COMPUTERISED ADMISSION PROCESS FOR POST GRADUATE COMPUTER SCIENCE PROGRAMME.

(A CASE STUDY OF FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA)

BY

AYANLOWO AYANWUYI SALIU

PGD/ MCS/ 99/ 2000/ 966.

A PROJECT SUBMITTED TO THE

DEPARTMENT OF MATHEMATICS / COMPUTER SCIENCE

FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA,

NIGER STATE.

FOR THE AWARD OF POST GRADUATE DIPLOMA

(PGD) IN COMPUTER SCIENCE

SEPTEMBER, 2001

CERTIFICATION

This is to certify that this project: **COMPUTERISED ADMISSION PROCESS FOR POST GRADUATE COMPUTER PROGRAMME** is carried out by **MR. AYANLOWO AYANWUYI SALIU** meets the requirements for the award of Post Graduate Diploma in Computer Science of Mathematics/Computer Department, Federal University of Technology, Minna, Niger state.

DR. Y. AIYESIMI **(Project supervisor)**	**DATE**
DR. S. A. REJU **(Head of Department)**	**DATE**
External Examiner	**DATE**

DEDICATION

This project work is dedicated to Almighty God.

ACKNOWLEDGEMENT

My special thank goes to God for His special wisdom and understanding and grace to go through this course.

Great thanks, especially to my able supervisor, Dr. Y. Aiyesimi who continuously assisted, encourage me and had time to go through my work and also gave all the necessary ideas, view, meaningful contribution thoughts, then made all the necessary corrections that needed to be made. God bless you, I appreciate your effort.

I wish to also express my appreciation to my H.O.D, Dr. S.A. Reju, my able lecturers: Prince R.O. Badmus, Mallam Audu Isha, Mr L.N. Ezeakor, Prof. A. Y Adeboye, Mr. Danjuma, Mr. Hakimi Danladi, Mr. Kola Abdulraheem, Mr. Saliu Adam Muhammed and the rest lecturers in the department for their wonderful support, fatherly advice, opinions and encouragement during the course of this project. God bless you all.

My deepest gratitude to my uncle, in person Alhaji Yahaya Raman, for his help. Also to my dear and loving wife, Mrs. Munirat Ayanlowo for her moral and financial support. May the Almighty Allah bless you richly.

TABLE OF CONTENT

CHAPTER THREE

CHAPTER FOUR

CHAPTER FIVE

ABSTRACT

Today, computer is widely used to solve different types of problems. This study computerized Post Graduate Diploma (PGD) Computer Admission Process, was proposed to improve the current manual systems by the use of computer. To evolve the new system, system development life cycle approach was chosen. This consists of feasibility study, system analysis, system design, system software development and system implementation and evaluation.

CHAPTER ONE

1.1 INTRODUCTION

With the computers becoming an integral part of our daily activities, we need to focus on how it can improve our lives. It would improve our lives by improving our ways of thinking, our ways of working and lastly our day-to-day decisions. There is the drive to encourage the use of new technologies on a regular basis. This trend in our society seems to be more of a daunting task.

The role of the Technological Institutions especially the Universities in the technological development of the nation will be counter productive if an automated information storage, processing and management culture is not integrated into the system.

The idea behind the design of the project management system is to provide a formalized procedure for the coordination of the programme (the P.G.D) with appropriate information from all relevant sources Internal e.g. the individual course lecturers or external e.g. Programme moderators if any for the purpose of planning and coordinating the activities for which he is responsible.

The system to be designed provide the necessary solutions to early compilation and release of results, determination of the area of in-depth study of the students as against their actual performance in the course and other relevant information.

1.2 SCOPE AND LIMITATIONS OF THE STUDY

The scope of this study is limited to the management of the P.G.D students' programme of the Computer Science Department since the design of the project management of the entire departmental programme is a very wide task. This project work covers the students' registration, course registration and result compilation.

The critical area of interest shall therefore be limited to the design of the followings:

a) Admission and registration of the students
b) Management of the essential records of the registered student.

c) P.G.D project planning and result processing of the students.

d) Monitoring of the progress of the departmental thesis or projects as approved for the programme.

1.3 OBJECTIVES OF THE STUDY

The automation of the management of the bridge programme of the department of Maths/Computer science prompted the study. The aim is to develop project management system (PMS) for the management of the vital records of the P.G.D program of the department with the following objectives.

1. To provide an integrated database system for the student for each academic session.
2. To ascertain the effectiveness and suitability of the project with the use of the present inputs (e.g. Class rooms, Lecturers, Equipment etc)
3. To determine the performance & consequently the areas of the program that may require attention.
4. To make available when needed relevant statistics of the program.

1.4 DEFINITION OF TERMS

1.4.1 ADMISSION: Admission in relation to this project study could be defined as entering or being allowed to enter a school (F. U. T as a case study) for the purpose of studying Post Graduate Diploma in Computer Science.

1.4.2 COMPONENTS OF A TYPICAL COMPUTER SYSTEM:

The hardware or physical components of a microcomputer system consists of six elements.

1. The central Processing unit (CPU), which is the brain of the computer. The CPU or microprocessor resides on the computer's motherboards. The microprocessor contains a small amount of internal memory or registers as well as the arithmetic logic unit (ALU), which is responsible for performing all mathematical and logical operations.
2. A monitor is a display device that allows you to see information. The monitor is also sometimes referred to as the visual display unit or CRT (Cathode Ray Tube).

3. The CPU receives information from the ***Input Devices.*** The most common of which are keyboard and mouse. Other input devices may include the microphone, scanners, & digital cameras.
4. The CPU sends information to output devices. A common output device is a printer, which allows the computer to put information on paper, producing a hardcopy. A monitor is the most common output device. Other output devices are speakers, plotters etc.
5. Just like human, computer store information in memory. The two types of computer memory are Read-only memory (ROM) and Random Accessed Memory (RAM). ROM is a permanent memory where the computers instruction set resides. No instruction can be added or written to ROM.

   RAM on the other hand is temporary memory or workspace by the computer's software program. RAM is volatile memory unlike the ROM and its content is erased when the computer is shut down.
6. Since RAM is erased when power is removed, there is a place to store the programs. This storage device allows programs to be saved for easy retrieval at some other time. Such devices are typically called disks and are further classified as Hard disk, Zip disk or floppy disk.

In contrast to computer hardware, the computer software consists of the application or program that allow you work on the computer. A few typical software program include, word processor, spreadsheets, databases, games and of course the designed Project Management System (PMS).

Lastly, there is an operating system on each computer that works to coordinate the activities of the computer. An example is the Windows 98™ which function to provide an easy graphical users interface (GUI).

1.5 MOTIVATION

Prior to introduction/ invention of computer, jobs were performed manually with so many personnel involved. These jobs were prone to error and can be slow in operation due to

sluggishness of some workers. Hence the era was being referred to as "Dark ages" of Data.

Nowadays, computer is the order of the day; it has almost taken manual processing in some organisations like banking, industries, ministries, and Research centres, space mission e.t.c. This is the result of its accuracy, speed and its effectiveness.

Taking all these into consideration, the role of the technological institution development of the universities in technological development of the nation will be counter productive if an automated information storage, processing and management culture is not integrated into the system.

Hence the need of designing a project management system to provide a formulised procedure for the coordination of the programme (PGD) with appropriate information from all relevant sources internal e.g individual course lecturers or external e.g programme moderators if any for the purpose of planning and co-ordinating of the activities for which he is responsible.

1.6 FEATURES OF THE PROJECT

Essentially, the features of this project: Computerised Admission process for post Graduate computer science programme includes:

(1) PGD project planning

(2) Result processing of the students

(3) Management of the essential records of the registered students.

(4) Departmental project management

(5) Proper monitoring of the progress of the departmental projects as approved for the programme.

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 BRIEF HISTORY OF THE PGD COMPUTER PROGRAMME

2.1.1 INTRODUCTION

The Programme is introduced with the view to training holders of Bachelors degree of recognised institution in pure science and social science based discipline. Also, holders of appropriate professional qualifications like the Higher National Diploma (HND), ACCA, SIS, and ICAN. The programme is tailored towards more practical applications than theoretical. For this reason, people with some years of working experience can obtain higher qualifications in preparation for more theoretical and research oriented degrees.

2.1.2 COURSES AND DEGREE AWARDED

The Courses shall be known by the name of Post-Graduate Diploma in Maths with Computers, abbreviated PGD-MCS as listed below:

FIRST SEMESTER

Course Code	Course Title	Credit Unit
DCPT 011	Introduction to computer	3
DCPT 013	General Mathematics	3
DCPT 014	Algorithms	3
DCPT 015	Programming	4
DCPT 016	Pre-written Software Packages	4

DCPT 017	Hardware Components and Systems	3
DCPT 018	Introduction to Statistics I	3

SECOND SEMESTER

Course Code	**Course Title**	**Credit Unit**
DCPT 020	Numerical Analysis	3
DCPT 021	Operating System	3
DCPT 024	Compiler Construction	3
DCPT 026	Data Base Management	3
DCPT 027	Introduction to Statistics II	3
DCPT 028	System Analysis and Design	3
DCPT 029	PROJECT	6

The award given is Post Graduate Diploma in Computer Science (PGDCS).

The conditions for the award above (PGDCS) are:

1. The student is expected to take and pass 25 Units core courses. On the whole a student must earn a minimum of 32 units to qualify for the award.
2. The student must satisfy all other conditions specified by Federal University of Technology, Minna, Post Graduate School for the award.

2.1.3 DURATION OF COURSE

The Programme shall be run on full-time basis for one calendar year of twelve months and part-time basis for two calendar years of twenty-four months.

2.1.4 ENTRY REQUIREMENTS

The minimum entry requirement for the programme is normally a Bachelors degree of Federal University of Technology, Minna or any other recognised institution in pure science, Engineering and social science based discipline.

