STUDENTS PROJECT DATABASE MANAGEMENT SYSTEM

(A CASE STUDY OF KADUNA POLYTECHNIC LIBRARY)

BY

PETER, Mathias Abayi

PGD/MCS/2008/1257

SUBMITTED TO DEPARTMENT OF MATHEMATICS/COMPUTER SCIENCE

FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA

JANUARY, 2011.

1.2 STATEMENT OF THE PROBLEM

Record management systems are changing drastically in today's technological dispensation; however, keeping tracks of records manually in files, cabinets and jackets has proved to be ineffective, because of the large volume of records of students' projects involved, considering the fast growth of the institutions The management of projects information in the Library of Kaduna Polytechnic (Isa Kaita Library) has been in constant utilization of the manual method of record management whereby cabinets and file jackets are used for all forms of file processing activities like storing, sorting retrieving of file record.

Undoubtedly, the use of this manual system of record management has been an evident problem in the institution. Thus the setbacks are enumerated below;

1. Duplication of data files which if not updated leads to inconsistency of files and wastage of office stationeries.
2. Prone to human error: The manual system in use leads to data errors as recording data is a lengthy process; it makes the whole exercise monotonous boring and hectic.

3. Misplacement/loss of projects files (records), which automatically eliminates the particular student project from being reviewed.

4. Time factor: Due to large number of students' projects available in the institutions' library, it takes more time to process any piece of

STUDENTS PROJECT DATABASE MANAGEMENT SYSTEM

(A CASE STUDY OF KADUNA POLYTECHNIC LIBRARY)

BY

PETER, Mathias Abayi

PGD/MCS/2008/1257

SUBMITTED TO DEPARTMENT OF MATHEMATICS/COMPUTER SCIENCE

FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA

IN PARTIAL FULFILMENT OF REQUIREMENTS LEADING TO THE AWARD OF POSTGRADUATE DIPLOMA (PGD) IN COMPUTER SCIENCE

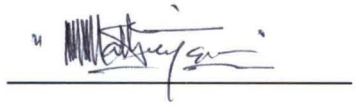
FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA.

JANUARY, 2011.

DECLARATION

I hereby declared that this project has been conducted solely by me under the supervision of Mallam Yisa Yakubu of the department of Mathematics and Computer Science, Federal University of Technology (FUT) Minna, Niger State.

I have neither copied someone's work nor someone else done it for me. In accordance with the ethics of research work, all materials used have been adequately referenced.



14/01/2011

Mathias, Abayi Peter
PGD/MCS/2008/1257

Date

CERTIFICATION

This project work has been read and approved as meeting the requirements of the Department of Mathematics/Computer Science, Post Graduate School, Federal University of Technology, Minna.

.....	.....
Mallam Yisa Yakubu **(Project Supervisor)**	**Date**
.....	.....
Dr. U. Y. Abubakar **(Head of Department)**	**Date**
.....	.....
External Examiner	**Date**

DEDICATION

This project is dedicated to God almighty for all that he has done for me and to my parents Mr. & Mrs. Peter A. Manga and family, My Lovely wife and daughter Mrs Mimi Mathias and Gieveh Victoria Mathias. And to my brothers and sisters, you stood by me in prayers, moral and financial supports God bless you and reward you richly.

ACKNOWLEDGEMENT

This work is acknowledged to God for his mercies, protection and courage given to me from the beginning of this course to the completion.

My sincerely gratitude goes to Mallam Yisa Yakubu my project supervisor for dedicating his time and series of phone calls made, as well as Mallam A. Ndanusa, the PGD Coordinator.

My sincere appreciation to Prof. N. I Akinwande, Prof. K. R. Adeboye, Prof. Y. M. Aiyesimi, Dr. U. Y. Abubakar who is the Head of Department, Dr. L. N. Ezeako, Dr. V. O. Waziri, Dr. Jiya Mohammed the PG Coordinator and other Lecturers and the entire staff of the Department of Mathematics/Computer Science FUT Minna who assisted in various ways to see that this project work is done successful.

I am grateful to my parents Mr. & Mrs. Peter A. Manga for without their blessing and everyday prayers, i wouldn't have gotten to this stage in life. My wife Mrs Mimi Mathias, my daughter Gieveh Victoria Mathias you bear my absence during the program and God really kept you.

My brothers and sisters, Engr & Mrs Taimako E. Usman and family, Arc & Mrs Reuben Manga and Family, Lieut Felix Oveye Manga, Pharm. Esla Ephraim Manga, Mr. George A. Manga, Miss Charity A. Manga, Joseph Alu Manga, Martha Sati Akawu, My nieces and nephews Lucy, Alice, Gloria, Isaac Aso, Rachael, Jeal, Miriam Ayinda and Jeremiah Atari.

I wish to express my sincere gratitude to my friends Abubakar Ibrahim and Family, Nuhu Simon Kasai and family, my colleagues in NNPC Retail Ltd, my Manager, Strategy and Business Development Mrs Gloria Y. Finbong, My course mates Mr.

Magnus Ojoenu, Mr. Cosmos Ezenma, Mr. Jacob Akor, Mr. Segun Aremu, Mr. Godwin Joshua Awan and Miss Augusta.

I also acknowledged my brothers and sisters, friends and colleagues that their names falls under those that are too numerous to mention as the end of every good prayers is amen, so also every good thing is thank you. Thank you for all the good time spent God will reward you accordingly. A very big thank you and pure love.

