Development of Multi-Modal Result Processing Software for Tertiary Institution

Oyeyinka I.K.
Computer Technology,
Yaba College of Technolgy, Yaba, Lagos.

Oladipo O.
Center for IT & Mgt.
Yaba College of Technolgy, Yaba, Lagos.

ABSTRACT

Result processing is a major task for any educational institution. In the past, result processing software has been written on different platform such as stand-alone database system, distributed system e.t.c. The era of web based multi-tier system has necessitated that these software be re-engineered. This paper presents the re-engineering of result processing software of the Yaba College of Technology into a multi-modal system. The system analysis of existing software was conducted to know people pains and develop the requirement for the new system. A multimodal, 3-teir (separating the application into data access layer, program logic layer and the user interface) methodology was adopted in re-writing this software. The tools used include visual studio, Borland Delphi and SQL Server database. A modular design approach was adopted with five major modules designed each module serve as an interface for specific user group with assign privileges to ensure people only manipulate data that belongs to them. The implementation produced a robust, highly secured, multimodal system. The system was tested and found to cope with change in policies, unavailability of staff which might be due to industrial strike action, occasional unavailability of network and power failure. It operates in online and offline mode as such it works well and is efficient in managing students' results seamlessly. It has ability for forward and backward integration with student's portal; this has made the software highly efficient as registration data are handled directly by the portal while results are posted on the portal.

General Terms

Software Engineering

Keywords

Multi-modal, result processing, software, tertiary institution.

1. INTRODUCTION

The world today is moving at the speed of light and equivalent accuracy to ensure quality [2]. This has given birth to the involvement of ICT in all facet of the economy today, as it's known to introduce speed with consistent accuracy providing data output of optimum integrity.

The quality of result processing and period of its delivery in tertiary institution is a major determinant in measuring management and administration performance metric and hence should be given serious attention. Tertiary institutions knowing this, establish unit to take care of general data and result processing. Various developers working with these units have come up with various algorithm, models and approaches to automate result processing and delivery to students. As the result produced are used as input for subsequent course registration, pre-requisite courses eligibility, graduation eligibility and it's basically the bases

for student assessment as to know the level of anticipated preparedness. To achieve this, the research exploits the use of a new programming paradigm called multimodal programming for result processing to take care of contingencies that could result in delay of result processing and delivery.

Multimodal programming is one concerned with alternate input methods combined to make a unified implication (Here on the centralized database) [9]. Basically, the essence of this approach is to,

1. Increase usability; where the weakness of one modality is offset by the strengths of another.

2. Enhance accessibility; which makes the design able to cope with unforeseen contingencies such as network availability and malfunction.

The information that is presented via several modalities is merged and refers to various aspect of same process. The use of several modalities for processing exactly same information, as a result provides an increased bandwidth of information transfer [7].

The aim of the research is to design and implement multimodal (multiple input interfaces) result processing software which is flexible enough to cope with periodic changes in academic policies and various staff strength, to produce results devoid of irregularities in ample time. The research objectives include: Provision of a dynamic web client system that provide interface for various users based on the login credential, Provision of custom logs for all write activities, which enhance back trace in case of result fraud, presentation of result in a user friendly interface for moderation and correction during vetting for Head of Departments, coordinators or moderator as case may be and presenting a user friendly score sheet report for management endorsement.

The remaining part of this paper is organized as follow; Section 2 earns out a literature review of existing systems. Section 3 discusses the research methodology. Section 4 discuses the system implementation and the survey carried out to evaluate the performance of the software. Section 5 concludes the research work

2. REVIEW OF EXISTING LITEARTURE

In recent years, post basic institutions have seen academic value of result processing and the need for it to be done internally. Hence, a lot of documentations exist in the line of research. [5] Presented the Design and Implementation of a Client Server Distributed Database for Student Results Processing. He proposed a relational database designed in a way that each academic department in the university has its

own database including the Central Record Processing Unit (CRPU), Exams and Record Unit, Student Affair Division, Dean's Offices, and Senate. The master database is hosted at CRPU. Microsoft Visual Basic 6.0®and Structured Query Language (SQL) were used to design a prototype of a client server distributed database for processing student records.

Work on Object-Oriented Software Model for Students' Registration and Examination Result Processing in Nigerian Tertiary Institutions was presented in [1]. They presentment application was an instance of a well modeled software system to testify to the beauty, power and supremacy of Object-Oriented paradigm. They further used software predictor metrics to indicate that the software developed is reliable and elegant.