Holders of appropriate professional qualifications like the Higher National Diploma (HND), ACCA, SIS, and ICAN etc may be considered on individual merit.

2.1.5 PROGRAMME STRUCTURE

The course work shall cover two semesters of lectures, tutorials and practical. The Project work follows immediately after the second semester examination and will occupy the remaining period.

2.1.6 EXAMINATIONS

Taught courses will be examined at the end of each semester through the combined process of continuous assessment and examination. The continuous assessment, which may consist of assignments, tests, seminars, etc, carries 40% while the examination carries 60%. The pass grade for examinations will be in the following categories:

(i)	70% and above	-	A
(ii)	60-69%	-	B
(iii)	50-59%	-	C
(iv)	45-49%	-	D
(v)	40-44%	-	E
(vi)	0-39%	-	F

CGPA		
4.5 and above	-	Distinction
3.5 - 4.4	-	Upper Credit
2.5 - 3.4	-	Lower Credit
1.0 - 2.4	-	Pass
0 - 0.9	-	Fail

2.1.7 THESIS / PROJECT WORK

The thesis/project will be written on the basis of designed work. The project topic would be chosen to solve practical industrial/domestic problems based on the candidate's working experience.

Candidates are required during long vacation and after the Second Semester Examinations to travel to some places (or places of work) to collect data and materials.
The project will be examined and orally defended. The oral examination shall be conducted by an External Examiner, Members of the Advisory Committee of the candidate and the Head of the Department.

2.1.8 GRADUATION REQUIREMENTS

For graduation, a candidate is expected to take and pass 25 Units core courses. On the whole a candidate must earn a minimum of 32 units to qualify for the award of Post Graduate Diploma in Computer Science (PGDCS).

2.2 THE AIM AND OBJECTIVES OF PGD COMPUTER PROGRAMME.

AIMS:

It is well known fact that in recent years many establishments, namely, Government, Bank, Companies and even individuals, have accepted the necessity to not have the opportunity of having computer education. The main aim of this programme is to provide broad based training in some aspect of Computer Science and related subjects that are of great need in industry and other establishments.

OBJECTIVES:

The objective is to produce from graduate of Computer Science and other graduates, Professional Computer Scientists and technologists with good applicable knowledge of Computer Science and technology in the daily needs of the nation. It is intended to adequately equip professionals with high computing skills in various aspects of computer science in order to enhance their industrial productivities.

Furthermore, National event as presented by lack of critical records and the inefficient way of retrieving available record was a factor that motivated this research. The aim of this program therefore is to provide in a very simplified form a data based management system for the management of the vital record of this program, or shall enable the department to present several options of the course statistics as well as ease the vigour usually associated with the manual management of the schools record.

2.3 DEPARTMENTAL PROJECT MANAGEMENT

One of the management problems in most institutions of higher learning today is lack of adequate data. These problems often make the task of proper academic planning very difficult phenomenon. Consequently futuristic planning remains non – existent.

During this research, it was noticed that even where the required data are available, their reliability is open to question. Even when planning is to be politicalized as it does occur in our country, access to relevant data helps to eliminate some unnecessary duplication of efforts at least

No project can run efficiently without some form of organised management. Even an apparent simple sequence of action needs a certain amount of management. Therefore in any project there must be plan on what is to be done and there must also be some level of monitoring on how things are going in relation to the plan.

If the course is running below expectation, or if the courses are beginning to be attractive to the targeted audience, it is essential to discover this as soon as possible and to be able to see how the project as a whole is going to be affected. The sooner the problem is uncovered, the sooner the corrective action can be taken and the possible effect of the problem curtailed.

Furthermore, the availability of an accurate data on the result and Departmental courses/project is essential to be able to reduce the current upsurge in certificate racketing, thereby reducing the level of embarrassment to the institution.

The objective of the PGD Computer package therefore is to provide data needed for;

- Planning.
- Scheduling of the courses in the PGD programme.
- Using the available resources effectively.
- Monitoring and controlling of the progress of the project.
- Evaluating what is being produced (graduates) and for insistence on the importance of high – standard of learning and teaching.
- Identifying areas of difficulty for the courses.
- Identifying quickly the causes of problems and creating the flexibility to adjust person's plans if necessary.

The need for computer application to schools/course, management therefore cannot be over emphasised. Moreover, the system has the ability to aid management to identify areas of potential growth modification such as level of courses taught, level or quality of the students admitted for the program e.t.c. and to compensate for this by making further plans.

CHAPTER THREE

3.0 SYSTEM ANALYSIS AND DESIGN

3.1 SYSTEM ANALYSIS

System analysis is the process or activities involved in examining an already existing system for the new system to be introduced. The analysis was carried out with the primary aim of obtaining complete and authentic information, which will provide a real knowledge of the prevailing situation so that the feasibility of designing an effective computerised system can be known.

During this stage answers were collected to the processes and method of the system under investigation to be able to ascertain effectively how it works.

Preliminary investigation, fact-finding, analysis/system design, staggered implementation and finally the documentation were undertaken.

3.2 FEASIBILITY STUDY

Feasibility study is the broad study of a system to identify reasons, which justifies the development of the new system or favours non – execution of the project. For an efficient study, I required an assistant to carry out the study together which is very necessary because it helps to prevent wasting of time, efforts and other resources.

The objectives that accomplished include the followings: -

1 CLARIFYING AND UNDERSTANDING THE PROJECT REQUEST.

Research was carried out to determine the optimal attainment of the project, that is,

i. What is being done.

ii. What is required

2 DETERMINING THE SIZE OF THE PROJECT.

This was necessary to be able estimate the amount of time and the number of people required to develop and implement the project.

3 ASSESSING COST AND BENEFIT OF THE PROPOSAL AND ALTERNATIVE APPROACHES.

It was important to know the cost of the project including the cost of training and re-training endeavours of the new information technology system. There was also the need to determine the technical, financial and operational feasibility of alternative approaches.

Lastly, the findings were documented in form of this report to the supervisor with recommendation, and outlining the need for acceptance of the proposed new system.

3.3 METHOD OF RESEARCH

The method of investigation included the fact-finding techniques that were used to carry out investigation on the new system. The techniques used are the combinations of the followings: -

(a) Interview (b) observation and (c) several inspections.

Before the design of the new system, efficient and proper grasping of the existing work and information flow is of paramount importance.

This was carried out adequately and effectively with the above methods of investigation at the department and the Dean offices.

The facts obtained from the stages of research was documented and information flow of the proposed PGD Computer Software was reduced to the basic component of input, processing and output for easy and better understanding of the design technique of the "PGD Computer Software".

3.4 SYSTEM DESIGN, SPECIFICATION AND PROCEDURE

System design is the use of an Engineer's creative ability and sense in organizing a logical but literally feasible procedure for a computerized system. Specifically, the design stage of the PGD software design would produce three outputs as follows:

3.4.1 GENERAL SYSTEM DESIGNS.

The general system design includes the flow chart (procedure), a written explanation (Pseudocode). For this study, coding (programming) is used to depict the representation of the project. This is available in the project Appendix A.

3.4.2 DATABASE DESIGN

Database design, which is also known as file design, is the denominator of any system. It contains the raw material (Data) necessary to produce output.

In manufacturing, for example, it is only when the product to be made is decided upon that the raw material could be specified and ordered. In the process of developing an information system, output requirement are first decided upon before data are specified. In a sense, output requirement can be thought of as an input database design.

Therefore the database file design includes the entire database files used throughout the system for the proper storage of the course processing data. Below is the physical descriptive structure of the entire database file used.

1. MAINREC. DBF

This database file is the main database file containing all registration data about each student. It also contains all score and computation of result for each student. The structure of the database file is as shown below.

FIELD	FILE NAME	WIDTH	TYPE
1	REG. NO	20	Character
2	NAME	30	Character
3	MODE	15	Character
4	ADDRESS	50	Character

5	OPTION		
6	UNIV1	30	Character
7	UNIV2	30	Character
8	DATE1	4	Character
9	DATE2	4	Character
10	YEAR	9	Character
11	COURSE11	12	Character
12	COURSE13	12	Character
13	COURSE14	12	Character
14	COURSE15	12	Character
15	COURSE16	12	Character
16	COURSE17	12	Character
17	COURSE18	12	Character
18	COURSE20	12	Character
19	COURSE21	12	Character
20	COURSE22	12	Character
21	COURSE24	12	Character
22	COURSE26	12	Character
23	COURSE27	12	Character
24	COURSE28	12	Character
25	COURSE29	12	Character
26	SCORE11	5	Numeric
27	SCORE13	5	Numeric

28	SCORE14	5	Numeric
29	SCORE15	5	Numeric
30	SCORE16	5	Numeric
31	SCORE17	5	Numeric
32	SCORE18	5	Numeric
33	SCORE20	5	Numeric
34	SCORE21	5	Numeric
35	SCORE22	5	Numeric
36	SCORE24	5	Numeric
37	SCORE26	5	Numeric
38	SCORE27	5	Numeric
39	SCORE28	5	Numeric
40	SCORE29	5	Numeric
41	TCP1	7	Numeric
42	TNU1	3	Numeric
43	GPA1	5	Numeric
44	TCP2	7	Numeric
45	TNU2	3	Numeric
46	GPA2	5	Numeric
47	CGPA	5	Numeric
48	UNIT11	2	Numeric
49	UNIT13	2	Numeric
50	UNIT14	2	Numeric

51	UNIT15	2	Numeric
52	UNIT16	2	Numeric
53	UNIT17	2	Numeric
54	UNIT18	2	Numeric
55	UNIT20	2	Numeric
56	UNIT21	2	Numeric
57	UNIT22	2	Numeric
58	UNIT24	2	Numeric
59	UNIT26	2	Numeric
60	UNIT27	2	Numeric
61	UNIT28	2	Numeric
62	UNIT29	2	Numeric
63	TITLE11	40	Character
64	TITLE13	40	Character
65	TITLE14	40	Character
66	TITLE15	40	Character
67	TITLE16	40	Character
68	TITLE17	40	Character
69	TITLE18	40	Character
70	TITLE20	40	Character
71	TITLE21	40	Character
72	TITLE22	40	Character
73	TITLE24	40	Character

74	TITLE26	40	Character
75	TITLE27	40	Character
76	TITLE28	40	Character
77	TITLE29	40	Character
78	REMARK1	12	Character
79	REMARK2	12	Character
80	REMARK3	12	Character

2. COURSES. DBF

This database file is used to store courses information for all valid courses in the department. Below is the structure of the database file.