TABLE OF CONTENTS

TITLE PAGE

DECLARATION

CERTIFICATION

DEDICATION

ACKNOWLEDGEMENT

TABLE OF CONTENTS

ABSTRACT

LIST OF TABLES

CHAPTER ONE

1.0 INTRODUCTION

1.1 BACKGROUND OF THE PROBLEM

1.2 STATEMENT OF THE PROBLEM

1.4 OBJECTIVES OF THE RESEARCH

1.5 SIGNIFICANCE OF THE STUDY

1.6 RESEARCH QUESTIONS

1.7 SCOPE OF THE STUDY

1.8 LIMITATION OF THE STUDY

1.9 DEFINITION OF TERMS

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 INTRODUCTION

2.2 A REVIEW OF DATABASE TECHNOLOGY

2.3 INFORMATION SYSTEM REVIEWS

2.4 THE IMPORTANCE AND BENEFIT OF INFORMATION

2.5 REVIEW OF COMPUTER SOFTWARE SYSTEM

2.6 REVIEW OF SOFTWARE ENGINEERING
2.7 SOFTWARE DEVELOPMENT PROJECT
2.8 THE CONCEPT OF DATA PROCESSING
2.9 THE FUNCTION OF DATABASE IN INFORMATION SYSTEM
2.10 DATABASE MANAGEMENT SYSTEM
2.11 SOFTWARE DEVELOPMENT AND IMPLEMENTATION TOOLS
2.12 REVIEW OF OBJECT ORIENTED PROGRAMMING (OOP) TECHNOLOGY

CHAPTER THREE
3.0 DESIGN PROCEDURE
3.1 INTRODUCTION
3.2 RESEARCH AND DESIGN
3.2.1 CASE STUDY RESEARCH DESIGN
3.3 AREA OF THE STUDY
3.4 POPULATION OF THE STUDY
3.5 SAMPLING TECHNIQUE EMPLOYED
3.5 SAMPLE OF THE STUDY
3.6 INSTRUMENTS FOR DATA COLLECTION
3.7 INSTRUMENT USED FOR DATA ANALYSIS
3.8 DATA PRESENTATION AND ANALYSIS
3.9 DATABASE STRUCTURE
3.10 SYSTEM REQUIREMENT

CHAPTER FOUR
4.0 CONSTRUCTION AND PACKAGING
4.1 INTRODUCTION
4.2 SAMPLE OUTPUT
4.3 PROGRAM CODE LISTING
4.4 SYSTEM FLOWCHART
4.5 PROGRAM FLOWCHART
4.6 OPERATIONAL MANUAL

CHAPTER FIVE

5.0 SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 INTRODUCTION

5.2 SUMMARY

5.3 CONCLUSION

5.4 RECOMMENDATION

REFERENCES

ABSTRACT

This project is aimed at eliminating the stress normally imposed on both structures in final year and lecturers on selecting project topic. And also implementing a database that allows you to have access to approved project topics.

A sort of mini computerized library that comprises all the required information and also certain a link to approved projects topics submitted to the Departments.

LIST OF TABLES

TABLE **PAGES**

Table 3.1: Sample size population.....

Table 3.2: Sample opinions on the inefficiency of the record management of projects records in the institution...

Table 3.3: Contingency table.....

Table 3.4: Sample opinions on the need to have a computerized projects database system in the institution

Table 3.5: Contingency table

Table 3.6: Graduate

Table 3.3: Projects

CHAPTER ONE

1.0 INTRODUCTION

1.1 BACKGROUND OF THE PROBLEM

Undoubtedly, computer technology has gained enormous grounds and impact today as it forms the platform for various multi dimensional developments in vast ramifications human endeavors; this is because computers have found wide applications in almost every aspect of business, industry, education, government, and socio-economic activities. One of the widely used applications of computers is in the area of data processing and manipulations; which makes it an imperative tool in a world of endless data management and processing functions.

In Nigeria today, computerization is fast gaining ground as most of its records and data processing operations are giving a global seal of computerization. Certainly, every research oriented institution and organizational bodies' keeps electronic information for the purpose of referencing, review, evaluations, processing et cetera. Thus, the method employed for the management of these essential tasks has proven ineffective and obsolete; the organization of records and information with the use of jackets, files, file cabinets has remained an ancient monument, which does not concur with modern data processing and management standards.

In perspective, this research work is aimed at providing a student project database management system for Kaduna Polytechnic graduates project record system.

1.2 STATEMENT OF THE PROBLEM

Record management systems are changing drastically in today's technological dispensation; however, keeping tracks of records manually in files, cabinets and jackets has proved to be ineffective, because of the large volume of records of students' projects involved, considering the fast growth of the institutions The management of projects information in the Library of Kaduna Polytechnic (Isa Kaita Library) has been in constant utilization of the manual method of record management whereby cabinets and file jackets are used for all forms of file processing activities like storing, sorting retrieving of file record.

Undoubtedly, the use of this manual system of record management has been an evident problem in the institution. Thus the setbacks are enumerated below;

1. Duplication of data files which if not updated leads to inconsistency of files and wastage of office stationeries.
2. Prone to human error: The manual system in use leads to data errors as recording data is a lengthy process; it makes the whole exercise monotonous boring and hectic.
3. Misplacement/loss of projects files (records), which automatically eliminates the particular student project from being reviewed.
4. Time factor: Due to large number of students' projects available in the institutions' library, it takes more time to process any piece of

work (information) and this brings about delay in getting necessary information on demand.

5. Cross reference: -It is often difficult to cross reference between files because of the file system arrangement and this leads to redundancy of some data files within the system.

6. Security factor: Security of files is not guaranteed in the current system, there is possibility of unauthorized persons having access to student projects files.

7. Lastly, no reports is generated by this system and graduates projects information are not made available on demand; this is because it is not easy retrieving a file of a particular graduate research work on demand since the process would involve going through voluminous files.

In view of all these overwhelming problems marred by the current system or records and information management, it becomes imperative to embrace the concept of the design and development of a student project database management system for the institution.

1.4 OBJECTIVES OF THE RESEARCH

The main objective of the research work is to design and implement a robust student project database management system, however find below a list of explicated sub-objectives of the research: -

1. To ensure that graduates projects records are updated regularly in the institutions' library i.e. as at when due either bi-annually, monthly, and weekly as the need arises and this will aid efficiency.

2. To ensure that the institution's library has all the necessary information about graduates projects, such as the topic, researcher, year, synopsis, supervisors etc.

3. To ensure that graduates projects records are retrieved easily, accessibility is made easy and comprehensive statistical reports are made available on demand either by the management, students or external researchers.

