Online Course Allocation System for Academic Staff in Multi-Campus Universities

Case Study: Uganda Martyrs University

NKUGWA FRED

2017-B072-30025

A Research Project Report Submitted to the Faculty of Science in Partial Fulfillment of the Requirements for the Award of the Bachelor of Science in Information Technology of Uganda Martyrs University

September, 2020.

DECLARATION

I, NKUGWA FRED, declare that this research is my original work and it has never been submitted in any Institution for any award. Works of other authors used as sources of literature have all been acknowledged and referenced appropriately.

Name of Researcher: NKUGWA FRED

Researcher's Signature: 

Date: _____03/12/2020_____

APPROVAL

This research has been done under my supervision and is submitted with my approval.

Signature  _____ Date____03/12/2020_____

Name of Supervisor: Babirye Nanteza Lucy

DEDICATION

This piece of work is dedicated to the lovely family of my mum Mrs. Najjumba Rose, my sister, Nanyingi Florence, my brother, Ssebalu Frank who worked tirelessly sacrificing all they had towards my education, my dear sponsor, Ramiro Zelada, Grandmother, Mrs. Nakalema Maxesia, my dear girlfriend, Nakimbejja Lilian and friends and lastly my dear supervisor, Ms. Babirye Nanteza Lucy who has provided continuous support towards the production of this report.

May the almighty bless and reward you abundantly.

ACKNOWLEDGMENTS

First and foremost, I am ever grateful to the almighty God who has always made every effort possible. His guidance, care, and love are amazing. Secondly, my heartfelt appreciation goes to my family, girlfriend, and friends who allowed me to have the time I needed to write and carryout this research. Thank you very much for your contributions and support towards me.

My sincere appreciation goes to my supervisor Ms. Babirye Nanteza Lucy for her intellectual guidance, support, and encouragement. I am very appreciative of your patience and motivation you have provided to me up to this stage of submission. I would also like to thank all the respondents especially the administrators at Uganda Martyrs University Nkozi and lecturers. I thank them for their acceptance to be part of this study.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

CSS	Cascading Style Sheet
D.V.C.A	Deputy Vice Chancellor Academics
E-education	Electronic education
GB	Giga Byte
HTML	Hypertext Markup Language
H.O.D	Head of Department
HR	Human Resource Manager
ID	Identification
MYSQL	My Standard Query Language
OS	Operating System
OCASFAS	Online Course Allocation System Academic Staff
PHP	Hypertext Preprocessor
RAM	Random Access Memory
UML	Unified Modeling Language
UMU	Uganda Martyrs University

ABSTRACT

This project is about the design and implementation of an Online Course Allocation System for Academic staff for Uganda Martyrs University to replace the currently used manual system. Course allocation involves the scheduling of a certain number of academic staff (lecturers) to teach courses over a definite period. It is usually done by the head of the department which takes into consideration the teaching load per week and contact hours for each academic staff according to their preferences. This process is costly and prone to human errors. Moreover, there is also a lot of repetitiveness were several lecturers can be allocated the same course. As well there are challenges in assigning the right course to the right lecturer. As a result, there are many allocation mistakes where some lecturers may be overwhelmed, while others would be underutilized. The project was therefore aimed at reducing the time and costs incurred during the course allocation process, as well as enhancing the work of allocating courses to the academic staff at this university.

A study was carried out to justify the relevance of the new system. The researcher followed a design-oriented research approach to collect and analyze the required data. Questionnaires, interview guides, and document analysis were used to collect the required data. The data collected was analyzed using frequency tables and the results of the analysis of this data were also presented. The analysis was aimed at investigating the various loopholes of the current system to come up with the functional, non-functional, and user requirements of the new system. During the study, it was found out that the currently used systems is too tedious and also prune to human errors hence a need to come up with a new system.

To solve the above problems, models representing the system were designed and implemented. Here several methodologies and tools were used. Dreamweaver embedded with HTML and PHP, UML, JavaScript, CSS MySQL, and Ms. Vision was also used through the planning, analysis, design, and implementation of the new system. The researcher used the prototyping technique to come up with the system which was tested and validated for the expected performance and ensure that the required requirements were met.

The proposed system makes it easier to update the schedule and have the changes effected and made conveniently accessible by all stakeholders where ever they are. It also provides quicker and more efficient services for the course allocation process hence saving time.

CHAPTER ONE

INTRODUCTION

This chapter explains the background of the study, problem statement, objectives, and scope of the study. It also gives details of the significance, and justification of the study.

This research is in the field of the e-educational process which can be defined as the use of information technology to enhance and improve on the quality of the teaching processes especially in higher institutions of learning (Sheikh, 2012). The management of the teaching process is an important aspect of the educational process (Galanc, 2011). Course allocation is one aspect of the teaching process. Course allocation involves the scheduling of a certain number of academic staff (lecturers) to teach courses over a definite period (Ekhosuehi, 2016). Scheduling is one of the most critical issues in the planning and managing of educational processes. Scheduling is usually done by the head of the department which takes into consideration the teaching load per week and contact hours for each academic staff according to their preferences. Coming up with a perfect course allocation schedule for the lecturers at the university level has been such a complex process for over years (Soyemi, et al., 2017). This problem has been seen in universities that have more than one campus with several programs that are inter-related by shared courses and lecturers (Siame, 2018).

Today, the use of information technology is vital in enhancing and improving the academic and learning processes whereby most academic institutions have adopted using automated systems to improve their learning processes. These systems have reduced the recurrent costs faced by different institutions. They no longer have to spend a lot of money on printing services and expansion of the storage areas for storing paperwork (Jeremy, 2012). This advantage has increased the popularity of automated systems because most people are looking for a way to improve the efficiency of their work while reducing costs. In this case, there was a need to adopt in technology to smoothen the course allocation and scheduling processes in universities whereby the online course allocation system would reduce clashes during the preparation of schedule for lecturers.

1.1 Background

Several institutional papers and reports have described the common challenges academic, administrative and workload allocation experienced by multi-campus universities compared to single-campus universities (Rudova, 2011). These challenges include delivering the same subjects, programs, and courses with equivalent quality on several campuses. According to Simon Perks (2013), developing an effective workload model takes time and effort. However, if done well it can provide valuable insight into academic workloads and yield tangible benefits for academics, departments and institutions alike. It can also, crucially, facilitate a more constructive dialogue between academic departments and the university administration.

According to Stensaker (2017)in Australia, Deakin University is a multi-campus university located in Victoria. This university had challenges with academic work allocation for the lecturers. The author noted that an individual academic staff member had a concern about his/her work allocation that was either excessive or unreasonable, the issue had to be discussed with his/her supervisor. Where it was not possible to resolve the concern by this means, the academic staff member would write to the Executive Dean of their Faculty or equivalent, if the staff member was not employed in that Faculty where he/she was seeking a work allocation review. The Executive Dean after 4 weeks of receiving a valid request identify issues being considered in the review would invite directly affected staff members. Then a work allocation review would be completed and a response provided to the staff member within two weeks of the application for a review. If the matter was not resolved by the Executive Dean of their Faculty, the staff member would then take his/her concerns to the Deputy Vice-Chancellor. Then the Deputy Vice-Chancellor would decide in two weeks and provide feedback for the pending work allocation review. This is a time-consuming and a resource wasteful task faced by this university during the course allocation process.

In a research conducted by Rauf, et al. (2018) in Malaysia, it was noted that the University of Selangor had a challenge of improper course allocations for the lecturers which was done by using software such as Microsoft Excel. In this university, each lecturer had a preferable course basing on the expertise field. Challenges came when a lecturer had been allocated to more courses as some were in higher positions and other lecturers were doing their research on the

time of their work. The manual methods that were used to allocate courses incurred a lot of mistakes and errors. This affected the lecturers' expertise field which they had done their majoring studies. Therefore, the university administration adopted to use a computerized system to handle the course allocation for lecturers' workload according to their preferred and expertise field. But this system considered the only allocation of courses to lecturers in a single campus university. Hence, there is a need to come up with a new system that can enable the allocation of courses to lecturers in multi-campus universities.

In Africa, University course allocation and scheduling was a very common problem. For instance, in Zambia, university administrators found challenges to allocate courses especially in the case where a university had two campuses running several programs that were inter-related by shared courses and lecturers, for example, Mulungushi University (Siame, 2018). In this case, they were using paper-based manual systems and Microsoft excel sheets to develop these schedules for the lecturers. This made the process of developing a course allocation or a schedule a very difficult task that could take human brains many hours to develop and it still contained errors (Siame, 2018).

According to Kiconco(2011), in Uganda, the course allocation process has been manually done on papers by most universities and then entered into Microsoft excel sheets thus in case of an error like time allocation errors, it is hard to correct. Due to this problem, lecturers are assigned to more than two courses on different campuses and sometimes may be assigned courses that may not be their expertise and high workload especially to high position lecturers. Therefore, using manual course allocation means that they could not get the preferred courses and a lot of issues have been developed due to this allocation method which has been conducted manually using software called Microsoft Excel (Rauf, Selvan, & Adnan, 2018).

Uganda Martyrs University (UMU) is a faith-based private University owned by the Episcopal Conference of the Catholic Bishops of Uganda. Uganda Martyrs University is located along the Equator at Nkozi, 80 km west of Kampala, the capital city of Uganda. Opened in October 1993 with 84 students and two academic departments, currently, it has 7 Faculties, 1 Institute, 3 Schools, 3 Directorates and about 5,000 students of whom about 1,500 are full-time residents on

campus. The remaining 2,500 and 1,000 participate in distance learning and part-time programs respectively at the university campuses in Kabale, Kampala, Masaka, Mbale, and Lira.

Currently, at UMU, course allocation for academic staff has been manually done whereby the lecturers submit their preferred courses to the heads of the department (HOD) who allocates courses (Dennis, 2011). There are different departments and, in each of them, there is a head of department who allocates courses to the lecturers. This is done based on the lecturer's preferences, contact hours and the position held. For instance, a teaching assistant should not be assigned more than 12hours and neither can he/she be assigned a masters course even though it may be in his or her preferences. The HOD schedules courses at different levels for example Data warehousing at Bachelors level, introduction to computer science at Diploma level, introduction to Information Systems at Masters level. A list of detailing the scheduled courses together with the number of lecturers within that department is compiled using Microsoft Excel and is then sent to the faculty dean for review. The faculty dean forwards the assigned courses to the Deputy Vice-Chancellor Academics(DVCA) for approval. Finally, DVCA sends the scheduled courses to the human resources manager who publishes the scheduled courses. However, lecturers usually complain that they are sometimes allocated to courses and contact hours which do not favor them to work at different campuses of this university. This is due to allocation mistakes and errors made during the course allocation process whereby some lecturers might exceed their teaching workload, while others would be underutilized. This means that the allocation process is always changing due to some allocation errors which might delay the learning process. So, if these errors are to be corrected, the whole process has to be done again which is time-consuming and a resource wasteful task. In addition to that, lecturers find difficulties to access the administrators to make allocation changes. And the extent to which these activities are carried out by the academic staff to meet the set standards determines the quality of learning and knowledge gained (Silva.D & Ribeiro, 2017). Therefore, this research was aimed at finding a solution to the course allocation problems faced by the head of departments and academic staff (lecturers) through the development of a viable online course allocation system for academic staff at Uganda Martyrs University. The proposed system would enhance the quality of courses allocated to the lecturers, reduce on the time taken to correct allocation mistakes.

1.2 Problem Statement

Multi-campus institutions should have a Course Load Allocation process that can ensure delivery of the same subjects, programs, and courses with equivalent quality on all campuses Rudova(2011). However, the time and effort taken during the process is not optimal. Additionally, this process is costly and prone to human errors. Moreover, there is also a lot of repetitiveness were several lecturers can be allocated the same course. As well there are challenges in assigning the right course to the right lecturer. As a result, there are many allocation mistakes where some lecturers may be overwhelmed, while others would be underutilized. As well some staff may be allocated courses that they are not well conversant with. This compromises the quality of delivery of courses in the different campuses.

To solve the above problems, an Online Course Allocation System for Academic Staff was developed. The system would ease the work of assigning courses to lecturers such that courses are effectively allocated without clashes. The proposed System would make it easier to update the schedule and have the changes effected and made conveniently accessible by all stakeholders where ever they are.

1.3. Objectives

This section details objectives of the research project. It gives both the major and specific study goals.

1.3.1. Major objective

This research project is intended to develop an online course allocation system for academic staff that can minimize time and costs incurred during the course allocation process, and also improve the quality of education.

1.3.2. Specific Objectives

i. To analyze and investigate the existing course allocation system to find out the requirements of the new system.
ii. To design an Online Course Allocation System for Academic Staff (OCASFAS)in order to come up with models representing the system
iii. To implement the OCASFAS in order to transform the design into a working system.

iv. To test and validate the new OCASFAS for the expected performance and ensure that the requirements are met.

1.4. Research Scope

The scope of the study includes the time scope, content scope and the geographic scope where the study was conducted.

1.4.1. Time Scope

The research was carried out in a period of seven months that is October 2019 to April 2020.

1.4.2. Content Scope

This research was aimed at finding out how the use of an automated system can help in course allocation for academic staff in the Computer Science and Information Systems department of Uganda Martyrs University. The system developed can enable the:

i. Head of the department to register the lecturer profile, course details and store the details in the database. The head of department can also allocate courses to lecturers and generate a schedule report per semester, per campus or per staff.
ii. Faculty Dean to log onto the system, review allocated courses to the academic staff in that faculty and also comments on the scheduled courses.
iii. The deputy vice chancellor academics to log onto the system, view courses, programmes, lecturer profile, schedule reports from the departments comment on the scheduled courses and finally approve the scheduled courses.
iv. Human Resource Manager to log onto the system, view scheduled courses, lecturer profile details, and comments for quality assurance and finally publish the scheduled courses to the academic staff.
v. Academic staff to log onto the system, view the assigned courses, submit preferred courses and teaching areas personal profile details, comment on the assigned load.
vi. Academic registrar to log onto the system, view reports on registered Programmes, courses, schedule and academic staff in the department.

1.4. 3. Geographical Scope

The researcher used the course allocation processes of Uganda Martyrs University(UMU). UMU is located along the Equator at Nkozi, 80 km west of Kampala, the capital city of Uganda. The research was limited to the course allocation processes of the Computer Science and Information Systems department.

1.5. Research Significance

The research could be a source of literature for future researchers.

This study is significant to many stakeholders, including, the lecturers, administrators in such a way that an automated system can reduce the time involved in allocating and re-allocating for the courses which led to delays in the academic work. According to Kiconco(2011), an online course allocation system can provide better services to lecturers, faculties/departments and administration by providing meaningful, consistent and timely information thus promoting the academic services to reduce the mistakes and errors made during the course allocation process. Besides, the system can also enable stakeholders' access to information in real-time and in an interactive manner through a web-based system thus eliminating delays in getting feedback (Kiconco, 2011).

Rauf, et al (2018) noted that the OCASFAS can also promote efficiency by converting paper processes to an electronic form hence it is more flexible than the traditional manual methods. If any lecturer or management has requested to add or drop the courses, the system is online-based where the lecturers and administrators can access the system anytime and anywhere (Rauf, et al., 2018). Therefore, the administrators can make changes in the allocation part by adding, deleting or change the lecturer assigned to the course.

The system can accurately allocate courses and printing out an output of a list stated for the required information concerning the lecturers' workloads basing on their preferences in the department of computer science and information systems at Uganda Martyrs University.

CHAPTER TWO

LITERATURE REVIEW

2.0 Introduction

This chapter presents the review of the related literature on course allocation system for lecturers in higher institutions of learning. A literature review involves the review of the related course allocation systems (web-based application) to establish what other scholars have covered the problem. This helped the researcher to identify the gaps in the literature and how these can be used to solve the problem under study and finally a conclusion.

2.1. Universities and Education Management

According to the Zina and Houhamdi (2017), a university was an educational institution for higher learning that typically includes a college that is composed of a set of departments, then a department offers one or more programs, a program has a study plan composed of a set of courses, a course is composed of one or more sections. The course is taught by an instructor who specialized in that course.

According to Scott, et al. (2007) multi-campus universities are defined as institutions with three or more campuses that geographically dispersed and campus each with a substantial student load. In the multi-campus model, academic services, resources, and support facilities are not concentrated on any one site, but are typically dispersed and often replicated across campuses and managed through a centralized administrative system. Kasozi (2003) also describes universities as centers of higher learning where knowledge is created, preserved and disseminated.

2.1.1 Education Management

According to (Rauf, et al., 2018) education management refers to the improvement of the learning and teaching processes. It also provides the idea of management education flow of an automated system rather than transforming it. He further argues that information technology is being used to automate the delivery of information to users. It also provides the evaluation of

sustainability in the management of education which is providing consistency to integrate the level of education. With the above-mentioned arguments, education management should be upgraded to enable the flow of teaching and learning through the design of the education management system. In conclusion, the literature review will be conducted on the previous education management systems which improvise the education management from the current situation to be automated to help the users (lecturers) receive information concerning the allocated courses in a timely way and with ease.

2.2. Course Allocation System

According to Ekhosuehi (2016) Course allocation involves the scheduling of a certain number of academic staff (lecturers) to teach courses over a definite period. This scheduling activity is done by the university administrators that include the head of departments, and the deans who schedule, courses and programs to the lecturers. The major emphasis of the university course allocation is to find a method to allocate whole events that are to happen per semester to fix predefined timeslots for each program (Siame, 2018).

2.2.1. Benefits of Course Allocation System

Over the years, there has been an increased reliance on digital information and the technologies that support it in virtually every aspect of the educational, research and administrative processes (T.D.Nguyen, 2018). Some benefits of digital information are the operational speed, accuracy, convenience and better service to users in such a way that the systems produce information for recipients who may need it sooner to make decisions that control their operations. In this study, information technology should be adopted in all business operations especially for higher institutions of learning to reinforce educational processes such course allocation system. An online course allocation system for academic staff eases the work of assigning courses to lecturers without clashes such that teaching workload is delivered to lecturers in time. The system can also be accessed anytime and anywhere by both lecturers and administrators hence saving costs and time. The system would ease the work of updating the course schedule and it is conveniently accessible by all stakeholders where ever they are. It provides quicker and more efficient services for the course allocation process through an automated system hence saving time. The course allocation system is a more vital important part because it minimizes the

courses given to the higher position lecturers to avoid overwhelming them (Rauf, et al., 2018). All lecturers are having their preferable courses where their expertise field plays the most important role in that issue. An automated system serves as a reference file for the university administrators and lecturers since the course, contact hours, campus and staff in charge are all centralized into the database hence flexibility when there is a need (Rauf, et al., 2018). Therefore, the system can reduce the time needed for the processing of departmental courses and attend to other subsequent information needed by the management system.

2.3. Overview of Systems

A system is a collection of independent but interrelated elements or components organized in a meaningful way to accomplish an overall goal (Hayajneh, 2007). The function of any system is to convert or process materials, energy, and/or information (inputs) into a product or outcome for use within the system, or outside of the system (the environment) or both. Inputs include raw material, data, and resources processed to produce the outputs of the organization. Throughput is the processes used by the system to convert raw materials or data (inputs) from the environment into products or services that are usable by either the system itself or the environment. The output is the product or service which results from the system's throughput or processing of technical, social, financial & human input.

Finally, feedback is information about some aspect of data or energy processing that can be used to evaluate and monitor the system and to guide it to more effective performance. Therefore, the effectiveness of a system can be measured on its efficiency to provide feedback to the users (information) with ease and on time (Zina, 2017). This efficiency includes services of high quality with responses that are fast and provided when they are needed or requested. Concerning the study, there was a need to use systems, tools and modern technology to streamline educational management and services delivery in universities. This indicates the importance of information in all spheres of life and among other things, it is an essential ingredient in planning, decision making and implementation of different activities by people and management staff of organizations likes universities. Hence automated systems can enable decision-makers to have quick and easy access to information that is to say the administrators of universities whenever there is a need thus reducing delays to make decisions.

2.4. Review of literature related to the current system.

The strengths and weaknesses of the existing and developed systems are discussed. Also, the way how these systems work is also discussed below;

2.4.1 The University of Selangor course allocation system

This is a web-based system developed to ease and smoothen the course allocation process by printing out a report of a list stated for lecturers and their preferred courses.

The university Selangor was having a constraint on course allocations for the lecturers by using manually software such as Microsoft excel (Rauf, et al., 2018). This affected the lecturer's expertise field which they had done the majoring studies. In this university, every lecturer was having a preferable course to teach and some lecturers were also doing their research on the time of their work. Due to the manual allocation of courses, it affected their research timing. Some lecturers were also in higher positions given more courses to lecture although they were to be given fewer courses to teach. Due to this happening in this university, they decided to develop a system that can handle the allocation of the courses for the lecturers according to their expertise and their position.

2.4.1.1 Strength of Selangor course allocation system.

The system minimized the courses given to the higher position lecturers. The system was also user-friendly which eased the work of the administrators (Head of Department) to provide the lecturers their preferable courses. It also authorized the administrators to make changes in the system by adding new courses if there is a need.

2.4.1.2 Weaknesses of Selangor course allocation system.

The system cannot handle a different curriculum that is it cannot handle courses and programs at different year levels. The system a lecturer cannot comment or send feedback when there is a need for reallocation. It does not restrict certain courses to academic staff according to their positions and grade levels.

2.4.2 Texas A&M University Classroom allocation system

This is a web-based system that was developed to help the staff of any university to assign classrooms for a given semester.

At Texas A&M University, administrators had problems with the handling allocation process. They were using a manual procedure which was time-consuming and tedious as well as prone to human errors (Fall, 2016). Compounding this problem, there was no procedure in place to compare alternate allocation strategies. Therefore, in most cases, the allocation of classrooms was done basing on previous semesters allocations as well as the expertise of both the representatives and the Registration staff. Hence a web-based tool was created to help the staff of the registration office overcome this problem manual classroom allocation. It also gave a better user experience and reduced the time taken by the manual process and also eliminated the possibility of human error.

2.4.2.1. Strength of Classroom Allocation System.

The system monitors all classrooms on campus taking into consideration the details about those rooms.

It automatically recommends effective pairing of course to classroom pairing with the courses offered for a given semester.

This system takes into consideration, the most essential elements such as the classroom size, course capacity and preferred building of a department.

2.4.2.2. Weaknesses of Classroom Allocation System.

The system cannot support the allocation of courses to several campuses. It does consider the personal choice and comfort of the respective faculty. The system cannot generate an automatic email to notify the faculty members of the allocated courses and rooms. Hence there was a need to develop a new system (extension) that integrates these weaknesses and limitations.

