DESIGN AND IMPLIMENTATION OF WEB – BASED ACADEMIC INFORMATION SYSTEM (WEBAIS) FOR NIGERIAN UNIVERSITIES

By

MUKTAR MUHAMMAD ALIYU

DEPARTMENT OF MATHEMATICS

AHMADU BELLO UNIVERSITY, ZARIA

NIGERIA

APRIL 2015

DESIGN AND IMPLEMENTATION OF WEB – BASED ACADEMIC INFORMATION SYSTEM FOR NIGERIAN UNIVERSITIES

By

Muktar Muhammad ALIYU, Degree CS (WUDIL) 2007
MSc/Scien/07688/2009-10

A THESIS SUBMITTED TO THE SCHOOL OF POSTGRADUATE STUDIES, AHMADU BELLO UNIVERSITY, ZARIA

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD
OF A
MASTERS DEGREE IN COMPUTER SCIENCE.

DEPARTMENTOF MATHEMATICS,
FACULTYOF SCIENCE
AHMADU BELLO UNIVERSITY,
ZARIA

NIGERIA

APRIL, 2015

DECLARATION

I hereby declare that the work in this thesis titled "Web – Based Academic Information System for Nigerian Universities" was performed by me in the Department of Mathematics under the supervision of Prof. S. B. Junaidu and Dr. S.E. Abdullahi. The information derived from the literature has been duly acknowledged in the text and list of references provided. No part of this work has been presented for another degree or diploma at any university.

MUHAMMAD, MUKTAR ALIYU	_____	_____
Name of student	Signature	Date

CERTIFICATION

This thesis entitled "Web – Based Academic Information System for Nigerian Universities" meets the regulation governing the award of the degree of M.Sc. Computer Science of the Ahmadu Bello University Zaria, and is approved for its contribution to knowledge and literary presentation.

Professor S.B. Junaidu Date

Major Supervisor

Dr. S. E. Abdullahi Date

Minor Supervisor

Dr. B. Sani Date

H.O.D Mathematics

Professor H. Zoaka Date

Dean,School of Postgraduate Studi

DEDICATION

To my parents and my family.

ACKNOWLEDGMENTS

All praises are due to ALLAH (S.W.A) who grant me all the possibilities, opportunities, health and ability to conduct this research, I pray for more of his support throughout my life, Ameen. I thank my supervisor, Prof. Sahalu Junaidu, for his guidance and support throughout this research. He was gracious to provide me with the opportunity to conduct this research. His passion for research inspired me to pursue this program. I benefited tremendously when I was conducting this research. We achieved big success in WeBAIS and I finally discovered that I have more passion for system development than research. I appreciate his support for me to become who I am and in what I am doing now.

I also thank Dr. Abubakar Musa and Dr. Abdulhadi Aminu, two great Doctors I had the privilege to learn from, for the comments and feedback that enriched this thesis.

I would like to thank my parents who supported me with all their hearts. Finally, I thank my wife for being my cheer leader and always being there for me.

ABSTRACT

A common challenge faced by universities is effective management of large amounts of data that are encountered in day-to-day operations. Access to these data requires various privileges and restrictions to cater for the various user categories. Traditional paper-based approach to managing these data, as practiced in many Nigerian universities, is costly, error-prone and ineffective. Existing solutions to this problem, where they exist, are ad hoc or not comprehensive. In this thesis, we developed a web-based information (WeBAIS) capable of managing large bodies of data effectively for Nigerian universities. Evaluation feedback of WeBAIS received from the two pilot universities where it was deployed showed that WeBAIS met its design objectives. We believe WeBAIS will receive further favorable assessment as further research is conducted to polish its other modules.

TABLE OF CONTENTS

1 DECLARATION iii

CERTIFICATION iv

DEDICATION v

ACKNOWLEDGMENTS vi

ABSTRACT vii

TABLE OF CONTENTS viii

LIST OF FIGURES xiii

CHAPTER ONE: GENERAL INTRODUCTION 1

1.1 Introduction 1

1.2 Research Motivation and Goals 2

1.3 Research Question 2

1.4 Research Objectives 3

1.5 Methodology 3

1.6 Contributions to Knowledge 4

1.7 Outline of Thesis 5

CHAPTER TWO: LITERATURE REVIEW 6

2.1 Introduction 6

2.2 Related Work 6

2.3 Student Evaluation of Teaching (SET) 10

2.4 Modern IS Design 12

2.4.1 Information System 12

2.4.2 Computer-Supported Cooperative Work 13

2.4.3 Usability 14

2.4.4 Database Design 15

2.5 Summary 17

CHAPTER THREE: DESIGN OF WeBAIS 18

3.1 Introduction 18

3.2 Students: WeBAIS-graduate and WeBAIS-undergraduate 20

3.3 WeBAIS-graduate Specific Functionality 21

3.3.1 Graduate Admissions 22

3.4 WeBAIS-undergraduate Specific Functionality 23

3.4.1 Student Admission Module 23

3.4.2 Student Registration Module 23

3.4.3 Student Result Checking Module 24

3.4.4 Student Evaluation of Teaching (Questionnaire) (SET) 24

3.5 Faculty (WeBAIS-faculty) 24

3.6 Department Programs and Processes: WeBAIS-department 25

3.7 Staff: WeBAIS – Staff 25

3.7.1 Staff Module 25

3.7.2 Level Coordinator (Course Advisor) Module 26

3.7.3 Departmental Examination Officer Module 26

3.7.4 Head of Department Module 26

3.8 Summary 27

CHAPTER FOUR: IMPLEMENTATION OF WeBAIS 28

4.1 Introduction 28

4.2 System Design 29

4.3 System Architecture 29

4.4 Application Architecture 31

4.5 System Design (Data Flow Diagram – (DFD)) 36

4.5.1 Difference with Flowcharts 37

4.6 Web Interface Design 38

4.7 Database Design 39

4.7.1 User Role Database Structure 40

4.7.2 Content Management DB Structure 41

4.7.3 SET – Database Structure 42

4.8 Code Design 43
4.8.1 Web Utility Classes 43
4.8.2 Authentication Related Classes 45
4.8.3 Data Object Related Classes 46
4.8.4 Tool Related Classes 46
4.8.5 Summary 47
4.9 System Security 47
4.9.1 Authentication 48
4.9.2 Authorization 50
4.9.3 Encryption and Decryption 51
4.10 Summary 52
CHAPTER FIVE: DEPLOYMENT and APPRAISAL 53
5.1 Introduction 53
5.2 WeBAIS Implementation Status 53
5.2.1 WeBAIS Deployment Requirements 56
5.3 Deployment 57
5.3.1 Questionnaire design 57
5.3.2 Results Presentation and Analysis 58
5.4 Summary 64

CHAPTER SIX: SUMMARY, CONCLUSION AND RECOMMENDATION 65

6.1 Summary 65

6.2 Conclusion 65

6.3 Recommendation.....67

References.....68

Appendix A: Data Flow Diagram of WeBAIS.....72

Appendix B: WeBAIS E – R Diagram and Tables.....94

Appendix C: WeBAIS Questionnaire.....96

LIST OF FIGURES

Figure 2.1: Student Information System Architecture 10

Figure 3.1: Functional Diagram of WeBAIS 18

Figure 4.1: System Architecture 30

Figure 4.2: Application Architecture (Simple Application Logic). 32

Figure 4.3: Application Architecture (User Group and Views). 33

Figure 4.4 : Information System Architecture (functional decomposition of the system). ..34

Figure 4.5: Student Evaluation of Teaching (SET) Module Diagram 35

Figure 4.6: Data Flow Diagram Symbols 36

Figure 4.7: User Role DB Structure 40

Figure 4.8: Content Management DB Structure 41

Figure 4.9: SET – Database Structure 42

Figure 4.10: Function diagram for web utility classes 45

Figure 5.1: Efficiency and Service Delivery 61

Figure 5.2: Student Evaluation of Teaching (SET) 62

Figure 5.3: Robustness and security of WeBAIS 63

Figure 5.4: WeBAIS' user-friendliness and ease of use 63

Figure A.1: Data Flow Diagram of the Application 72

Figure B.1: Database Tables and Diagram 94

CHAPTER ONE: GENERAL INTRODUCTION

This chapter discusses the introductory part of the thesis which includes background of the study, research motivations and goals, the research questions the thesis attempts to answer, the methodology used to answer those questions and finally the summary of the thesis contribution to knowledge.

1.1 Introduction

A major challenge faced by university academics is the effective management of large amounts and various types of data that are encountered in day-to-day operation, ranging from personal data to varied types of documents. The data have various access privileges and restrictions and will be input by a variety of sources, ranging from individuals themselves (e.g., a current student or applicant submits a resume), to a member of the faculty to a member of the departmental administration (e.g., Head of Department who manages the affairs of the Department), to automatic uploading of transcripts or test scores for an applicant. Automatically generated statistics regarding this information will be used by many different stakeholders, both inside and outside the department. Traditional paper-based operation is not only costly but also ineffective.

We designed and implemented a Web-Based Academic Information System(WeBAIS), to effectively manage school information. The design of the overall system follows the guidelines from the Design Science for Information System (Hevner A., *et* al, 2004)and benefits from the study in Computer-Supported Co-operative Work (CSCW) (Grundin, Computer Supported Cooperative Work History and Focus, 2005). To increase the usability

of our system, we followed Computer-Human Interaction (The Apache Software, 2007) and Iterative Design guidelines (Denning, 1997).

1.2 Research Motivation and Goals

The greatest challenge of today's information system design is to provide users with the required functionality needed for the organization which, in many cases, happens not to be possible or requires huge resources including money, time and highly skilled personnel. Information systems of many Nigerian Universities suffer from some of the following challenges:

a. Ineffective management of the large amounts and different types of data
b. Inability to cater for different user categories with various access privileges and restrictions
c. Lack of related components that need to be integrated together such as registration and academics.
d. Lack of robust, portable and extensible components that support efficient data management.

The objective of the research in this thesis is to develop a robust information system with a single integrated data store that overcomes these limitations and more.

1.3 Research Question

The study is set up to answer the following question:

Can an information system be developed for Nigerian Universities that is robust, portable, extensible and user-friendly?

1.4 Research Objectives

The objectives of the research undertaken in this thesis is to design and implement a web-based academic information system (WeBAIS) that

a. is robust, portable and user-friendly
b. makes it easier for stakeholders (managers, staff and students) to carry out their normal activities
c. improves efficiency of delivering university services
d. includes a student evaluation of teaching (SET) module for improving the learning process

1.5 Methodology

The following are the proposed steps needed for the realization of this research work.

a. Literature review to determine best practice in modern information system design
b. Develop the architecture of the proposed WeBAIS system
c. Implement the WeBAIS system
d. Deploy WeBAIS in pilot universities
e. Collect feedback from WeBAIS users
f. Analyze user-feedback against WeBAIS objectives

1.6 Contributions to Knowledge

The main contribution of this work is the design and implementation of a university information system that significantly improves efficiency of a university's daily operations. The application has been tested over a list of more than ten thousand (10,000) undergraduate students from various departments and faculties for three academic sessions. It also captures the list of all members of staff both academic and non-academic staff of Kano University of Science and Technology, Wudil, and North-West University, Kano.

WeBAIS significantly improves the efficiency for many daily university operations. For example, the traditional paper-based graduate program application involves a sequence of procedures: paper submission of all materials to be sorted and collated into paper – based folders, which are passed sequentially from admissions chair to faculty members to review. The average processing time is several months. Identifying high priority applicants was difficult and often results in losing good ones. With the help of WeBAIS, we are able to electronically submit almost all application materials from applicants, references, admission related documents and perform automatic importing of test scores, undergraduate institution, etc. As a result, an application can be reviewed by all faculty members immediately and decisions can be made in days, making it easy to identify excellent applicants.

We can now do many things with WeBAIS that we could not do before. First, WeBAIS makes information sharing possible and convenient, e.g., in undergraduate/graduate admission, HODs' of the faculty members are able to view any application and recommend; Admission Committee Chairman/Member is able to view and grant admission

to applicants. Second, generating statistics is very easy in WeBAIS while it required much bookkeeping and manual processing before.

In WeBAIS, we designed and implemented a set of web utility classes. These classes are general, self-contained, and greatly increase the reusability and maintainability of our code. By building on top of the web utility classes, the development of WeBAIS application becomes relatively easy. Furthermore, the web utility classes can be used for other web applications. We plan to make them open source software in the future.

Although WeBAIS is an information management system for the university (KUST – Wudil as case study), it can easily be configured to be used by Colleges, Monotechnics, Polytechnics and the like. WeBAIS can be extended to support other services apart from academic system such as administrative system of a university. So far, WeBAIS has generated a lot of interest in KUST and North-West Universities, encouraging further research of the application.

1.7 Outline of Thesis

Chapter II describes related work from different fields that are used to inform the design of WebAIS. Chapter III presents the definition of requirements and functionality of our system, WeBAIS. Chapter IV presents the design of the system. Chapter V describes the deployment, user appraisal and results implementation status for our system. We conclude in Chapter VI.

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

The main objective of WeBAIS is to improve the effectiveness and efficiency of the university academic work through a paperless interface to facilitate information transition, processing, and storage. In this chapter, we concentrate on reviewing some related works concerned with student information systems, client/server systems, web-based systems, distributed systems and result processing systems. Ideas were drawn from these diverse areas that inform the design and implementation of our WeBAIS system.

2.2 Related Work

Academic institution is an educational institution dedicated to education and research, which grants academic degrees (Obasi, *et* al 2013). Teaching and research remain the primary activities in higher institutions, but there are other important activities as well, like managing of students' data at the different levels of higher education (undergraduate, postgraduate, and doctorate levels); tracking of students' progress at each level; as well as other administrative and managerial activities. The creation and management of accurate, up-to-date information regarding students' academic career is critically important in the university as well as colleges. Student information system deals with all kind of student details, academic related reports, college details, course details, curriculum, batch details, placement details and other resource related details too (Bharamagoudar, *et* al 2013).

With emergence of information technology, “the higher education administrational processes are undergoing a significant transition. The first major shift in these technologies occurred when they evolved from mainframe programs to client server solutions” (Kumar, 2011). Today, these technologies are more central than ever to colleges and universities, the changing technology landscape together with data reporting demands has compelled colleges and universities to evaluate their major software systems. Now universities are mostly using centralized web based systems that is based on industry standard technology. These latest generation systems allow sharing of data easily with other systems and provide option to communicate important information to relevant stakeholders. Institutions are under constant pressure to demonstrate both willingness and capacity to incorporate the latest developments in CIS. With younger generation having grown up with technology, students now expect to get information and do business with colleges and universities on the web. Ease and convenience for the customer is essential for any university and we believe the student services on the web play a major part in this.

Custom developed software with proper analysis, design and implementation, the final product should meet the requirements defined by the user. User acceptance should be higher because of the input end users have in the design. Custom development often allows the organization to avoid ‘big bang’ implementation of a system through phased implementation (Kumar, 2011). Some of the limitations could be that design and development phases result in longer and larger projects.The key architecture choice is between client server system and web-based system (Bharamagoudar, *et* al 2013).

The client/server products require a server for each site and use special software that must be installed and maintained on each computer running the CIS (Bharamagoudar, *et* al

2013). In contrast web-based systems usually have centralized server that allows users to access data via a secured internet connection without special software installations (Kumar, 2011). The World Wide Web has emerged as powerful and appealing technology to utilize in the migration of main frame systems. Ease of development cost, platform independence and accessibility are some of the reasons that using the web is appealing (Hevner, *et* al 2004). Continuing exponential growth in the availability of computing processor cycles, memory, storage and network bandwidth together with the rapid growth of World Wide Web have made it possible to develop modular and flexible systems (Kumar, 2011).

We review some recent related works aimed at improving the earlier systems in the following paragraphs.

Obasi, *et* al 2013, developed a web-based student academic records information system which implements a wide range of students' functionality. Limitations of this work include: only students use the system; no roles for administrators and staff, no candidate application module, no admission processing, no SET module on the system and it does not use any of the IS moled.

Youh 2010, presented a distributed system for student result processing. Their system emphasizes advantages of distributed system and allows each department to maintain its database. Limitations of this system include lack of roles for students and staff, lack of application, admission, registration and SET modules. Their system is also difficult to manage, maintain and upgrade.

Obiniyi and Absalomin 2010, addressed the problems arising from result processing on the intranet, which focuses on speeding up collection of students' academic data to expedite

processing of results and transcripts at various levels. It also allows online access of results for students. Limitation of this system include only two services (result processing and library resources management) were supported, no candidate application module, no course management, staff records, course allocation and roles for administrators and staff have not been implemented.

From the much documented in the Staff User manual, this system does not provide the kind of management information that would be expected for management activities.

The approach adopted for result update is a major concern. These concerns include:

a. Possibility of the inbuilt formula in Microsoft Excel file used to collate results broken.
b. No much difference from the earlier approach used by the University, except for the fact that the results collated using Microsoft Excel is eventually stored in a database.
c. Amount of time taken to manually search through the Excel file to update a score.
d. Unnecessary duplication of data as a result of versioning. This leads to wastage of storage.

From the above literatures, in summary, we observed and derived the simple architecture for the related information system on the Figure 2.1 below:

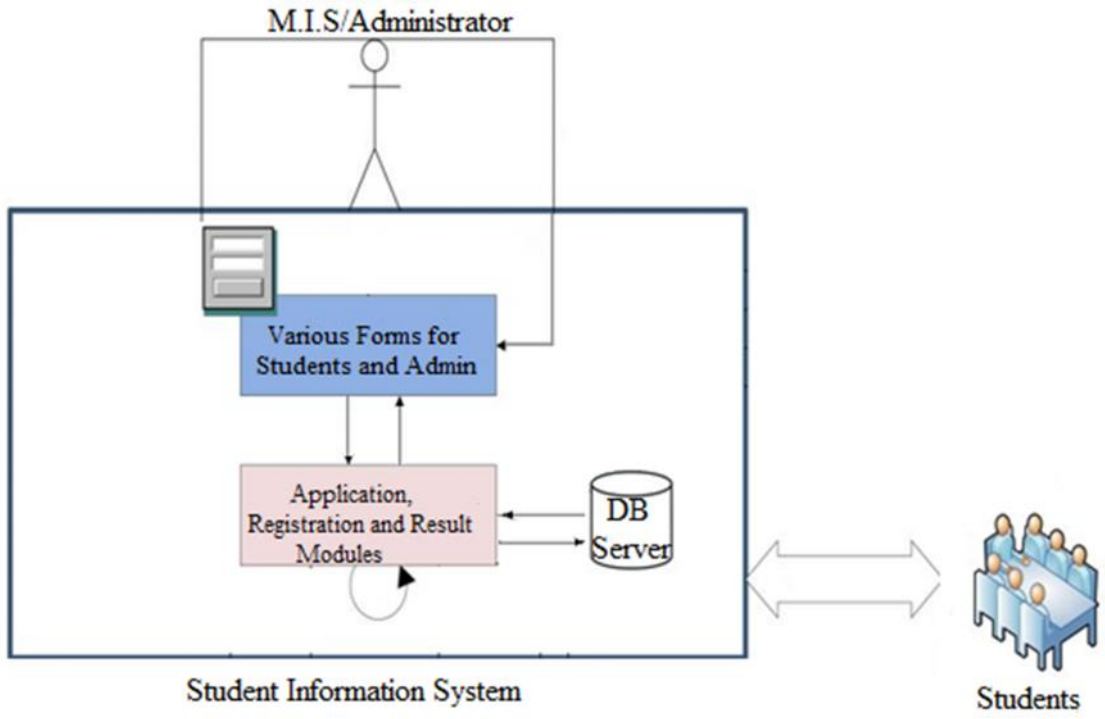


Figure 2.1: Student Information System Architecture

2.3 Student Evaluation of Teaching (SET)

SET is a paper or electronic questionnaire, which requires a written or selected response answer to a series of questions in order to evaluate the instruction of a given course.

Over the past century, student ratings have continued to take precedence in teaching evaluation systems in many part of the world. However, this is not the case with many Nigerian Universities (Chen & Hoshower, 2003). A key problem in this research is to provide a systematic way of making sure that each student participates in evaluating course – teacher before his/her academic results are released. This will enable faculty/management as well as teacher to improve student learning and teaching motivation. The idea is to

integrate both the academic and course – teacher evaluation together, so as to have both the academic records as well as the course – teacher evaluation records to make student's participation possible.

Since student ratings are the most, if not the only, influential measure of teaching effectiveness, active participation by and meaningful input from students can be critical in the success of such teaching evaluation systems. Nevertheless, very few studies have looked into students' perception of the teaching evaluation system and their motivation to participate. This application employs methodology to make student participation in evaluating their teachers as easy as possible, so as to motivate students to participate in the teaching evaluation process. Students generally consider an improvement in teaching to be the most attractive outcome of a teaching evaluation system (Chen & Hoshower, 2003). The second most attractive outcome was using teaching evaluations to improve course content and format. Using teaching evaluations for teacher tenure, promotion and salary rise decisions and making the results of evaluations available for staff, Departmental Head, and other stakeholders of the school decisions on course and instructor selection were much important. Some of the key points of these applications include:

a. Teaching/Course evaluation improves the learning process
b. Student ratings have been used in many part of the world
c. Many Nigerian Universities do not use SET
d. The National Universities Commission (NUC) has recommended making SET mandatory to Universities
e. Rating motivates instructors to improve teaching

f. It motivates students to participate in the teaching evaluation process.

2.4 Modern IS Design

A growing amount of interdisciplinary work explores the number of goal: answering how the activities of complex systems can be coordinated (Malone & Crowston, 1994). Some work focused on coordination in parallel and distributed computer systems while others on coordination in human systems (Malone & Crowston, 1994). However, the common theme is that the targeted complex systems usually include both people and computers; our project shares the same characteristic.

In modern IS design, some inter disciplinary areas enlighten on how to design IS that is robust, reliable and cost – effective. Below are some of these inter-disciplinary fields and recent findings about them which will help us in laying down the theory background of our work.

In most of the papers reviewed above they described in detail how they designed and implemented there IS, none of them discovered and used the latest discovery of IS by either by selecting Design Science or Behavioral Science (March & Smith, 1995).

2.4.1 Information System

According to Hevner, *et* al 2004, Information Systems (IS) are "implemented within an organization for the purpose of improving the effectiveness and efficiency of that organization". Two paradigms that characterize much of the research in the IS discipline are behavioral science and design science. The behavioral science paradigm seeks to

develop and verify theories that explain or predict human or organizational behavior. The design science paradigm seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts.

These two paradigms are complementary but distinct (March & Smith, 1995). The behavioral science paradigm has its root in natural science research methods. It seeks to develop and justify theories that explain or predict organizational and human phenomena (Hevner, *et* al 2004). The design science paradigm has its roots in engineering and the science of the artificial (Simon, 1996). It is fundamentally a problem solving paradigm and seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished (Denning, 1997).

Our work obviously falls in the realm of the design science due to the problem solving nature of the work. As the IS literature recognizes, while the importance of design is well recognized, designing a useful system is complex. We built on the work of the design science paradigm and followed the literature suggested guidelines in (Hevner, *et* al 2004).

