A Novel Web-Based Student Academic Records Information System

Nmaju Obasi, E. O. Nwachukwu, and C. Ugwu

Department of Computer Science, University of Port Harcourt,
Port Harcourt, Rivers State, Nigeria
nmajuobasi@yahoo.com, enoch.nwachukwu@uniport.edu, chidiebere.ugwu@uniport.edu.ng

Abstract

The system presents a single platform that will be used to manage and process data for all categories of students in a seamless and interactive manner. The design technology adopted for the implementation is a client/server technology, with MYSQL as the server technology and Visual Basic.NET as the client technology. Internet Information Server (IIS) is used as the Web server. The software development methodology adopted is the incremental model in conjunction with prototyping technique. The data used were obtained from the University of Port Harcourt and an empirical evaluation of the system shows that the system expedites processing of students' results and generation of other related academic information. This system increases efficient service delivery and provides added advantage in academic records management.

Keywords: Student Information System, Web Server, Client/Server, MYSQL

1.0 Introduction

Academic institution is an educational institution dedicated to education and research, which grants academic degrees [13]. This definition may well cover the different types of academic institutions, although each type of academic institution offers the same services in varying degrees. The basic types of academic institutions include: Primary Schools, Secondary Schools, and Advanced Educational Institutions. Our interest following there-from is in advanced educational institutions and our focus in this paper is how they manage students' information for the purpose of processing grades and administering results.

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. In many countries of the developed world, these activities are handled by automation. This, however, is far from being the case in developing countries as evident in Nigeria. Many higher institutions in Nigeria still adopt the manual method of managing students' data which is time-consuming and demanding, and are often prone to a variety of errors and disasters. Hence, it brings to the fore the need to properly address how these shortcomings (in managing students' data) could be resolved and improved. The solution to these shortcomings lies in an efficient information management system, or simply, information system.

Information systems, simply stated, transform data into useful information. It is an arrangement of people, data, processes, and interface that interact to support and improve day-to-day operations in a business as well as support the problem-solving and decision-making needs of management and users [9].

A student information system is a software application for education establishment to manage student data [13]. Synonyms include Student Information Management System, Student Records System, Student Management System, Campus Management System, or School Management System. These systems encompass a wide range of functions and capabilities, and therefore vary in size, scope, and capability. This could range from systems implemented in relatively small scale to cover students records alone, to enterprise-wide solutions that aims to cover most aspects of running large multi-campus organizations with significant local responsibility [13].

Most student information systems in use today are server-based, with the application residing on a central computer server and being accessed by client applications at various places within and even outside the school [13]. Also, according to [13], student information systems have been moving to the web since the late 1990s and that trend is accelerating as institutions replace their older systems. There are several forces that have been driving this evolution of student information systems, and as a result, leading many institutions to replace theirs. These forces are

- Demand for 24/7 web-based access to information by students, teaching staff, and (in primary and secondary education) parents.
- Increasing demands in the amount and frequency of data reporting for accountability and other purposes.
- Importance of integrating student information system with other tools, especially relating to instructions, courses and learning.

Thus, this paper is concerned with the design and implementation of a novel web-based student academic information system for higher institutions that will address some of the identified functions of a student information system. The identified functions this paper offers include: providing means to effectively maintain accurate, up-to-date student database that can be quickly and easily accessed; providing an efficient means to collect, collate, interpret, and administer students' results error-free; enabling prompt processing and releasing of relevant academic document (like transcripts, statement of results, etc) in real-time fashion; providing effective means to secure and protect students' data against infiltration and unforeseen disaster; providing an automated means for nstitution management to extract decision-making information about students' academic performance; providing a seamless communication interface between students and the institution.

Hence, the objectives of this paper therefore include:

i. To develop an information system that will provide a single platform to manage and process the data of different categories of students.
ii. To develop information system that will serve as an interface between students and institution management to enable students promptly check their grades, as well as track their progress.
iii. To develop an information system that will produce information to aid decision-making at management level.

2.0 Related Works

In the past, information systems of this nature were manually operated – it usually involved high clerical costs, delay in production of required information, reduction in work throughput, and many at times grave errors and/or omissions occur. To overcome these shortcomings, the use of information technology is now employed. But many institutions still have not put the potentials of information technology to work as observed. For instance, many institutions use different forms of spreadsheet applications to collate and process academic results. The deficiencies with this approach include: it dose not offer the benefits of a database management system; it does not provide timely information and therefore cannot be relied upon to aid decision-making; the availability of results and other information are not real-time. This approach could be summed up in one word – inefficient.

Some recent related works aimed at improving the earlier systems reviewed include:

a) “Client Server Distributed Database for Student Result Processing”, [10]. The main highlights in this work include:
   i. It will allow each academic department to maintain its own database and control their data.
   ii. It emphasizes on advantages of distributed system over centralized database system.
   iii. It focuses on improving communication between various departments’ local database system in a bid to improve computation of students results.

b) “Design and Implementation of Students’ Information System for Tertiary Institutions Using Neural Networks: An Open Source Approach”, [1]. The main highlights in this work include:
   i. It focuses on speeding up collection of students’ academic data to expedite processing of results and transcripts at various levels.
   ii. It would allow online access of results for students.

c) “Interactive Intranet Portal for effective Management in Tertiary Institution”, [4]. The main highlights in this work include:
   i. Their work addresses the problems arising from result processing, tuition fee payment, and library resources management.
   ii. As observed from this work, emphasis is on bringing some identified services to the academic community on one platform.
   iii. Only two of the mentioned services (result processing and library resources management) were discussed in the work and they didn’t meet the design expectations as at the time of publication of the paper.

d) Web portal developed for University of Port Harcourt by Cinfores – a private organization (2012)
   i. The portal offers the following services: *Information about the Institution and the various schools, departments, facilities, and other information that needs public view; Account maintenance; Online payments and payment records; Email services; SMS services;* and *Students Exams and Records Management.*
   ii. The last service is presented as a web application, and it addresses the following tasks: Students’ Registration; Academic Course Registration; Tests and Exams scores and records; Students’ personal and academic records; Academic course management and allocation; Academic Result approval; and CGPA check.
   iii. 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.
   iv. The approach adopted for result update is a major concern. These concerns include:
      - Possibility of the inbuilt formula in Microsoft Excel file used to collate results broken.
      - Not 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.

- Amount of time taken to manually search through the Excel file to update a score.
- Unnecessary duplication of data as a result of versioning. This leads to wastage of storage.

The proposed system presents an enhanced and efficient means to manage and process students' personal and academic data with a lot of added functionalities. Unlike the reviewed works, this work provides the following benefits:

- Would provide information beyond that required for routine data processing. Such information will support decision-making and managerial activities.

- Would allow generation of the necessary academic documents (Transcripts, Statement of Results, etc.) in a flexible and dynamic manner.

3.0 Materials and Methods

The software development model adopted in this project is the *incremental model* in conjunction with *prototyping* technique. This methodology has been adopted because of the nature of the work, which are iterations of analyze-design-implement activities called *builds*. Each build is improved and functionality added until it becomes a new build.

Hence, to address the identified problems, as well as the deficiencies in the existing system, we employed the following methods:

- Interviewed stakeholders, including management staff and other staff of the University of Port Harcourt.
- Surveyed and analyzed the different academic documents the institution presents to students.
- Reviewed best practice literature regarding information technology projects in government and private settings.
- Reviewed other student information collection and management methods and systems to gain in-depth understanding of what the system should be like; what it should do; and how it should perform.
- Collected a comprehensive academic course data from the University of Port Harcourt used for testing the system.
- Simulated a fictitious 4-year academic result scores for 20 students per session for Computer Science department (which we used as the test department).

