EEJOURNALS OPEN ACCES
EURASIAN EXPERIMENT JOURNAL OF ENGINEERING (EEJE) ISSN: 2992-409X
©EEJE Publications Volume 4 Issue 1 2023

Online Student Result and Archive Management System in KIU-College of Applied Sciences and Technology

Ogwal Robert Rubble

Faculty of Applied Sciences and Technology Kampala International University

ABSTRACT

This study was based on online student results and an archive management system, a case study of KIU-college of applied sciences and Technology under the following research objectives: to understand how the existing system worked in order to identify requirements for the new system, to design a student record archiving system that would help students and parents view or print academic information without necessarily coming to the university, to implement the designed student record archiving system, and to test and validate the implemented student record archiving system for any errors and to ensure that it meets the user requirements. This study employed a cross-sectional research design strategy, data was collected using researcher-devised questionnaires and the following findings were registered. In regard to the existing system, the study established the following weaknesses: Inaccurate data capture and recording resulting from the collecting of incomplete manifests and other relevant data hence unreliable data which resulted in the processing of wrong information, Loss in productivity because staff members spent valuable time moving from place to place in search of data to be analyzed, too much reliance on students who may at their own discretion deliberately or unknowingly conceal vital information and a lot of paperwork involved which resulted into errors and inconsistent results. In regard to the requirements of the proposed system, the study found two types of requirements which included functional and non-functional requirements. The system was designed using PHP flat form and Wamp Server as the backend application. The system passed through a series of successful testing which included among others unit testing, module testing, sub-system testing and acceptance testing. It was later validated and the system was found able to authenticate authorized users only, capture, store and retrieve information. The research objectives of this system were effectively achieved. The challenges of the existing system were readily addressed by the provided user requirements. Recommendations for the developed system included: operation on Windows operating system flat form, hardware requirements (CPU= 40GB, RAM= 1GB or higher, Processor speed= 1 GHz or higher), used with Wamp server 2.0 or higher, used under PHP Framework and user training should be implemented so that users may not find the system difficult to use though it is user friendly.

Keywords: Student record archiving system, Online student results, Wamp Server, System, PHP flat form.

INTRODUCTION

Institutions of higher learning in Uganda maintain their own student records, besides the records kept by the Uganda National Examination Board (UNEB) and the Ministry of Education and Sports (MoES). The MoES is responsible for the central collection, processing, and utilization of data and information; and coordinating it for effective decision-making on the national level [1-10]. It is also responsible for formulating education policies in the

country as well as rendering educational services through its various bodies and departments. Institutions of higher learning require a system that will enable them to extract the appropriate information from the available data that they receive from the students' former schools/institutions during admission [11-14]. The absence of such a system in the country has not only led to the loss of student information but has also limited the sharing of this information in the country. Such a system is urgently required to track an individual student's progress and performance at every level of education. Such a system makes it quicker and easier for managers to identify, access and retrieve this information [15-19]. For such a system to run efficiently, it must integrate organizational objectives into its strategic framework [1]. This study will therefore develop a Student Result and Archiving Management System which is an educational portal for university institutions. This online portal will handle everything from enrollment of students to making individual results of students available online directly on the school database. School management will be in a position to tell at a glance how many students are there in the school at any given point in time. The beauty of it all is that it will help make students' results available to them from anywhere in the world provided there is internet connectivity. All they will need to do is log in to their account and view/print their results from the database. Student Result and Archive Management System will be able to turn manual means of results output to digital over an online hosting platform. It will be in a position to automatically compute cumulative scores and grades according to figures being keyed in. It will be in a position to call up a particular student's result at any point in time. Students will be able to view results several years after graduation. It will help reduce the rate of human errors, for example, the Mis-calculation of cumulative scores. It will also be able to automatically award remarks to students based on grades.

Statement of Problem

The university since its inception has been facing student result-related challenges. The university has for a number of times had complaints from students of their missing/lost results (coursework and examination marks). This problem has left many students repeating (retaking) the same course unit or at worse failing to graduate. This system was therefore intended to solve the above problem by storing students' records in an archive database that would help track students' records in case of any complaints.

Aim

To develop a web-based system that would enable the College of Applied Sciences and Technology (KIU Main campus) to keep students' records in an archive setting long even after they have graduated. This would help both the students and the parents to print or view this information even after a long period of time without stepping on the university premises.

Specific objectives:

i. To understand how the existing system works in order to identify requirements for the new system.
ii. To design a student record archiving system that would help students and parents view or print academic information without necessarily coming to the university.
iii. To implement the designed student record archiving system.
iv. To test and validate the implemented student record archiving system for any errors and to ensure that it meets the user requirements.

Research Questions

i. What are the challenges faced by the existing system?
ii. What are the user requirements for the proposed system?
iii. How will the new system be implemented?
iv. How will the new system be tested and validated?

METHODOLOGY

Research design

This study employed a cross-sectional research design strategy. According to [2], a cross-sectional survey design is a research design where data is collected to make inferences about a population of interest. The researcher used this strategy because it provides a numeric description of some part of the population, and describes and explains events as they are, as they were or as they would be. This enabled the researcher to rapidly collect data and provided him with the ability to understand a population from part of it.

Study Population

The target population will include a total of 50 respondents from the College of Applied Sciences and Technology. This will include 10 lecturers, 30 students and 10 lab assistants. The lecturers will be involved in the study because

they are the ones who face the challenges of recording and keeping students' marks. The lab assistants will be included in the study because they will be the ones in charge of maintaining the system. The students will be included in the study because they are the ones affected by the old system where there is rampant loss of marks exam and coursework marks.

Research Instruments

The research tools that were utilized in this study included the following: a face sheet to gather data on the respondents' profile (gender, age, educational level, and work experience); researcher-devised questionnaires were used to understand the existing system and determine the requirements for the proposed system.

Data Gathering Procedures

i. After the proposal was approved, the researcher secured a list of qualified respondents from the College of Applied Sciences and Technology and selected through systematic random sampling from this list to arrive at the minimum sample size.
ii. The respondents were explained about the academic nature of the study.
iii. The respondents were requested to answer completely and not to leave any part of the questionnaires unanswered.
iv. The researcher emphasized retrieval of the questionnaires within five days from the date of distribution.
v. On retrieval, all returned questionnaires were checked if all were answered.
vi. The data gathered were tallied, encoded into the computer and statistically treated using the Statistical Package for Social Sciences (SPSS).

Data Analysis

The frequency and percentage distribution were used to determine the profile of the respondents and the challenges facing the existing system.

Planning the Development Process

The consideration here defined a product life cycle model. The software lifecycle encompassed all activities required to define, develop, test, deliver, operate and maintain a software product [3].

Waterfall Model

The waterfall model is a sequential design process, often used in the software development process, in which progress is seen as flowing steadily downwards (like a waterfall) through the phase of conception, initiation, analysis, design, construction, testing, production/implementation and maintenance[4]. The waterfall model is illustrated in Figure 1 below:

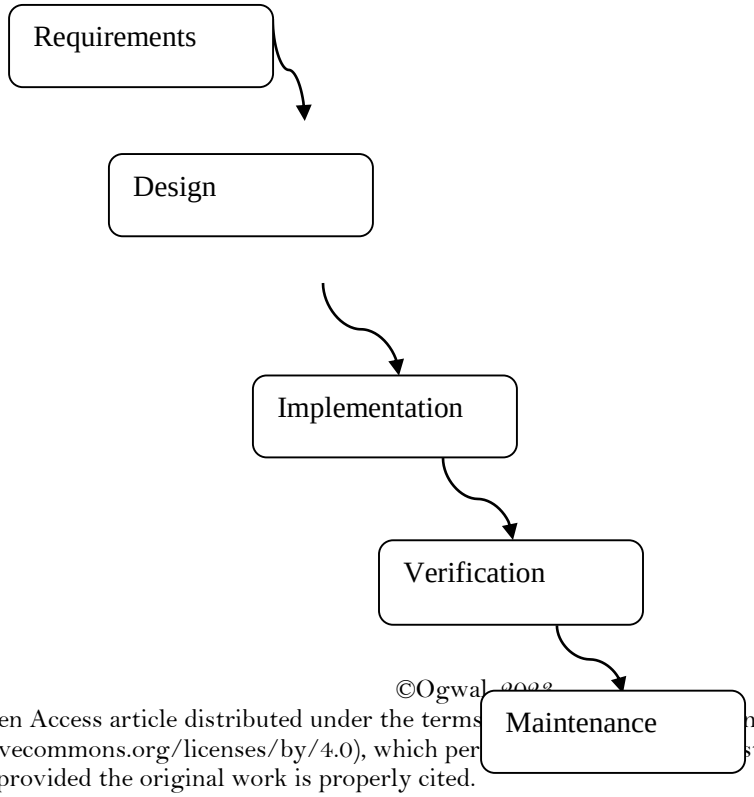


Figure 1: shows the waterfall model

Preliminary Development Schedule

This study used a preliminary development schedule to estimate its system design

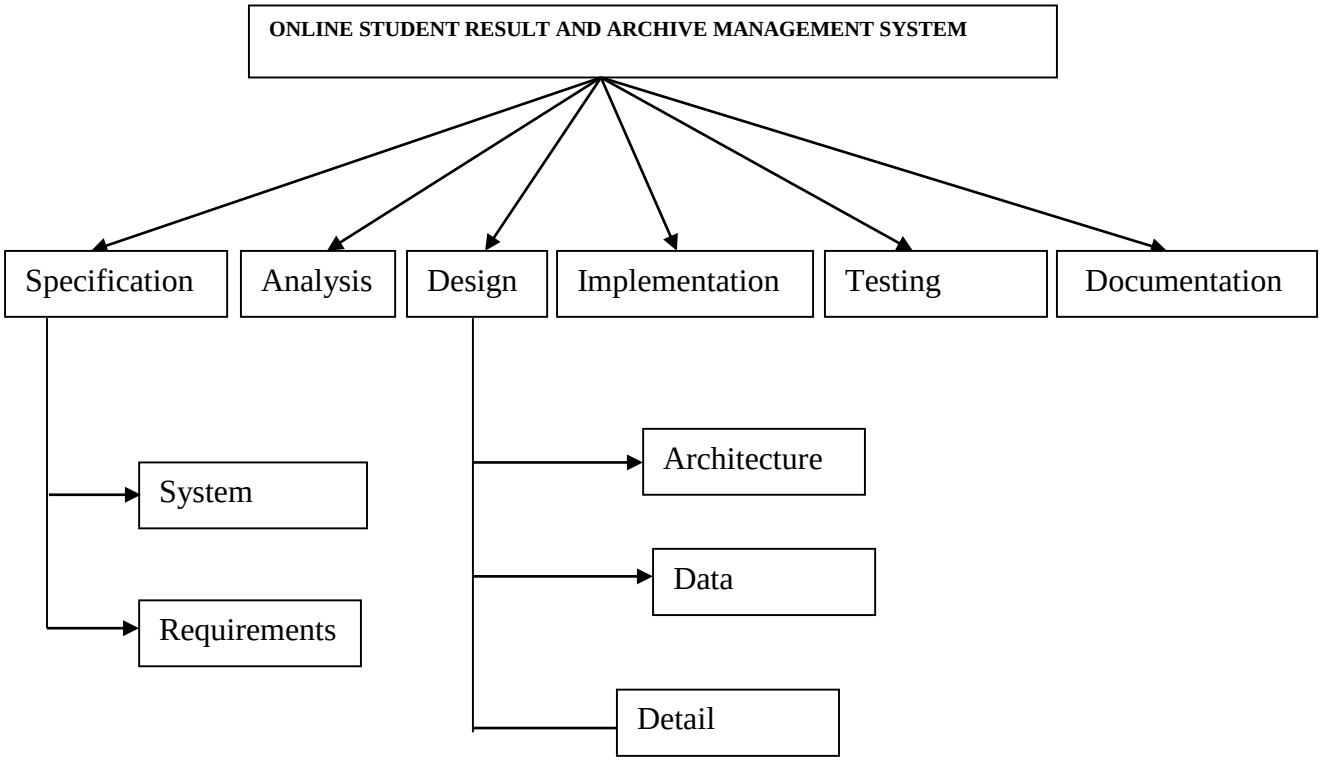


Figure 2: Shows the work breakdown structures Tools that were used

Table 1: Shows tools that were used

Software Requirements	**Description**
PHP programming language	For designing the front end plate form
Wamp Server (Mysql)	For developing the database
Microsoft word 2007	For documentation
Operating system	Windows 7
Hardware Requirements	**Description**
CPU (central process unit)	Pentium IV
Processor speed	2.0 GHz

RAM (Random Access Memory)	1.0 GB
Hard disk	500 GB

Ethical Considerations

To ensure the confidentiality of the information provided by the respondents and to ascertain the practice of ethics in this study, the following activities were implemented by the researcher:

i. The respondents were coded instead of reflecting their names.
ii. The researcher solicited permission through a written request to the principal of the College of Applied Sciences and Technology.
iii. The researcher acknowledged the authors quoted in this study through citations and referencing.
iv. The researcher presented the findings in a generalized manner.

RESULTS

Study of the existing system

The study was carried out at the Offices of the College of Applied Sciences and Technology (KIU Main campus). The purpose of this study was to develop a web-based system that would enable the College of Applied Sciences and Technology to keep Students' records in an archive setting long even after they have graduated. This would help both the students and the parents to print or view this information even after a long period of time without stepping on the university premises. The research was to evaluate the existing system's effectiveness, reliability and performance in order to identify user requirements.

Weaknesses of the Existing System

i. There is a demand for storage space for paperwork. This space could be utilized to accommodate other activities such as offices or for expansion of the staff room.
ii. Inaccurate data capture and recording resulting from the collecting of incomplete manifests and other relevant data hence unreliable data which may result in the processing of wrong information.
iii. Loss in productivity because staff members spend valuable time moving from place to place in search of data to be analyzed.
iv. Too much reliance on students who may at their own discretion deliberately or unknowingly conceal vital information.
v. A lot of paperwork is involved which can result in errors and inconsistent results.
vi. The paper-based system in use gives little opportunity to share data across the units in the department. This is because each unit has its own files with contents relevant to it.
vii. Same data kept on the subject in different files is inconsistent and therefore lacks integrity and may be unreliable.
viii. Retrieval of information is tedious and time-wasting. It is therefore necessary to automate the system so as to overcome the various shortcomings associated with the current system.

Proposed system

Functional requirements

These are the statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations [5]. They explain what the system should do and what it entails:

I. The system will authenticate authorized users of the system.
II. The system will enable the administrator to capture and store details about the users.
III. It will be in a position to automatically compute cumulative scores and grades according to figures being keyed in.
IV. It will be in a position to call up a particular student's result at any point in time.
V. Students will be able to view results several years after graduation.
VI. It will help reduce the rate of human errors, for example, the Mis-calculation of cumulative scores.
VII. It will also be able to automatically award remarks to students based on grades.

VIII. Query the data in the database.
IX. Update, delete, refresh, exit and add are some of the functionalities that the system will have.
X. There is a need for sufficient hard disk space to manage the system, at least 60GB.
XI. Enough memory (RAM) is needed to ensure quicker and better responsiveness, at least 512MB.
XII. There will be a need for a medium that offers fast and efficient communication transmission that has a minimum error performance.

Non Functional Requirements

These are systematic qualities that defined the system's properties and constraints. They include the following,

Performance: The system will be expected to have a good response time in order to successfully perform data manipulation.

Security: The system will be expected to have security functionalities like the user name and password to avoid unauthorized users from accessing the system.

Accessibility

Efficiency, reliability, speed and retrieval of information needed will be made easier for the system users.

User-friendly

The system uses commands, for system navigation that the user will find easy to learn, because of reduced complexity.

Versatility

The system will be designed to fit other types of operating systems such as Windows 7 Vista, XP, Linux, etc.

Availability

Access to the system is a bit simplified to allow users to be able to start work as smoothly as possible. Maintenance by the system administrator should be done regularly to keep the system available to the users.

System Specification

This specifies the functionality of the system and the constraints in its operation. System specifications are intended to establish what services are required from the system and the constraints on the system's operation and development. This stage is very delicate because errors at this point inevitably lead to later problems in the system design and implementation [6].

In this research study, the researcher observed the following on system specification:

I. An estimate of whether the identified user needs may be satisfied using the current software and hardware technologies and whether the proposed system is cost-effective.
II. Deriving system requirements through observing the existing system, and discussing with potential users and procurers.
III. Defining a set of requirements and recording it to produce a requirements document.
IV. Checking the requirements for realism, consistency and completeness. Errors in the requirements document are inevitably discovered thus correcting them. The systems requirements document, also known as the functional specification, should be precise. It may serve as a contract between the system buyer and the software developer.

