



A SOFTWARE APPLICATION FOR UNIVERSITY STUDENTS RESULTS PROCESSING

¹EYO O. UKEM, ²FRANCIS A. OFOEGBU

^{1,2}Department of Physics, University of Calabar, Calabar, Nigeria.

¹E-mail: ukemeo@yahoo.com

ABSTRACT

The end-of-course grades assigned by lecturers are intended to convey the level of achievement of each student in the class. These grades are used to make a multitude of decisions. Unless a sufficiently accurate and efficient method is used for the grading, the grades are apt to convey misinformation and lead the decision-maker astray. In itself, the processing of results is found to be rather tedious, especially when carried out manually, and when the number of students is large. It is time-consuming and error prone. The process, however, becomes a lot easier and much more accurate when carried out with a computer running a suitable software application. To find a sufficiently fast and accurate method of carrying out this processing is a problem that is both challenging and interesting. In this work, a computer software application was developed to facilitate the automated processing of the results. The software was developed in Java programming language in the form of a database, employing MYSQL Relational Database Management System. The developed software performed well and produced expected results on completion. With it, it was possible to compute Grade Point Average and Cumulative Grade Point Average for each student based on examination scores entered.

Keywords: *Examination Scores, Automated Processing, Cumulative Grade Point Average, Software Application, Results Grading, Database.*

1. INTRODUCTION

The errors associated with the existing manual method of processing of students results in most universities in Nigeria, including the University of Calabar, make it not only desirable but imperative that computerized approach be used to the full in measuring students' progress. The manual methods being employed suffer a number of set-backs. They make the process to be time-consuming and prone to error. They lead to examination results being published late, sometimes with wrong grades being entered and students' grade point averages being wrongly computed as a result, and ultimately leading to wrong conclusions being arrived at on class of degree awarded. Some students could end up with undeserved good class of degree, while others could be unfairly victimized, bringing about frustration and bad blood. The Departments concerned, and the whole University, could become tarnished. The problem, therefore, that arises is to find a method of processing examination results that would be sufficiently accurate and reasonably timely. The software

application in this article is intended to bring relieve by providing for timely and accurate processing of students results using the processing power of the computer. The rest of the article discusses the background to this work, the materials and method used in the work, and the results recorded. The article ends with a conclusion.

2. BACKGROUND

At present the students grading system used by all universities in Nigeria is the Five-Point Grading System, established by the Nigerian Universities Commission (NUC) in 1989. The system is also called the "Carry Over" system, because a student is allowed to carry over a course up to two times (or attempt a course a maximum of three times) before a final "Fail" grade is recorded if he/she still does not pass the course. A few exceptions to the rule are the General Studies and Communication Skills (or GSS) courses, which must be passed for a student to graduate. The NUC five-point grading system [1] is shown in Table 1 below.



Table 1: The NUC Five-Point Grading System

SCORE (%)	LETTER GRADE	GRADE VALUE	REMARK
70-100	A	5	Excellent
60-69	B	4	Very Good
50-59	C	3	Good
45-49	D	2	Average
40-44	E	1	Pass
00-39	F	0	Fail

The courses offered in a degree programme are allocated a number of credit hours which vary from one course to another, because the courses vary in their needs and scope. Hence some are allocated greater credit hours than others. The measure of performance of a student in any course is given by the grade-points obtained in that course. The grade-points obtained by a student in

any course are determined by multiplying the value of the grade (numeric grade) by the credit hours of the course. The total grade-points are obtained by summing up the grade-points of all the courses offered. The Grade-Point Average (GPA) is computed by dividing total grade-points by the sum of credit hours of all the courses offered in that period. Thus:

$$\text{GPA} = \frac{\text{Total Grade-Points of the courses offered in the year}}{\text{Summation of the credit hours of the courses in that year}}$$

The table below (Table 2) illustrates how a student's GPA may be calculated manually. If a student obtains the grades as shown at the end of,

say, Year One, the computation of GPA is as shown.