A software application for university students' results processing was presented in [10]. The system was developed to facilitate the automated processing of the results. The software was developed using Java programming language and MYSQL Relational Database Management System. With it, it was possible to compute Grade Point Average and Cumulative Grade Point Average for each student based on examination scores entered.

3. RESEARCH METHODOLOGY

This research was carried out at the Center for Information technology and Management (CITM), Yaba College of technology, Yaba, Lagos. During the system analysis, various complaints about the existing system were collated. It was discovered that stakeholders and users' opinion and requirement differs with different cadres. Stakeholders in this project are people directly or indirectly affected by the performance of the result processing software example include the students, management, parents and the likes. Some of the issues collated were:

1. Industrial action results to unnecessary delay of students' result and management tend to incur heavy overhead to pay staff to work during industrial actions. (Management's opinion, students and parents opinion)

2. Delayed report. Such as, Number of references for each course for the entire school, eligibility status reports which indicates the status of each student (Active, Expelled, Repeating, Withdrawn, Stepdown and reason and differed). Since, school programmers needed to collate from different databases in each schools/faculty. (Management's and CITM staff's opinion)

3. Un-uniform results as output were from stand alone systems which could easily be re-programmed. (management's opinion)

4. Lack of adequate monitoring and logging system. (Management's and CITM) staff's opinion)

5. Lack of database and application security. Access to the application automatically guarantees access to the database table, which makes them vulnerable. (CITM staff's opinion)

6. Isolated data scheme as students registrations are re-entered from nominal row printed-out by school officers. (Data entry staffs opinion)

7. Obsolete database technology and approach. The former system uses the Ms assess database which is a database known with indexing error as data grows. (CITM staff's opinion)

8. Distributed standard; which could lead to inconsistency and logic error. Since there is no centralize logic, each of the software installation instances can be configured to any output, resulting in un-uniform result sheet presentation. (CITM staff's and management opinion)

9. Raw scores and final score mismatch, due to typographical error or fraud. (Lecturer's opinion).

In the bid to ensure all the opinion are captured and cared for, the application software developed implements a multiple redundant interface-module structure, to ensure the unavailability of some staff groups would not hamper result processing exercise. It further, implements a centralized logic and database system as the application to be accessed by every member of staff involved in the exercise is hosted on a server. The database was made to keep logs of every write transaction made and change in scores could be tracked from various interfaces-module ranging from course lecturer itself to the Head of Departments (HOD) (The information accessible from various interface-module depend on the privilege and the jurisdiction of the users).

During the system design, various templates were developed to describe the operation carried out by different module. Each module is the entire application span accessible to a user group. The available modules were:

1. Lecturers' web module

2. Programmers' web module

3. Data entry staff web module

4. HOD/ Coordinators' web vetting module

5. Class advisers' web module.

The programmers in this context are the CITM representative in each school. They are responsible for result processing exercise and other CITM activities in the schools. CITM is the unit of the institution that has the prerogative to develop, maintain software application, train various users' to use developed application, generate management report from applications, support applications developed, maintain computers and associated hardware in the school.

The application was further developed on two platforms to ensure security. The web application part and the windows desktop application part. The web part is designed to ensure easy access by every member of staff involved in result processing and the windows desktop application is made to run on specific users' system to ensure restricted access. The windows application serves as the management console of the multimodal result processing software. It has the following functions

- Create new staff and other users

- Enforce disciplinary actions such as expel, step down and repeat

- View profile information and reset password

- Change grading policies

- Trace log on changes made to results.

- Upload specific result to online server

- Print all report sheets

- Manipulate courses on course dashboard

- Work on student's registration on student dashboard.