User Requirements of the New System

These are the statements in natural language plus diagrams of what services the system is expected to provide and the constraints under which it must operate. Readers of the user requirements are: - client manager (ICT manager), staff members (end-users) and system developers (researchers).

- Update records.
- Delete any records.
- Edit and refresh records.
- Add records that will need to be added to the system's database.
- Search for relevant records.
- Generate reports.

Security Requirements

These specify system behaviour that is disallowed rather than the behaviour that is expected of the system. The following security checks will be observed:

- Unauthorized users should not be allowed to access the database. This can be enhanced by the use of passwords and user names.
- Relevant information should be made available to the relevant people.
- Use of security measures such as firewalls and anti-virus to trap unwanted information that might harm the system.

Organizational Requirements

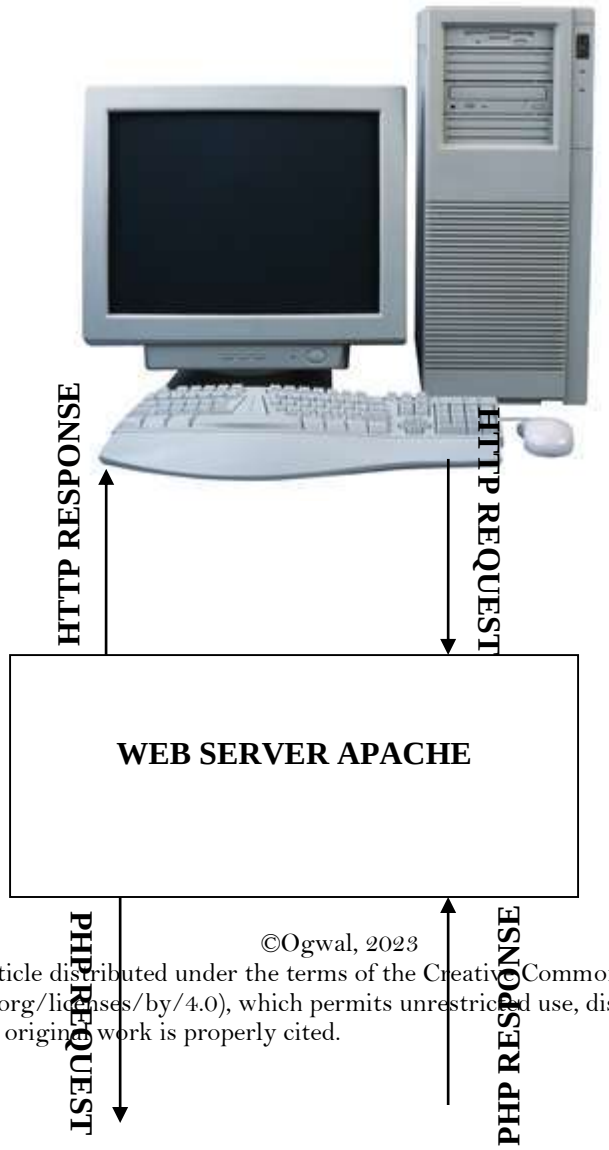
With the new system in place, there was a need to maintain the database. Therefore, some costs would need to be budgeted for in terms of operational and maintenance costs.

System Design

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

Architectural Design

The architecture design explains the different components of the system. This is illustrated in Figure 3 below.



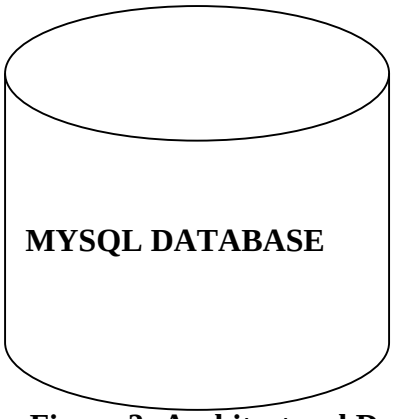


Figure 3: Architectural Design

Entity relationship diagram

The entity relationship diagram of the system is described in Figure 4. It illustrates the relationships between the different system entities.

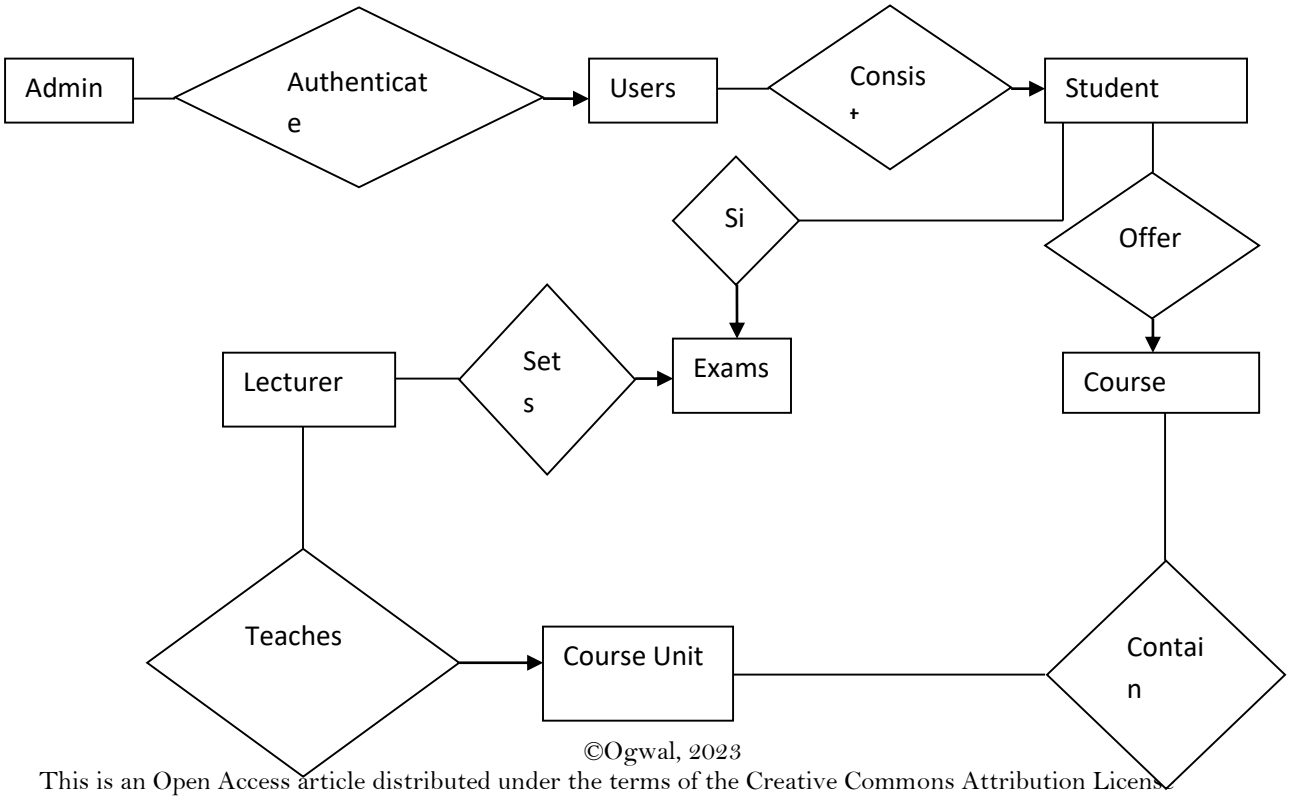


Figure 4: Logical database design

Is a relational schema mapped from classes to be used to design relational structures for database implementation.
i. Admin (id (Primary Key), Idnum, Password, Fname, lname, mname, birth, status, gender)
ii. Course (id (Primary Key), CourseCode, description)
iii. Student (id (Primary Key), fname, mname, lname, idnumber, password, course, yearlevel, section, gender, status, bday)
iv. Lecturer (id (Primary Key), fname, mname, lname, idnumber, password, work, gender, status, bday)
v. Course Unit (id (Primary Key), subject, section, course, level)
vi. Users (id (Primary Key), id number, password, position)

Physical Data Design

The physical database design deals with the description of how the database implementation will be carried out.