FIELD	FILE NAME	WIDTH	TYPE
1	ID-CODE	1	Character
2	CODE	7	Character
3	TITLE	40	Character
4	UNIT	1	Numeric

3.5 COST AND BENEFIT ANALYSIS

3.5.1 COST ANALYSIS

The cost benefit analysis is the analysis of the total cost (expenses) that is needed in order to put the PGD software under development into actual implementation. It is quite

important to note that cost and benefit analysis at times is difficult to quantify at best, but must be done in order to estimate the financial and operational impact within the organization.

Below is the breakdown of the cost of implementing the PGD software.

3.5.2 DEVELOPMENT COST

S/No.	Description of items	Units	Rate	Amount
A	1. System Analysis & Design	5 weeks	6,000.00	30,000.00
	2. Software development	8 weeks	5,000.00	40,000.00
	3. Equipment procurement	Lot		150,000.00
	4. Installation	Lot	35,000.00	35,000.00
	5. Training of (2 Staff)	2 weeks	5,000.00	20,000.00
TOTAL COST ON ITEM A				**140,000.00**

3.5.2 OPERATING COST

S/No.	Description of items	Units	Rate	Amount
B	1. Supplies of computer accessories & stationeries	Lot	70,000.00	70,000.00
	2. Equipment maintenance	Lot	50,000.00	50,000.00
	3. Application software as may be required	Lot	150,000.000	150,000.00
	4. Labour	Lot	25,000.00	25,000.00
TOTAL COST ON ITEM B				**160,000.00**

Grand total A & B = 300,000.00

3.6 BENEFIT ANALYSIS

The benefit analysis of any management information system cannot be overemphasised. It functions to enable the manager to make high and quality decisions. An improved PGD software design system should result for example, in more effective project management.

The PGD software will ensure a timely and accurate production of critical courses reports that will enable management to take proper decision as regards the progress and importance of the PGD Computer course program.

Efforts have been made to produce a comprehensive, convenient and an adaptable program for the PGD Computer course.

CHAPTER FOUR

4.1 PROGRAM DEVELOPMENT/ IMPLEMENTATION

The process of software development is a co-operative effort of the users of the software and system Analysist. While the Analysists are the professionals that deals with the technology and its application to project information processing needs, the users of the software on the other side have an in depth familiarity with the respective manual functional areas.

The skill and knowledge of these two groups complement each other and can be combined to create any type of information systems during the course of the software development process.

However, because software development is a team effort, most programmers have adapted a standardized "System/Software Development Methodology", that provides a framework for cooperation and successful development of a new system.

This step-by-step system development procedures is as illustrated below in a tabular form.

STEPS	ACTIVITIES	MEASURABLE OUTPUT
Analysis and specification	Appraisal of existing situation identification of users requirement	
Design	Design the overall program structure Design the detailed processor Processing logic	Process logic specifications, using Pseudocode or flow chart techniques.
Programming	Writing of code in the appropriate processor language Entry of written code into the computer	Hand written program code Computer printed list out of code.
Testing and debugging	Removal of syntax and logic errors. Final testing of program	Error free Program execution
Installation and maintenance	Error free program installation User training	Program documentation and up grade procedures.

However, software development process is essentially the same, be it for Inventory Management System, Currency Processing, airline reservation system etc. As members of a "Project Team" progress through the procedures outlined in a software development

methodology, the result of one step provided the input for the next step and/or subsequent steps. The project team typically is made up of both users and Computer Engineers.

The methodological approach to software development is a tool information services and users employ to coordinate the effort of a variety of people engaged in a complex process. Hence for a successful program development of PGD software, the writer of this project work is part of the object team as Programmer. The above five (5) steps in the table were followed one after the other in order to design and develop a customized software package called the "PGD Software".

4.2 CHOICE OF PROGRAMMING LANGUAGE

From the previous analysis, it is pertinent to say that the proposed system is going to be used to store large number of data/information, and time-to-time retrieval of record. Due to this fact, the choice of the programming language chosen for the development of the system is Database Management System (DBMS) package with special preference for Visual Fox Pro.

4.2.1 FEATURES OF THE PROGRAMMING LANGUAGE CHOSEN

The choice of Visual Fox Pro arose because of the following reasons and features posed by the application software.

- It is easy to write an interactive user interface program and also simple to understand.
- It is user's friendly.

- It reduces data redundancy.
- Data integrity can be maintained
- Provide easy and greater access to information.
- Individual database file can be designed to meet specification requirement of particular functional unit of an organisation.
- The Visual Fox Pro allows the source program to be compiled to an executable file thereby allowing the program to be run independently of the application software that was used in coding the program e.g. Visual Fox Pro.
- It has quality graphical users' interface.

4.3 PHYSICAL AND LOGICAL DESIGN OF THE SYSTEM

The physical and logical design of the system deals with the physical construction of the logical design of the proposed system. It has to do with program specification of output, input file and processing into computer software.

The design of the computer software is important to ensure that the actual program produced performs the entire task intended and to allow for future modification to be performed efficiently and with minimum destruction to the design of the system.

4.4 SYSTEM IMPLEMENTATION

System implementation is a broad term that encompasses testing and debugging, hardware and software requirement, system installation and system conversion.

It is also the coordination of the fact, which is necessary in ensuing the operation of the new system.

4.5 SYSTEM TESTING AND DEBUGGING.

The essence of program testing is to make sure that program is error free and that all the logic involved are well defined and straightforward. However, it is often seen as a means of establishing that a program is error free and that it does what is required. It is virtually impossible to test a program so thoroughly that it can be claimed to be free of error. In most cases fixing one error gives rise to host of others, which in turn have to be corrected exhaustively. It is much more realistic to think of testing as a "process of finding error". When a stage is reached, when the program appears to run perfectly, this does not mean that there are no more errors in the program, it simply mean that those errors have not been discovered.

Hence, the PGD software have been developed, tested with real live data, irrespective of people and environment and it was found to be error free.

4.6 HARDWARE AND SOFTWARE REQUIREMENT

To make maximum utilization of the developed system, certain hardware and software needs to be installed.

4.6.1 HARDWARE REQUIREMENT

This comprises of all the physical component of the computer and its accessories. Therefore the choice of the computer requirement is done to suit both the current and the future needs of the organization with respect to the volume and types of data to be processed. In summary, a computer system with the following minimum requirement is required.

4.6.2 SOFTWARE REQUIREMENTS

Software requirement are the basic and other relevant application software that need to be installed on the computer in order to make maximum utilization of the computer system and the developed PGD software. It is because of this fact that the following software must be installed on the system.

However, other application software apart from the ones mentioned above may be installed as may be required by the staff and organization concerned.

4.7 PROGRAM INSTALLATION

This is the process of transferring the developed software from the floppy disk to a permanent storage device called the Hard disk.

However, due to vast improvement in recent operating systems, the installation of the software is a very simple operation.

The procedure as follows:

STEPS	PROCEDURE	RESULT
1. Go to start	Click	Start popup menu is displayed.
2. Run submenu	Click	Run dialog box displayed
3. Insert PGD SOFTWARE diskette into A:\	-	-
4. Type the source drive (A:\)	Browse	Content of A:\ displayed
5. Select set-up	Double – click	Installation begins
6. Follow the instruction that Follows	Click	Installation in progress
7. Destination	-	Software installed C:\ successfully