It should be noted that these functions were made available based on login parameters and privilege levels. Regular school programmers are restricted to these functions for their school/faculty alone, while super administrator/programmers have access to all functions. Other users' group have access to report sides only in their respective jurisdictions. This is shown in Figure 1 use case diagram.
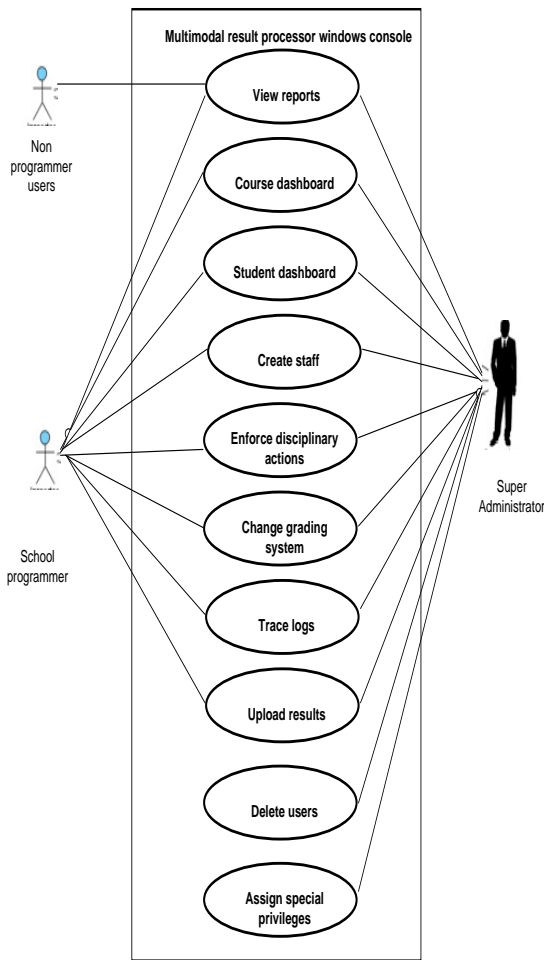
Fig 1: Use case diagram for the desktop windows application part

The Lecturers' Web Module allows authenticated lecturers to view courses allocated to him/her on a departmental basis and enter scores for each student per course. The scores were by default initialized to '−1' which indicates that the student has not been scored for the registered course. In addition, the module has a special support for offline input; it allows user to upload scores from Ms-excel file which can be done off-line in the absence of network facility. Figure 2 is a UML activity diagram that explains the system design for the lecturers' web module. The programmer module is a supplementary modality that takes care of lecturers that could

not key in their results before deadlines. The results are obtained in bulk and transferred to excel sheet. Here, results in different courses for various students can be collated into excel sheet and uploaded to the result portal. The record arrangement in the excel sheet could be in two format as shown in Table 1 and Table 2 respectively. N there is the total number of courses which is dynamic since the dataset is populated at run-time based on courses registered by students in the specified class. These templates are generated in the application from the programmer's module. The ultimate duty of a school programmer is to process different classes' results; this is when result's Grade Point Average , Cumulative GPA, Total Number of Unit, Weighted Grade Points, highest scorers and number of references (if any) was generated. The UML activity diagram that shows the developed prototype is shown in Figure 3.

Table 1. Excel template to collate different scores in diverse courses for various students

Matriculation Number	Course code 1	Course score1	..	Course Code N	Course Score N

Table 2. Excel template 2 to collate different scores in diverse courses for various students

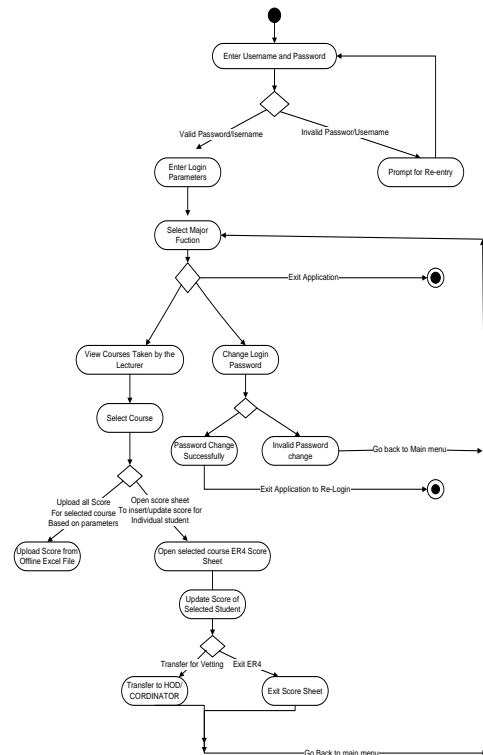| Matriculation Number	Course code 1	Course Code 2	..	Course Code N