4.0 Analysis of Proposed System

Two levels of system design have been made using models. The first model, logical model, shows what the system is. The two logical models used include data model and process model, respectively. The identified entities and business assertions for these entities are as discussed below, and a normalized and fully attributed entity-relationship diagram (ERD) representing the data model is as shown in Figure 1.

The entities identified include Faculties; Departments; Degree Courses; Academic Courses; Assigned Academic Courses; Academic Awards; Degree Course Levels; Degree Course Awards; States; Local Government Areas (LGAs); Students; Users; Academic Sessions; Students Registration; Academic Courses Registration; Faculty Colour Codes.

The business assertions as would be represented in the entity-relationship diagram (ERD) include:

- We need to store data about Faculties. For a Faculty, we need to know the faculty name, faculty code, faculty colour, as well as the Faculty Dean. The value of FacultyID uniquely identifies one and only one Faculty.
- We need to store data about Departments. For a Department, we need to know the faculty, department code, department name, as well as the Head of Department. The value of DeptID uniquely identifies one and only one Department.
- We need to store data about Degree Courses. For a Degree Course, we need to know the department, and degree course name. The value of DegreeID uniquely identifies one and only one Degree Course.
- We need to store data about Academic Courses. For an Academic Course, we need to know the course code, course title, course unit, course semester, course department, and the course description. The value of CourseID uniquely identifies one and only one Academic Course.
- We need to store data about Academic Courses Assigned to Lecturers. For an Assigned Academic Course, we need to know the Academic Session, Lecturer, Academic Course, HOD Result Approval, Faculty Dean Result Approval, and Senate Result Approval. The value of AssignID uniquely identifies one and only one Assigned Academic Course to a Lecturer for a given Academic Session.
- We need to store data about Academic Awards. For an Academic Award, we need to know the award code, award name, and degree type. The value of AwardID uniquely identifies one and only one Academic Award.
- We need to store data about Faculty Colour Codes. For a Faculty Colour, we need to know the colour code and colour name. The value of ColorCode uniquely identifies one and only one Faculty Colour.
- We need to store data about Degree Courses' Levels. For a Degree Course Level, we need to know the degree course, department, degree course level, first semester maximum credit unit, first semester minimum credit unit, second semester maximum credit unit, and second semester minimum credit unit. Also, we need to know the academic adviser. The value of LevelID uniquely identifies one and only one Degree Course Level.
- We need to store data about Degree Courses' Awards. For a Degree Course Award, we need to know the degree course, award code, and award duration. The value of DegAwardID uniquely identifies one and only one Degree Course Award.
- We need to store data about Academic Sessions. For an Academic Session, we need to know the session name and session status. The value of SessionID uniquely identifies one and only one Academic Session.

- We need to store data about States. For a State, we need to know the State's name. The value of State uniquely identifies one and only one State (of the Federation).
- We need to store data about Local Government Areas. For a Local Government Area, we need to know the State and Local Government's name. The value of LGA uniquely identifies one and only one Local Government Area (of the Federation).
- We need to store data about Students. For a Student, we need to know surname, first name, other names, sex, title, religion, marital status, date of birth, place of birth, hometown, Local Government Area, State, Nationality, home address, email address, admission year, admission number, degree course, department, faculty, degree award code, degree award name, degree award type, programme type, programme duration, picture, signature, PIN (Personal Identification Number), Password and graduation status. We also need to know the graduation year. The value of StudentID uniquely identifies one and only one Student.
- We need to store data about Users. For a User, we need to know the user code, user's name, user's department, user's class, user's category, user's password, and user's status. The value of UserID uniquely identifies one and only one User.
- We need to store data about Students Registration. For a Student's Registration, we need to know the student, academic session, study year, degree course level, degree course, first semester maximum credit unit, first semester minimum credit unit, second semester maximum credit unit, and second semester minimum credit unit. We also need to know the registration date. The value of StudentRegID uniquely identifies one and only one Student Registration.
- We need to store data about Academic Courses Registered for each Registered Student, and Scores for each Registered Academic Course for each Registered Student. For an Academic Course Registration, we need to know the student the course is registered for, academic session, semester, academic course, course unit, lecturer in-charge, and if the registered course is a carry-over course or not. Also, we need to know the score, grade, and grade point as well as the user that entered or modified the score, date of update/modification, and previous score and grade before the modification. The value of CourseRegID uniquely identifies one and only one Academic Course Registration.

The proposed system would be difficult to fully understand when viewed as a whole (single process) – we need to break down the system into smaller subsystems. This is known as **decomposition**. The decomposition diagram is as shown in Figure 2. The decomposition diagram shows each distinct operation that need to be performed by the system. In some cases, an operation is further broken down to sub-operations that will be performed. Some operations are dependent on other operations, and therefore cannot be performed until the operations they are dependent upon are performed. For instance, an academic course cannot be assigned to a lecturer if that academic course has not been pre-defined in the system. Furthermore, a student cannot register for an academic course for a given academic session if the academic course has not been assigned to a lecturer for that academic session. By this, Academic Course Registration is dependent on Assign Academic Courses, and Assign Academic Courses is dependent on Academic Course Definition.

The second model, physical model, shows how the system is physically and technically implemented. The output of the physical modeling is the **database model**.

The database models representing some of the entities are:
:

Table 1: Users table

Entity: Users; **Table name**: tblUsers			
Attributes	**Data Type**	**Constraints**	**Remarks**
UserID	NUMBER (9)	NOT NULL	Primary Key; No Duplicates; Auto-generated
UserCode	VARCHAR (10)	NOT NULL	Size=10; No Duplicates
UserName	VARCHAR (50)	NOT NULL	Size=50
UserCategory	VARCHAR (25)	NOT NULL	Size=25
Password	VARCHAR (10)	NOT NULL	Size=10; Encrypted
UserDept	NUMBER (9)	NOT NULL	Foreign Key; Duplicates OK
UserClass	VARCHAR (20)	NOT NULL	Size=20
UserStatus	VARCHAR (15)	NOT NULL	Size=15
Email	VARCHAR (100)	NULL	Size=100

Table 2: Academic Courses table

Entity: Academic Courses; **Table name**: tblAcademicCourses			
Attributes	**Data Type**	**Domain**	**Remarks**
CourseID	NUMBER (9)	NOT NULL	Primary Key; No Duplicates; Auto-generated
CourseCode	VARCHAR (10)	NOT NULL	Size=10; No Duplicates
CourseTitle	VARCHAR (100)	NOT NULL	Size=100
CourseUnit	NUMBER (1)	NOT NULL	
CourseSemester	VARCHAR (6)	NOT NULL	Size=6
CourseDescription	VARCHAR (MAX)	NULL	Unlimited size of text
DeptID	NUMBER (9)	NOT NULL	Foreign Key; Duplicates OK