2.4.2 Computer-Supported Cooperative Work

Recently, a great deal of interest has been placed on designing computer tools to help people work together more effectively. This new field designs systems called Computer-Supported Cooperative Work (CSCW) (Grundin, Computer Supported Cooperative Work History and Focus, 2005) or groupware (Fersko-Weiss, 1992). According to Ellis et al. in (Ellis, Gibbs, & Rein, 1991), “the work in CSCW usually looks at how groups work and seeks to discover how technology (especially computers) can assist them.” Groupware is

the technology used to support collaborative work (Fersko-Weiss, 1992). According to Ellis *et* al 2001, groupware consists of "computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment." CSCW stresses the outcome or product while groupware is the enabling technology.

In our work, we studied closely how different components of the department administration functions and designed a system that helps the complex coordination. Therefore, the work in CSCW provided a lot of insights for our work.

2.4.3 Usability

Usability is one of the focuses of the field of Computer-Human Interaction (Baecker, *et* al 2005). As the name suggests, usability has to do with bridging the gap between people and machines. For a tool to be effective, it must allow intended users to accomplish their tasks in the best way possible. Usability depends on a number of factors including how well the functionality fits users' needs, how well the flow through the application fits users' tasks, and how well the response of the application fits users' expectations. According to Nielsen 2004, usability has multiple components and is traditionally associated with five attributes. First, **Learnability**: the system should be easy to learn so that the user can rapidly start getting some work done with the system. Second, **Efficiency**: the system should be efficient to use, so that once the user has learned the system, a high level of productivity is possible. Third, **Memorability**: the system should be easy to remember. Fourth, **Errors**: the system should have a low error rate, so that users make few errors during the use of the system. Fifth, **Satisfaction**: the system should be pleasant to use (Nielsen, 1994).

The key principle for maximizing usability is to employ an iterative design, which

progressively refines the design through evaluation from the early stages of design (Gould, 2000). The evaluation steps enable the designers and developers to incorporate users' and clients' feedback until the system reaches an acceptable level of usability.

The preferred method for ensuring usability is to test actual users on a working system. Achieving a high level of usability requires focusing design efforts on the intended end-user of the system.

Our work provides various types of web interface for users and the usability of our web interface results in the effectiveness of our system. We included usability early in the design/development process and included users in the design process to improve the usability attributes described in (Nielsen, 1994). We also allowed usability and users' needs to drive our design and test our system frequently throughout the design process.

2.4.4 Database Design

Our work involves massive data management which requires a sophisticated database design. Database design is defined as: "the logical and physical structure of one or more databases to accommodate the information needs of the users in an organization for a defined set of applications" (Date, 1990).

According to Date 1990, the design process follows several steps: 1. planning and analysis; 2. conceptual design; 3. choice of Database Management System (DBMS); 4. Logical design; 5. Physical design and 6. Implementation.

The following section briefly walks through each step (Date, 1990).

The goals of the requirements analysis (Step 1) are to determine the data requirements of the database in terms of primitive objects; to classify and describe the information about these objects; and to identify and classify the relationships among the objects (Date, 1990).

A data model is a conceptual representation of the data structures that are required by a database (Introduction to Data Modeling, 2007). The data model is one part of the conceptual design process (Step 2). The Entity-Relationship (ER) (Chen, 1976) model is widely used for the data modeling. A basic component of the ER model is the Entity-Relationship (ER) diagram which is used to visually represent data objects. In the ER model, "Entities" are the principal data objects about which information is to be collected (Chen, 1976). Entities are usually recognizable concepts, either concrete or abstract, such as people, places, things, or events which have relevance to the database. Some specific examples of entities are student, degree program, and document. A "Relationship" represents an association between two or more entities (Chen, 1976). An example of a relationship would be: students are enrolled in a degree program; a degree program involves different documents.

A Database Management System (DBMS) is computer software designed for the purpose of managing databases. Typical examples of DBMSs include Oracle, Microsoft SQL Server, and MySQL. Step 3 involves making choices of DBMS.

In logical design step (Step 4), the ER diagram is converted into tables in the database. Physical database design (Step 5) consists of executing tables in an actual DBMS software file. Database system implementation (Step 6) consists of entering data and maintaining the DBMS.

2.5 Summary

To build a user friendly, reliable, flexible, and effective paperless University information management system, we needed insights from different disciplines. This chapter presented the key concepts and review of the literature. None of the literature above meets our defined target of building a robust, effective and user friendly information system for Nigerian Universities.

In the next chapter we present system requirements, functional requirements and non-functional requirements of WeBAIS.

CHAPTER THREE: DESIGN OF WeBAIS

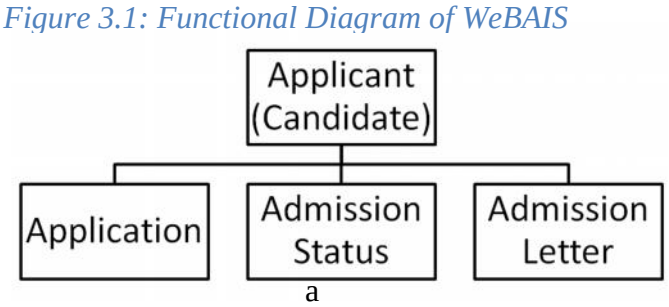
3.1 Introduction

Different components of WeBAIS share common functionality in different ways. Some components are very similar to each other. For example, WeBAIS-graduate and WeBAIS-undergraduate are parallel components that share significant functionality. Some common functionality is required in all components. For example, all components need functionality to upload and manage documents. Some similar functionality is used in all applications, but they differ from each other slightly. For example, all online applications involve some kind of review process, although they may not be exactly the same. In the following, we provide an overview of the requirements and functionality for each component.

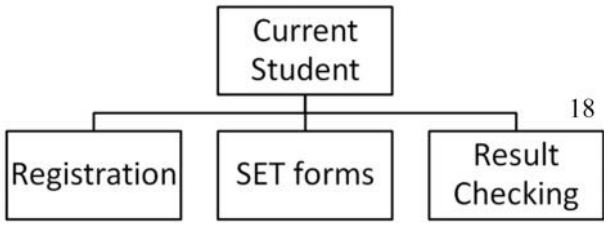
Figure 3.1 shows the functional diagram of the WeBAIS system whose main components we briefly describe next.

• WeBAIS-applicant – a component used to manage information and procedures related to students' applications and admission Status. dfdf

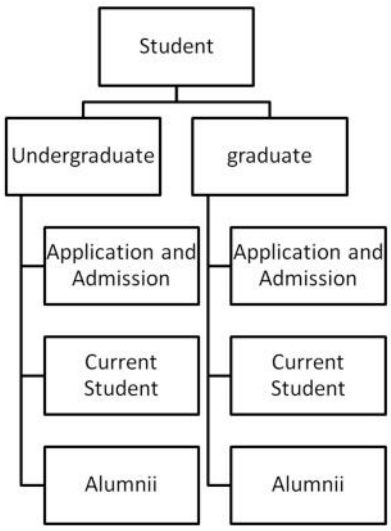
Figure 3.1: Functional Diagram of WeBAIS



a



18



a. WeBAIS-Student – a component used to manage information related to Students (both new and returning). c

b

In this work, we also provide a detailed case study of the Student Evaluation of Teaching (SET), which is a tool to monitor staff teaching performance response by the students. The SET interacts with all WeBAIS components: students who apply for admission (WeBAIS-applicant) will pay application fee as well as processing fee. Also, students who secure admission (WeBAIS-student) will continue with course registration from entry point to graduation level where all the information for the student will be available for various accesses according to their respective privileges.

a. WeBAIS-undergraduate – much of the planned functionality for this component is already in production use. It includes undergraduate admissions (on-line submission of application materials by applicants, on-line review by the department, and on-line decision by the admission committee); current undergraduate student information management (degree progress monitor, dismissal/withdrawal process, application, assignment, enrollment history); and former undergraduate student information management (listing of degree, graduate year).
b. WeBAIS-graduate – much of the planned functionality for this component is in progress. It includes graduate admissions (on-line submission of application materials by applicants and references, on-line review by the department/faculty, and on-line decision by the admission committee); current graduate student information management (degree

progress monitor, application, review and assignment, enrollment history); and former graduate student information management (listing of degree, graduate year).

c. WeBAIS-faculty – on-line submission of application materials by applicants and references, online review by the faculty, some support for the interviewing process.

d. WeBAIS-department – Research Experience for Undergraduate (REU) program (on-line submission of application materials by applicants and references and review by the faculty) and academic programs administration (semester course planning and scheduling support, instructor assignments and attendance, etc.)

3.2 Students: WeBAIS-graduate and WeBAIS-undergraduate

WeBAIS-graduate and WeBAIS-undergraduate are parallel components that share a large amount of functionality. In this section, we first introduce the common functionality for both components and then discuss the component-specific functionality.

Academic program information is essential for an academic department. Providing tools for students, faculty, and staff to easily access/manage the various types of academic information is important. WeBAIS-graduate and WeBAIS-undergraduate provide tracking for academic information throughout students' stay in the program. For example, students' basic information, tests scores and various types of degree program related information. Data are mainly and maintained by the level coordinators, but students are allowed to input and/or edit certain data. Most data are available (accessible) to student and all data are available to all faculty members. In this application, we plan to support the following functionality for current students:

a. Student basic information: Name, contact information, gender, citizenship, ethnicity, Nigerian residency, photo, etc.
b. Degree progress monitor: Provides information about milestones (degree plan, prelim exam, final exam, etc.), enrollment, academic performance (GPA and CGPA).
c. Registrations and Payments Records, history, selection, and evaluation (RPR)
d. Email support: Listserv management and other mailer tools
e. Statistics and report generation (admission, graduation, enrollment).

Maintaining contact information and tracking the accomplishments of former students are important aspects of an academic department. We plan to support the following functionality for alumni:

a. List of alumni to be used to generate list on public web pages and statistics accessible on internal web pages.
b. Tracking of alumni (where they went and how they are doing) to build a network of former students to assist current students and for university purposes.
c. Alumni events, newsletters, and other services to be provided for the university's alumni.

3.3 WeBAIS-graduate Specific Functionality

WeBAIS-graduate supports one major component that are not needed for WeBAIS-undergraduate: graduate admissions.

3.3.1 Graduate Admissions

A Graduate Admission is a very important factor that affects the quality of a graduate program. Thus, having an effective way to process graduate admissions is crucial. Traditionally, applicants and their references send all application related documents in hard copy to the department. It takes a significant amount of time and resources to process these documents. From the department's point of view, various types of documents need to be processed manually and then sent to faculty members for review. Since it involves a sequence of steps, the entire procedure is very time consuming and inefficient. To address this problem, WeBAIS-graduate provides an online system for graduate admissions. The system allows applicants and their referees to be able to use electronic submission for all application related materials; faculty members in the department to view all applications and the supporting materials and provide reviews online; the admissions committee to make decisions that can be made available to both applicants and all faculty members (at different times if desired). In this application, we support the following functionality:

a. Online submission of application materials by applicant and references,
b. Online review by faculty members,
c. Online decisions by the Admission Committee,
d. Automatic data import from internal and or external sources such as the Office of Admissions, and
e. Support for advising e.g., to send email to applicants regarding their decision and to manage other application related information.
f. Online submission for corresponding parties (student's referees able to submit

recommendation letter), and

g. Online review system for admission committee.

3.4 WeBAIS-undergraduate Specific Functionality

In addition to the common functionality, WeBAIS-undergraduate supports or will support the following functionality:

3.4.1 Student Admission Module

a. Application Form

b. Applicants List for Processing

c. Post – UTME Result Imports and Processing for Admission Evaluation

d. Application Recommender (by the member of the Department (HOD))

e. Application Processing (Granting/Revoking by the Admission Committee)

f. Application Status Tracking by the Applicant

g. Admission Letter Report to successful applicants

3.4.2 Student Registration Module

a. Bank Payment Registration by both returning and fresh students

b. Automatic Registration Number Generation to New students

c. Automatic Carry Over Tracking and Registration to returning Students

d. Time based Registration exercise monitor and control

e. Access to previous registrations and payments

3.4.3 Student Result Checking Module

a. Access to Semester/Session Result for all Examinations

b. Access to Transcript (Graduate/Alumni – Protected Copy)

3.4.4 Student Evaluation of Teaching (Questionnaire) (SET)

a. Evaluate tutor of any registered course for teaching performance

3.5 Faculty (WeBAIS-faculty)

WeBAIS-faculty supports procedures and processes related primarily to faculty, such as faculty search, faculty progress report submission and archiving, staff list Reports, statistics generation and data archiving, and support for the Appointment and Promotion and Committee (A and PC).

The faculty search application provides an online system for student/staff information, department search and programs. Much of the functionality in this application is similar to those in the graduate admissions application. Additionally, the faculty search application will provide for SET statistic report for all faculty staff and for individual.

WeBAIS-faculty will also support a number of departmental procedures where faculty needs to support materials for evaluation by departmental committees. For example, the submission of annual progress reports by all faculties (to be evaluated by the review committee, consisting of the department head and members of A and PC, and promotion and tenure materials (to be evaluated by the A and PC committee). In both cases, at least

initially, WeBAIS-faculty will only handle material submission by faculty – the reviewing and evaluation will be handled outside WeBAIS.

3.6 Department Programs and Processes: WeBAIS-department

WeBAIS-department supports departmental procedures and processes, such as course planning and scheduling, course allocation, result processing and transcript. It also includes staff, Academic staff, level coordinator, examination officer and Head of department's modules/roles.

Course planning and scheduling provides semester by semester course planning and scheduling support (course offerings, course instructor, etc.).

3.7 Staff: WeBAIS – Staff

This implements most of the associated functions that relate and restrict staff and their various roles as follows:

3.7.1 Staff Module

a. Access his/her profile and user account
b. Academic Staff Module
c. Access all courses taken on each semester/session
d. Access list of students register with course(s) taken
e. Access and keep Attendance Records of the students
f. Import Result file of each course taken (to the Server)

g. View Result import report

h. View Statistical Report on SET

3.7.2 Level Coordinator (Course Advisor) Module

a. View Students Advisees

b. View Students' Courses Registered

c. Make Add/Drop to Students (Monitored by Management)

d. View Students Detail Information

e. View Students Detail Results

3.7.3 Departmental Examination Officer Module

a. View Courses Allocation on each semester/session

b. View Result files Reports (uploaded by the Lecturer)

c. Upload Result to the Database after Departmental Approval

d. View Examination Detail/Summary Reports for Senate

e. Track student exams records and reports

3.7.4 Head of Department Module

a. View All Staff of the Department

b. View All Students of the Department

c. View All Courses taken by the Department

d. View and Allocate courses to the Academic Staff

e. Track Results Upload Reports by the Lecturers

f. View Statistical Report on (SET) for all Staff

g. View Examination Detail/Summary Reports for Senate

h. Track student detail/exams records and reports

There are many more staff roles from high ranking staff to principal officers.

3.8 Summary

In this chapter, detail of the requirements of the system have been discussed. These include functional requirements like admission, registration, course management and allocation, staff profiles, examination and records. Also, some of the important key roles of the system such as student role, staff, exam officer, head of department, etc.

CHAPTER FOUR: IMPLEMENTATION OF WeBAIS

4.1 Introduction

The main purpose of this research project is to improve the efficiency with which academic and students services are delivered in a university environment. To achieve this goal, the need to adopt a sequence of steps adopted by information system developers as the methodology. The following steps observed in developing the system:

a. Extensive Literature review to determine best practice in modern IS design (Chapter II)
b. Feasibility Study and Requirement Analysis (Chapter III)
c. Development of the architecture of the proposed WeBAIS
d. Design the application interface and Database backend with portability consideration
e. Implementation of the WeBAIS system
f. Deployment of the WeBAIS in pilot university
g. Collection of feedback from WeBAIS users
h. Analysis of user–feedback against WeBAIS objectives

In this chapter we present the system design, system architecture, application architecture, web interface design, database design, code design and system security design. System implementation and appraisal are presented in the next chapters.

4.2 System Design

Design is very important in interactive systems and needs to be evaluated at each stage of development. We followed the Iterative Design approach (Gould, 2000) which states that once a component has been partially or completely implemented, we can receive feedback from users. Based on that feedback we can find out problems in the design of the interface or problems in the functionality of the system, and re-evaluate alternatives. Our design has to take into account the various types of data and users in different applications.

We split our design into the system architecture, application architecture, data flow diagram (DFD), the web interface, the database, and the code. For DFD we applied DFD Standard using IS framework (Whitten & Bentley, 2008), for the design of the web interface, we applied User Centered Design (Petrelli, *et* al 1999). Working closely with the final users allowed us to know their specific needs. For the design of the database, we used the Entity-Relationship (ER) (Chen, 1976) model for the data modeling. The object-oriented programming paradigm was used to design and implement the code. This paradigm will ease maintainability. Also, object-oriented programming offers a natural way of defining functions and data in terms of a class hierarchy.

In this section we first provide an overview of the system architecture. We then describe our design for the web interface, database and code.

4.3 System Architecture

A big challenge for WeBAIS is that it needs to support a large number of users whose use

for the system varies greatly. This challenge is a key factor that affects our design of the system architecture and the authentication.

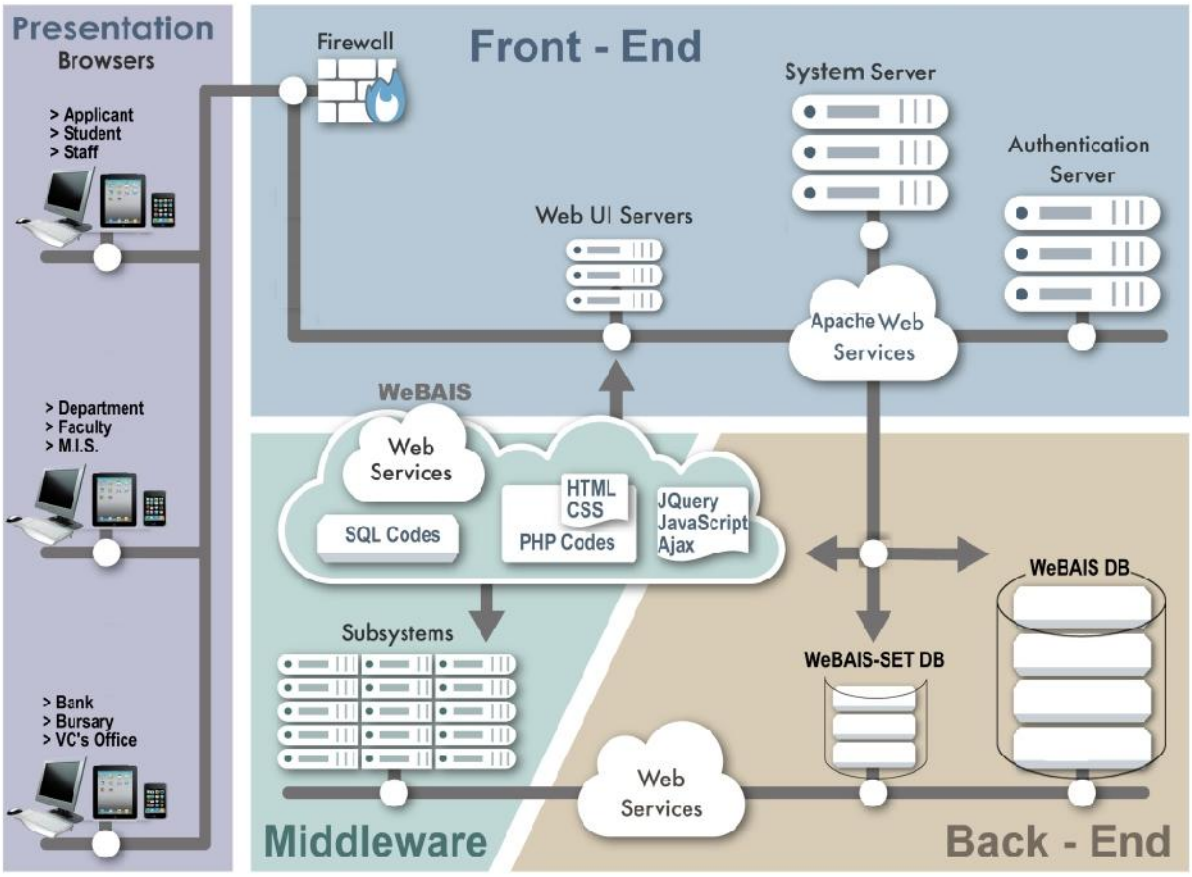


Figure 4.1: System Architecture

The system architecture of WeBAIS is shown in Figure 4.1 User requests over the Internet are received by a web server, and the corresponding query is sent to a database server. The data are sent back to the web server and the resulting web page is generated and sent back to the user. Our system runs in Windows with Apache (The Apache Software, 2007) and PHP for web serving and MySQL (SQL, n.d.) for database. All functionality is written in

PHP with MySQL API.

There are five different parts in Figure 4.1, namely i. presentation ii. Front – end iii. Middleware, iv. Back–end and v. the WeBAIS system

The presentation layer is where users interact with the system using different media like personal computer, table or even mobile phones. All they need is the browser which is at the front – end. In the front end, the browser is needed which is used to make a request through URL and translate the response from HTML to present it to the user in an organized way using both HTML, CSS and JavaScript.

The middleware and back – end encapsulate data services of the system, all the database responses pass through the middleware. Because this application is a data driven web application, there is high – level of abstraction which communicates with the database intelligently in these two layers. The Application Architecture, Dataflow Diagrams and Web Interface design are discussed in Sections 4.4, 4.5 and 4.6.

4.4 Application Architecture

Another challenge for WeBAIS is that it needs to cover a wide range of functionality for different user – groups, as such the Application architecture needs to be modeled in line with the software specifications.

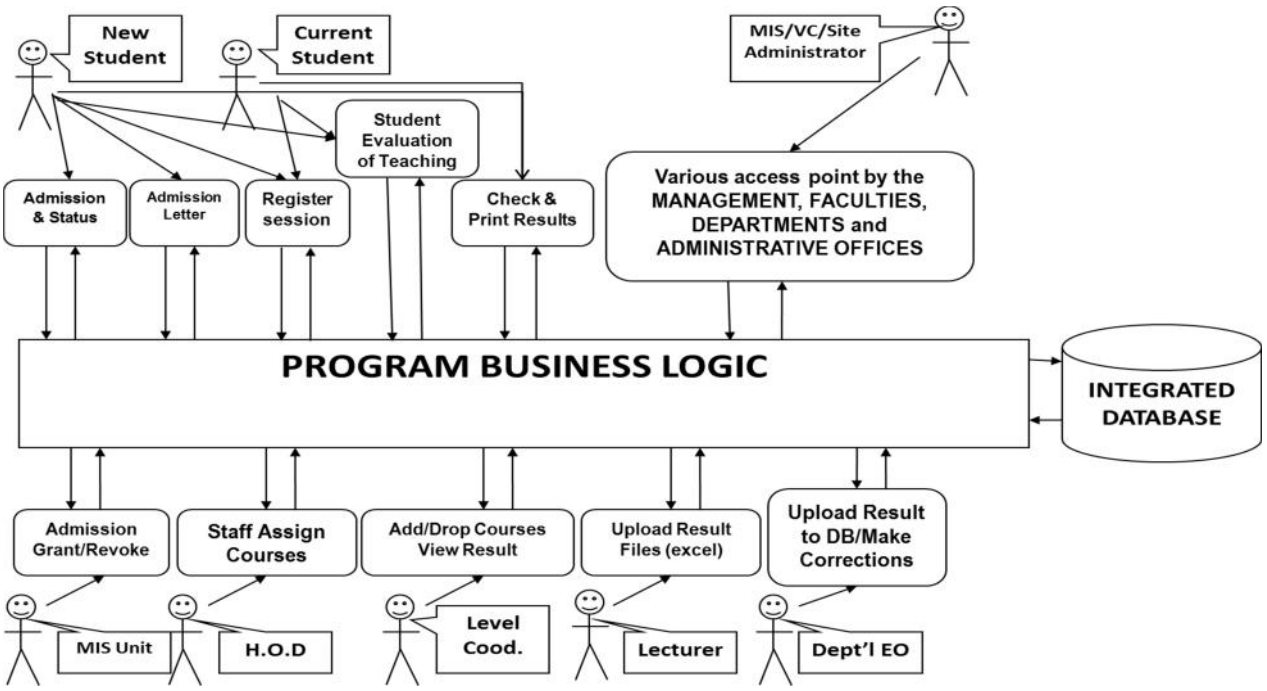


Figure 4.2: Application Architecture (Simple Application Logic).