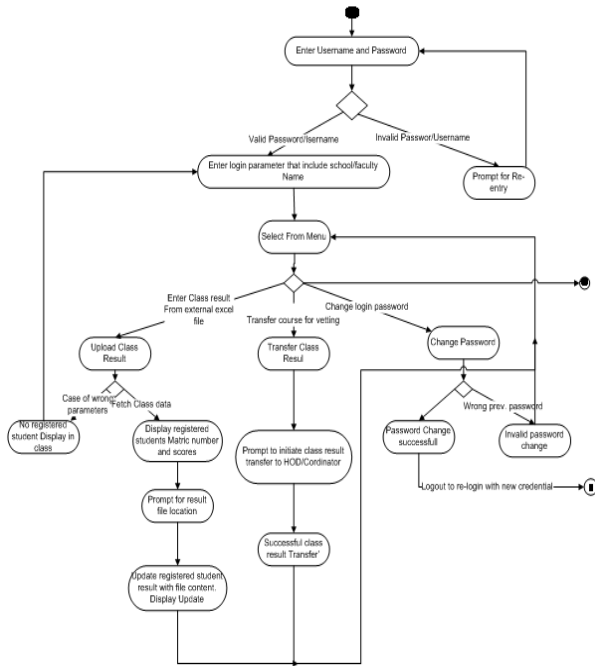**Fig 2: UML activity diagram for lecturer's module**

Fig 3: Activity diagram for programmer's module

HOD/Coordinator's web module is a web based vetting module that allows Head of Departments or Coordinators to vet result during moderation. This allows per-courses vetting and a general course vetting mode. The prototype for the designed is shown in Figure 4. Data entry staff web module creates a per-student view of results. Individual matriculation number is inputted to fetch corresponding scores. Cases where results were not correctly inputted for a student can amend and scores can be inputted afresh too.
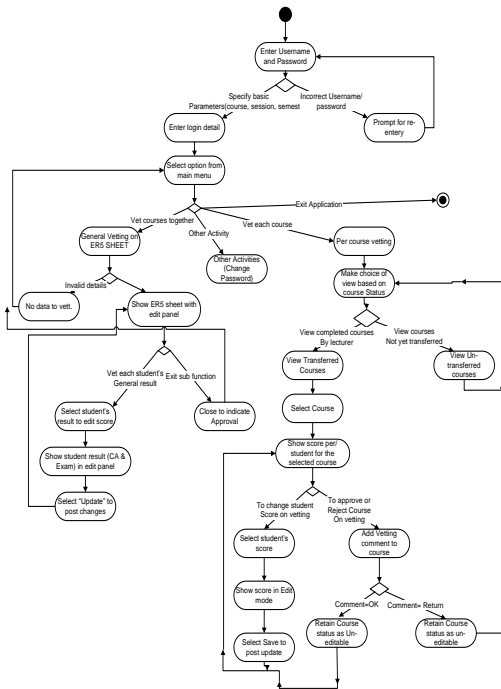
Fig 4: HOD/Coordinator's vetting module prototype

The class adviser is an academic staff that attends to results and registration of students in a particular class. A class

adviser can collate class scores into an excel sheet as shown in sample Table 2 and upload from his interface. The application was made to be functional when it online and offline. The offline mode is demonstrated in the use of excel template off-connection with the application server. Various excel template downloaded could be completed at users' convenience and loaded back to the application whenever users have access to the local network. The application server is mounted in the server room which was powered by an online inverter that can go more than 48 hours without power supply this is to ensure that the application is always available for use. Even when there is power outage users can still connect with the application on their laptop computers.

To build the application the software tools used include

- Visual studio 2008

- Borland Delphi 6

- Microsoft Sql server 2008 Database

- IIS Web server

The web part was implemented in a visual studio environment using ASP.NET with VB.NET and the windows desktop application part was implemented using Visual Pascal (Borland Delphi) in Delphi 6.0 environment. Delphi was the language of choice in implementing the windows application due to its ease of data access using various data aware component, flexible quick report and the easily customizable grid component to render data on forms [6]. The application uses the state of art database designed by Microsoft (Microsoft SQL Server 2008) because of it stability even as data grows in the database, it supports for many database transactions and some other features which include ease of installation, better performance, built in encryption and better security [3][5]. The web hosting and delivery used the Microsoft Internet Information Service (IIS) because of it compatibility with Microsoft ASP.net [8].

The database implemented has twelve (12) user's tables; Eight (8) of them are directly involved in result data manipulation and processing. These are the published registration table (named as pubreg), the elaborated registration view table (named as er5view), the elaborated view summary (named as er5summary) the departments table (named as deptTable), the course detail table (named as cosTable) and the step down students table (named as steptab), the student history table (named as studhistory) and the log detail table (named as the logtab). The database entity relationship (ER) diagram of the basic tables is as shown in Figure 5. Two (2) of the tables are for user's authentication and profiling. These are the staff identification table (named as staffed) and the staff detail table (named as stafftab) the ER diagram is shown in Figure 6. The last two (2) are functional tables as shown in Figure 7.