Table 2: Calculation of GPA

Course Code	Credit Hours	Scores	Grade	Grade-Points
GSS 1101	2	42	E	1 x 2 = 02
GSS 1131	2	60	B	4 x 2 = 08
PHY 1111	3	46	D	2 x 3 = 06
PHY 1141	3	30	F	0 x 3 = 00
PHY 1180	2	75	A	5 x 2 = 10
MTH 1111	3	50	C	3 x 3 = 09
CHM 1011	3	70	A	5 x 3 = 15
GSS 1102	2	58	C	3 x 2 = 06
GSS 1112	2	45	D	2 x 2 = 04
GSS 1122	2	41	E	1 x 2 = 02
PHY 1152	3	68	B	4 x 3 = 12
MTH 1352	3	80	A	5 x 3 = 15
CHM 1022	3	67	B	4 x 3 = 12
	33			101

$$\begin{aligned} \text{Total Grade-Points} &= 101 \\ \text{Total Credit Hours} &= 33 \\ \text{Grade Point Average (GPA)} &= 101/33 \\ &= 3.06 \end{aligned}$$

The Cumulative Grade-Point Average (CGPA) in any year is obtained by dividing the cumulative sum of the total grade-points over the years by the cumulative sum of the credit hours over the same period. Thus,



$$\text{CGPA} = \frac{\text{Cumulative sum of Total Grade-Points of the courses from Year One to the Present Year}}{\text{Cumulative sum of the Credit Hours of the courses from Year One to the Present Year}}$$

The CGPA is a very important measure, as this is what determines whether a student can move on to the next level or be made to repeat a year or even withdraw totally from the programme. The final CGPA determines the class of degree awarded to the candidate on eventual completion of the programme. Obviously, for the first year, the CGPA is equal to the GPA.

Manually calculating and tracking CGPA for each student is rather laborious, especially with increasing number of students, and is prone to error. With a computer the task becomes much easier, faster, and more accurate. All that is needed is a suitable software application to run on the hardware, and a solution is established. With the appropriate software in place, the system would simply capture raw scores as entered by individual lecturers for various students in the different courses, and then process accordingly.

3. LITERATURE REVIEW

Some previous work has actually been carried out in this area. Two of such are the work by Moses Ekpenyong [2] and that by Ukem and Onoyom-Ita [3]. In his work, Ekpenyong employs Microsoft Excel spreadsheet program to build an Intelligent Knowledge-Based System (IKBS), making use of various programming facilities provided by that application (Excel). The central issue here is that the programming is hard coded into the cells, and cell referencing is used to monitor and track students performance (cumulative points, etc.). The system has been reported to be working fine. However, it appears to be rather restrictive, and calls for substantial expertise in programming.

In the work by Ukem and Onoyom-Ita, Adobe Dreamweaver, an Integrated Development Environment, is used to create the Graphic User Interface and to write the codes. MYSQL Server, a Relational Database Management System, is used to create the database tables and data. Personal Home Page Pre-Processor (PHP) is used to communicate with and manipulate the database. This application, though tested and found to be working as expected, has however not been put to use widely.

Apart from these two, there are undoubtedly several other similar applications in

existence. There is, however, always room for improvement. This new application is intended to have reduced complexity and greater ease of use, in order to enhance maintainability while still retaining good speed and accuracy.

4. MATERIALS AND METHODS

The computer software application is required to be independent of any platform. It is desired to have three sessions, namely: the "Super Administrator" session, the "Staff/Administrator" session, and the "Staff" session. The Head of Department (HOD) should have the authentication/permission of the Super Administrator. He should be able to perform the following functions.

- Register students in individual units of the Department
- Register staff in the Department
- Register courses offered in the Department
- Assign courses to registered staff in the Department
- Assign examination officers to individual units
- And also perform the role of a staff

The Examinations Officer of a unit should have the authentication of the Staff/Administrator. He/she needs to be a registered Staff, and thus have a valid username and a password. He should be able to perform the following functions:

- Enter students scores and view students' grades in the courses he coordinates
- Process students results in his Unit, which includes calculating the GPA and CGPA
- View all the students' results in his unit

A Course Coordinator should have the authentication of the Staff. Hence, he should be registered by the Super Administrator, should have a valid username and password. He should be able to perform the following functions:

- Enter students' scores for courses he coordinates
- View the grades he entered

These details enumerated above, along with other details of the application, are shown in the architectural design of the application [4], which is illustrated in Fig.1.