Figure 4.2 shows the simple program flow for the main steps for different roles and how they interact with the system. The idea here is to create a role-based information system where each user has a specific role to play on the system. Most roles have been fully implemented with quantitative amounts of data for testing and assurance for the security and reliability of the system.

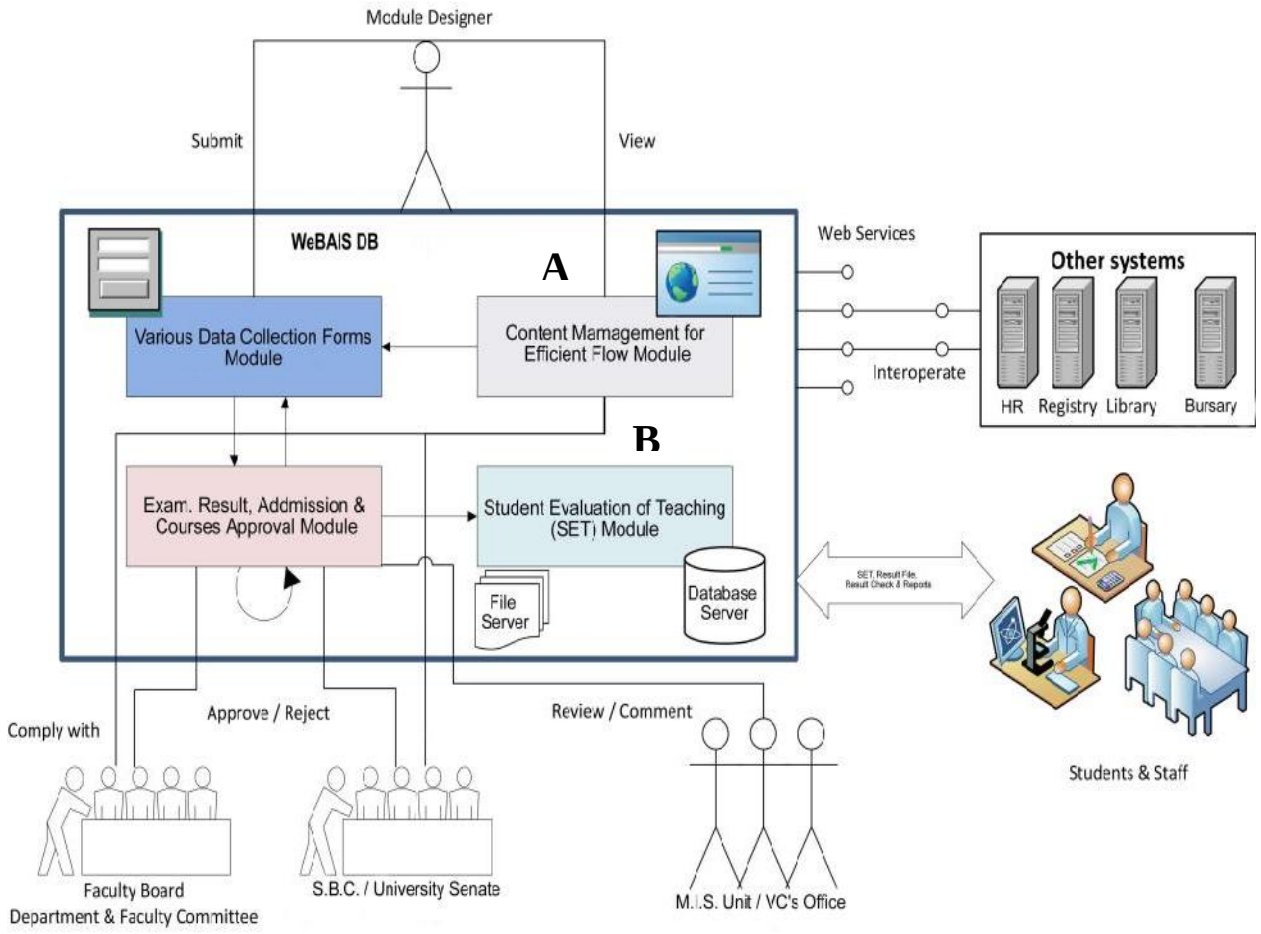


Figure 4.3: Application Architecture (User Group and Views).

Application with many users involved need careful designing, figure 4.3 shows how user – group are related and what are the new modules added from the previous work. Box labeled "A" is a content management module which is added from previous work to have an efficient management of the system. Content management defined how all the system components will be inter – linked and how will each content be identified uniquely so that only known contents goes to a specific role.

Box labeled "B" is a SET module, which is aiming at improving learning process generally,

no any university implement and integrate this module to its information system. SET is being design to communicate with student's courses registered as well as staff courses allocation together to identify who teach what course to what particular set of students. Student will view all the courses registered and access this module to assess each staff taken a course from the courses registered by student. Detail of this SET box will be shown on figure 4.5.

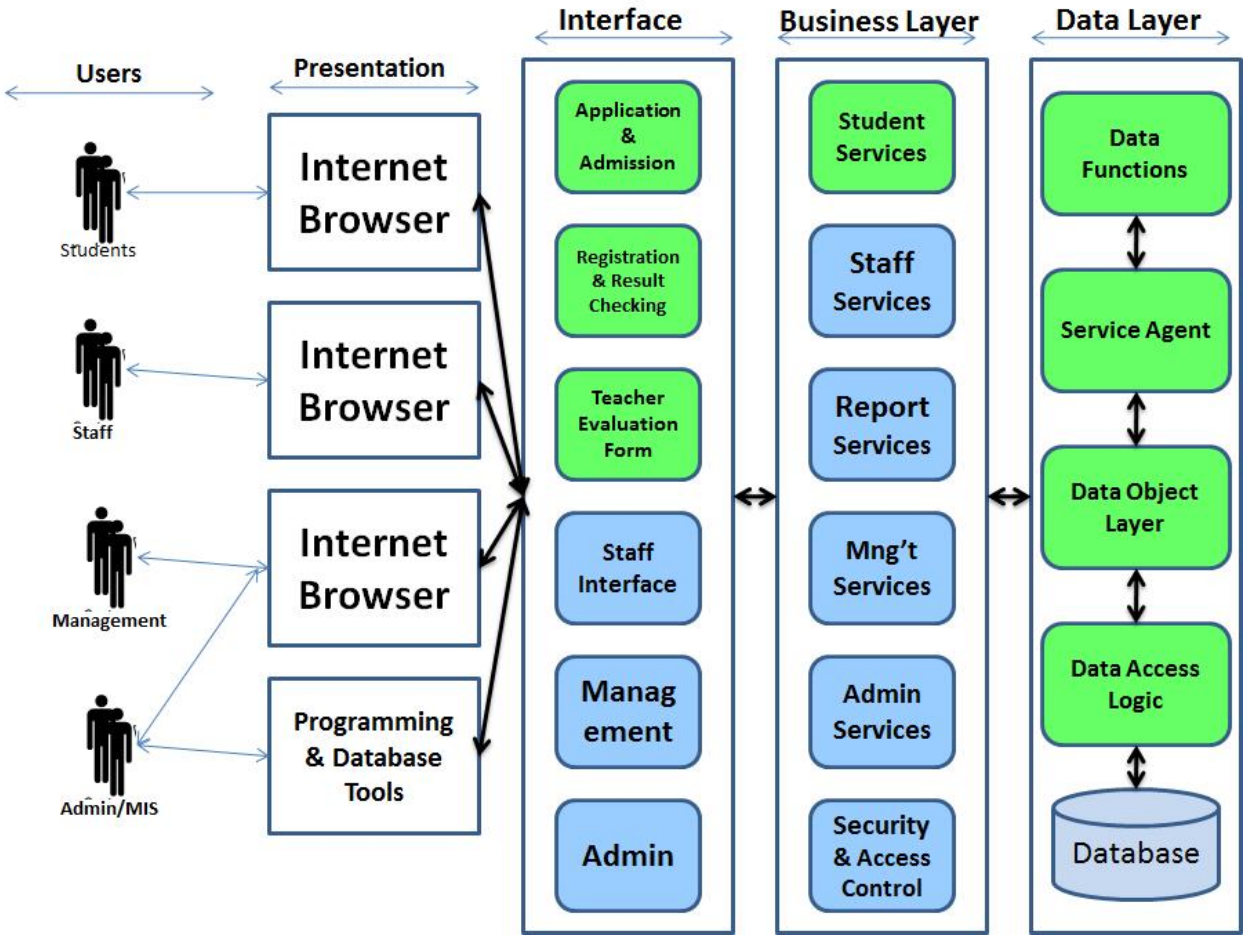


Figure 4.4 : Information System Architecture (functional decomposition of the system).



Figure 4.4 shows the detailed architectural diagram of the entire system. The system was built in stages. The green ovals in the figure shows components completed in phase I and the pink ovals shows complements for Phase II. All users of the system except system developer interact with the system via internet browsers of any kind via any device. The system developer can access the system using programming and database tools, too.

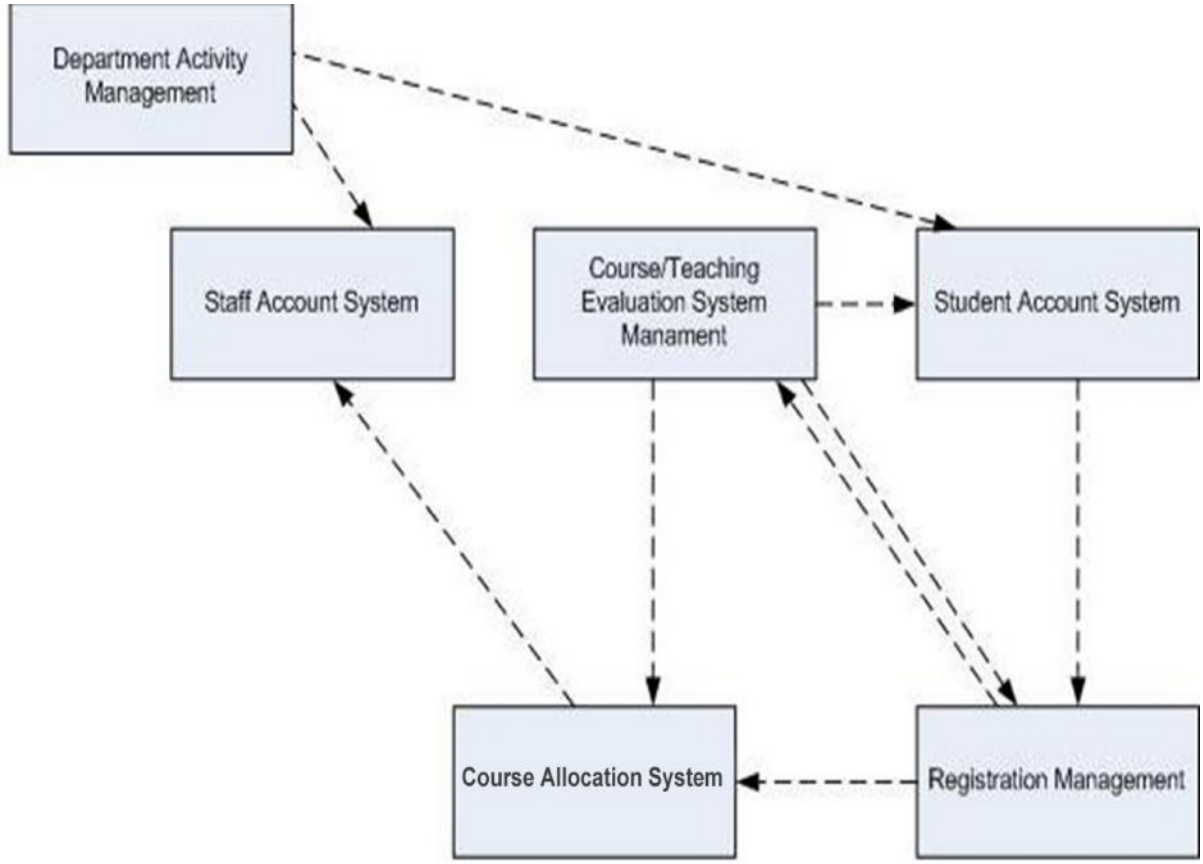


Figure 4.5: Student Evaluation of Teaching (SET) Module Diagram

Figure 4.5 depicts the functional flow of the student evaluation of teaching (SET) system. This module is designed specifically to improve learning and teaching processes. As explained in this section above, this module has been integrated between courses allocated to staff and courses registered by the students. The result of the student's assessment will be available to various role including staff himself, Head of Department, Dean Faculty and other selected management staff if permitted. The feature will show the result in percentage and display the overall score of the total score.

4.5 System Design (Data Flow Diagram – (DFD))

We now present the detailed design of each composition of Figure 4.6 using the DFD notation. The DFD tool depicts the flow of data through a system and the work or processing performed by that system (Whitten & Bentley, 2008).There are only three symbols and directional arrow keys.

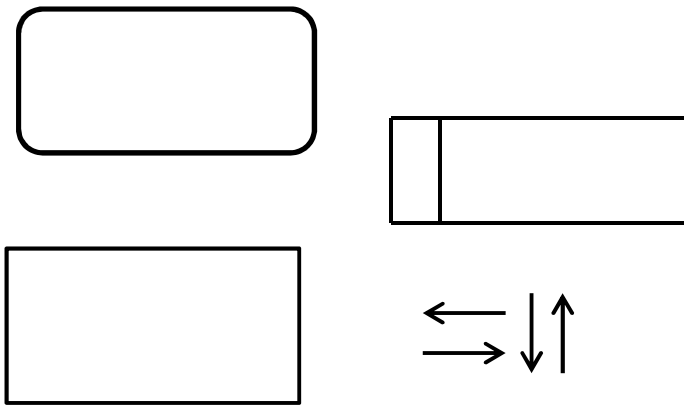


Figure 4.6: Data Flow Diagram Symbols

In fig. 4.6, the rounded rectangles represent *process or* work to be done. Notice that they are illustrated in the PROCESS color from your Information system framework.

a. The squares represent *external agents-the* boundary of the system. Notice that they are illustrated in the INTERFACE color from your information system framework.
b. The open-ended boxes represent *data store;* sometimes called files or databases. These data stores correspond to all instances of a single entity in a data model. Accordingly they have been illustrated with the DATA color from information systems framework.
c. The arrows represent *data flows, (*or inputs and outputs) to and from the processes.
d. The detailed DFD for WeBAIS is given in Appendix B.

4.5.1 Difference with Flowcharts

There is sometimes a tendency to confuse data flow diagrams with flowcharts because program design frequently involves the use of *flowcharts.* However, *data flow* diagrams are very different. Let's summarize the differences:

a. Processes on a data flow diagram can operate in parallel. On the other hand, processes on flowcharts can execute only one at a time.
b. Data flow diagrams show the flow of data through the system. Their arrows represent paths down which data can flow. Looping and branching are not shown. On the other hand, flowcharts show the sequence of processes or operations in an algorithm or program. Their arrows represent pointers to the next process or

operation. This may include looping and branching.

c. DFD can show processes that have different timing. For example, a single DFD might include processes that happen hourly, daily, weekly, yearly, and on demand. This doesn't happen in flowcharts. (Whitten & Bentley, 2008)

4.6 Web Interface Design

Users are a central part in the design of web interfaces. In general, we tried to reduce the number of clicks the users have to make to get to the information they need. For all applications, we had different levels of user groups study before we released the applications. During this preliminary release, we got feedback from them regarding the way the information was delivered, the ease of use, how intuitive the interface was, and what other options users would like WeBAIS to provide. We also discovered some user preferences which were easy to accommodate and greatly improves user's efficiency in our system. We accept feedback from active users after the release the aim of which is to improve the user's experience.

WeBAIS supported over 9,000 users in the first year. These users vary greatly in the way they interact with the system which affects the specification of how we deliver information through web interface to different users. For applicants, we provided navigation links to all information they need to access in the header of the home page. When a user is a current member of our University, they normally have different reasons for using our system. Therefore, we provided a user friendly index page with all the options that are needed. We made the access to this index page easy to access by putting a link to it in the header of

every page.

We make it based on various user role accesses which provide user with available options defined by that role.

4.7 Database Design

We followed the outlines in Chapter II Section D for database design. In particular, we did a user group study to analyze system requirements before the design. We then use the ER model for the data modeling. In term of DBMS, we chose MySQL, which is one of the world's most popular open source databases. About 60 tables support the production use outlined in Chapter I (See Appendix C for database tables used in WeBAIS). Note that the web interface, which is implemented using PHP in our code, is the main way we provide for all users to manipulate the data in the database.

Below are some key points on the role manager, the content manager and the SET Database.

4.7.1 User Role Database Structure

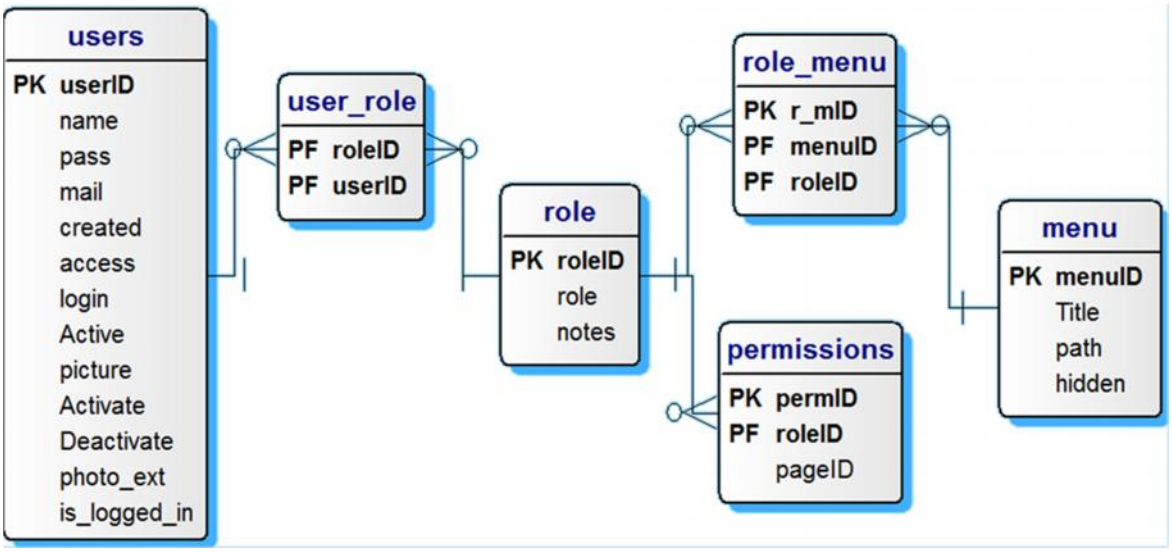


Figure 4.7: User Role DB Structure

Figure 4.7 describes the detailed E–R diagram of the users and their corresponding roles. Each user has menu and roles. Menus are the list of options displayed before the user while permissions are the set of available access points. In this case a user may have access permission to a particular content that did not include in the list of options or menu.

4.7.2 Content Management DB Structure

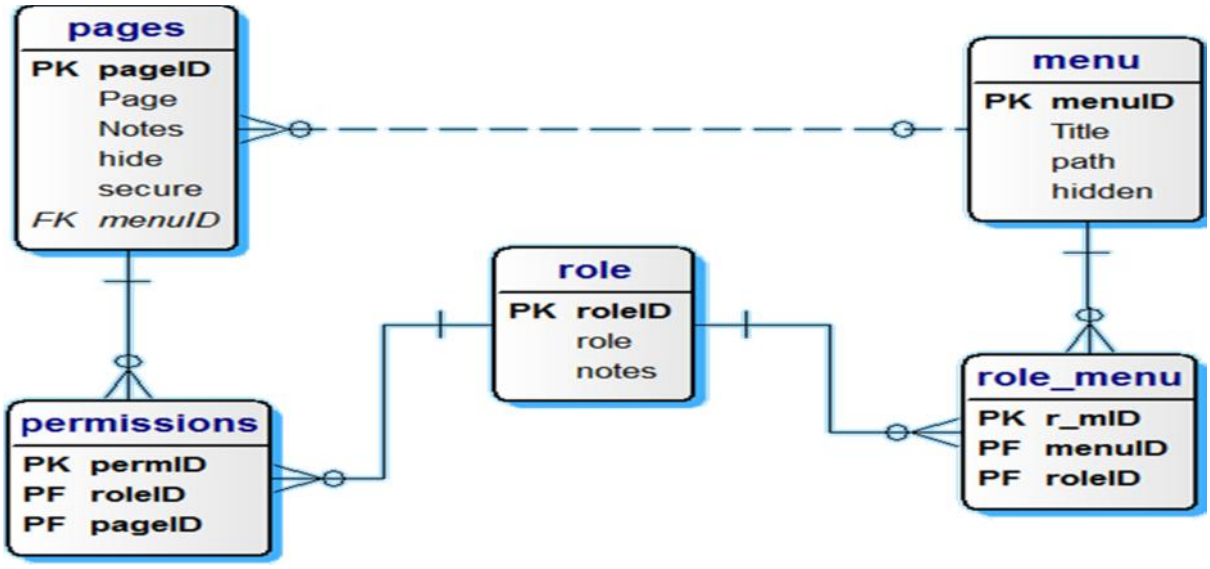


Figure 4.8: Content Management DB Structure

Figure 4.8 shows the E–R diagram of the contents. Each content is managed separately to identify who has privilege to access the content, and also which content is to be secured and which is to be access publically. It also associates contents with menus so as to let users have contents links in their accounts.