4. To ensure that past graduates projects records can be made available on demand (easy accessibility) for either review or referencing purposes.

5. To eliminate inconsistency, redundancy, duplication of data files and to enhance integrity and security of data files.

6. To design a computerized database system that will enable an effective management of the university's graduates projects information for the library.

1.5 SIGNIFICANCE OF THE STUDY

The significances of this project are: -

1. The system would enable the library management have an up-to-date record about all graduates projects information which will in turn enhance the management of these peculiar information.

2. By computerizing the graduates' projects information, the technological and computer appreciation rating of the institution will be enhanced; thus, setting a pace for other tertiary institutions in the country to imitate.

3. Computerization generally would result in improvement of the operation of the library projects management system and institution since it would be timesaving, less strenuous, and less prone to errors, in view of the fact that most of the manual tasks would be eliminated and this would aid decision-making and successful goal achievement.

4. To cap it all, it would integrate the entire record keeping standard to modern standards of what is obtained in other educational institution.

1.6 RESEARCH QUESTIONS

1. Are graduates projects records updated? How often is it done? Monthly, weekly, yearly.
2. Does the institution's library have all the facts about all graduates projects that are available in the library?
3. Do you have difficulties retrieving graduates projects records and producing comprehensive reports on demand?
4. Can past projects records be made available on demand?
5. Are there instances where there are inconsistency, redundancy, insecurity and duplication of data files?

1.7 SCOPE OF THE STUDY

The scope of the study is to create a project database for the institution's library. The software system would include comprehensive information on graduates' projects, which will enable an effective storage, retrieval and overall management procedures of the institutions graduate database system.

1.8 LIMITATION OF THE STUDY

This research work is limited to building a comprehensive projects database system for data entry, storage, retrieval and updating of projects records of students in the College of Science and Technology (CST) in the institutions library, the proposed software to be developed will not include the operations of the library, neither does it store the entire contents of any given project work as it will only capture information about the project such as the title of the project, abstract/synopsis, student information and project evaluation details.

1.9 DEFINITION OF TERMS

1. Auditing: A formal examination, correction, and official endorsing of financial accounts, especially those of a business, undertaken annually by an accountant.

2 Computerized: The implementation of computer software and hardware in the organization and processing of processes. The processing of information using the computer system.

3 Database Management System: They are computer programs developed and configured to create, store, and manipulate databases.

4 Inventory: list of things, especially items of property, assets, or other resources that are stocked.

5 Information: Information can be described as any organized computer data. The meaningful material derived from computer data by organizing it and interpreting it in a specific way.

6 Information Technology: The use of technologies from computing, electronics, and telecommunications to process and distribute information in digital and other forms.

7 Management: The organizing and controlling of the affairs of a business or a sector of a business.

8 Network: A system of two or more computers, terminals, and communications devices linked by wires, cables, or a telecommunications

system in order to exchange data. The network may be limited to a group of users in a local area network, or be global in scope, as the Internet is.

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 INTRODUCTION

The main objective of this research work is to design, code test and implement a project database management system for the institution's library, which will help eliminate the problems that have been discussed in the preceding chapter. Therefore, this chapter attempts to thoroughly review relevant literatures that are of great significance to the research under study.

2.2 A REVIEW OF DATABASE TECHNOLOGY

The concept and application of database system have evolved over the years to improve and enhance the way in which records and information are managed in different data and record processing tasks. Beyond doubt, the most valuable possession of any organization or institution is the information at its disposal; information forms the basis for the processing operations and activities carried out in any organization because it forms the platform for the realization of the purpose of establishing such scheme and helps management in making calculated and improvised decisions.

Database Technology includes theory and experimental methodology for building computer systems that handles large data volumes. Central is development of concepts, languages, software, and methods for describing, storing, searching, analyzing, distributing, and other data processing to make access of data simple, efficient, scalable, reliable, and adaptable for new application areas.

Database research deals with storing, searching, and analyzing very large information volumes. The area has been very influential for today's IT-systems. Internet created vastly increased demand for systems to handle large data volumes, DataBase Management Systems (DBMS). Three important technical achievements have driven the technology: fast and efficient data storage, query languages for searching data, and data distributed over many computers. Advanced queries are central in modern software for inventory control, analysis, prognosis, planning, etc. DBMSs are inside most large systems. New applications put new requirements on the technology, e.g., for scalable search and analyzes in scientific and technical applications within physics, medicine, and engineering.

New applications require management of large information volumes stored in different formats and sources. E.g., some data may be stored in conventional relational databases while other data are produced as streams from sensors.

2.3 INFORMATION SYSTEM REVIEWS

Becker (1989) represented an information system in a broader capacity as a system, which stores information and data and also makes these available to users. Currently, the efficient and effective collection, organization, documentation and retrieval of data frequently requires a data processing system with real-time processing facilities and direct access to databases, thus improving the way in which data are been manipulated.

2.4 THE IMPORTANCE AND BENEFIT OF INFORMATION

DENNIS et al (1995), explains information as any communication or reception of knowledge such as facts, data or opinion, including numerical, graphics or maintained in any medium including computerized database, paper, microform or magnetic tapes.

Undoubtedly, information remains a very important tool in every educational system and the need to organize effectively all educational based information such as the graduates' projects details cannot be ignored.

2.5 REVIEW OF COMPUTER SOFTWARE SYSTEM

Bustard (2000), Explains software as computer programs; instructions that cause the hardware (the machines) to do work. He categorized software as a whole into a number of categories based on the type of work done. The two primary software categories are operating systems (system software), which control the workings of the computer, and application software, which addresses the multitude of tasks for which people use computers. System software thus handles such essential, but often invisible, chores as maintaining disk files and managing the screen, whereas application software performs word processing, database management, and the like. Two additional categories that are neither system nor application software, although they contain elements of both, are network software, which enables groups of computers to communicate, and language software, which provides programmers with the tools they need to write programs.

