A WEB-BASED NOTICEBOARD SYSTEM

CASE STUDY: KYAMBOGO UNIVERSITY

BY

BASOOMA JOHN
(08/U/277/ITD/GV)

SUPERVISOR: Mr. MATOVU MOSES

A RESEARCH PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF A BACHELORS DEGREE OF INFORMATION TECHNOLOGY AND COMPUTING OF KYAMBOGO UNIVERSITY.

JUNE, 2011

DECLARATION

I **Basooma John** hereby declare that the work presented in this project report is the original work of my own hands and brain. It is not a photocopy, neither is it an edited version of the work presented earlier anywhere by other people.

Signature... Date......................................

APPROVAL

This report has been submitted with my approval as the supervisor.

Signature.. Date.....................................
Mr. Matovu Moses
Department Of Computer Science
Kyambogo University

DEDICATION

This work is dedicated to the beloved family of Mr. Basooma George.

ACKNOWLEDGEMENT

Above all, I acknowledge the Almighty God, who has taken me all through my studies right from the very beginning up to where I am now.

Secondly, I take the opportunity to acknowledge My parents Mr. Basooma George and Mrs Basooma Aidah, My brothers Mr. Mwidu Alex,and Mr. Muwanika Richard and their families, My dearest sister ms Kisengesa Winfred Moreen, and the rest of my family for having been supportive to me in my endeavours. May God reward them abundantly.

I also want to express my sincere gratitude to my classmates and friends Okware Aldo, Bayega James, and Muhuruzi Dorothy for being there for me during the hard times through the course of my study inside the university. May God bless them.

I also deliver my sincere appreciation to my lecturer and supervisor Mr. Matovu Moses for the knowledge and love he has extended to the entire BITC class throughout the course of study in Kyambogo University. May God reward and protect him.

I also deliver my sincere appreciation to my lecturer Mr. Matovu Henry for the very good computer programming skills he delivered to me during my three year study inside Kyambogo University. my God reward him abundantly

Lastly but not least, I acknowledge the entire teaching staff of the Department Of Computer Science of Kyambogo University for the knowkedge and skills delivered to us at the University

Contents

List of Figures

List of Tables

LIST OF ACRONYMS

DBMS	Database Management System.
FK	Foreign Key
HTML	Hypertext Mark up Language
I.T	Information Technology.
ICT	Information and Communication Technology.
ITEK	Institute of Teacher Education Kyambogo.
KYU	Kyambogo University.
PK	Primary Key
SMS	Serial Messaging Service
SQL	Structured Query Language.
UNISE	Uganda National Institute of Special Education.
UPK	Uganda Polytechnic Kyambogo.
URI	Universal Resource Indicator
W3C	World Wide Web Consortium
WCMS	Web Content Management System.
WWW	World Wide Web.
XHTML	Extended Hypertext Mark up Language

ABSTRACT

This report is a summary of the results of the study carried out on the noticeboard system of Kyambogo University. The research was aimed at analysing the requirements of the noticeboard system of Kyambogo University, and develop a web-based noticeboard system to serve those requirements. The requirements analysis was carried out on the basis of interviews, questionnaires, observation, participation and documentary review.

After system analysis, the system was modelled using data flow diagrams and entity relationship diagrams.

The system was later implemented with the capability to capture and store information, retrieve it on demand, and display it to the system users in a format that is easy to understand. The capability to capture and process student results was also incorporated into the system.

The system was implemented Using MySQL relational database management system, PHP server-side scripting languages alongside other web technologies.

Chapter 1

INTRODUCTION

1.1 Brief Introduction

This chapter gives a brief background of the case study for which the project was designed, and states the problem that was solved by the project. The significance and scope of the project are also stated.

1.2 Background

Noticeboards are one of the most heavily used means of distributing information both internally and externally in organisations, hospitals, schools, institutions and so many other societies. Noticeboards have thus become a very significant part of institutions as far as communication is concerned.

Finished documents are printed and pinned on wall noticeboards for people to read and get the information. This is a very effective way of distributing information as long as the target audience keeps visiting the noticeboard [14].

Kyambogo University (KYU) is Uganda's third public university established by the Universities and Other Tertiary Institutions Act 2001 . It is a merger of the former Uganda Polytechnic Kyambogo (UPK), the Institute of Teacher Education, Kyambogo (ITEK), and the Uganda National Institute of Special Education (UNISE) [15].

Kyambogo University is located 8km from Kampala City Centre along the Kampala Jinja highway. It is accessible through the Banda Trading Centre or the Kyambogo T-junction. There is also an access route through the Ntinda- Kiwatule road [15].

To keep pace with global trends, Kyambogo University has built capacity, and continues to do so in Information and Communications Technology (ICT). The objective is to enable all students to have access to computer training. All its academic programmes have an in-built component of ICT to enable the students appreciate the value of technology and to generate interest for further learning and use of ICT [15].

Despite its efforts to meet the global standards and find a better place on the international scene, Kyambogo University has not fully utilized the power of computers and the World Wide Web to facilitate management and distribution of information within the university and its branches. The university entirely relies on manual noticeboards to pass vital information to its entire community. Information pinned on these noticeboards inside the university is always not accessible to the students outside the university such as students in affiliated teacher's colleges and long distance students. Besides the information pinned on the manual noticeboards is also hard to keep track of for future reference [15].

The main purpose for carrying out this project was to develop a user friendly computerised web-based noticeboard system that would capture information from users, keep track of that information, and retrieve it for display on demand.

1.3 Problem Statement

Kyambogo University entirely relies on manual noticeboards to pass information to its students. printed copies are pinned on the manual noticeboards in the various departments of the university for the students to read.

The students find it sometimes difficult to access this information especially when they are out of the University.

The inconveniences caused by the manual noticeboard to the students can be solved by developing a computerised web-based noticeboard system.

1.4 Objectives

There were several objectives for carrying out this project. These were grouped into general and specific objectives.

1.4.1 General Objective

The main objective of this project was to develop a computerised noticeboard system to enable students to easily access information posted on the noticeboard at their convenience.

1.4.2 Specific Objective

The specific objectives of this research included the following

i. To identify the system requirements by studying the current noticeboard system of Kyambogo University.

ii. To design the web-based noticeboard system.

iii. To implement the system.

iv. To test and validate the system.

1.5 Scope of the Study

The geographical scope of the system will be limited to the faculty of science of Kyambogo University, but can be later extended to serve the entire University.

The system will be limited to processing and publishing students results, keeping track of students' information, and processing and publishing general notices.

1.6 Significance of the Study

The significance of this research is to improve on the ease and convenience with which information is distributed and accessed at Kyambogo University.

It will save the university top administrators the time spent in duplicating and pinning the same documents on different noticeboards of the University, and increase the rate at which information reaches the target audience .

The system will also improve the convenience with which lecturers send lecture notes, and results to the students.

The system will also enable students to easily access information in the University, receive lecture notes and coursework results, and form on-line discussion groups.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

This chapter gives a brief overview of a series of books, documents, articles and journals that the researcher reviewed to clear the air on the need for a web-based noticeboard system at Kyambogo University. The chapter particularly gives more concentration to database driven web-based information systems and web applications rather than stand-alone computer applications.

2.2 Information

Information can be defined as processed data. Information is data that has been made useful for problem solving and decision making [3].

2.3 System

A system is a collection of components that performs related activities to attain a common objective. A system must be robust in order to prevent data loss or corruption in case of any failures.[3]

2.4 Information system

An information system is a collection of interrelated components that capture data, and process it to output and store information that can be used for critical decision making in an organisation or institution such as a university. An information system can be manual, semi computerized or computerized [3].

2.4.1 Components of an information system

An information system consists of the following components [3]:

i. **Technology.** This involves the input, process, storage and telecommunication equipment of the system; That is hardware and software.

ii. **Data**. These are the raw input to the system.

iii. **Procedures.** These are guidelines and rules for an information system that govern data processing.

iv. **People**. These are the users who operate the system.

v. **Information.** This is that data which is refined and organized by processing and purposeful intelligence .

2.5 The World Wide Web (WWW)

The World Wide Web (WWW) is a collection of documents interconnected by hyper links that are distributed on web servers around the world . The documents on the web range from simple personal web pages to complex database driven, mission critical web applications [2].

To facilitate the growth and development of the World Wide Web, a full body of professional web designers and developers called the World Wide Web Consortium (W3C) was put in place to set the standards for web technologies. This body sets the syntax and semantics for mark up languages and web scripting languages [5].

The WWW is a technologically active environment, and information systems developed for the WWW differ in significant areas from traditional applications. Existing literature has suggested that software development methodologies for traditional applications may have to be modified, or indeed replaced to meet the needs of the WWW. Consequently, a number of software development methodologies have emerged to support the development of WWW-based information systems [1]. Currently, the standard mark up language for developing web-based applications is the Extended Hypertext Mark up Language (XHTML) which evolved from Hypertext Mark up Language (HTML) [4].

To aid in developing database driven web-based applications, a variety of server-side scripting language have been introduced to manage the exchange of information between web servers and database servers. these include PHP: Hypertext preprocessor, Perl, Rubby on rails, Active Server Pages.Net (ASP.Net), Java server pages(JSP), Cold fusion, among others. These have been instrumental in the development of the World Wide Web [6].

Web technology differs from the traditional information technology in that "IT might be labeled as a new type of information system, but it is fundamentally a new medium of human communication" [10].

The WWW incorporates the idea of a boundless information world in which all the items have a reference by which they can be retrieved; That is a Universal Resource Indicator (URI) [9].

The late 1990s has seen the WWW emerge as a strategic business tool, driven by the commercial interests of companies [7].

2.5.1 Web Based Information Systems

The applications that have emerged on Intranet and Extranets using WWW technologies, can be referred to as WWW-based information systems. However, a WWW-based information system should be distinguished from a standard WWW application or page; the nature and the type of information that is made available to the user is different. The standard WWW page is unidirectional in the way it provides information to the user; with the user often making requests in catalogue or directory based sites by following hypertext links. A WWW-based information system can be defined as an application that not only disseminates information, but also pro actively interacts with the user to aid them in their task. Information is therefore presented to the user in a bi-directional manner in a WWW-based information system [11].

The emergence of database centric applications on the WWW has led to a change in the nature of WWW pages. This emerging change is supported by the observations of Tomas and Michael [6], as they categorise WWW pages as being either static or dynamic. Static hypertext systems, as commonly provided by standard WWW pages, can be characterised by both static links and static pages. Database-centric applications are characterised by dynamic page creation but have a static link structure. Dynamic applications, however, provide both dynamic link structure and page creation. WWW-based information systems fall into the database-centric and dynamic application classes [6].

WWW-based information systems are fundamentally different from traditional systems in several critical areas. Tomas and Michael suggest that WWW applications often involve people with differing skill sets, including authors, content designers, artists as well as programmers. WWW applications also involve the capturing and organising of the structure of a complex information domain, whilst making it clear and accessible to the user [6].

2.6 Existing Web Based Information Systems

There are so many web based information systems that have been developed by people else where in the World. These include the following:

2.6.1 A Campus Web Based Information System (Karadeniz Technical University)

The campus web based information system for Karadeniz Technical University was developed by Arif Cadgas and Tahsin Yomraliogru in September 2002, to collect University's various data consisting of management components and topography, and Land use [12].