BRINGING THE SOFTWARE FROM C:\ TO WINDOW PROGRAM SUBMENU

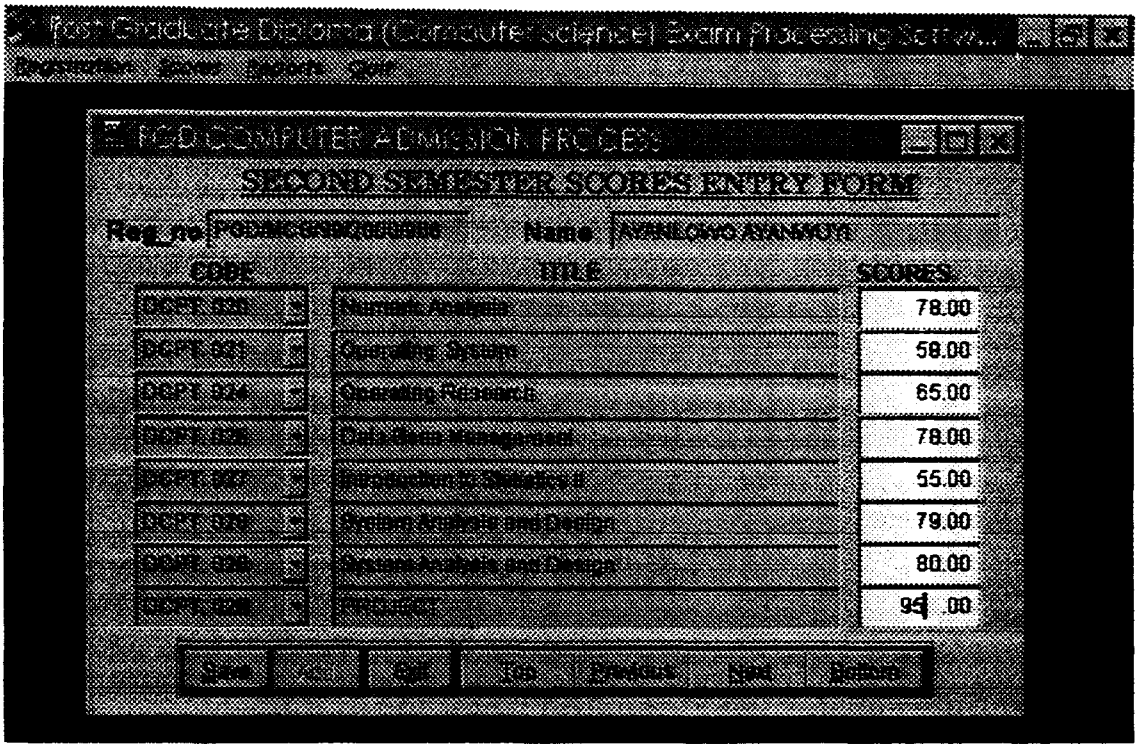
STEPS	PROCEDURE	RESULT
1. Go to start	Click	Start popup menu displayed
2. Setting submenu	Click	-
3. Choose taskbar & start menu	Click	Taskbar dialog box
4. Select start menu program	click	-
5. Browse to select PGD software	Double - click	-
6. Select folder (program)	Click	-
7. Destination	click	PGD software copied into program submenu successfully.

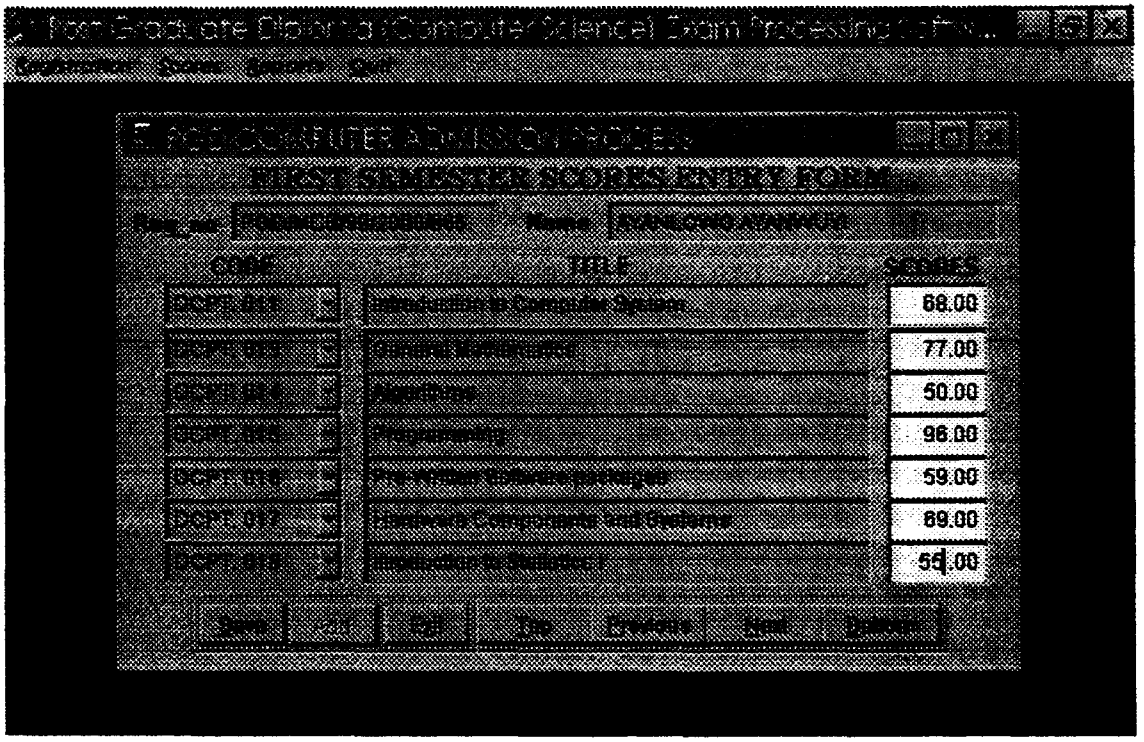
At the end of the installation, the floppy diskette becomes a backup and should be well safeguarded against any damages for future use.

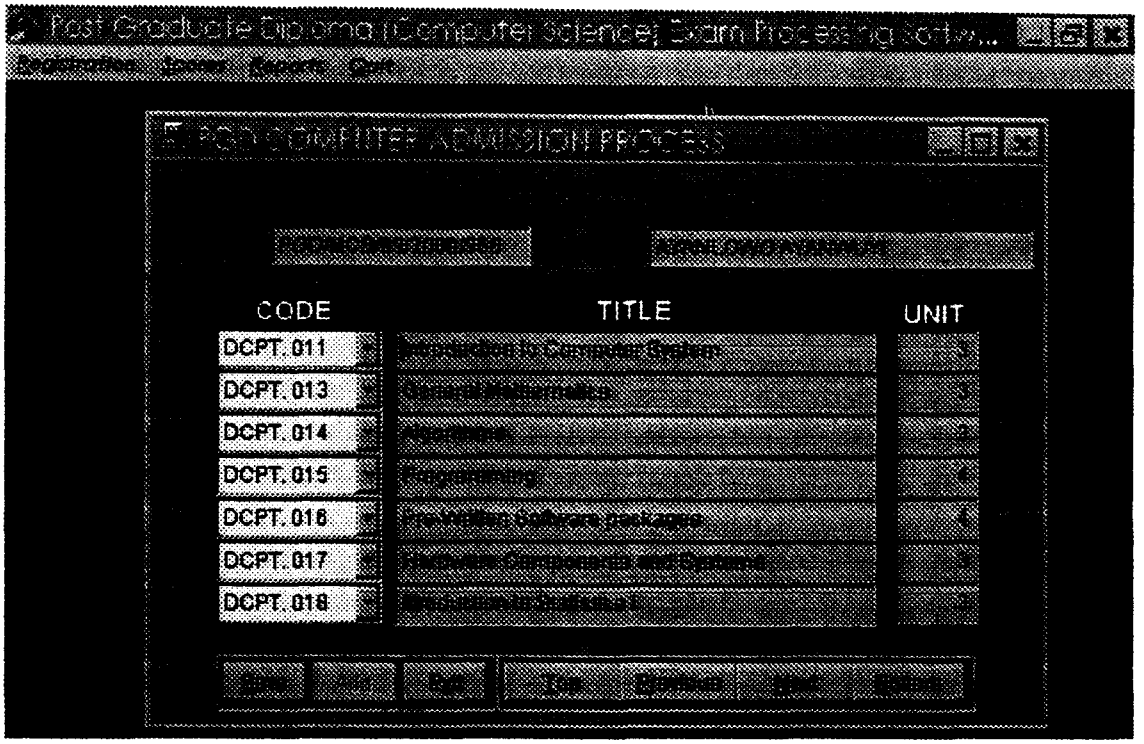
To start the system after successful installation of the PGD software, all one need to do is to go to start and run the folder name you had created.

