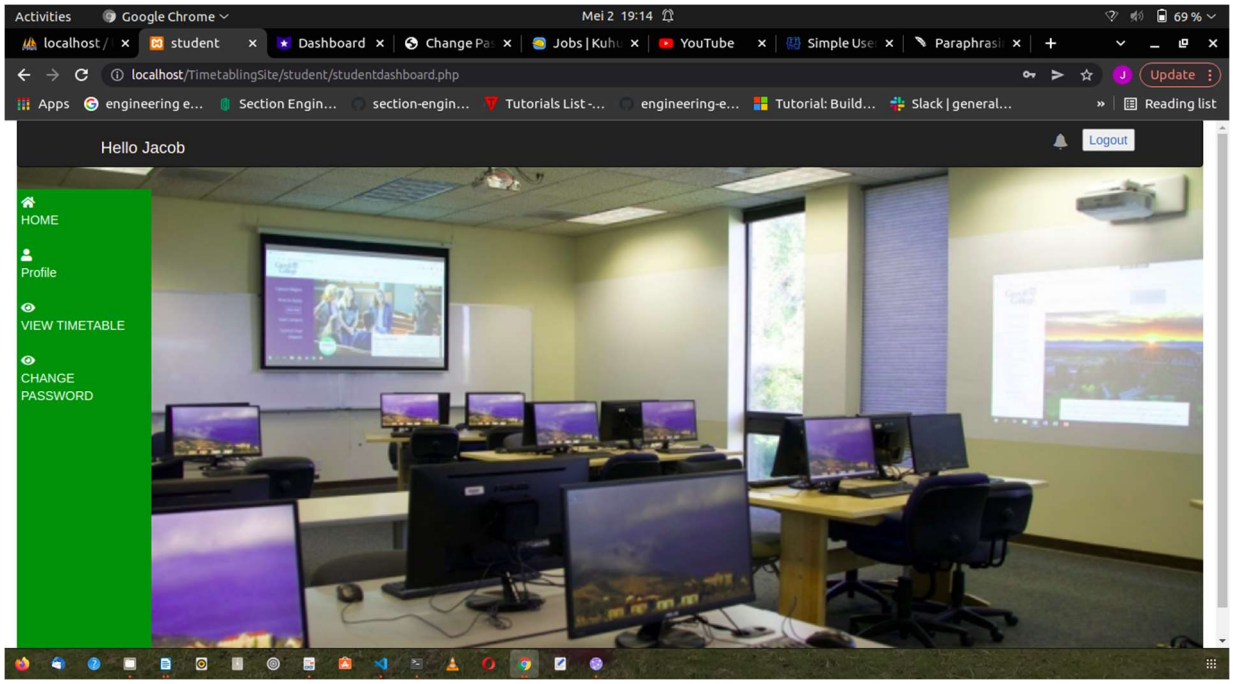

Activities
Google Chrome
Mei 5 22:13
14 %
localhost/TimetablingSite/lecturer.php
Update
Apps
Reading list
sign in
Home
About us
username
UserName
password
Login