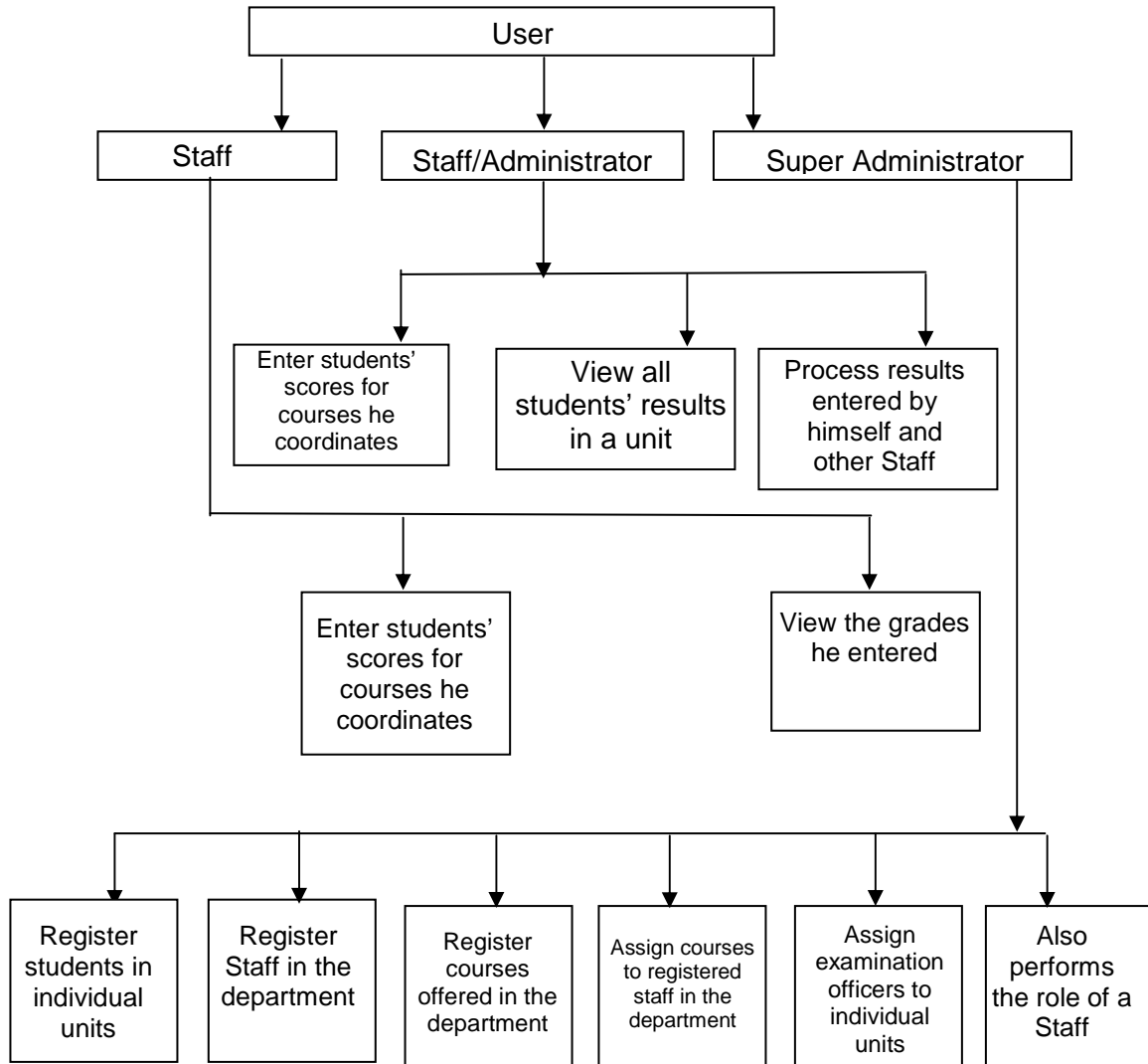


Fig. 1: The Architectural Design of the Software Application

The application was developed in the form of a database, using the Waterfall software life-cycle model. The Waterfall software life-cycle model is called the traditional model because it was the first widely used software development life cycle. It is part of Structured Software Engineering or the Structured Paradigm, which is the older of the two earliest approaches to formal software engineering [5]. (The other approach is Object-Oriented Software Engineering). The decision to implement the application in the form of a database was informed by the consideration that various types of data would need to be held, and a database approach would be more appropriate due to the advantages that the database file system has over other forms of file systems.