According to the developers, the system was developed with the main objective of facilitating the sharing and distribution of information in Karadeniz Technical University .

The system developed by Arif Cadgas and Tahsin Yomraliogru made use of a geographical information system program called arcIMS, and the internet to improve on the use of geographical information to facilitate managerial decision making in Karadeniz Technical University.

The system was also intended to keep track of changing topology and around Karadeniz Technical University, and provide accurate values that can be used to deduce a reliable trend of the topology changes around the University.

To attain these objectives, the campus information system of Karadeniz Technical University was supposed to collect graphic, and non graphic data relating to the University and its departments and sub-departments, and then transform this data into a computer platform, store, query, analyse, and present this data to the users in various formats including maps, charts, graphs, reports, and other web based views. This information was supposed to be availed to all the departments of the university whenever needed.

Basing on the observations of Arif Cadgas and Tahsin Yomraliogru, building the web based information system for Karadeniz Technical University came with quite a lot of benefits to the university including the following:

The graphic data relating to Karadeniz Technical University such as roads, landscape, parking area, infrastructure, vegetation, among others was collected, and made easily available to the users in form of maps [12].

The topological data relating to Karadeniz Technical University that was collected was digitised, and stored in a form that can easily be modified in case of any changes in the topology.

Generally, the system simplified the process of accessing topological information in Karadeniz Technical University.

Much as the objective of collecting ad distributing data to the project was attained, the project could still be extended to enable it to handle student and staff data in the University. This functionality was left out.

2.6.2 A Web Based Information System (UNIX Servers)

This project was carried out by Madhavan Raghavan, as a partial fulfilment for the award of a master of science in computer science of the University of Wales [13].

The project attempted to develop and implement a web based system to act as a single point of access for server information, especially UNIX based computer systems. The system was supposed to regularly collect server information from a given number of UNIX servers without manual intervention, Store this data in a central data store, and avail it to the users who need it in a suitable web view [13].

In large organisations where system performance is a critical factor in the operation of business, the point of knowing how many servers are running at a particular point in time is necessary since this information is useful for system optimisation and troubleshooting. This is supported by the observations of Madhavan in his research during the development of a single server access point [13].

In most cases, system administrators access this information by using manual methods such as referring to the system documentation of the system, or logging the server information into a log file whenever a server request is made [13].

According to Madhavan, the first method, has a disadvantage of requiring someone who constantly keeps updating the document, besides being an inconvenience to the users since the information is not easily available [13].

The second method on the other hand, though semi computerised leads to duplication of server data since server information is logged each time a request is made to the server. This is a common problem with semi computerised systems [13].

2.7 Manual Information Distribution Systems

This section discusses some of the manual and semi computerised systems used to distribute information.

2.7.1 Magazines

Magazines, periodicals, glossies or serials are publications, generally published on a regular schedule, containing a variety of articles. They are generally financed by advertising, by a purchase price, by pre-paid magazine subscriptions, or all three [14].

Magazines can be distributed through the mail; through sales by book stores or other vendors; or through free distribution at selected pick-up locations.Sales models for distribution fall into three main categories:

i. Paid circulation: In this model, the magazine is sold to readers for a price, either on a per-issue basis or by subscription, where an annual fee or monthly price is paid and issues are sent by post to readers [14].

ii. Free circulation: This means that there is no cover price and issues are given away, for example in street dispensers, airline in-flight magazines or included with other products or publications [14].

iii. Controlled circulation: This is the model used by ”insider magazines” or industry-based publications distributed only to qualifying readers, often for free and determined by some form of survey. This latter model was widely used before the rise of the World Wide Web and is still employed by some titles. For example, in the United Kingdom, a number of computer-industry magazines, including Computer Weekly and Computing, and in finance, Waters Magazine [14].

2.7.2 Bulletin Boards

A bulletin board (pin board or notice board in British English) is a surface intended for the posting of public messages, for example, to advertise things to buy or sell, announce events, or provide information. Bulletin boards are often made of a material such as cork to facilitate addition and removal of messages or it can be placed on the computer so people can leave and erase messages for other people to read and see [14].

Bulletin boards are particularly prevalent at universities. Many sport dozens, if not hundreds or thousands of public bulletin boards, used for everything from advertisements by extracurricular groups and local shops to official notices. Dormitory corridors, well-trafficked hallways, lobbies, and free standing kiosks often have cork boards attached to facilitate the posting of notices. At some universities, lampposts, bollards, trees, and walls often become posting sites in areas where official boards are sparse in number [14].

Internet forums are becoming a global replacement for traditional bulletins. Online bulletin boards are sometimes referred to as message boards. The terms bulletin board, message boards and even internet forums are interchangeable, although often one bulletin board or message board can contain a number of internet forums or discussion groups. An on-line board can serve the same purpose as a physical bulletin board. Magnet boards, or magnetic bulletin boards, are a popular substitute for cork boards because they lack the problem of board deterioration with the placement of pins over time [14].

2.8 Database

A Database is a structured collection of data that is centrally managed to meet the needs of a community of users.

The structure is achieved by organizing the data according to a database model. The model in most common use today is the relational model. Other models such as the hierarchical model and the network model use a more explicit representation of relationships [8].

The earliest known use of the term database was in November 1963, when the System Development Corporation sponsored a symposium under the title Development and Management of a Computer-centered Data Base. Database as a single word became common in Europe in the early 1970s and by the end of the decade it was being used in major American newspapers [8].

The first database management systems were developed in the 1960s. A pioneer in the field was Charles Bachman. Bachman's early papers show that his aim was to make more effective use of the new direct access storage devices becoming available: until then, data processing had been based on punched cards and magnetic tape, so that serial processing was the dominant activity [8].

A computer database relies upon software to organize the storage of data. This software is known as a database management system (DBMS). Databases management systems are categorized according to the database model that they support. The model tends to determine the query languages that are available to access the database. A great deal of the internal engineering of a DBMS, however, is independent of the data model, and is concerned with managing factors such as performance, concurrency, integrity, and recovery from hardware failures. In these areas there are large differences between products [8].

2.9 Summary

The Literature review indicated that web-applications are becoming a critical businesses tool.

The existing web-based information systems however are not customised enough to handle student data and noticeboard information effectively. For example, The web information system for Karadeniz Technical University can effectively handle topological data while the Web based UNIX server access system can effectively handle server data. On the other hand, the manual systems such as magazines have quite a lot of loopholes including difficulty of trucking data.

The Web based noticeboard system developed in this Project is designed to effectively capture, store, retrieve and distribute students information in the Kyambogo University. In the web-based noticeboard system, the researcher used MYSQL database management system to develop the database for the system because of it high security features and compatibility with other web technologies. A web-based user interface was developed for entering data into the system, retrieving, formatting , and displaying data to users in an easily comprehensible format. The researcher also used the latest versions of the incorporated technologies with backward compatibility to enable the system to interact with new and old technologies.

Chapter 3

METHODOLOGY

3.1 Introduction

This chapter describes the techniques and approaches that were be used to achieve the objectives of the study. It covers the requirements elicitation process, system analysis and design, implementation, testing and validation.

3.2 Identification of Requirements

To identify the requirements of the system, various approaches were used as explained below.

3.2.1 System Study and Investigation

There was a thorough study of the existing noticeboard system at Kyambogo University in order to understand its loop holes before developing the web-bases noticeboard system. This was achieved through interactions, use of questionnaires, interviews, observations and participation in the current system.

3.2.2 Documentary Review

Some of the information was collected by studying documents from within and out of the University. Information from literature reviews clearly brought out the need for a computerised system to serve as a noticeboard in the University and how time critical information circulation should be in such an environment like a University. However, most of the documents that had been pinned on the noticeboard some time back could not be retrieved since they had already been lost. This clearly indicated the need for a computerised system that can keep and retrieve such information on demand.

Documentary review also helped the researcher to perceive the views of people else where in the World towards web-based systems. The researcher's findings clearly indicated that there was a need for a web-based noticeboard system at Kyambogo University

3.2.3 Questionnaires

In this approach, questionnaires were specifically designed with relevant questions to study the current noticeboard system in Kyambogo University. The questionnaires were filled by members of Kyambogo University community mostly students. This approach was instrumental in finding out what view they had in relation to the manual noticeboard system at the University.

3.2.4 Interviews

Oral interviews were conducted by personal between the researcher and the noticeboard Users at Kyambogo University mostly students. This was instrumental in gathering information about the current manual noticeboard system and its loopholes since it gave the researcher a chance to interact with the system users . However, time and resources did not allow the researcher to exhaust the majority of the entire Kyambogo university community.

3.2.5 Observation

This method was used as an additional approach to clearly and fully understand the flow of processes in the existing system. It helped the researcher to determine requirements that could not easily be expressed by the interviewees due to lack of technical knowledge.

3.3 System Analysis and Design

During the analysis phase, system analysis and design was done using data processing models like data flow diagrams, entity relationship diagrams and context models. This was so as to clearly understand the flow of information between processes geared by the three fold design; the conceptual, logical and physical designs.

The conceptual design involved identifying the relationships between the major entities interacting with the system and their matching attributes and identifying the relationships between entities where entity relationship diagram was the end product.

The logical design involved identifying the various data structures that were used to hold entity data. These include database tables, files and many others.

The physical design involved designing the database tables and assigning them the necessary field sizes and types. The result accomplished the design of the system.

3.4 System Implementation

Like any other web-based information system the web-based noticeboard system was implemented using a relational database management system and web technologies. During the implementation of the system, I used MYSQL relational database management system to create the tables that were supposed to hold data for the system. MYSQL was chosen because of its very good security features, compatibility and easy integration with various web technologies on top of being open source.

To develop the web interface, a combination of web technologies were used including a mark up language, specifically XHTML(Extended Hypertext Mark up Language)was used to build the interface components, A server-side scripting language (PHP:Hypertext pre processor) was used to process data coming in and out of the system , A client-side scripting Language (Javascript) was used to perform client-side data validation, and cascade styling sheets (CSS) was used to improve the appearance of the interface components.

3.5 System Testing and Validation

This section briefly discusses the steps that were used to test and validate the system.

3.5.1 System Testing

The system was tested using unit testing in the preliminary stages of the testing phase. This involved testing individual components (methods and modules) in the code to judge the functionality of each in isolation using test data. This was so in order to ensure that each module that was integrated in the system was working as expected. A successful test here was integrated into a complete system that will also be tested as a whole (integration testing). Techniques like black box testing were also employed so as to achieve the system's final objective. These involved allowing other people who did not have knowledge of the internal working of the system especially users to test the system using there own data.

3.5.2 System Validation

Several validation tests such as the data and security validation tests were carried out to ensure that the system can validate data input from the users, reject any data which is supplied in a wrong format, and prevent unauthorised users from accessing the system resources. Subsections 5.4.1 and 5.4.2 describe the security and data validation tests that were carried out on the system.

Chapter 4

SYSTEM STUDY, ANALYSIS AND DESIGN

4.1 Introduction

This chapter discusses the design technicalities and procedures that were used to clearly study the existing system, which was t he drive to coming up with the new system. It involves all the structured diagrams used to model the existing and the newly designed system.

4.2 System Study

This section gives the various steps that were taken by the researcher during the study of the noticeboard system at kyambogo University.