Table 2 Admin

Field Name	Datatype	Size	Constraints	Comments
Id	Int	11	Not Null	Auto-increment
Idnum	Varchar	30	Not Null	
Password	Varchar	30	Not Null	
Fname	Varchar	30	Not Null	
Lname	Varchar	30	Not Null	
Mname	Varchar	30	Not Null	
Birth	Varchar	30	Not Null	
Status	Varchar	30	Not Null	
Gender	Varchar	30	Not Null	

Table 3: Course

Field Name	Datatype	Size	Constraints	Comments
Id	Int	11	Not Null	Auto-increment
Course code	Varchar	30	Not Null	
Description	Varchar	30	Not Null	

Table 4: Student

Field Name	Datatype	Size	Constraints	Comments
Id	Int	11	Not Null	Auto-increment
Fname	Varchar	30	Not Null	
Mname	Varchar	30	Not Null	
Lname	Varchar	30	Not Null	
Idnumber	Varchar	30	Not Null	
Password	Varchar	30	Not Null	
Course	Varchar	30	Not Null	
Yearlevel	Varchar	30	Not Null	
Section	Varchar	30	Not Null	
Gender	Varchar	30	Not Null	
Status	Varchar	30	Not Null	
Bday	Varchar	30	Not Null	

Table 5: Lecturer

Field Name	Datatype	Size	Constraints	Comments
Id	Int	11	Not Null	Auto_increment
Fname	Varchar	30	Not Null	
Mname	Varchar	30	Not Null	
Lname	Varchar	30	Not Null	
Idnumber	Varchar	30	Not Null	
Password	Varchar	30	Not Null	
Work	Varchar	30	Not Null	
Gender	Varchar	30	Not Null	
Status	Varchar	30	Not Null	
Bday	Varchar	30	Not Null	

Table 6: Course Unit

Field name	Datatype	Size	Constraints	Comments
Id	Int	11	Not Null	Auto-increment
Subject	Varchar	30	Not Null	
Section	Varchar	30	Not Null	
Course	Varchar	30	Not Null	
Level	Varchar	30	Not Null	

Table 7: Users

Field Name	Datatype	Size	Constraints	Comments
Id	Int	11	Not Null	Auto-increment
Id number	Varchar	30	Not Null	
Password	Varchar	30	Not Null	
Position	Varchar	30	Not Null	

DISCUSSIONS

System Implementation

Systems implementation is the delivery of that system into production (meaning day-to-day implementation). The implementation phase delivered the production system into operation. The functional system from the construction phase was the key input to the implementation phase of the system. The deliverable of the implementation phase was the operational system, the operation and support stage of the life cycle [7].

System implementation involves:

- Conduct a system test

The system was tested to ensure that it operates as required to avoid inconveniences while under operation by the users. Different tests were carried out as explained below under program testing.

- Prepare a conversion plan

Once a successful system test was completed, the researcher began preparations to place the new system into operation. Using the design specifications, the researcher developed a detailed conversion plan for the new system.

- Install the database

To place the system into operation, the researcher loaded the database. The purpose of installing the database was to populate the new system's database with existing data from the old system. Each record loaded must be input, edited and confirmed before the database table was ready to be placed into operation. The researcher calculated the database size and estimated the time required to perform the installation. Data entry then kicked off. This task was crowned with restructured existing data that has been populated in the database for the new system. The new

system was now put into operation. The functional system from the construction phase was the key input to system implementation [8]. The users were trained using various manuals, and files and the database was loaded and the final testing was performed. System users provide continuous feedback as new problems and issues arise. To provide a smooth transition to the new system, a conversion plan must be prepared. This plan may call for an abrupt switch where the old system is terminated and replaced by the new system. Alternatively, the old system may run in parallel with the new system until the new system has been deemed acceptable to replace the old system.