Such advantages include data consistency, data integration, data sharing, data independence, data control and minimal data redundancy [6].

The database system used in this work to create the database tables and data was MYSQL, a Relational Database Management System (RDBMS). A relational database management system is a database management system in which data is stored in the form of tables and the relationship among the data is also stored in the form of tables. Each table has multiple columns and rows. The Relational system has been the most commonly used of all databases, and has various features that make it simple yet elegant [7]. MYSQL is an acronym for My Structured Query Language. It is a relational database management

system computer language designed for querying, retrieving, modification, and management of data in relational database management systems. MySQL is very fast, reliable, and easy to use, and its connectivity, speed, and security make it highly suited for accessing databases [8].

The entire application was developed in Java. Java is a programming language used to build programs that can work on stand-alone computers and on the internet. Its primary features are that it is object-oriented and a cross platform language [9]. By cross platform, it means that the programs can run across several platforms such as Microsoft Windows, Apple Macintosh, Linux, and

so on. Java is not only used for stand-alone applications and Net based programs but also used to create consumer devices and accessories programs such as cellular phones, Palm pilots and other gadgets. Java is different from most high-level programming languages in that it is not translated into a machine-specific code; instead Java is translated into an intermediate-level language called Java Byte code. This translation is done by Java compiler called Javac [10].

Coding of the application produced over 4,000 lines of code. The flow of the program logic is shown in Fig.2.