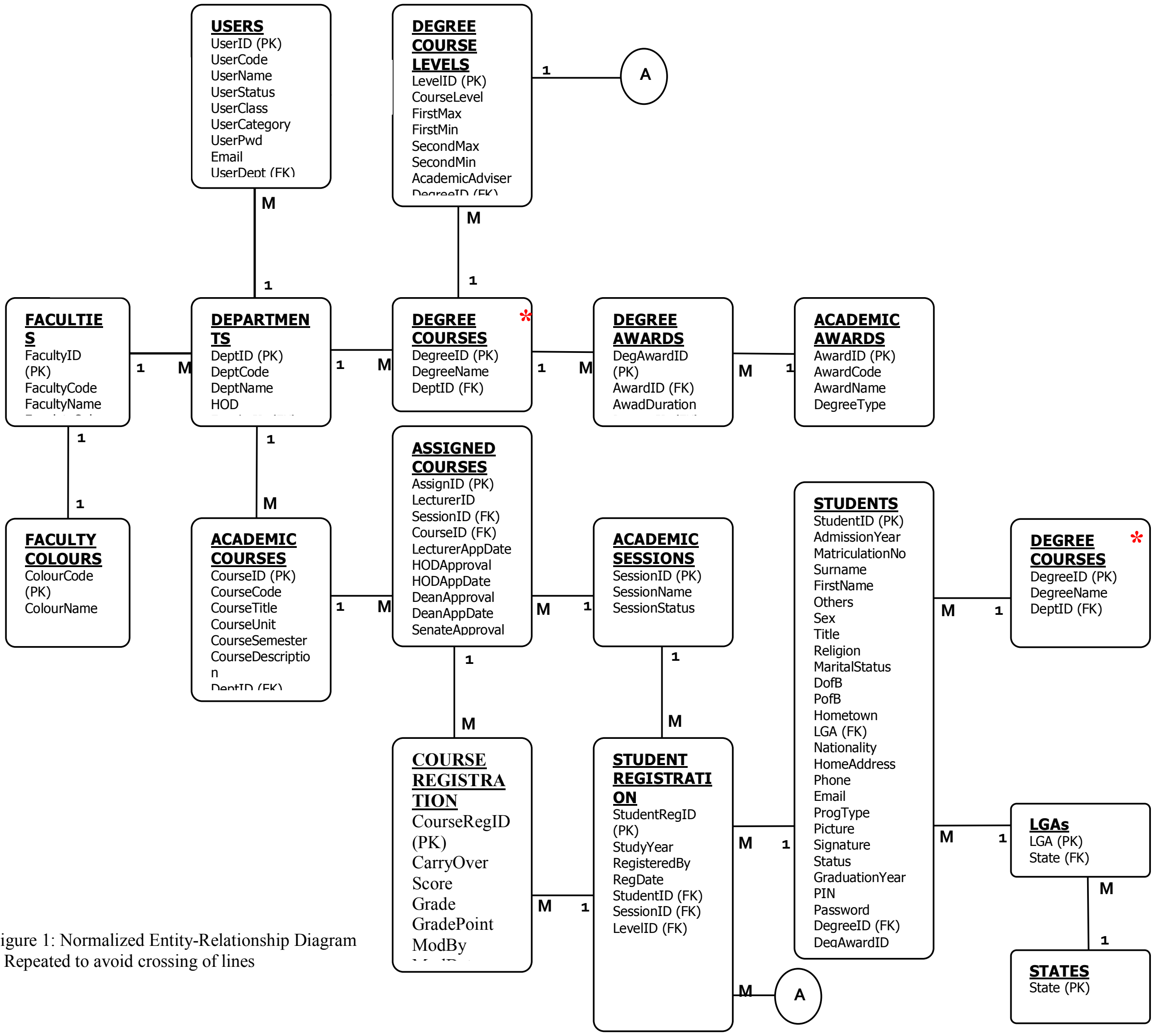


Figure 1: Normalized Entity-Relationship Diagram
* Repeated to avoid crossing of lines