2.5. Conclusion

In conclusion, therefore, according to the literature reviewed, the researcher identified that a manual system of allocation of courses and contact hours usually wastes a lot of time, delays other departmental duties that are important, high costs of data processing which results in the inefficiency of course allocation. Due to those mentioned problems, therefore, automation of departmental course allocation system for the lecturers should be adopted to reduce the workload of the staff especially the Head of Departments. Also, the importance of automated systems is highlighted this explains why organizations should adopt technology thus automate their business processes.

CHAPTER THREE

RESEARCH METHODOLOGY

3.0 Introduction

This chapter mainly explains the various methods and techniques that were used in the acquisition of the necessary data which was later analyzed to find the desired requirements for the system development. It goes on to explain which tools were used and applied in the development of the system.

3.1 Research Design and System development model

The research design provides overall guidance for the collection and analysis of data about the study. The researcher used descriptive and cross sectional research designs. The descriptive research involved describing the process through which the course allocation system can be used by the users and Cross sectional in choosing the respondents from which data was to be gathered as well as the research questions that were to be answered by the respondents. These were used to argue for the relevance of the new system and the evaluation of the different versions of the system.

The researcher chose Prototyping Model in order to get the product in time. A prototype is usually an initial version of a system used to demonstrate concepts and how they interact indicating even the most appropriate design options. Sommer, (2011) define prototyping as an iterative process of systems development in which requirements are converted into a working system that is continually revised through close collaboration between an analyst and the system users. The researcher used this method in order to minimize development costs and allows the developer to receive user feedback and then rework on the necessary changes until an acceptable prototype is finally achieved from which a complete system can now be developed. The model also enabled the researcher to prevent major design and functional errors Such errors were identified earlier and then corrected. This helped the researcher in ensuring improved design quality, improved system usability, improved maintainability and reduced development effort hence meeting user and system requirements.

The phases of the prototyping model are depicted in figure 3.1 below

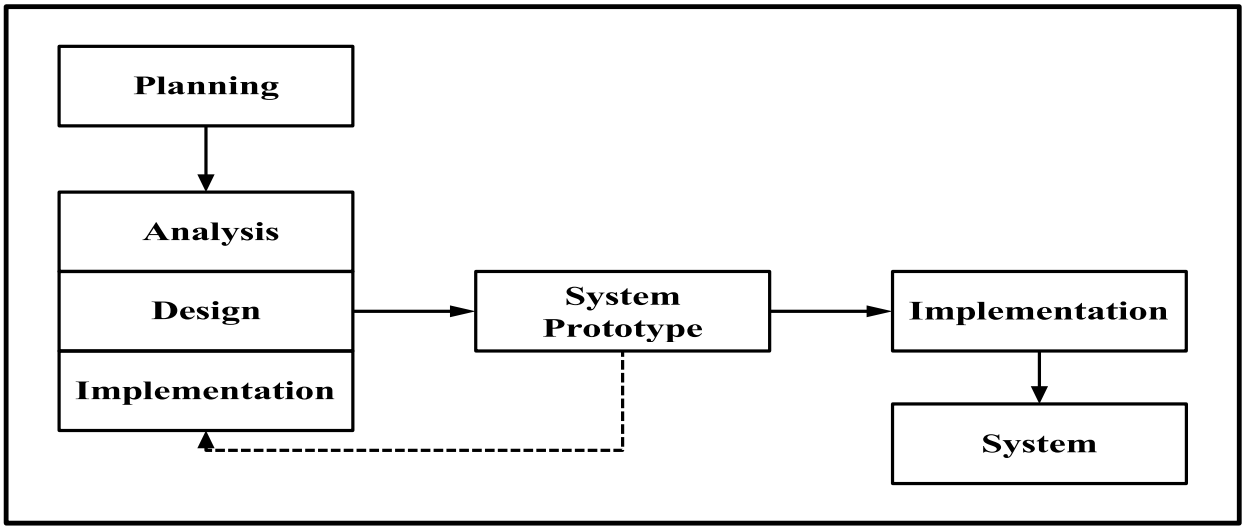


Figure 3.1: Prototype Model

3.1.1 Planning Stage

At the planning stage, the investigation of the problem was done and a feasibility study was carried out for the proposed system. This involved checking whether the proposed system was realistic in accordance to researcher's capability. This involved identification of the study area, target population, the sample size was estimated, sample selection and data collection techniques.

3.2 Area of study

The researcher used the course allocation processes of the department of Computer Science and Information Systems, Faculty of Science- Uganda Martyrs University. The researcher used Nkozi and Masaka campuses as a case study because at these campuses lecturers complain that they are sometimes allocated to courses that do not favor them to travel and teach at different campuses of this university.

3.3 Target Population

According to Dunn (2009) Population refers to a complete set of objects, cases, and individuals that the researchers intend to generalize the research results. Thus, it refers to the collection of elements, people among others that can be used to investigate a particular fact or situation.
However, due to the large sizes of populations, researchers often cannot test every individual in the population because it is too expensive and time-consuming (Dunn, 2009).

In this study, the targeted population was 30 people constituting university administrators, for example, academics dean (1), head of department (1), system administrator (1) and academic staff (27) in the department of Computer Science and Information Systems at both Masaka and Nkozi campuses respectively that were to be interviewed and given questionnaires to fill and thereafter, analyze the collected data. Since the new system is to be used by the administrators mainly and academic staff during the course allocation process and operations like adding, deleting, searching, updating the data in the system.

3.3.1 Sample Size

The sample size was calculated scientifically using Slovin's formula. This formula was used because it's a random sampling technique formula.
The Slovin's Formula is given as follows:
n = N/ (1+Ne2), where n is the sample size, N is the population size and *e* is the degree of freedom which is equal to 0.05 given 95% confidence interval.

Table 3.1: Showing sample size.

Population Category	Category Size(N)	Sample Size(n)
Academic Staff	27	25
University Administrators	3	2
Total	30	27

3.3.2 Sampling Technique

The sampling technique that was used to sample some of the university administrators such head of department, systems administrator and academic staff at Uganda Martyrs University Nkozi

was simple random sampling. In simple random sampling each member of population is equally likely to be chosen as part of the sample. It has been stated that "the logic behind simple random sampling is that it removes bias from the selection procedure and should result in representative samples" (Taherdoost, 2016).

3.4 Data Collection Method

In this section, all the methods and tools that the researcher used to collect data are documented. Data for the proposed system was collected using the following methods;

3.4.1 Questionnaire Method

Questionnaires were used to gather information from the academic staff i.e. lecturers, assistant lecturers and teaching consultants. This method was selected to enable the research handling a large number of participants (Kakinda, 2010). The researcher designed questionnaires which were distributed to the academic staff in order to obtain their views and gain insight about the existing course allocation system for the academic staff and an online course allocation system for the academic staff that was to be developed. The researcher provided both closed and open-ended questions closed-ended questions are structured such that a respondent has to choose from the predetermined and open-end questions are those questions that are structured to let the respondents write down thoughts in the space. The questionnaire is attached in ***Appendix A***.

3.4.2 Interview Method

An interview is a data collection method where the interviewer asks the interviewee guided questions about the topic of study through a face to face conversation. The researcher designed questions to ask respondents and the interviews were mainly semi-structured. This was in-depth and among the people interviewed were the head of the department, systems administrator and the faculty dean. The researcher conducted interviews because it was the convenient method to collect data from the university administrations who had tight schedules to ensure if there was a need for the new system, therefore firsthand information was obtained. The interview guide is attached in ***Appendix B***.

3.4.3 Documents Reviews

The researcher also reviewed the relevant documents, guides and laws governing an online course allocation system for the lecturers that have been used for past years. The researcher also made a study on business processes that were to be used to come up with accurate allocation of courses, records storage, and retrieval. The main source includes journals about an online course allocation system for lecturers and internet sources. Document analysis has been proven to be an extremely valuable alternative source of data, especially where there is a lack of access to research subjects (Bryman, 2007).

3.5 Data Analysis

The collected data was analyzed to determine the system and user requirements to come up with design models for the new system after studying the current system. The key data collected were categorized, coded, summarized and displayed in Microsoft excel as frequency tables interpreted to determine the requirements for the new system. Frequency tables were used because they save time, inexpensive to analyze findings and ability to provide accurate results (Saunders, 2007).

3.6 Design Phase

The purpose of this stage was to create a technical solution that satisfies the functional requirements for the system. At this point, the information gathered during the analysis phase was used to formulate models that represented the solution of the system. Also in this phase, the researcher designed logical, functional and physical design of the database, user interfaces, system architecture, network, system security and program design which were used later in the implementation phase. Here, the researcher followed an object-oriented approach to design and used the different notations in the Unified Modeling Language to produce detailed designs. The object-oriented approach enabled the researcher to break down a complex system into smaller, more-manageable modules, work on the modules individually, and easily put the modules back together to form an information system.

3.6.1 System architectural design

Architecture was designed to provide a blue print of the OCASFAS system. The architecture represents a broad structure of the new system. Architecture involves the software tools that are used to construct the system such as modeling tools, databases, compilers and packages (Wiley, 2015). In the technical architecture design, the software can be broken down into major components that cover items such as the user interfaces, storage and detailed processing. This involves the details of architecture and also the details of how individual components of the system are constructed.

3.6.2 Modeling

Microsoft Visio as a modeling tool was used to come up with logical and physical designs of the system. This application has a powerful template that allows for simple brainstorming and recording of ideas.

The UML is the current standard for programming in object-oriented programming languages (Wiley, 2015). UML was used to come up with process diagrams and class diagrams. When creating classes and other objects with relationships between each other, UML was used to visually describe these relationships. UML also helped the researcher to plan a program before the programming took place. Object Oriented diagrams such as use case diagrams, class diagrams, activity diagrams and sequence diagrams were used by the aid of Microsoft Visio software tool. The use case diagrams were used to demonstrate the sequence of operations that the user can perform with different modules of the system along with the associations. The activity diagrams were used represent the various states of objects in the system modules and the modes of relationships between the various action states of the objects. Sequence diagrams are the interaction diagrams that were used show how objects operate with one another and in what order. The class diagrams were used to show the associations between different classes with objects of the system and helped to provide the graphical notations for representing such data models.

3.7 Implementation

This section explains how the system was implemented using different tools and technologies.

3.7.1 Dreamweaver

This is a system development tool that enables programmers to build complex websites using Html, JavaScript, and server-side programming languages. The researcher used this as a text editor and a code generator to design and develop the proposed system because it simplifies the construction of interfaces and connecting them to the server on which they run.

3.7.2. MYSQL (My Structured Query Language)

It is a relational database management system equipped with an interface for application programming in PHP by (Jerzy,2013). Relational database design was used to design the database. A relational database management system is an excellent tool for organizing a large amount of data and defining the relationship between the datasets consistently and understandably. The researcher used it to create databases because of its scalability and easy to connect with user interfaces built in PHP scripting language hence can accommodate large amounts of data and also provide high-level security for stored information.

3.7.3 PHP (Hypertext Processor)

It is an open-source server-side scripting language designed for web development to produce dynamic web pages. PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document. PHP was chosen as the scripting language for the online course allocation system for lecturers because it is compatible with most web servers and operating systems. It can be used with many relational database management systems.

3.7.4. CSS

It stands for Cascading Style Sheet. This was used by the researcher to define the look and feel of the system pages. It is preferred because it allows a developer to separate content from design as it takes care of the layout while HTML works on the content (Bhaumik, 2015).

3.7.5. HTML

It's the predominant markup language for web pages which is written in the form of HTML elements consisting of tags, enclosed in angle brackets within 22 the web page content. It was used to create forms and tables. The browser does not display the HTML tags but uses the tags to interpret the content of the page.

3.7.6 JAVASCRIPT

It is a scripting language that enables web authors to design interactive sites. This scripting language enabled the designer to spice up the system with dynamic content and to make the system's functionalities active (Jerzy, 2013). It was used by the researcher to make the proposed system's functionalities active through the generation of dynamic content.

3.7.7 Bootstrap

According to Bhaumik (2015), bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites. The researcher used bootstrap because it is completely free to download and use.

3.8 System Testing

Testing and integration is the process of ensuring that the entire system successfully works together as a single entity.

A combination of testing approaches that were used include unit, integration and system testing to ensure that system program, database structure and performance requirements are working according to the specifications. HTML, MySQL, and PHP scripts were run independently to test the system's functionalities. Then integration testing was carried out to test the compatibility of the various system components. HTML was used to create forms, tables and user interfaces. MySQL was used to create and connect relational tables to the database and it was tested on Firefox. PHP was used to process queries. To ensure data integrity, user input is checked and in case of wrong inputs, the user will receive a notification and prompted to enter the correct data. For example, a non-user of the system cannot access it because it requires a username with a

corresponding password and the incorporation of HTML with PHP scripts to enable the enhanced validity of the system. Finally, system testing was carried out under different running environments like windows 7,8,10 and Ubuntu.

3.9 Conclusion

This chapter looked at data collection techniques that were used to attain the information needed to design the proposed system and also system development.

CHAPTER FOUR

PRESENTATION OF RESULTS, SYSTEMS ANALYSIS AND DESIGN

4.1 Introduction

This chapter presents findings from the data collected which closely look at the current system with strengths and loopholes. It also involves explaining the current system processes (inputs and outputs) and the users that have access to it. The results from the questionnaires were analyzed using Microsoft Excel, data from interviews and documents were discussed and presented in this chapter. The data collected is analyzed to find out the requirements for the new system as required by the various stakeholders and the system design is also explained.

4.2 Analysis and presentation of data

The findings from the data collected and the results of the analysis of this data are presented. The analysis was aimed at identifying the various loopholes of the current system to come up with the functional, non-functional and user requirements of the new system. To acquire information concerning the functioning of the current system, the researcher collected data using several data collection tools which included interviews and questionnaires. The researcher made appointments with the university administrators who are concerned with the course allocation process and a total of 2 people were interviewed.

The researcher also printed out 25 copies of questionnaires and these were distributed to several lecturers, consultants, and assistant lecturers. The respondents were asked about how easy is it to have access to a computer connected to the internet; the results are shown in the table 4.1 below:

Table 4.1: Accessibility to a computer connected to the internet.

Access to a computer	**Number of Respondents**	**Percentage(%)**
Easy	22	81.4
Very easy	3	11.2
Not easy	2	7.4
Difficult	0	0
Total	**27**	**100**

The results from the table above indicate that the biggest number (81.4%) of respondents have access to a computer connected to the internet. This implies that most of the staff will be able to access the online system if it is deployed.

The respondents were also asked whether there is an urgent need to change from the current system to an online computerized system for allocating courses to lecturers and avail of access to the scheduled information. The results are shown in the table below;

Table 4.2: Opinions on the development and implementation of a new computerized system for allocating courses to academic staff

access to a computer	**Number of Respondents**	**Percentage(%)**
As soon as possible	19	70.3
Soon or later	8	29.7
No	0	0
Indifferent	0	0
Total	**27**	**100**

According to the table above, the biggest number (70.3%) of respondents' opinion was that a new computerized system for allocating courses to academic staff at Uganda Martyrs University should be adopted hence the need to develop a new system.

4.3 System Analysis and Design

System design is the process of defining the architecture, components, interfaces, modules and data for a system to meet specified requirements. The system design contains logical design and physical design where the logical design describes the structure and features such as output, input, procedures and the database. The physical design describes a working system in details.

4.3.1 Description of the existing system

Uganda Martyrs University uses a manual system (Microsoft excel and word applications) for allocating courses to lecturers. In the university, there are several different departments and, in each department, there is a head who allocates courses to the lecturers for that particular department. The head of department schedules courses based on the lecturers' preferences and sends the details of the scheduled courses together with the number of lecturers within that

department to the head of a faculty for review. Then the head of a faculty forwards the assigned courses to the Deputy Vice-chancellor Academics for approval. Finally, the duty vice-chancellor sends the scheduled courses to the human resources manager who publishes the scheduled courses.

Table 4.3 The current business processes and the responsible person

Current Business Process	Responsible Person
Allocation of courses	Head of Department
Reviewing the allocated courses	Head of Faculty
Approval of the allocated courses	Deputy Vice-Chancellor Academics
Publishing scheduled courses	Human Resources Manager

4.3.1.1 Current Records Storage infrastructure

No integrated course schedule database allows view, retrieval, simultaneous sharing of data, and auto-generation of reports. Currently, university administrators are using standalone applications (Microsoft Word and Microsoft Excel) to store scheduled data. Therefore, to make the allocation process easier, hence the need to have the OCASFAS database server in the university server room and the database always available online which can enable the computerized devices to access the system on the network.

4.3.1.2 Weakness of the existing system

Basing on the results from the analysis, the current system was characterized by the following problems:

1) The system in place does not automatically allow update of lecturer profile, adding a course to year level, adding, deleting and changing of lecturer schedule. This is fed in manually which is tedious and time-consuming. It also leads to delays in formulating schedules to lecturers every semester.

2) The manual system does not allow flexibility, for instance, it does not facilitate the addition and deletion information but rather every time an update is needed, a new document is created which increases paperwork and duplication.

3) The current system is not flexible which is costly for lecturers who have to physically come to the main campus every time there is a need for information.

4) There is no centralized database to hold scheduled course allocation information. A large volume of information available makes it difficult to collect, organize and link documents using the manual system.

5) The system cannot automatically generate reports when required by the administrators.

6) There is too much paperwork which has led to storage space problems.

7) Difficulties in retrieving information when needed due to the numerous files created in the process of keeping updates.

8) The security of the lecturer's records is not guaranteed in the current system.

9) Difficulties in sending every lecturer's assigned courses and workload by the administrator due to disparate systems.

10) Stand-alone office packages process and store data leading to a lot of duplication of the same files kept by various offices and departments.

4.3.2 The Proposed System

The proposed Online Course Allocation System for Academic Staff is meant to streamline the processes of allocating courses to the academic staff and storage of the course scheduled records to ease the work done by the administrators and enable easy access to this information on the time at Uganda Martyrs University.

4.3.2.1 Advantages of the proposed system

i. It eases the work of assigning courses to lecturers without clashes such that teaching workload is delivered to lecturers in time.
ii. The proposed system makes it easier to update the schedule and have the changes effected and made conveniently accessible by all stakeholders where ever they are.

iii. It provides quicker and more efficient services for the course allocation process hence saving time.

iv. It be can accessed anytime and anywhere by both lecturers and administrators hence saving costs and time.

4.3.2.2 Use Case Scenario for the proposed system

Use-case scenarios were used to develop use case diagrams following a sequence of events as shown below.

Table 4.8: Showing Use Case Scenario

Use Case Scenario	Use Case Description
Use case Scenario A: Creating User accounts	To access the core modules, the system administrator is prompted to log in before performing any tasks. Once logged in he/she can register all other users. The details recorded include: User_id, User_role, First_Name, Last_Name, Email, Phone_Number, Username and Password.
Use case Scenario B: Adding programs and courses.	Once the head of department has successfully logged in, the system allows him to add a new Programme by clicking on the link —" Course Structure". This is prompted by a click on the link "Add Programme". The system will also allow him to add a new course and adding a new teaching area.
Use case Scenario C: Adding a new academic staff (lecturer).	Once the head of department has successfully logged in, he can add a new academic staff by clicking on the link "Academic Staff". This is prompted by a click on the link "add academic staff". He can also edit or delete academic staff.
Use case Scenario D: Adding a new campus.	Once the head of department has successfully logged in, he can add a new campus by clicking on the link "Campus". This is prompted by a click on the link "Add Campus".
Use case Scenario E: Adding a new academic year.	Once the head of department has successfully logged in, he can add a new academic year by clicking on the link "Academic Year". This is prompted by a click on the link "Add Academic Year".
Use case Scenario F: Adding a new schedule.	Once the head of department has successfully logged in, he can add a new schedule by clicking on the link "Schedule". This is prompted by a click on the link "add schedule". He can also edit or delete schedule.

Use case Scenario G: Checking academic staff profile, allocated courses.	The lecturer who wants to check their profiles and allocated courses, the system prompts the academic staff to login. He/she will be authenticated by the username and password to access his/her account.
Use case Scenario H: Adding a preference and send a comment.	Once the academic staff has successfully logged in, he/she can add his/her preferred courses by clicking on the link "Preference". He can also send a comment by clicking on the link "Add Comment".
Use case Scenario I: View Reports	The system produces report on the scheduled courses, Programmes records, Courses records and academic staff records. These reports are viewed by respective head of department who may want to edit or delete allocated courses, Academic Registrar who may want to view the registered programs and courses, Faculty Dean who may want to review the scheduled courses, Deputy Vice Chancellor Academics to approve the allocated courses, Human Resource who may want to publish the scheduled courses and the systems administrator who may want to add or delete a user.

Source: Research Findings

4.3.2.3 Use Case Diagram for OCASFAS

The use case diagram demonstrates the sequence of operations that the user can perform with different modules of the system along with the associations. The user can interact with the system after a successful login as shown in the figure 4.1 below;

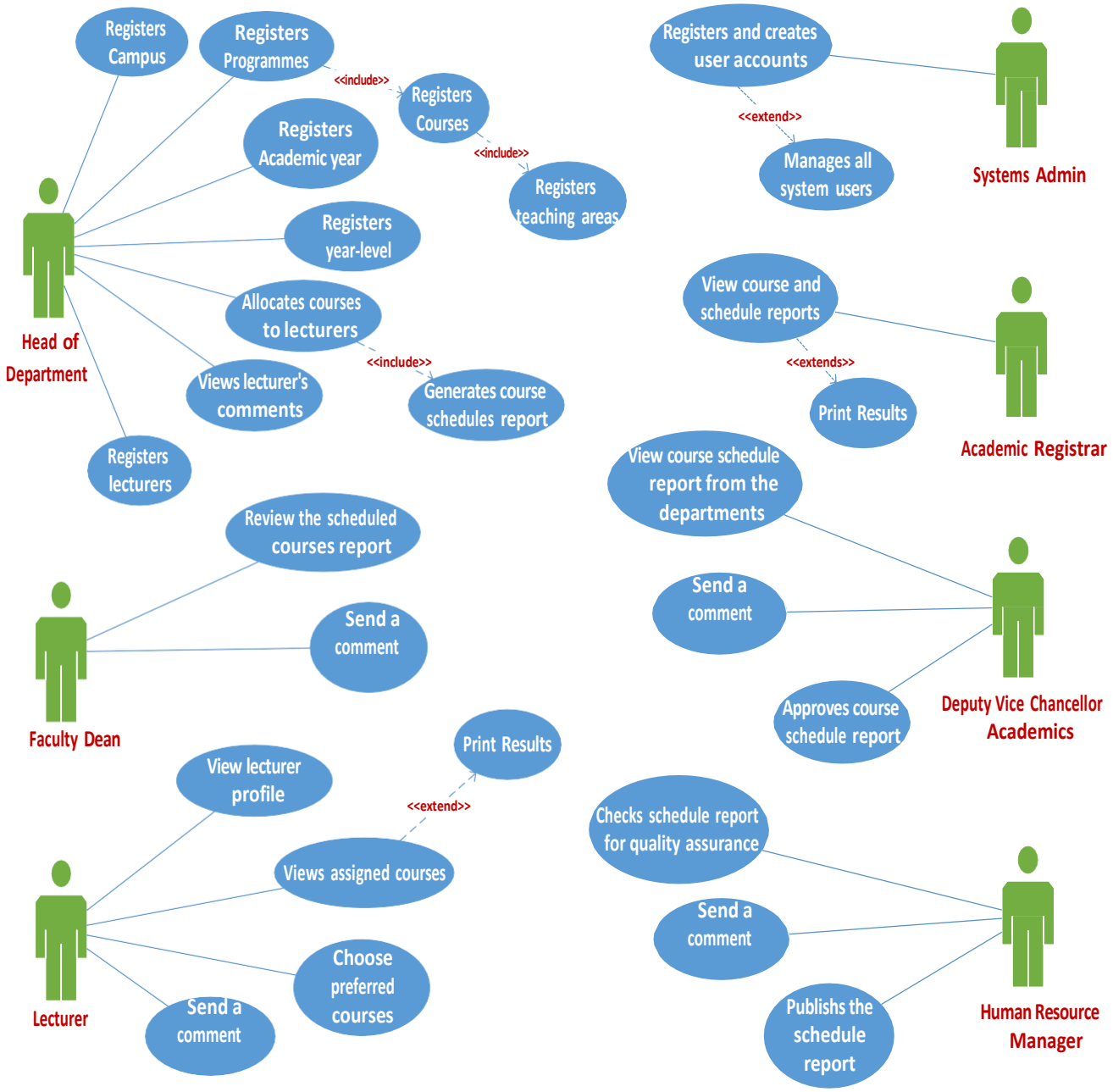


Figure 4.1: Showing the use case diagram for OCASFAS.