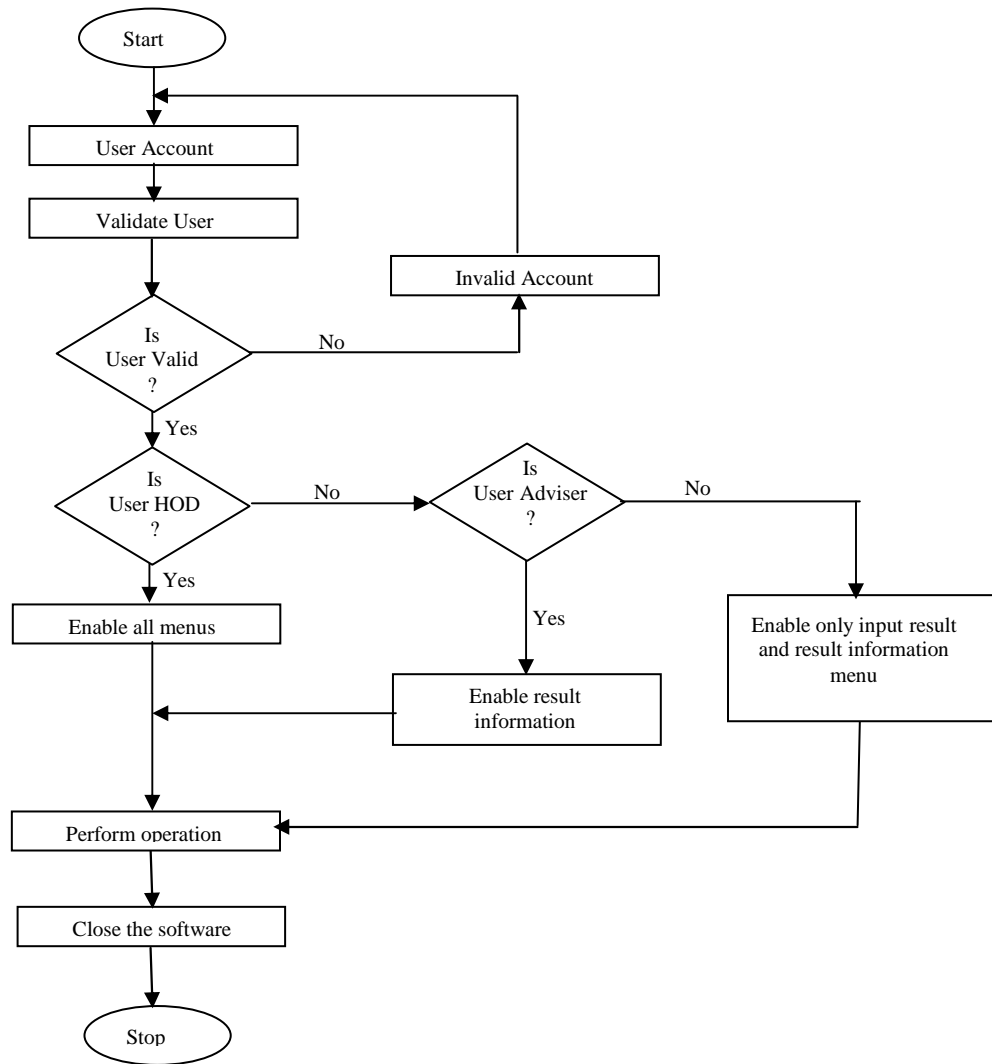


Fig.2: The program's flow chart

The system is designed and implemented such that the following are carried out during its use:

User validation: To be able to use the software, staff are to be registered by the HOD with a default username and password on the first login to the software.

Students Registration: Students in the Department are to be registered on the system.

Course Registration: Courses offered from first year through final year should be registered.

Course Assignment: After course registration, staff are assigned courses that they will teach.

Usage: At the end of the period (semester or session) staff will login to the software and enter students' marks for any course they are assigned. Staff can also view results already submitted, if they want, or change their password when desired,

while academic advisers will have the authentication to process students' results (i.e. calculate GPAs and CGPAs) for their respective units for any academic session they choose. An academic adviser can also view any student's GPA and CGPA in his unit.

Finally, HOD can view all staff, students and course registration, as well as be able to view any student's result irrespective of the student's unit.

5. RESULTS AND DISCUSSIONS

The developed software application was run on the system and found to operate as expected. The login screen is as shown in Fig. 3.

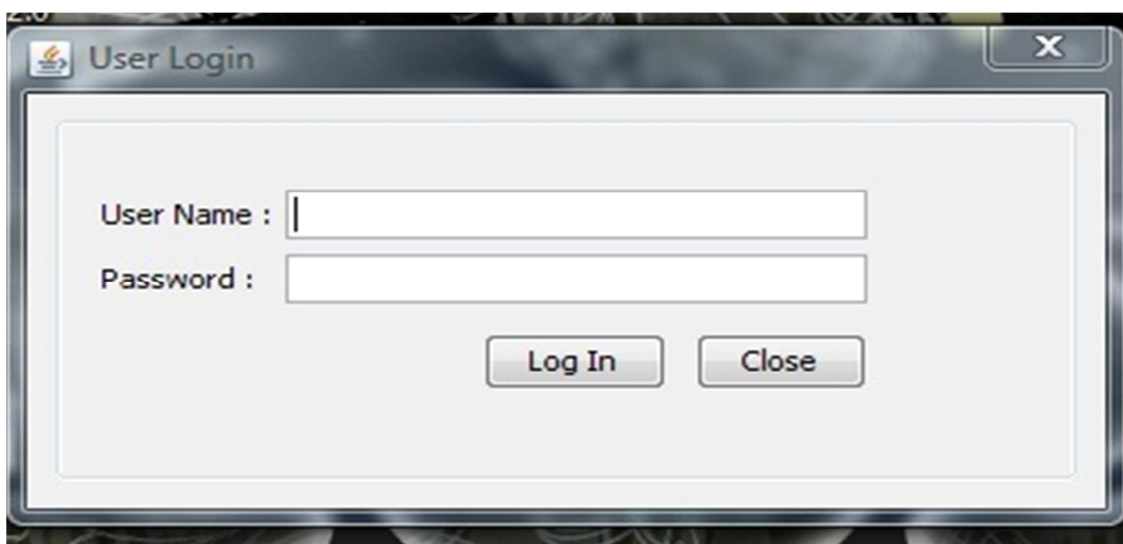


Fig. 3: Log in Page (Validates user account before access is granted)

Once the user is able to log in, the main window appears. The main window has three menus, namely; **Result Computation** Menu, **Administration** Menu, and **Help** Menu. Within the Result Computation and Administration menus, there are sub-menus that appear once they are clicked. For example, when Result Computation is clicked, sub-menus such as "Record Result", "View Result", "Process Result", "GPAs", and "Exit" will appear. When Administration is clicked, the sub-menus "Student Registration", "Change Account", "Staff

Registration", "Course Registration", "Assign Courses", and "All Views" appear. Once a sub-menu is clicked, a frame appears at the centre of the window. This is where the main action is performed. A sample screen, the Students Registration Window, is shown in Fig. 4. Similar windows exist for staff and student registration. Drop down menus are provided to allow for easy selection of items in some cases, while in others, radio buttons are used. Also available are some text fields. These do not have drop down menus and desired text needs to be typed in.