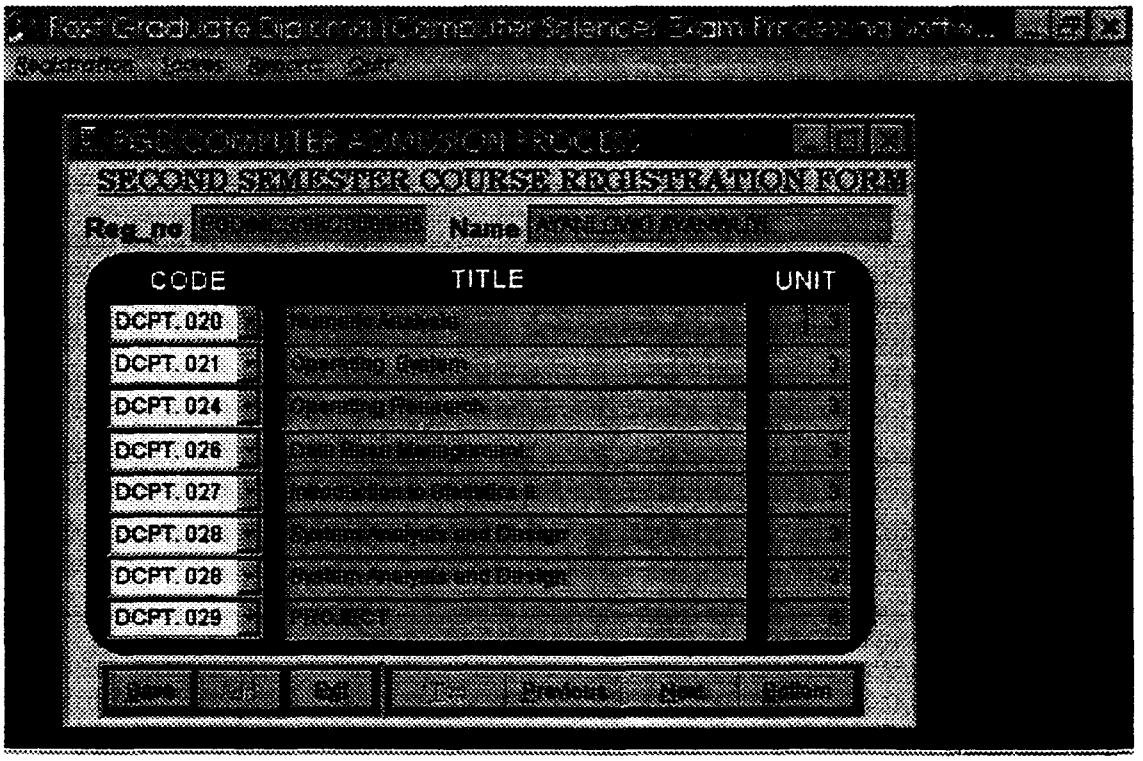
4.8 PROGRAM LAYOUT DESIGN

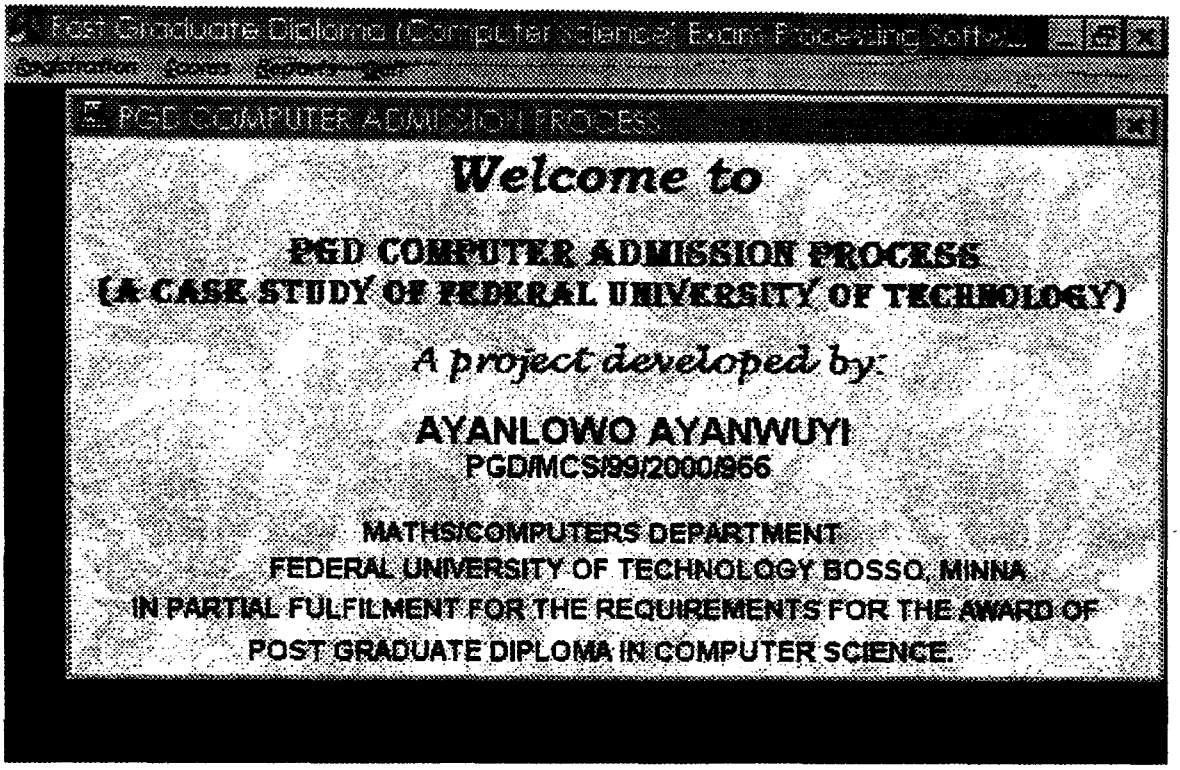
The main menu of the PGD software is in form of rectangular square containing pop up or pulls down menu options. The user of the system is expected to use the mouse or up and down arrow key to highlight the option. This is preceded by either enter key or double clicking to carry out the execution. Below is the structure of the main menu.

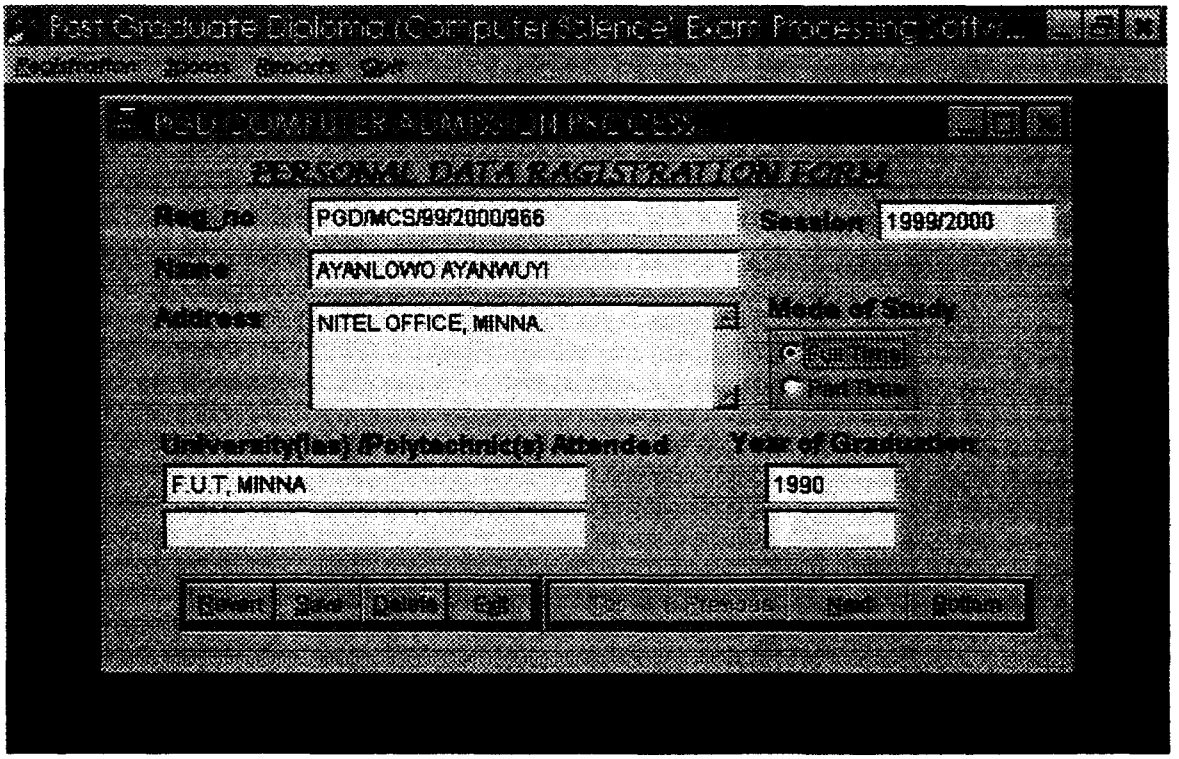

SECOND SEMESTER SCORES ENTRY FORM
Reg_no
Name
CODE
TITLE
SCORES
78.00
58.00
65.00
78.00
55.00
79.00
80.00
Save
Exit
Top
Previous
Next
Bottom


FIRST SEMESTER SCORES ENTRY FORM
CODE
TITLE
SCORES
68.00
77.00
50.00
96.00
59.00
89.00
Save
Exit
Top
Previous
Next
Bottom


CODE
TITLE
UNIT
DCPT. 011
DCPT. 013
DCPT. 014
DCPT. 015
DCPT. 016
DCPT. 017
DCPT. 018


SECOND SEMESTER COURSE REGISTERATION FORM
Reg_no
Name
CODE
TITLE
UNIT
DCPT. 020
DCPT. 021
DCPT. 024
DCPT. 026
DCPT. 027
DCPT. 028
DCPT. 028
DCPT. 029


PGD COMPUTER ADMISSION PROCESS
Welcome to
PGD COMPUTER ADMISSION PROCESS
(A CASE STUDY OF FEDERAL UNIVERSITY OF TECHNOLOGY)
A project developed by:
AYANLOWO AYANWUYI
PGD/MCS/99/2000/966
MATHS/COMPUTERS DEPARTMENT
FEDERAL UNIVERSITY OF TECHNOLOGY BOSSO, MINNA.
IN PARTIAL FULFILMENT FOR THE REQUIREMENTS FOR THE AWARD OF
POST GRADUATE DIPLOMA IN COMPUTER SCIENCE.


PERSONAL DATA RAGISTRATION FORM
Reg_no
PGD/MCS/99/2000/966
Session
1999/2000
Name
AYANLOWO AYANWUYI
Address
NITEL OFFICE, MINNA.
Mode of Study
Full Time
Part Time
University(ies) /Polytechnic(s) Attended
Year of Graduation
F.U.T, MINNA
1990
Revert
Save
Delete
Edit
Next
Return

4.9 PROGRAM TESTING/DEBUGGING & HANDOVER

It's the desire of the author of this program for it to be used in solving the real life problem for which it is intended. Normal program testing and debugging should last for a period of six months, which is not feasible for this design.

In view of the above therefore the author have prolonged all design specification and codes for necessary debugging when the need arises. The author is also ready to be available for further modification when needed.

Once the above procedure has been observed, the author should naturally withdraw, and prolong involvement of the System Engineer with a working system should be avoided. The system maintenance should be a responsibility of the maintenance group within the department. The user must be satisfied that the system works properly and meet their entire requirement by the time of handover. It is essential, therefore that the handover takes place formally with a clear understanding on all sides.

With the above requirement done, it is suffix to say that the PGD software developed has satisfied all the necessary rudiments of system implementation

CHAPTER FIVE

5.0 SYSTEM CONVERSION, DISCUSSION OF RESULTS, CONCLUSION AND RECOMMENDATION

5.1 SYSTEM CONVERSION

This is the process of changing the old system to the new one. There are four methods of dealing with a system conversion. They are: -

(a) PARALLEL SYSTEM:

In this method, the old system is operated along side with the new one. It offers the greatest security, since the organisation can still fall back to the old system without loss of time, money or service, in the event of problems such as errors in processing. The disadvantages are the double operating cost since the two systems are maintained. Also, the new system may not get fair trail.