The application developed was rolled out and tested by the school programmers to rectify before a full scale deployment for the entire school community to process students' results. Before making different modules available for users group, trainings were conducted to intimate users with different feature available on the software. Subsequently, feedback was obtained from different users to make continuous updates and ensure software served users well and efficiently. The software is also continually updated to ensure it properly optimized and scaled to accommodate more users and users group. The summary of the SDLC is shown in Figure 8.
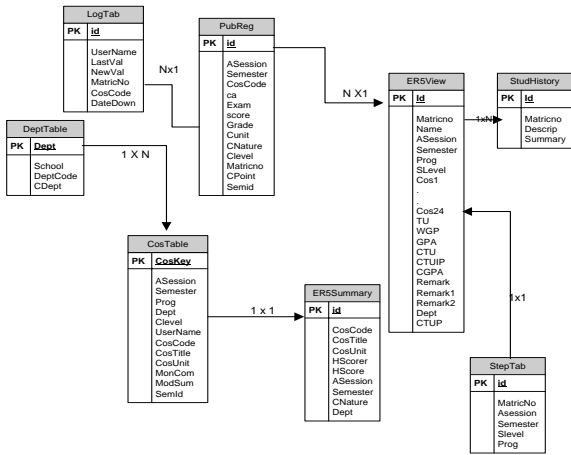
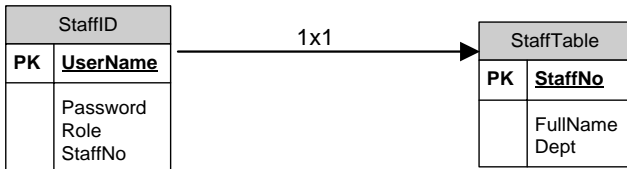**Fig 5: Database Entity relationships diagram of basic tables**

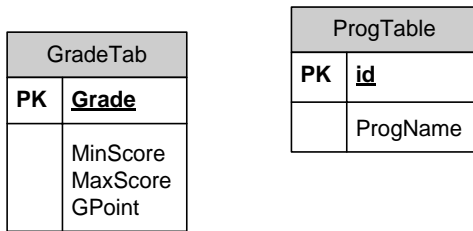**Fig 6: Database Entity relationship of user based tables**

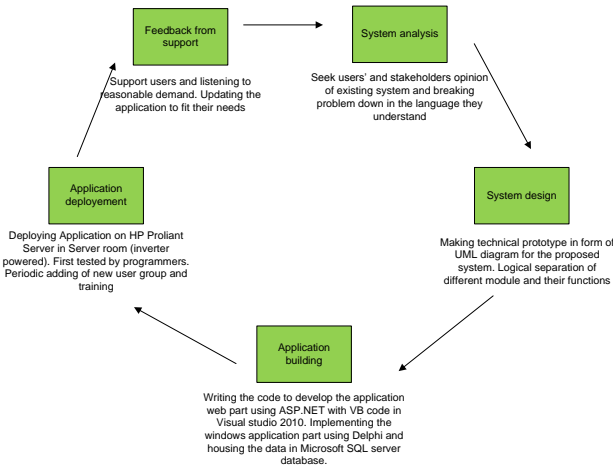**Fig 7: Other functional tables**

Fig 8: Application SLDC

4. IMPLEMENTATION

Figure 9-11 shows various interfaces while running the multimodal result processing software developed. Figure 9 shows the login page to the result portal. Every user is given a user name and password that uniquely identifies them and attached to this is the privilege level that determines what and how much of data the user sees. An HOD has access to all classes in a department while a class adviser has access to a class in a department. Figure 10 shows an opening dialog box