2.3 View timetable Allows the student to access the timetable and to view the allocated class with respective lecturers

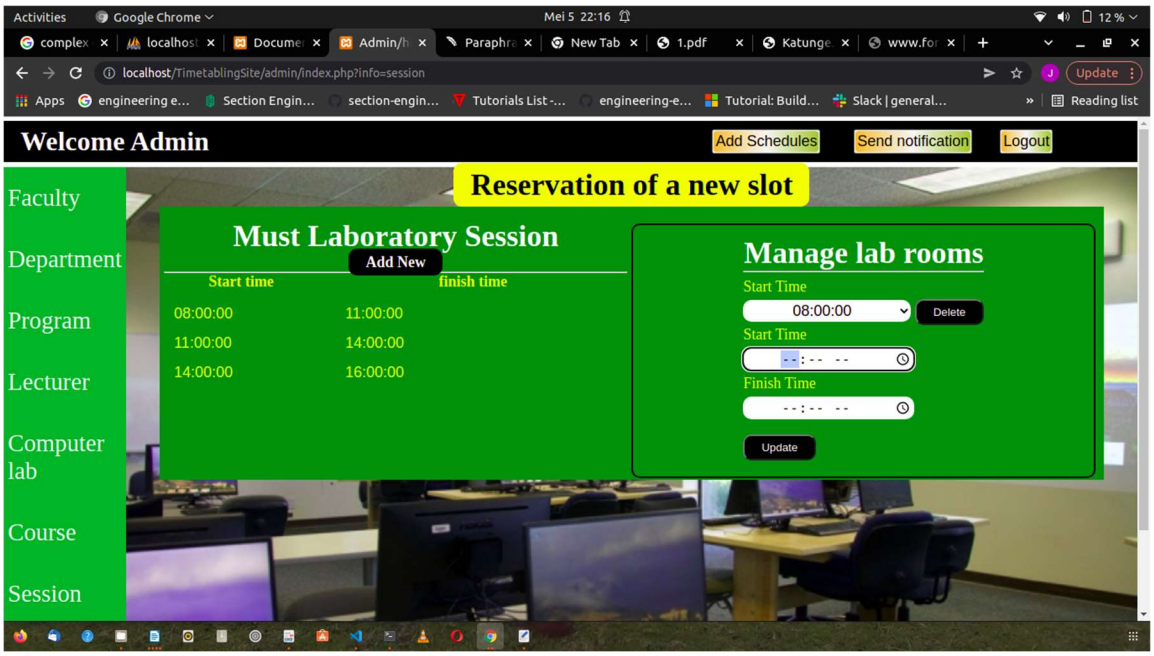


Days	08:00:00--11:00:00	11:00:00--14:00:00	14:00:00--16:00:00
Monday	Mary Lab 20 CIT 3291	John Lab 10 CIT 3250	John Lab 2 CIT 3250
Tuesday	Thuo Lab 17	Mary Lab 17 CIT 3151	
Wednesday		Simon Lab 17 CIT 3201	Thuo Lab 10 CIT 3153
Thursday	Simon Lab 17 CIT 3201	Lab 11 CIT 3154	
Friday	Mary Lab 3 CIT 3250		Simon Lab 1 CCS 3302

The Timetabler login allows the timetabler to allocate rooms and edit other system information he is are able to navigate through these pages and perform these tasks using the side bar menu.



2.4 Reservation of a slot it contains details of where you user needs to reserve a slot



2.5 Booking a slot

It shows the laboratory capacities, date and start time and finish time of the slot.

1.9 References

[1] P. S. Chen and Z. Y. Zeng, "Developing two heuristic algorithms with metaheuristic algorithms to improve solutions of optimization problems with soft and hard constraints: An application to nurse rostering problems," *Appl. Soft Comput. J.*, vol. 93, Aug. 2020, doi: 10.1016/j.asoc.2020.106336.

[2] A. O., U. K. C., N. J., and O. F.B, "A Soft-Computing Based Course Timetabling System for Schools in Nigeria," *NIPES J. Sci. Technol. Res.*, vol. 2, no. 3, p. 283, Aug. 2020, doi: 10.37933/nipes/2.3.2020.28.

[3] A. Le Platte, D. Bartholomew, E. Jack, and A. Nikov, "THE UNIVERSITY OF WEST INDIES FACULTY OF SCIENCE AND AGRICULTURE DEPARTMENT OF COMPUTING AND INFORMATION TECHNOLOGY COMP 6650 Web Usability Semester Project Report Project Title: User centered redesign of the website-www.tntschools.com," 2010. [Online]. Available: www.tntschools.com.

[4] A. M. Hambali, Y. A. Olasupo, and M. Dalhatu, "Automated university lecture timetable using Heuristic Approach," *Niger. J. Technol.*, vol. 39, no. 1, pp. 1–14, 2020, doi:

10.4314/njt.v39i1.1.

[5] S. Gaku, “UNIVERSITY OF NAIROBI SCHOOL OF COMPUTING AND INFORMATICS SYSTEM: WEB ENABLED AGENT-BASED UNIVERSITY TIMETABLING ASSISTANT.”

[6] ChayaAndradi and SamindaPremaratne, “Utilization of Timetable Management System to a Medium Scaled University,” *Int. J. Mod. Res. Eng. Technol. www.ijmret.org*, vol. 1, no. 2, pp. 8–16, 2016.

[7] J. S. Tan, S. L. Goh, G. Kendall, and N. R. Sabar, “A survey of the state-of-the-art of optimisation methodologies in school timetabling problems,” *Expert Syst. Appl.*, vol. 165, Mar. 2021, doi: 10.1016/j.eswa.2020.113943.

[8] R. E. Balcita and T. D. Palaoag, “Integration of school management systems using a centralized database (Ismscd),” *Int. J. Inf. Educ. Technol.*, vol. 10, no. 9, pp. 704–708, Sep. 2020, doi: 10.18178/ijiet.2020.10.9.1446.

[9] U. Majid, “Research Fundamentals: Study Design, Population, and Sample Size,” *Undergrad. Res. Nat. Clin. Sci. Technol. J.*, vol. 2, no. 1, pp. 1–7, 2018, doi: 10.26685/urncst.16.

[10] R. E. Balcita and T. D. Palaoag, “Integration of school management systems using a centralized database (Ismscd),” *Int. J. Inf. Educ. Technol.*, vol. 10, no. 9, pp. 704–708, 2020, doi: 10.18178/ijiet.2020.10.9.1446.

[11] haftom hailay *et al.*, “Web Based Student Information Management System Related papers A Review of Comput erized Payroll Syst em pronot i roy Aut onomous Int egrat ed Library

E-Proct ored Model: Elect ronic Solut ion Archit ect for Exam Derelict ion in Kenya Web Based Student Information Management System," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 2, 2013, [Online]. Available: www.ijarcce.com.

[12] A. M. Hambali, Y. A. Olasupo, and M. Dalhatu, "Automated university lecture timetable using Heuristic Approach," *Niger. J. Technol.*, vol. 39, no. 1, pp. 1–14, Apr. 2020, doi: 10.4314/njt.v39i1.1.