Some of the screenshots of the proposed system

Figure 5: Home Page

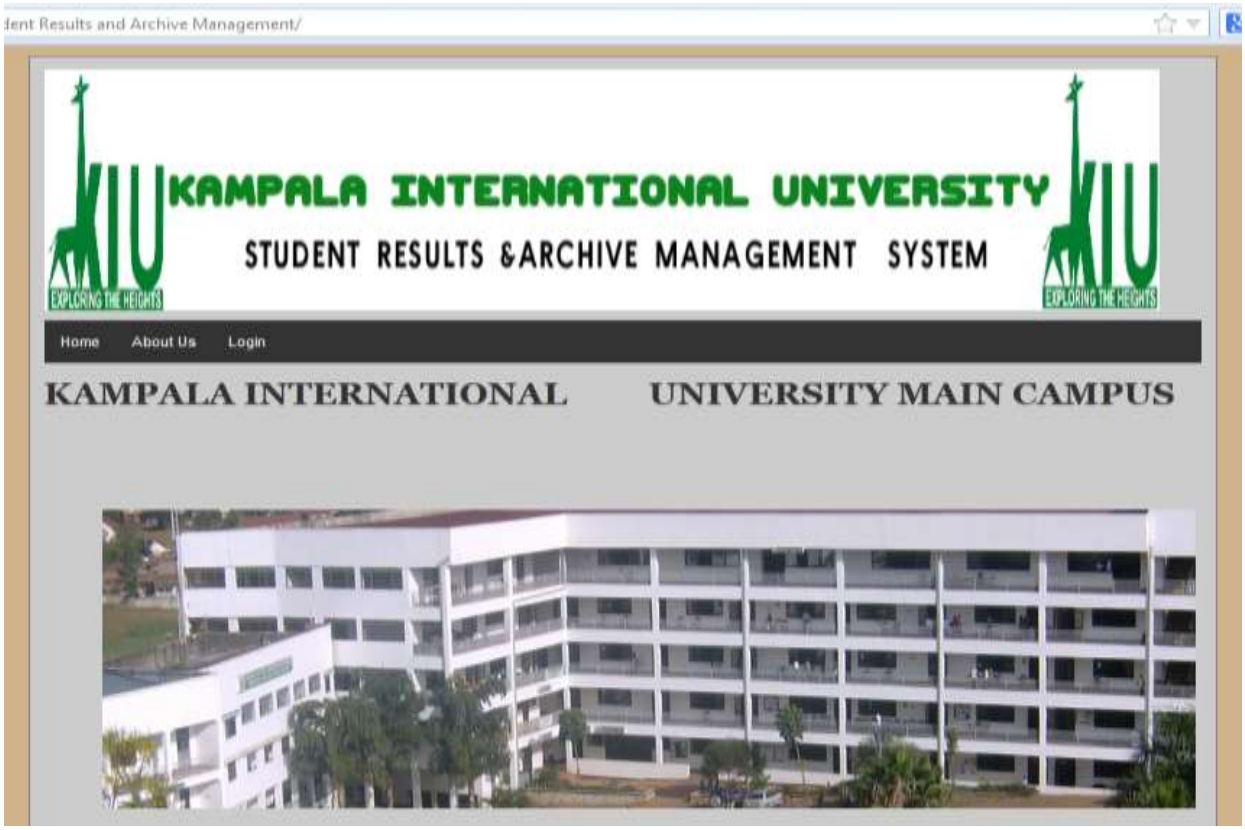


Figure 6: Login Interface

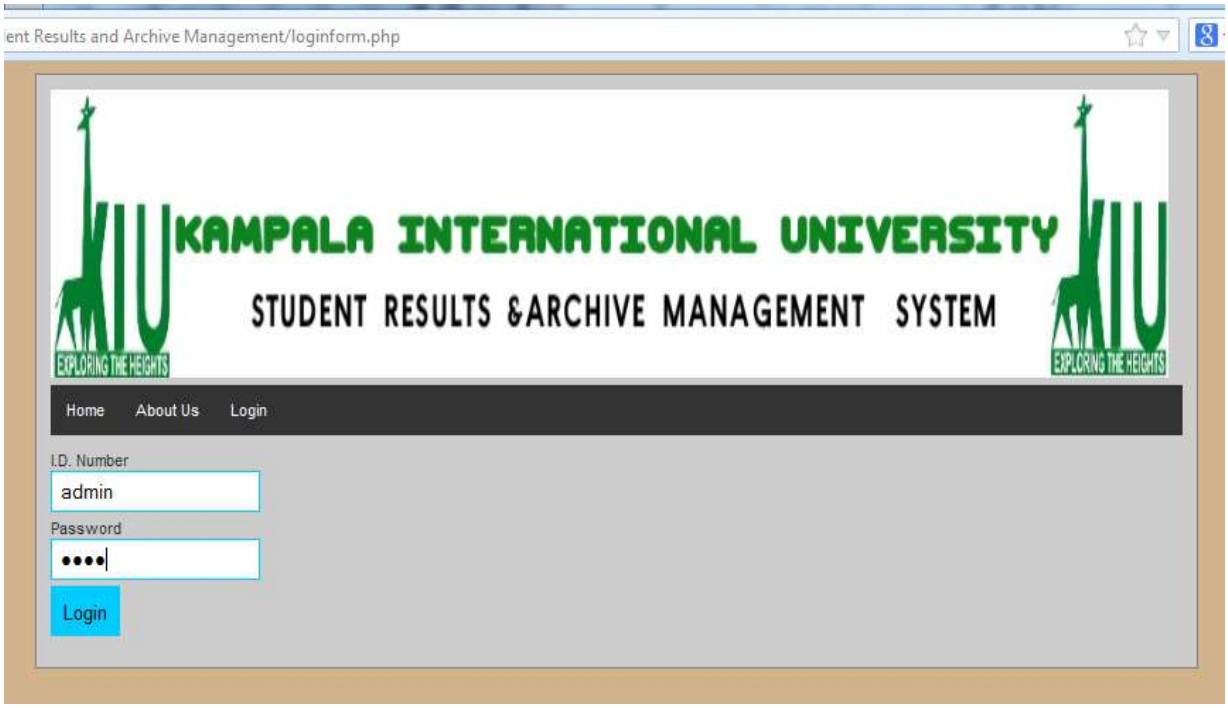


Figure 7: Admin Account

a) Profile

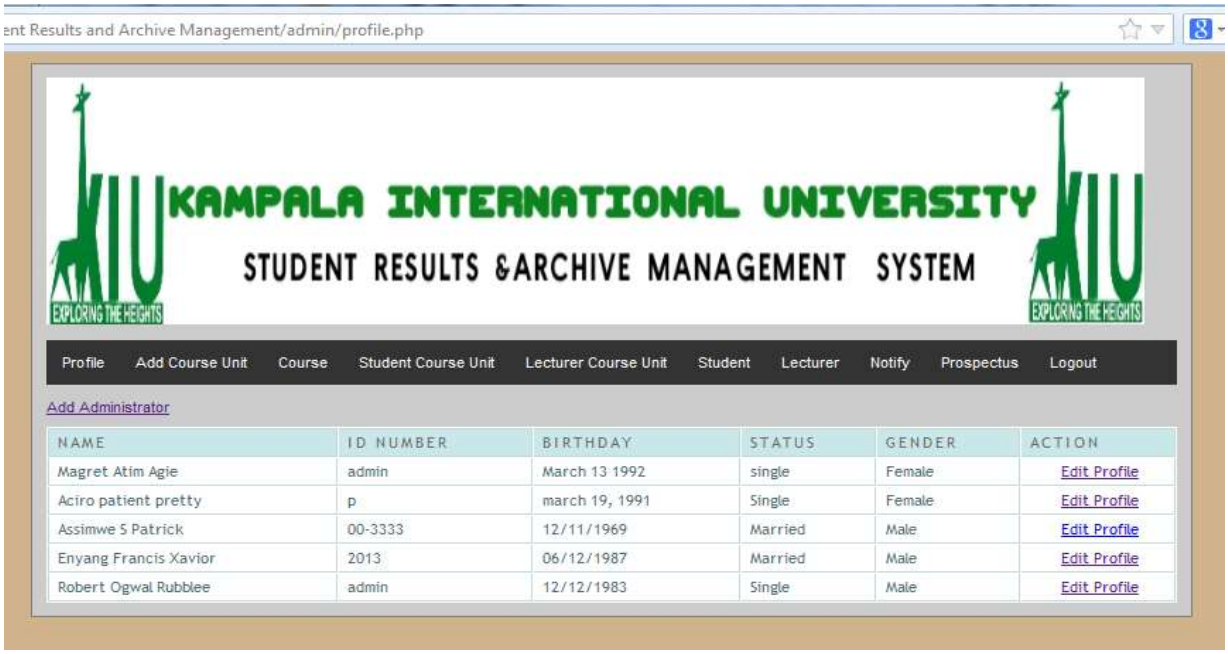


Figure 8: Add Admin

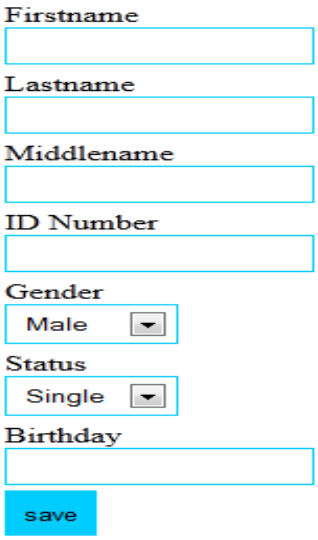


Figure 9: Add Course Unit

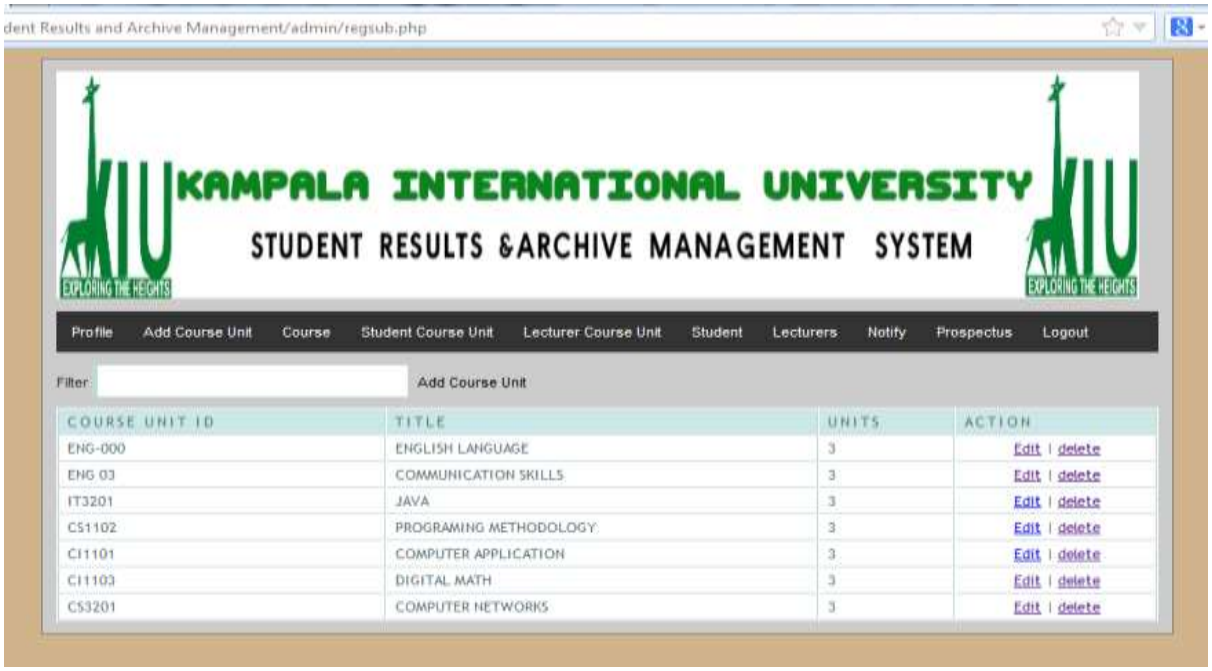


Figure 10: Student