4.3.3 Requirements of the proposed system

The features and behavior of a system are described as shown below;

4.3.3.1 User Requirements

User requirements refer to the various requirements that users require from a system. The requirements below were gathered from the respondents during data collection. They were asked what they expected of the system to do and the following were the findings. The user requirements for the online course allocation system for academic staff are divided into two categories; functional requirements and non-functional requirements.

Table 4.4: Functional requirements

Module	**Functional Requirement**
Academic Staff(lecturer)	**The system shall allow the academic staff to:** i. Login with valid usernames and passwords to gain access to the dashboard. ii. Register preferred teaching areas and courses. iii. Comment on the assigned courses. iv. View their assigned courses, allocated campus, profile and online on login.
System Administrator	The system administrator can add, delete and manage users of the system who are given different access rights depending on their status and roles.
Head of Department	The head of the department should (add, edit, delete, update course information, lecturer profile, allocating courses, generate course schedule).

Module	Functional Requirement
Faculty Administrator	The faculty administrator(dean) should be able to (add, edit, delete, update course information and review course schedules).
DVCA	The deputy vice-chancellor academics should be able to view allocated courses, year levels, view course details, view lecturer profile and approval of the course schedules.
Human Resource Manager	The human resources manager should be able to view course schedules, lecturers profile for quality assurance and publish the Scheduled courses.
Academic Registrar	The academic registrar should be able to view reports on schedule, Programmes, academic staff and course unit in the department for proper records management.

4.3.3.2 System requirements

System requirements can be defined as the requirements written from the developer's perspective on what a system will do. These may include steps undertaken by the system when scheduling. Course allocation involves the process of assigning a set of academic staff to a given set of courses taking into account a set of considerations. The considerations for scheduling algorithm were described as shown below;

The algorithm should consider the qualifications of the staff for example teaching assistants should not be able to teach masters courses. It should consider teaching areas selected by the academic staff. The algorithm should also put into consideration the available programs at different campuses in a particular academic year. The algorithm should consider the number of credits, the maximum teaching load which is usually 12 hours per week meaning that the total number of credits should not exceed 12 per staff. If it does, then that should be indicated as extra

load. Therefore, the extra load should be only allocated if all the qualified staff has been allocated courses.

Before scheduling, a staff would select the teaching areas from the lecturer's account. Then the HOD would select the academic year, program, year level, department, campus, and semester and click generate schedule.

Evolutionary Algorithm

Evolutionary algorithms are general purpose search procedures based on the mechanisms of natural selection and population genetics (Khaled .M, 2004). This algorithm can be used to solve the required constraints and considerations during allocation.

We have 2 Sets that is Courses set and Staff set.

Let S be Staff

Let C be Courses

$S = \{S_1, S_2, S_3, S_4 \ldots\ldots\ldots\ldots\}$ Staff set

$C = \{C_1, C2, C3, C4 \ldots \quad\}$ Courses set

Let S_1 prefer $C_1 > C_2 > C_3 > C_4$

S_2 prefer $C_1 > C_2 > C_4 > C_3$

S_3 prefer $C_3 > C_1 > C_2 > C_4$

S_4 prefer $C_4 > C_2 > C_1 > C_3$

Let Staff want 3 courses and the course capacities be 3.

S_1 is allocated C_1, C_2 and C_4

S_2 is allocated C_1, C_3 and C_2

S_3 is allocated C_2, C_4 and C_3

S_4 is allocated C_4, C_3 and C_1

4.3.3.3 Non-functional requirements

Efficiency

The system should able to provide the correct results when correct input has been fed into it and prevent the users from making mistakes by validating their input. It should also display informative alerts and warning messages that trigger the user's attention in case of wrong or incorrect inputs.

Validation of Entries: The system should be able to validate data that has been entered by the users. The integrity of data is the ability to prevent the system from accepting wrongs entries.

Response times

The system should have a quicker response time. This was achieved through the use of light graphical illustrations to decrease on the screen refresh times and loading speed.

Reliability: The system shall be reliable by being available 24 hours. The new system has enough capacity to accommodate all the required information for the system to function normally. The information should display the attributes of accuracy, consistency, and reliability of content and processes.

Security and Accessibility: The system must provide required access levels restricted to appropriate users for security purposes because the data held by the system is so important to the university management.

Performance: The system performs data manipulations according to commands feed-in by the users in the shortest time.

User Friendly: The system should be user friendly and easy to understand with a simple logical interface that can be understood and followed easily. Therefore, the system has been developed with good user interfaces to enable users to interact with it easily.

Growth Requirements

The system should be in position to handle the growth and extension requirements of the university. The system should allow growth, extension and upgrading.

4.3.3.4 Security Requirements

The users of the system require assurance about the security features of the new system. This can be achieved by providing several access levels to different users of the system whereby it should allow varying privileges to users according to their ranks and roles.

4.3.3.5 Hardware Requirements

The hardware requirements of the proposed system are described as shown below;

Table 4.6: Specification for the server machine

Hardware	**System requirements**
Processor Speed	2.5GHz
RAM	4GB RAM and above
Hard Disk	500GB and above

Table4.7: Specifications for the client machine

Hardware	**System requirements**
Processor Speed	1.1GHz
Hard Disk	80GB and above
RAM	512MB RAM and above
Network	Wireless connection and Ethernet wire cable
Internet Connection	ADSL Broadband: 4.5 Mbps download 09 Mbps upload

4.3.3.6 Software Requirements

Software requirements can be described as a list of programs that are required to operate or develop the system or a program. The table below shows the required software requirements for OCASFAS.

Software Requirements	
Operating System	Windows family
Application Server	Xampp v3.2.2

Web designing languages	HTML5,CSS6 and Bootstrap
Scripts	JavaScript
Server side Script	PHP version: 5.6.30
Database	MySQL
Database Connectivity	PhpMyAdmin
Browsers Support	Firefox,Chrome,Explorer
Antivirus	Avast anti-virus

4.4. System design

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements (Hoffer, 2008). The system design contains logical design and physical design where the logical design describes the structure and features of the system such as input, output, procedures, and database. From the logical design, you come up with the physical design which describes actual software and working system.

4.4.1 Activity Diagram for the system

The activity diagram below represents stages that data undergoes including processing stages and various states of objects in the system.

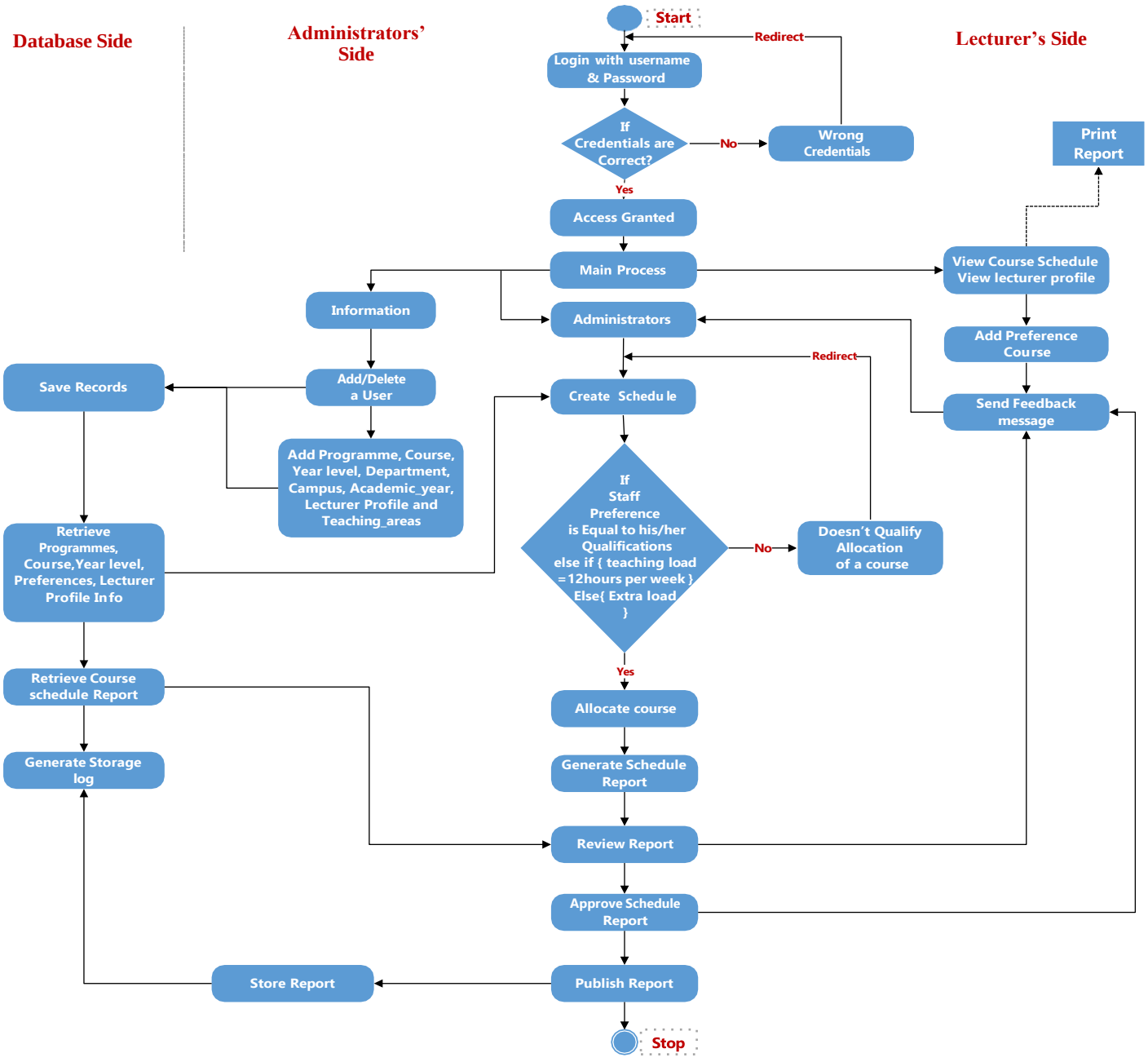


Figure 4.3: Showing the activity diagram that represents the various states of objects in the system.

4.4.2 Sequence Diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order.

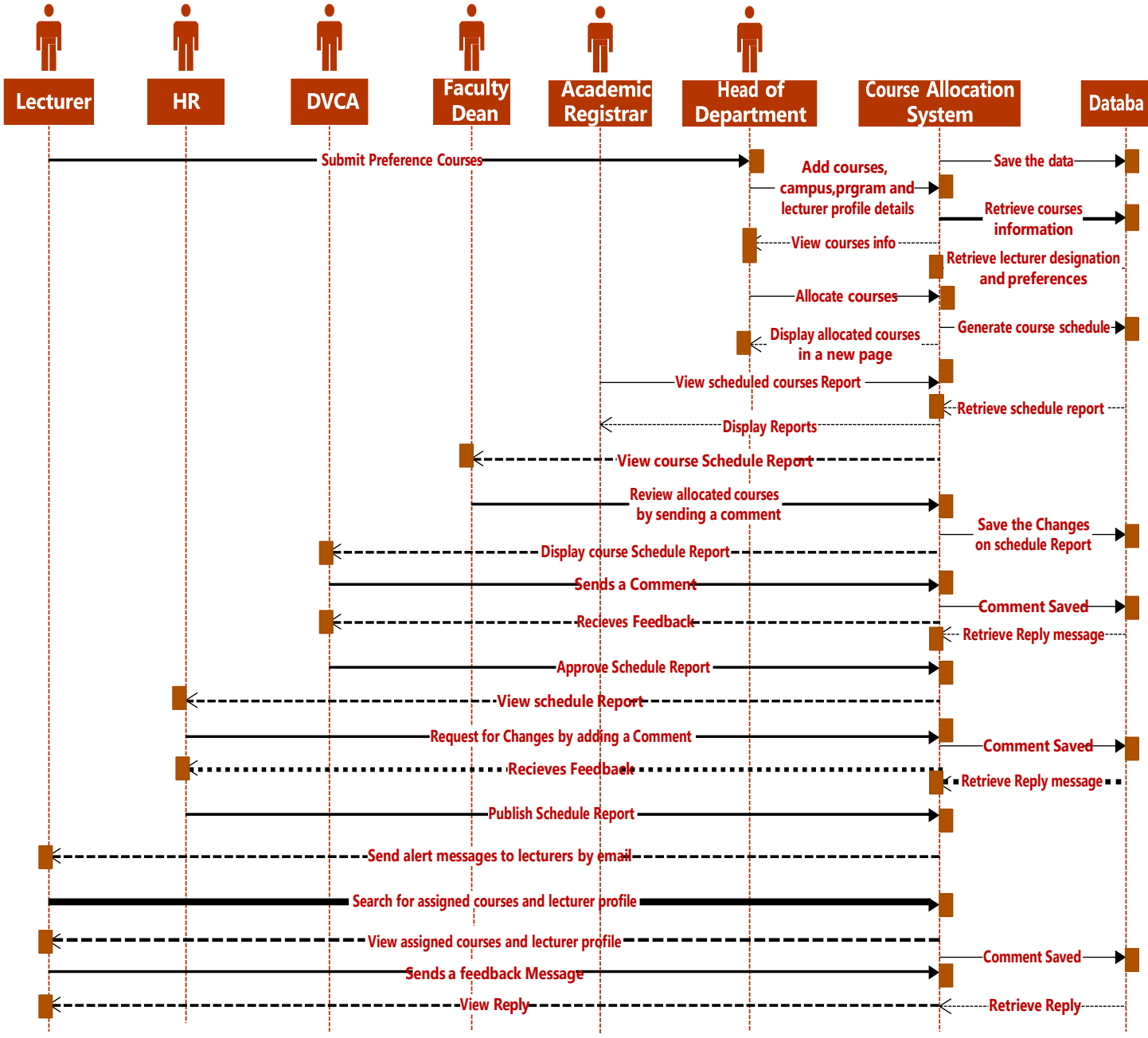


Figure 4.3: Showing a construct of a message sequence chart

4.4.3. Class diagram for OCASFAS

The class diagram shows the associations between different classes of the system. After the user logs into the system, a session is initiated and maintained until the user logs out. The figure below is the class diagram displaying the OCASFAS entities, their attributes and methods.

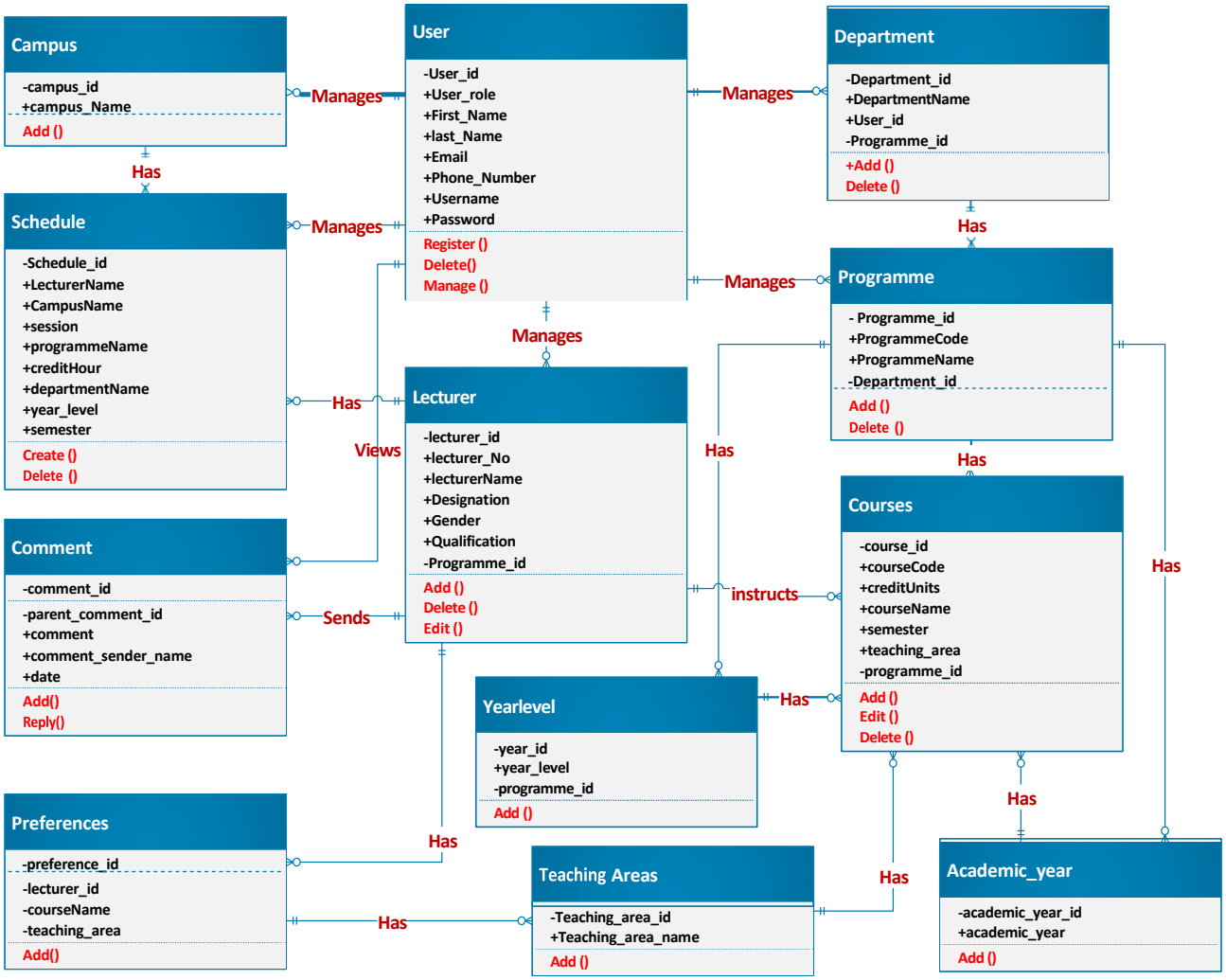


Figure 4.2: Showing a class diagram and associations between different classes of the system.

4.4.4 Database design

An online course allocation system for academic Staff is a database based application which means that the program interacts with a database at some point in its execution. In this case, a relational model is used which means that data is logically structured within tables. Each table has a name and it is made up of named attributes of data and also every row in a table contains at

most one value for each attribute. The database design comprises designing the logical, conceptual and physical database structure.

Additionally, the data model represents an important element in the determination of the system requirements.

Table 4.9: Showing Structure for table User.

Entities	Description	Attributes
Lecturer	This is the academic staff member in the department who teaches courses.	Lecturer_ID,Lecturer_No,LecturerName, Designation, Gender, Qualifications, Programme_id.
Programme	This represents a program within the faculty.	Program_id,Program_Name,Program_code, Department_id.
Department	This represents a department in the faculty.	Department_id,Department_Name,user_id, programme_id.
User	This represents the user or administrator who has the right to access the database to view, make changes and manage data.	User_id, User_role, First_Name,Last_Name, Email, Phone_Number,User_Name,Password.
Course	This represents a course taught by the lecturer.	Course_id,Course_code,Semester ,CourseName,CreditUnits,programme_id.
Comment	This represents a feedback message sent by the user	Comment_id,Parent_comment_id, Comment,Comment_sender_name, Date.
Preferences	This represents lecturer' subject choice	Preference_id,Lecturer_id,CourseName, Teaching_area .
Year level	This represents a program year level.	Year_id,Year_level,programmme_id,
Campus	This represents several	Campus_id,Campus_Name.

	university campuses.	
Schedule	This represents the courses assigned or allocated to the lecturer.	Schedule_id, LecturerName, CampusName, Session,Year_level, ProgrammeName, DepartmentName. Semester, CreditHour
Academic_Year	This represents the academic year of a program enrolment in a curricula	Academic_year_id,Academic_year.
Teaching_areas	This represents areas of specialization.	Teaching_area_id,Teaching_area_name

4.4.4.1 Data Dictionary for OCASFAS

Table 4.10: Showing Structure for table User.