4.7.3 SET – Database Structure

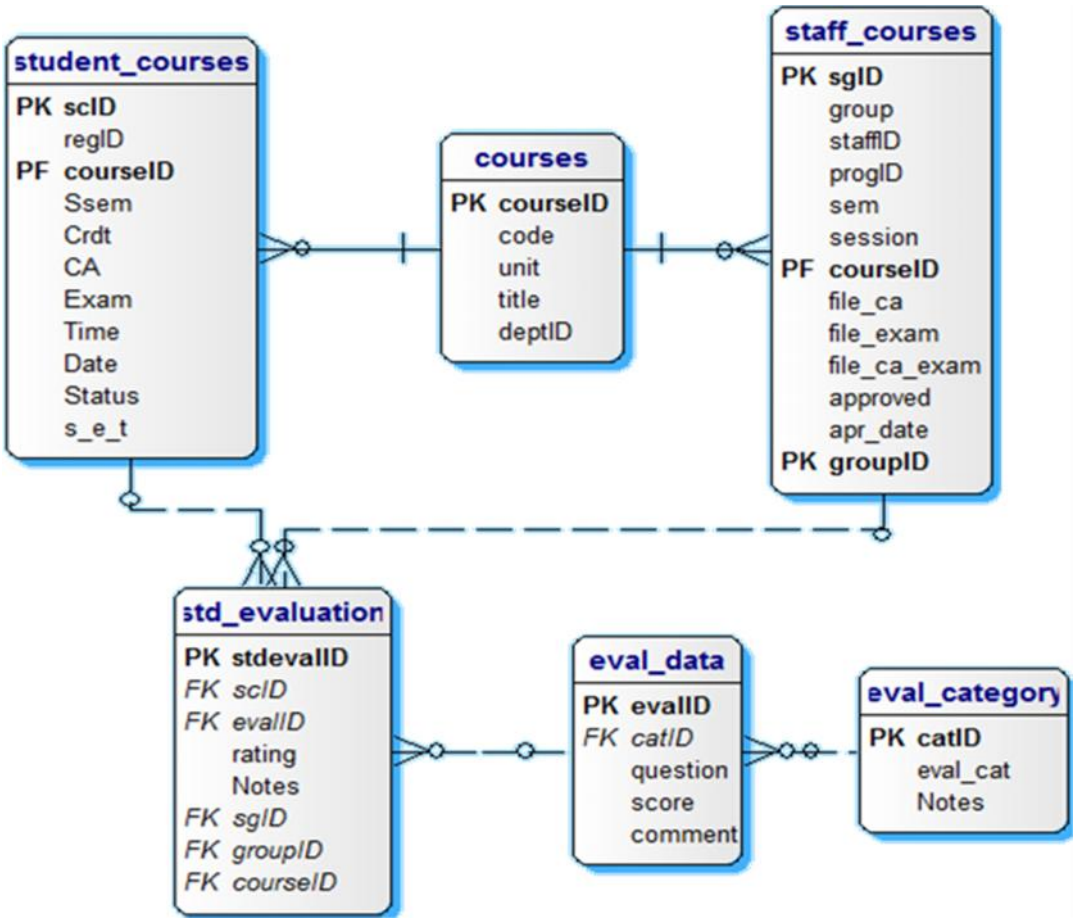


Figure 4.9: SET – Database Structure

Figure 4.9 show the E–R diagram of the Student Evaluation of Teaching (SET) which is one of the significant modules on this system. It relates students to courses taken and staff teaching those courses at those particular times. The evaluation questions are associated with each course taken by both students and staff. The questions can be changed any time by administrator.

4.8 Code Design

To develop a system that can be easily maintained, we applied software engineering techniques. The Waterfall model (Sommerville, 1996) has been widely used to develop large systems. It starts with definition and analysis of requirements so that the system can be divided into different components. The next phase is the design of those components, followed by the implementation and evaluation of each component, and finally integrating them and testing the system as a whole.

Code in WeBAIS is written in PHP using the object-oriented programming technique. Code is the bridge between the web interface and the database. All web interfaces that users see and all underlying database operations are implemented in different classes. WeBAIS implementation has around 75 classes totaling about 56,000 lines of code.

4.8.1 Web Utility Classes

Different components of WeBAIS share common functionality in different ways. Some common functionality is required in all applications. For example, all applications need functionality to upload and manage documents. Some similar functionality is used in many applications but they differ from each other slightly.

To increase the re-usability and maintainability of our code, we introduced the idea of web utility classes. We designed and implemented around 40 utility classes (Figure 4.10). They are base classes for all components in WeBAIS. Building on top of the web utility classes, each component only needs to provide component-specific functionality. As the base

classes become more and more complete, building a new application becomes easier.

We partitioned these classes into four sets: authentication related classes, user information related classes, on-line application related classes, and tool related classes. The first three sets of classes in Figure 4.10 share some common functionality.

a. Web interface functionality includes the function to display various web interfaces for different group of users (an individual input form for data input, non-editable table for data display, summary information for a set of records, etc.).
b. Database management functionality includes all database operations (load data from the database, manipulate data in the database, search/sort data by different criteria, etc.).

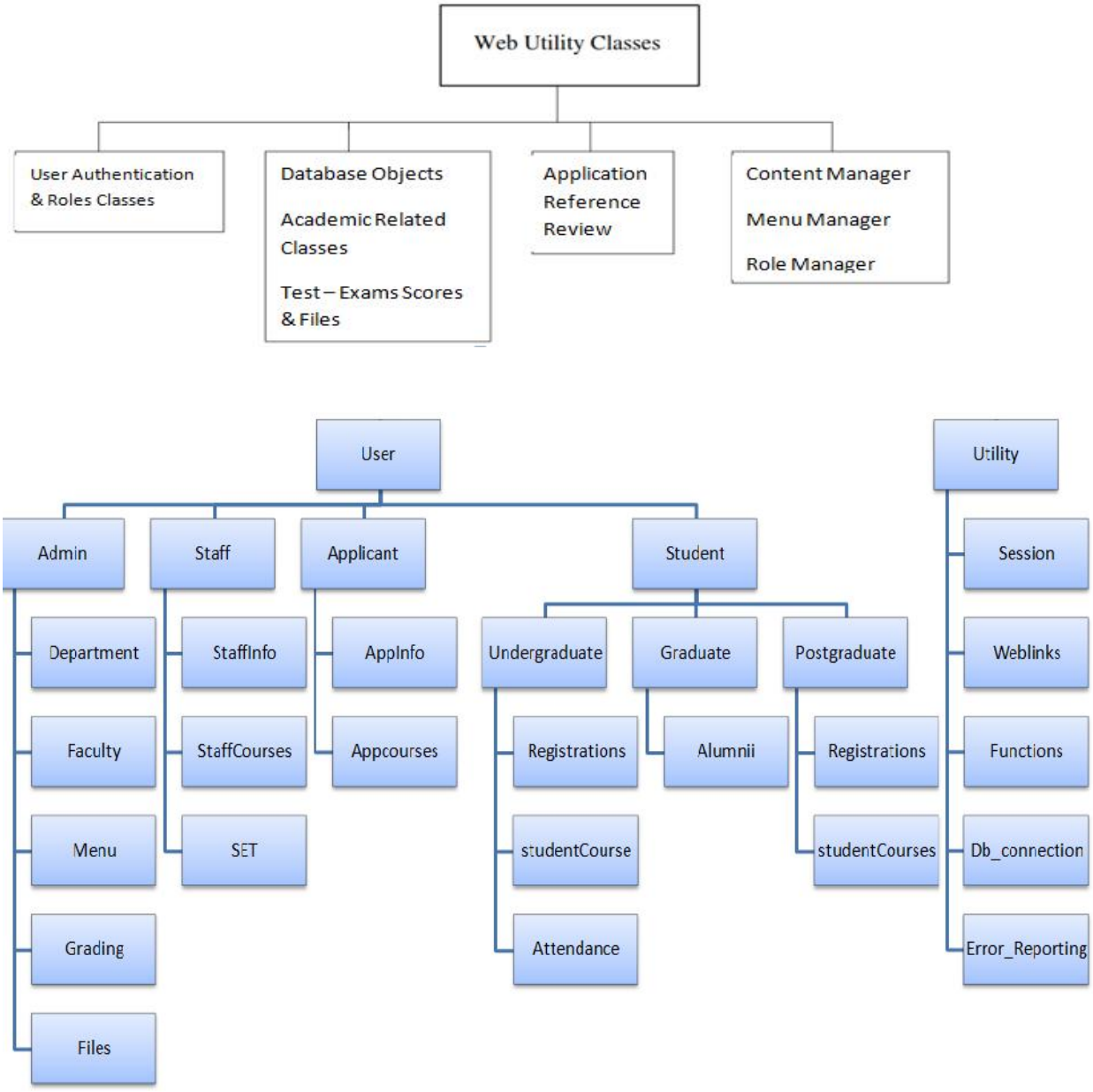


Figure 4.10: Function diagram for web utility classes

4.8.2 Authentication Related Classes

This set of classes provides support for authentication and management of user group membership information.

Authentication related classes provide support for secure ID authentication. This is derived from a base authentication class which provides login and logout functionality.

A single user might have multiple roles at the same time for his/her usage of our system. The same user's role could change over time; a graduate applicant becomes a student. We define a group as a set of individuals identified by a common characteristic. We use group membership to store the user's group information over a period of time. The group membership class has functionality to check whether the user's group membership is valid or not.

4.8.3 Data Object Related Classes

This set of classes provides support for maintaining WeBAIS related information. We provided classes to manage information such as department, faculty, program, courses, etc. It also provides tools to manage a data object's specific information, department's programs, faculty's department, program courses, e.t.c.

4.8.4 Tool Related Classes

This set of classes provides support for different functionality needed in web interfaces. For example, the document manager provides support for document uploading and management for documents: editing of associated information and deletion of documents. HTML utility classes support various HTML input (text input box, drop down menu, etc).

4.8.5 Summary

These utility classes are general and can be directly used or customized for a particular application. Building on top of the web utility classes, the development for each application becomes easier since we only need to add in application specific functionality. For example, WeBAIS-undergraduate uses almost all the classes from utility classes.

4.9 System Security

Information Systems Security is one of the most pressing challenges facing all kinds of organizations today. Although many companies have discovered how critical information is to the success of their business or operations, very few have managed to be effective in maintaining their information secure, avoiding unauthorized access, preventing intrusions, stopping secret information disclosure, etc.

There are various definitions of security, but all of them basically agree on the same components. Security in information systems considers the protection of information and of the systems that manage it, against a wide range of threats in order to ensure business continuity, minimize risks and maximize the return on investment and business opportunities (Whitten & Bentley, 2008).

Security is, therefore, currently a widespread and growing concern that covers all areas of society: business, domestic, financial, government, and so on. In fact, the so-called information society is increasingly dependent on a wide range of software systems whose mission is critical, such as air traffic control systems, financial systems, or public health

systems. The potential losses that are faced by businesses and organizations that rely on all these systems, both hardware and software, therefore signify that it is crucial for information systems to be properly secured from the outset.

4.9.1 Authentication

WeBAIS supports a large number of users. These users vary greatly in the way they interact with the system. There are three main user groups who use WeBAIS.

a. **Long Term Users** are the main user group for WeBAIS. Their usage is a few years. All of them have some association with the University, and they use WeBAIS on a regular basis. In particular, this group includes all staff and students within the University. The total number as at now is getting to 12,000.

b. **Short Term Users** is the most active user group for WeBAIS. Their usage is a few months. This group includes various types of applicants, graduate applicants undergraduate applicants. In the first year, the total number of users in this group was approximate 5,000. Note that a portion of these users will transition to “long term user” once they affiliate with the WeBAIS, e.g., a graduate applicant becomes a graduate student.

c. **One Time Users**is the largest user group for WeBAIS, but their usage for WeBAIS is very limited – could be as little as one time or a few times. In particular, it includes references for different University Websites visitors. In the first year, the total number of users in this group was approximately 30,500.

Two main requirements affect our design of the authentication for WeBAIS. First, we want

it to be as secure as possible. Second, we want it to be as convenient as possible for the user. With a large number of users and a big difference between user groups, no single solution can meet all requirements. WeBAIS provides Central Authentication System (CAS) authentication as described below.

a. **Central Authentication System (CAS) authentication**: CAS authentication is provided by the Web site and all users who have an association with the University are able to use this authentication. It meets both the security and the convenience requirements very well and is our first choice for authentication. In WeBAIS, all long term users and graduate applicants use CAS authentication.

Here is the PHP coding for login authentication

```
if(!$session->is_logged_in() && isset($_GET['username']) && isset($_GET['password'])){
        $user=user::authenticate($_GET['username'], md5($_GET['password']));
        $session->login($user);
        if($session->is_logged_in()){
                $user=user::find_by_id($session->user_id);
                $user->load_menu();
        }
        else {
                redirect_to('access_denied.php')
```

```
    }
}
```

4.9.2 Authorization

The **authorization** allows you to configure the user accounts that can access your site or application. Use authorization in combination with authentication to secure access to content on your server. Authentication confirms the identity of a user, while authorization determines what resources users can or cannot access.

a. Each role has been assigned two functions
    i. Menu Option
    ii. Access Privilege (Permission).
b. Menu: User access functionality options
c. Permission: Access privilege assigned to role.

Below is the code to check for authorization

```
if($user->has_permission($_SERVER['PHP_SELF'])){
        // Do stuff here…
}
else {
        redirect_to("access_denied.php");

}
```

4.9.3 Encryption and Decryption

Some information inside the DB are sensitive

Using an encryption algorithm, some data elements have been encrypted. These include:

User Password (md5 encryption, no decryption)

CA and Exam Score (encryption/decryption)

Here is the code for encryption and decryption

a. Code for encryption

```
function encrypt($sData, $sKey='mysecretkey'){
    $sResult = '';
    for($i=0;$i<strlen($sData);$i++){
        $sChar    = substr($sData, $i, 1);
        $sKeyChar = substr($sKey, ($i % strlen($sKey)) - 1, 1);
        $sChar    = chr(ord($sChar) + ord($sKeyChar));
        $sResult .= $sChar;
    }
    return encode_base64($sResult);
}
```

b. Code for decryption

```
function decrypt($sData, $sKey='mysecretkey'){
    $sResult = '';
    $sData   = decode_base64($sData);
    for($i=0;$i<strlen($sData);$i++){
        $sChar    = substr($sData, $i, 1);
        $sKeyChar = substr($sKey, ($i % strlen($sKey)) - 1, 1);
        $sChar    = chr(ord($sChar) - ord($sKeyChar));
        $sResult .= $sChar;
    }
    return $sResult;
}
```

4.10 Summary

In this chapter, we describe what is needed to design WeBAIS; explained the design architecture, application architecture; we also explain web interface design, database design, code design and system security design. In the next chapter we will discuss System Deployment and Appraisal.

CHAPTER FIVE: DEPLOYMENT and APPRAISAL

This chapter discusses results and usage statistics on WeBAIS in the first month of production and implementation status.

5.1 Introduction

WeBAIS has made a big impact on the University day-to-day operation and receive recognition from different types of users.

Within the first year of development, we have successfully released the following main applications: undergraduate admission, undergraduate registration, undergraduate result processing and graduation. We have implemented about 60 classes (around 56,000 lines of code) in PHP and several tools for publishing and data importing from Excel and PDF.

The total number of users is around 12,000. In particular, 37% of our current users are references for various types of applicants, 49% of our current users are various types of students, and the remaining 14% of our current users are current members of University Staff.

5.2 WeBAIS Implementation Status

Chapter III described the requirements we determined that were necessary for a system such as WeBAIS to support the information management for a modern university. In this section we outline the implementation status of the WeBAIS system.

We give a list of functionalities implemented and deployed according to different user categories as follows.

a. Applicant

i. Applicant Admission Tracker

ii. Applicant Biodata

iii. Applicant JAMB/DE Form

iv. Applicant JAMB/DE Slip

v. Applicant SSCE/GCE Result

b. Bank

vi. Bank Page for Applications and Registrations

vii. Sales Report by Bank User

c. Admission Committee

viii. Committee Admission Process Page

ix. Course Evaluation Form

d. HOD, Exam Office

x. Departmental Course Groups

xi. Departmental Courses

xii. Departmental Staff List

xiii. Departmental Students List

xiv. HOD Admission Recommender

xv. Exam. Semester Result Summary

xvi. First Semester Exam Result

xvii. Letter Grade Exam Report

xviii. Level Coordinators of the Dept.

xix. List of Uploaded Records by Lecturers

xx. Semester Result Letter Grade

e. Staff

xxi. Staff Profile

f. Teaching Staff

xxii. Staff Courses Allocation by HOD

xxiii. Upload Exams Records (Excel Files)

xxiv. Uploading Approved Results to Database

g. Student

xxv. Student Biodata

xxvi. Student Course Registration

xxvii. Student Evaluation Menu

xxviii. Student Registration Menu

xxix. Student Result Semester

xxx. Student Result Session

xxxi. Students List per Level

Furthermore, all requirements for the University Academic (e.g., student management, course management, course allocation, results processing and collection, etc) except matters relating to administration are implemented.

5.2.1 WeBAIS Deployment Requirements

WeBAIS is a web – based application run on aphace server, it requires server – side tools and client side tools as follow:

a. Server Side (minimum) Requirement

i. Hardware (PC)

a) RAM 1.0GB

b) Processor 1.70 GHz

ii. **Software**

a) Windows/Linux/Mac OS

b) Aphace Server

c) MySQL Server

d) PHP Program Engine

e) JQuery

f) Browser

b. Client Side (minimum) Requirement

i. **Hardware**

a) PC, Tablet, Smart Phones/iPoids

ii. **Software**

a) Browser (JavaScript Enabled)

5.3 Deployment

WeBAIS was deployed at two universities for pilot study. In both universities, users were given acess to the application according to their user groups and roles. The deployment universities and sites are:

a. KUST – Wudil (kust.com.ng)
b. North West University, Kano (nwu-ces.org)

In the following sections, we briefly outline how questionnaires were designed to appraise WeBAIS, WeBAIS deployment requirements, presentation and analysis of questionnaire results.

5.3.1 Questionnaire design

In order to evaluate WeBAIS, we developed a questionnaire targeted at different users of the system for their feedback. The questionnaire was meant to solicit users' input on the following aspects of WeBAIS:

a. WeBAIS' potential for improving efficiency of service delivery in a University
b. SET's prospects for improving the learning process
c. Robustness and security of WeBAIS
d. WeBAIS' user-friendliness and ease of use

There are a total twenty five (25) questions in the questionnaire as shown in Appendix B.

5.3.2 Results Presentation and Analysis

Based on this questionnaire above, we produce 100 copies of which we are able to collect back 82 from which we analysed their responses below:

a. We obtained responses from 82 (Paper Based Form)
b. Staff category (32 responses)
c. Exam officer/Level Coordinator category (15 responses)
d. Management category (8 responses)
e. Students category (38 responses)
f. Total responses(82 responses)

From the above responses we observed each question and summarized the result for each question below:

Table 5.1: Questionnaire Response Summary

	Options (1-Strongly Agree, 2-Agree, 3-Not sure, 4-Disagree, 5-Strongly Disagree),(1-Yes, 2-No), (1-100%, 2-50%, 3-25%, 4-10%, 5-Other)					
Questions	**1**	**2**	**3**	**4**	**5**	**Total Responses**
1	8	15	21	38	0	**82**
2	43	39	0	0	0	**82**
3	8	19	23	16	16	**82**
4	13	24	16	17	13	**82**
5	10	19	19	26	8	**82**
6	12	12	25	21	12	**82**
7	11	25	20	18	8	**82**
8	23	38	21	0	0	**82**
9	20	33	23	0	0	**82**
10	0	6	4	6	5	**21**
11	1	13	6	1	0	**21**
12	7	8	4	1	1	**21**

13	8	9	3	1	0	**21**
14	9	17	10	0	0	**36**
15	5	24	7	0	0	**36**
16	11	17	8	0	0	**36**
17	11	17	8	0	0	**36**
18	7	14	8	0	0	**29**
19	3	2	3	0	0	**8**
20	7	12	10	0	0	**29**
21	10	15	4	0	0	**29**
22	7	15	13	3	0	**38**
23	8	14	13	2	1	**38**
24	14	19	5	0	0	**38**
25	14	14	10	0	0	**38**

Below are some graphical representation of some key aspect of the questionnaire:

Q4. WeBAIS improves efficiency of service delivery in University

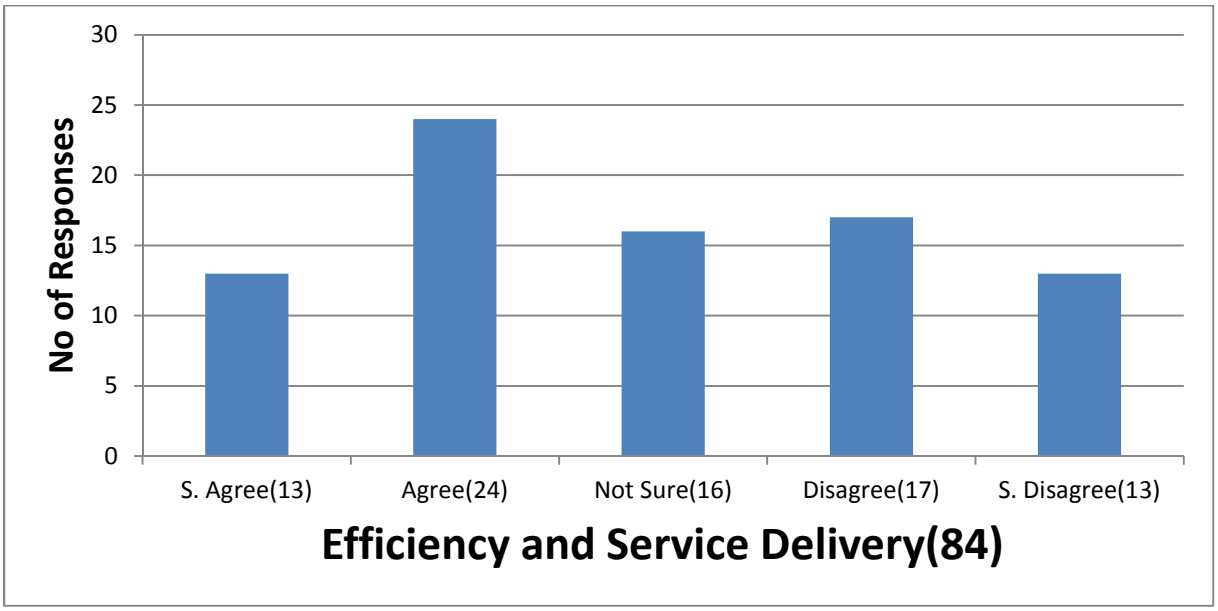


Figure 5.1: Efficiency and Service Delivery

a. The results show that WeBAIS can improve efficiency of service delivery by up to 56%

SET's prospects for improving the learning process

Q10. As instructor, student ratings using applications like WeBAIS, motivates me to improve teaching

And Q22. As a student, rating instructors using applications like WeBAIS, motivates me to participate more in the teaching evaluation process

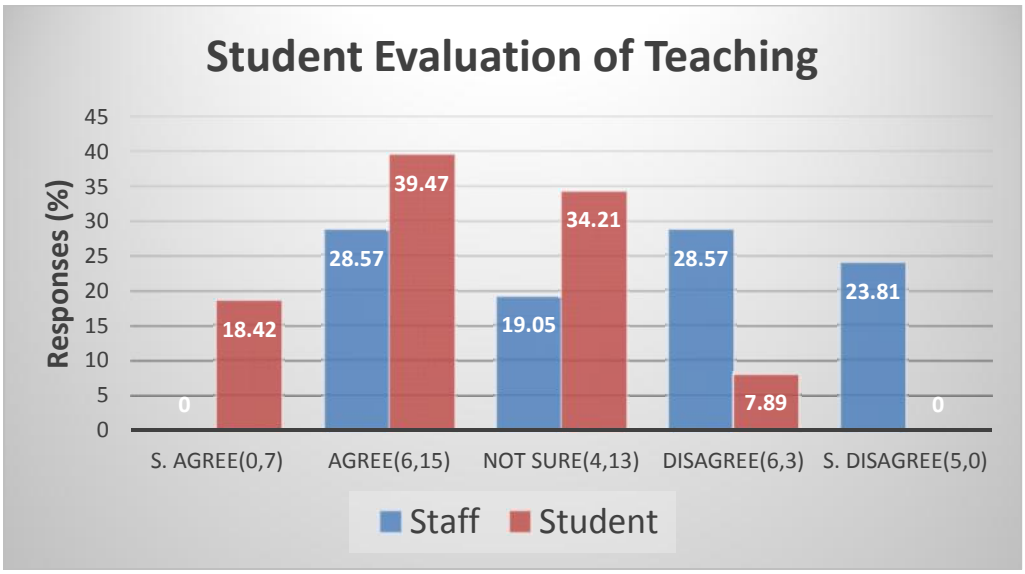


Figure 5.2: Student Evaluation of Teaching (SET)