(b) DIRECT CUTOVER:

Here, the old system is replaced by the new one abruptly, sometimes over a weekend or even overnight. This method forces users to make sure the new system works. This means the users have no other method to fall back on, in the event of any problem. The method therefore requires careful planning.

(c) PILOT SYSTEM

In this method, the working version of the system is implemented first in one department of the organisation. Thereafter, based on the outcome, the system is then installed in the rest of

the organisation by either direct cutover or phase – in method. Pilot system gives experience as live test before implementation.

(d) **PHASE – IN METHHOD**

This method involves the gradual implementation of the new system in segments across the users. Having looked at the advantages and disadvantages of all the four method, in relation to this study, it is recommended that the parallel system of the conversion to be used.

5.2 DISCUSSION OF RESULTS

This study is about computerised PGD Admission process (PGD Student record) in the Department of Mathematics/ Computer Science, Federal University of Technology, Minna, Niger State.

Student record is a vital document of the department, which contains useful and detail information on each student. Such information could be on the student's biography, courses registered and examination result.

Like other information systems, the design of the PGD Computer program involves the following transaction process : - Data Collection , Raw data input, Data Processing, and information output, which thereafter is used by the management as a tool for decision making and action and by the Student to know their progress in the various courses.

In conclusion, the various Output generated are in Appendix III

5.3 CONCLUSION

The design of the page PGD Computer programme was intended to solve a real life problem. It was aimed at reducing the task of the coordinator of the programme in the processing /management of the enormous data of the PGD Computer programme Project.

The design covers important critical areas of the project such as the:

1. Student registration with special emphasis to some key areas of their bio data.
2. Student course registration for the two semesters.
3. Examination processing by semester and by session.
4. Production of project reports of items 1-3.
5. Generation of the student transcript.

5.3 RECOMMENDATION

Though the project was a success as it effectively solved the problem for which it was addressed, there is room for further research on the project.

Further areas of interest for development includes the followings:

(1) Provision for the program to sort the list of registered students by the institutions attended. This may be required for proper monitoring of the admitted student's performance and with the possibility of advising the admission's office of the past records of available Polytechnics or Universities.

(2) Provision for the sorting of the student either by part time or full time.

(3) Provision of the necessary fields for the record of student's performances in the continuous assessment programme.

REFERENCES

AKOR, F. (1994) Understanding Microcomputers published by Aflon Nig. Ltd, Abuja

BADAMOSI, R. O. (1996) System Analysis, Federal University of Technology Minna. (Unpublished) Lecture note.

CAROL, B. & JILL, D. (1996) Software system development (2nd Edition) Mcgraw Hill International

Approved Syllabus for the Full –Time and Party-Time Post Graduate Diploma Course in Computer Science, Department of Mathematics/Computer Science, Federal University of Technology, Minna.