4.2.1 Requirements Analysis

This process involved the discovery of the requirements of the new system through discussion with potential users of the existing system.

It involved the researcher working interacting with stakeholders who use the current noticeboard system. During this stage, the researcher was able to clearly identify the constraints of the web-based noticeboard system.

4.2.2 Operation

In the current noticeboard section of Kyambogo University, all information that is supposed to be viewed by the students is printed on papers, and pinned on manual noticeboards.

4.2.3 Weaknesses

- Information can not easily be stored for future reference because it is plucked off the manual noticeboard to create space for new posts.

- The manual noticeboard has a limited capacity of information that can be pinned there.
- information pinned on the manual noticeboard is not secure since it can easily be plucked off by anyone
- The system is not easily accessible since it is only situated in one place.
- It takes a very long time for students to access their results since they have to wait until they are pinned to the noticeboard.

4.3 System Analysis

This section discusses the procedures taken by the researcher to analyse the noticeboard system at Kyambogo University, and gather the requirements of the system.

4.3.1 Requirements Specification

This involved activities of translating the information collected during system analysis phase into a document that defined a set of requirements for the new system. There are two types of requirements which were looked at, namely; user requirements and the system requirements.

4.3.1.1 User Requirements

These were specifications that were required of the system by the users. These requirements were represented in statements in natural language, and also modelled into diagrams such as data-flow diagrams, entity relationship diagrams, for representation in a way that can easily be understood by the users.

The user requirements were divided into two categories: that is functional and non-functional requirements

4.3.1.2 Functional Requirements

These are the requirements that described what the system must do. They clearly state all the functionality that is required of the system. The functional requirements included the following:

i. The system shall accept input of notice data from users from users.

ii. Should Store the in put information into one data storage are specified on the active database server.

iii. The system shall be able to easily retrieve the stored notice data on demand

iv. The system shall allow users to input student assignment and examination results, and compute the final mark, grades and student progress.

v. The system shall allow students to form on-line discussion groups, and conduct discussions on line.

vi. The system shall allow students to access their results on line.

vii. The system shall not allow students to be able to edit their results while they retrieve them on line.

viii. All students and staff should have user accounts that they use to login to the system.

4.3.1.3 Non functional Requirements

Non-functional requirements were described as the features on the services the system was to provide. These included the following:

i. System users shall be authenticated while they access the system resources in order to limit different users to only data they are allowed to access.

ii. The system shall be able to handle concurrent access without any failure or delays

iii. The system shall be able to return informative messages to users when they enter incorrectly formatted data into the system.

iv. The system shall always give a brief description to the users about its usage at each section.

v. The system shall not always exhibit any delays during data retrieval by the staff or students.

vi. The system shall not fail frequently.

vii. The system shall be able to recover from failures in case of any quickly and without any data loss.

4.3.2 System Requirements

The system requirements are categorised into hardware and software requirements as described below.

4.3.2.1 System Hardware Requirements

These constitute of the minimum hardware that on which the system must be installed. Table 4.1 shows the hardware requirements for the web based noticeboard system.

Hardware	System Requirement
Central processing Unit (CPU)	Intel Pentium duo core or higher version
Physical Memory	2GB Of Random Access Memory (RAM) or more
Hard disk space	100 GB or more

Table 4.1: Hardware Requirements

4.3.2.2 System Software Requirements

These include the minimum software platform on which the system must be installed. Table 4.2 below shows the minimum software requirements of the web based noticeboard system.

Operating System	Windows Server 2003, Linux Or higher Operating systems
Webserver	Apache 2.2.17 or higher version
Database Server	MYSQL 5.1 or hiher version
Web client	Mozira firefox, Microsoft Internet explorer or other web browsers

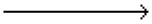
Table 4.2: Software Requirements

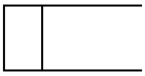
4.4 System Design

This section explains the various methods that were used to model both the current and the proposed system. It includes the use of context diagram, process and data flow diagrams accompanied with entity relationship modelling for the database design.

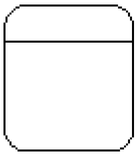
4.4.1 Description of symbols

The figures below show the symbols used in the design of the data flow diagrams. They are not included in the list of figures.


Data flow


Data store

Entity


Process

4.4.2 Context Diagram for the Web Based Noticeboard System

The methods discussed above were also used to model the expected system in order to understand the expected system's operational context and anticipated flow of processes.

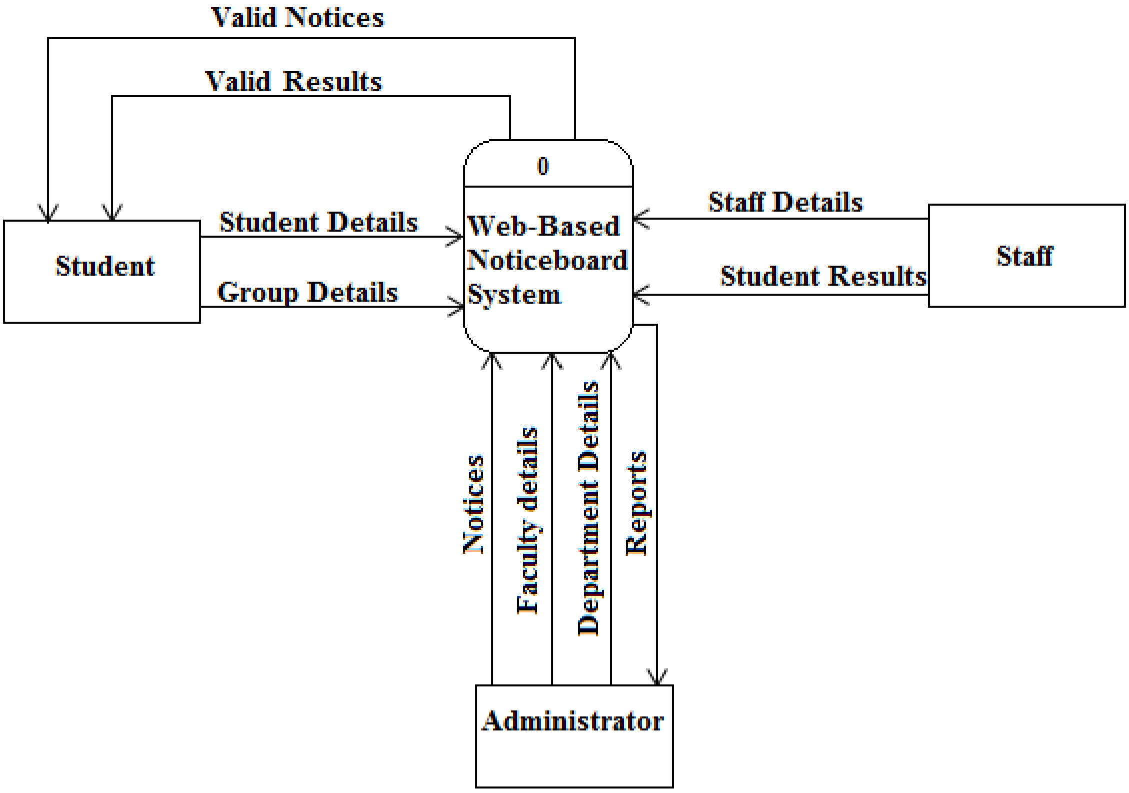


Figure 4.1: Context diagram for the Web-based Noticeboard System

4.4.3 Level One Data Flow Diagram for the System

The context digram above was later split into various processes and stores, to further understand the flow of data through the system.

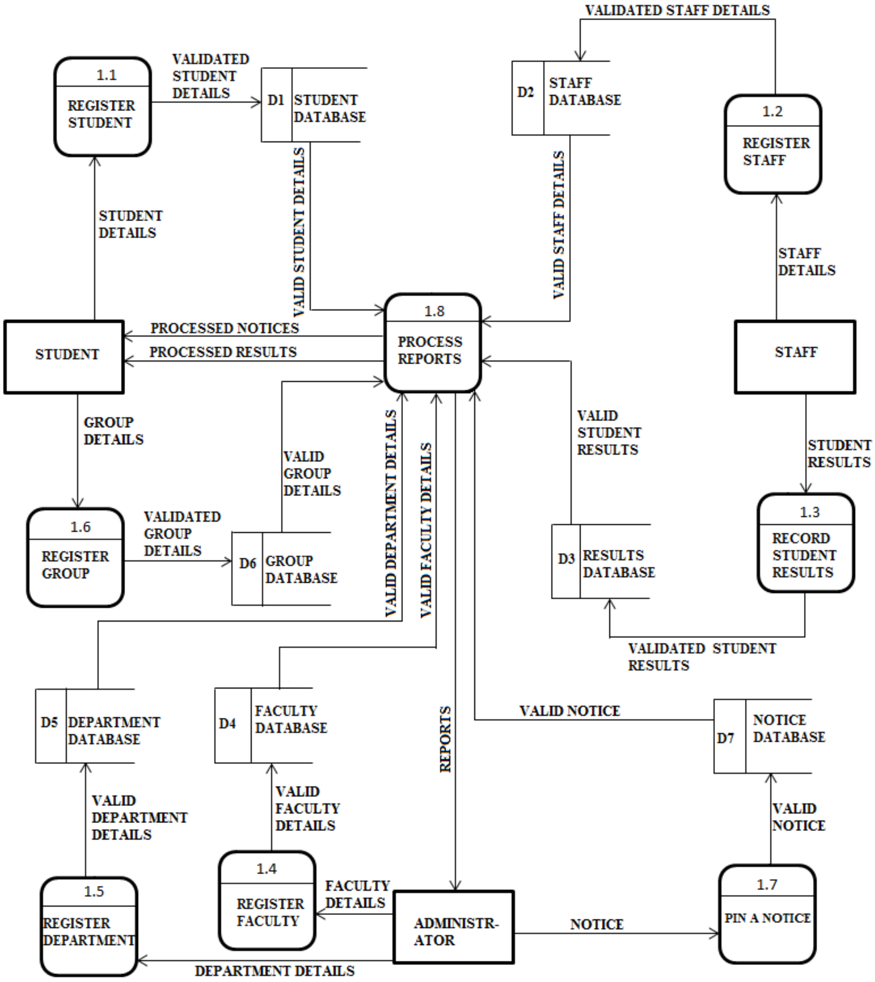


Figure 4.2: Level 1 data flow diagram for the web-based noticeboard system

4.4.4 Description of Entities

There are various entities that interact with the web base noticeboard system. The table 4.3 below describes these entities.

Entity	**Description**
Student	This is a person who studies inside the university.
Staff	This is a member of the university teaching and non teaching staff.
Administrators	This is anyone of the top administrators in the university

Table 4.3: Entities

4.4.5 Description of Processes

There are various processes involved in the web based noticeboard system. The table 4.4 below shows a description of the processes.

Process No.	**Process**	**Description**
1.1	Register student	Captures validates and stores student data
1.2	Register staff	Captures validates and stores staff details
1.3	Record students' results	Captures validates and stores Student results
1.4	Register faculty	Captures, validates and stores faculty details
1.5	Register Department	Captures, validates and stores department details
1.6	Register discussion group	Captures, validates and stores group details
1.7	Pin a notice	Captures, validates and stores a notice
1.8	Process Reports	Processes and outputs reports from the system

Table 4.4: Processes

4.4.6 Description of Data Stores

There are various data stores involved in the web based noticeboard system. The table 4.5 below shows a description of the data stores.