[13] G. Alnowaini and A. A. Aljomai, "Genetic Algorithm for Solving University Course Timetabling Problem Using Dynamic Chromosomes," *2021 Int. Conf. Technol. Sci. Adm. ICTSA 2021*, 2021, doi: 10.1109/ICTSA52017.2021.9406539.

[14] 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, pp. 410–414, 2014, [Online]. Available: www.ijcsmc.com.

[15] N. I. Ilham, E. H. M. Saat, N. H. A. Rahman, F. Y. A. Rahman, and N. Kasuan, "Auto-generate scheduling system based on expert system," *Proc. - 7th IEEE Int. Conf. Control Syst. Comput. Eng. ICCSCE 2017*, vol. 2017-Novem, no. November, pp. 6–10, 2018, doi: 10.1109/ICCSCE.2017.8284370.

[16] A. Muklason, N. D. Angresti, P. C. Bwananesia, V. A. Supoyo, and S. Hidayatul, "Automated Examination Timetabling Optimization Using Greedy-Late Acceptance-Hyperheuristic Algorithm; Automated Examination Timetabling Optimization Using Greedy-Late Acceptance-Hyperheuristic Algorithm," 2018.

[17] S. Limanto, N. Benarkah, and T. Adelia, "Thesis examination timetabling using genetic

algorithm," *Int. Electron. Symp. Knowl. Creat. Intell. Comput. IES-KCIC 2018 - Proc.*, pp. 6–10, 2019, doi: 10.1109/KCIC.2018.8628572.

[18] D. Kusumawardani, A. Muklason, and V. A. Supoyo, "Examination timetabling automation and optimization using greedy-simulated annealing hyper-heuristics algorithm," *Proc. 2019 Int. Conf. Inf. Commun. Technol. Syst. ICTS 2019*, pp. 164–169, 2019, doi: 10.1109/ICTS.2019.8850932.

[19] P. G. Daniel, D. A. O. Maruf, and D. B. Modi, "Paperless Master Timetable Scheduling System," *Int. J. Appl. Sci. Technol.*, vol. 8, no. 2, 2018, doi: 10.30845/ijast.v8n2a7.

[20] K. Jilcha Sileyew, "Research Design and Methodology," *Cyberspace*, no. August, 2020, doi: 10.5772/intechopen.85731.

[21] N. Pawar, "6. Type of Research and Type Research Design," *Soc. Res. Methodol.*, vol. 8, no. 1, pp. 46–57, 2020, [Online]. Available: https://www.kdpublications.in.

[22] S. Shukla, "Concept of Population and Sample," *How to Write a Res. Pap.*, no. June, pp. 1–6, 2020, [Online]. Available: https://www.researchgate.net/publication/346426707_CONCEPT_OF_POPULATION_AND_SAMPLE.

[23] N. Pawar *et al.*, "Genetic Algorithm For University Course Timetabling Problem," *Angew. Chemie Int. Ed. 6(11), 951–952.*, vol. 8, no. 1, pp. 1–6, 2020, doi: 10.26685/urncst.16.

[24] M. Elfil and A. Negida, "Sampling methods in clinical research; an educational review," *Archives of Academic Emergency Medicine*, vol. 7, no. 1. 2019.

[25] M. E. Buchanan, "Methods of data collection," *AORN J.*, vol. 33, no. 1, 1981, doi:

10.1016/S0001-2092(07)69400-9.

[26] P. G. Daniel, D. A. O. Maruf, and D. B. Modi, “Paperless Master Timetable Scheduling System,” *Int. J. Appl. Sci. Technol.*, vol. 8, no. 2, 2018, doi: 10.30845/ijast.v8n2a7.

[27] R. S. P. and P. B. R. Maxim, “Software Engineering : A Practitioner’s Approach Eight Edition" in Process Model.” p. 880, 2015.

[28] I. Sommerville, *TENTH edition Tenth Edition*. 2016.

[29] S. Shukla, “Research design Research design,” *Res. Soc. Sci. Interdiscip. Perspect.*, no. September, pp. 68–84, 2017, [Online]. Available: file:///E:/Documents/dosen/buku Metodologi/[John_W._Creswell]_Research_Design_Qualitative,_Q(Bookos.org).pdf.

[30] M. Mukherjee and S. Roy, “Feasibility Studies and Important Aspect of Project Management,” *Int. J. Adv. Eng. Manag.*, vol. 2, no. 4, p. 98, 2017, doi: 10.24999/ijoaem/02040025.

[31] A. O. Approach, *SYSTEMS ANALYSIS & DESIGN An Object-Oriented Approach with UML*. .

[32] H. Techie-Menson and P. Nyagorme, “Design and Implementation of a Web-Based Timetable System for Higher Education Institutions,” vol. 7, no. March, pp. 1–13, 2021.

[33] A. S. Gaikwad, F. A. Kadri, S. S. Khandagle, and N. I. Tava, “Review on Automation Tool for ERD Normalization,” *Int. Res. J. Eng. Technol.*, vol. 4, no. 2, pp. 1323–1325, 2017, [Online]. Available: https://irjet.net/archives/V4/i2/IRJET-V4I2259.pdf.