In addition to these task-based categories, Bustard listed several types of software as described based on their method of distribution. These include the so-called canned programs or packaged software developed and sold primarily through retail outlets; freeware and public-domain software, which is made available without cost by its developer; shareware, which is similar to freeware but usually carries a small fee for those who like the program; and the infamous vaporware, which is software that either does not reach the market or appears much later than promised.

Bitter (1986) describes software as the Instructions that humans write to tell the computer how to do jobs and perform certain operations. He explained that the computer needs the hardware and software in order to work properly. It needs software to tell it what to do and hardware to actually carry out the work.

Furthermore to Bitter (1986), software is another name for Computer programs. Programs are actual instructions that the user gives the computer. People who write these instructions are called programmers. A program might tell the computer how to calculate an employee's paycheck or locate a book in the school library; thus, they are important links between the computer and the users.

2.6 REVIEW OF SOFTWARE ENGINEERING

Ulster (2000) identified Software Engineering as the discipline of developing software products. A software product is the complete set of computer

programs, procedures, documentation, and data specified for delivery to a customer; development is concerned with all technical and managerial activities necessary to create the product; and carrying out the development effectively means meeting customer requirements within constraints of time, cost, and quality.

Additionally, Ulster (2000) described the software engineering as a concept that emerged from series of workshops conducted by the North Atlantic Treaty Organization (NATO) in 1968 to consider what was then described as the "software crisis". Too many software development projects were failing, and the failures were attributed to rapid increases in the scale and complexity of the software concerned. It was recognized that a more systematic approach to software development was needed and that it should be based on established engineering principles.

2.7 SOFTWARE DEVELOPMENT PROJECT

Bustard (2000) explains that Software evolves through many versions in its lifetime, as errors are corrected, operation is enhanced, and changing requirements are encountered and dealt with. This work is often referred to as "maintenance". Each new version is created through a software development process. Typically, this is divided into four main phases: (1) requirements analysis and specification, which establishes what the software product is to achieve; (2) design, which determines how the software product will meet its requirements; (3) implementation, which creates the software product as designed (this combines the development of new components with the reuse

or modification of existing ones); and (4) testing, which ensures that the software product operates as required. Intermediate products, such as requirements specifications and software designs, are also reviewed thoroughly as a basis for moving from one development phase to another.

He however, said that Software has not always been developed in a controlled manner and some systems are now proving very difficult to maintain. The International Standards Organization (ISO) has defined requirements for a general Quality Management System (QMS) covering all product development (ISO 9001) and has specific guidelines for the application of this standard to software development (ISO 9000-3). An organization implementing a QMS to this standard can be audited and can receive formal certification for its development process.

Software engineers are involved in a large and ever-widening range of application areas. Examples include supporting rapid share transactions in the stock market; facilitating information storage, exchange, and presentation over the World Wide Web; providing entertainment through video games; enhancing images from astronomical telescopes; controlling heart pacemakers; and so on. In all cases, software engineering principles help to ensure that the resulting systems are reliable and function as required.

2.8 THE CONCEPT OF DATA PROCESSING

Encarta (2005) describes Data Processing as the analysis and organization of data by the repeated use of one or more computer programs. Data processing is used extensively in business, engineering, and science, and to an increasing extent in nearly all areas in which computers are used. Businesses use data processing for such tasks as payroll preparation, accounting, record keeping, inventory control, sales analysis, and the processing of bank and credit card account statements. Engineers and scientists use data processing for a wide variety of applications, including the processing of seismic data for oil and mineral exploration, the analysis of new product designs, the processing of satellite imagery, and the analysis of data from scientific experiments.

Data processing is divided into two kinds of processing:

a. Database processing and

b. Transaction processing.

2.8.1 DATABASE BASIC IN DATA PROCESSING

A database is a collection of common records that can be searched, accessed, and modified, such as bank account records, school transcripts, and income tax data. In database processing, a computerized database is used as the central source of reference data for the computations. Transaction processing refers to interaction between two computers in which one computer initiates a transaction and another computer provides the first with the data or computation required for that function.

Most modern data processing uses one or more database at one or more central sites. Transaction processing is used to access and update the databases when users need to immediately view or add information; other data processing programs are used at regular intervals to provide summary reports of activity and database status. Examples of systems that involve all of these functions are automated teller machines, credit sales terminals, and airline reservation systems.

2.8.2 THE DATA-PROCESSING CYCLE

The data-processing cycle represents the chain of processing events in most data-processing applications. It consists of data recording, transmission, reporting, storage, and retrieval. The original data is first recorded in a form readable by a computer. This can be accomplished in several ways: by manually entering information into some form of computer memory using a keyboard, by using a sensor to transfer data on to a magnetic tape or floppy disk, by filling in ovals on a computer-readable paper form, or by swiping a credit card through a reader. The data are then transmitted to a computer that performs the data-processing functions. This step may involve physically moving the recorded data to the computer or transmitting it electronically over telephone lines or the Internet.

Once the data reach the computer, the computer processes the information. The operations the computer performs can include accessing and updating a database and creating or modifying statistical information. After processing the data, the computer reports summary results to the operator of the program.

As the computer processes the data, it stores both the modifications and the original data. This storage can be both in the original data-entry form and in carefully controlled computer data forms such as magnetic tape. Data are often stored in more than one place for both legal and practical reasons. Computer systems can malfunction and lose all stored data, and the original data may be needed to recreate the database as it existed before the crash.

The final step in the data-processing cycle is the retrieval of stored information at a later time. This is usually done to access records contained in a database, to apply new data-processing functions to the data, or—in the event that some part of the data has been lost—to recreate portions of a database. Examples of data retrieval in the data-processing cycle include the analysis of store sales receipts to reveal new customer spending patterns and the application of new processing techniques to seismic data to locate oil or mineral fields that were previously overlooked.