Store No..	**Store**	**Description**
D1	Student database	Stores valid student details.
D2	Staff database	Stores valid staff details
D3	Department database	Stores valid department details.
D4	Faculty database	Stores valid faculty details.
D5	Results database	Stores valid student results
D6	Groups database	Stores valid group details.
D7	Notice database	Stores valid notices.

Table 4.5: Data Stores

4.4.7 Entity Relationships

This is a representation of how different entities in the web-based noticeboard system are mapped to each other. The figures below show conventions are used to represent the relationships between entities. They are not included in the list of figures.

i. ——< This represents a one to many relationship. This is a situation in which a single occurrence of a record in one entity is mapped to one or more records in the other entity.

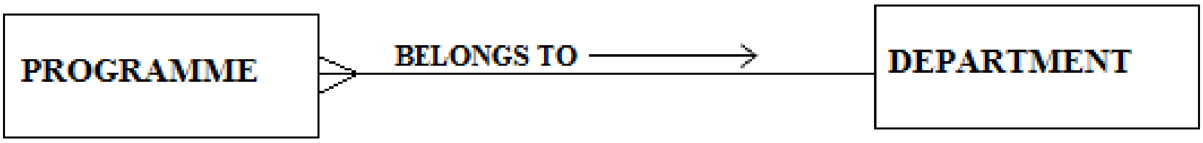
ii. —— This represents a one to one relationship. This is where an occurrence of a record in one entity can be mapped to at most one record in the other entity.

iii. >——< This represents a many to many relationship. This is where several records in one entity can be mapped to several records in another entity.

The following figures represent the relationships between the entities in the system. They are not included in the list of figures.

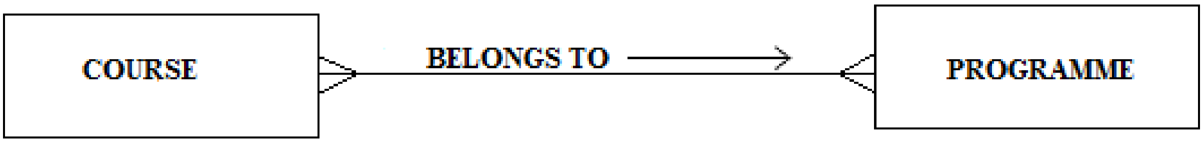
Programme - Department Relationship

A programme belongs to a single department, and a department consists of one or more programmes.



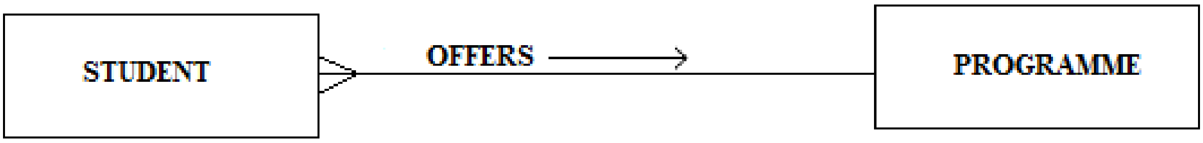
Course - Programme Relationship

A programme is made up of several courses. A single course can belong to several programs.



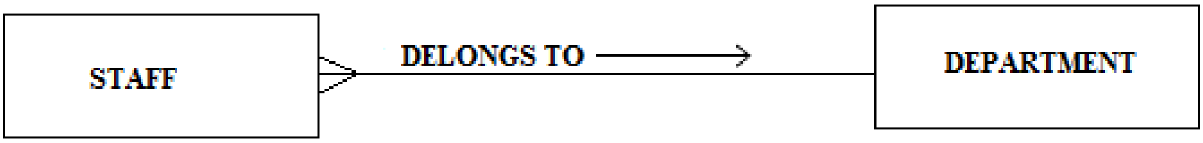
Student - Programme Relationship

A student offers a single programme, and a programme is offered by one or more students.



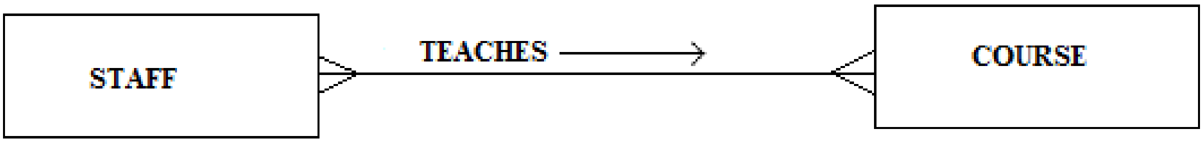
Staff- Department Relationship

A staff member belongs to a single department, and a department consists of one or more staff members.



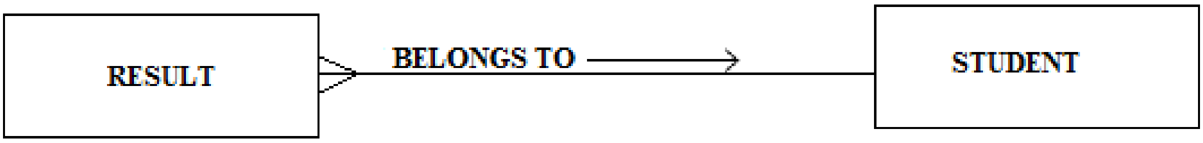
Staff -Course Relationship

A course in a particular programme is taught by one or more staff members. A staff member can teach one or more courses.



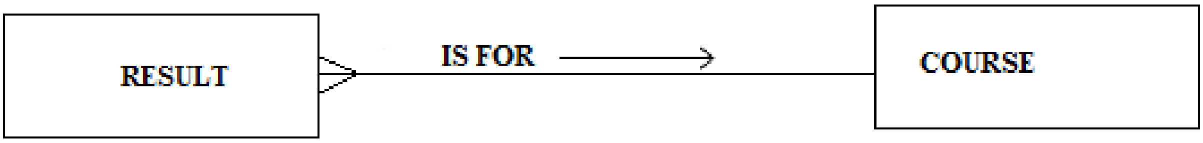
Student - Result Relationship

A student can have several results from different courses, while a particular occurrence of a result belongs to a single student .



Course - Result Relationship

A given result belongs to a single course. A course can have results for one or more students.



Notice - Staff Relationship

A notice is posted by one staff member, while a staff member can post several notices

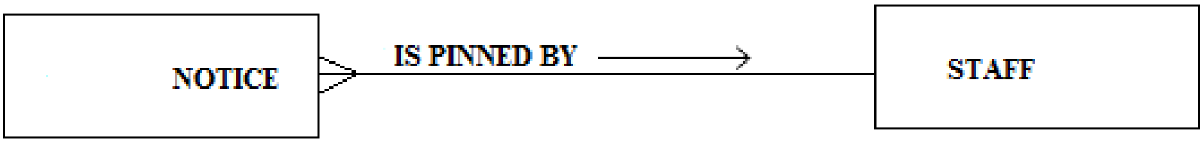


Figure 4.3 shows a comprehensive entity relationship diagram for the system.

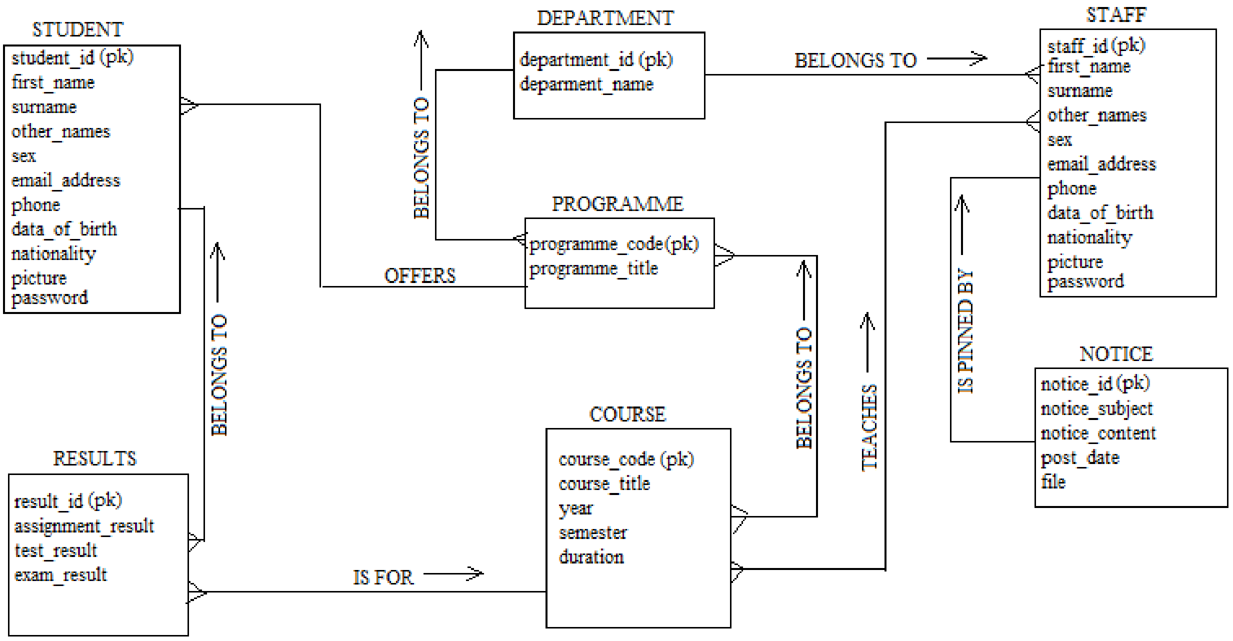


Figure 4.3: Entity Relationship Diagram for the Web-based Noticeboard System

4.4.8 Database Design

This involved the conversion of the relations into real table designs where fields were assigned corresponding field types and sizes.

The following terms were used to describe field constraints:

i. **Primary key (PK)**. This is a field that holds a unique value for each field in the table.

ii. **Foreign Key (FK)**. This is a field in the child table which is of the same data type as the primary field in a parent table, and is used as the point at which the parent table links to the child table.

All primary and foreign keys in the database were defined with fixed length for easy indexing, and speeding up database searches.

Table 4.6 is used by the system to keep a list of students in the university, and their related information.

Field Name	Data Type	Constraints	Description
Student ID	char(20)	Not Null, PK	The unique student registration number
First_Name	varchar(20)	Not Null	The first name of the student
Surname	varchar(20)	Not Null	The surname of the student
Other_ Names	varchar(20)	Null	Any other names of the student
Sex	varchar(7)	Not Null	The gender of the student
Date_of_Birth	date	Not Null	The date of birth of the student.
Nationality	int(3)	Not Null	The unique ID of the country of origin of the student
Email_Address	varchar(40)	Not Null	The email address of the student
Phone_number	varchar(15)	Not Null	The telephone number of the student
Programme_code	varchar(5)	Not Null, FK	The the code of the programme offered by the student
Password	varchar(40)	Not Null	The hashed password used by the student to log into the system.
Picture	varchar(30)	Not Null	The file path to the student's profile photo

Table 4.6: Student details

Table 4.7 is used by the system to keep a list all departments in the university.

Field Name	Data Type	Constraints	Description
Department_ID	char(10)	Not Null, PK	The unique Identification number of the department
Department_name	varchar(200)	Not Null	The name of the department
Faculty_ID	char(10)	Not Null	The unique Identification number of the faculty to which the department belongs

Table 4.7: Departments

Table 4.8 is used by the system to store a list of all staff members of the university, and their related information.