Column Name	Constraint	Data Type	Description
user_id	Not Null	Int(11)	User identifier
user_role_id	Not Null	Int(11)	User role identifier
first_name	Not Null	Varchar(255)	User's first name
last_name	Not Null	Varchar(255)	User's last name
Email	Not Null	Varchar(255)	Email address
Contactno	Not Null	Varchar(15)	Phone number
Username	Not Null	Varchar(200)	Username
Password	Not Null	Varchar(50)	User password

Table 4.11: Showing Structure for table lecturer

Column Name	Constraint	Data Type	Description
lecturer_id	Not Null	Int(12)	Lecturer identifier
lecturer_No	Not Null	Varchar(30)	Lecturer number
lecturerName	Not Null	Varchar(50)	Lecturer name
Designation	Not Null	Varchar(50)	Lecturer position and title
lecturer_gender	Not Null	Varchar(8)	Lecturer gender
Qualification	Not Null	Varchar(255)	Lecturer's qualifications
Programme_id	Not Null	Int (11)	Lecturer's area of specialization

Table 4.12: Showing Structure for table Schedule

Column Name	Constraint	Data Type	Description
scheduleID	Not Null	Int(6)	Schedule identifier
programmeName	Not Null	Varchar(100)	Pragramme name
Semester	Not Null	Varchar(15)	Semester
Session	Not Null	Varchar(30)	School year session
courseName	Not Null	Varchar(35)	Course name
Lecturer	Not Null	Varchar(50)	Lecturer Name
campusName	Not Null	Varchar(50)	Campus name
departmentName	Not Null	Varchar(50)	Department name
Year_level	Not Null	Varchar(15)	Year level

Table 4.13: Showing Structure for table Academic_year

Column Name	Constraint	Data Type	Description
academic_year_id	Not Null	Int(5)	Session identifier
academic_year	Not Null	Varchar(30)	Academic_ year

Table 4.14: Showing Structure for table Programme

Column Name	Constraint	Data Type	Description
programme_id	Not Null	Int(5)	Programme identifier
programmeName	Not Null	Varchar(30)	Programme name
programmeCode	Not Null	Varchar(60)	Programme code
department_id	Not Null	Int(11)	Department identifier

Table 4.15: Showing Structure for table Course

Column Name	Constraint	Data Type	Description
course_id	Not Null	Int(5)	Course identifier
courseCode	Not Null	Varchar(30)	Course code
Semester	Not Null	Varchar(30)	Semester
courseName	Not Null	Varchar(60)	Course name
creditUnits	Not Null	Int(12)	Course unit hours
Programme_id	Not Null	Int(12)	Programme identifier
Teaching_area	Not Null	Varchar(60)	Teaching area

Table 4.16: Showing Structure for table Comment

Column Name	Constraint	Data Type	Description
comment_id	Not Null	Int(11)	Course identifier
Parent_comment_id	Not Null	Int(11)	Parent comment identifier
Comment	Not Null	Varchar(600)	Sent message
Comment_sender_name	Not Null	Varchar(40)	The sender of the message
Date	Not Null	Timestamp	The date when the message was sent

Table 4.17: Showing Structure for table Preferences

Column Name	Constraint	Data Type	Description
preference_id	Not Null	Int(11)	Preferences identifier
Lecturer_id	Not Null	Int(11)	Lecturer identifier
CourseName	Not Null	Varchar(40)	Course name
Teaching_area	Not Null	Varchar(40)	Teaching area

Table 4.18: Showing Structure for table Year level

Column Name	Constraint	Data Type	Description
year_level_id	Not Null	Int(12)	Year level identifier
year_level	Not Null	Varchar(30)	Year level
programmme_id	Not Null	Int(12)	Course identifier(FK)

Table 4.19: Showing Structure for table department

Column Name	Constraint	Data Type	Description
department_id	Not Null	Int(11)	Department identifier
departmentName	Not Null	Varchar(50)	Department name
User_id	Not Null	Int(11)	User identifier
Programme_id	Not Null	Int(11)	Programme identifier

Table 4.20: Showing Structure for table Teaching_areas

Column Name	Constraint	Data Type	Description
teaching_area_id	Not Null	Int(5)	Teaching_area identifier
teaching_area	Not Null	Varchar(30)	Teaching area

Table 4.21: Showing Structure for table Campus

Column Name	Constraint	Data Type	Description
Campus_id	Not Null	Int(5)	Campus identifier
campusName	Not Null	Varchar(30)	Campus name

4.4.4.2 Logical database design

The list below describes the main entities identified for the proposed system with their associated attributes defined with the primary keys underlined and the foreign keys [FK].

Programme (**programmeID**, programmeCode, programmeName, **department_id [FK]**).

Department (**department_id**, departmentName,**user_id[FK],progarmme_id [FK]**).

Lecturer (**lecturer_id**, lecturerName, designation, lecturer_gender, lecturer_No**,** Qualification, **programme_id[FK]**).

Course (**course_id**, courseCode, courseName, semester, creditUnits,Teaching_area, **programme_id[FK]**).

User (**user_id**, **user_role [FK],** First_Name, Last_Name, Email, phoneNumber, username, password).

Academic_Year(**academic_year_id**,academic_year).

Campus (**campus_id**, campusName).

Comment (**comment_id**, **parent_comment_id[FK]**,comment,comment_sender_name,date)

Preferences (**preference_id**,lecturer_id,teaching_area,**courseName**)

Schedule (**schedule_id**, lecturerName, CampusName, session, courseName, year_level, programmeName, semester, DepartmentName, creditHour).

Teaching_areas (**teaching_area_id**, teaching_area_name).

Year_level (**year_id**, year_level, **programme_id [FK]**).

4.4.4.3 Physical design of the database

Below is the query that was used to build the database, tables and data dictionary for the proposed system.

Creating the database

CREATE DATABASE course_scheduling

USE course_scheduling

SET SQL_MODE=”NO_AUTO_VALUE_ON_ZERO”,

Table structure for table ‘User’

CREATE TABLE ‘User’ (‘user_id’ int (11) NOT NULL auto_increment, ’user_role_id’ int (11) NOT NULL, ’first_name’ varchar (255) NOT NULL, ’last_name’ varchar (255) NOT NULL, ’email’ varchar (255) NOT NULL, ’Contactno’ varchar (15) NOT NULL, ’username’ varchar (200) NOT NULL, ’password’ varchar (50) NOT NULL, PRIMARY KEY (user_id));

Table structure for table ‘lecturer’

CREATE TABLE ‘lecturer’ (’lecturer_id’ int (12) NOT NULL auto_increment, ’lecturer_No’ varchar (30) NOT NULL, ’lecturerName’ varchar (50) NOT NULL, ’designation’ varchar (50) NOT NULL, ’lecturer_gender’ varchar (8) NOT NULL, ’Qualification’ varchar (255) NOT NULL, ’Programme_id’ int (11) NOT NULL, PRIMARY KEY (lecturer_id));

Table structure for table ‘schedule’

CREATE TABLE ‘schedule’ (’scheduleID’ int (6) NOT NULL auto_increment, ’programmeName’ varchar (300) NOT NULL, ’semester’ varchar (15) NOT NULL, ’session’ varchar (30) NOT NULL, ’courseName’ varchar (35) NOT NULL, ’lecturer’ varchar (50) NOT NULL, ’campusName’ varchar (50) NOT NULL, ’departmentName’ varchar (50) NOT NULL, ’year_level’ varchar (15) NOT NULL, PRIMARY KEY (‘scheduleID’));

Table structure for table ‘academic_year’

CREATE TABLE ‘academic_year’ (’academic_year_id’ int (5) NOT NULL auto_increment, ’academic_year’ varchar (30) NOT NULL, PRIMARY KEY (‘academic_year_id’));

Table structure for table ‘programme’

CREATE TABLE ‘programme’ (’programme_id’ int (5) NOT NULL auto_increment, ’programme’ varchar (30) NOT NULL, ’programmeCode’ varchar (60) NOT NULL, PRIMARY KEY (‘programme_id’));

Table structure for table ‘course’

CREATE TABLE ‘course’ (’course_id’ int (5) NOT NULL auto_increment, ’courseCode’ varchar (30) NOT NULL, ’semester’ varchar (30) NOT NULL, ’first_name’ varchar (255) NOT NULL, ’courseName’ varchar (60) NOT NULL, ’creditUnits’ int (12) NOT NULL, ’Programme_id’ int (12) NOT NULL, ’teaching_area’ varchar (60) NOT NULL, PRIMARY KEY (‘course_id’));

Table structure for table ‘comment’

CREATE TABLE ‘comment’ (’comment_id’ int(11) NOT NULL auto_increment, ’parent_comment_id’ int(11) NOT NULL, ’comment’ varchar (600) NOT NULL, ’comment_sender_name’ varchar (40) NOT NULL, ’date’ Timestamp NOT NULL, PRIMARY KEY (‘comment_id’));

Table structure for table ‘preferences’

CREATE TABLE ‘preferences’ (’preference_id’ int (12) NOT NULL auto_increment, ’courseName’ varchar (40) NOT NULL, ’lecturer_id’ int (12) NOT NULL, ’teaching_area’ varchar (40) NOT NULL, PRIMARY KEY (‘preference_id’));

Table structure for table ‘Year level’

CREATE TABLE ‘year_level’ (’year_level_id’ int (12) NOT NULL auto_increment, ’year_level’ varchar (30) NOT NULL, ’programme_id’ int (12) NOT NULL, PRIMARY KEY (‘year_level_id’));

Table structure for table ‘department’

CREATE TABLE ‘department’ (’department_id’ int (11) NOT NULL auto_increment, ’departmentName’ varchar (50) NOT NULL, ’user_id’ int (11) NOT NULL, ’programme_id’ varchar (11) NOT NULL, PRIMARY KEY (‘department_id’));

Table structure for table ‘teaching_areas’

CREATE TABLE ‘teaching_area’ (’teaching_area_id’ int (5) NOT NULL auto_increment, ’teaching_area’ varchar (30) NOT NULL, PRIMARY KEY (‘curricula_id’));

Table structure for table ‘campus’

CREATE TABLE ‘campus’ (’campus_id’ int (5) NOT NULL auto_increment, ’campusName’ varchar (30) NOT NULL, PRIMARY KEY (‘campus_id’));

4.5. Conclusion

In this chapter, the data was analyzed and user/system requirements were determined. The requirements of the proposed system were discussed and analyzed through class diagram, activity diagram, use case diagram and sequence diagram. The functionality and security specification of the new system was also determined in this chapter.

CHAPTER FIVE

SYSTEM IMPLEMENTATION AND TESTING

5.1 Introduction

In this chapter, the researcher explains how the system designs were translated into a working system. It includes the implementation plan, outputs (screenshots) for the various user interfaces, database implementation and system testing.

5.2 Implementation plan

This section shows the different activities, deliverables and tools that were carried out during the implementation of the system as shown in the table below.

Table 5.1: Implementation plan

ACTIVITY	DELIVERABLES	TOOLS
Coding	-Implement components of the system design. -Connect the user interfaces to the database.	Dreamweaver,PHP,HTML, JavaScript,Bootstrap.MySQL and Xampp server.
Testing Plan	-Review of code for error correction. -Test correctness, performance and system reliability. -User testing of the system. -Security testing and authentication. -Test the documentation of the results.	-Dreamweaver -Firefox and Chrome browsers.
Documentation	-System documentation and key features.	-MS word and PDF.

5.3 Coding

At this point, the logical, physical design specifications and models designed earlier were converted into working computer codes.

5.3.1 Database implementation

The database was created using MySQL and the database was run on MySQL server. The database and all the defined components were designed and generated using MySQL server, the tables were normalized by use of “Primary keys” to uniquely identify each entry in the database and the “foreign key” to show the relationships by linking different tables.

Below is a screen shot of the database and database tables.

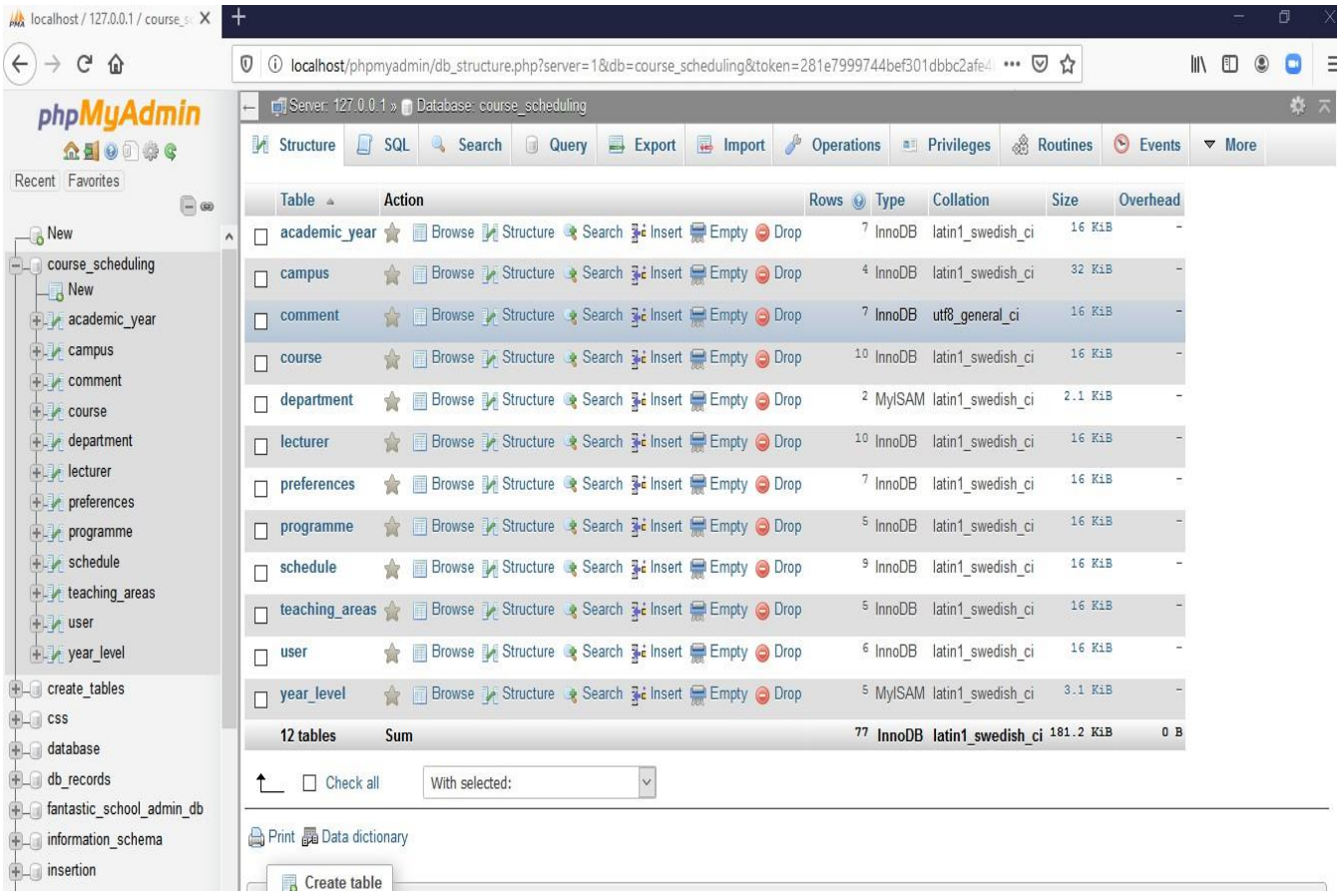


Figure 5.1: Screenshot of the database and database tables.

5.3.2 User interface implementation

The system was developed as an interactive mechanism between the user at the interface and the database through the web-browser. Dreamweaver a web page program editor was used for coding and de-bugging errors. PHP was used to build the front end and building the logic to connect the interface to the database in the backend, CSS for styling up the interfaces and bootstrap. The user interface enables the user to interact with the database to enter, edit, retrieve and view data according to user roles and privileges granted.

5.3.2.1 The login page

Index.php

This script is the login page of the OCASFAS and it provides access to the registered users. This allows all users to login and their views are determined by the administrator's selection of their roles in the system. The user can enter his/her user name and password. Running the index.php (http://localhost/course/index.php)automatically starts up the login page as shown below.

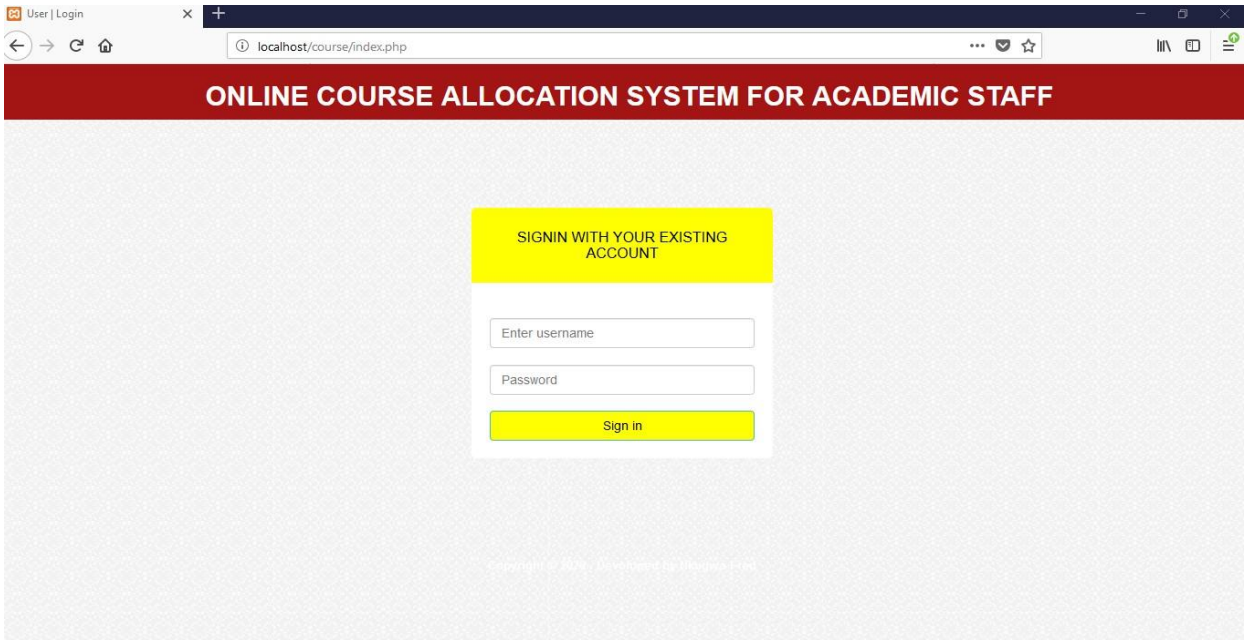


Figure 5.2: Screenshot of login page

5.3.2.2 Interface for system administrator

http://Iocalhost/course/dashboard.php

This script opens the system administrator control panel. He/she has the privileges to access and use this page to do his/her duties like adding new users to the system and managing users of the system.

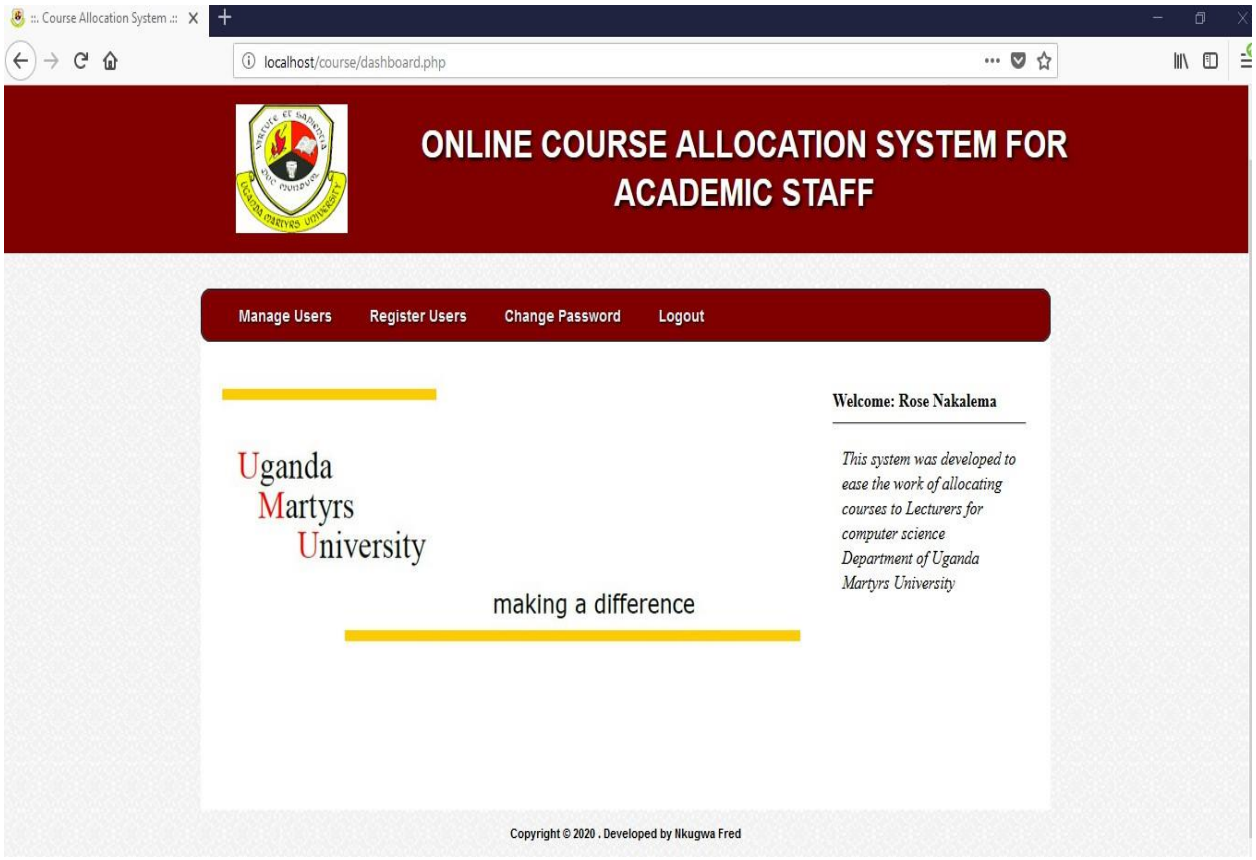


Figure 5.3: Screenshot of System administrator page
http://localhost/course/sysadmin/managerusers.php

This script opens the interface where the systems administrator manages users of the system.