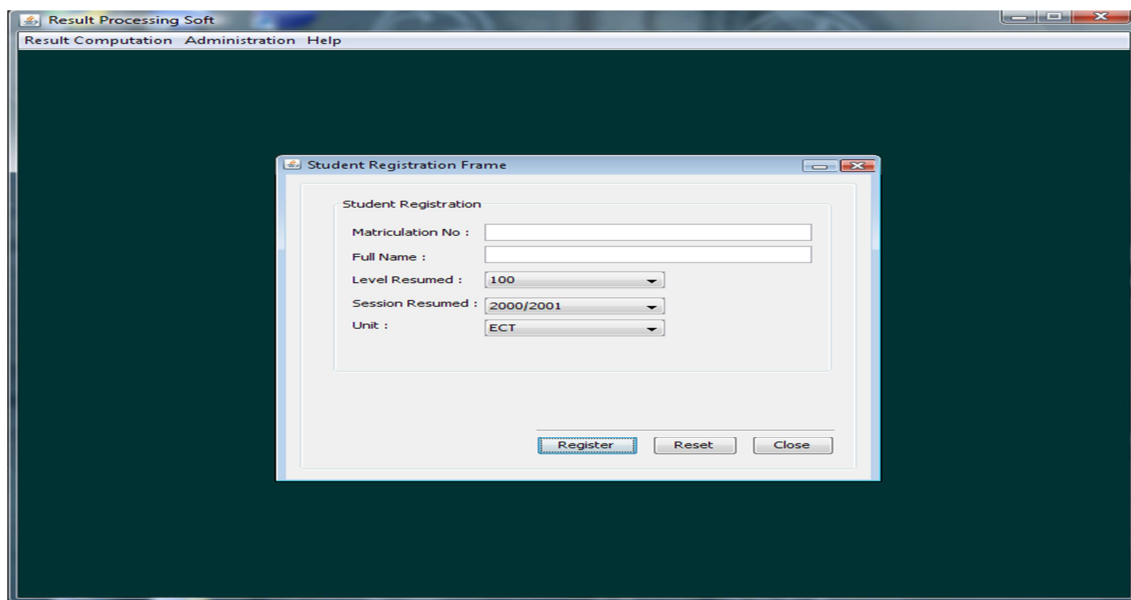


Fig.4: Students Registration Window (For registering all the students in the Department)

The outputs from the system are various. They include a listing of registered staff, registered courses, and registered students. Processed results can be viewed for one student at a time, or for all the students in a Unit. A sample output is shown in

Figure 5. This shows the result for a single student in all the courses at 100-level (Year 1) for one academic session. The result also shows the GPA, and since this is the first year, there is no CGPA as such.