Field Name	**Data Type**	**Constraints**	**Description**
Staff_ID	char(20)	Not Null, PK	The unique Identification number of a staff member
First_Name	varchar(20)	Not Null	The first name of the staff member
Surname	varchar(20)	Not Null	The surname of the staff member
Othe_ Names	varchar(20)	Null	Any other names of the staff member
Sex	varchar(7)	Not Null	The gender of the staff member
Job_ID	char(20)	Not Null, FK	The Identification number of the staff member's job description
Department_ID	char(10),	Not Null, FK	Identification number of the staff member's department
Date_of_Birth	date	Not Null	The date of birth of the staff member.
Nationality	int(3)	Not Null	The Unique ID of the country of origin of the staff member
Email_Address	varchar(40)	Not Null	The email address of the staff member
Phone_number	varchar(15)	Not Null	The telephone number of the staff member
Password	varchar(40)	Not Null	The hashed password used by the staff member to log into the system.
Picture	varchar(30)	Not Null	The file path the staff member's profile photo

Table 4.8: Staff details

The table 4.9 is used by the system to store a list of all programmes offered in the university, and their details.

Field Name	**Data Type**	**Constraints**	**Description**
Programme_code	char(5)	Not Null, PK	The unique programme code
Programme_title	varchar(255)	Not Null	The full programme title
Duration	int	Not Null	The duration in years taken by a student to complete the programme
Department_ID	varchar(10)	Not Null, FK	The unique identification number of the department to which the programme belongs

Table 4.9: Programmes

The table 4.10 is used by the system to store a list of all courses attended in the university, and their details.

Field Name	**Data Type**	**Constraints**	**Description**
Course_code	char(8)	Not Null, PK	The unique code of the course
Course_Title	varchar(255)	Not Null	The full title of the course
Semester	int(1)	Not Null	The semester in which the course is done
Year	int(1)	Not Null	The year of study in which the course is done
Credit_Units	int(1)	Not Null	The number of hours per week for which the course is taught

Table 4.10: Courses

Table 4.11 is used by the system to store all notices posted to the noticeboard by the staff members.

Field Name	Data Type	Constraints	Description
Notice_ID	int(11)	Not Null, PK, Auto_increment	Unique Identifier of a notice
Department_ID	char(10)	Not Null, FK	The unique identifier of the department for which the notice is intended
Staff_ID	char(15)	Not Null, FK	The Unique identifier of the staff member who pins the notice
Notice_Subject	varchar(255)	Not Null	The subject that the notice is about
Notice_content	text	Not Null	The message in the notice
Post_date	TimeStamp	Not null, Default Current Timestamp	The date and time at which the notice was posted
file	varchar(60)	Not Null, Default '0'	The path to the file attached to the notice if any

Table 4.11: Notices

Table 4.12 is used by the system to store all examination, test and assignment results of every student in the university

Field Name	Data Type	Constraints	Description
Result_ID	int(11)	Not Null, PK, Auto_increment	The unique result record identifier
Course_code	char(7)	Not Null, FK	The unique code of the course unit for which are the results in question
Student_ID	char(20)	Not Null, FK	The Unique Identifier of the student owning the results
Test_result	decimal(3,1)	Null	The test score of the student for the course unit in question
Assignmet_result	decimal(3,1)	Null	The assignment score of the student for the course unit in question
Exam_result	decimal(3,1)	Null	The examination score of the stuednt for the course unit inquestion

Table 4.12: Results

Chapter 5

SYSTEM IMPLEMENTATION, TESTING AND VALIDATION

5.1 Introduction

This chapter comprises of the series of steps that were taken to implement, test, and validate the system. It concentrates mainly on translating the procedures and diagrams in chapter 4 into a computer program that is executed in the back end of the system to output the necessary interfaces (forms) that enhance user interaction with the system, building reports and querying the database to retrieve the data desired by the system users. Part of the computer code written to attain this functionality is shown in appendix D.

5.2 Site map

The Web-based noticeboard system was implemented as represented in the site map in Figure 5.1.

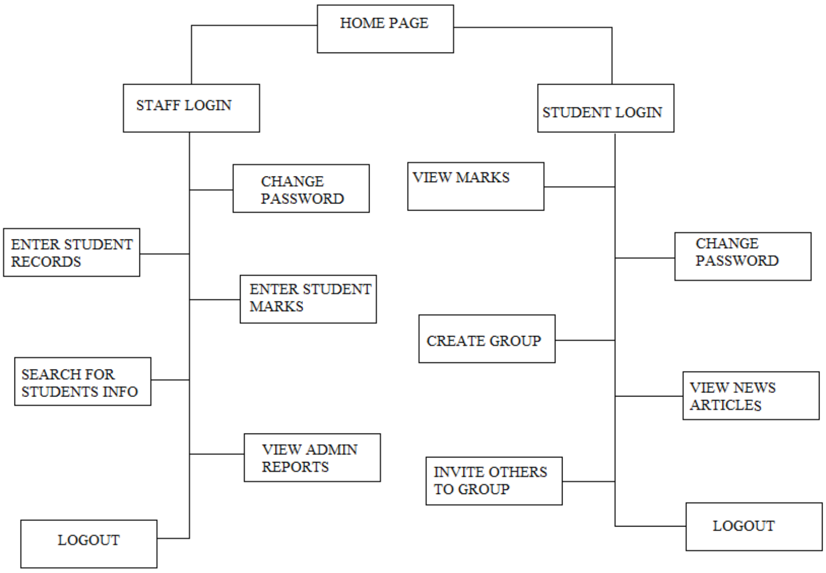


Figure 5.1: Site map for the web-based noticeboard system

5.3 Interfaces

Various interfaces were designed to enable the users to interact with the web-based noticeboard system, insert and retrieve data to and from the database, generate reports, and perform lots of other tasks. This section gives a brief description and relevancy of some of the interfaces in the system.

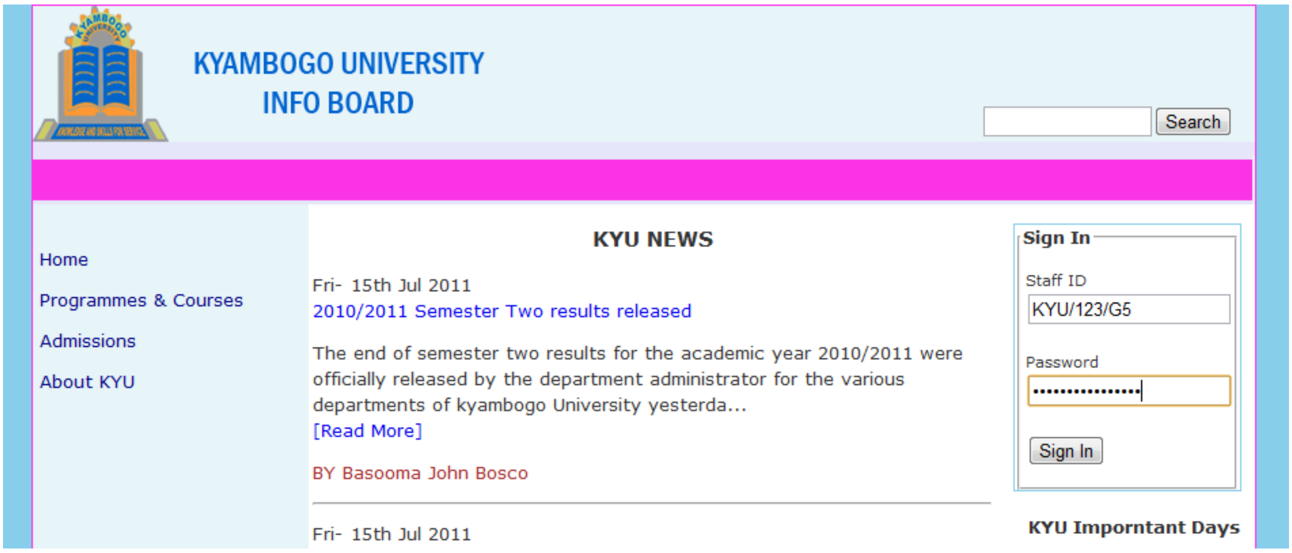
5.3.1 Login interfaces

These are authentication centres into the system that are used by the system to ensure that only authorized system users could access the services of the system. These users included staff and students of kyambogo University. The interfaces capture the unique identification numbers and passwords of the users, encrypts the password using the sha1 algorithm, and queries the database for the existence of the supplied identification number and password combination. If a match is found in the database, the system allows the user into the system, and presents them with a menu of the tasks they can carry out depending on their rights in the system. If a match is not found, the system displays a message that informs the users that they have supplied incorrect login credentials as shown in Figure 5.6.

5.3.1.1 Staff Login Interface

This interface is used for authenticating staff members as they login to the system. It is used by the system, in conjunction with the procedure described in subsection 5.3.1 above to authenticate the staff members into the system.

After staff members are logged into the system successfully, they can view or enter student results, post notices, register new students and perform so many other tasks depending on the rights assigned to them by the system administrator during registration.



KYU NEWS

Fri- 15th Jul 2011

2010/2011 Semester Two results released

The end of semester two results for the academic year 2010/2011 were officially released by the department administrator for the various departments of kyambogo University yesterda...

[Read More]

BY Basooma John Bosco

Fri- 15th Jul 2011

Staff ID: KYU/123/G5

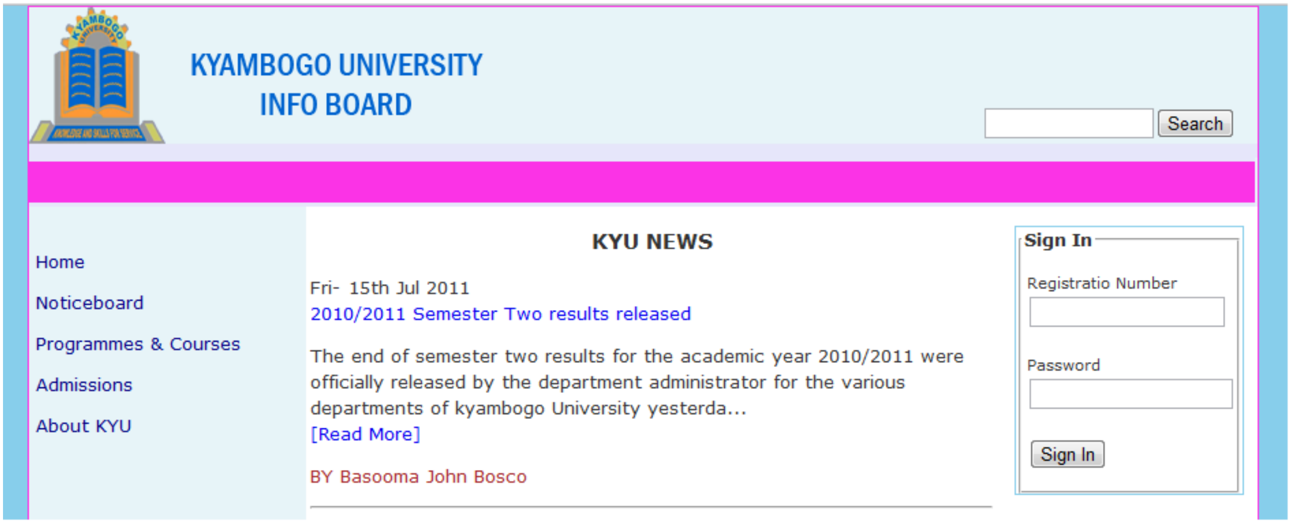
Password: ••••••••••••••••

Figure 5.2: Staff Login Interface

5.3.1.2 Student Login Interface

This interface is used by the system to authenticate students using the steps described in subsection 5.3.1.