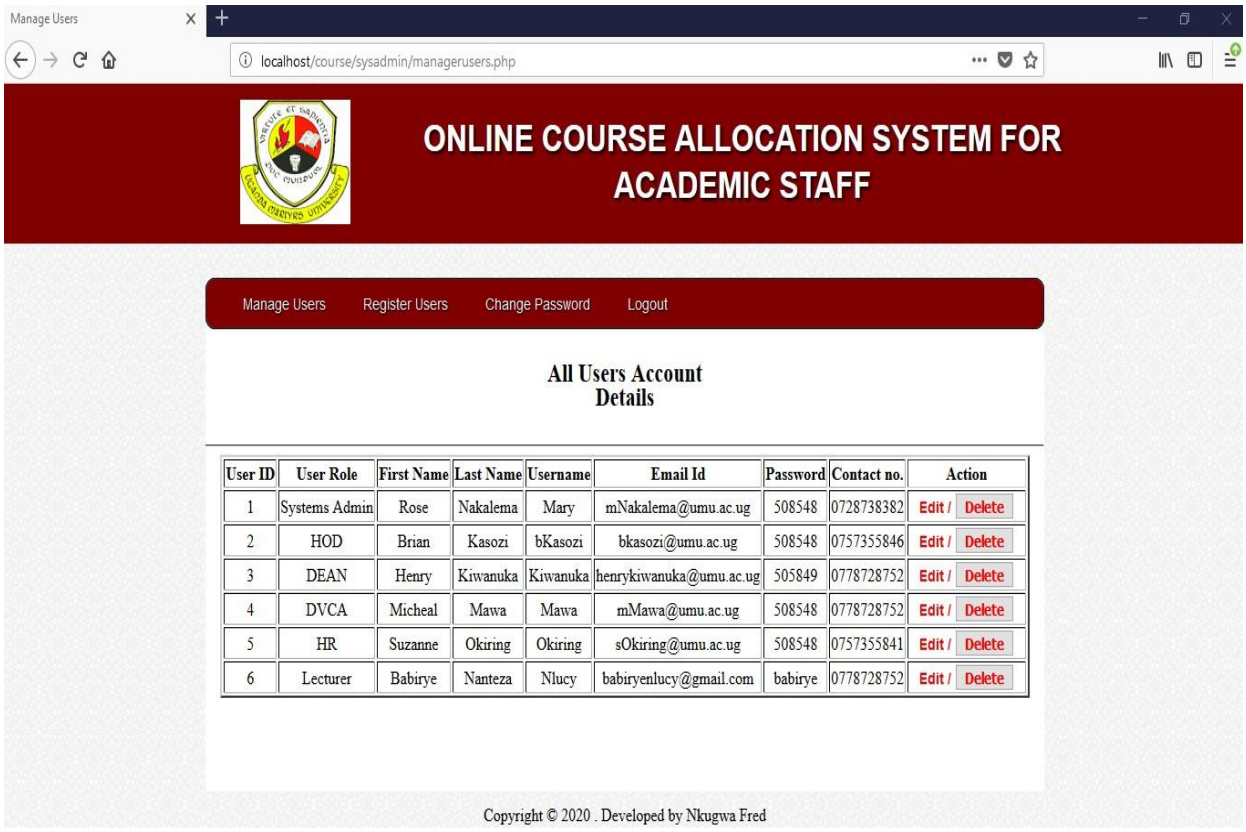


Figure 5.3: Screenshot of System administrator page for managing the users.

5.3.2.3 Interface for Course structure

http://localhost/course/admin/Program.php

This script is used to layout the program and courses in the system. A course cannot be registered without an existing program under which this course belongs. The page makes it possible for the head of department to register new programs and courses as shown in the figures below.

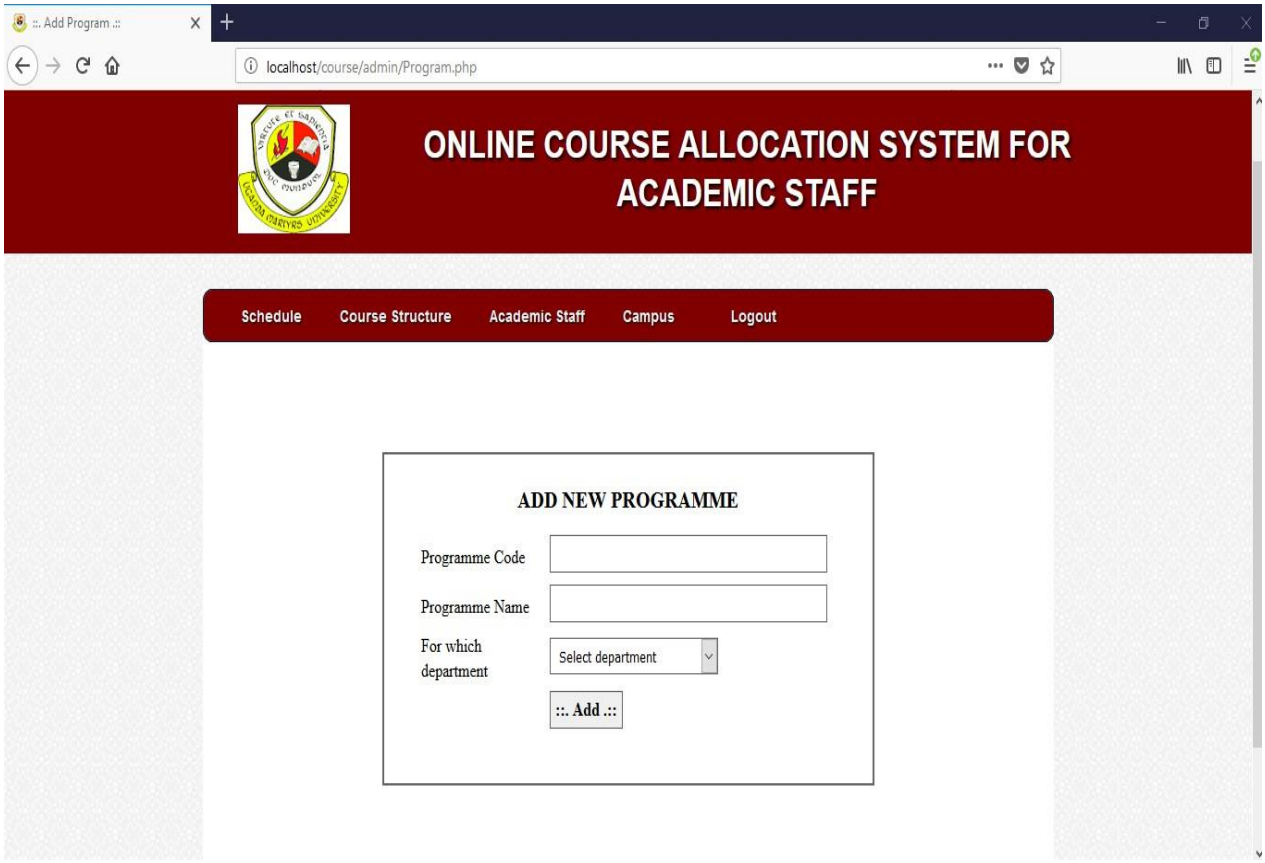


Figure 5.5: Screen shot of adding a new program.

Course registration

http://localhost/course/admin/add_course.php

The add a new course, there must be an existing program under which that course is to be registered. The system allows the head of the department to enter new courses as shown below.

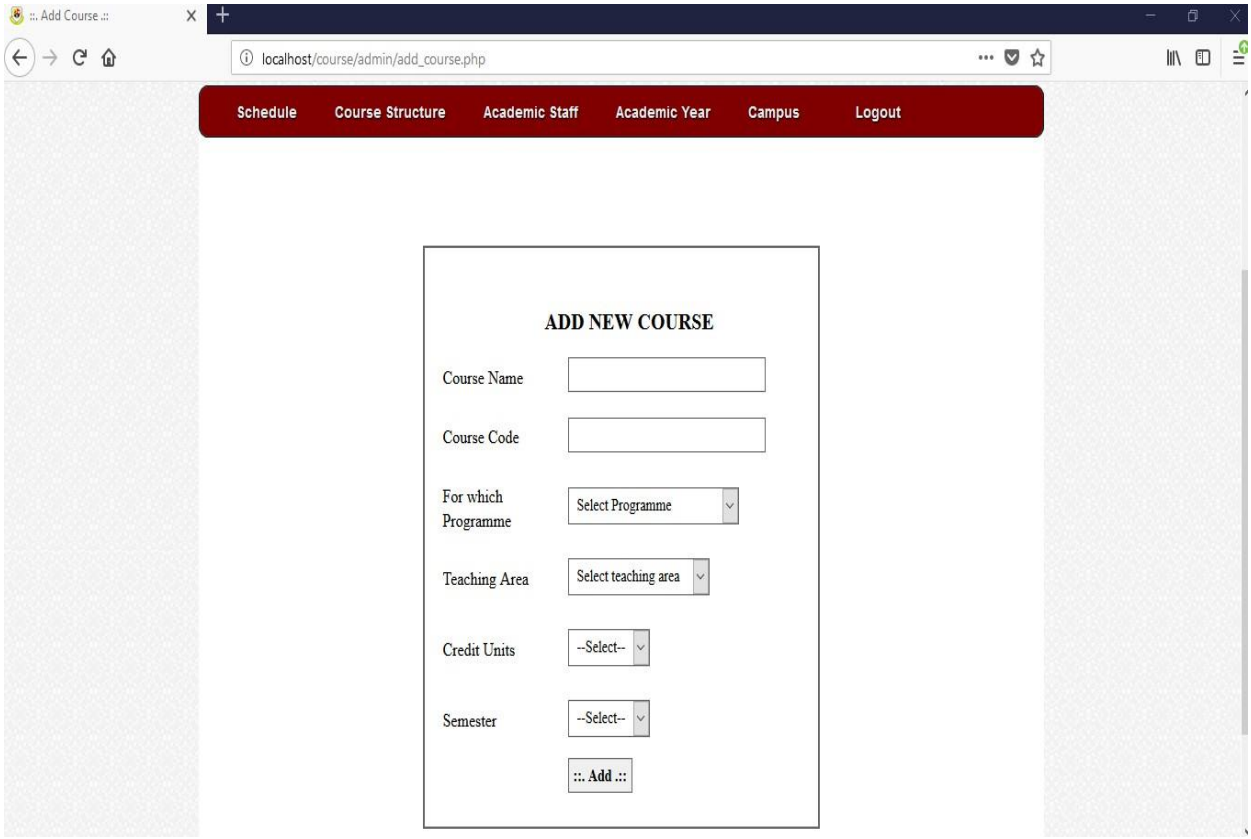


Figure 5.6: Screen shot of adding new courses

View programs interface

http://localhost/course/admin/ViewPrograms.php

Each program has courses offered and to view these courses, the link “course Structure” is clicked, then click on view programs link to view the programs list, finally click view courses under that program and a list of courses under that program is displayed as shown in the two figures below.

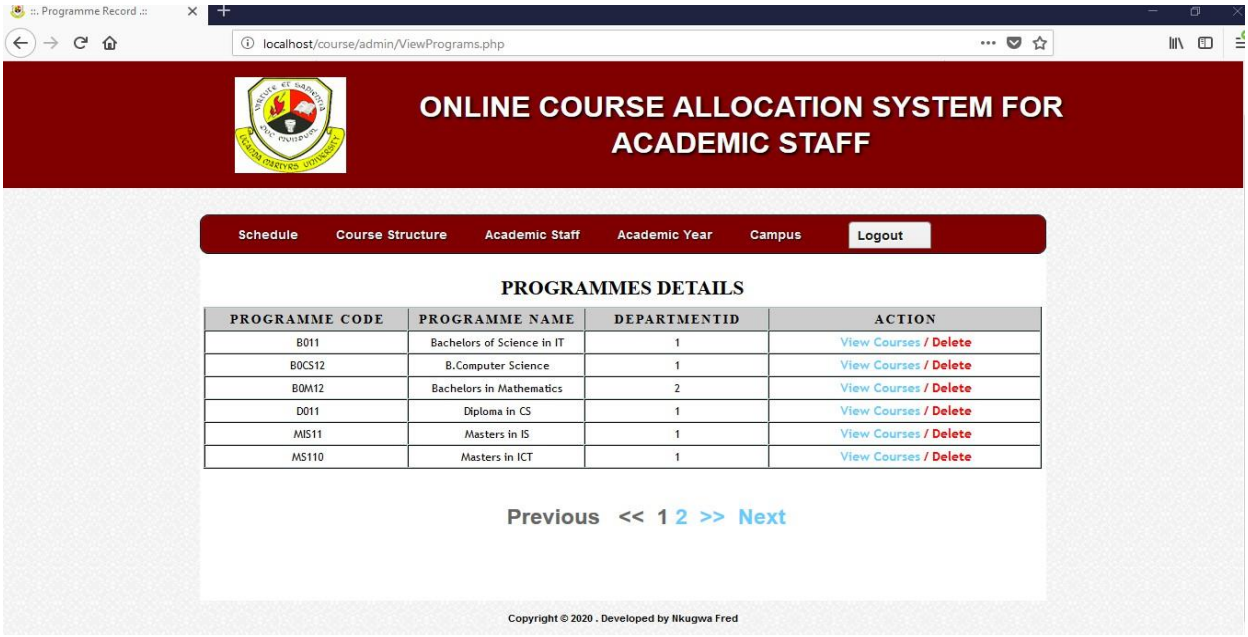


Figure 5.7: Screen shot of displaying programs list.

View Courses

http://localhost/course/admin/courserecord.php

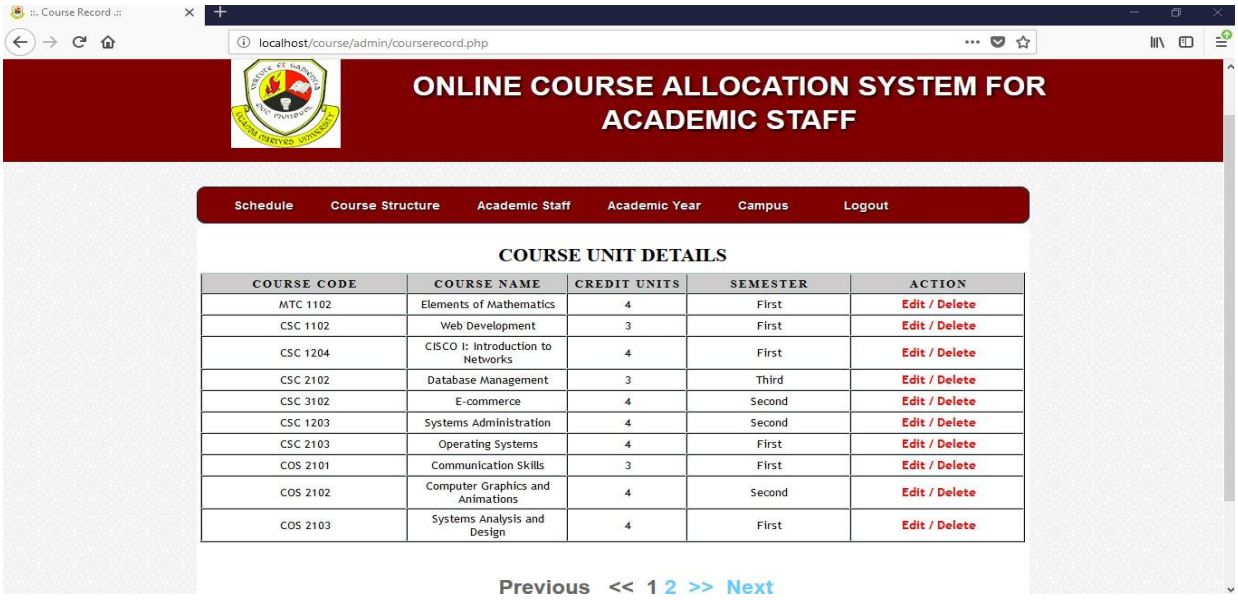


Figure 5.8: Screenshot of courses reports page.

5.3.2.4 Head of department interface for managing academic staff.

This link is used to structure the academic staff module. This is where the academic staff is registered into the system and it links to other functions that execute the queries to the academic staff entity in the database. The following are the functions implemented under this interface.

Register academic staff

The system can only allow the head of the department to add a new academic staff if he/she belongs to that faculty/department in order to be allocated course workload.

http://localhost/course/admin/add_lecturer.php

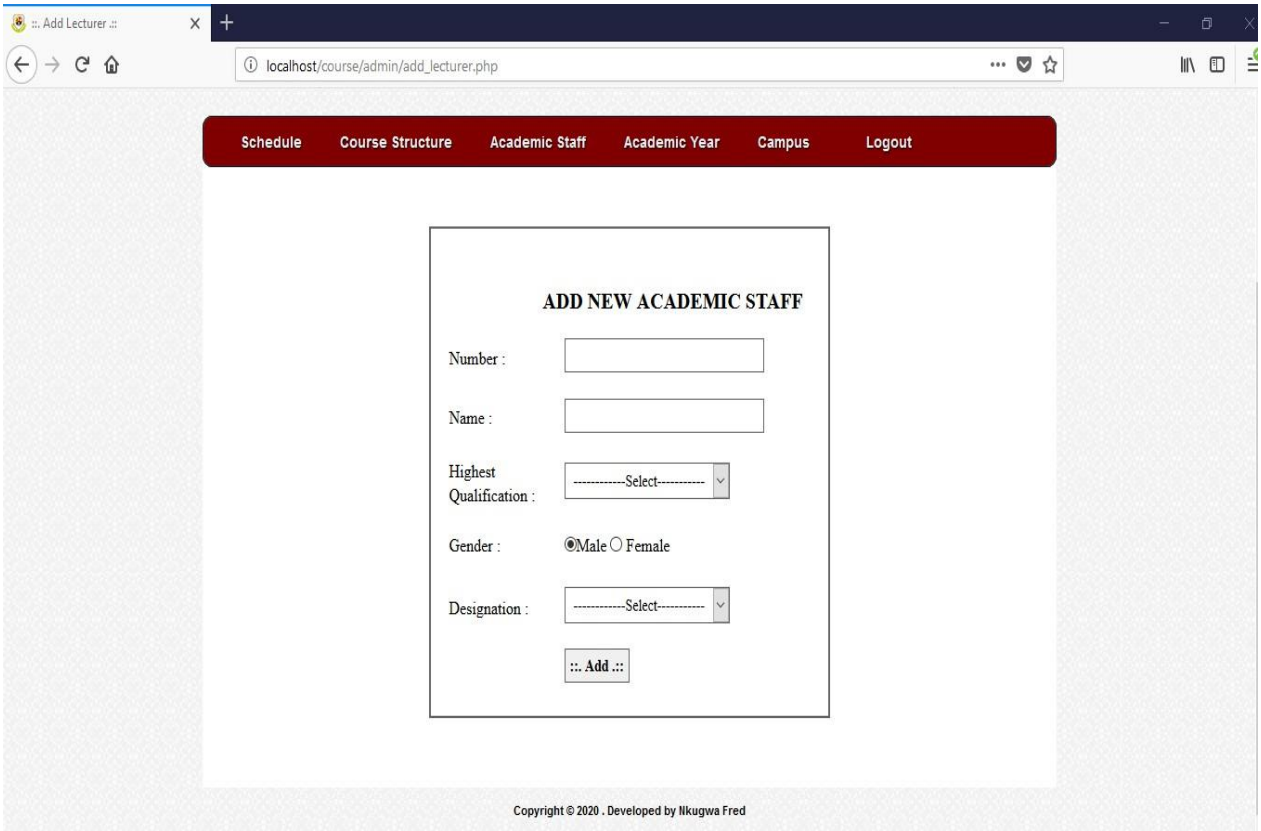
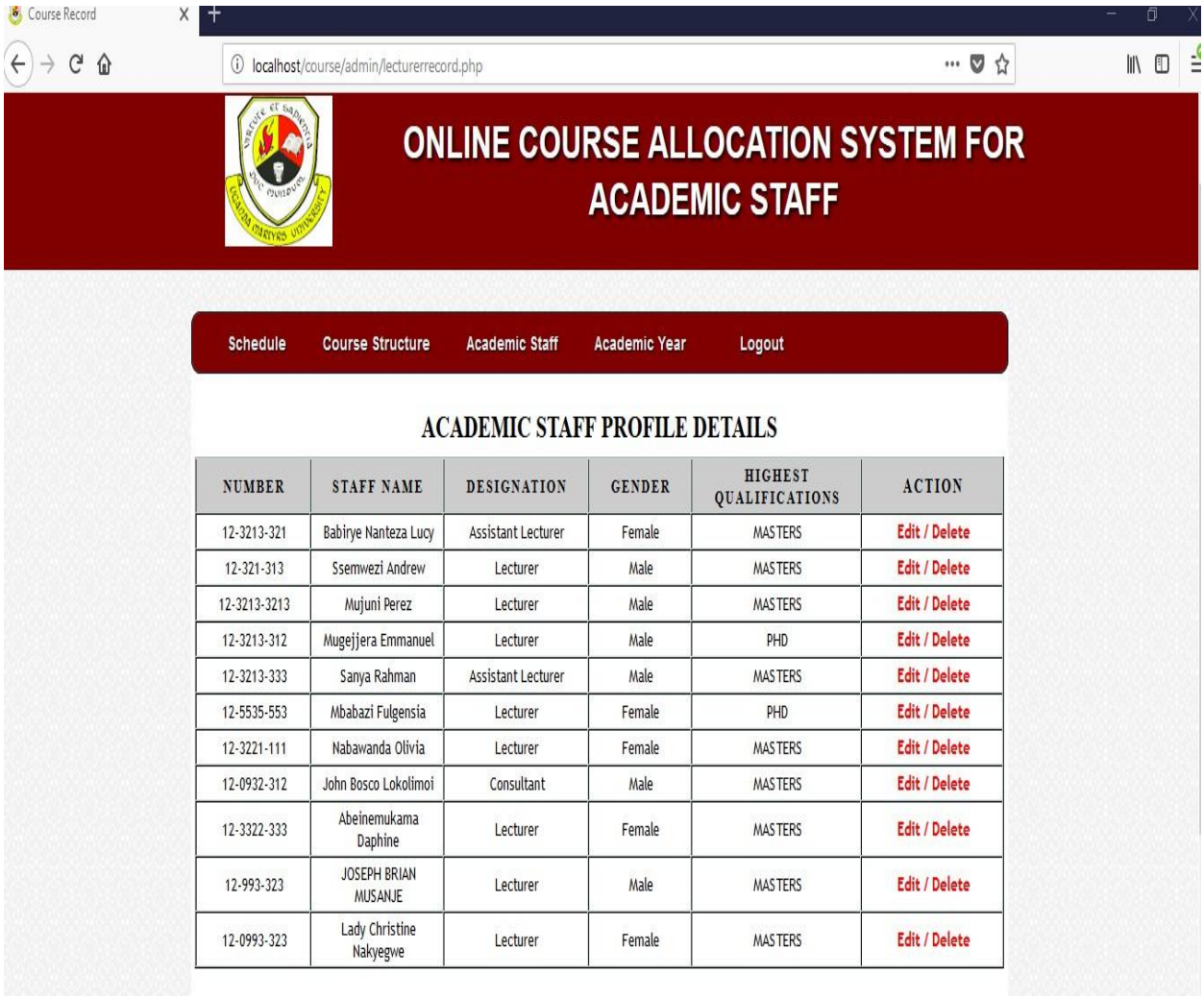


Figure 5.9: Screenshot for adding a new academic staff page.