APPENDIX II

```
**********************
*Menu Section
**********************
Set device to screen
Clear
*RUN \projinfo\info.EXE
SET SYSMENU OFF
CLOSE ALL
CLEAR
Set safety off
Set color to w/b
Set exclusive on
Set date to brit
Set century on
Set status off
Set t talk off

Clear
Set path to \ayanlowo
PUBLIC MY, myear
Store space(9) to myear


_screen.icon = "seccast.ico"
_screen.caption = "Post Graduate Diploma (Computer Science) Exam Processing Software"
_screen.picture = "waves.bmp"


my =0
do form frmlogin
sele 1
use mainrec again exclusive

sele 2
use courses again exclusive

sele 1
do pgdmenu.mpr
read events
return

PROCEDURE Load

        public maxw,maxh,wid,hei

        store 0 to maxw,maxh,wid,hei

        maxw = _screen.left + _screen.width
        maxh = _screen.top + _screen.height
```

```
        wid = thisform.left
        hei = thisform.top
    ENDPROC


    PROCEDURE Click
        *release thisform
    ENDPROC


    PROCEDURE timer1.Timer

        if (wid + thisform.left <= maxw)
        thisform.left = thisform.left + 1
        else
        thisform.left = -15
        endif
    ENDPROC


    PROCEDURE timer2.Timer
        thisform.release()
    ENDPROC


    PROCEDURE timer3.Timer
        i = 0.025
        DO WHILE(THISFORM.WIDTH >= 0 AND THISFORM.HEIGHT >= 0)
            IF !(THISFORM.WIDTH = 0 OR THISFORM.HEIGHT = 0)
                THISFORM.WIDTH = THISFORM.WIDTH - i
                THISFORM.HEIGHT = THISFORM.HEIGHT - i
            ELSE
                THISFORM.VISIBLE = .F.
                THISFORM.RELEASE
                EXIT
            ENDIF
        ENDDO
    ENDPROC


ENDDEFINE
DEFINE CLASS form1 AS form


    Top = 6
    Left = 53
    Height = 289
    Width = 528
```

```
DoCreate = .T.
Caption = "PGD COMPUTER ADMISSION PROCESS"
Icon = "..\ayanlowo\pc04.ico"
BackColor = RGB(239,186,160)
Name = "Form1"


ADD OBJECT txtreg_no AS textbox WITH ;
        Comment = "", ;
        ControlSource = "mainrec.reg_no", ;
        Height = 23, ;
        InputMask = "PGD/MCS/99/9999/9999", ;
        Left = 109, ;
        MaxLength = 20, ;
        TabIndex = 1, ;
        Top = 28, ;
        Width = 239, ;
        Name = "txtReg_no"


ADD OBJECT lblreg_no AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
        FontSize = 11, ;
        WordWrap = .T., ;
        BackStyle = 0, ;
        Caption = "Reg_no", ;
        Left = 26, ;
        Top = 30, ;
        Width = 55, ;
        TabIndex = 12, ;
        Name = "lblReg_no"


ADD OBJECT txtname AS textbox WITH ;
        Comment = "", ;
        ControlSource = "mainrec.name", ;
        Height = 23, ;
        Left = 109, ;
        MaxLength = 30, ;
        TabIndex = 2, ;
        Top = 57, ;
        Width = 239, ;
        Name = "txtName"


ADD OBJECT lblname AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
```

```
        FontSize = 11, ;
        WordWrap = .T., ;
        BackStyle = 0, ;
        Caption = "Name", ;
        Left = 26, ;
        Top = 59, ;
        Width = 40, ;
        TabIndex = 13, ;
        Name = "lblName"


ADD OBJECT edtaddress AS editbox WITH ;
        Comment = "", ;
        Height = 61, ;
        Left = 109, ;
        TabIndex = 3, ;
        Top = 86, ;
        Width = 239, ;
        ControlSource = "mainrec.address", ;
        Name = "edtAddress"


ADD OBJECT lbladdress AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
        FontSize = 11, ;
        WordWrap = .T., ;
        BackStyle = 0, ;
        Caption = "Address", ;
        Height = 20, ;
        Left = 26, ;
        Top = 86, ;
        Width = 58, ;
        TabIndex = 14, ;
        Name = "lblAddress"


ADD OBJECT txtuniv1 AS textbox WITH ;
        Comment = "", ;
        ControlSource = "mainrec.univ1", ;
        Height = 23, ;
        Left = 30, ;
        MaxLength = 30, ;
        TabIndex = 4, ;
        Top = 175, ;
        Width = 234, ;
        Name = "txtUniv1"
```

```
ADD OBJECT txtuniv2 AS textbox WITH ;
        Comment = "", ;
        ControlSource = "mainrec.univ2", ;
        Height = 23, ;
        Left = 30, ;
        MaxLength = 30, ;
        TabIndex = 5, ;
        Top = 200, ;
        Width = 234, ;
        Name = "txtUniv2"


ADD OBJECT txtdate1 AS textbox WITH ;
        Comment = "", ;
        ControlSource = "mainrec.date1", ;
        Height = 23, ;
        Left = 360, ;
        MaxLength = 4, ;
        TabIndex = 8, ;
        Top = 175, ;
        Width = 73, ;
        Name = "txtDate1"


ADD OBJECT txtdate2 AS textbox WITH ;
        Comment = "", ;
        ControlSource = "mainrec.date2", ;
        Height = 23, ;
        Left = 360, ;
        MaxLength = 4, ;
        TabIndex = 9, ;
        Top = 200, ;
        Width = 73, ;
        Name = "txtDate2"


ADD OBJECT opgmode AS optiongroup WITH ;
        AutoSize = .T., ;
        ButtonCount = 2, ;
        Comment = "", ;
        Value = 1, ;
        ControlSource = "mainrec.mode", ;
        Height = 46, ;
        Left = 363, ;
        Top = 101, ;
        Width = 81, ;
        TabIndex = 7, ;
        Name = "opgMode", ;
        Option1.Caption = "Full Time", ;
```

```
        Option1.Value = 1, ;
        Option1.Height = 17, ;
        Option1.Left = 5, ;
        Option1.Top = 5, ;
        Option1.Width = 69, ;
        Option1.AutoSize = .T., ;
        Option1.Name = "Option1", ;
        Option2.Caption = "Part Time", ;
        Option2.Height = 17, ;
        Option2.Left = 5, ;
        Option2.Top = 24, ;
        Option2.Width = 71, ;
        Option2.AutoSize = .T., ;
        Option2.Name = "Option2"


ADD OBJECT label1 AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
        FontSize = 11, ;
        BackStyle = 0, ;
        Caption = "Mode of Study", ;
        Height = 20, ;
        Left = 362, ;
        Top = 81, ;
        Width = 104, ;
        TabIndex = 15, ;
        Name = "Label1"


ADD OBJECT label3 AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
        FontSize = 11, ;
        BackStyle = 0, ;
        Caption = "University(ies) /Polytechnic(s) Attended", ;
        Height = 20, ;
        Left = 29, ;
        Top = 156, ;
        Width = 281, ;
        TabIndex = 16, ;
        Name = "Label3"


ADD OBJECT label4 AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
        FontSize = 11, ;
        BackStyle = 0, ;
```

```
        Caption = "Year of Graduation", ;
        Height = 20, ;
        Left = 341, ;
        Top = 155, ;
        Width = 136, ;
        TabIndex = 17, ;
        Name = "Label4"


ADD OBJECT cmdgrpeditor AS commandgroup WITH ;
        AutoSize = .T., ;
        ButtonCount = 6, ;
        BackStyle = 1, ;
        Value = 1, ;
        Height = 33, ;
        Left = 38, ;
        Top = 236, ;
        Width = 201, ;
        TabIndex = 10, ;
        BackColor = RGB(255,0,255), ;
        Name = "cmdgrpeditor", ;
        Command1.AutoSize = .F., ;
        Command1.Top = 5, ;
        Command1.Left = 5, ;
        Command1.Height = 23, ;
        Command1.Width = 48, ;
        Command1.Caption = "\<Add", ;
        Command1.Name = "cmdadd", ;
        Command2.AutoSize = .F., ;
        Command2.Top = 5, ;
        Command2.Left = 53, ;
        Command2.Height = 23, ;
        Command2.Width = 48, ;
        Command2.Caption = "\<Save", ;
        Command2.ColorScheme = 2, ;
        Command2.Name = "cmdsave", ;
        Command3.AutoSize = .F., ;
        Command3.Top = 5, ;
        Command3.Left = 100, ;
        Command3.Height = 23, ;
        Command3.Width = 48, ;
        Command3.Caption = "\<Delete", ;
        Command3.Name = "cmddelete", ;
        Command4.AutoSize = .F., ;
        Command4.Top = 5, ;
        Command4.Left = 148, ;
        Command4.Height = 23, ;
        Command4.Width = 48, ;
        Command4.Caption = "E\<xit", ;
```

```
        Command4.Name = "cmdexit", ;
        Command5.AutoSize = .F., ;
        Command5.Top = 5, ;
        Command5.Left = 5, ;
        Command5.Height = 23, ;
        Command5.Width = 48, ;
        Command5.Caption = "\<Revert", ;
        Command5.Name = "cmdrevert", ;
        Command6.AutoSize = .F., ;
        Command6.Top = 5, ;
        Command6.Left = 53, ;
        Command6.Height = 23, ;
        Command6.Width = 48, ;
        Command6.Caption = "\<Modify", ;
        Command6.Name = "cmdmodify"


ADD OBJECT cmdgrpnavigator AS commandgroup WITH ;
        AutoSize = .T., ;
        ButtonCount = 4, ;
        BackStyle = 1, ;
        Value = 1, ;
        Height = 33, ;
        Left = 238, ;
        Top = 236, ;
        Width = 271, ;
        TabIndex = 11, ;
        BackColor = RGB(255,0,255), ;
        Name = "cmdgrpnavigator", ;
        Command1.Top = 5, ;
        Command1.Left = 5, ;
        Command1.Height = 23, ;
        Command1.Width = 66, ;
        Command1.Caption = "\<Top", ;
        Command1.Name = "cmdtop", ;
        Command2.AutoSize = .F., ;
        Command2.Top = 5, ;
        Command2.Left = 70, ;
        Command2.Height = 23, ;
        Command2.Width = 66, ;
        Command2.Caption = "\<Previous", ;
        Command2.Name = "cmdprevious", ;
        Command3.Top = 5, ;
        Command3.Left = 135, ;
        Command3.Height = 23, ;
        Command3.Width = 66, ;
        Command3.Caption = "\<Next", ;
        Command3.Name = "cmdnext", ;
        Command4.AutoSize = .F., ;
```

```
        Command4.Top = 5, ;
        Command4.Left = 200, ;
        Command4.Height = 23, ;
        Command4.Width = 66, ;
        Command4.Caption = "\<Bottom", ;
        Command4.Name = "cmdbottom"


ADD OBJECT txtxyear AS textbox WITH ;
        Comment = "", ;
        Enabled = .T., ;
        Format = "!", ;
        Height = 23, ;
        InputMask = "9999/9999", ;
        Left = 422, ;
        MaxLength = 9, ;
        ReadOnly = .T., ;
        TabIndex = 6, ;
        Top = 30, ;
        Width = 99, ;
        Name = "txtxYear"


ADD OBJECT lblyear AS label WITH ;
        AutoSize = .T., ;
        FontBold = .T., ;
        FontSize = 11, ;
        WordWrap = .T., ;
        BackStyle = 0, ;
        Caption = "Session", ;
        Left = 358, ;
        Top = 33, ;
        Width = 57, ;
        TabIndex = 18, ;
        BackColor = RGB(0,0,0), ;
        Name = "lblYear"


ADD OBJECT label2 AS label WITH ;
        FontBold = .T., ;
        FontName = "Lucida Handwriting", ;
        FontSize = 11, ;
        FontUnderline = .T., ;
        BackStyle = 0, ;
        Caption = "PERSONAL DATA RAGISTRATION FORM", ;
        Height = 25, ;
        Left = 77, ;
        Top = 5, ;
        Width = 360, ;
```

```
        TabIndex = 19, ;
        ForeColor = RGB(0,64,128), ;
        Name = "Label2"


PROCEDURE Refresh
        *thisform.txtxyear.value = myear
ENDPROC


PROCEDURE Activate
        public mreg_no, mname, mmode, maddress, moption, muniv1, ;
                        muniv2, mdate1, mdate2, mmyear


        Thisform.setall("Readonly",.T.,"Textbox")
        Thisform.setall("Readonly",.T.,"combobox")
        Thisform.setall("Readonly",.T.,"EDITbox")
        thisform.cmdgrpnavigator.enabled = .T.


        thisform.cmdgrpeditor.cmdadd.visible = .T.
        thisform.cmdgrpeditor.cmdmodify.visible = .T.
        thisform.cmdgrpeditor.cmdrevert.visible = .F.
        thisform.cmdgrpeditor.cmdsave.visible = .F.
        thisform.cmdgrpeditor.cmdsave.enabled = .F.

        thisform.cmdgrpeditor.cmdadd.setfocus
ENDPROC


PROCEDURE Load
        sele a
        *use mainrec again
ENDPROC


PROCEDURE Init
        public modified

        modified = .f.
ENDPROC


PROCEDURE cmdgrpeditor.cmdadd.Click
        *sele proj1
        APPEND BLANK

        modified = .f.
```

```
        Thisform.setall("Readonly",.F.,"Textbox")
        Thisform.setall("Readonly",.F.,"combobox")
        Thisform.setall("Readonly",.F.,"EDITbox")
        *thisform.txtyear.readonly = .T.
        thisform.cmdgrpnavigator.enabled = .F.

        thisform.cmdgrpeditor.cmdadd.visible = .F.
        thisform.cmdgrpeditor.cmdmodify.visible = .F.
        thisform.cmdgrpeditor.cmdrevert.visible = .T.
        thisform.cmdgrpeditor.cmdsave.visible = .T.
        thisform.cmdgrpeditor.cmdsave.enabled = .T.

        *
        THISFORM.REFRESH()
ENDPROC


PROCEDURE cmdgrpeditor.cmdsave.Click

                replace reg_no with THISFORM.txtreg_no.VALUE
                replace name with THISFORM.txtname.VALUE
                replace mode with THISFORM.opgmode.VALUE
                replace address with THISFORM.edtaddress.VALUE
                *replace option with THISFORM.opgoption.VALUE
                replace univ1 with THISFORM.txtuniv1.VALUE
                replace univ2 with THISFORM.txtuniv2.VALUE
                replace date1 with THISFORM.txtdate1.VALUE
                replace date2 with THISFORM.txtdate2.VALUE
                replace xyear with thisform.txtxyear.value


        thisform.cmdgrpeditor.cmdadd.visible = .T.
        thisform.cmdgrpeditor.cmdrevert.visible = .F.

        thisform.cmdgrpeditor.cmdmodify.visible = .T.

        thisform.cmdgrpnavigator.enabled = .T.
        Thisform.setall("Readonly",.T.,"Textbox")
        Thisform.setall("Readonly",.T.,"combobox")
        Thisform.setall("Readonly",.T.,"EDITbox")
ENDPROC


PROCEDURE cmdgrpeditor.cmddelete.Click
        store 0 to rep1
        rep1 = messagebox("Are you really sure ?", 36, "Want to Delete ?")
        if rep1 = 6 then
        DELETE
        PACK
```

```
        THISFORM.REFRESH()
        endif
        if this.parent.cmdadd.visible == .F.
                this.parent.cmdadd.visible = .T.
                this.parent.cmdrevert.visible = .F.
                this.parent.cmdsave.enabled = .F.
        endif
ENDPROC


PROCEDURE cmdgrpeditor.cmdexit.Click
        THISFORM.RELEASE()
ENDPROC


PROCEDURE cmdgrpeditor.cmdrevert.Click
        if modified == .f.
        use mainrec again exclusive
        GO BOTTOM
        DELETE
        PACK
        else

                        replace reg_no with mreg_no
                        replace name with mname
                        replace mode with mmode
                        replace address with maddress
                        replace option with moption
                        replace univ1 with muniv1
                        replace univ2 with muniv2
                        replace date1 with mdate1
                        replace date2 with mdate2
                        replace Xyear with mXyear

        endif
        Thisform.setall("Readonly",.T.,"Textbox")
        Thisform.setall("Readonly",.T.,"combobox")
        Thisform.setall("Readonly",.T.,"EDITbox")

        thisform.cmdgrpeditor.cmdadd.visible = .T.
        thisform.cmdgrpeditor.cmdmodify.visible = .T.
        thisform.cmdgrpeditor.cmdrevert.visible = .F.
        thisform.cmdgrpeditor.cmdsave.enabled = .F.


        thisform.cmdgrpnavigator.enabled = .T.

        THISFORM.REFRESH()
ENDPROC
```

```
PROCEDURE cmdgrpeditor.cmdmodify.Click

	Thisform.setall("Readonly",.F.,"Textbox")
	Thisform.setall("Readonly",.F.,"combobox")
	Thisform.setall("Readonly",.F.,"editbox")
	*thisform.txtyear.readonly = .T.

	thisform.cmdgrpnavigator.enabled = .F.

	modified = .t.
			mreg_no = thisform.txtreg_no.value
			mname = thisform.txtname.value
			mmode = thisform.opgmode.value
			maddress = thisform.edtaddress.value
	*		moption = thisform.opgoption.value
			muniv1 = thisform.txtuniv1.value
			muniv2 = thisform.txtuniv2.value
			mdate1 = thisform.txtdate1.value
			mdate2 = thisform.txtdate2.value
			mXyear = thisform.txtXyear.value

	thisform.cmdgrpeditor.cmdadd.visible = .F.
	thisform.cmdgrpeditor.cmdmodify.visible = .F.
	thisform.cmdgrpeditor.cmdrevert.visible = .T.
	thisform.cmdgrpeditor.cmdsave.visible = .T.
	thisform.cmdgrpeditor.cmdsave.enabled = .T.


	*
	THISFORM.REFRESH()
ENDPROC


PROCEDURE cmdgrpnavigator.cmdtop.Click
	GOTO TOP
	THISFORM.REFRESH()


	this.parent.cmdprevious.enabled = .F.
	this.enabled = .F.

	this.parent.cmdnext.enabled = .T.
	this.parent.cmdbottom.enabled = .T.
ENDPROC


PROCEDURE cmdgrpnavigator.cmdprevious.Click
```

```
        IF !BOF()
                SKIP -1
                this.parent.cmdbottom.enabled = .T.
                this.parent.cmdnext.enabled = .T.
                IF BOF()
                        GO TOP
                        this.parent.cmdtop.enabled = .F.
                        this.enabled = .F.

                        this.parent.cmdnext.enabled = .T.
                        this.parent.cmdbottom.enabled = .T.
                else

                        this.parent.cmdtop.enabled = .T.
                        this.enabled = .T.
                ENDIF
        ENDIF
        THISFORM.REFRESH()
ENDPROC


PROCEDURE cmdgrpnavigator.cmdnext.MouseDown
        LPARAMETERS nButton, nShift, nXCoord, nYCoord
ENDPROC


PROCEDURE cmdgrpnavigator.cmdnext.Click
        IF !EOF()
                SKIP
                this.parent.cmdtop.enabled = .T.
                this.parent.cmdprevious.enabled = .T.
                IF EOF()
                        GO BOTTOM
                        this.parent.cmdbottom.enabled = .F.
                        this.enabled = .F.

                        this.parent.cmdprevious.enabled = .T.
                        this.parent.cmdtop.enabled = .T.
                ENDIF
        ENDIF
        THISFORM.REFRESH()
ENDPROC


PROCEDURE cmdgrpnavigator.cmdbottom.Click
        GO BOTTOM
        THISFORM.REFRESH()
```

```
        this.parent.cmdnext.enabled = .F.
        this.enabled = .F.

        this.parent.cmdtop.enabled = .T.
        this.parent.cmdprevious.enabled = .T.
    ENDPROC


ENDDEFINE
*
*-- EndDefine: form1
**************************************************************************
```