2.9 THE FUNCTION OF DATABASE IN INFORMATION SYSTEM

James (1994) defines database system as a collection of data on computer: a systematically arranged collection of computer data or information, structured so that it can be automatically retrieved or manipulated. As a matter of fact, the information concerned can be anything that is defined to be of significance to the organization system of serving that is, any thing that may be necessary to the decision making process involved in the management of that organization. In actual fact, the components of a database are the same records and files in traditional files; the only difference is the integration of

these files. A DBMS might use sequential, direct, indexed sequential, or standard

Mark (1984) defines a database as a collection of all data within a particular area or application. He establishes that the database should be independent of the application programs, which access, updates, or change the data items. These are stored in such a way that relations between them can be established. All information system in its complexity needs a good database management system in order to store and retrieve data and information effectively. Characteristically databases hold huge quantities of data, which have to be stored in complex ways in order to enable fast for the maximum proportion of queries.

2.10 DATABASE MANAGEMENT SYSTEM

Bambooweb (2002) describes a database management system (DBMS) as a computer program (or more typically, a suite of them) designed to manage a database, a large set of structured data, and run operations on the data requested by numerous users. Typical examples of DBMS use include accounting, human resources and customer support systems. Originally found only in large companies with the computer hardware needed to support large data sets, DBMSs have more recently emerged as a fairly standard part of any company back office.

DBMS's contrast with the more general concept of database applications in that they are designed as the "engine" of a multi-user system. In order to fill

this role, DBMSs are typically built around a private multitasking kernel with built-in networking support. A typical database application will not include these features internally, but may be able to support similar functionality by relying on the operating system to provide these features for it.

2.11 SOFTWARE DEVELOPMENT AND IMPLEMENTATION TOOLS

Visual Basic 6 was the programming languages used in the design of the front end application of the Project database Management System for Kaduna Polytechnic. Hence, this section of the chapter attempts to present an overview of the programming language platform.

2.11.1 AN OVERVIEW OF MICROSOFT VISUAL BASIC 6

Visual Basic (**VB**) is the third-generation event-driven programming language and integrated development environment **(IDE)** from Microsoft for its COM programming model. VB is also considered a relatively easy to learn and use programming language, because of its graphical development features and BASIC heritage.

Visual Basic was derived from BASIC and enables the rapid application development (RAD) of graphical user interface (GUI) applications, access to databases using Data Access Objects, Remote Data Objects, or ActiveX Data Objects, and creation of ActiveX controls and objects. Scripting languages such as VBA and VBScript are syntactically similar to Visual Basic, but perform differently.

A programmer can put together an application using the components provided with Visual Basic itself. Programs written in Visual Basic can also

use the Windows API, but doing so requires external function declarations. The final release was version 6 in 1998. Microsoft's extended support ended in March 2008 and the designated successor was Visual Basic .NET (now known simply as Visual Basic).

2.11.2 LANGUAGE FEATURES

Like the BASIC programming language, Visual Basic was designed to be easy to learn and use. The language not only allows programmers to create simple GUI applications, but can also develop complex applications. Programming in VB is a combination of visually arranging components or controls on a form, specifying attributes and actions of those components, and writing additional lines of code for more functionality. Since default attributes and actions are defined for the components, a simple program can be created without the programmer having to write many lines of code. Performance problems were experienced by earlier versions, but with faster computers and native code compilation this has become less of an issue.

Although programs can be compiled into native code executables from version 5 onwards, they still require the presence of runtime libraries of approximately 1 MB in size. This runtime is included by default in Windows 2000 and later, but for earlier versions of Windows like 95/98/NT it must be distributed together with the executable.

Forms are created using drag-and-drop techniques. A tool is used to place controls (e.g., text boxes, buttons, etc.) on the form (window). Controls have

attributes and event handlers associated with them. Default values are provided when the control is created, but may be changed by the programmer. Many attribute values can be modified during run time based on user actions or changes in the environment, providing a dynamic application. For example, code can be inserted into the form resize event handler to reposition a control so that it remains centered on the form, expands to fill up the form, etc. By inserting code into the event handler for a keypress in a text box, the program can automatically translate the case of the text being entered, or even prevent certain characters from being inserted.

Visual Basic can create executables (EXE files), ActiveX controls, or DLL files, but is primarily used to develop Windows applications and to interface database systems. Dialog boxes with less functionality can be used to provide pop-up capabilities. Controls provide the basic functionality of the application, while programmers can insert additional logic within the appropriate event handlers. For example, a drop-down combination box will automatically display its list and allow the user to select any element. An event handler is called when an item is selected, which can then execute additional code created by the programmer to perform some action based on which element was selected, such as populating a related list.

Alternatively, a Visual Basic component can have no user interface, and instead provide ActiveX objects to other programs via Component Object Model (COM). This allows for server-side processing or an add-in module.

The language is garbage collected using reference counting, has a large library of utility objects, and has basic object oriented support. Since the more common components are included in the default project template, the programmer seldom needs to specify additional libraries. Unlike many other programming languages, Visual Basic is generally not case sensitive, although it will transform keywords into a standard case configuration and force the case of variable names to conform to the case of the entry within the symbol table entry. String comparisons are case sensitive by default, but can be made case insensitive if so desired.

The Visual Basic compiler is shared with other Visual Studio languages (C, C++), but restrictions in the IDE do not allow the creation of some targets (Windows model DLL's) and threading models.

2.12 REVIEW OF OBJECT ORIENTED PROGRAMMING (OOP) TECHNOLOGY

Object Oriented Programming (OOP) is described as a system of programming that permits an abstract, modular typing hierarchy, and features polymorphism, inheritance, and encapsulation. Three basic features of Object Oriented Programming: -

1. **Polymorphism**: An object-oriented programming term that is used to refer to the ability to have methods with the same name, but different content, for related classes. The procedure to use is determined at run time by the class of the object. For example, related objects might both have Draw methods. A procedure, passed such an object as a

parameter, can call the Draw method without needing to know what type of object the parameter is.

2. **Inheritance:** An object-oriented programming term used to refer to the ability of a subclass to take on the characteristics of the class it's based on. If the characteristics of the parent class change, the subclass on which it is based inherits those characteristics. For example, if you add a new property, Is Bold, to an editing control, any subclasses based on your control will also have an IsBold property.