View the academic staff report

When the head of the department clicks on the link "View academic staff", the list of registered academic staff is displayed. The head of department can edit academic staff's profile or delete academic staff record.

http://localhost/course/admin/lecturerrecord.php



NUMBER	STAFF NAME	DESIGNATION	GENDER	HIGHEST QUALIFICATIONS	ACTION
12-3213-321	Babirye Nanteza Lucy	Assistant Lecturer	Female	MASTERS	Edit / Delete
12-321-313	Ssemwezi Andrew	Lecturer	Male	MASTERS	Edit / Delete
12-3213-3213	Mujuni Perez	Lecturer	Male	MASTERS	Edit / Delete
12-3213-312	Mugejjera Emmanuel	Lecturer	Male	PHD	Edit / Delete
12-3213-333	Sanya Rahman	Assistant Lecturer	Male	MASTERS	Edit / Delete
12-5535-553	Mbabazi Fulgensia	Lecturer	Female	PHD	Edit / Delete
12-3221-111	Nabawanda Olivia	Lecturer	Female	MASTERS	Edit / Delete
12-0932-312	John Bosco Lokolimoi	Consultant	Male	MASTERS	Edit / Delete
12-3322-333	Abeinemukama Daphine	Lecturer	Female	MASTERS	Edit / Delete
12-993-323	JOSEPH BRIAN MUSANJE	Lecturer	Male	MASTERS	Edit / Delete
12-0993-323	Lady Christine Nakyegwe	Lecturer	Female	MASTERS	Edit / Delete

Figure 5.10: Screen shot for the academic staff report

View preferences report

When the head of department clicks on the link “View preferences”, the list of preferred courses registered by the academic staff is displayed. This can be used during allocation of courses to the academic staff.

http://localhost/course/admin/PreferenceRec.php

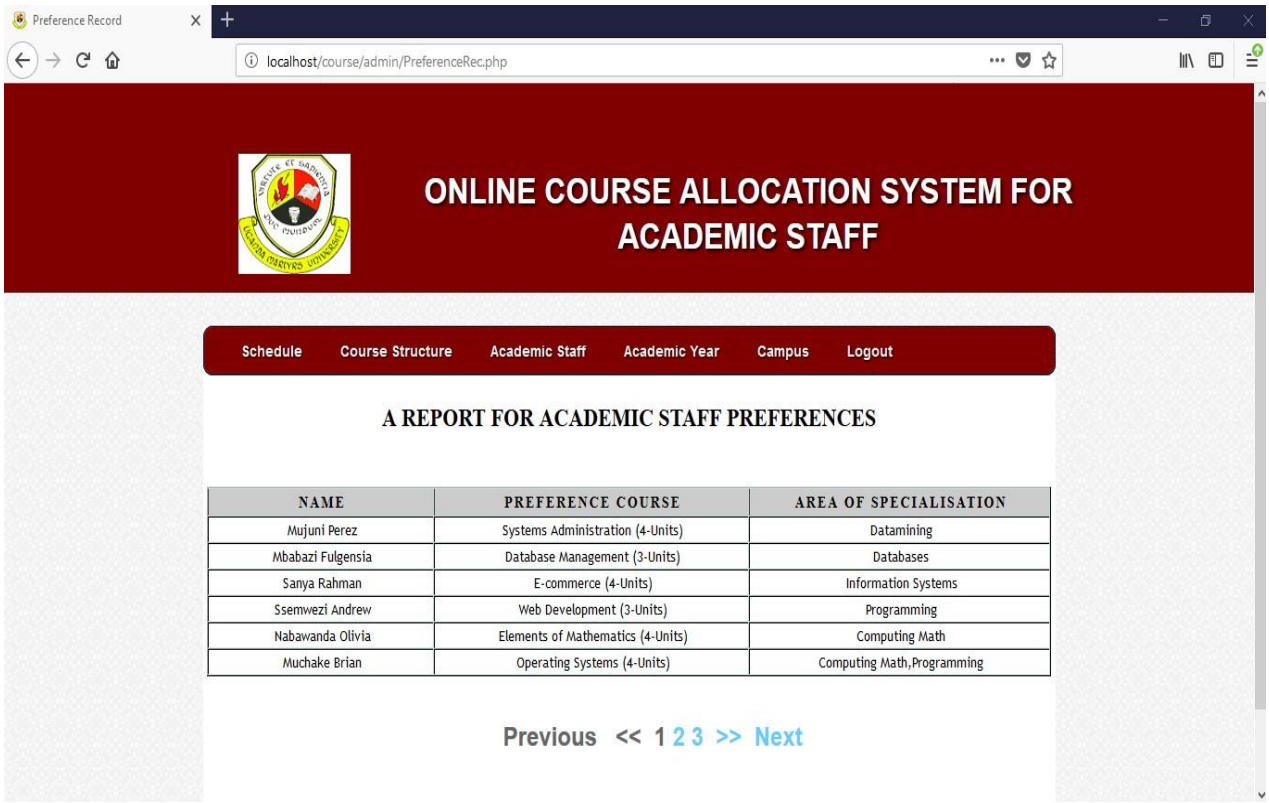


Figure 5.11: Screen shot for the academic staff report.

5.3.2.4 Campus interface

http://localhost/course/admin/Add_Campus.php

This script enables the head of department to add a new campus by clicking on the link “Add campus”.

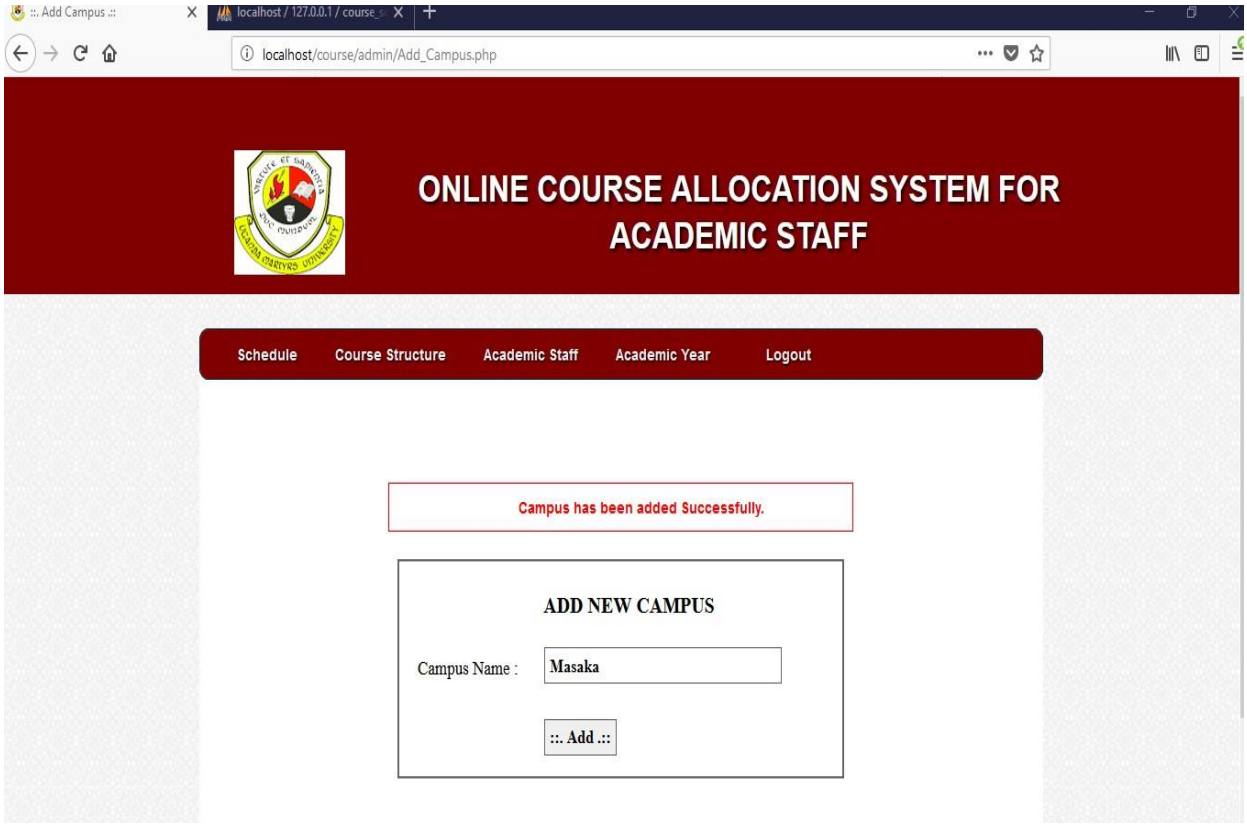


Figure 5.12: Screen shot of the add campus page.

5.3.2.5 Academic year interface

http://localhost/course/admin/add_session.php

This script enables the head of department to add a new academic year by clicking on the link "Add academic year"

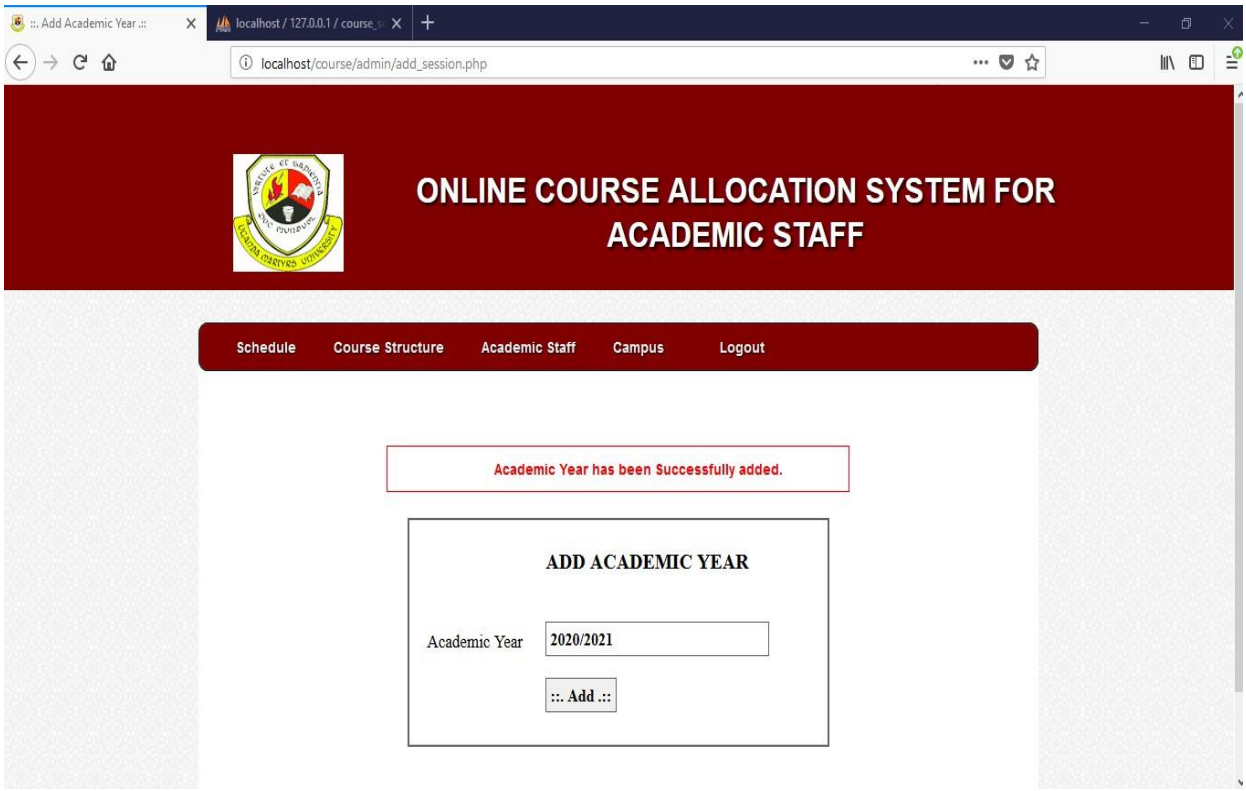


Figure 5.13: Screen shot for adding a new academic year.

5.3.2.6 Scheduling interface

This link enables the head of the department to allocate courses to the academic staff and the schedule is registered into the system. It links to other functions that execute the queries to the schedule entity in the database. The following are the functions implemented under this interface.

Create Schedule interface

http://localhost/course/admin/add_schedule.php

The page makes it possible for the head of department to generate schedule to the academic staff as shown in the figures below.

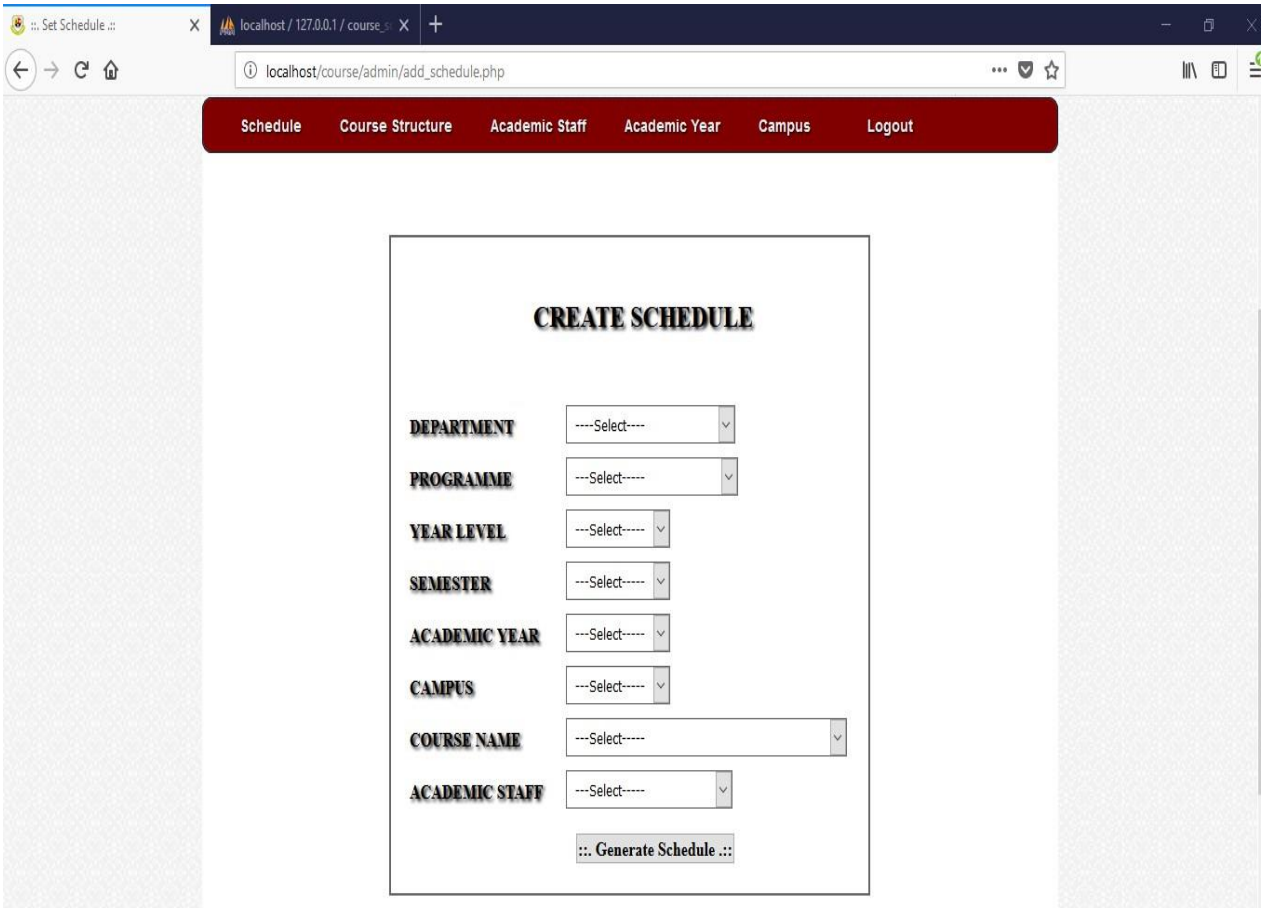
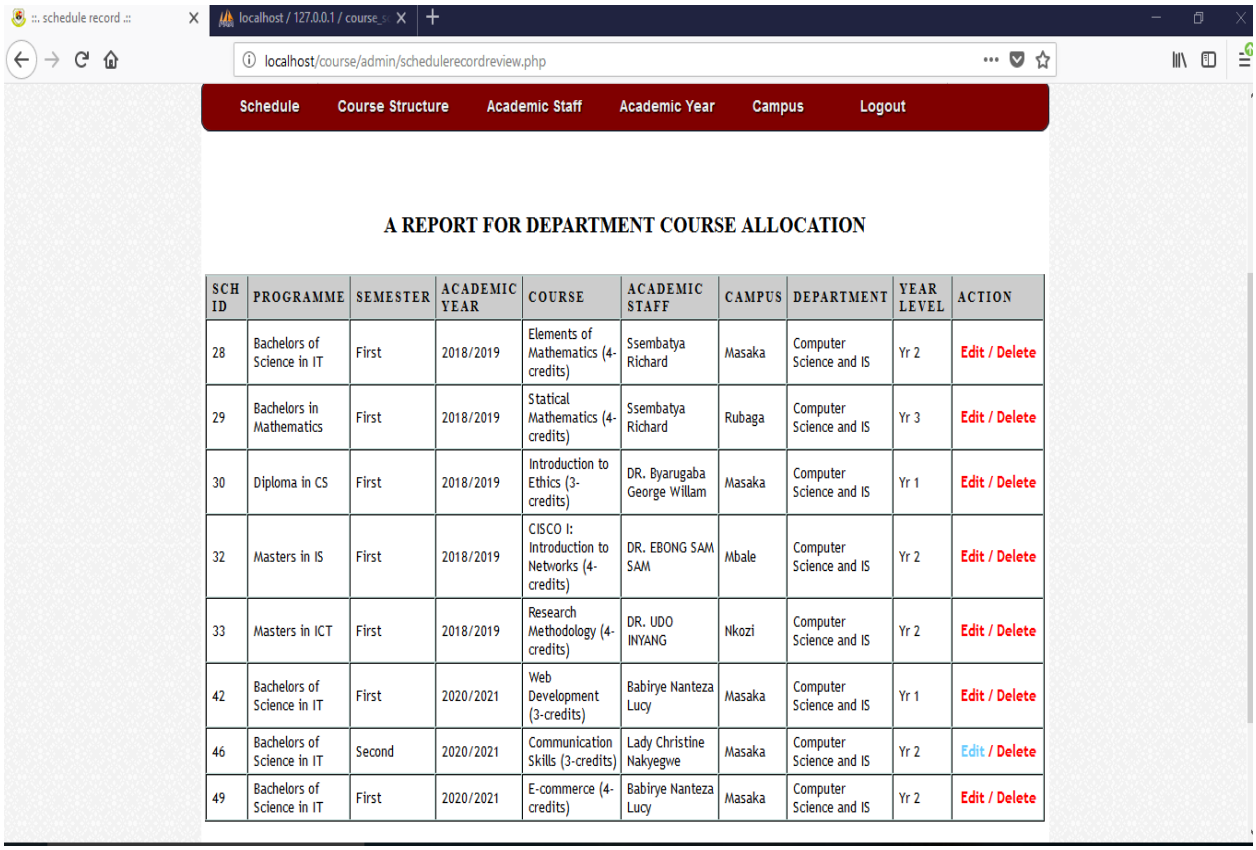


Figure 5.14: Screenshot of creating schedule page.

Review schedule Report

http://localhost/course/admin/scheduleerecordreview.php

The page makes it possible for the head of the department to review the schedule courses to the academic staff and also perform other functions like editing and deleting of records as shown in the figures below.



SCH ID	PROGRAMME	SEMESTER	ACADEMIC YEAR	COURSE	ACADEMIC STAFF	CAMPUS	DEPARTMENT	YEAR LEVEL	ACTION
28	Bachelors of Science in IT	First	2018/2019	Elements of Mathematics (4-credits)	Ssembatya Richard	Masaka	Computer Science and IS	Yr 2	Edit / Delete
29	Bachelors in Mathematics	First	2018/2019	Statical Mathematics (4-credits)	Ssembatya Richard	Rubaga	Computer Science and IS	Yr 3	Edit / Delete
30	Diploma in CS	First	2018/2019	Introduction to Ethics (3-credits)	DR. Byarugaba George Willam	Masaka	Computer Science and IS	Yr 1	Edit / Delete
32	Masters in IS	First	2018/2019	CISCO I: Introduction to Networks (4-credits)	DR. EBONG SAM SAM	Mbale	Computer Science and IS	Yr 2	Edit / Delete
33	Masters in ICT	First	2018/2019	Research Methodology (4-credits)	DR. UDO INYANG	Nkozi	Computer Science and IS	Yr 2	Edit / Delete
42	Bachelors of Science in IT	First	2020/2021	Web Development (3-credits)	Babirye Nanteza Lucy	Masaka	Computer Science and IS	Yr 1	Edit / Delete
46	Bachelors of Science in IT	Second	2020/2021	Communication Skills (3-credits)	Lady Christine Nakyegwe	Masaka	Computer Science and IS	Yr 2	Edit / Delete
49	Bachelors of Science in IT	First	2020/2021	E-commerce (4-credits)	Babirye Nanteza Lucy	Masaka	Computer Science and IS	Yr 2	Edit / Delete

Figure 5.15: Screenshot of reviewing the schedule results.

Filter schedule records

http://localhost/course/admin/schedulerecord.php

This script makes it possible for the head of department to filter schedule results according to a specified criterion as shown in the figures below.