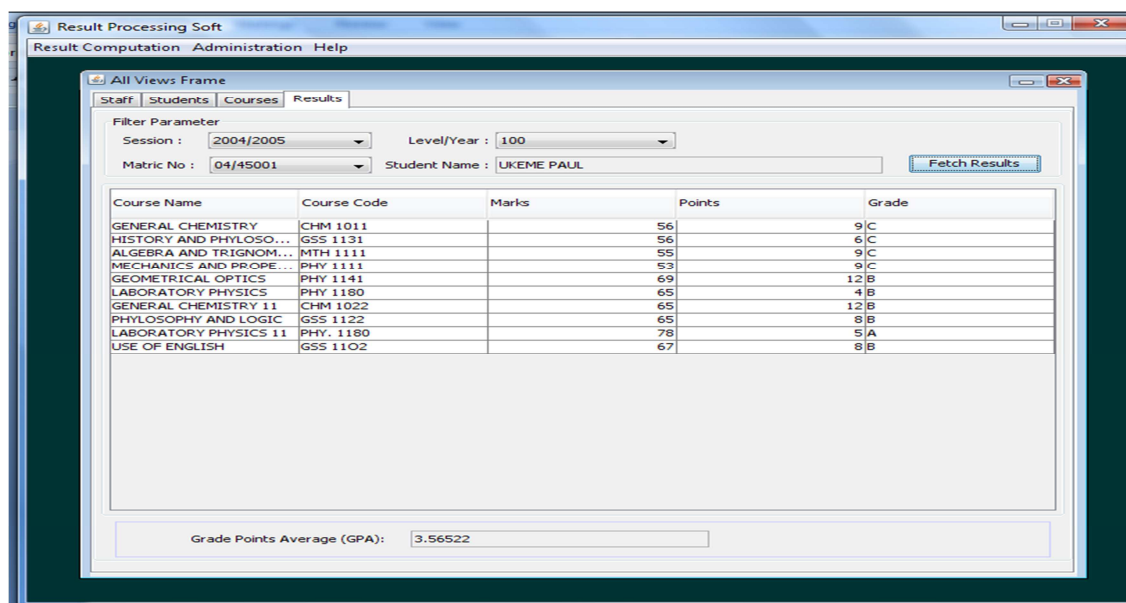
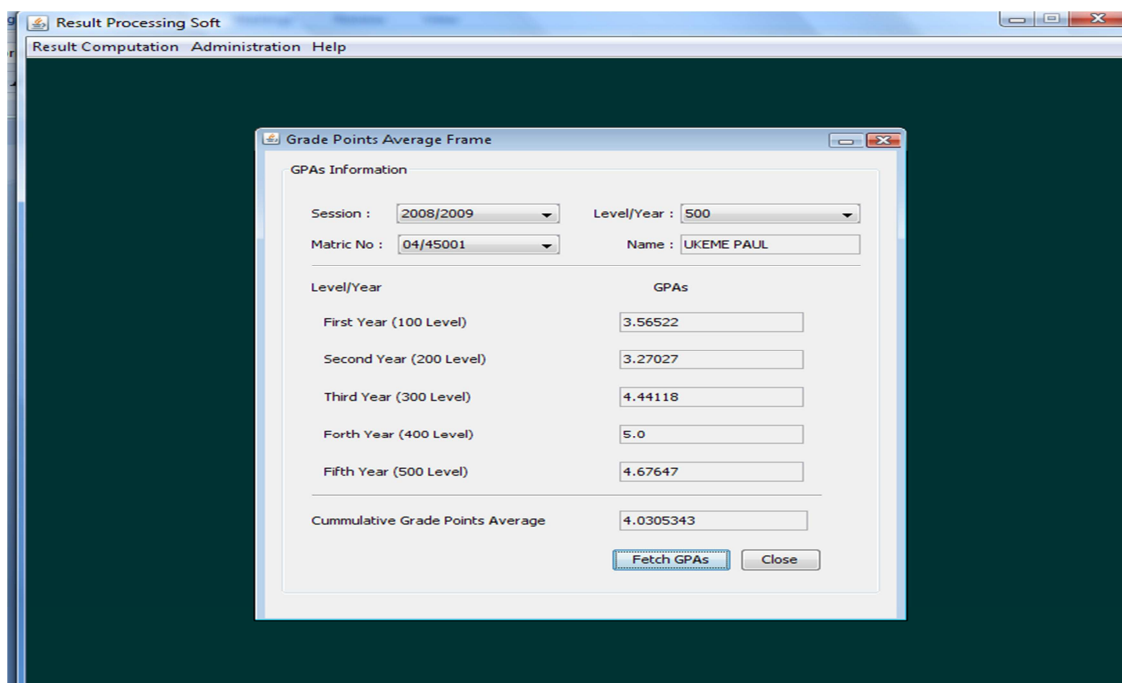


Fig.5: Student's Results Window (For checking student's results)

Figure 6 shows the summary of the results of a student over an entire five-year period (for a five-year program, for example). The GPA for each

year (level) is displayed, with a final CGPA at the end, which determines the class of degree to be awarded.



The screenshot shows a software window titled "Grade Points Average Frame" with a menu bar containing "Result Computation", "Administration", and "Help". The window contains the following information:

GPA Information	
Session :	2008/2009
Level/Year :	500
Matric No :	04/45001
Name :	UKEME PAUL
Level/Year	GPA
First Year (100 Level)	3.56522
Second Year (200 Level)	3.27027
Third Year (300 Level)	4.44118
Forth Year (400 Level)	5.0
Fifth Year (500 Level)	4.67647
Cummulative Grade Points Average	4.0305343

At the bottom of the window, there are two buttons: "Fetch GPAs" and "Close".