3. **Encapsulation:** An object-oriented programming term for the ability to contain and hide information about an object, such as internal data structures and code. Encapsulation isolates the internal complexity of an object's operation from the rest of the application. For example, when you set the Caption property on a command button, you don't need to know how the string is stored.

CHAPTER THREE

3.0 DESIGN PROCEDURE

3.1 INTRODUCTION

In this chapter the research method used to derive answers to the research question will be discussed. Areas of importance that will be treated effectively are the instruments used to conduct the study, the procedure followed, and the research design and data analysis, system requirement specifications and also the database structure.

3.2 RESEARCH AND DESIGN

A Research can be described as an enquiry into the unknown; it involves the search for solutions to problems or answers to questions. Comparatively, Research design is a plan that specifies how data relating to a given problem should be collected and analyzed. It provides the procedural outline for the conduct and procedure of any given investigation.

3.2.1 CASE STUDY RESEARCH DESIGN

The Case Study Research Design has been employed for the development of this research work in which Kaduna Polytechnic graduate projects record system was considered with both students and staff sample opinion on the importance to have a comprehensive yet functional projects database information system.

3.3 AREA OF THE STUDY

The area of the study of this research is Kaduna Polytechnic.

3.4 POPULATION OF THE STUDY

The population of the study consists of all 630 CST graduating students and 20 library staff of the Isa Kaita Library, Kaduna Polytechnic.

3.5 SAMPLING TECHNIQUE EMPLOYED

The Simple Random Sampling Technique is the sampling technique used in this research study. In the application of this sampling technique, every one component or entity of the population has equal and independent possibility of being included in the simple random sampling technique. For example a population size of 1000 elements, the probability or chance of drawing each element is 1/1000. Samples resulting from the application of this procedure are said to be unbiased and are therefore representative of the population.

Nonetheless, the simple random sampling is the easiest and simplest probability sampling technique, in terms of conceptualization and application. It does not inevitably require knowledge of the exact composition of the population, so long as we reach all member of the population.

3.5 SAMPLE OF THE STUDY

A sample size n is taken from the population by using stratified random sampling technique. Each respondent category representing a stratum, a 30% sample size n is distributed in the two stratums i.e. 20 Staff of Isa Kaita Library, Kaduna Polytechnic and 630 Students from College of Science and Technology (CST)

The distribution of the two stratums is shown below.

N = the population

n = Sample size of N

nh = Sample size of each stratum

Nh = Total number of staff in each stratum

30% of N = n

n = 20/100 X 30/1 = 6

n = 6

h = 1(1st Stratum)

n1 = nHn/N = the population from the first stratum

N1 = the total number of staff in the first stratum = 6

30% of N = n

n = 630/100 X 30/1 = 189

n = 189

h2 = 2(2nd Stratum)

n2 = nHn/N = the population from the second stratum

N2 = the total number of students in the

Second stratum = 54

TABLE 3.1

Strata	Population Size	Number administered	Percentage (%)	Total
Staff	20	6	30%	60
Students	630	189	30%	

3.6 INSTRUMENTS FOR DATA COLLECTION

A fundamental aspect of any research work is the data collection process. It involves the instruments and procedures for uncovering the information demanded by a problem. Nonetheless, the methods utilized for the gathering of data were basically reading, interviewing and the questionnaires methods.

a. **QUESTIONNAIRE:** This gathered data on individual opinion and idea. The survey questionnaire provides meaningful responses during the evaluation process. The questionnaire was used to determine the current status of opinions, intentions and motives of the respondent specifically the open-end questions used whereby the researcher asked only questions pertinent to the problem.

b. **READING:** A secondary research technique which involves the gathering of relevant information from secondary sources like newspapers, related journal, textbooks, magazines, past projects and other relevant sources. In this research, books related to algebra were discreetly studied and analyzed.

c. **INTERVIEWING:** An interview is a conversation between two or more people (the interviewer and the interviewee) where questions are asked by the interviewer to obtain information from the interviewee. In this research, it was used to obtained first hand information relating to the project scope.

3.7 INSTRUMENT USED FOR DATA ANALYSIS

The chi-square (χ^2) test is the statistical tool applied to draw inference and conclusions on the data collected. The chi-square test is a non-parametric inferential statistical method use in the analysis of frequencies or nominal data.

3.7.1 A Non-Parametric Statistics

As a non-parametric statistics, it makes no restrictive assumptions about the distribution of scores in question and so it can use where the assumptions of parametric statistics about the distribution are not satisfied.

Accordingly, this statistic has found widespread applications in the field of education and other behavioral sciences, particularly in the analysis of data in the form of frequencies or categories.

3.7.2 A Two-Tailed Test

The chi-square is a two-tailed test; it can only indicate whether or not a set observed frequencies differ significantly from the corresponding set of expected frequencies not possibly the direction in which they differ. The general formula for the computation of the chi-square statistic is given by:

$$\chi 2 = \frac{\sum (O - E)2}{E}$$

Where

O = Observed frequency

E = Expected or theoretical frequency

Σ = "Sum of or summation"

The formula above implies that we have to determine the expected or theoretical frequencies first. The expected (or theoretical) frequencies are those frequencies which occur the null hypothesis, while the observed frequencies correspond to the frequencies obtained by direct observation of the phenomenon or event under consideration. Having obtained the expected frequencies we next calculate the square of the difference between the observed and expected frequencies. These squared differences are divided by the corresponding expected frequencies and the ratios summed up to get χ^2.

The calculated χ^2 is then compared with the critical or table χ^2. If the calculated value of the calculated value of the χ^2 exceeds the critical value, then we reject the null hypothesis; otherwise, we do not reject the null hypothesis. To find the critical or table value, we have to (as in the case of the t-statistic) decide on an appropriate alpha level (level of significance) and obtain the associated degree of freedom. Critical values of x^2 can be obtained from the sampling distribution table of χ^2 for df $\leq$ 30 for df > 30; the formula below, as an approximately normal sampling distribution, is used.

$$Z = \sqrt{2\chi 2} - \sqrt{2\,df - 1}$$

Since the sampling distribution is approximately normal we now compare whether value we get from this formula with the appropriate critical value of the z-statistic.