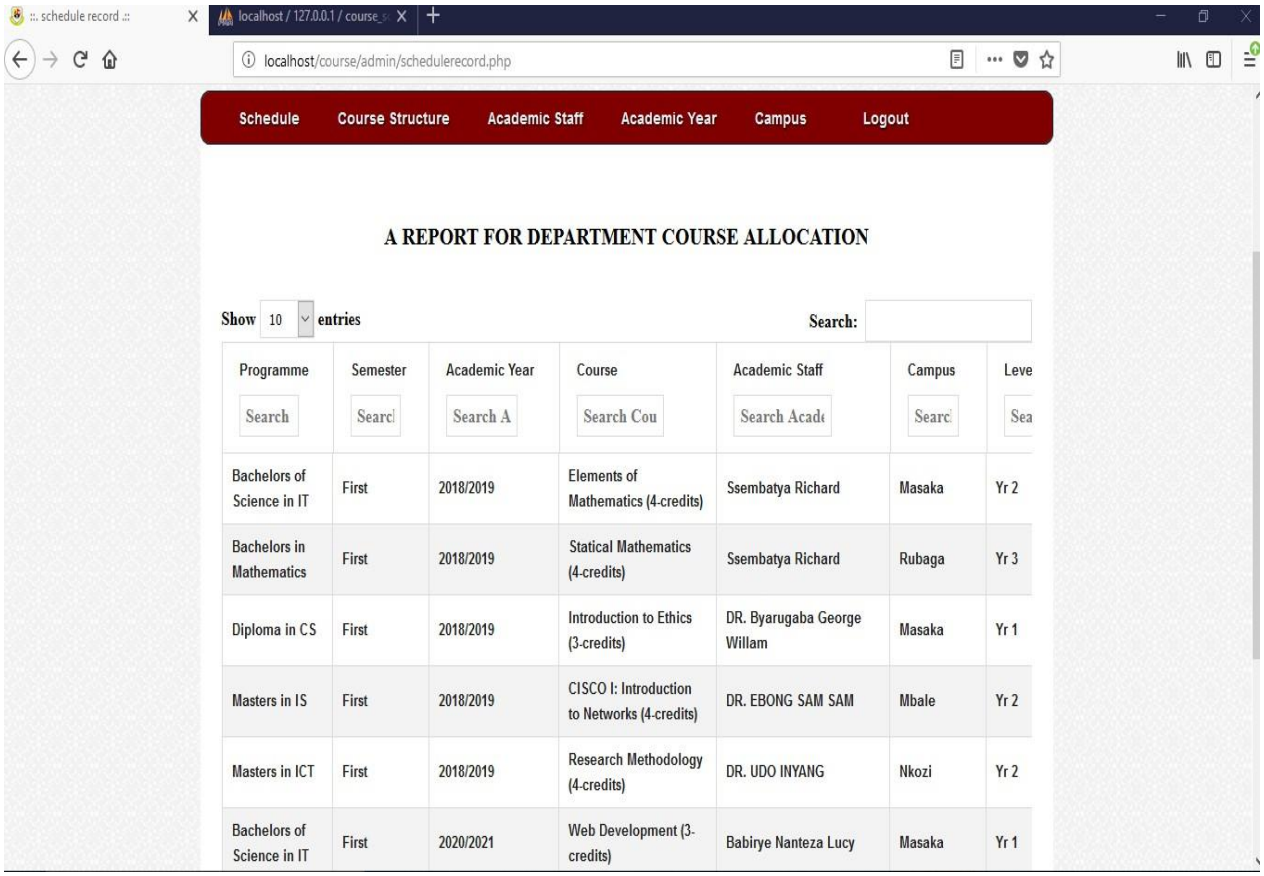


Figure 5.16: Screenshot of the filter schedule records.

View sent Comment

localhost/course/admin/Comment.php

This script makes it possible for the head of department to view comments sent by various academic staff concerning the allocated courses and any other review required as shown in the figure below.

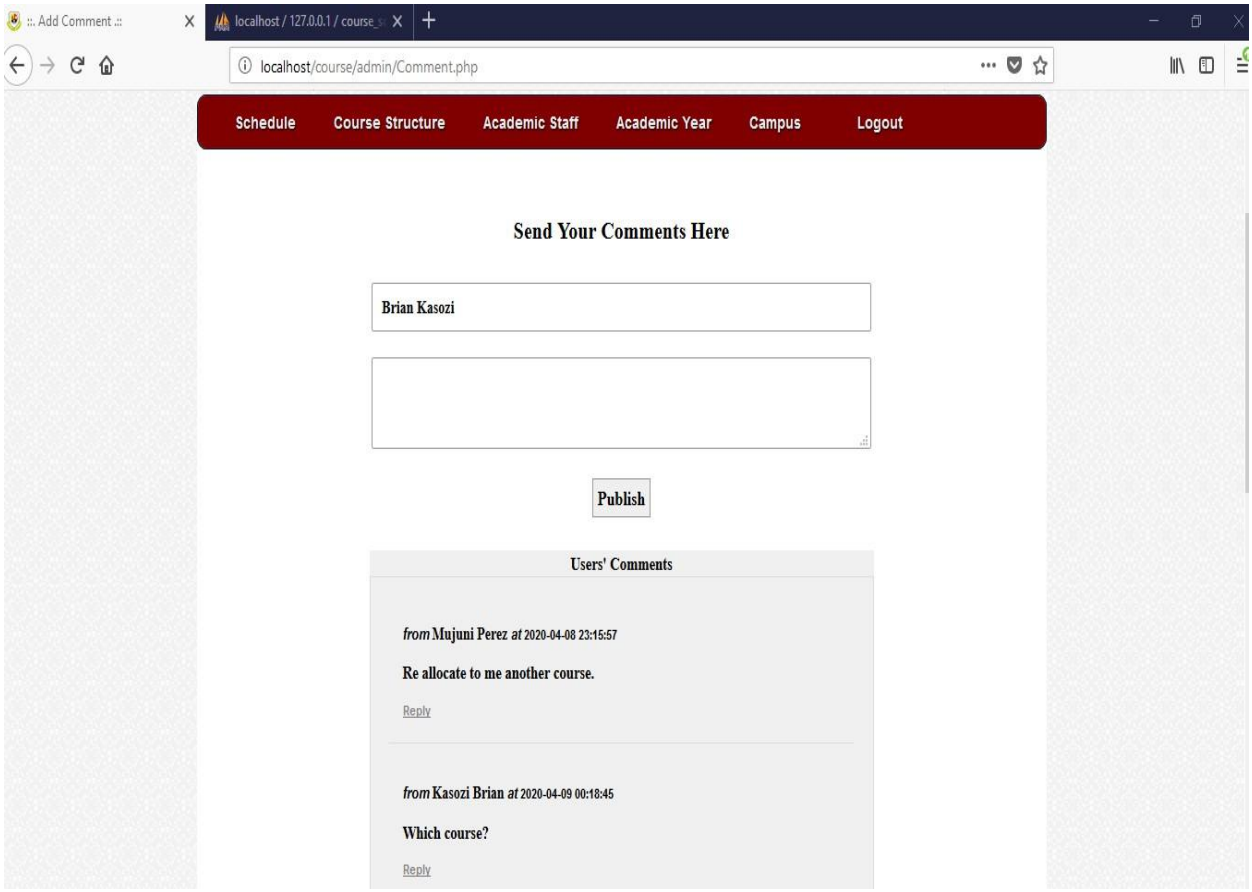


Figure 5.15: Screenshot of the display sent comments from the users.

5.3.2.7 Academic staff schedule interface

This page enables the academic staff to search for his/her teaching load by filtering according to academic year and semester as shown in the figures below.

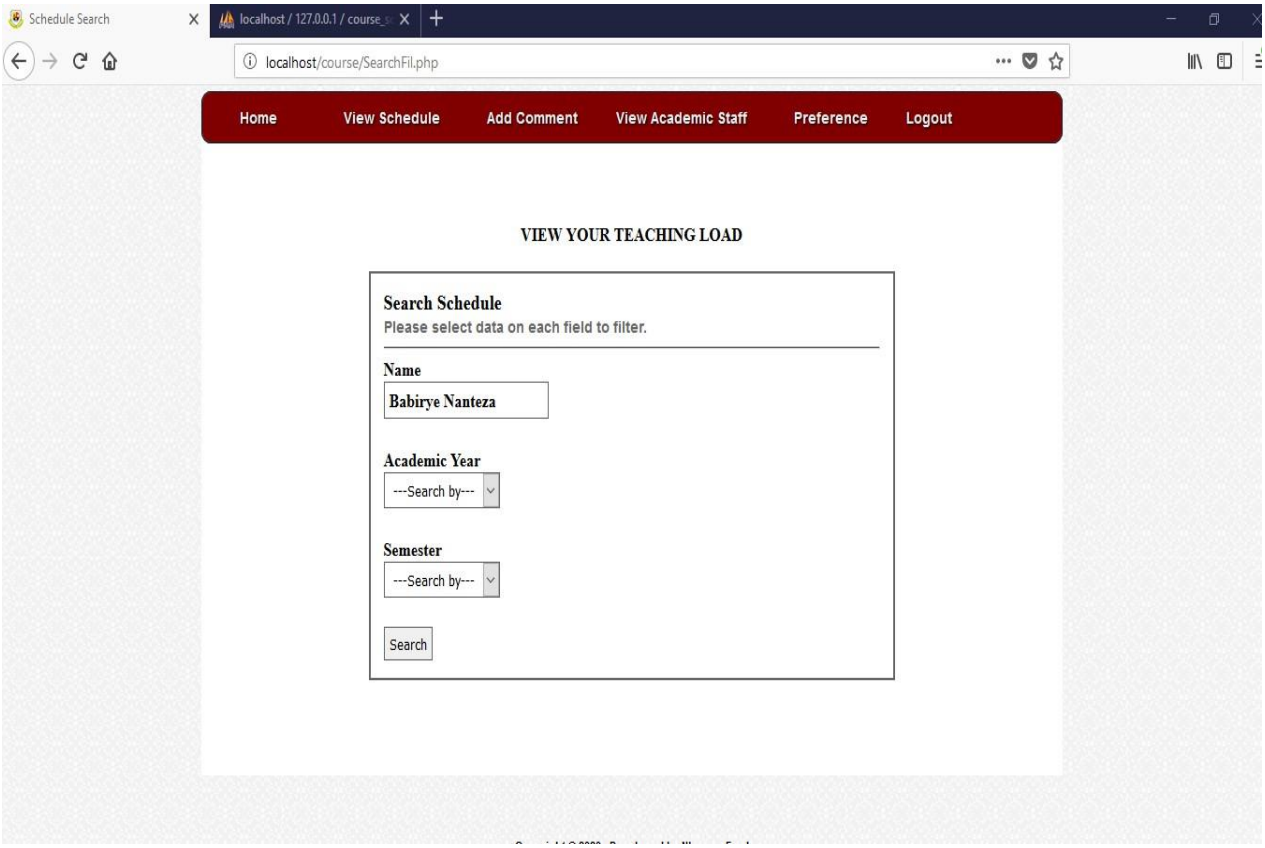


Figure 5.16: Search academic staff schedule

Academic staff schedule report

http://localhost/course/Filter.php

When the academic staff has applied a search for his/her teaching load by filtering according to the academic year and semester, the report below is displayed.

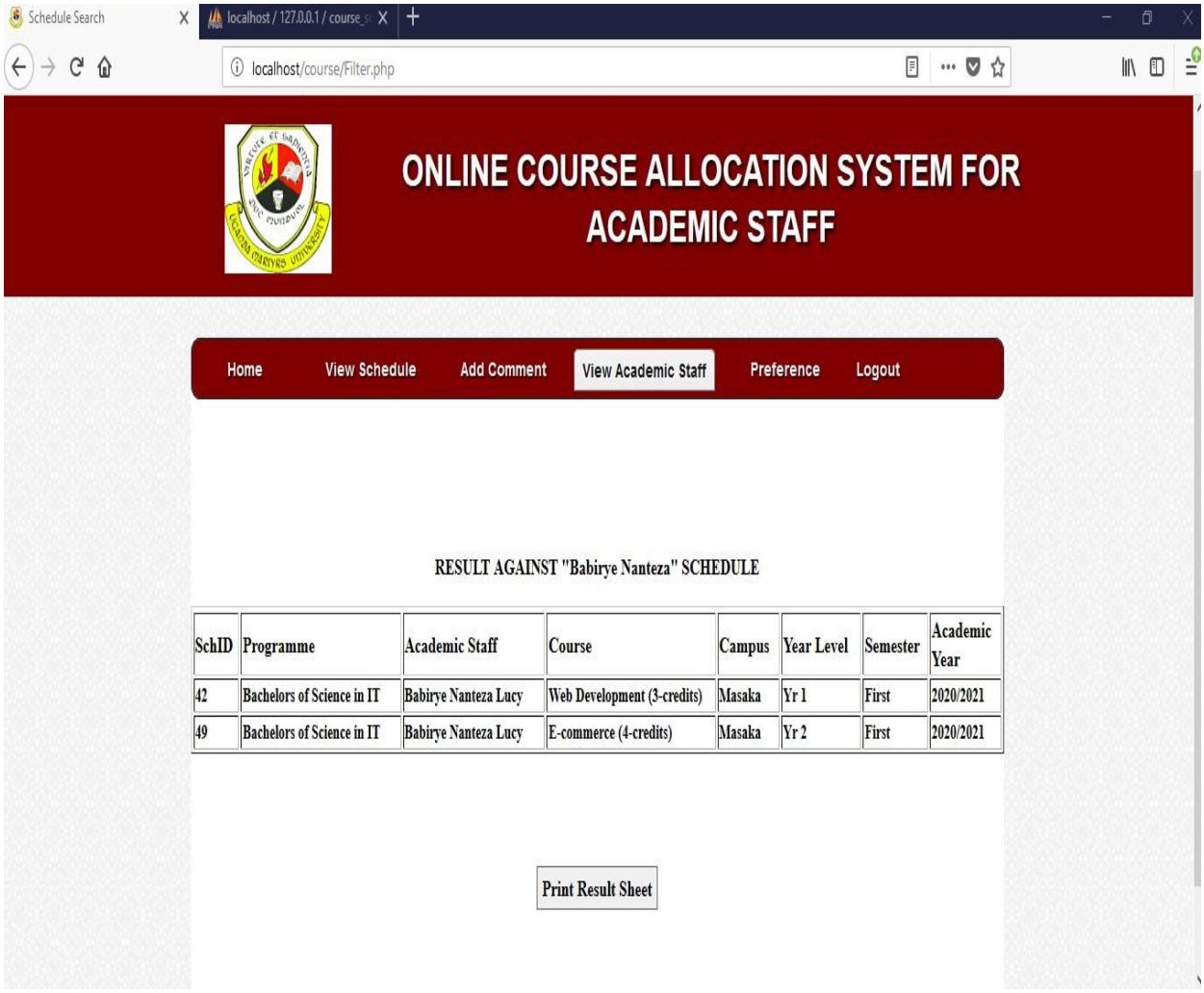


Figure 5.17: Academic staff schedule report

5.3.2.8 View academic staff profile

http://localhost/course/lecturerProfile.php

This script enables the academic view his/her profile registered in the system by clicking on the link" View Academic Staff", a criteria according to the name of that academic staff is picked from the logged-in session which is then used to query records against that academic staff from the database as shown below.

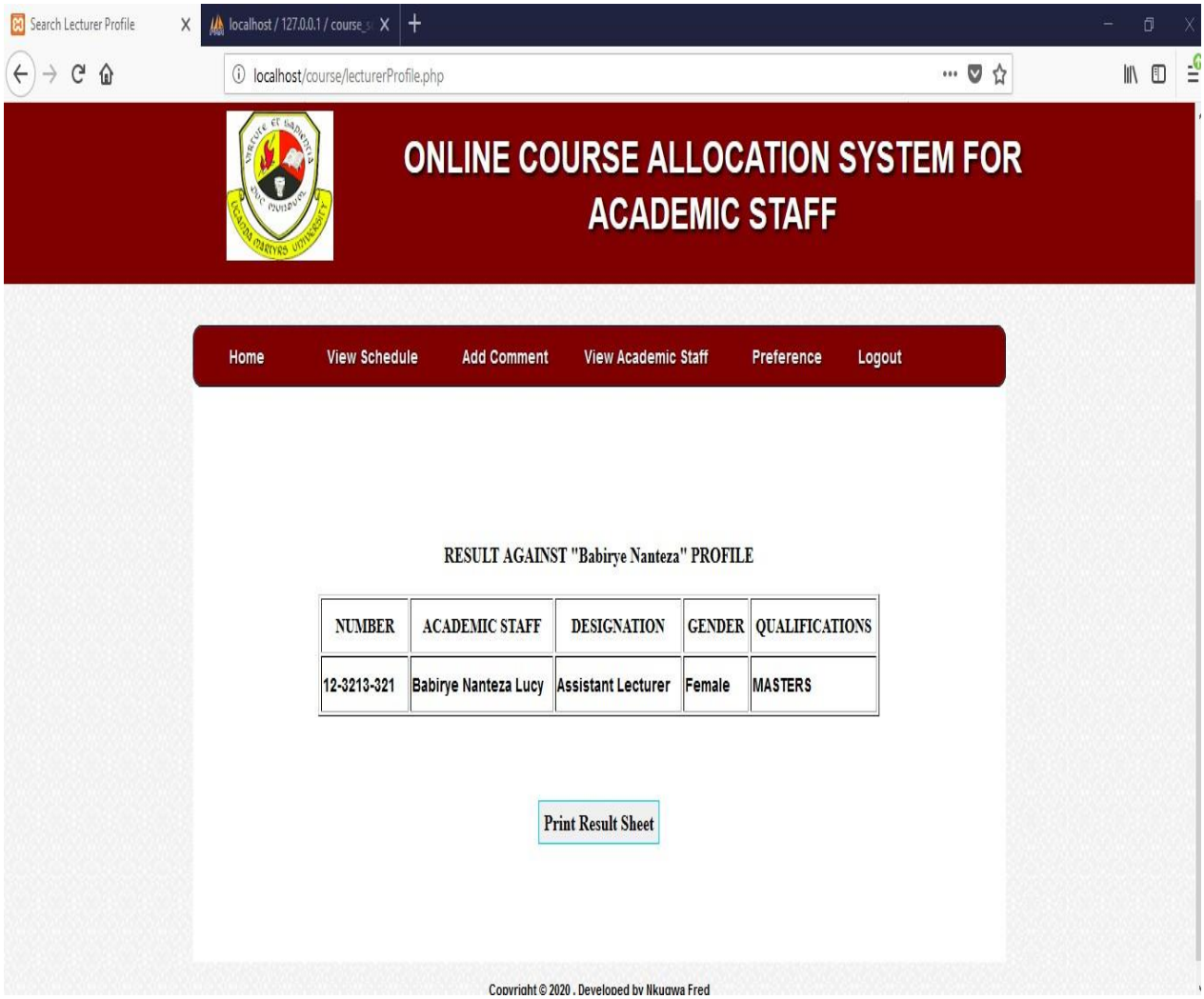


Figure 5.18: Academic staff profile report.

5.3.2.9 Register Preference

http://localhost/course/Preferences.php

This script makes it possible for the academic staff to register his/her preferred courses and area of specialization as shown in the figure below.

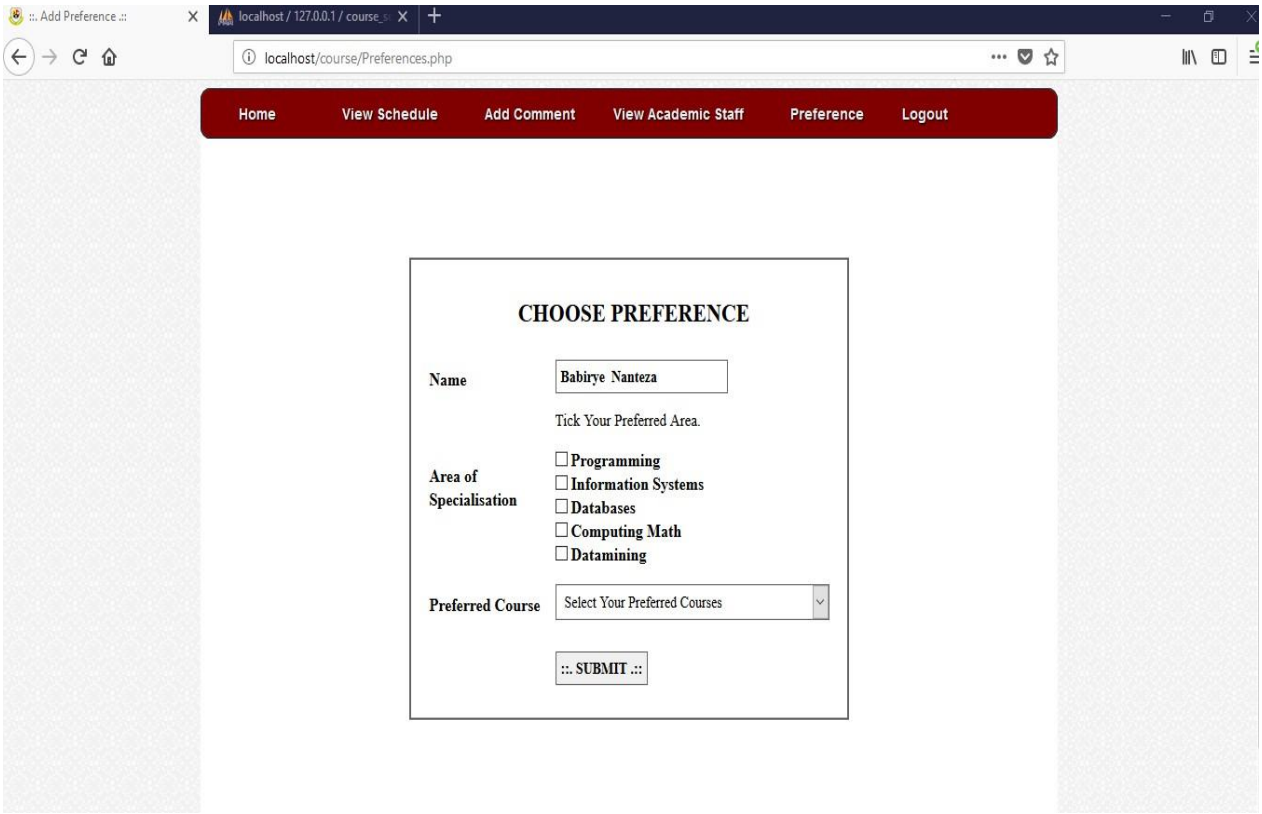


Figure 5.19: Register preference.

Post a comment

http://localhost/course/Comment.php

This page enables the academic staff to post his/her comment concerning the scheduled courses to the head of department for reallocation by clicking on the link "Add Comment" as shown below.