When students login successfully into the system, they can view, but not edit their results, and notices posted by the staff members.



KYU NEWS

Fri- 15th Jul 2011
2010/2011 Semester Two results released

The end of semester two results for the academic year 2010/2011 were officially released by the department administrator for the various departments of kyambogo University yesterda...
[Read More]

BY Basooma John Bosco

Figure 5.3: Student Login Interface

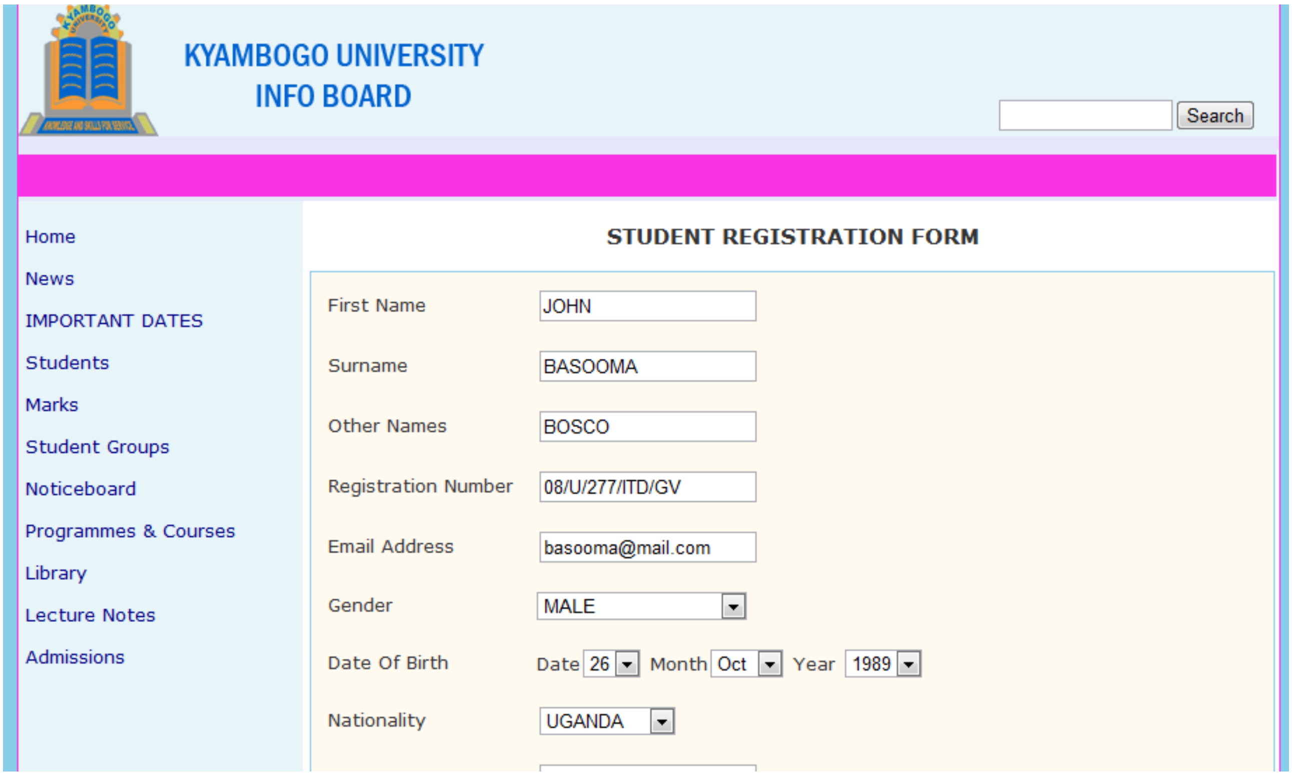
5.3.2 Registration Interfaces

These are interfaces that are used by the system administrator to register new users, departments , courses, programmes, and so many other entities into the system. The interface collects the user data from the administrator, validates that the data entered is of the correct format, and sends that data to the scripts that encrypts any fields that need to be encrypted, and run the necessary database queries to store this data into the database.

5.3.2.1 Student Registration Interface

This interface is used by the University registrar to register new students into the system. The interface captures and validates student data, and then sends it to the PHP script that encrypts the student's password, and store the student data in the database. When the students are registered, they are able to login to the system with their registration number and password combinations. The students are able to change their passwords, but not edit any other data entered by the registrar about them.

Figure 5.4 shows a snapshot of the student registration interface.



STUDENT REGISTRATION FORM

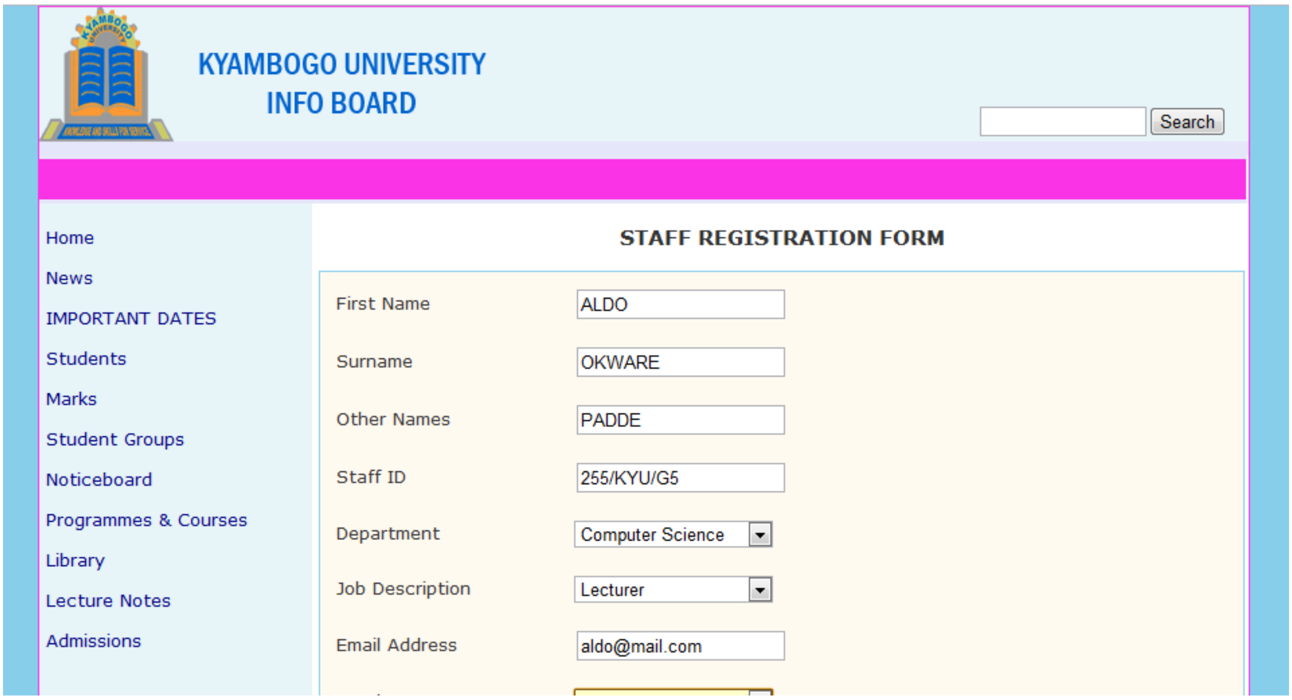
First Name	JOHN
Surname	BASOOMA
Other Names	BOSCO
Registration Number	08/U/277/ITD/GV
Email Address	basooma@mail.com
Gender	MALE
Date Of Birth	Date 26 Month Oct Year 1989
Nationality	UGANDA

Figure 5.4: Student registration interface

5.3.2.2 Staff Registration Interface

This is used by the system administrator for registering new staff members into the system. During staff registration, staff members are assigned different rights by the system administrator. These rights depend on the job description of the staff member, and they absolutely determine what a given staff member can or cannot do with the system. For example if the system administrator assigns one staff member the job description of a department administrator, and assigns another the job description of a lecturer, the department administrator can view, edit, and enter students marks, while a lecturer can only view, but not edit or enter student's marks.

Figure 5.5 shows a snapshot of the staff registration interface.



STAFF REGISTRATION FORM

First Name	ALDO
Surname	OKWARE
Other Names	PADDE
Staff ID	255/KYU/G5
Department	Computer Science
Job Description	Lecturer
Email Address	aldo@mail.com

Figure 5.5: Staff registration interface

5.4 System Testing

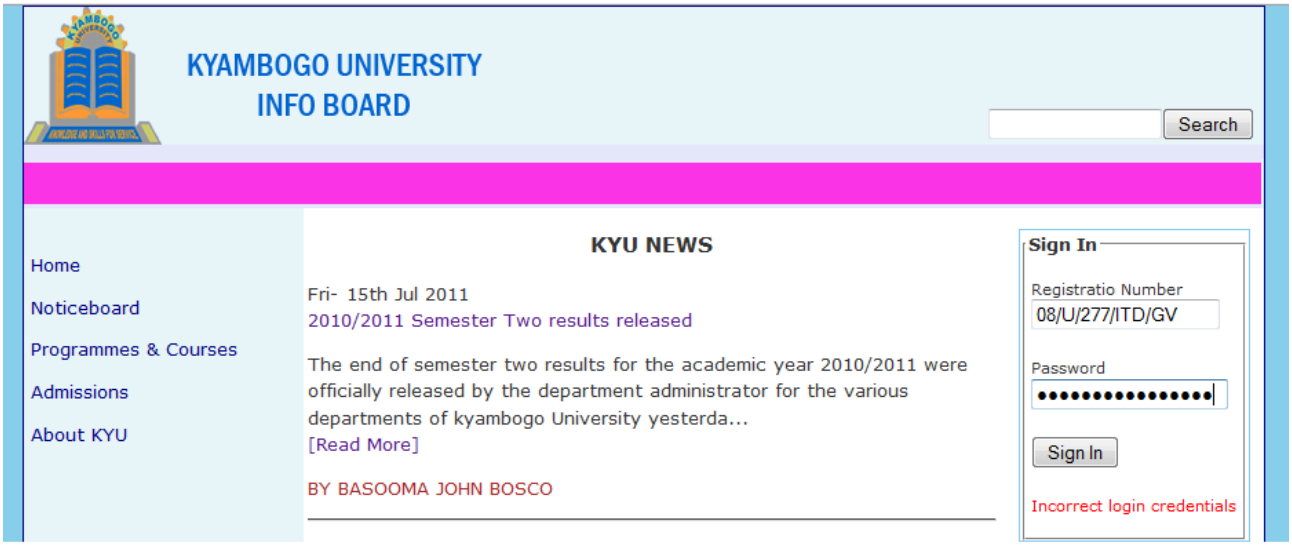
After implementation, the system was tested by a team consisting of system users and the researcher. Among the aspects on which the system was tested were security, data validation, and so many others.