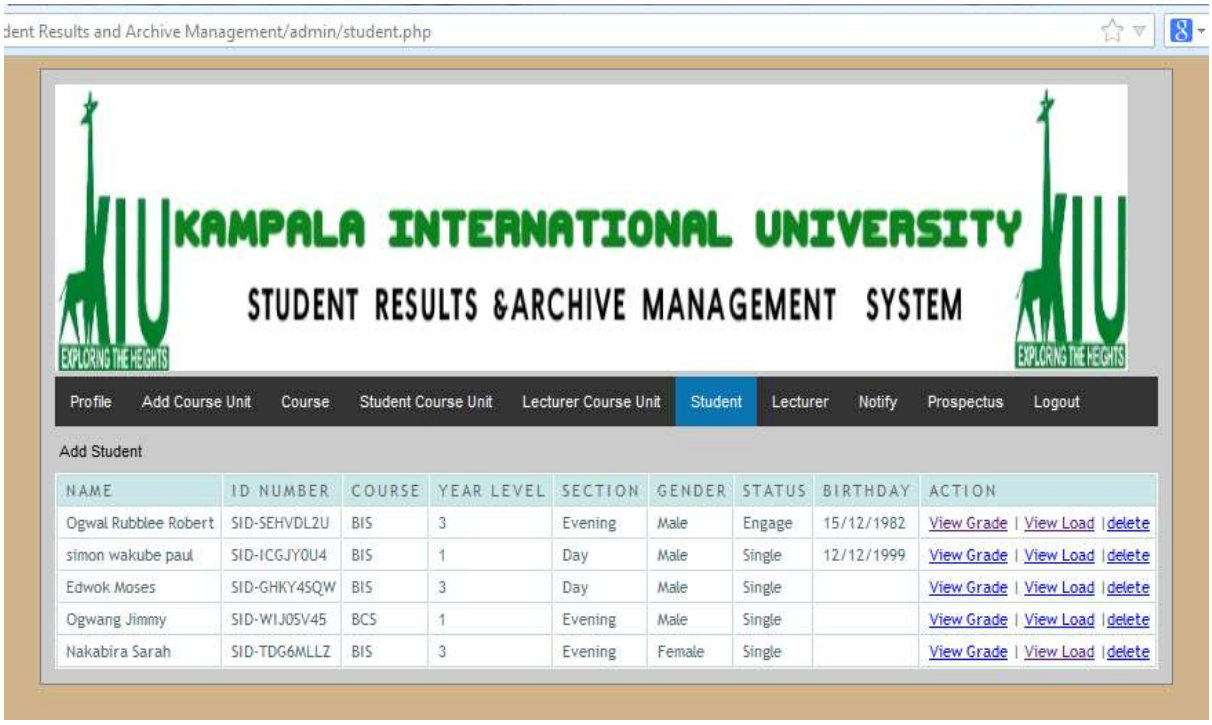


Figure 11: Add Student

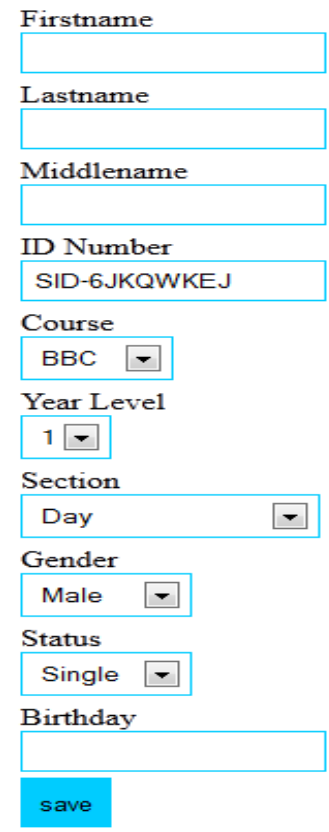


Figure 12: Lecturer

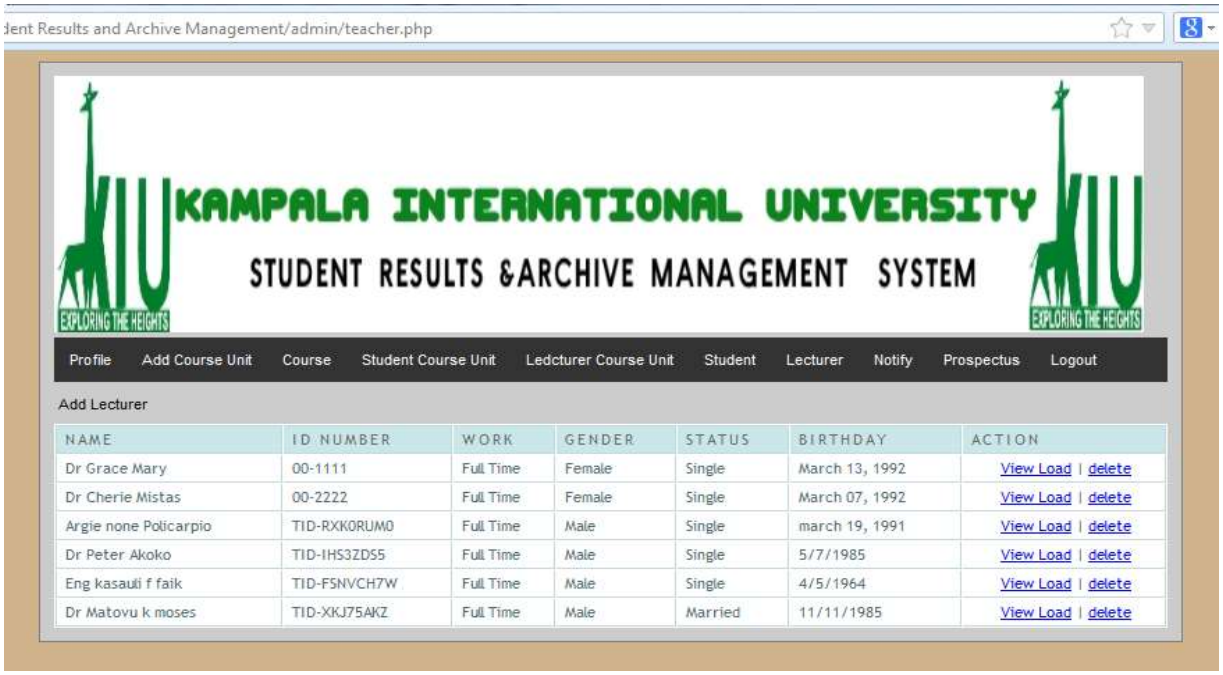
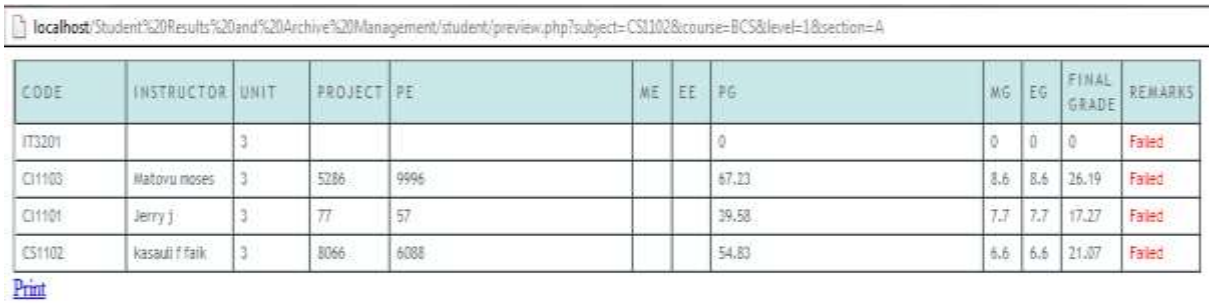


Figure 13: Student Account Profile

Figure 14: View Record

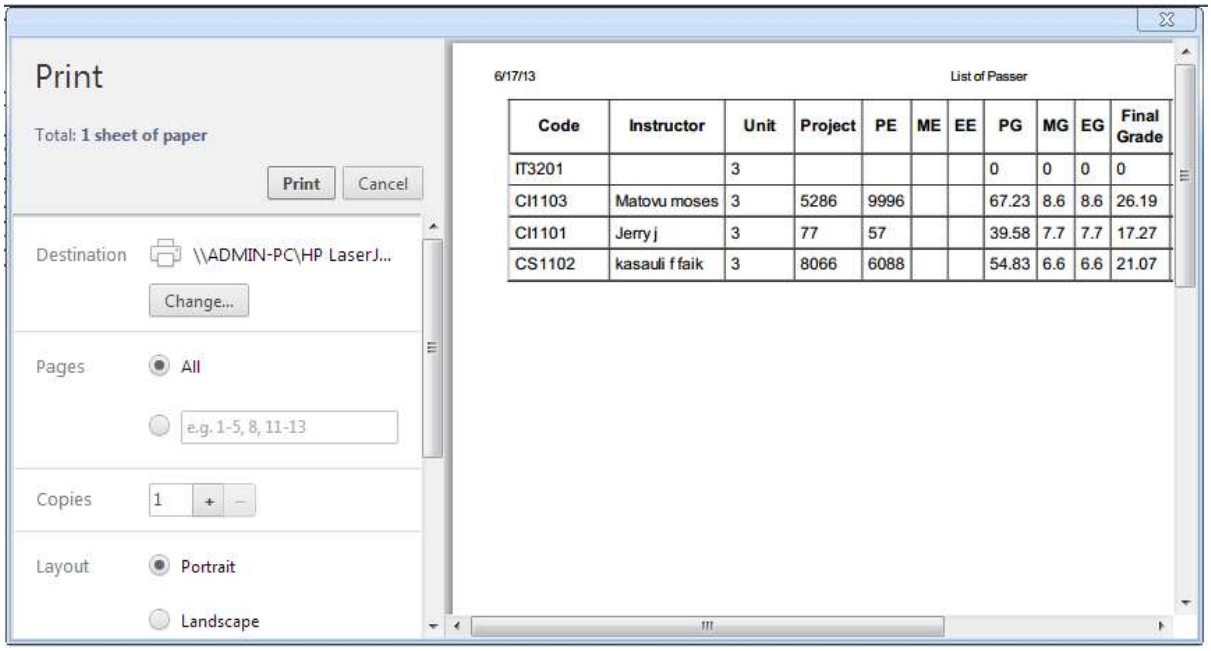


localhost/Student%20Results%20and%20Archive%20Management/student/preview.php?subject=CS1102&course=BCS&level=1&section=A