The SET has not been accepted well by the teaching staff but accepted by students:

– Only 28.6% of staff agree or strongly agree that SET can improve the learning process

– About 58% of students agree or strongly agree that SET can improve the learning process

Robustness and security of WeBAIS

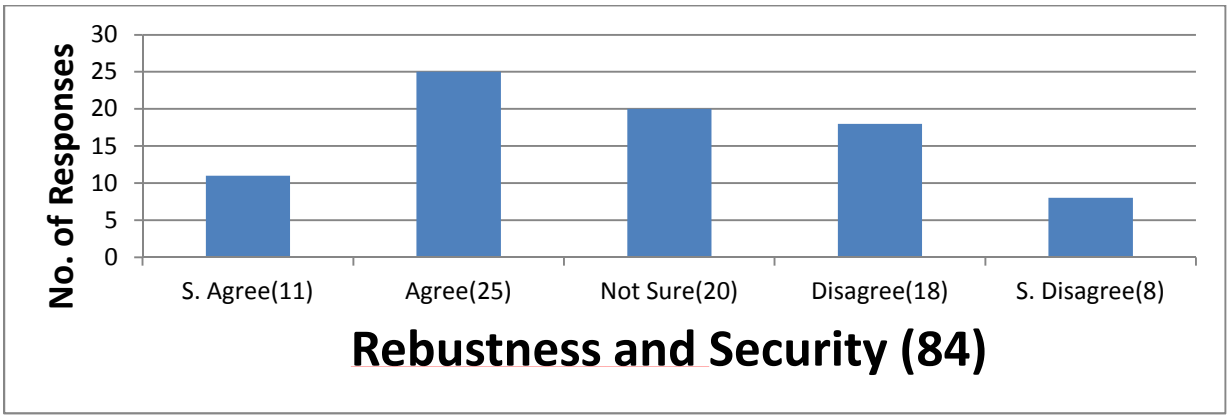


Figure 5.3: Robustness and security of WeBAIS

a. The results show that WeBAIS is robust and can improve security by up to 58%

WeBAIS' user-friendliness and ease of use

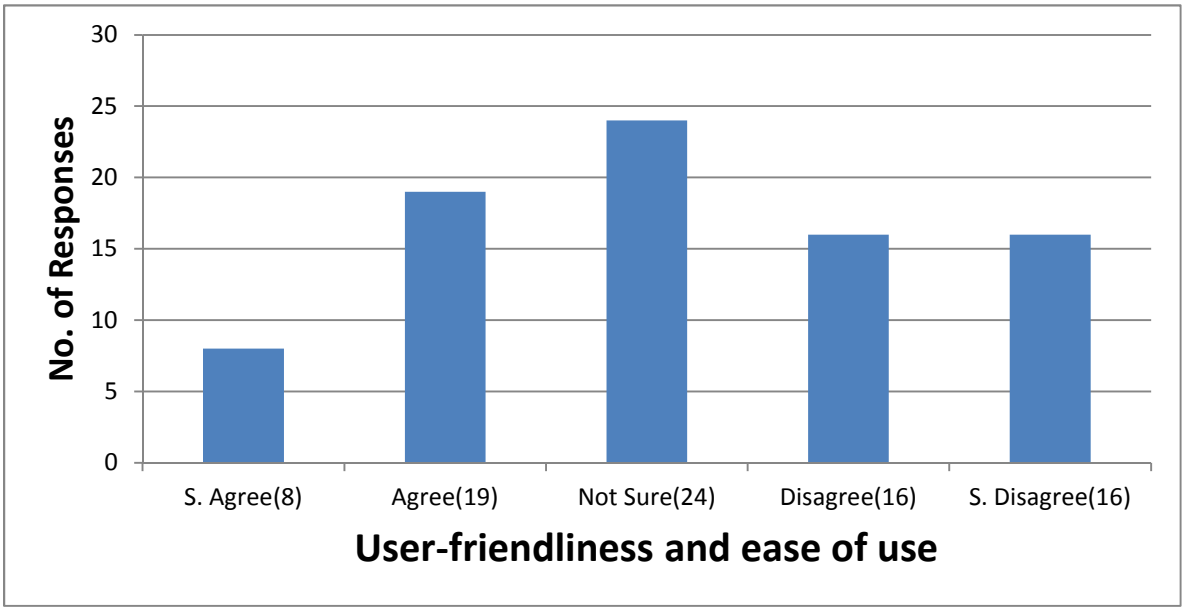


Figure 5.4: WeBAIS' user-friendliness and ease of use

Only 32.9% of the responses agree or strongly agree that WeBAIS is user – friendly

a. About 28% not sure

b. Result proves that WeBAIS needs to be improved on user friendliness of the system

5.4 Summary

Despite the challenges, time consuming, cost implication as well as difficulties in the entire project design, we are able to present some of the expected goals have been set by this research, we have seen the significant improvement in university academic management, in terms of robustness, efficiency, user friendliness and teaching/learning evaluation for improving teaching quality.

In the next chapter we will discuss about general summary of the research, limitation and future work to be implemented.

CHAPTER SIX: SUMMARY, CONCLUSION AND RECOMMENDATION

6.1 Summary

In this report, WeBAIS has been introduced and defined, research question have been raised as well as objective of the research (Chapter I). Relative literatures have been critically observed and some of the important key problems to those literatures have been pointed out; and way to overcome those problems as the basis for developing WeBAIS (Chapter II). In (Chapter III), definition of requirements and functionality of the system needed to be implemented have been described in details. Chapter IV presents the design of the system including architecture, database and coding method. Chapter V describes the deployment, user appraisal and results of the user responses.

Finally, in the last chapter, a summarized paragraph (section 6.1) presented of all the chapters except the last chapter, following the conclusion and recommendation for the future work of this thesis.

6.2 Conclusion

We presented a University Web – Based Academic Information System called WeBAIS. It was developed following Information Science design paradigm, Software Engineering techniques, and Computer-Human Interaction guidelines. It is a robust, large-scale information system for Nigerian Universities, well-designed object- oriented application. We also used Student Evaluation of Teaching (SET), which is a tool to monitor and for the purpose of improving teaching performance by the teaching staff. This feature

has not been implemented by any university digital system in Nigeria. The Application has been deployed and in use at two universities and attested by users to meet its design objectives, largely. It has also been deployed currently to Kano State Polytechnic for Application, Admission and Registration exercise (after some adjustments). Many more Colleges, Institutions are about to deploy the system including Universities.

WeBAIS has made a big impact on the University daily operation. It improves the efficiency of many operations. The difference in processing time is significant. There are many things we can do now with WeBAIS that we could not do before.

In WeBAIS, we designed and implemented a set of web utility classes which are general and self-contained. They greatly increase the re-usability and maintainability of our codes. The web utility classes can be used by other web applications. We plan to make them open source in the future.

Although WeBAIS is a customized information system for the University Academics (of Kano University of Science and Technology, Wudil) as a case study, it can be easily configured to be used by other Universities, Mono &Polytechnics, Colleges and other Institutions.

Lastly, WeBAIS achieved its designed objective as obsebed in chapter V. User responses show that WeBAIS made significant impact to university daily operation.

6.3 Recommendation

WeBAIS only implement some aspect of Academic Activities of University. It has not cover all aspect of university activities that are part of academic. It has not also include many of administrative activities. Below are some of these activities that have not been included:

a. No Schedulers {Timetables}

b. No administrative functions such as dispatch and approvals

c. No HR, Payroll, Registry, Library, Accommodation and etc.

In the future, we recommend this work to implement all of the above listed functions to make a complete University information System (UIS):

References

1. Baecker, R., Grudin, J., Buxton, W., and Greenberg, S. (2005). *Reading in Human-Computer Interaction: Toward the Year 2000.* New York: Morgan Kaufmann Publishers Inc. 2nd Edition.
2. Baecker, R., J.Grundin, Buxton, W., and Greenbergs, S. (2000). Reading in Human-computer Interaction. *New York*, 1.
3. Bharamagoudar, S., Geeta, R., and Totad, S. (2013). Web Based Student Information Management System. *nternational Journal of Advanced Research in Computer and Communication Engineering Vol. 2*, 2343.
4. Chen, P.P.S. (1976). The Entity-relationship Model Toword a Unified View of Data. *ACM Transactions on Database Systems*, 9-36.
5. Chen, Y., and Hoshower, L. B. (2003). Student Evaluation of Teaching Effectiveness: an assessment of student perception and motivation. *Assessment and Evaluation in Higher Education, Vol. 28, No. 1*, 71.
6. Crowston, K., and Malone, T. (1994). The Interdisciplinary Study of Coordnation. *ACM Computing Surveys*, 119.
7. Date, C. J. (1990). *An Introduction to Database.* Boston,addition-wesley.
8. Denning, P. (1997). A New Social Contract for Research. *Communication of the ACM*, 132,134.
9. Ellis, C., Gibbs, S., and Rein, G. (1991). Groupware: Some Issues and Experience. *Communication of the ACM*, 58.

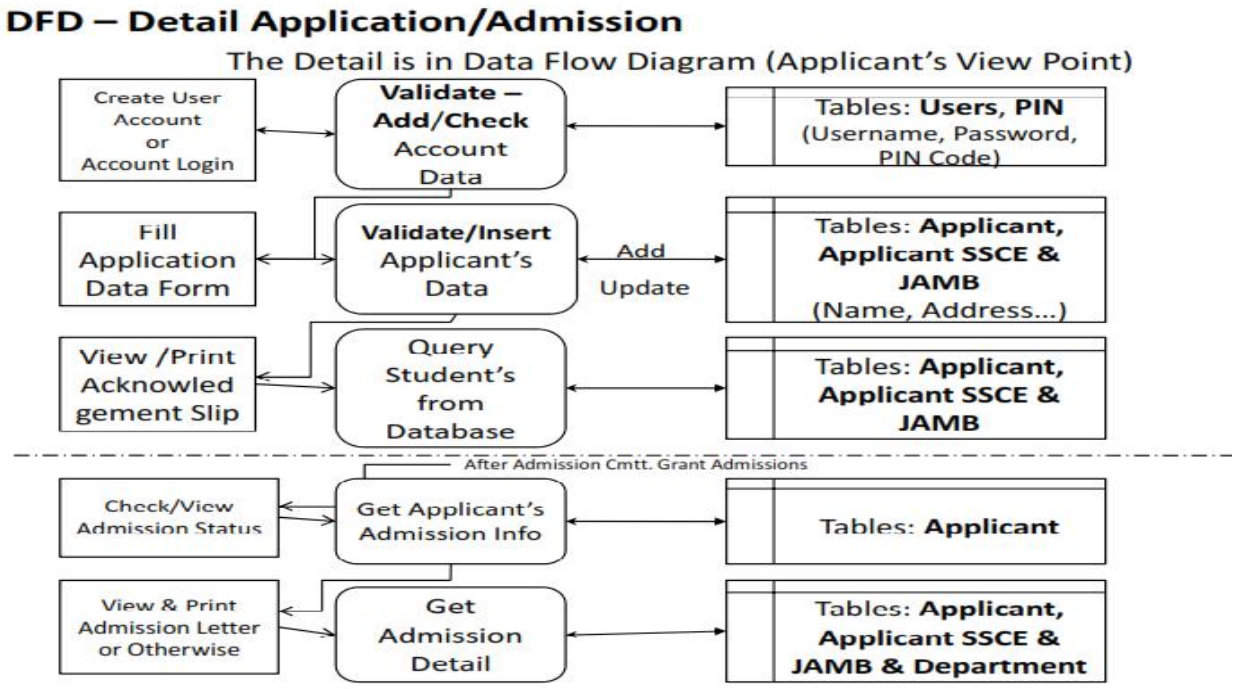
10. Fersko-Weiss, S. O. (1992). *Enhancing Productivity in Networked Organization.* new york: Van Nostradreihold.
11. Gonçalves, N. P., and Sapateiro, C. M. (2008). Aspects for Information Systems Implementation: challenges and impacts, A higher education institution experience. *Revista de Estudos Politécnicos Polytechnical Studies Review*, 9.
12. Gould, J. (2000). *How to Design Usable System.* Newyork: Morgan Kaufmann.
13. Grundin, J. (1998). *Problem in the Design and Evaluation of Organization Interfaces.* IEEE.
14. Grundin, J. (2005). *Computer Supported Cooperative Work History and Focus.* IEEE.
15. Hevner, A. R., March, S., Park, J., and Ram, S. (2004). Design Science Research in Information System. *MIS Quarterly*, 105.
16. Hevner, A., March, S., Park, J., and Ram, S. (2004). Design Science Research in Information System. *MIS Quarterly*, 105.
17. ICEIS Special Sessions - WOSIS, 2. (2014, febuary 23). *Security in Information Systems*. Retrieved from 17th International Conference on Enterprise Information Systems - ICEIS 2015: http://www.iceis.org/wosis.aspx
18. Identifiers(UUID), Universal Unique. (n.d.). *Universal Unique Identifiers(UUID)*. Retrieved January 2007, from Universal Unique Identifiers(UUID): http://www.itu.int/ITU-T/studygroups/com17/oid.html
19. *Introduction to Data Modeling*. (2007, March 14). Retrieved january 2007, from University of Texax: http://www.utexas.edu/its/windows/database/

20. Kumar, B. A. (2011). THIN CLIENT WEB-BASED CAMPUS INFORMATION SYSTEMS FOR FIJI NATIONAL UNIVERSITY . *International Journal of Software Engineering and Applications (IJSEA), Vol.2, No.1,* 13.
21. Malone, T., and Crowston, K. (1994). The Interdisciplinary study of Coordination. *ACM Computing Surveys*, 119.
22. March, S., and Smith, G. (1995). Design and Natural Science Research on Information Technology. *Decision Support System*, 2661.
23. Method, S. a. (n.d.). *System Analysis and Design Method.*
24. Nielsen, J. (1994). *Usability Engineering.* San Francisco.
25. Obasi, N., Nwachukwu, E. O., and Ugwu, C. (2013). A Novel Web-Based Student Academic Records Information System. *West African Journal of Industrial and Academic Research* , Vol.7 No. 1.
26. Obiniyi, A. A., and Absalom, E. E.-S. (2010). Design and Implementation of Students' Information System for Tertiary Institutions Using Neural Networks. *International Journal of Green Computing*, 1-15.
27. Petrelli, D., Angeli, A., and Convertino, G. (1999). A User-centered Approach to User Modelling. *In Proc. ACM, International Conference on User Modelling*, (p. 2255).
28. Philipa, I., Kenneth, A., and Agajo, J. (2011). Interactive Intranet Portal for effective Management in Tertiary Institution. *International Journal of Advanced Computer Science and Applications*, 122-129.

29. Preprocessor, the PHP Hypertext. (2012, August). *The PHP Hypertext Preprocessor*. Retrieved january 2007, from The PHP Hypertext Preprocessor: http://www.php.net
30. Simon, H. A. (1996). *the science of the artificial*. cambridge.
31. Smith, G., and March, S. (1995). Design and Natural Science Research on Information Technology. *Decision Support System*, 2661.
32. Sommerville, I. (1996). *software engineering.* Boston.
33. SQL, M. (n.d.). *MY SQL*. Retrieved JANUARY 2007, from MY SQL: http://www.mysql.com
34. The Apache Software. (2007, january). *The Apache Software Foundation*. Retrieved from The Apache Software Foundation: http://www.apache.org
35. web. (n.d.).
36. Whitten, and Bentley. (2008). *System Analysis and Design Methods.* Sanfransisco: McGrow Hill.
37. Youh, D. X. (2010). Design and Implementation of a Client Server Distributed Database for Student. *The Pacific Journal of Science and Technology* , 288.

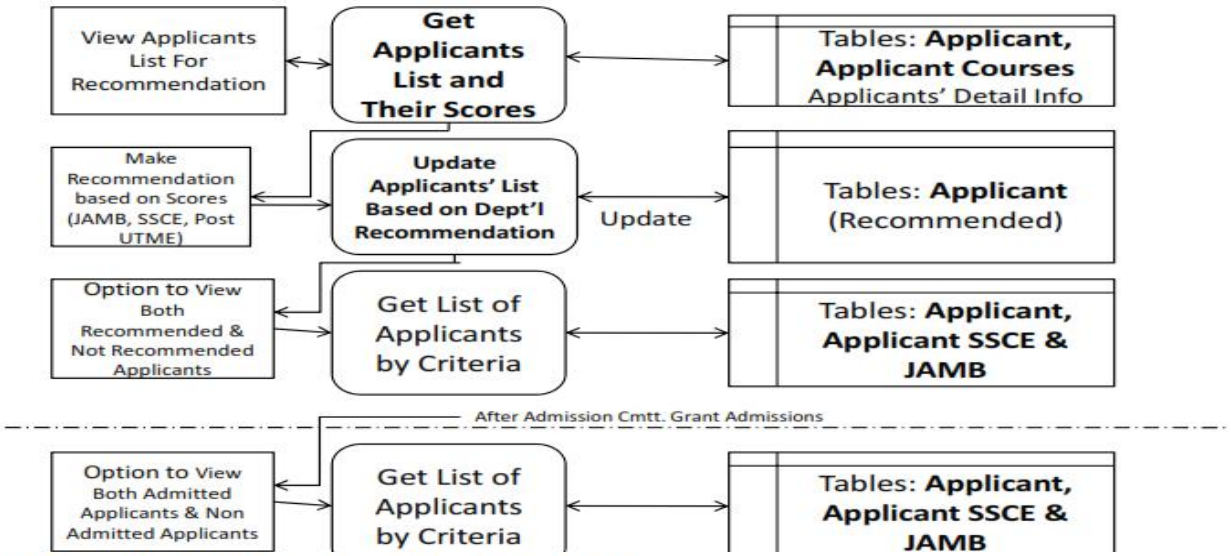
Appendix A: Data Flow Diagram of WeBAIS

Figure A.1: Data Flow Diagram of the Application



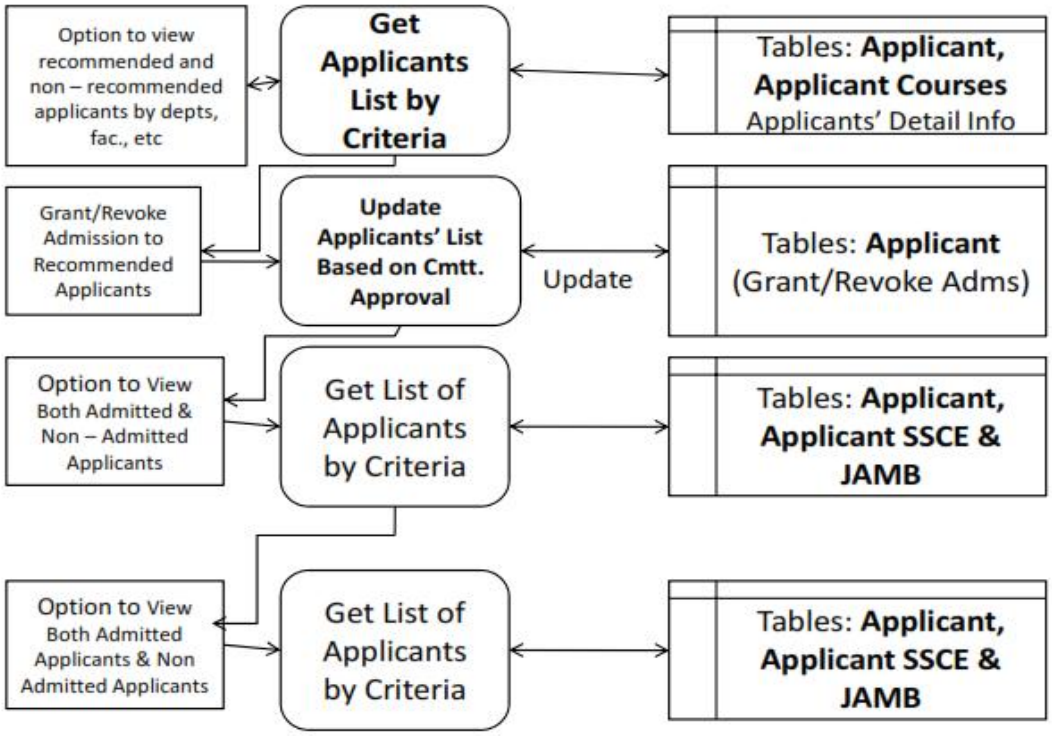
DFD – Detail Application/Admission

The Detail is in Data Flow Diagram (Departmental View Point (HOD))



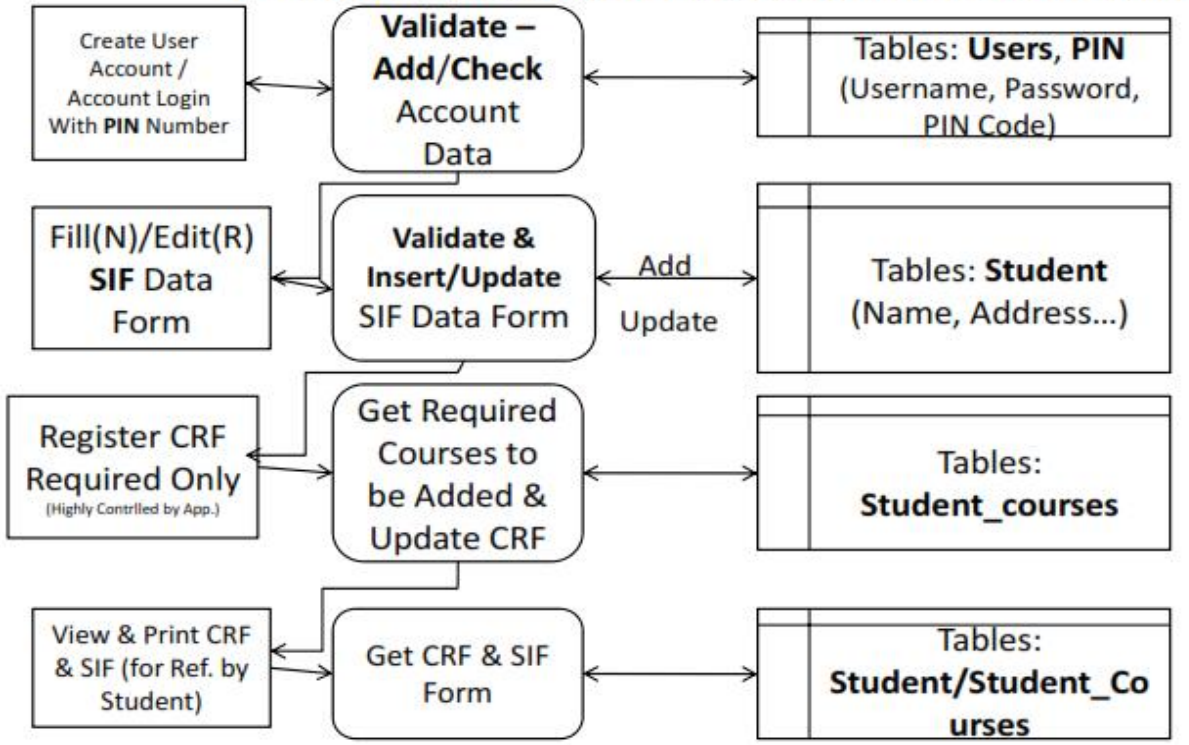
DFD – Detail Application/Admission

The Detail is in Data Flow Diagram (Central Admission Cmtt. View Point)