The figures 5.6 and 5.7 show snapshots of the results of the security and data validation tests respectively.

5.4.1 Security Validation Test

This test was carried out to find out how the system responds if a user supplies incorrect authentication credentials.

Figure 5.6 shows an incident in which a student supplied a wrong password during login. Using the authentication procedure described in subsection 5.3.1, a match for the supplied registration number - password combination was not found. The results show that the system displayed an informative error message to the student telling him that he had supplied incorrect credentials.



KYU NEWS

Fri- 15th Jul 2011
2010/2011 Semester Two results released

The end of semester two results for the academic year 2010/2011 were officially released by the department administrator for the various departments of kyambogo University yesterda...
[Read More]

BY BASOOMA JOHN BOSCO

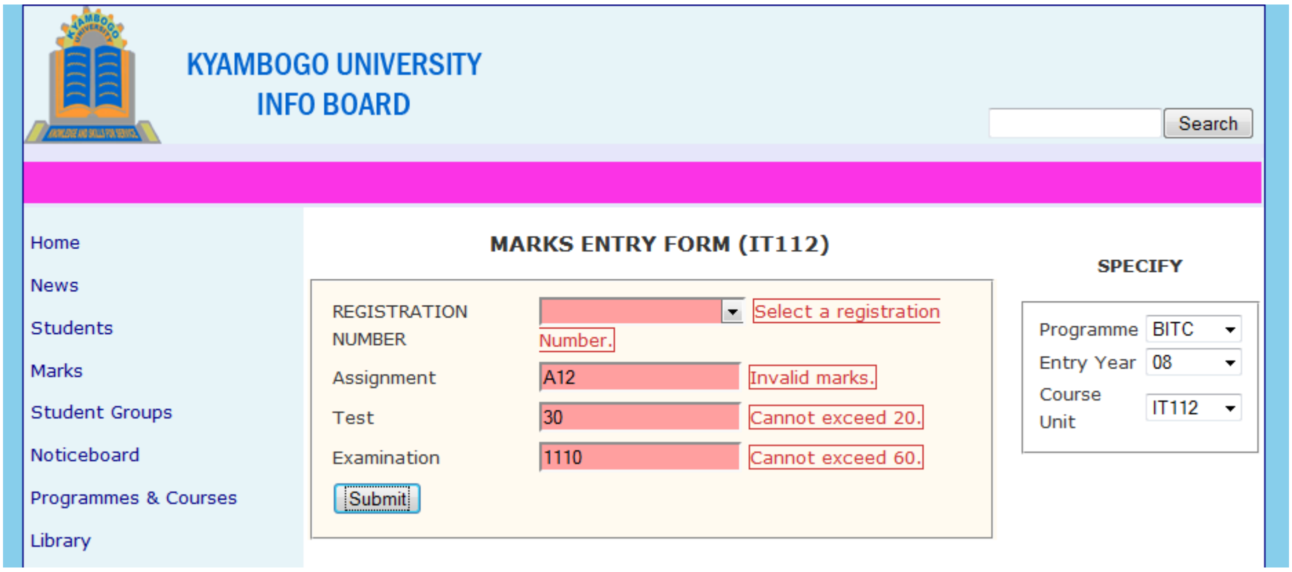
Registratio Number: 08/U/277/ITD/GV
Password: ●●●●●●●●●●●●●●●●

Incorrect login credentials

Figure 5.6: Login Validation Test

5.4.2 Data Validation Test

This test was carried out to find out how the system responds in case a user enters an incorrect data format, or data that may cause the system to malfunction. In figure 5.7, a department administrator did not select a registration number for the student whose marks he was entering, the system displayed an error message to show that the registration number was required. In the same figure, the department administrator entered a non integer value for the assignment marks. The system displayed an error message to show that he had entered an invalid value for that field. The test marks are not supposed to exceed 20, while the examination marks are not supposed to exceed 60. The system displayed informative errors when the user entered values that were greater than the maximum values as shown in figure 5.7.



MARKS ENTRY FORM (IT112)

REGISTRATION NUMBER: Select a registration Number.
Assignment: A12 Invalid marks.
Test: 30 Cannot exceed 20.
Examination: 1110 Cannot exceed 60.

SPECIFY

Programme: BITC
Entry Year: 08
Course Unit: IT112

Figure 5.7: Data Validation Test

5.5 System Validation

The tests carried out in section 5.4 showed that the system carries out data validation, security validation and other forms of validation efficiently. shows that the system is robust.

Chapter 6

DISCUSSION, CONCLUSION AND RECOMMENDATIONS

6.1 Introduction

This chapter comprises of a detailed discussion and evaluation of the results from the project research. It clearly indicates the challenges met by the researcher during the whole process of research and gives round prospective recommendation that is in line with the observations along with the way forward to solving the problems faced in Kyambogo university due to the manual noticeboard system.

6.2 Discussion

The system embraces one of the World's fastest growing and cross platform technologies; that is the World Wide Web (WWW). This is so because in the early stages of initiation of the project, the researcher was concerned about the ease of deploying the system for use.

At every stage of development of the project, the researcher tried his best to achieve what takes place in the manual noticeboard system in the most convenient possible way, and display the results to the users in the simplest most comprehensible format and on demand. For example displaying only results that belong to the current logged in student, displaying only selected student lists, displaying only selected notices, providing a search box for desired information, and so much more. The system therefore tries with the most possible intelligence to sieve the information displayed to the user.

The system is also equipped with controls both at the client side and the server side to guard against any data entry mistakes that may break the systems data integrity, and consequently Inform the user of what should be done.

If the system is used properly, the researcher is hopeful that it will reduce the long queues at the noticeboard especially when student results are released, improve on the convenience with which the students access general information pinned on the noticeboard, and their specific information.

The system will also improve on the interaction between students and alumni of the university since it supports on-line discussions.

6.2.1 Achievements

In line with the proposed scope of the system, the researcher was able to fully attain the following functionality of the system:

i. Registration of students and staff into the system and effective processing of student and staff information.

ii. Secure authentication of students and staff while they login to the system.

iii. Convenient entry and display of students marks on line and on demand.

iv. A fully fledged Noticeboard section for public notices

v. Ability of students to create and participate in on line discussion groups.

vi. Secure processing and publication of Kyambogo University news.

vii. An on-line gossip section for the university students where they share their "informal" views on events in the university.

6.2.2 Challenges

During the course of development of the system, the researcher faced the following challenges:

i. Given the scope of the project, and the time allocated for the project, the system scope was way too large to be wrapped up by the researcher since the system required a lot of research and security controls.

ii. The researcher did not have sufficient resources such as funds to facilitate the development of the system.

6.2.3 Conversion

The parallel mode of conversion should be adopted that is to say; the new system should be used concurrently with the existing traditional Manual noticeboard system for better adaptability to the new system.

6.3 Conclusion

The researcher though with limited resources managed to come up with a university web-based noticeboard system with the above listed functionality. However, the system needs to be improved with the above listed enhancements to make it of more use.

6.4 Recommendations

The web-based noticeboard system has a lot of improvements that can be made to it to give it more functionality. These include the following:

i. Incorporation of an on-line student admission section.

ii. Incorporation of an on-line Library management section.

iii. Incorporation of an on-line course content administration section for the university.

iv. Incorporation of an on-line university hospital management section.

v. Incorporation of an on-line inventory tracking section for Kyambogo university.

vi. Incorporation of an on-line fees payment tracking section for Kyambogo university.

vii. Incorporation of a live chat and instant messaging section.

viii. Incorporation of an on-line help forums.

ix. Incorporation of an email messaging section.

x. Incorporation of an SMS gateway, and so much more.

REFERENCES

[1] Abbot, Robert, K., and Hector Garcia-Molina.(1992) Scheduling Real-Time Transactions: a Performance Evaluation. Stanford University and Digital Equipment Corp. ACM.

[2] Baron, John P., Michael J. Shaw, and Andrew D. Bailey (2000). Web-based E-catalog Systems in B2B Procurement. Communications of the ACM, 43(5), 93-100.

[3] Baewicz, Jacek; Krolikowski, Zbyszko; Morzy, Tadeusz (2003). Handbook on Data Management in Information Systems. Springer, pp. 18. ISBN 3540438939.

[4] Friedlein, A. (2003). Maintaining and Evolving Successful Commercial Web Sites. San Francisco: Morgan Kaufmann.

[5] Hitt, F.J. (2002). Web Content Workflow Management. CyberTeams, Inc. A Whitepaper. Published October 30, 2002. Retrieved April 2oo11 from http://www.cyberteams.com/company/whitepapers/WebContentManagement.html.

[6] Isakowitz, Tomas, Michael Bieber, and Fabio Vitali (1998). Web Information Systems. Communications of the ACM, 41 (7), 78-80.

[7] Mowshowitz, Abbe (1997). Virtual Organizations. Communications of the ACM, 40 (9), 30-37.

[8] Okereke, C.G, Eseyin, E, Onwe, O.J, and Bimbola, E.U.A. (2009). MBA 758 DATABASE. MANAGEMENT SYSTEM. National Open University Of Nigeria. Abujja. ISBN: 978-058-331-9.

[9] Peter H. Carstensen, Lasse Vogelsang (2001) Design of Web-Based Information Systems

[10] Turoff, Murray, and Star Roxanne Hiltz (1998). Superconnectivity. Communications of the ACM. 41 (7), 116.

[11] West, K., Huff, R., Turocy, P. (2000). Managing content on the Web. Network Computing. Manhasset: October 30, 2000. Vol. 11, Issue 21.

[12] Cadgas. A .A, Tahsin. Y (2002). Web based Campus Information System, Department of Geodesy and photogrammety Engineering, Trabzon, Turkey

[13] R Madhvan (2008). Web based Campus Information System for UNIX servers,Master of science in Computer science University of Wales

[14] http://www.well.com/art/maghist02.html, Article Retrieved on Sunday, August 21, 2011.

[15] http://www.kyu.ac.ug, Retrieved on Wednesday, June 1st, 2011.

APPENDICES

- APPENDIX A: Time schedule

Item NO.	**Activity**	**Duration**	**Deliverables**
1	Writing the concept paper	March 2011 (1 week)	Concept Paper approval
2	Writing the project proposal	April 2011 (2 weeks)	Proposal approval
3	Requirements analysis and elicitation	April 2011	Requirements documentation
4	System and Software design	May 2011	System models
5	System Implementation	May 2011	Application
6	System Testing and validation	June 2011	Customer feedback
7	System documentation	June 2011	Application documentation

Table 6.1: Time Schedule

APPENDIX B: Project budget.

Item No.	**Item**	**Cost**
1	Sytem analysis tools	US$131
2	Transport	US$174
3	Phone calls	US$131
4	Development Software	US$65
5	Development Faciltation	US$9565
5	SMS Gateway	US$4000
6	Testing	US$218
7	Deployment	US$2050

Table 6.2: Project Budget

Appendix C: Questionnaire

My Name is Basooma John, Registration Number 08/U/277/ITD/GV, a third year student of Bachelor of Information Technology and Computing of Kyanbogo University.

I am carrying out a research on the current noticeboard system of Kyambogo University. Please help me to answer these questions.

Your name ....