Chi-square is usually applied in two type of test namely:-

a. *The goodness–of–fit test*

b. *The test of independence*

3.7.3 THE TEST OF INDEPENDENCE

The test of independence was employed in the data analysis of this research. In this test, two factors or variables, each having two or more levels/categories are involved and the researcher wants to test whether or not the two variables are dependent or not. The table in which the observed and expected frequencies associated with the various levels of the two variables are presented is called a Contingency Table.

3.8 DATA PRESENTATION AND ANALYSIS

TABLE 3.2:

Sample opinions on the inefficiency of the record management of projects records in the institution

POSITION	CATEGORY		
	STAFF	STUDENTS	TOTAL
Strongly Agree	3	140	**143**
Agree	2	20	**22**
Undecided	0	20	**20**
Disagree	1	9	**10**
Strongly Disagree	0	0	**0**
Total	**6**	**189**	**195**

Aim: To investigate whether respondents' views concurs with the fact that the institution's projects record management is inefficient. The Chi-square test of independence at the 5% level of significance is applied.

Hypothesis

H_0: The institution's graduate project record management system is considered inefficient.

H_1: The institution's graduate project record management system is not considered inefficient.

Level Of Significance

$\propto = 0.05$

Test Statistic

$$\chi 2cal = \frac{\sum_{i=n}^{r} \sum_{j=m}^{c} (oij - eij)2}{eij}$$

Decision Criterion

Reject H_0 if χ^2_{cal} exceeds $\chi^2_{0.05,4} = 9.48773$

TABLE 3.4:

Sample opinions on the need to have a computerized projects database system in the institution

POSITION	CATEGORY		
	STAFF	STUDENTS	TOTAL
Strongly Agree	3	140	**143**
Agree	2	20	**22**
Undecided	0	20	**20**
Disagree	1	9	**10**
Strongly Disagree	0	0	**0**
Total	**6**	**189**	**195**

Aim: To investigate whether respondents' views concurs with the fact that the institution's needs a computerized graduate projects database system for its record management system. The Chi-square test of independence at the 5% level of significance is applied.

Hypothesis

H_0: The institution needs a computerized graduate project database system for its record management system.

H_0: The institution does not need a computerized graduate project database system for its record management system.

Level Of Significance

$\propto = 0.05$

Test Statistic

$$\chi 2cal = \frac{\sum_{i=n}^{r} \sum_{j=m}^{c} (oij - eij)2}{eij}$$

Decision Criterion

Reject H_0 if χ^2_{cal} exceeds $\chi^2_{0.05,4}$ = 9.48773

Computation

The computations are summarized in the contingency table below:

TABLE 3.6

Contingency Table

O_{ij}	e_{ij}	$O_{ij}-e_{ij}$	$(O_{ij}-e_{ij})^2$	$\frac{(O_{ij}-e_{ij})^2}{e_{ij}}$
3	2.083	0.917	0.840	0.404
140	138.567	1.433	2.053	0.015
2	1.117	0.883	0.779	0.698
20	19.233	0.767	0.588	0.031
0	0.000	0.000	0.000	0.000
20	19.233	0.767	0.588	0.031
1	0.125	0.875	0.765	6.12
9	8.117	0.883	0.779	0.096
0	0.000	0.000	0.000	0.000
0	0.000	0.000	0.000	0.000
Total				**7.395**

INTERPRETATION AND CONCLUSION

Since χ^2_{cal} =7.395 does not exceed χ^2_{tab} = 9.48773 **we accept H_o.** Hence, we conclude that respondents agree with the urgent need of the institution to have a computerized project database system, which will enable an effective management of the institutions' graduates' projects records.

3.9 DATABASE STRUCTURE

A Database can be described as an organized collection of data organized for storage in a computer memory and designed for easy access by authorized users, or organized collections of related records. A database enables easy and effective manipulation of records that have been stored in a computer system; thus, the need for database in any information system or integrated information system cannot be overlooked.

The database name is **Projects**, which was created using Microsoft Access 2003 as the backend and the application program developed in Microsoft visual basic 6 as the front end;

TABLE 3.7

GRADUATE

DESCRIPTION	FIELD NAME	FIELD TYPE	SIZE
Student Registration No	RegNo	Text	15
Student Last Name	LName	Text	15
Student First Name	FName	Text	15
Sex	Sex	Text	8
Programme	Programme	Text	30

Department	Depart	Text	30
Year of Graduation	Year	Number	4

TABLE 3.8

PROJECTS

DESCRIPTION	FIELD NAME	FIELD TYPE	SIZE
Student Registration No	RegNo	Text	15
Project Topic	Topic	Text	100
Project Abstract	Abstract	Text	200
Project Category	Category	Text	30
Project Supervisor	Supervisor	Text	30
Remark	Remark	Text	30

3.10 SYSTEM REQUIREMENT

The system requirements are the minimum hardware and software requirements for the development and full implementation of the projects database system. Below are the minimum hardware and software for the development.

3.10.1 HARDWARE REQUIREMENT

The hardware requirements for the system are listed below: -

- Pentium IV 1Ghz or higher
- 15 inch Visual Display Unit (Super Video Graphic Adapter)
- 1GB RAM or higher
- UPS (Uninterruptible Power Supply)
- Power stabilizer
- Printer

3.10.2 SOFTWARE REQUIREMENT

The software requirements for the system are listed below: -

- Windows XP Operating System or higher
- Microsoft Visual Basic 6.0 developers' tools.
- Microsoft Office 2000 or higher
- Microsoft Access and SQL (Structured Query Language) server 2000.