DFD – Detail Registration

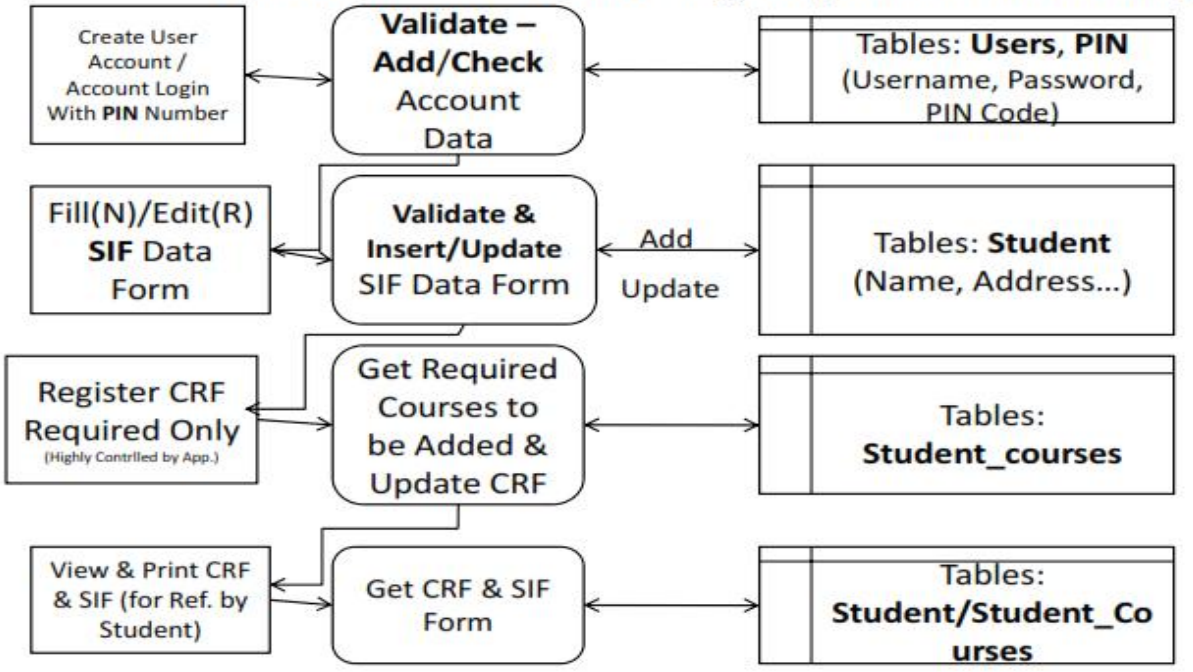
The Detail is in Data Flow Diagram (Student's View Point)



+

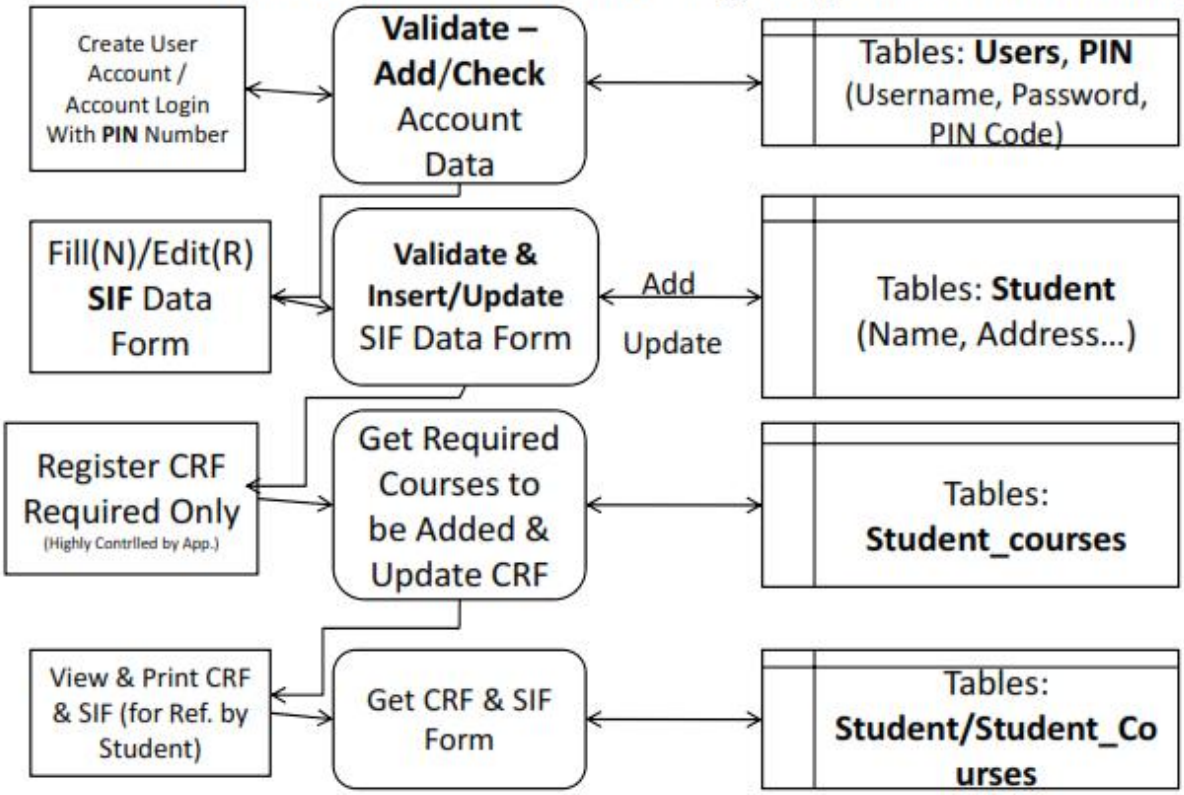
DFD – Detail Registration

The Detail is in Data Flow Diagram (Student's View Point)



DFD – Detail Registration

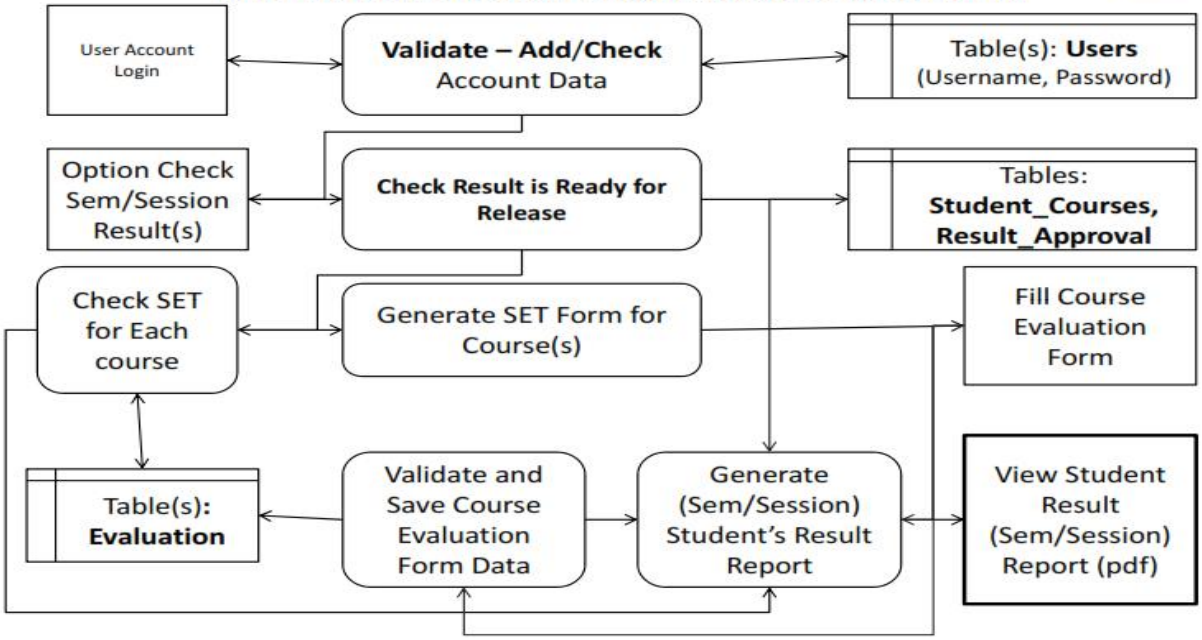
The Detail is in Data Flow Diagram (Student's View Point)



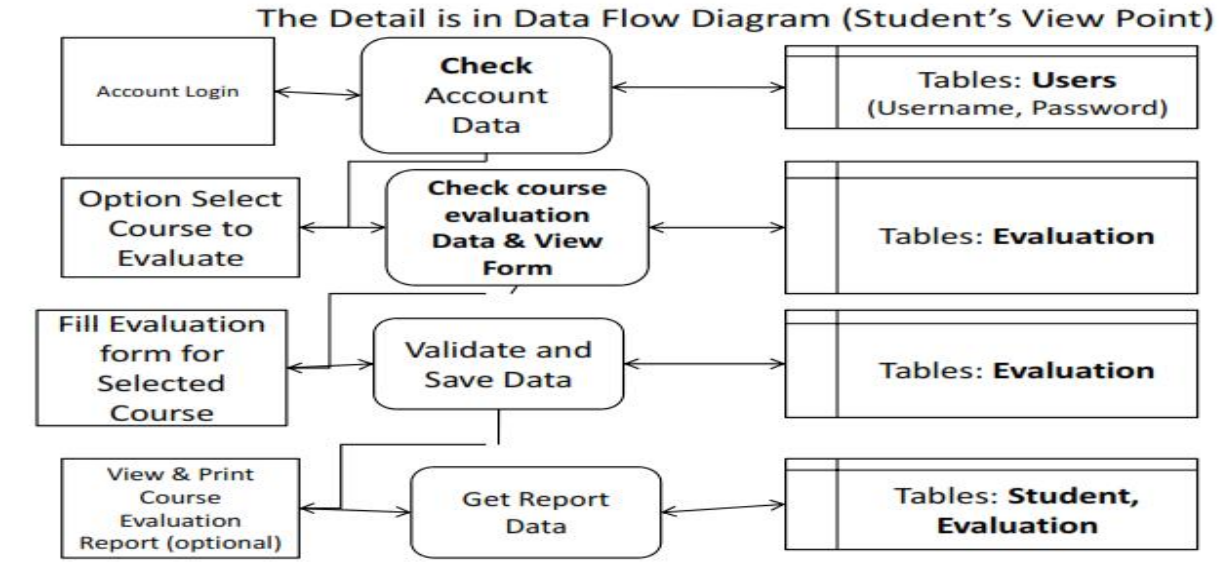
DFD – Detail Result Checking

Back

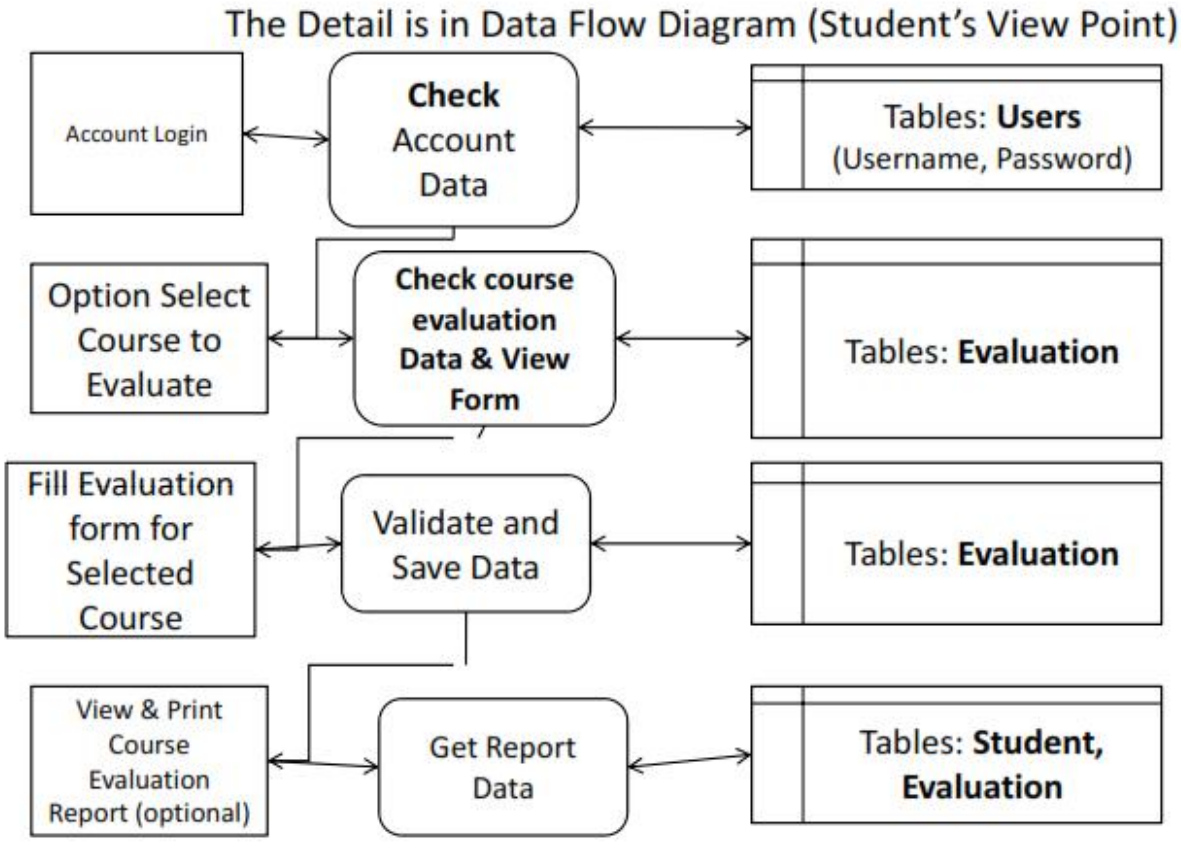
The Detail is in Data Flow Diagram (Student's View Point)



DFD – Detail Teacher Evaluation Form

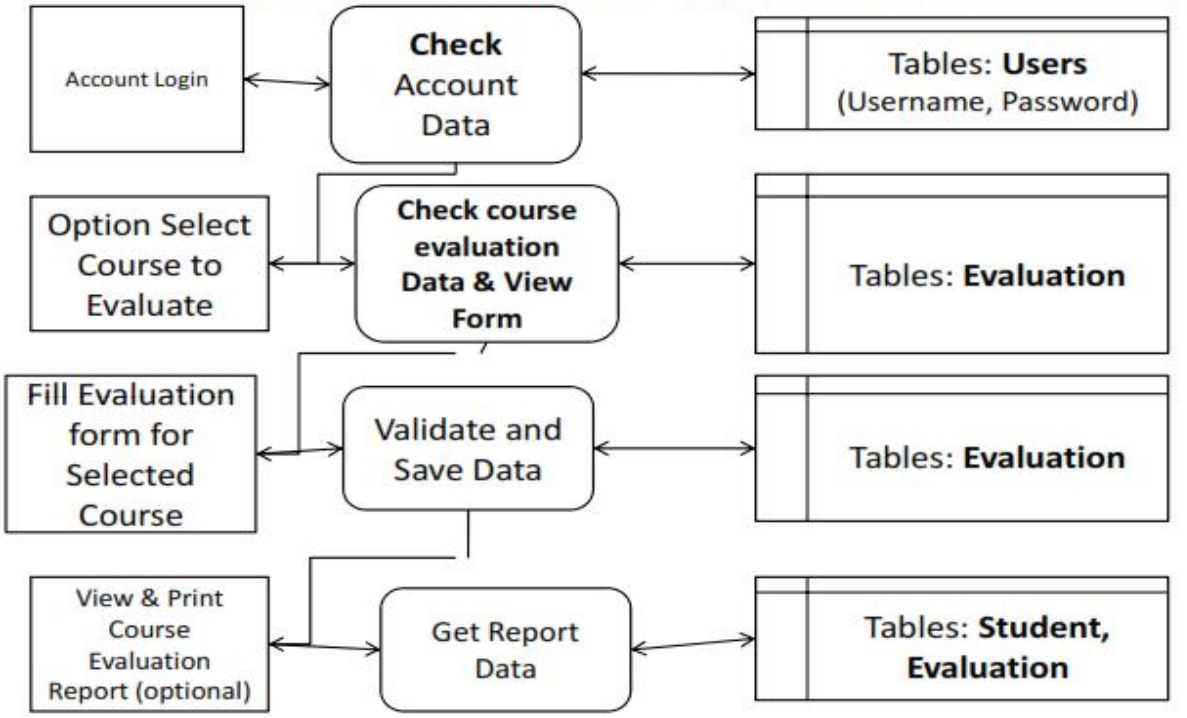


DFD – Detail Teacher Evaluation Form



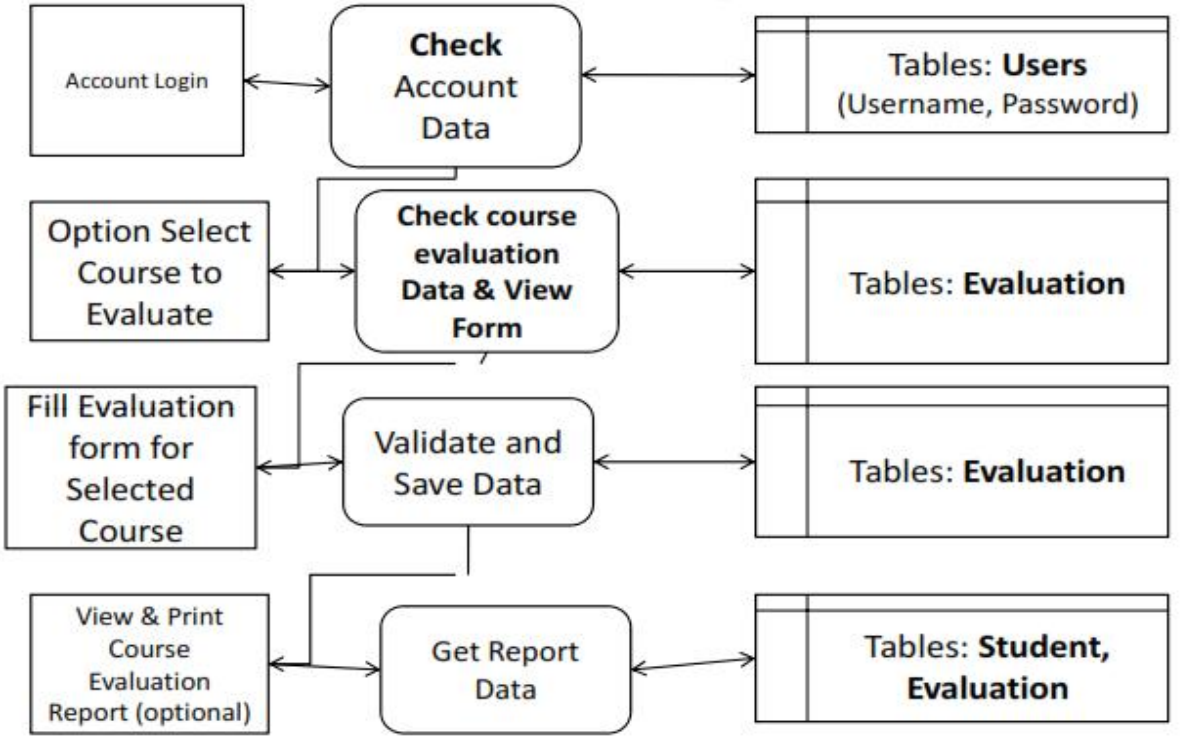
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



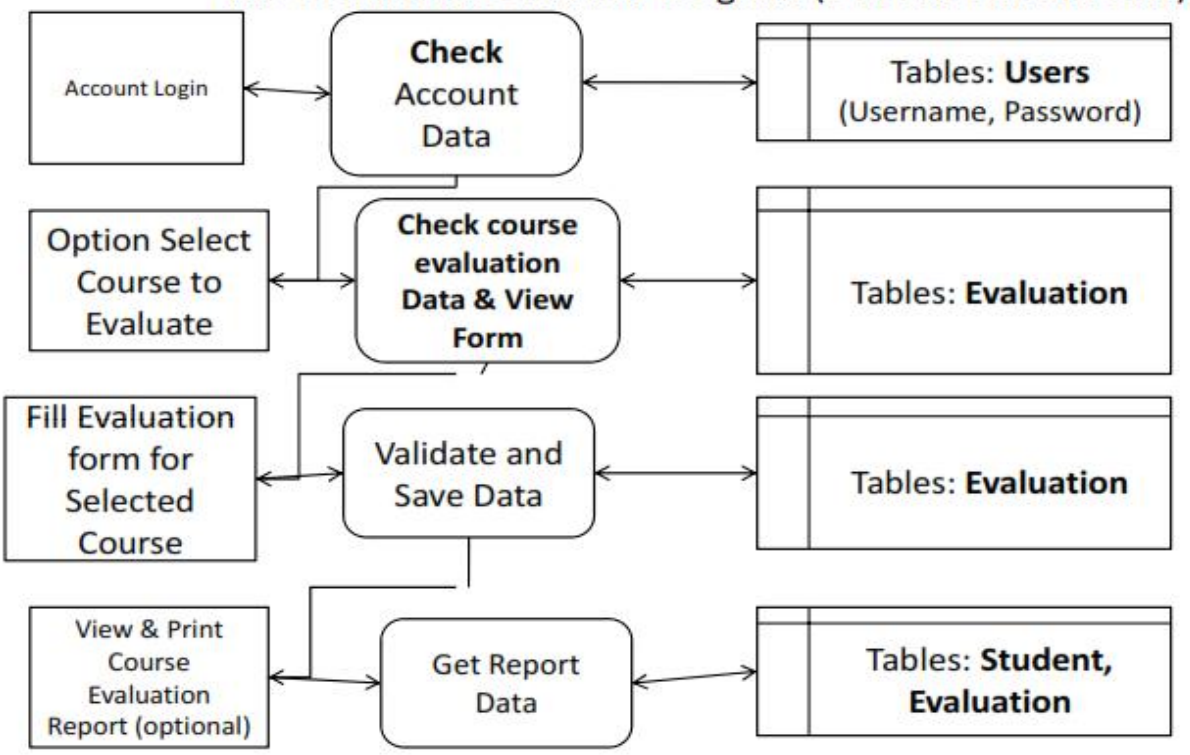
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