CODE	INSTRUCTOR	UNIT	PROJECT	PE	ME	EE	PG	MG	EG	FINAL GRADE	REMARKS
IT3201		3					0	0	0	0	Failed
CI1103	Matovu moses	3	5286	9996			67.23	8.6	8.6	26.19	Failed
CI1101	Jerry j	3	77	57			39.58	7.7	7.7	17.27	Failed
CS1102	kasauli f faik	3	8066	6088			54.83	6.6	6.6	21.07	Failed

Print

Figure 15: Print Result



Code	Instructor	Unit	Project	PE	ME	EE	PG	MG	EG	Final Grade
IT3201		3					0	0	0	0
CI1103	Matovu moses	3	5286	9996			67.23	8.6	8.6	26.19
CI1101	Jerry j	3	77	57			39.58	7.7	7.7	17.27
CS1102	kasauli f faik	3	8066	6088			54.83	6.6	6.6	21.07

Figure 16: Prospectus

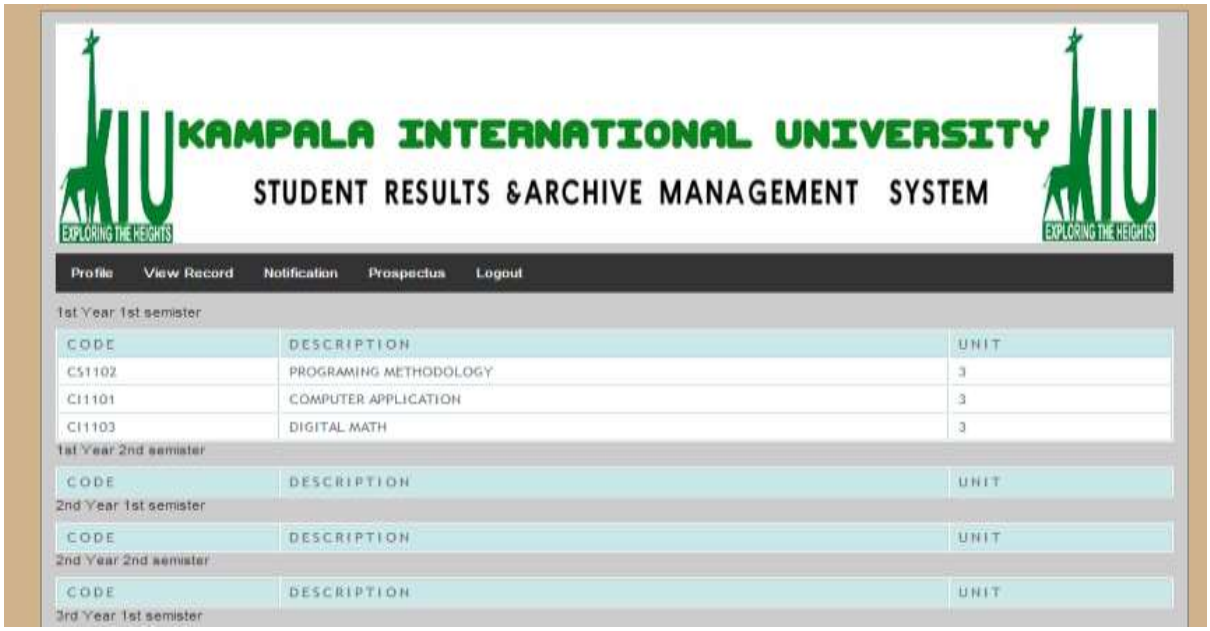


Figure 17: LECTURER ACCOUNT Profile

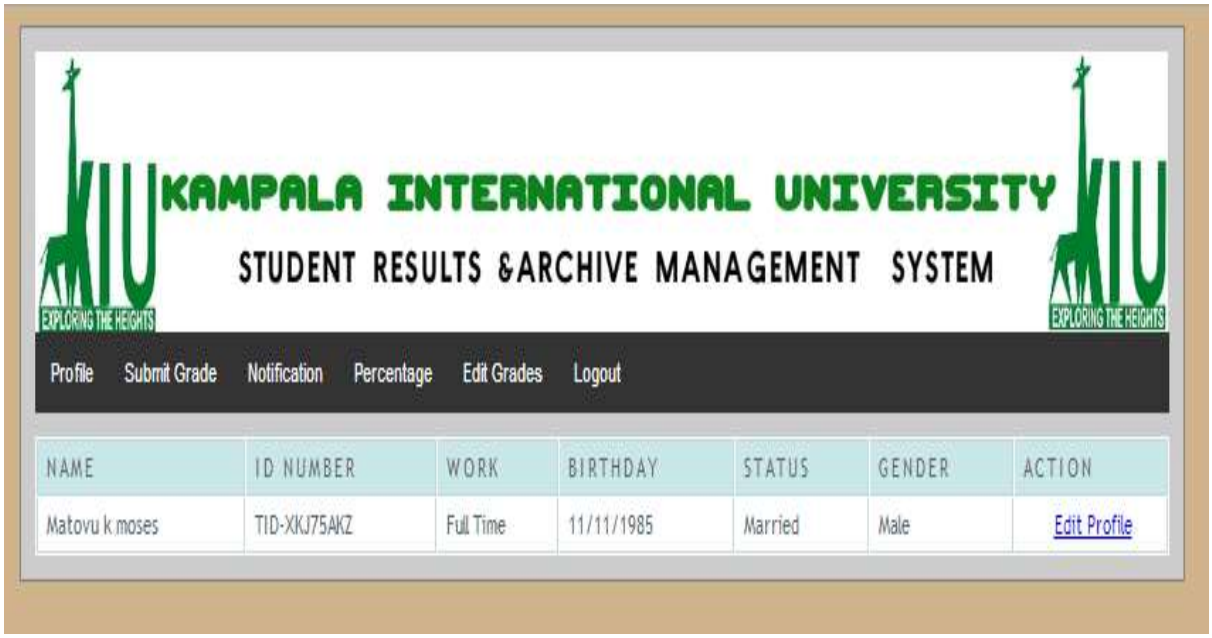


Figure 18: Submit Grades

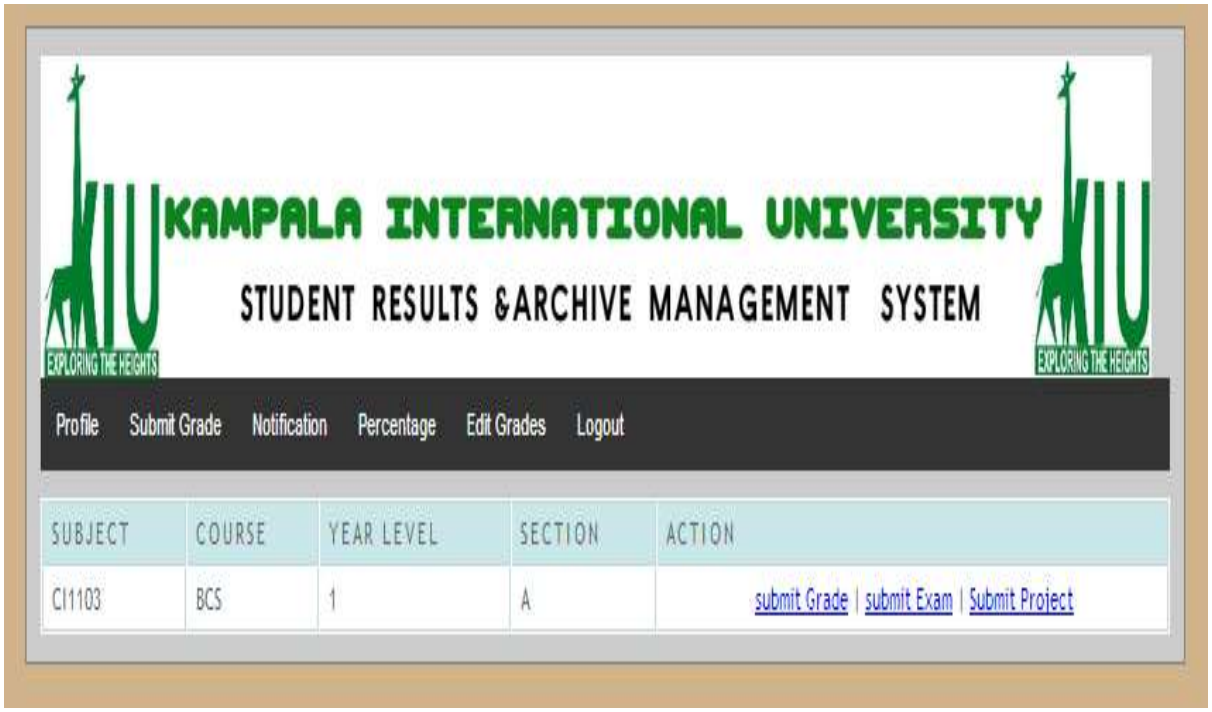
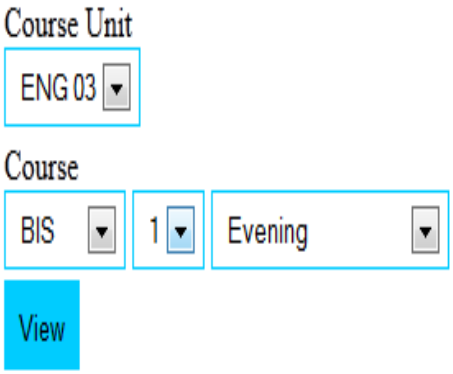