Gender

Male □ Female □

Are you a staff member or student of Kyambogo University? Tick an option.

Staff Member □ Student □

Telephone Number ....

1. Are the noticeboards in Kyambogo University useful to you? Tick your option.

Very useful □ Useful □ Not useful □

2. How do you describe the current noticeboard system in Kyambogo University?

Manual □ Semi computerised □ Computerised □

3. How convenient is the current noticeboard system in Kyambogo University?

Very convenient □ Convenient □ Not convenient □ Very inconveniencing □

4. How often do you visit the noticeboard?

Very often □ Often □ Rarely □ Not at all □

5. How easy is it to track information pinned on the noticeboard?

Very easy □ Easy □ Hard □ Very hard □ Impossible □

6. How long does it take to process and release semester results?

Not long □ Long □ Very long □

7. How easy is it to access results and other information when you are outside the University?

Very easy □ Easy □ Hard □ Very hard □ Impossible □

8. How convenient do you think it would be if semester results and other public notices could be published and accessed over the internet?

Very convenient □ Convenient □ Inconveniencing □ Very inconveniencing □

Thank you for your time madam/sir.

APPENDIX D: Sample Code Database access class

```
<?php

    /*
    * This class contains most of the database related methods.
    * All the other classes in this project will inherit these
    * methods by extending this class.
    */

    class mysql_database
{
private $username="root";
private $hostname="localhost";
        protected $connection;
        protected static $table_name;

        /*
* This constructor creates a connection to the database as soon as the
        * database class is instantiated and selects the necessary database.
*/
        function __construct()
        {
            $this->connect_to_db();
        }

        /*
* Method to create a database connection.
*/
        protected function connect_to_db()
        {
            /*
* connect to the database using the constants imported in config.php
*/
            $this->connection=mysql_connect($this->hostname, $this->username);
            if(!$this->connection)
            {
/*
* If the connection fails, stop the script with this message.
*/
                die("Sorry: There was an error connecting to the database");
            }
            else
            {
          /*
* If the connection succeeds, try to find the noticeboard database.
*/
                if(!mysql_selectdb("noticeboard", $this->connection))
                {
                    //If the database is not found, die with this message.
                    die("Sorry: There was an error finding the necessary database");
                }
            }
        }

        /*
* This method  queries the database
* parameter: SQL string
* returns records extracted from the database
*/
        public function query($sql="")
        {
            if(!$result=mysql_query("{$sql}", $this->connection))
            {
                die("A fatal error occurred while retrieving the
                 requested data’’);
            }
            else
            {
                return $result;
            }

        }


   /*
* This method extracts records from a query result array.
* parameter: Query result array.
* Returns an associative array of a database record
*/
        public function extract_record($query_result="")
        {
            if(!empty($query_result))
            {
                $record=mysql_fetch_assoc($query_result);
                return $record;
            }
        }

/*
* This method finds a record in a database table with a particular id
* and populates the object properties.
* Parameters: id_name, id.
*returns void.
*/
        public function find_by_id($id_name="", $id="")
        {
            $sql="SELECT * FROM ";
            $sql.=static::$table_name;
            $sql.=" WHERE {$id_name}=’";
            $sql.=$id."’";
            $result=$this->query($sql);
            $result=$this->extract_record($result);
```

```
      $this->initialise_object($result);
    }



/*
* This method returns all records from a given table
* Parameters: string table_name,order field, integer limit value.
* return array of database records.
*/

    public function find_all($table, $order="", $limit="")
    {

  $sql="SELECT * FROM ";
  $sql.=static::$table_name;
  if($order!="") $sql .=" ORDER BY {$order} ";
  if($limit!="") $sql .=" LIMIT {$limit}";
  $result=$this->query($sql);
  return $result;
    }


 /*
* This mothod deletes a record from a given table in the database
* * Parameters: id_name, id.
* return void
*/

    public function delete($id_name="", $id="")
    {
      $sql="DELETE FROM ";
      $sql.=static::$table_name;
      $sql.=" WHERE {$id_name}=’";
      $sql.=$id."’ LIMIT 1";
      $result=$this->query($sql);
    }


    /*
* This method sanitises user input that may violate MySQL syntax
    * Parameter: String user_input.
* returns Sanitised string.
*/
    public function sanitise_input($input="")
    {
      if(!empty($input))
      {
        if(function_exists("mysql_real_escape_string"))
        {
          if(get_magic_quotes_gpc())
          {
```

```
                        $input=stripcslashes($input);
                        $input=mysql_real_escape_string($input);
                        return $input;
                    }
                    else
                    {
                        $input=mysql_real_escape_string($input);
                        return $input;
                    }
                }
                else
                {
                    if(get_magic_quotes_gpc())
                    {
                        return $input;
                    }
                    else
                    {
                        $input=addslashes($input);
                        return $input;
                    }

                }
            }
        }


        /*
* This method finds class variables.
* Parameter: void
* Return Array of class variables and their current values.
*/
        public function find_all_vars()
        {
            return get_object_vars($this);
        }

        /*
* This method checks if a class variable has been declared.
* Parameter: String variable name.
* return bool.
*/
        public function variable_exists($var="")
        {
            $vars=$this->find_all_vars();
            if(array_key_exists($var, $vars))
            {
                return true;
            }
            else
            {
                return false;
```

```
      }
    }


    /*
* This method initialises a class object.
* parameter: Associative array of a database record.
* return void
*/
    public function initialise_object($values)
    {
      if(is_array($values)&&!empty($values))
      {
        foreach($values as $key=>$value)
        {
          if($this->variable_exists($key))
          {
            $this->$key=$value;
          }
        }
      }
      else
        {
          die("Sorry: A fatal error occured while retrieving
          the necessary data.");
        }
    }



   /*
* This method iserts a record in a given database table.
* Parameter: String Table_name.
* Return bool.
*/

    public function insert($table="")
    {
      $fields=array();
      $values=array();
      foreach($_POST as $key=>$value)
      {
        $fields[]=$key;
        $values[]=$this->sanitise_input($value);
      }
      if(!empty($fields)&&!empty($values))
      {
        $sql="INSERT INTO ";
        $sql.=$table ." (";
        $sql.=join(", ", $fields);
        $sql.=") VALUES (’";
        $sql.=join("’, ’",$values);
```

```
                $sql.="’)";
               $result=$this->query($sql);
            }
        }

        /*
* This method returns the number of records in a given database table
* Parameter: string $table_name.
* return int number of record
*/
        public function total($table="")
        {
            $sql="SELECT count(*) FROM ";
            $sql.=$table;
            $result=$this->query($sql);
            $result=$this->extract_record($result);
            $result=array_shift($result);
            return $result;
        }


       /*
* This function updates a record in a aparticular record in the database.
* Parameter: table name, where condition field, where condition valua
*return bool
*/

         public function update($table,$filter_field,$filter_value)
       {
        /*
Generating an array of update fields and values from the post array
*/
$update_fields=array();
        foreach($_POST as $key=>$value)
        {

            $value=$this->sanitise_input($value);
            $field="{$key}";
            $field.="=’";
            $field.="{$value}’";
            $update_fields[]=($field);
        }
        $sql="UPDATE ";
        $sql.=$table;
        $sql.=" SET ";
        $sql.=join(", ",$update_fields);
        $sql.=" WHERE {$filter_field}=’";
        $sql.=$filter_value."’";
        $this->query($sql);

return true;
       }
```

```
  /*This method uploads a file and creates a database record.
*Parameters: the upload file path without extension,
*the table whre the file path will be stored,
*the primary key field of the table, the primary
* key value if the field associated with the file.
* Return bool.
*/

    public function upload_photo($upload_path,$table_name
    ,$pri_key,$pri_key_value)
    {
        $uploaded_file_info=pathinfo($_FILES['picture']['name']);
        /*Returns an array containing the path information of the file
        being uploaded, such as file name, extension, directory among others.*/
        if($this->no_file_upload_error())
{
    $file_extension=$uploaded_file_info['extension'];
    if($file_extension!="jpg" && $file_extension!="gif" && $file_extension!="png")
    {
echo"<p><font color=red>Invalid file format</font></p>";
return false;
    }
    else
    {
$new_file_path=$upload_path.".".$extension;
if(move_uploaded_file($_FILES['file']['tmp_name'],$new_file_path))
{
    $_POST['picture']=$new_file_path;
    $this->update($table_name,$pri_key,$pri_key_value);
    return true;
}
    }

}
    }



    /*
    *Function for checking and reporting file upload errors
    */
     public function no_file_upload_error()
    {
        switch($_FILES['picture']['error'])
        {
            case UPLOAD_ERR_OK:
                return true;
                break;
            case UPLOAD_ERR_CANT_WRITE:
                echo"<font color=red>ERROR: Could not write
                uploaded file to disk</font><br/>";
                return false;
            case UPLOAD_ERR_EXTENSION:
                echo"<font color=red>ERROR: File extension not
                 supported</font><br/>";
                return false;
            case UPLOAD_ERR_FORM_SIZE:
                echo"<font color=red>ERROR: File exceeded
                 maximum size</font><br/>";
                return false;
            case UPLOAD_ERR_INI_SIZE:
                echo"<font color=red>ERROR: File exceeded
                maximum size</font><br/>";
                return false;
            case UPLOAD_ERR_NO_FILE:
                echo"<font color=red>ERROR: No file was
                 selected for upload</font><br/>";
                return false;
            case UPLOAD_ERR_NO_TMP_DIR:
                echo"<font color=red>ERROR: Could not find temporary
                directory for file upload</font><br/>";
                return false;
            default:
                echo"<font color=red>File upload failure</font><br/>";
                return false;
        }

    }

}
    $database=new mysql_database();
    ?>
```

Staff Class

```
<?php
    require_once('database.class.php');
class staff extends mysql_database{
        private $taff_id;
        public $first_name;
        public $surname;
        public $other_names;
public static $table_name='staff';
public $dept_id;
public $job_id;

     public function login($staff_no="", $password="")
    {
$password=sha1(trim($password));
$sql="SELECT * FROM staff WHERE staff_id='{$staff_no}'
AND password='{$password}'";
$result=$this->query($sql);
if(mysql_numrows($result)!=1)
{
    echo"<p><font color=red> Incorrect login credentials</font></p>";

        }
else
{
    $result=mysql_fetch_assoc($result);
    $_SESSION['current_staff_id']=$result['staff_id'];
    header("Location:index.php");

}
    }

    public function logout()
    {
        $_SESSION=array();
        header("Location:index.php");
    }
}

$staff=new staff();
?>
```

The student class

```
<?php
    require_once('database.class.php');
class student extends mysql_database{
private $student_id
        public $first_name;
        public $surname;
        public $other_names;
public static $table_name='student';


     public function login($reg_no="", $password="")
    {
$password=sha1(trim($password));
$sql="SELECT * FROM student WHERE student_id=
'{$reg_no}' AND password='{$password}'";
$result=$this->query($sql);
if(mysql_numrows($result)!=1)
{
    echo"<p><font color=red> Incorrect login credentials</font></p>";

        }
else
{
    $result=mysql_fetch_assoc($result);
    $_SESSION['student_regno']=$result['student_id'];
    header("Location:home.php");

}
    }
    public function logout()
    {
        unset($_SESSION['student_regno']);
        header("Location:/infoboard/info/index.php");
    }
}
$student=new student();
?>
```