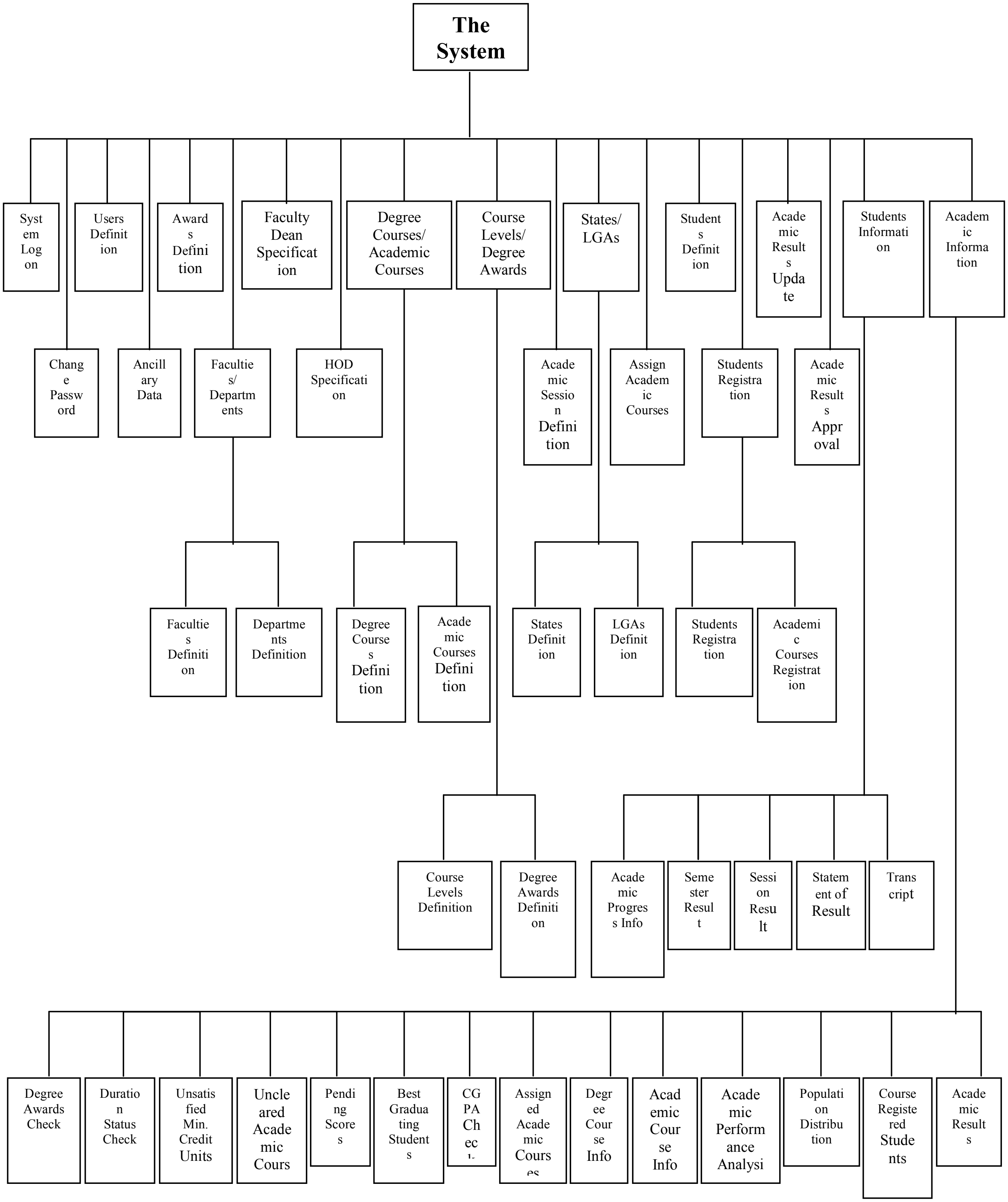


Figure 2: Decomposition diagram of the proposed system

Figure 3 shows the System Logon flowchart representing the logon operation

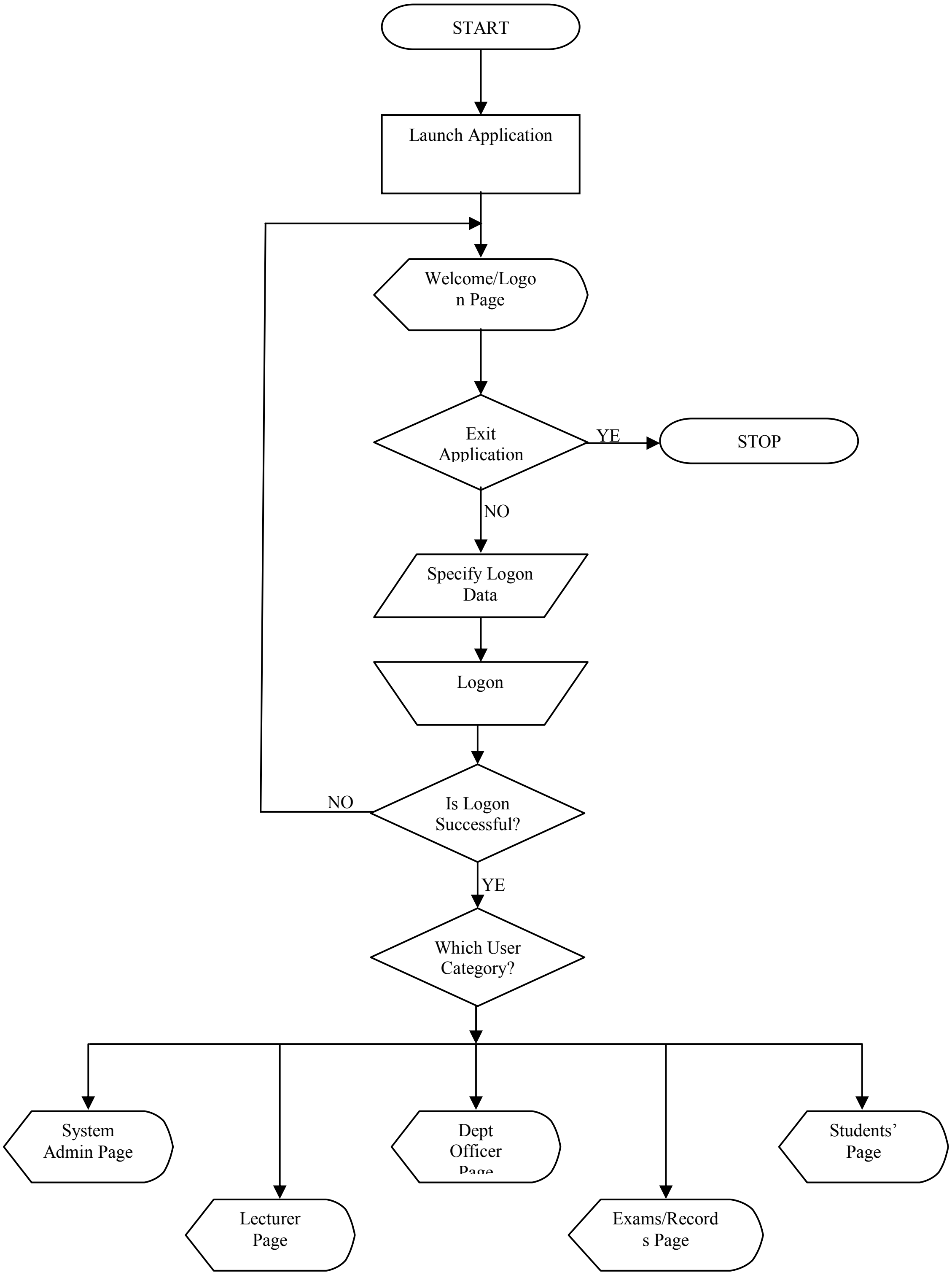
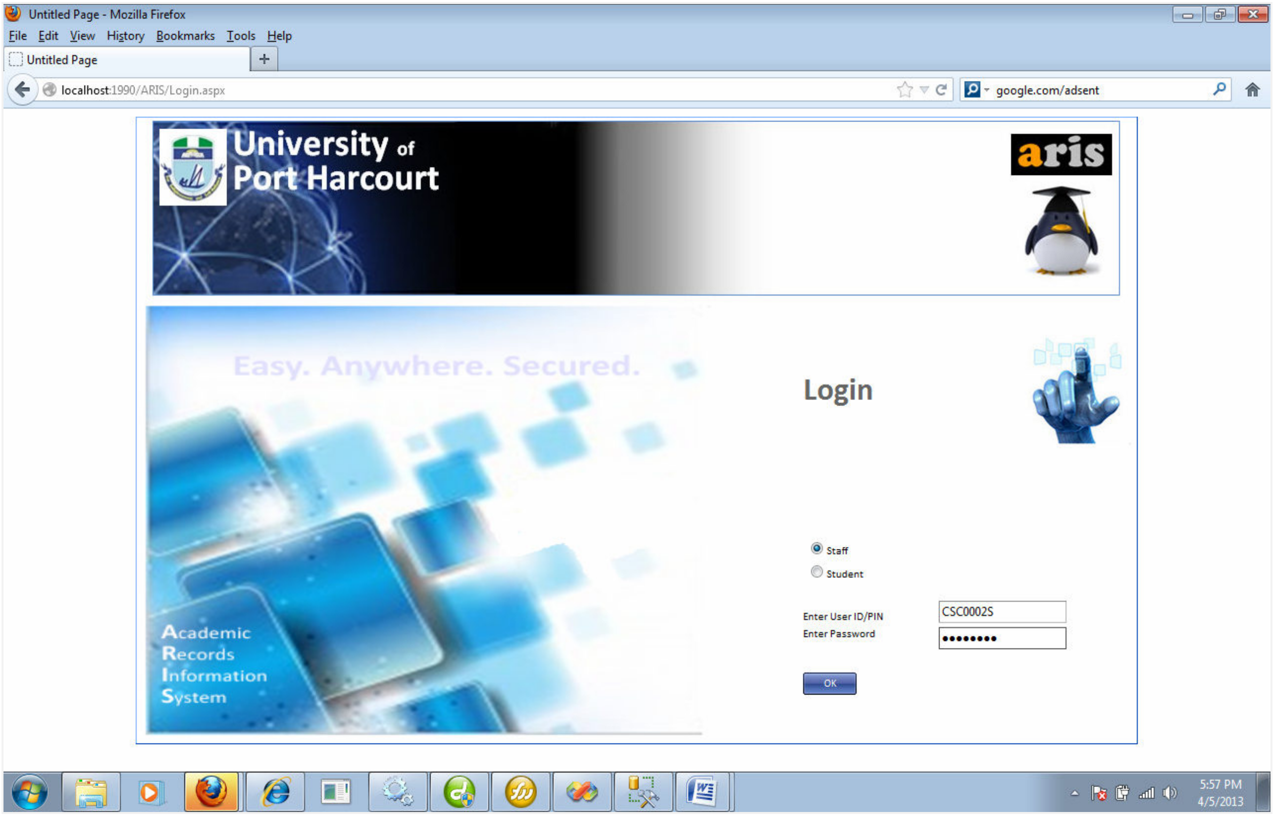


Figure 3: System Logon Flowchart

5.0 Implementation of ARIS

A working student information system, called ARIS, was developed using Visual Basic.NET as the client-side technology and MYSQL as the server-side technology.