Fig.6: GPA and CGPA window (For Exam Officers only)

The application provides a simple and convenient approach to the processing of students' results, including GPAs and CGPAs. It is reasonably secure because, due to password protection, no unauthorized person can gain access to alter the data. The information obtained from the system has a reasonably high degree of accuracy, because all the computations are automated. Any errors found would probably be those introduced by human error in the keying in of the raw scores. Computations are carried out very speedily by the system, once all raw data has been entered, and required information is available almost immediately.

This application suffers a setback where power supply is not reliable, such as in underdeveloped and even developing countries. Without adequate power supply, the system simply cannot be used, and the benefits cannot be enjoyed. Another issue that may be of concern is the security. In its present form the application does not have rigorous checks built in. This could

be improved by having sound backup policies in place, and the introduction of audit trail.

6. CONCLUSION

This paper has presented a software application meant to ease the processing of students results in a tertiary institution. The application was successfully developed, tested, and found to be working as expected. It is capable of storing and processing students' results with high speed and accuracy, and presenting output in certain required forms. It is easy to use due to the use of a GUI rather than command-line approach, reasonably secure, and enforces data integrity resulting from the use of a relational database management system. With this application, the processing of students' results can be automated to a large extent, thereby reducing processing time and increasing accuracy.

The future outlook of this work is to make the application web-enabled, which would greatly enhance its use. Users could then access



the application from anywhere via the Internet, and be able to carry out their work, and students with proper authorization would be able to view their results on-line. The issue of security could also be looked into, with a view to improving it so that users would have greater peace of mind, knowing that their data would not easily be compromised.

ACKNOWLEDGEMENTS

This work was facilitated by the Electronics and Computer Technology Unit of the Department of Physics, University of Calabar, whom we acknowledge accordingly. We also wish to acknowledge, with thanks, the contributions of "Chibex".

REFERENCES

- [1] Kola Adeyemi and Austine Osunde. "An Assessment of the Academic Achievement of Students in Two Modes of Part-Time Programme in Nigeria" in *The International Review of Research in Open and Distance Learning*, July 2005.
- [2] Moses E. Ekpenyong. "A Real-Time IKBS for Students Results Computation", *International Journal of Physical Sciences (Ultra Scientist of Physical Sciences)* Volume 20, Number 3(M), September – December, 2008.
- [3] E. O. Ukem and E. O. Onoyom-Ita. "A Software Application For The Processing Of Students Results", *Global Journal of Pure and Applied Sciences* Volume 17 No. 4 (2011)
- [4] F. A. Ofoegbu. "Development of a Computer Programme to Calculate Students GPA and CGPA Using Java". A final year project in the Department of Physics, University of Calabar, Nigeria, 2009.
- [5] G. Curtis and G. Longworth. *Guide to Software Engineering Paradigm*, (1998).
- [6] Tata (1995). *Database Management Systems*. Tata Consultancy Services, Nariman Point, Bombay, India.
- [7] Brian K. Williams and Stacey C. Sawyer. *Using Information Technology: A*

Practical Introduction To Computers & Communications. McGraw-Hill Irwin, Montreal, 2003.

- [8] <http://www.mysql.com>
- [9] <http://www.javaworld.com>
- [10] APTECH Team (2005): Java Simplified – 1. Aptech Limited, Mumbai, India.



AUTHORS PROFILES

Eyo O. Ukem holds the Bachelor of Engineering (B.Eng.) degree in Electrical Engineering, obtained from Ahmadu Bello University, Zaria, Nigeria. He also holds the Master of Engineering (M.Eng.) degree in Electronics and Telecommunications Engineering from the same university. His interests are mostly in Information and Communication Technology. He is currently a Senior Lecturer in the Electronics and Computer Technology Unit of the Department of Physics in the University of Calabar, Calabar, Nigeria.

Francis A. Ofoegbu obtained a Bachelor of Science (B.Sc.) degree in Electronics and Computer Technology from the University of Calabar, Nigeria, in 2009. He is currently serving in the compulsory National Youth Service Corps scheme of the Federal Republic of Nigeria. His interests lie in Electronics and Computer Technology.