Figure 19: Percentage

Figure 20: Edit Grades



System Testing

This was intended to ensure that the system conforms to its specification and that the system meets the expectations of the users. Large systems, as the new system, are built out of sub-systems which are built out of modules which are composed of procedures and functions. The testing process proceeded in stages where testing was carried out incrementally with system implementation. The following were the stages followed:

- Unit testing**:** Individual components were tested to ensure that they operate correctly. Each component was tested independently without other system components.
- Module testing: A module encapsulates related components, thus tested without other system modules. Examples of modules are object class, abstract data types and collection of functions and procedures.
- Sub-system testing: A collection of modules was integrated into sub-systems and then tested. This concentrated on the detection of module interface errors by rigorously exercising these interfaces.
- System testing: The sub-systems were integrated to make up the system. This process was concerned with finding errors that result from manipulated interactions between sub-systems and sub-system interface problems. It was concerned with validating the system so that it would meet its functional and non-functional requirements and testing the emergent system properties.
- Acceptance testing: This was the final stage in the testing process before the system was accepted for operational use. The system was tested with data supplied by the system customer rather than simulated test data. Acceptance tests revealed errors and omissions in the system requirements definition because the real data exercise the system in different ways from the test data. It also revealed requirements problems where the system facilitations do not really meet the users' needs or the system performance was unacceptable.

System Validation

The system was validated to check whether it meets users' requirements

Table

Functionalities	**Yes**
Authenticate authorized users only	√
Capture, store and retrieve information	√
Provide reports	√

Strengths of the system

Based on the end users of the system, some of the strengths of the student archive management system include;
The system can only be accessed by authorized users.

I. It is in a position to automatically compute cumulative scores and grades according to figures being keyed in.
II. It is in a position to call up a particular student's result at any point in time.

III. Students are able to view their results several years after graduation.
IV. It automatically awards remarks to students based on grades.
V. The system is easy to learn and use.

Weaknesses of the system

Based on the end users of the system, some of the weaknesses of the Drug order system include:

- The system requires an internet connection which may be costly for the College of Applied Sciences and Technology and the end users.

Conclusion

The research objectives of this system were effectively achieved. The challenges of the existing system were readily addressed by the provided user requirements. The new system was then developed with the following capabilities:

- Allowing user authentication.
- Allow an administrator to register more users, inserting, modifying and deleting records.
- Preparing the reports automatically and printing them in PDF format.
- Providing reports according to the selected semester.
- Keeping all information about students and users.

The system was later tested and validated and could successfully perform the user requirements.

Recommendations

Recommendations for the developed system include;
The system should be used on Windows operating system flat form. This is because its initial design was done using a Windows operating system and therefore is compatible with any widow operating system. It might not work well with another operating system such as Linux or Unix because of their complexity and user unfriendliness. Should be used with the following hardware requirements met (Hard Disk= 40GB, RAM= 1GB or higher, Processor speed= 1 GHz or higher). This is because the system operates under a heavy graphical interface and therefore with a small hard disk and memory space, the computer can be forced to "freeze" or "hang. Should it be used with Wamp server 2.0 or higher? This is because the Wamp server is the backend database and the system's database itself was designed with a higher version of the Wamp server. Therefore, when a lower version is installed, some of the items may end up missing hence the system might not run effectively and the database will not perform its functionalities as required.

REFERENCES

1. Maher, William J. (1992). . Metuchen, New Jersey: Society of American Archivists & the Scarecrow Press, Inc.
2. Sammon, D. & Finnegan, P. 2000. The Ten Commandments of Data Warehousing. The Database Advances in Information Systems, ACM SIGMIS, 31(4): 83-91.
3. Hasan, H., Hyland, P., Dodds, D. & Veeraraghvan, R. 2000. Approaches to the Development of Multi-Dimensional Databases: Lessons from Four Case Studies. The Database for Advances in Information Systems, 31(3): 10-21.
4. Elisam Magara (2006). a framework for an integrated student information management system for higher education in Uganda.
5. Galbraith, V.H. (1948). Studies in the Public Records. London. p. 3.
6. Adams, C, Beath C.M (1991). Issues relating to the implementation and management of an integrated student achievement management system. Journal of the American Society for information science.
7. Collins, R and Straub, D (1991). The delivery of information services within a changing information environment. Journal of the American Society for information science 42 (20): 120-123.
8. The University of Pittsburgh: electronic records management. Bulletin of the American Society for Information Science. 1993 Oct; 20(1): 21.
9. Masisani,WM. and Adabara,I.(2022).Implementation of Smart Grid Decision Support Systems. IDOSR JOURNAL OF SCIENTIFIC RESEARCH 7 (1), 50-57.

10. Masisani,WM. and Adabara,I.(2022).Monitoring with Communication Technologies of the Smart Grid.IDOSR JOURNAL OF APPLIED SCIENCES 7 (1), 102-112.
11. Nabiryo,P. and AE,Itodo(2022).Design and Implementation of Base Station Temperature Monitoring System Using Raspberry Pi.IDOSR JOURNAL OF SCIENCE AND TECHNOLOGY 7 (1), 53-66.
12. Natumanya,A.(2022).Design and Construction of an Automatic Load Monitoring System on a Transformer in Power Distribution Networks.IDOSR JOURNAL OF SCIENTIFIC RESEARCH 7 (1), 58-76.
13. Masisani,WM. and Adabara,I.(2022).Overview of Smart Grid: A Review.IDOSR JOURNAL OF COMPUTER AND APPLIED SCIENCES 7 (1), 33-44.
14. Ugwu,CN. and Eze,VHU.(2023).Qualitative Research.IDOSR JOURNAL OF COMPUTER AND APPLIED SCIENCES 8 (1), 20-35.
15. Namugenyi,MS. and Abdalla,F.(2022).An Automated Student Registration System: A Case Study of Lugazi Primary School. INOSR Experimental Sciences 10 (1), 25-41.
16. Kalulu, M. (2022).The Smart Grid and Its Security Challenges.INOSR APPLIED SCIENCES 9 (1), 9-12.
17. Eze, VHU. Ugwu, CN. and Ugwuanyi,IC.(2023).A Study of Cyber Security Threats, Challenges in Different Fields and its Prospective Solutions: A Review. INOSR Scientific Research 9 (1), 13-24.
18. Ogbonna,CC. Eze, VHU. Ezichi,SI. Okafor,OW. Onyeke, CA. and Oparaku,UO.(2023).A Comprehensive Review of Artificial Neural Network Techniques Used for Smart Meter-Embedded forecasting System.IDOSR JOURNAL OF APPLIED SCIENCES 8 (1), 13-24.
19. Okafor,OW. Edeagu,OS. Chijindu,CV. Iloanusi,NO. Eze,VHU.(2023).A Comprehensive Review on Smart Grid Ecosystem.IDOSR JOURNAL OF APPLIED SCIENCES 8 (1), 25-63.

Ogwal Robert Rubble (2023). Online Student Result and Archive Management System in KIU-College of Applied Sciences and Technology. *Eurasian Experiment Journal of Engineering*, 4(1):51-76