ARIS Welcome/Logon page allows users to log on to the system (Figure 4). As indicated in the System Logon flowchart (Figure 3), successfully logged on users are re-directed to the appropriate page, depending on their user categories


Untitled Page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Untitled Page
localhost:1990/ARIS/Login.aspx
google.com/adsent
University of Port Harcourt
aris
Easy. Anywhere. Secured.
Login
Academic Records Information System
OK
5:57 PM
4/5/2013

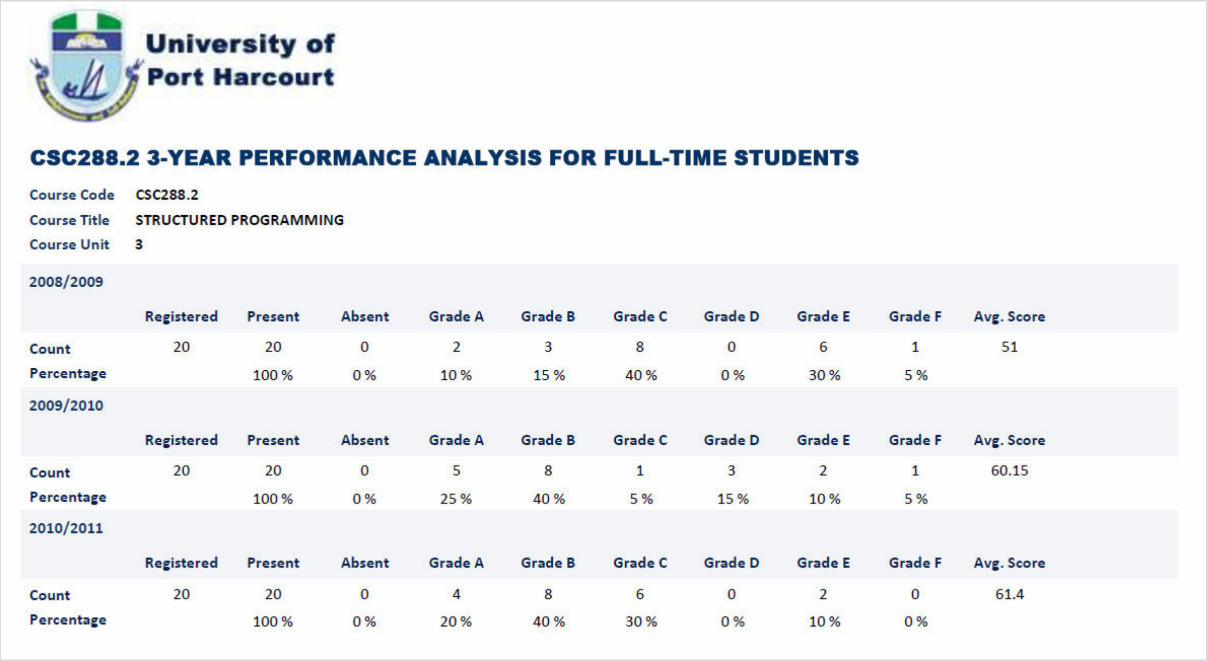
● Staff
○ Student

Enter User ID/PIN CSC0002S
Enter Password •••••••••

Figure 4: ARIS Welcome/Logon page

6.0 Sample Output/Results

Outputs present information to the system users, and are the most visible component of a working information system. As such, they are often the basis for the users' and system owners' final assessment of the system's value. Figures 5 to 9 are some of the output results of the. The output results are based on search criteria.



University of Port Harcourt

CSC288.2 3-YEAR PERFORMANCE ANALYSIS FOR FULL-TIME STUDENTS

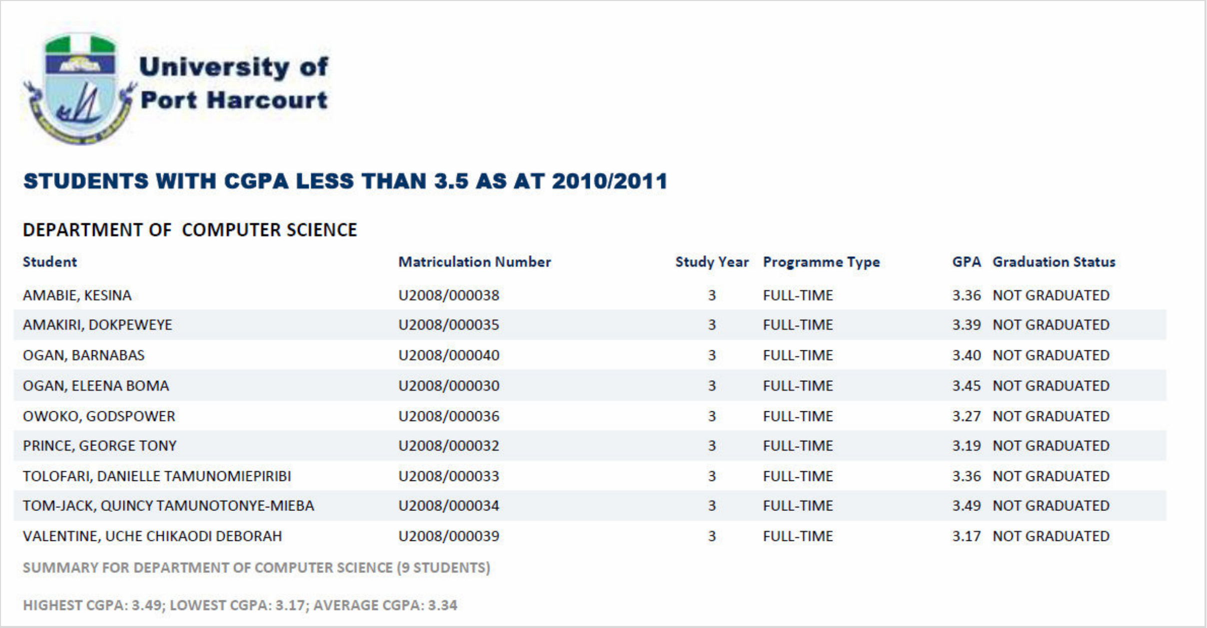
Course Code	CSC288.2
Course Title	STRUCTURED PROGRAMMING
Course Unit	3

2008/2009	Registered	Present	Absent	Grade A	Grade B	Grade C	Grade D	Grade E	Grade F	Avg. Score
Count	20	20	0	2	3	8	0	6	1	51
Percentage		100 %	0 %	10 %	15 %	40 %	0 %	30 %	5 %	

2009/2010	Registered	Present	Absent	Grade A	Grade B	Grade C	Grade D	Grade E	Grade F	Avg. Score
Count	20	20	0	5	8	1	3	2	1	60.15
Percentage		100 %	0 %	25 %	40 %	5 %	15 %	10 %	5 %	

2010/2011	Registered	Present	Absent	Grade A	Grade B	Grade C	Grade D	Grade E	Grade F	Avg. Score
Count	20	20	0	4	8	6	0	2	0	61.4
Percentage		100 %	0 %	20 %	40 %	30 %	0 %	10 %	0 %	