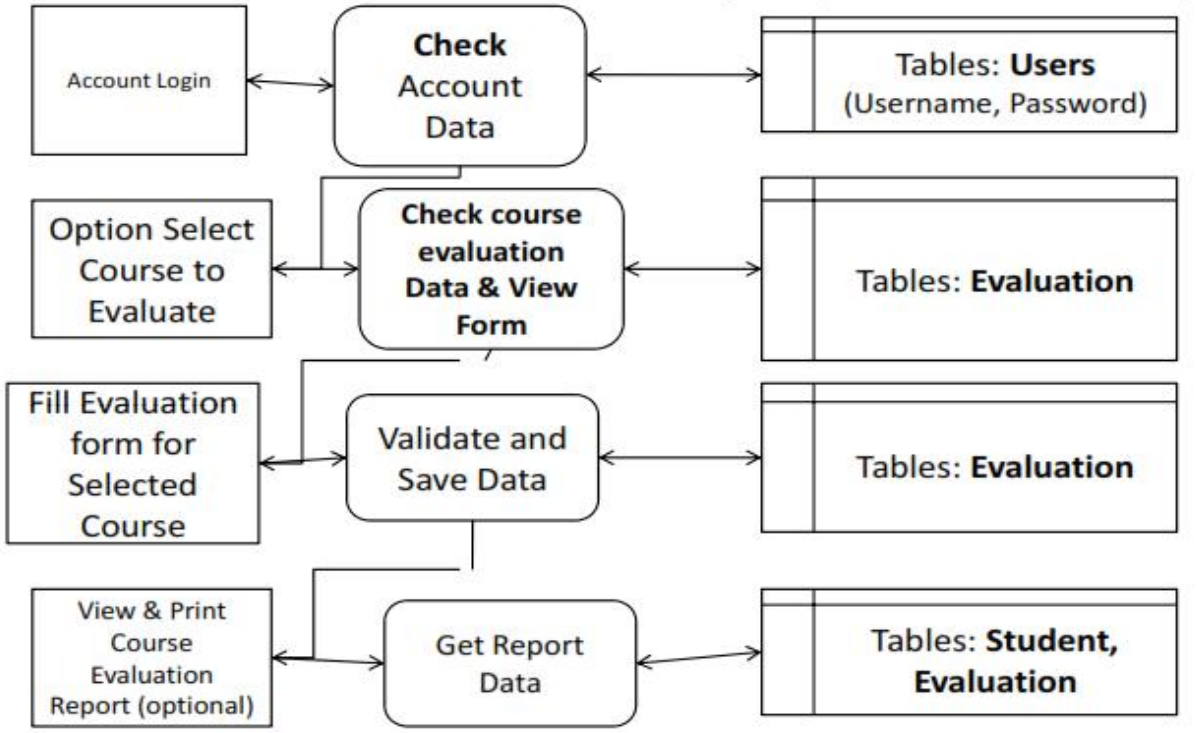
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



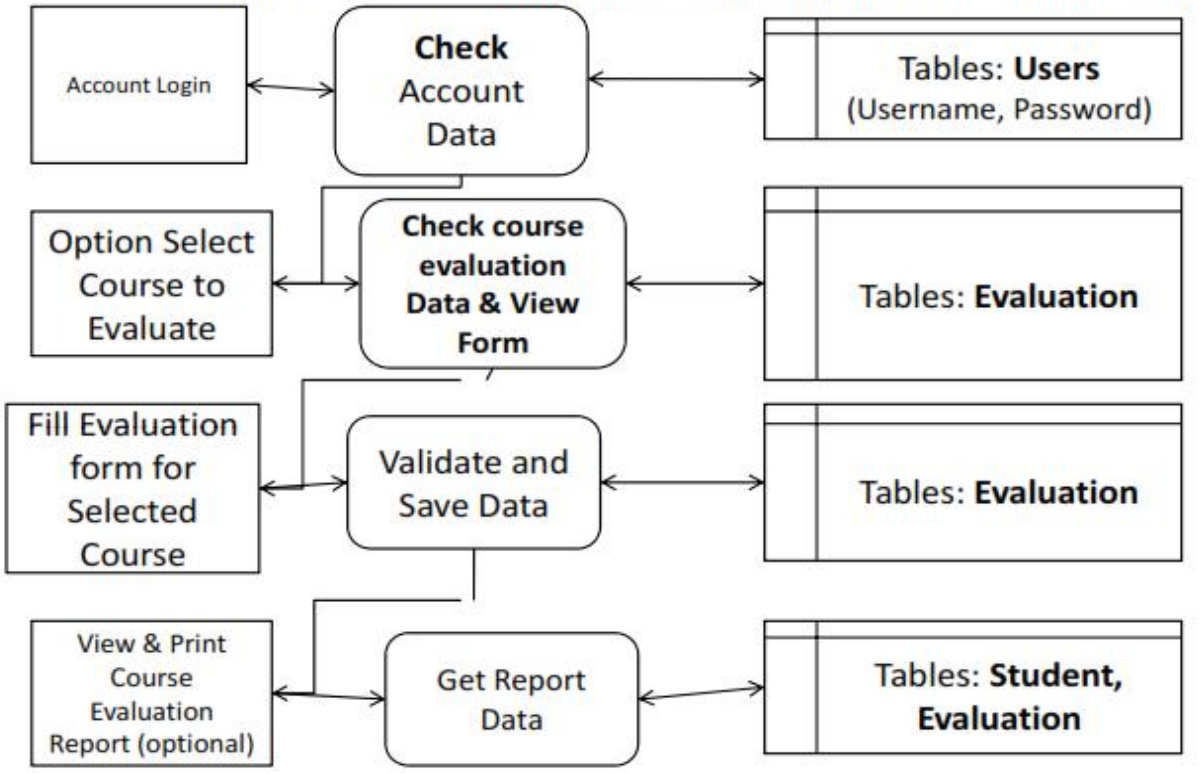
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



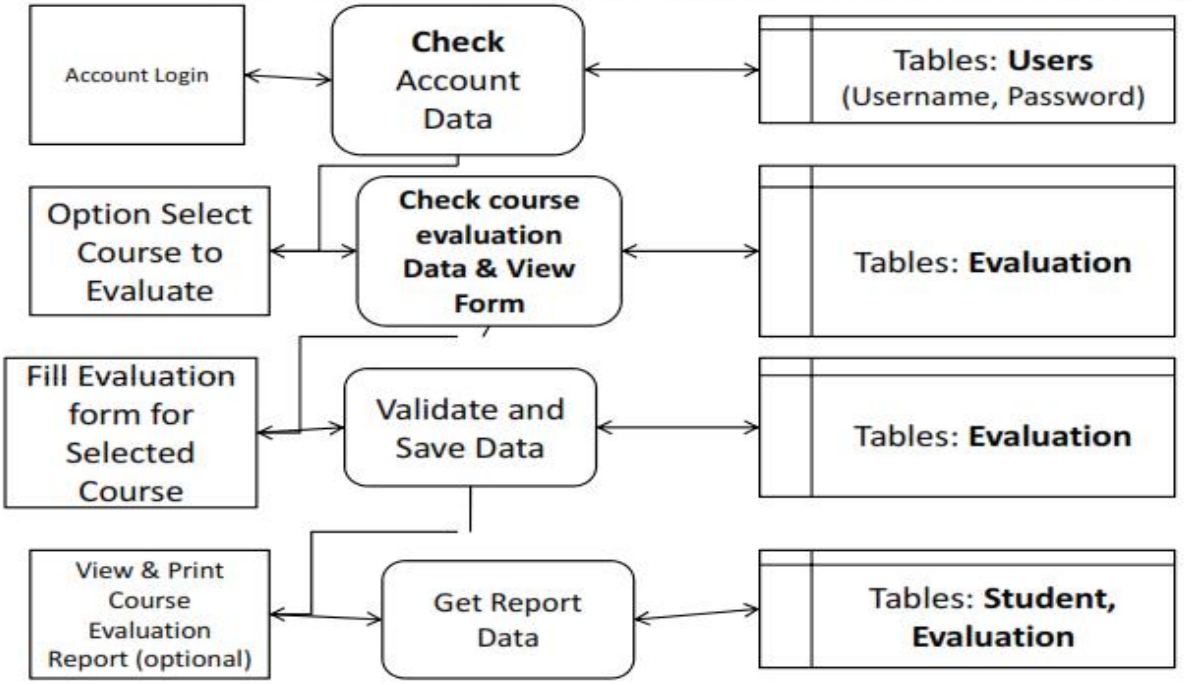
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



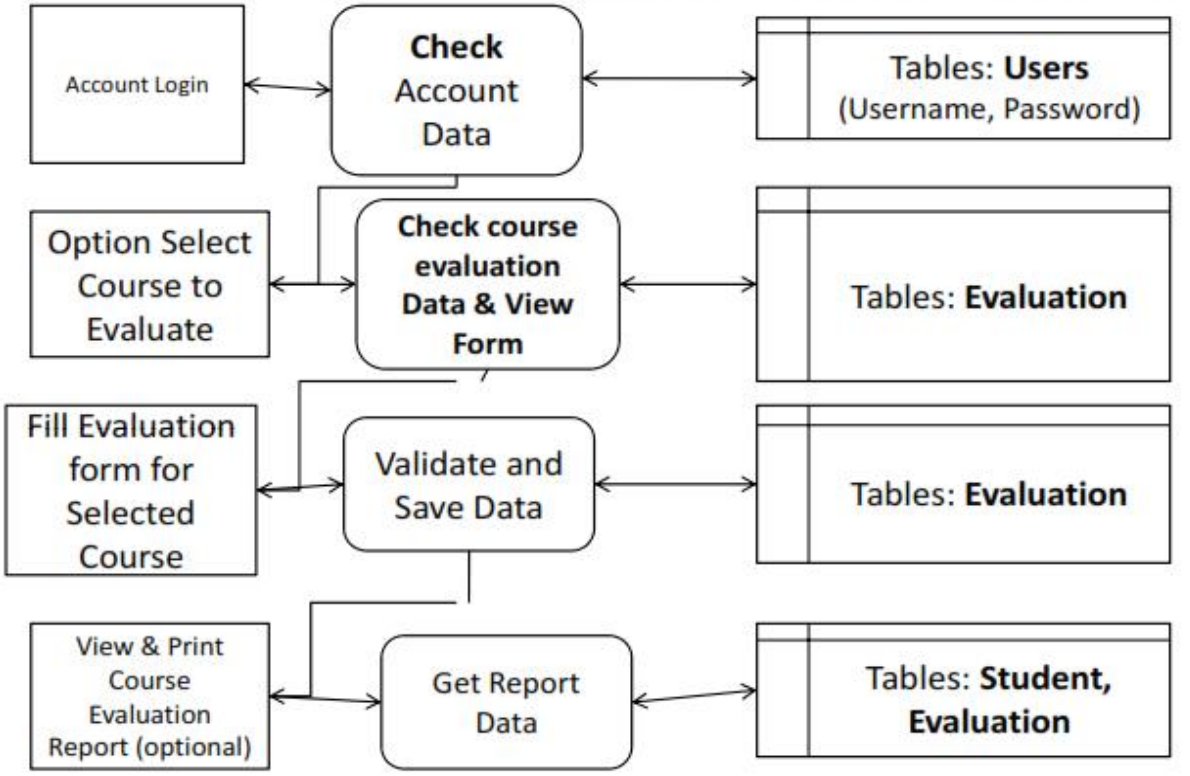
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



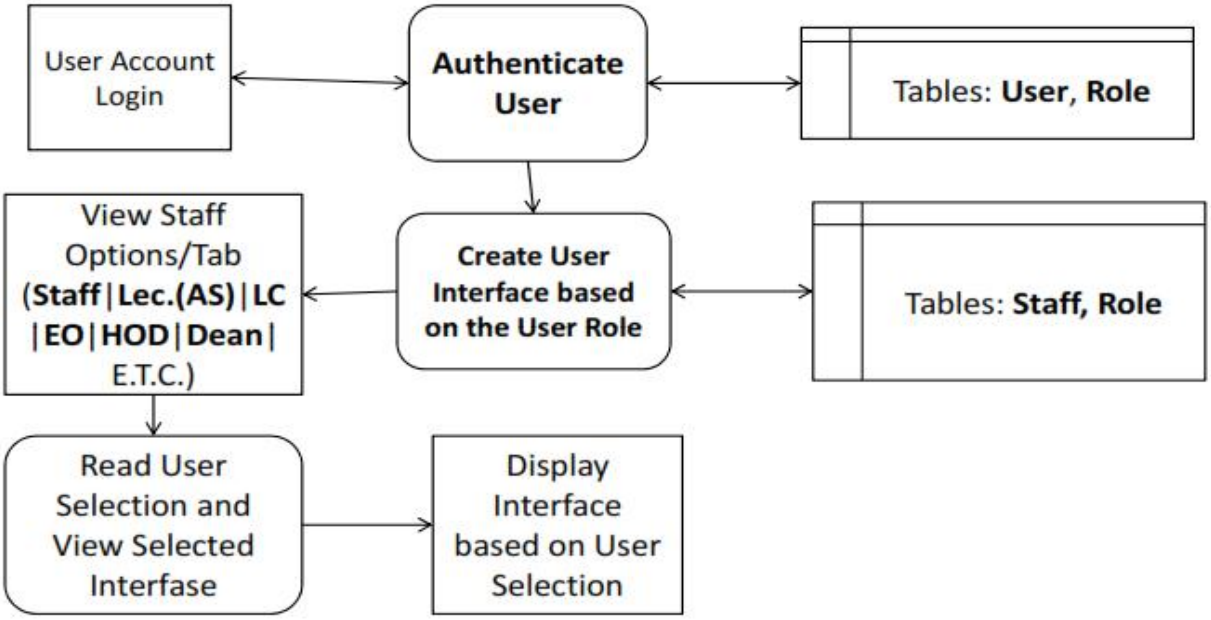
DFD – Detail Teacher Evaluation Form

The Detail is in Data Flow Diagram (Student's View Point)



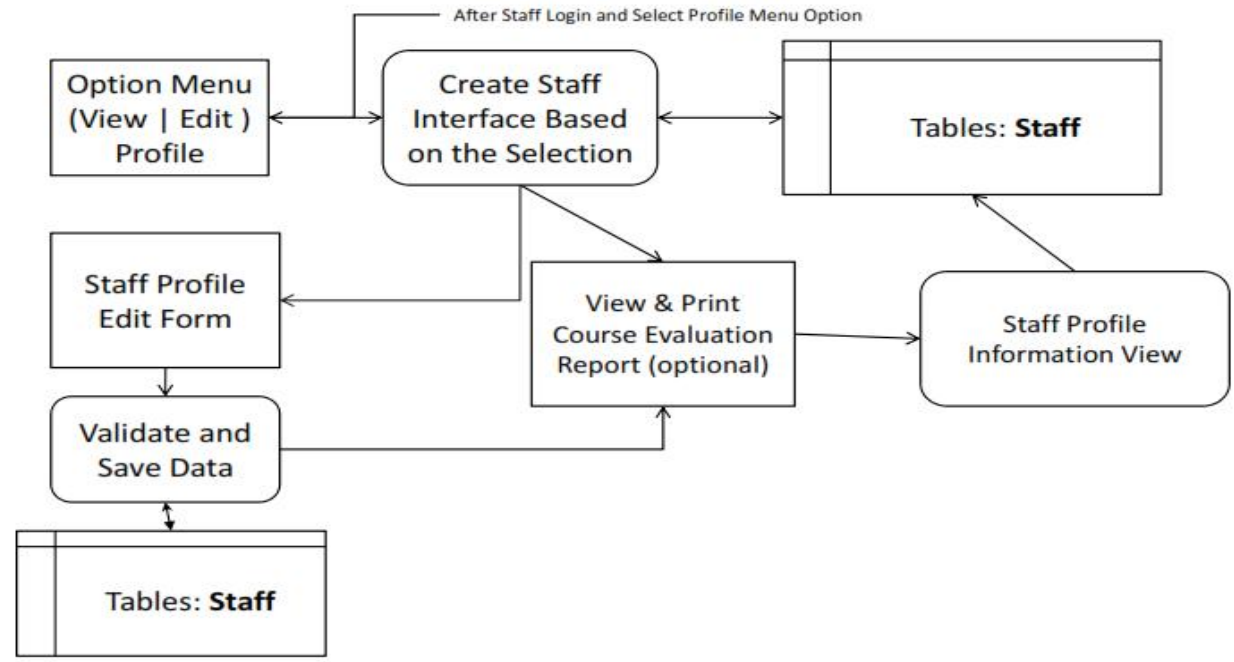
DFD – Staff

The Detail is in Data Flow Diagram (Staff Main Page Option View)



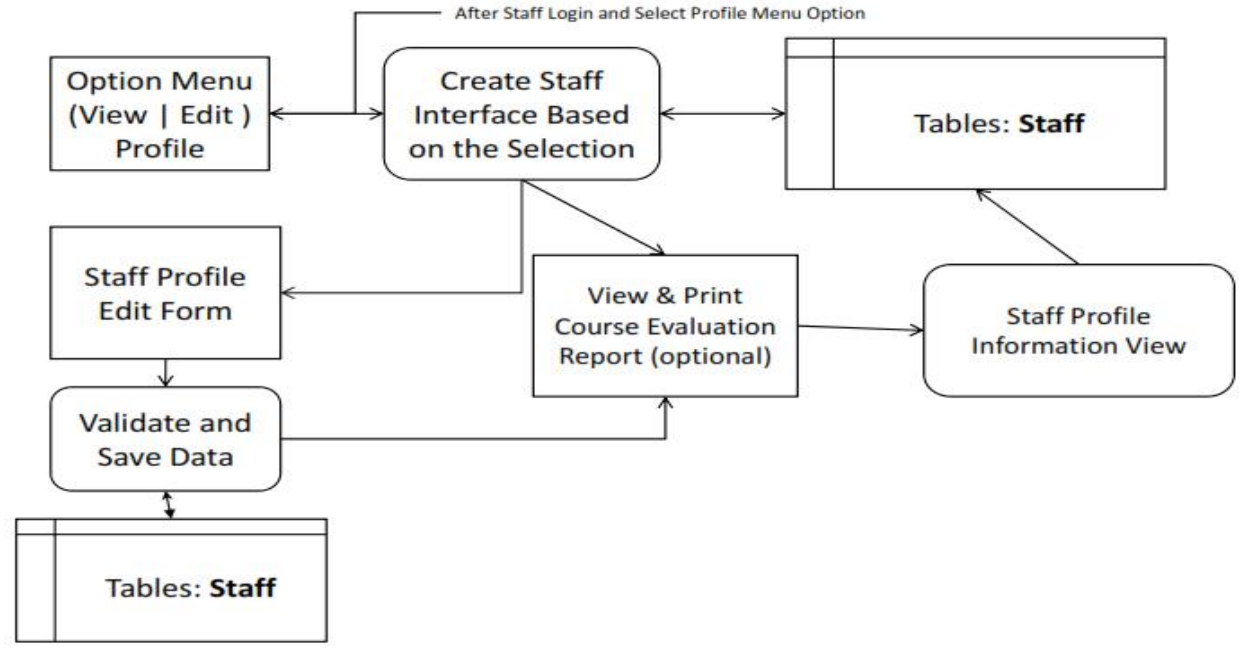
DFD – Staff

The Detail is in Data Flow Diagram (Staff – Profile Page View)



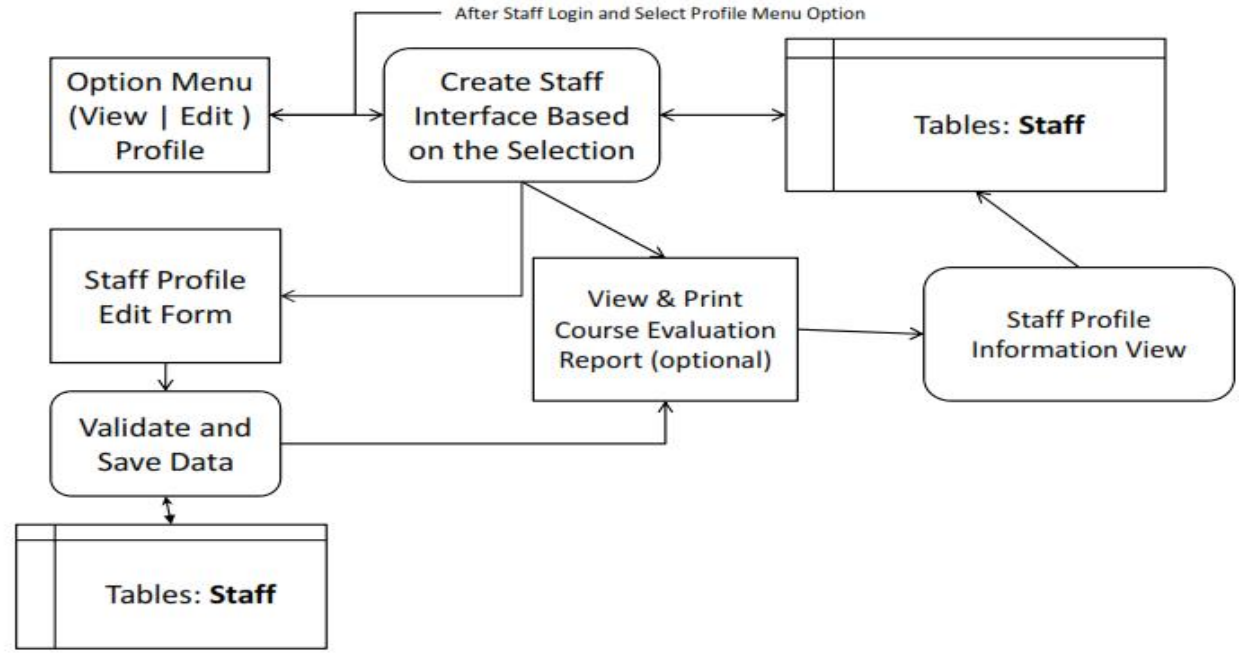
DFD – Staff

The Detail is in Data Flow Diagram (Staff – Profile Page View)



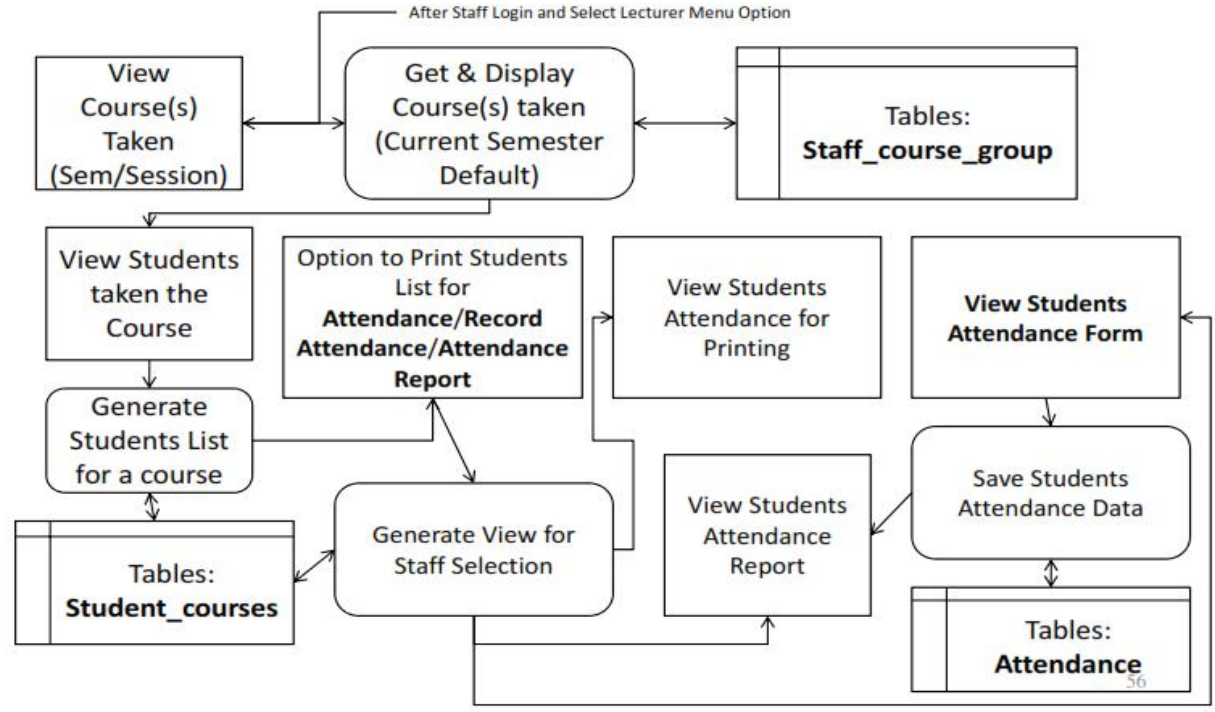
DFD – Staff

The Detail is in Data Flow Diagram (Staff – Profile Page View)