APPENDIX III

FEDERAL UNIVERSITY OF TECHNOLOGY

POST-GRADUATE DIPLOMA IN COMPUTER SCIENCE

FIRST SEMESTER RESULT

01/03/2001

Registration Number								Tcp	Tnu	Gpa	Remark
PGD/MCS/99/2000/966	DCPT. 011 68.00	DCPT. 013 77.00	DCPT. 014 50.00	DCPT. 015 96.00	DCPT. 016 59.00	DCPT. 017 69.00	DCPT. 018 55.00	89.00	23	3.87	Upper Credit
PGD/MCS/99/2000/967	DCPT. 011 86.00	DCPT. 013 54.00	DCPT. 016 97.00	DCPT. 014 21.00	DCPT. 015 13.00	DCPT. 018 45.00	DCPT. 017 23.00	50.00	23	2.17	Pass
PGD/MCS/99/2000/968	DCPT. 011 90.00	DCPT. 013 65.00	DCPT. 014 77.00	DCPT. 015 87.00	DCPT. 016 94.00	DCPT. 017 63.00	DCPT. 018 88.00	109.00	23	4.74	Distinction
PGD/MCS/99/2000/969	DCPT. 011 12.00	DCPT. 013 34.00	DCPT. 014 56.00	DCPT. 015 45.00	DCPT. 017 64.00	DCPT. 016 24.00	DCPT. 018 34.00	29.00	23	1.26	Pass
PGD/MCS/99/2000/965	DCPT. 011 75.00	DCPT. 013 53.00	DCPT. 014 40.00	DCPT. 015 26.00	DCPT. 016 48.00	DCPT. 017 68.00	DCPT. 018 84.00	62.00	23	2.70	Lower Credit

FEDERAL UNIVERSITY OF TECHNOLOGY

POST-GRADUATE DIPLOMA IN COMPUTER SCIENCE

SECOND SEMESTER RESULT

01/03/2001

Registration Number									Tcp	Tnu	Gpa	Cgpa	Remark
PGD/MCS/99/2000/966	DCPT. 021	DCPT. 024	DCPT. 026	DCPT. 027	DCPT. 028	DCPT. 028	DCPT. 029	DCPT. 020					
	50.00	65.00	78.00	55.00	79.00	80.00	95.00	78.00	120.00	27	4.44	4.18	Upper Credit
PGD/MCS/99/2000/967	DCPT. 022	DCPT. 024	DCPT. 026	DCPT. 027	DCPT. 028	DCPT. 029	DCPT. 020	DCPT. 021					
	45.00	86.00	78.00	23.00	46.00	68.00	59.00	68.00	87.00	27	3.22	2.74	Lower Credit
PGD/MCS/99/2000/968	DCPT. 022	DCPT. 024	DCPT. 026	DCPT. 027	DCPT. 028	DCPT. 029	DCPT. 020	DCPT. 021					
	49.00	38.00	70.00	94.00	60.00	50.00	84.00	84.00	96.00	27	3.56	4.10	Upper Credit
PGD/MCS/99/2000/969	DCPT. 022	DCPT. 024	DCPT. 026	DCPT. 027	DCPT. 028	DCPT. 029	DCPT. 020	DCPT. 021					
	23.00	63.00	45.00	72.00	62.00	15.00	39.00	12.00	45.00	27	1.67	1.48	Pass
PGD/MCS/99/2000/965	DCPT. 027	DCPT. 026	DCPT. 028	DCPT. 022	DCPT. 024	DCPT. 020	DCPT. 029	DCPT. 021					
	23.00	67.00	45.00	68.00	48.00	69.00	88.00	56.00	87.00	27	3.22	2.98	Lower Credit

FEDERAL UNIVERSITY OF TECHNOLOGY

POST-GRADUATE DIPLOMA IN COMPUTER SCIENCE

ACADEMIC TRANSCRIPT

Reg No PGD/MCS/99/20

Name AYANLOWO AYANWUYI **Year** 1999/200

CODE	TITLE	UNIT	SCORE
DCPT. 011	Introduction to Computer System	3	68.00
DCPT. 013	General Mathematics	3	77.00
DCPT. 014	Algorithms	3	50.00
DCPT. 015	Programming	4	96.00
DCPT. 016	Pre-Written Software packages	4	59.00
DCPT. 017	Hardware Components and Systems	3	69.00
DCPT. 018	Iitroduction to Statistics I	3	55.00

CODE	TITLE	UNIT	SCORE
DCPT. 021	Operating System	3	50.00
DCPT. 024	Operating Research	3	65.00
DCPT. 026	Data Base Management	3	78.00
DCPT. 027	Introduction to Statistics II	3	55.00
DCPT. 028	System Analysis and Design	3	79.00
DCPT. 028	System Analysis and Design	3	80.00
DCPT. 029	PROJECT	6	95.00
DCPT. 020	Numeric Analysis	3	78.00

Total Credit Points : 89.00
Total Credit Units : 23
Grade Point Average : 3.87
Semester Remark: Upper Credit

Total Credit Points : 120.00
Total Credit Units : 27
Grade Point Average : 4.44
Semester Remark: Upper Credit

Cummulative Grade Point Average: 4.18

Final Remark : Upper Credit

FEDERAL UNIVERSITY OF TECHNOLOGY

POST-GRADUATE DIPLOMA IN COMPUTER SCIENCE

LIST OF REGISTERED STUDENTS

01/03/2001

Reg No	Name	Year	Mode	Address
PGD/MCS/99/2000/966	AYANLOWO AYANWUYI .	1999/2000	Full Time	NITEL OFFICE, MINNA.
PGD/MCS/99/2000/967	YAHAYA USMAN	1999/2000	Part Time	NIGERIA ARMY, TRADOC DEPARTMENT, KOTANGORA.
PGD/MCS/99/2000/968	PETER ADAJI	1999/2000	Full Time	CENTRAL BANK OF NIGERIA, MINNA.
PGD/MCS/99/2000/969	KAMAR DANJUMA	1999/2000	Full Time	FEDERAL MINISTRY OF WORK & HOUSING, ABUJA.
PGD/MCS/99/2000/965	JOJO YAHOO	1999/2000	Part Time	RIVER BASIN AUTHORITY, MINNA.