Figure 5: Academic Course Performance Analysis



University of Port Harcourt

STUDENTS WITH CGPA LESS THAN 3.5 AS AT 2010/2011

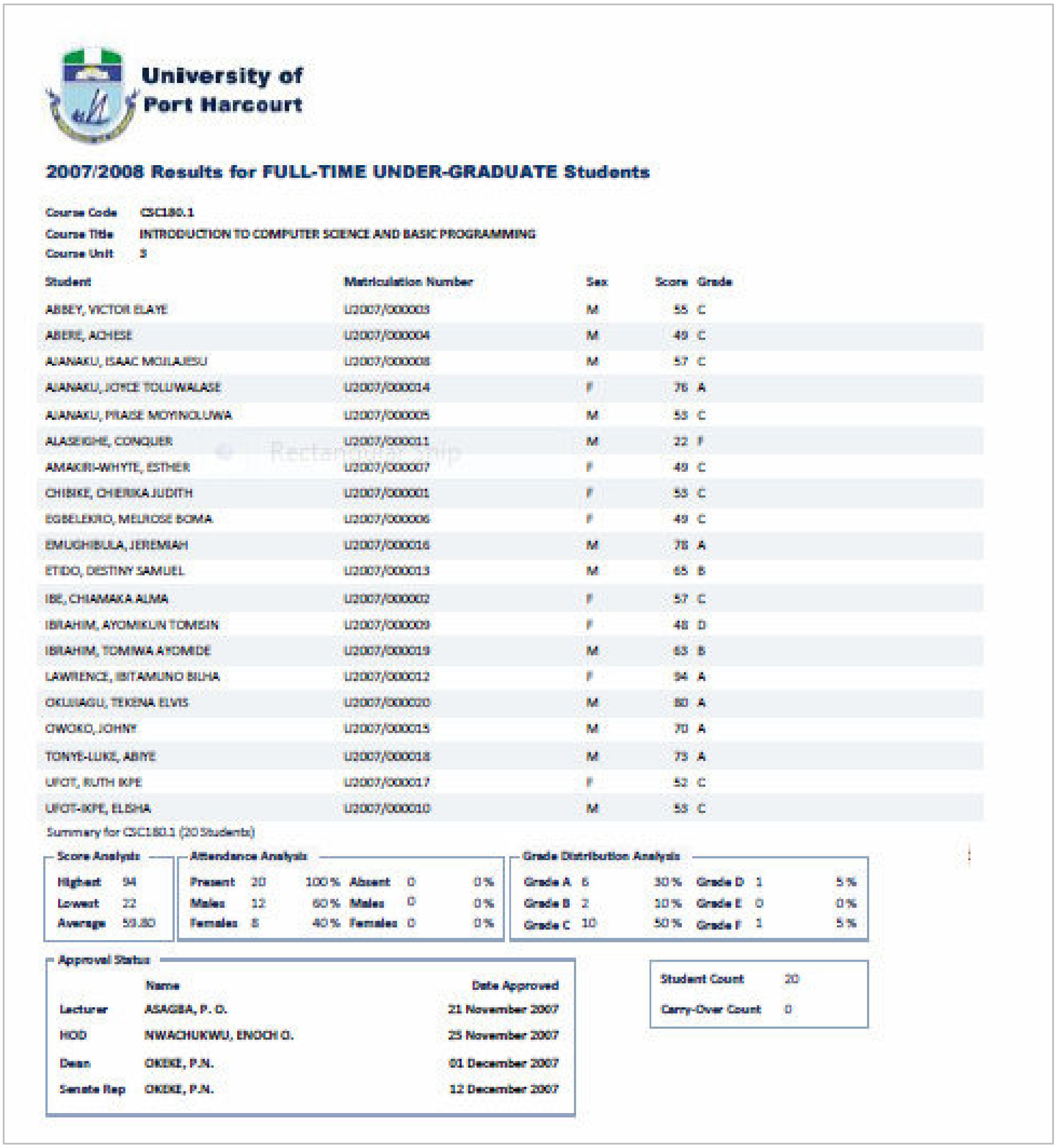
DEPARTMENT OF COMPUTER SCIENCE

Student	Matriculation Number	Study Year	Programme Type	GPA	Graduation Status
AMABIE, KESINA	U2008/000038	3	FULL-TIME	3.36	NOT GRADUATED
AMAKIRI, DOKPEWEYE	U2008/000035	3	FULL-TIME	3.39	NOT GRADUATED
OGAN, BARNABAS	U2008/000040	3	FULL-TIME	3.40	NOT GRADUATED
OGAN, ELEENA BOMA	U2008/000030	3	FULL-TIME	3.45	NOT GRADUATED
OWOKO, GODSPOWER	U2008/000036	3	FULL-TIME	3.27	NOT GRADUATED
PRINCE, GEORGE TONY	U2008/000032	3	FULL-TIME	3.19	NOT GRADUATED
TOLOFARI, DANIELLE TAMUNOMIEPIRIBI	U2008/000033	3	FULL-TIME	3.36	NOT GRADUATED
TOM-JACK, QUINCY TAMUNOTONYE-MIEBA	U2008/000034	3	FULL-TIME	3.49	NOT GRADUATED
VALENTINE, UCHE CHIKAODI DEBORAH	U2008/000039	3	FULL-TIME	3.17	NOT GRADUATED

SUMMARY FOR DEPARTMENT OF COMPUTER SCIENCE (9 STUDENTS)

HIGHEST CGPA: 3.49; LOWEST CGPA: 3.17; AVERAGE CGPA: 3.34

Figure 6: CGPA Information



University of
Port Harcourt

2007/2008 Results for FULL-TIME UNDER-GRADUATE Students

Course Code	CSC180.1
Course Title	INTRODUCTION TO COMPUTER SCIENCE AND BASIC PROGRAMMING
Course Unit	3

Student	Matriculation Number	Sex	Score	Grade
ABBEY, VICTOR ELAYE	U2007/000003	M	55	C
ABERE, ACHESE	U2007/000004	M	49	C
AJANAKU, ISAAC MOJLAJESU	U2007/000008	M	57	C
AJANAKU, JOYCE TOLUWALASE	U2007/000014	F	76	A
AJANAKU, PRAISE MOYINOLUWA	U2007/000005	M	53	C
ALASEIGHE, CONQUER	U2007/000011	M	22	F
AMAKIRI-WHYTE, ESTHER	U2007/000007	F	49	C
CHIBIKE, CHIERIKA JUDITH	U2007/000001	F	53	C
EGBELEKRO, MELROSE BOMA	U2007/000006	F	49	C
EMUGHIBULA, JEREMIAH	U2007/000016	M	78	A
ETIDO, DESTINY SAMUEL	U2007/000013	M	65	B
IBE, CHIAMAKA ALMA	U2007/000002	F	57	C
IBRAHIM, AYOMIKUN TOMISIN	U2007/000009	F	48	D
IBRAHIM, TOMIWA AYOMIDE	U2007/000019	M	63	B
LAWRENCE, IBITAMUNO BILHA	U2007/000012	F	94	A
OKUJAGU, TEKENA ELVIS	U2007/000020	M	80	A
OWOKO, JOHNY	U2007/000015	M	70	A
TONYE-LUKE, ABIYE	U2007/000018	M	73	A
UFOT, RUTH IKPE	U2007/000017	F	52	C
UFOT-IKPE, ELISHA	U2007/000010	M	53	C

Summary for CSC180.1 (20 Students)

Score Analysis

Highest	94
Lowest	22
Average	59.80

Attendance Analysis

Present	20	100 %	Absent	0	0 %
Males	12	60 %	Males	0	0 %
Females	8	40 %	Females	0	0 %