to open an excel file to read from and in-turn upload the extracted data to the selected course from a list of courses coordinated by the lecturer. Figure 11 shows an interface that displays scores imported from an excel sheet by a school programmer. Figure 12 shows a vetting sheet and figure 13 shows a printable result sheet, to be printed for management's approval.
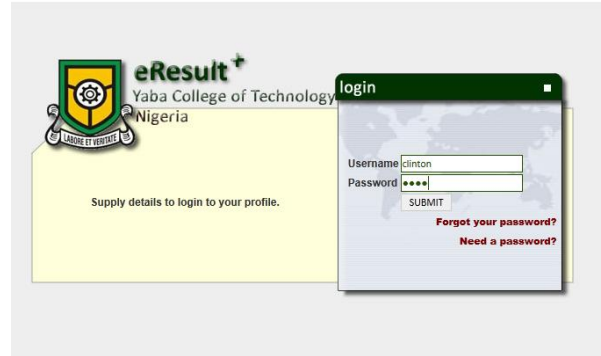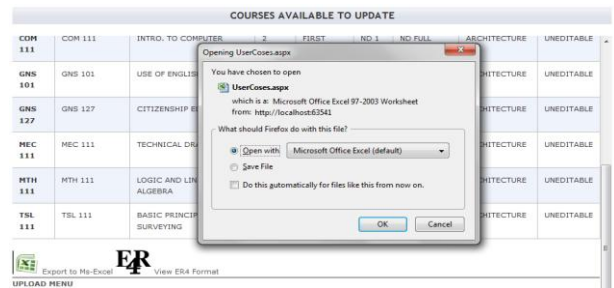

Fig 9: Web base part login page.

Fig 10: Prompt to open excel file and extract content

Fig 11: Programmers module displaying extracted scores

Fig 12: Vetting Sheet

Fig 13: Printable result.

5. SOFTWARE TESTING AND SURVEY

The research survey was carried out at the Yaba college of Technology, Yaba, Lagos, to evaluate the performance of the developed system. A scale to rate the application performance on a scale 1-5 (poor to excellent) based on user friendliness of the application, ease of generating and printing report, availability and ease of adapting to change in policies. 90 questionnaires were distributed (35 to class advisers, 18 to school programmers, 25 to HODs and 12 data entry staffs). The summary of respondents is shown in Table 3. The survey showed that the application developed is user friendly, efficient in generating report and adapt easily to changes in policies. This is shown in Table 3 as all the users rated the application performance 80% and above, based on the aforementioned factors. 7.78% of the respondent indicated that the application is only 60% availability this could be due to hitch on the school network which was later resolved.

Table 3. Summary of respondents

Factors queried	Respondents rating on scale from 1-5				
	1	**2**	**3**	**4**	**5**
User friendliness	0	0	0	67	23
Ease report generation	0	0	0	14	76
Software availability	0	0	7	52	31
Adaptation to change in policy	0	0	0	55	35

6. CONCLUSION

The result processing software developed in this research work is a robust, highly secured, user friendly, multi-modal computer application that has the ability to generate various reports on the fly. The output from this software is uniform regardless of when and where it is generated since the application logic is centralized. It also employ the three tier object oriented programming model (with separated data access layer, program logic and user interface) to ensure easy handing-over or continuation of development by another programmer. The application was built to cope with irregularities as a result of unforeseen change in policy such as that of Nigeria Board of Technical Education (NBTE). The application has Online and offline support.

The seamlessness advantage of the application lies in its ability to plug-in to fetch input from the student portal. Hence, information in the result processing software is same as that registered by the students themselves during registration. Real time monitoring feature of the application makes it log prove, every score changed is logged to the database and can be traced in a user friendly way. Ultimately, this software is very efficient in synchronizing locally processed result with available results online for students to check and printout a statement of academic performance for different semester. The usage of this software has greatly improved the result processing and administration in Yaba College of technology.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] Bamigbola, O.M., Olugbara, O.O., and Daramola, J.O. 2010. An Object-Oriented Software Model for Students' Registration and Examination Result Processing in Nigerian Tertiary Institutions.

[2] Bertucci, G. and Alberti, A. 2003. Globalization and the Role of the State: Challenges and Perspectives

[3] Boone, K. 2011. Microsoft SQL Server advantage.

[4] Djam, X.Y. 2011. Design and Implementation of a Client Server Distributed Database for Student Results Processing.

[5] Fernando, A.P. 2008. SQL Server 2008 Full-Text Search: Internals and Enhancements.

[6] Marco, C. 2001 Mastering Delphi 6. Sybex.

[7] Oviatt, S. 2002. Multimodal interfaces. , J. Jacko. Sears. 2 (Mar. 2002), 677-697.

[8] Schaefer, K. 2011. Core Server, Professional IIS 7. John Wiley and sons.

[9] Sarter, N.B. 2006. Multimodal information presentation: Design guidance and research challenges.

[10] Ukem, E. O. and Ofoegbu, F.A 2012. A Software Application for University Students Results Processing.

[11] Vitense, H.S., Jacko, J.A., and Emery, V.K. 2002. Multimodal feedback: establishing a performance baseline for improved access by individuals with visual impairments. In Proceedings of ACM Conf. on Assistive Technologies