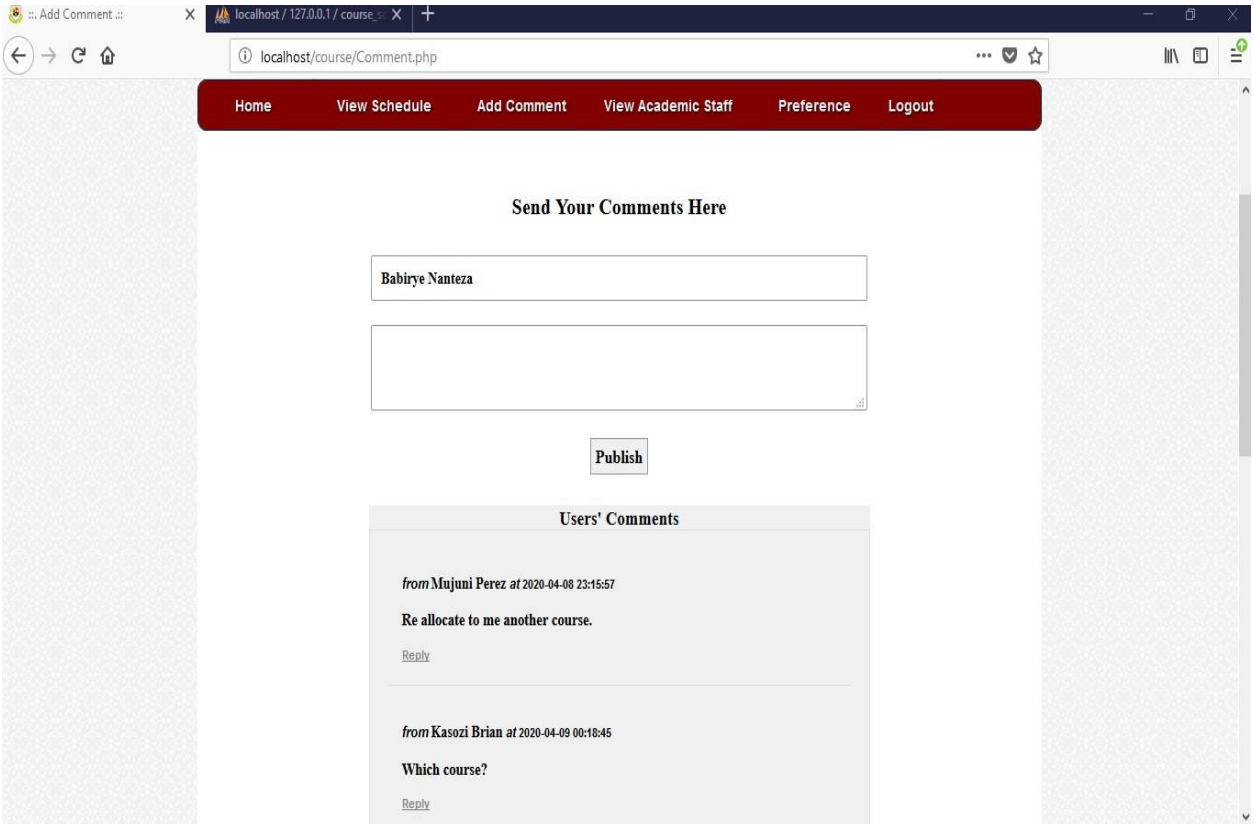


Figure 5.20: Post a comment page

5.4 System Testing

In this section, the functionalities of the system were tested to ensure that the system was working according to the specifications and also ensure that the set objectives for the project were met. This involved unity testing and testing validity of the results with data input, performance, checking code for any errors and authentication.

5.4.1 Validation

The system has the security features of ensuring proper form validation. When filling in forms, the system requires that all fields have to be validated before the user can submit the form. Each text field is preset to accept specific values and if wrong values are input, the error message window pops-up rejecting the input values. The screenshots below show the feature of form validation.

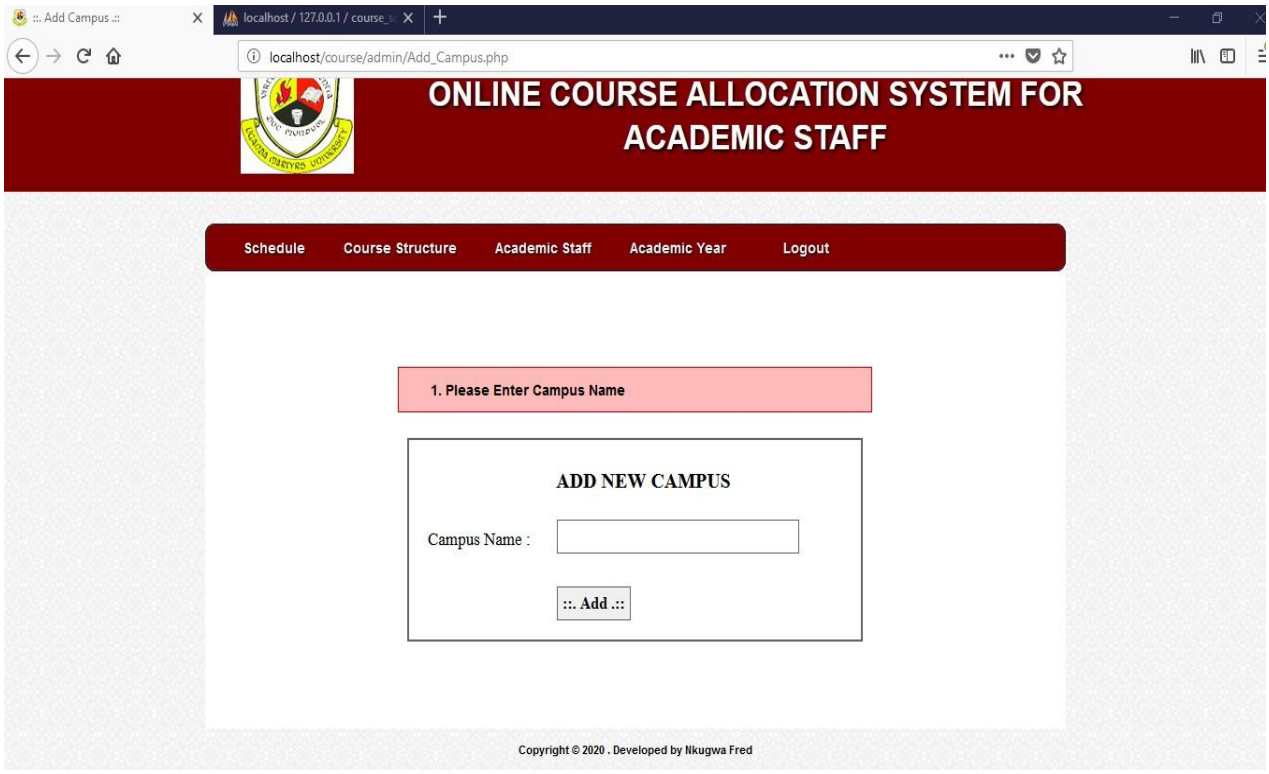


Figure 5.21: Validation of Input Data in the system

5.4.2 Unit testing

This was carried out by testing individual system codes, it was done by inputting test data to test their functionality against the pre-set required functionality.

5.4.3 Integration testing

This was carried out to test how compatible the individual components of the system can be integrated to work together as one complete system.

5.5 Conclusion

This chapter focused on the implementation of the OCASFA system and the presentation of the results or the final system design.

CHAPTER SIX

DISCUSSION, SUMMARY, RECOMMENDATION AND CONCLUSION

6.0 Introduction

This project was aimed at developing an online course allocation system for academic staff for UMU that can minimize time and costs incurred during the course allocation process, and also improve the quality of education. This chapter explains the researcher's discussion, conclusion after implementation of the new system, research summary, challenges faced, ethical research guidelines that were considered and recommendations as shown below.

6.1 Ethical Considerations

These were efforts to ensure that data was collected but not affecting the participants in any way. The researcher ensured that ethical requirements are upheld throughout the exercise. The researcher also ensured that the respondents made their own decisions to participate in the based on adequate knowledge, he made sure that the respondents; were informed about the procedures of the study, know the purpose, duration, benefits, and risks of their participation. The respondents were also assured of their privacy in the study. The respondents had the right to ignore the items that they don't wish to respond to.

6.2 Challenges Faced

During the course of this research project, the researcher met several challenges and these include;

System coding was a major challenge whereby the researcher had to learn and understand the required scheduling constraints to come up with a clear algorithm in PHP and MySQL that can satisfy those constraints. With limited knowledge and programming skills, this limited several functionalities that the researcher could implement into the system.

During the data collection stage, few respondents were met because the university is a busy institution. This made it very hard for the researcher to collect data from all targeted people. Also, the researcher learnt that some people are hard to deal with and others are easy.

Thirdly financial limitation, the researcher faced a challenge of unpredicted expenditure. This included high transport costs to meet the respondents at the main campus (Nkozi), internet costs where the researcher had to subscribe for dial-up internet connection to continue with the project during the country lockdown, thus had no access to the university Wi-Fi.

6.3 Discussion

This project will provide an alternative to the manual course allocation system for academic staff that is currently used by administrators at Uganda Martyrs University. It will ease the work of allocating courses to the academic staff whereby courses are effectively allocated without clashes hence reducing the time taken to come up with course schedule in the department and costs incurred during the course allocation process. It will make it easier to update the schedule and have the changes effected and made conveniently accessible by all stakeholders where ever they are(online).

The developed system provides a solution to the problems of the manual course allocation system such as allocation mistakes, repetitiveness, delays in the department activities since it directly solves them. Therefore, if the new system is adopted, it will improve the educational processes within Uganda Martyrs University and other multi-universities who would like to enhance their business operations. However, this system requires high-speed internet connection to provide better services.

6.4 Summary

This sub-section presents a summary of the study that was carried out at Uganda Martyrs University as part of the requirements for the award of a Bachelor's Degree in Information Technology. It presents the aim of the study, objectives, and how they were achieved as shown below.

The study aimed at minimizing the time and costs incurred during the course allocation process through the development of OCASFAS. To achieve this, the researcher set objectives to analyze and investigate the existing course allocation system for academic staff to identify the loopholes, challenges and requirements necessary for the development of OCASFAS which was done in chapter four during the analysis of the current system. Questionnaires, interviews and document

analysis were the fact-finding techniques used during this phase. The data collected about the current system was analyzed and presented using the frequency tables to derive requirements for the new system. And system design was done in chapter four in order to come up with models representing the system.

The implementation of the system was then done whereby the system design was transformed into a working system, this was done in chapter five.

Testing the implemented OCASFAS was done in chapter five during testing documentation so as to ensure that it meets the expected performance and user requirements set.

6.5 Recommendations

This research study will greatly improve the head of the department's work of allocating courses to the academic staff, easily update schedule and also have the changes effected and made conveniently accessible by all stakeholders where ever they are. However, this research also uncovered new areas that need further study. Therefore, the researcher recommends the need for further development of the OCASFAS in the areas shown below.

The developed system is currently limited to the faculty of science due to the limited time in which the project was planned to be accomplished and it was also very hard for the researcher to develop a system for the entire university. Therefore, the researcher recommends that future researchers should continue to improve this system so that it can take into consideration of other faculties at Uganda Martyrs Universities. Hence improving the educational processes within the university.

The sending of the notification messages module was out of the scope of this project. However, the researcher recommends that this module should be integrated into the future version of the OCASFAS. When the head of the department is done with the work of allocating courses to the academic staff, the dean has reviewed the schedule, the deputy vice-chancellor has approved the schedule then the human resource manager is tasked with publishing the schedule to academic staff. Therefore, the system should be incorporated in the away that it can automatically send a notification message to alert the academic staff about their new workload by email.

6.6 Conclusion

This chapter involves a summary of all other chapters that were discussed earlier in the study. It also includes areas that need further research to improve the study, ethical considerations that were followed to ensure that data was collected but not affecting the participants in any way. This chapter also takes into account the challenges faced by the researcher and lessons learnt from the whole study process.

References

Bhaumik, S., 2015. *Bootstrap essentials.* Birmingham, UK: Packet Publishing.

Bryman, A. &. B. E., 2007. *Business research methods.* 2nd ed. ed. Oxford: Oxford University Press.

Connolly, T. a. B. C., 2005. *Database Systems.* 4th ed. Harlow: Pearson Education Limited.

Dennis, L., 2011. *Study in Uganda.blogspot.com.* [Online]
Available at: https://studyinugandahe.blogspot.com/2011/08/uganda-martyrs-university.html
[Accessed 6th June 2020].

Dunn, O. J., 2009. *Basic statistics.* Hoboken, N.J.: John Wiley & Sons.

Ekhosuehi, V., 2016. University Course Allocation in a Department Using Linear. *Journal of Mathematics and Applications.*

Fall, S. N., 2016. *Design and Implementation of a Classroom Allocation System Prototype,* Texas: The Department of Computing Sciences.

Galanc, T. K. W. a. P. J., 2011. A quantitative method supporting the management of the teaching process. *Operations Research and Decisions,* 1(21), p. 33 – 51.

Hayajneh, Y., 2007. Systems & Systems Theory. *Management for Health Care Professionals Series.*

Hoffer, J. A. &. G. J. F., 2008. *Modern Systems Analysis and Design.* 5th ed. ed. New Jersey.: Pearson Education,.

Ishtiaq, M., 2019. Book Review Creswell, J. W. (2014). Research Design: Qualitative, Quantitative and Mixed Methods Approaches (4th ed.). Thousand Oaks, CA: Sage. *English Language Teaching,* 12(5), p. 40.

Jeremy, M. M., 2012. *Development of a STUDENTS ATTENDANCE AND RESULTS,* s.l.: Dissertation.

Kasozi, A., 2003. *University education in Uganda : Challenges and opportunities for reform.* Kampala, Fountain Publishers.

Khaled Mesghouni, S. H., 2004. Evolutionary Algorithms for course scheduling. *International Journal of Applied Math and Computations,* 14(1), p. 103.

Kiconco, C., 2011. *DESIGN AND DEVELOPMENT OF AN ONLINE STUDENT INFORMATION SYSTEM,* s.l.: Masters Dissertation.

Knotters, M., 2012. Purposive versus random sampling for map validation: a case study on ecotope maps of floodplains in the Netherlands. *Ecohydrology*, 6(3), pp. 425-434.

Kothari, C., 2004. Research Methodology Methods and Techniques. *New Age International Publishers, New Delhi. - References - Scientific Research Publishing*, Volume 2nd Edition.

Nelson, H., 2003. *The design way: intentional change in an unpredictable world.* New Jersey: Educational Technology Publications.

Nicholas , . O. O. & Ogwara, N. . O., 2018. Intelligent based Multi-Agent Approach for University Timetable Scheduling System. *international Journals of Computer Applications*, 182(2).

Rauf, F., Selvan, Y. K. & Adnan, Z., 2018. Course Allocation System for Lecturers. *International Journal of Computer Applications,* February, 180(22), pp. 10-14.

Rudova H, M. T. M. K., 2011. Complex university course timetabling. *Journal of Scheduling* , 14(2), pp. 187-207.

Saunders, M. e. a., 2007. *Research methods for business students.* 4th ed. London: Prentice Hall..

Scott, G., Grebennikov, L. & Johnston, K., 2007. Study of Australian Multi-Campus Universities. *Journal of Institutional Research,* p. 23.

Sheikh, M. A., 2012. *A Prototype of Lecturer Course Allocation System,* Malaysia: Universiti Utara.

Siame, A., 2018. University Course Timetabling using Bayesian based Optimization Algorithm. *International Journal of Recent Contributions from Engineering, Science & IT (iJES),* 6(2), p. 14.

Silva.D, A. & Ribeiro, M. e. a., January-March 2017. International Council for Distance Education. *Open Praxis Journal, vol. 9 issue 1,*, p. pp. 109–120.

Soyemi, J., Akinode, J. & Oloruntoba, S., 2017. Automated Lecture Time-tabling System for Tertiary Institutions. *International Journal of Applied Information Systems (IJAIS),* 12(5), p. 9.

Stensaker, B., 2017. Academic development as cultural work: responding to the organizational complexity of modern higher education institutions. *International Journal for Academic Development,* 23(4), pp. 274-285.

T.D.Nguyen, 2018. *Intelligent Systems Design and Applications: 1.* 8th International Volume 1 ed. India: Springer.

Taherdoost, H., 2016. Sampling Methods in Research Methodology;How to Choose a technique for Research. *International Journal of Academic Research in Management,* 5(2), p. 27.

Tetnowski, J., 2015. Qualitative Case Study Research Design. *Perspective on Fluency*, 25(1), p. 45.

Thomas M.Connolley, C. E. B., 2005. *Database systems, a practical approach to design, implementation, and management.*. United States Of America: Pearson Education Ltd.

Virtue, E., 2016. UNIVERSITY COURSE ALLOCATION IN A DEPARTMENT USING LINEAR PROGRAMMING TECHNIQUES. *The Journal of the Mathematical Association of Nigeria,43,* pp. 1-11.

Wiley, 2015. *Systems Analysis & Design, An Object Oriented Approach with UML*. 5th ed. United States of America: Don Fowley.

Zina H, B. A. R. A. ,. M. M. "., 2017. A Multi-Agent System for Course Timetable Generation. *TEM Journal*, Volume 8, pp. 211-221.

APPENDICES

DATA COLLECTION INSTRUMENTS

UGANDA MARTYRS UNIVERSITY

FACULTY OF SCIENCE

APPENDIX A: QUESTIONNAIRE

TOPIC: ONLINE COURSE ALLOCATION SYSTEM FOR ACADEMIC STAFF IN MULTICAMPUS UNIVERSITIES

Dear Respondent,

I am Nkugwa Fred a student at Uganda Martyrs University pursing a bachelor's degree of Science in Information Technology. I am carrying out a study by use of a questionnaire meant for undergraduate dissertation to be presented to the Faculty of science in partial fulfillment of the requirements for the award of Bachelors degree of science in Information Technology of Uganda Martyrs University and you have been randomly selected to participate. I therefore request you to spare some time and honestly answer the questions asked in this questionnaire. The answers will be used to provide the requirements for the design and development of an **Online Course Allocation System for Academic Staff in Multi-Campus University**. The information provided will be treated with utmost confidentiality.

1. SECTION A: BACKGROUND INFORMATION

i) Gender:

[1] Male [2] Female

ii) Faculty/Department …….....**Years of Service**.....

iii) Lecturer Category:

[1] On-campus/Full-time [2] Off-campus/part-time and/or distance

2. SECTION B: INFORMATION ACCESS ABOUT COURSE SCHEDULE DETAILS

i) How easy is it to have access to a computer connected to the internet?

[1] Very easy [2] Easy [3] Not easy [4] Difficult [5] Very difficult

ii) How do you access your information about your assigned workload for the semester?

[1] Electronic mail [2] Physically check on the university staff notice board [3] On what sup

iii) If there are course allocation mistakes made during the schedule process and changes are made by the head of department, how do you get that information?

[1] Receive email containing updates [2] Posted on the University Website [3] Posted on what sup

[4] Hear say [5] Go to the Department Head's office directly

iv) In your own opinion, what would be the best way to avail access to course schedule information?

[1] Online via course allocation system on login [2] Receive a letter through the post office

[3] Posted on what sup [4] Physically check on the staff notice board [5] 1 and 3

v) Given the opportunity to choose between Online access to my assigned courses to teach per semester information and physically coming to check on the Staff notice board to access the same, I would choose Online access than physically checking on the staff notice board.

[1] Strongly agree [2] Agree [3] Disagree [4] Strongly disagree

Please indicate the reasons for your choice.

……

……

……

SECTION C: PROBLEMS FACED WITH INFORMATION ACCESS

i) Does the institution have an online course allocation system for the lecturers (a web-based interface) where you can find all the assigned workload for courses to be taught by a lecturer without physically coming to UMU?

[1] Yes [2] No

[ii] If no, currently where do you access lecturer's schedule (assigned workload) information.

[1] Faculty office [2] Staff notice board [3] On what sup [4] Electronic Mail

ii) Do you sometimes get inconvenienced in planning your teaching schedule because your workload(schedule)information is delayed?

[1] Yes [2] No

[iii] If Yes, what do you think are the causes of such delays?

[1] Processing is done manually? [2] Checking on the notice board takes long time

[3] No online system to access this information [4] All the above

iv) What problems do you face when managing assigned workload at different campuses of UMU?

……

……

……

v) Are you satisfied with the way medium through which access to course schedule information is availed?

[1] Yes [2] No

vi) If a new online course allocation system for the lecturers is put in place, what would be your expectations from it?

[1] View my profile information

[2] View courses assigned to me

[3] View program changes online

[4] View and print my assigned courses and workload information online after login.

[5] Username and Password protection

[6] All the above

[ii] Any other expectation

………

………

………

THANK YOU

APPENDIX B: INTERVIEW GUIDE

Questions were formulated and written down prior to the process of obtaining data. These are guidelines during the interviewing of the administrators (Head of department, Human Resource Manager and ICT Staff) of Uganda Martyrs University.

ADMINISTRATORS

Dear Respondent,

I am Nkugwa Fred a 3rd year student at Uganda Martyrs University pursing a bachelor's degree of Science in Information Technology. I am carrying out a study by use of an interview meant for undergraduate dissertation to be presented to the Faculty of science in partial fulfillment of the requirements for the award of Bachelors degree of science in Information Technology of Uganda Martyrs University and you have been randomly selected to participate. I therefore request you to spare some time and honestly answer the questions that I am going ask in this interview. The answers will be used to provide the requirements for the design and development of an **Online Course Allocation System for Academic Staff in Multi-Campus Universities.** I promise that the information provided will be treated with utmost confidentiality.

Name: ……. Faculty……

Responsibility……...Rank……..

The following questions do not make up the full questionnaire for the interviews, they are simply guidelines to ensure a certain level of consistency.

1. What does the respondent know about the current system of allocating(scheduling) courses to be taught by lecturers in Uganda Martyrs University? How is it done
   ……
   ……
   ……
   ……

2. What are the current methods/tools used during course allocation?

……
……
……
……

3. How do lecturers access information about their assigned workload for the semester?

……
……
……
……
……

4. Does the respondent think the current system is efficient? Why?

……
……
……
……

5. What are the processes involved during course allocation for the lecturers?

……
……
……
……
……
……

6. How is lecturer's schedule data stored and retrieved?

……
……
……
……

7. What problems do you face when assigning courses to lecturers at different campuses of UMU?

……
……
……
……
……
……

8. Does the institution have an online course allocation system for the lecturers (a web-based interface) where you can find all the assigned workload for courses to be taught by a lecturer without physically coming to UMU?

[1] Yes [2] No

9. Does the respondent think there is need for a new computerized (web-based interface) system of managing these records at the university?

[1] Yes, As soon as possible

[2] Yes, Sooner or later

[3] NO

[4] Indifferent

10. What can be done to ensure that the proposed system works as expected?

……
……
……
……
……
……
……
……

THANKS FOR YOUR TIME

APPENDIX C: TIMELINE

ID	Task Name	Duration	Start	Finish
1	Developing a Concept Paper	21 days	Mon 9/2/19	Mon 9/30/19
2	Developing a Project Proposal	44 days	Tue 10/1/19	Fri 11/29/19
3	Data Collection and Analysis	44 days	Mon 12/2/19	Thu 1/30/20
4	Implementation Phase	55 days	Fri 1/31/20	Thu 4/16/20
5	Writing a Project Report	32 days	Fri 4/17/20	Mon 6/1/20