Grade Distribution Analysis

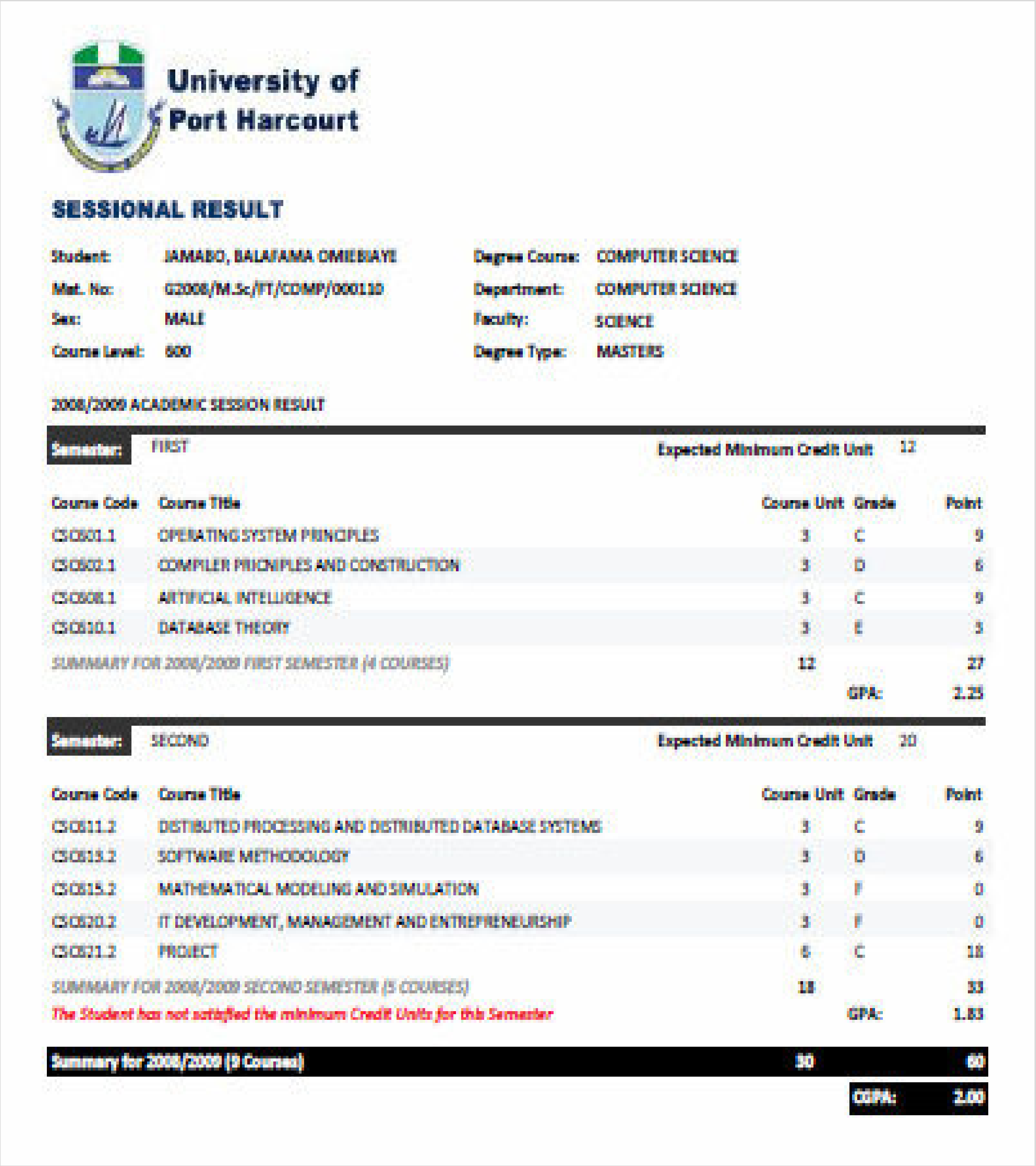
Grade A	6	30 %	Grade D	1	5 %
Grade B	2	10 %	Grade E	0	0 %
Grade C	10	50 %	Grade F	1	5 %

Approval Status

	Name	Date Approved
Lecturer	ASAGBA, P. O.	21 November 2007
HOD	NWACHUKWU, ENOCH O.	25 November 2007
Dean	OKEKE, P.N.	01 December 2007
Senate Rep	OKEKE, P.N.	12 December 2007

Student Count	20
Carry-Over Count	0

Figure 7: Academic Course Result



University of Port Harcourt

SESSIONAL RESULT

Student:	JAMABO, BALAFAMA OMIEBIAYE	Degree Course:	COMPUTER SCIENCE
Mat. No:	G2008/M.Sc/FT/COMP/000110	Department:	COMPUTER SCIENCE
Sex:	MALE	Faculty:	SCIENCE
Course Level:	600	Degree Type:	MASTERS

2008/2009 ACADEMIC SESSION RESULT

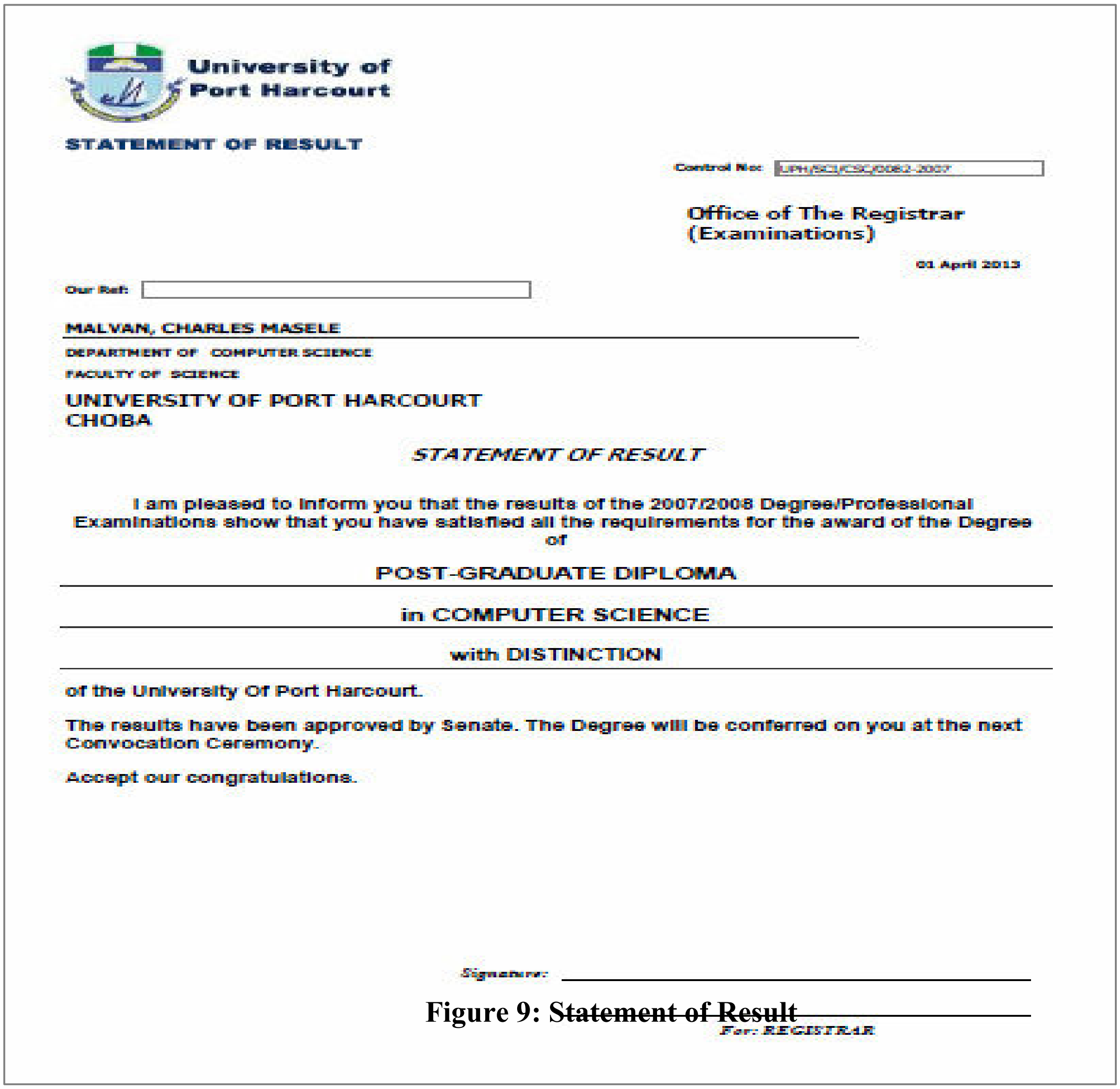
Semester: FIRST — Expected Minimum Credit Unit 12