DFD – Staff

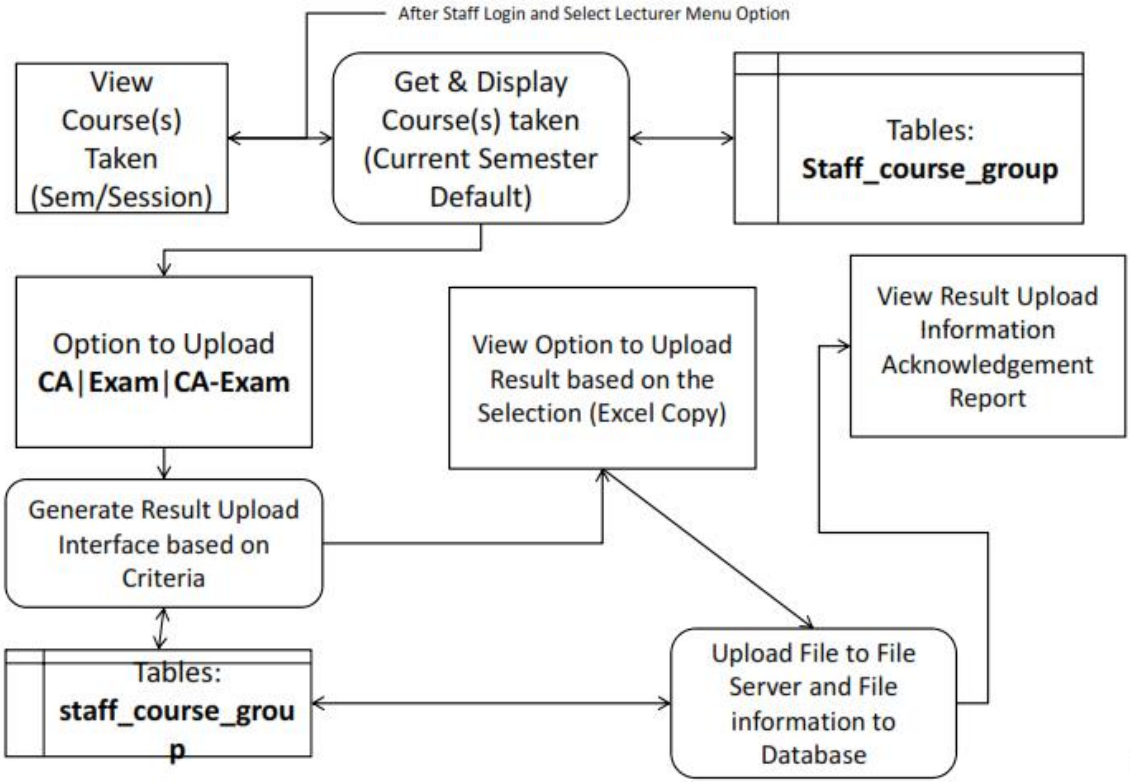
The Detail is in Data Flow Diagram (Academic Staff – Course – Page View) Back



DFD – Staff

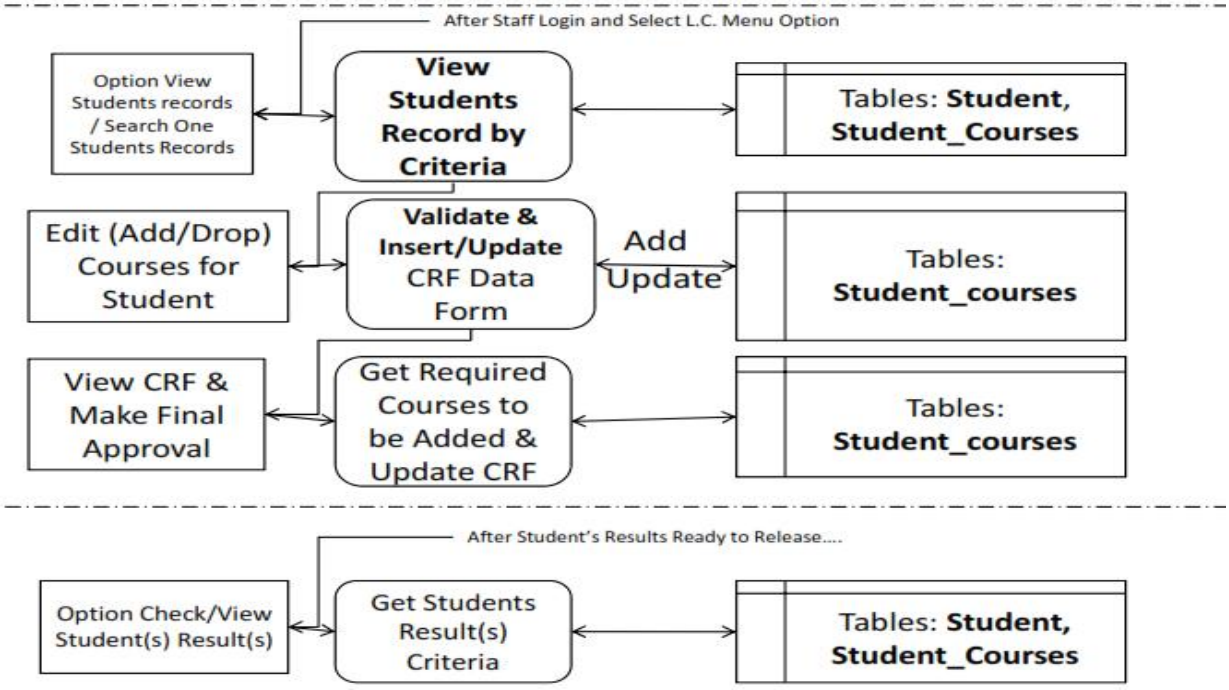
The Detail is in Data Flow Diagram (Academic Staff – Result Upload {Excel - File} View)

Back



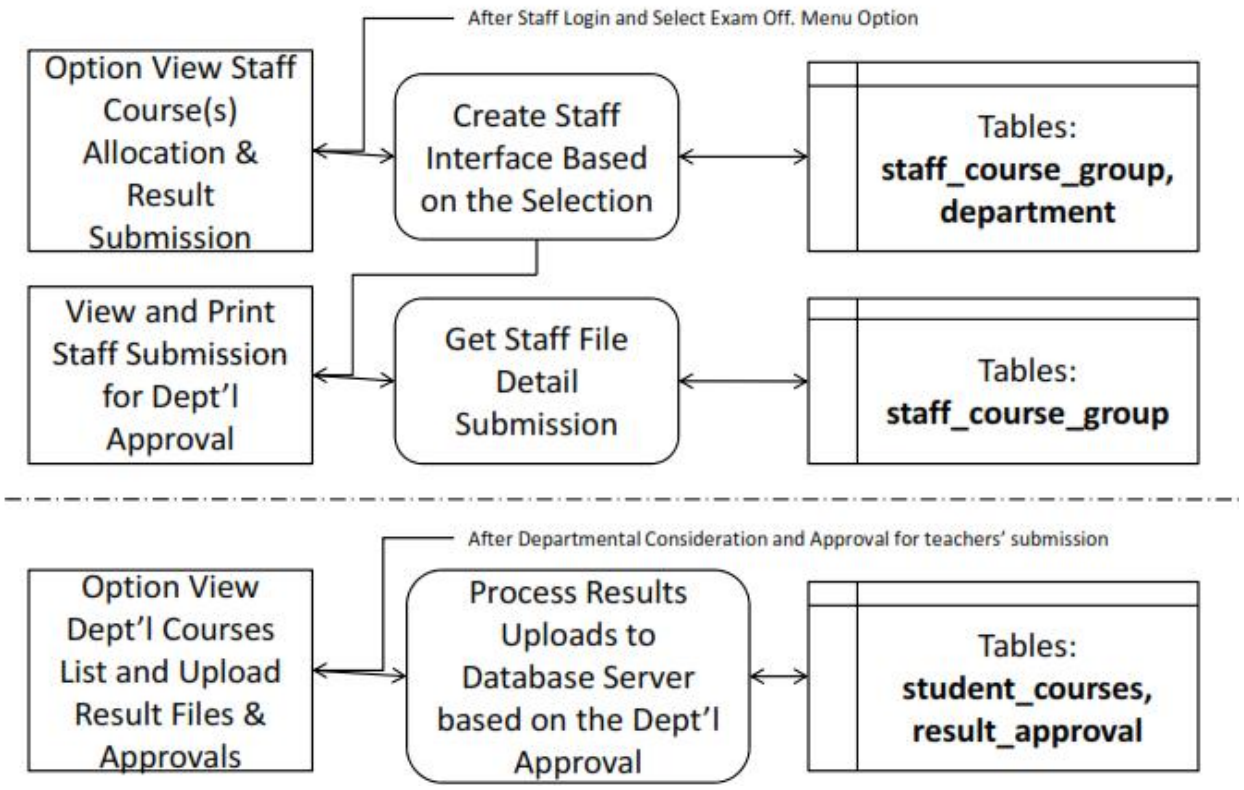
DFD – Staff

The Detail is in Data Flow Diagram (Dept. Level Cood. View Point)



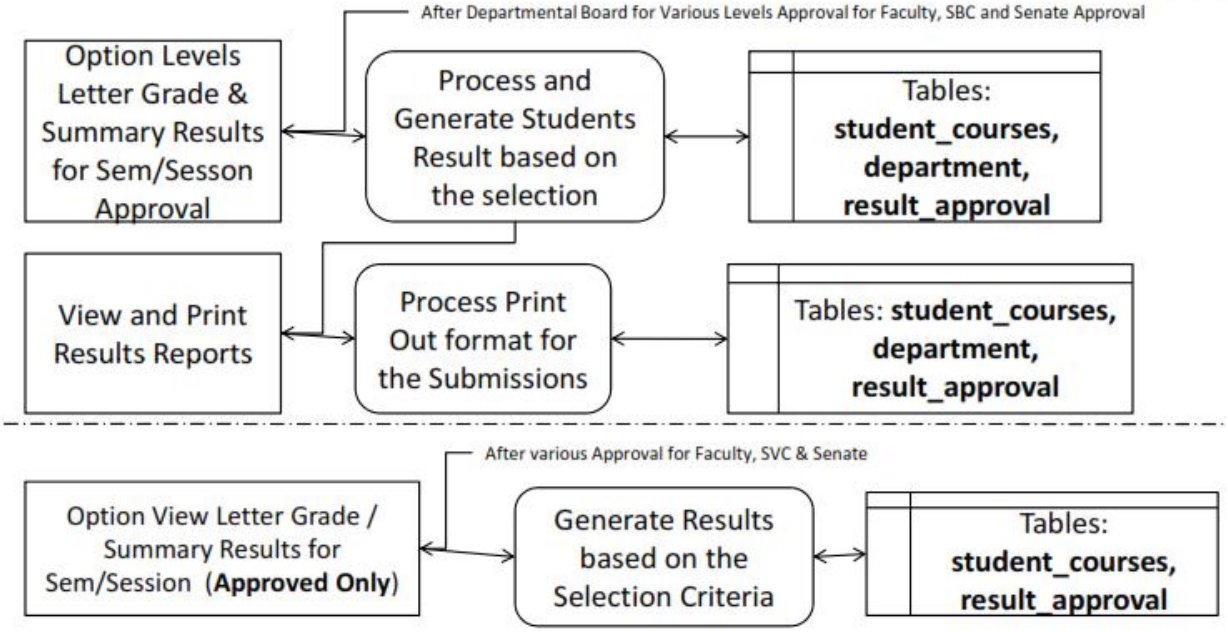
DFD – Staff

The Detail is in Data Flow Diagram (Examination Officer View Point)



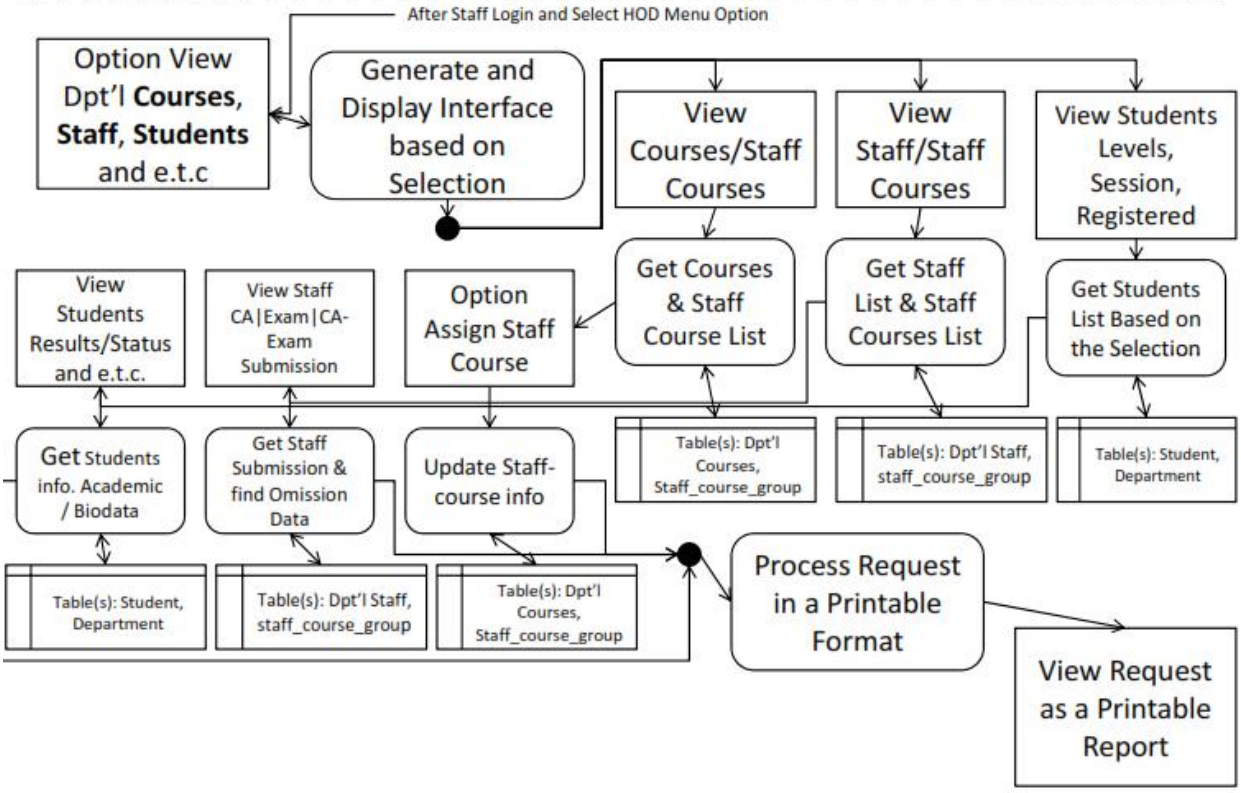
DFD – Staff

The Detail is in Data Flow Diagram (Examination Officer View Point) Back



DFD – Staff

The Detail is in Data Flow Diagram (Head of Department View Point) Back



Appendix B: WeBAIS E – R Diagram and Tables

Figure B.1: Database Tables and Diagram

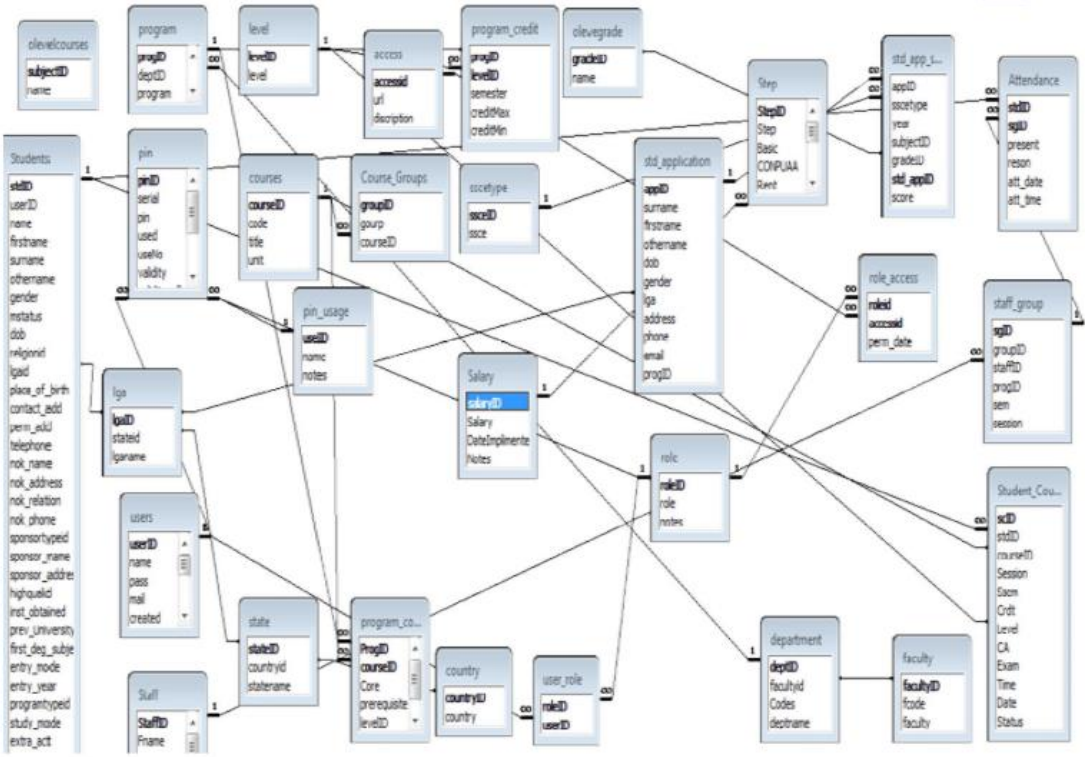
WeBAIS – Database Entity List

Back

1. appointment	21. penalty	41. staff_step
2. attendance	22. permissions	42. stafftype
3. bank	23. pin	43. state
4. bank_payments	24. pin_usage	44. std_app_subjects
5. country	25. program	45. std_app_subjectss
6. course_groups	26. program_copy	46. std_application
7. courses	27. program_courses	47. std_applications
8. department	28. program_credit	48. std_cgpa_results_view
9. eval_category	29. qual_grades	49. std_crs_view
10. eval_data	30. qualifications	50. std_evaluation
11. faculty	31. result_approved	51. std_gpa_results_view
12. grade	32. role	52. std_info
13. level	33. role_menu	53. std_registration
14. level_coordinator	34. salary	54. std_results_view
15. lga	35. school	55. step
16. menu	36. ssce	56. student_courses
17. nprogram	37. sscetype	57. student_courses-
18. olevegrade	38. staff	58. students
19. olevelcourses	39. staff_courses	59. user_role
20. pages	40. staff_courses_files	60. users

WeBAIS – Database Structure

Back



Appendix C: WeBAIS Questionnaire

Below is the draft of the WeBAIS Questionnaire

…after my address

Letter of Introduction

I am presently undergoing a research on "Web – Based Academic Information Source for Nigerian Universities" towards a Master of Science degree at Ahmadu Bello University, Zaria. Please spare some moments of your time to complete the research questionnaire attached.

The data obtained from this questionnaire would be used exclusively for the purpose of this research and all your comments would be kept confidential.

Thank you

Muktar Muhammad Aliyu

M.Sc/SCIE/07688/2009-2010

1. **Occupation**

Section 1: (General)

a. ☐ Management (Dean, HOD)

b. ☐ Exam Officer, Level Coordinator

c. ☐ Staff (Teaching, Non – Teaching)

d. ☐ Student (Applicant, Undergraduate, Graduate)

e. ☐ Other Specify: _____

2. Have you used a system (portal) similar to WeBAIS before?
    a. Yes
    b. No
3. WeBAIS reduces paper usage by about
    a. 100%
    b. 50%
    c. 25%
    d. 10%
    e. Others, specify:_____

4. Use of WeBAIS improves university daily operation

a. Strongly agree
b. Agree
c. Not sure
d. Disagree
e. Strongly disagree

5. Use of WeBAIS improves the teaching/learning process

a. Strongly agree
b. Agree
c. Not sure
d. Disagree
e. Strongly disagree

6. Use of WeBAIS improves the learning process

a. Strongly agree
b. Agree
c. Not sure
d. Disagree
e. Strongly disagree

7. WeBAIS response is fast and error free

a. Strongly agree
b. Agree
c. Not sure
d. Disagree
e. Strongly disagree

8. **WeBAIS provide security and is robust of service in University**
   a. Strongly agree
   b. Agree
   c. Not sure
   d. Disagree
   e. Strongly disagree
9. Application like WeBAIS may be recommended to all Nigerian universities
   a. Strongly agree
   b. Agree
   c. Not sure
   d. Disagree
   e. Strongly disagree

Administrators (Dean, HoD), pleasego to Section 4. Staff (teaching and non-teaching), please go to Section 2. Exam Officers/Level Coordinators, please go to Section 3. Students/Applicants/Graduates please go to Section 5.

Section 2: (Staff Teaching/Non – Teaching)

10. As instructor, student ratings using applications like WeBAIS, motivates me to improve teaching
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
11. With WeBAIS, I manage students' attendance taking more efficiently.
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree

12. With WeBAIS, it is easier and quicker for me to communicate with my students (publishing their CA and exam scores).
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
13. With WeBAIS, it is easier for me to check validity of my students' results published by the Exam Office.
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree

Section 3 (Exam Officer, Level Coordinator)

14. As Exam Officer, WeBAIS helps me to do results data entry quicker and easier
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
15. Result collection, processing and result release is easier and faster using WeBAIS compared with previous method
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
16. As Exam Officer/Level Coordinator, my work load has been reduced with WeBAIS
    a. Strongly agree
    b. Agree
    c. Not sure

d. Disagree
e. Strongly disagree

17. Managing Add/Drop using WeBAISis easier and faster
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree

Section 4: (Management {H.O.D, Dean})

18. As HOD/Dean, managing records of my Department/Faculty becomes easier
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
19. WeBAIS helps me identify and solve students' problems (course registration, etc) easier than before
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
20. With WeBAIS, we can generate various types of reports needed for Management decision
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
21. WeBAIS makes it easy to track students on probation, students who deferred their studies, etc
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree

Section 5: (Student, Applicant and Graduate)

22. As a student, rating instructors using applications like WeBAIS, motivates me to participate more in the teaching evaluation process
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
23. WeBAIS gives me quick access to my registrations/academic records
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
24. Lecturers are able to publish our CA/Exam scores quicker using WeBAIS
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree
25. Administrators are able to respond to students' problems quicker using WeBAIS
    a. Strongly agree
    b. Agree
    c. Not sure
    d. Disagree
    e. Strongly disagree