Course Code	Course Title	Course Unit	Grade	Point
CSC601.1	OPERATING SYSTEM PRINCIPLES	3	C	9
CSC602.1	COMPILER PRICNIPLES AND CONSTRUCTION	3	D	6
CSC608.1	ARTIFICIAL INTELLIGENCE	3	C	9
CSC610.1	DATABASE THEORY	3	E	3
SUMMARY FOR 2008/2009 FIRST SEMESTER (4 COURSES)		12		27
			GPA:	2.25

Semester: SECOND — Expected Minimum Credit Unit 20

Course Code	Course Title	Course Unit	Grade	Point
CSC611.2	DISTIBUTED PROCESSING AND DISTRIBUTED DATABASE SYSTEMS	3	C	9
CSC613.2	SOFTWARE METHODOLOGY	3	D	6
CSC615.2	MATHEMATICAL MODELING AND SIMULATION	3	F	0
CSC620.2	IT DEVELOPMENT, MANAGEMENT AND ENTREPRENEURSHIP	3	F	0
CSC621.2	PROJECT	6	C	18
SUMMARY FOR 2008/2009 SECOND SEMESTER (5 COURSES)		18		33
The Student has not satisfied the minimum Credit Units for this Semester			GPA:	1.83
Summary for 2008/2009 (9 Courses)		30		60
			CGPA:	2.00

Figure 8: Sessional Result.



University of
Port Harcourt

STATEMENT OF RESULT

Control No: UPH/SCI/CSC/0082-2007

Office of The Registrar
(Examinations)

01 April 2013

Our Ref:

MALVAN, CHARLES MASELE
DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF SCIENCE

UNIVERSITY OF PORT HARCOURT
CHOBA

STATEMENT OF RESULT

I am pleased to inform you that the results of the 2007/2008 Degree/Professional Examinations show that you have satisfied all the requirements for the award of the Degree of

POST-GRADUATE DIPLOMA

in COMPUTER SCIENCE

with DISTINCTION

of the University Of Port Harcourt.

The results have been approved by Senate. The Degree will be conferred on you at the next Convocation Ceremony.

Accept our congratulations.

Signature: ____________________

For: REGISTRAR

Figure 9: Statement of Result

7.0 Discussion of Results

Academic Course Performance Analysis output (Figure 5) provides performance analysis for a given academic course. The analysis is a 3-year performance analysis for students of a given programme type. The periods considered in the analysis include the specified academic session and the two immediate past academic sessions (if available). For instance, if the performance analysis in, say, CSC502.1 is required and the specified academic session is 2001/2002, the periods the system will consider include 1999/2000, 2000/2001, and 2001/2002 academic sessions, respectively. CGPA Information output (Figure 6) presents the names of students whose CGPA is exactly or less than or greater than a specified CGPA value as at a specified academic session. Academic Course Result output (Figure 7) provides the scores and grades of students who registered for a particular academic course for a given academic session. Also, the following information are presented: *Score Analysis (Highest Score, Lowest Score, Average Score); Attendance Analysis (Number of Students present and absent, broken down to males and* *females); Grade Distribution Analysis (Count of different Grades)*. Sessional Result output (Figure 8) shows the performance of a student at the end of a particular academic session, showing such information like name of student, student's matriculation number, department, registered courses for the semester and grades earned. Also, the student's GPA (for each semester) and CGPA for the academic session are automatically computed and displayed as well. Also, a remark will be shown for each semester if the student did not satisfy the minimum credit unit load for a semester. Statement of Result (Figure 9) provides information about a student's earned degree award. This report is usually produced at the end of the student's programme – the student having satisfactorily met all requirements expected of him or her. This report can only be printed if the following conditions are met: student must have satisfied the expected minimum total Credit Units; student must have reached the terminal year of the programme; student must have cleared all previously failed academic courses; there should be no pending score for any academic course registered for the student; the results of all academic courses undertaken must have been approved up to Senate level. These are some of the output the system can provide.

8.0 Conclusion

The system has been thoroughly tested and evaluated, and is certified to be made operational. The performance of the system is seen to meet the design expectations as shown in the results. The system will expedite efficient service delivery in academic records management by eliminating the delays associated with computing results and processing academic documents; enabling management to query the database in a dynamic manner to extract relevant information to aid decision-making; providing an interface between students and the institution. Therefore, we recommend that the university will get more added advantage using ARIS than the present system in use. Nevertheless, new systems usually represent a departure from the way business is currently done; therefore there must be provision for a smooth transition from the old system to the new system.

References

[1.] Ayodele O.A. and Absalom E.E. (2010). "Design and Implementation of Students' Information System for Tertiary Institutions using Neural Networks: An Open Source Approach", *International Journal of Green Computing*, Vol. 1, No. 1, pp. 1-15.

[2.] Bonig R. (2012). "Latest Trends in Student Information Systems: Driven by Competition", *Educause Review*, Vol. 47, No. 3, pp. 56-57.

[3.] Chapman J. R., "Software Development Methodology a.k.a System Development Life Cycle", Available at http://www.hyperthot.com/pm_sdm.htm Retrieved 05.06.2012

[4.] Idogho O.P., Akpado K., and Agajo J. (2011). "Interactive Intranet Portal for Effective Management in Tertiary Institution", *International Journal of Advanced Computer Science and Applications*, Vol. 2, No. 6, pp. 122-129.

[5.] Khan A. I., Quarashi R. J., and Khan U. A. (2011). "A Comprehensive Study of Commonly Practiced Heavy and Light Weight Software Methodologies", *International Journal of Computer Science Issues*, Vol. 8, Issue 4, No. 2, pp.

[6.] Nwachukwu E.O. (1998). Information Systems Development – A Structured Approach, Genesis Concepts, Owerri, Nigeria, pp. 34-36.

[7.] Pienaar H. (2003). "Design and Development of an Academic Portal", *Libri*, Vol. 53, pp. 118 – 129.

[8.] Sorensen R., "A Comparison of Software Development Methodologies" Available At sorenser@software.hill.af.mil Retrieved: 09-07-2012

[9.] Whitten J.L., Bentley L.D., and Dittman K.C. (2001). Systems Analysis and Design Methods, Irvin/McGraw Hill, New York.

[10.] Youh D.X. (2010). "Design and Implementation of a Client Server Distributed Database for Student Results Processing", *The Pacific Journal of Science and Technology*, Vol. 11, No. 2, pp. 288-295.

[11.] "Software Development Methodology Today", http://ptgmedia.pearsoncmg.com/images/01318725 08/ mplechapter/0131872508_ch01.pdf. Retrieved: 09-07-2012

[12.] http://social.msdn.microsoft.com/Forums/en/netfxnetcom/thread/a5c456c8-e432-4b46-b963-3b4559e0a81d Retrieved: 28-11-2012

[13.] http://en.wikipedia.org/w/index.php?title=Student_Information_systems&oldid=491197676 Retrieved: -11-2012

[14.] http://wiki.answers.com/Q/What_is_a_difference_web_based_application_and_windows_based_application#ixzz2ENq3NU7j Retrieved: 28-11-2012

[15.] http://wiki.answers.com/Q/What_is_an_advantage_of_web_based_application_compared_to_non_web-base application#ixzz2ENqNU736 Retrieved: 28-11-2012

[16.] http://wiki.answers.com/Q/What_is_an_advantage_of_a_web_based_application#ixzz2ENqd59Ei Retrieved: 28-11-2012