CHAPTER FOUR

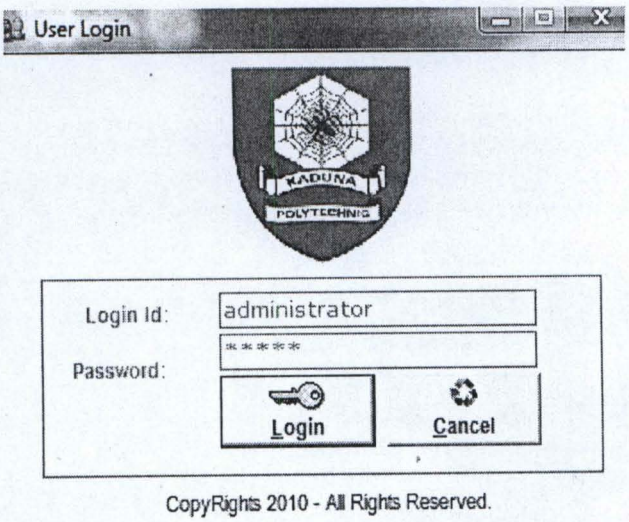
4.0 CONSTRUCTION AND PACKAGING

4.1 INTRODUCTION

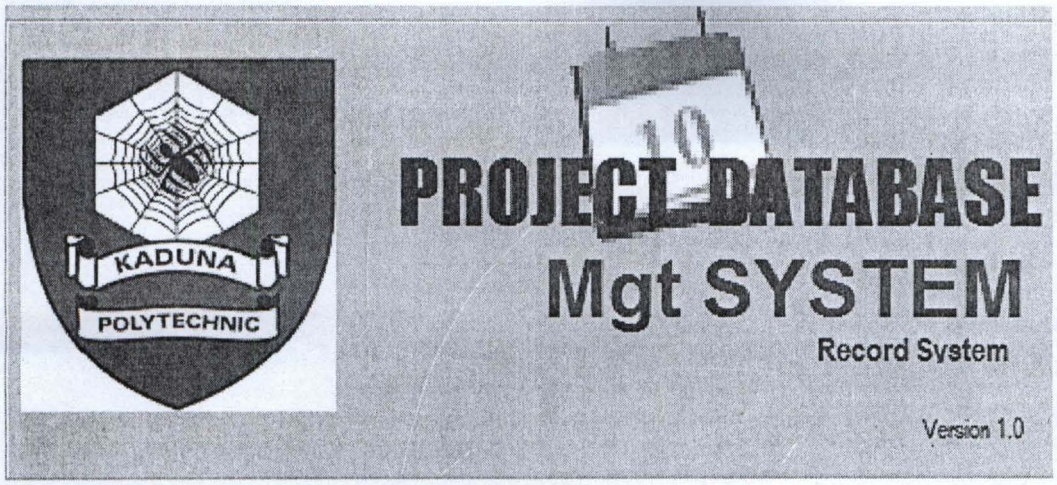
This chapter entails the system and program flowchart, program listings, sample outputs and the operational manual. However, the chapter presents a front end and back end documentation of the program development and implementation.

4.2 SAMPLE OUTPUT

SYSTEM LOGIN

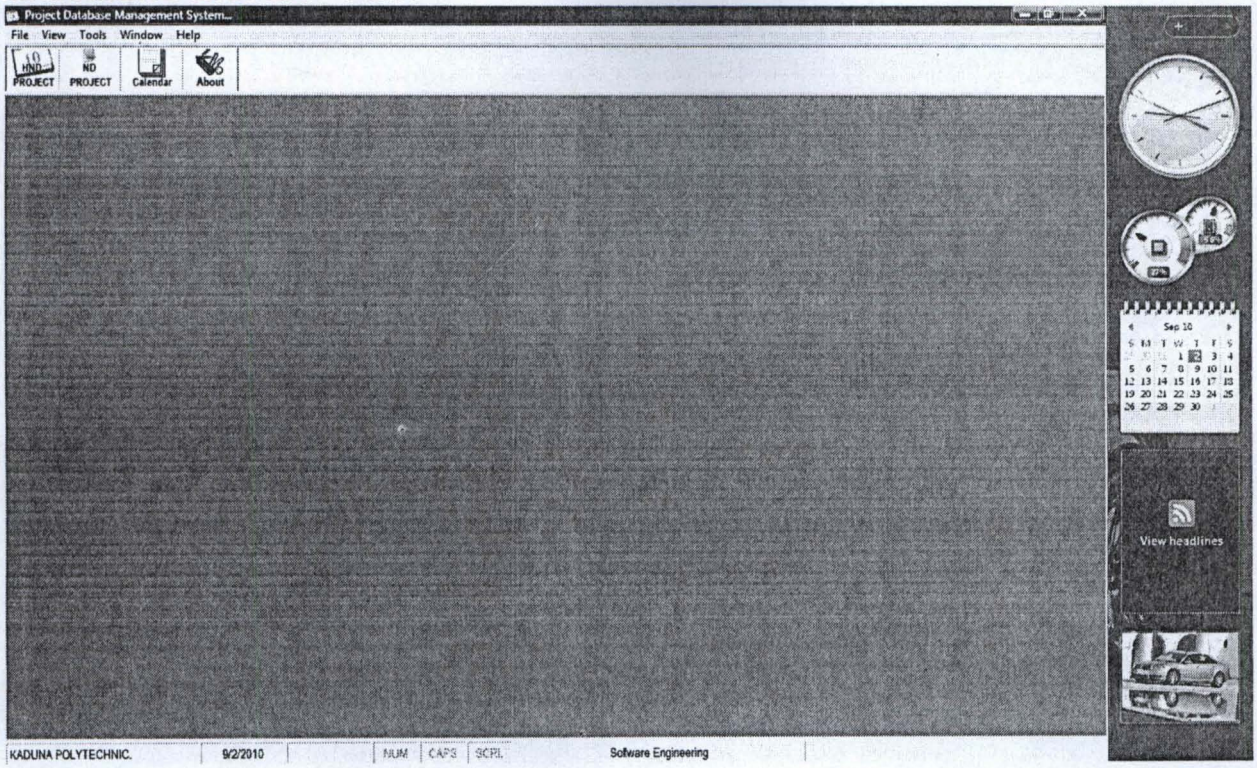


PROGRAM LOADING

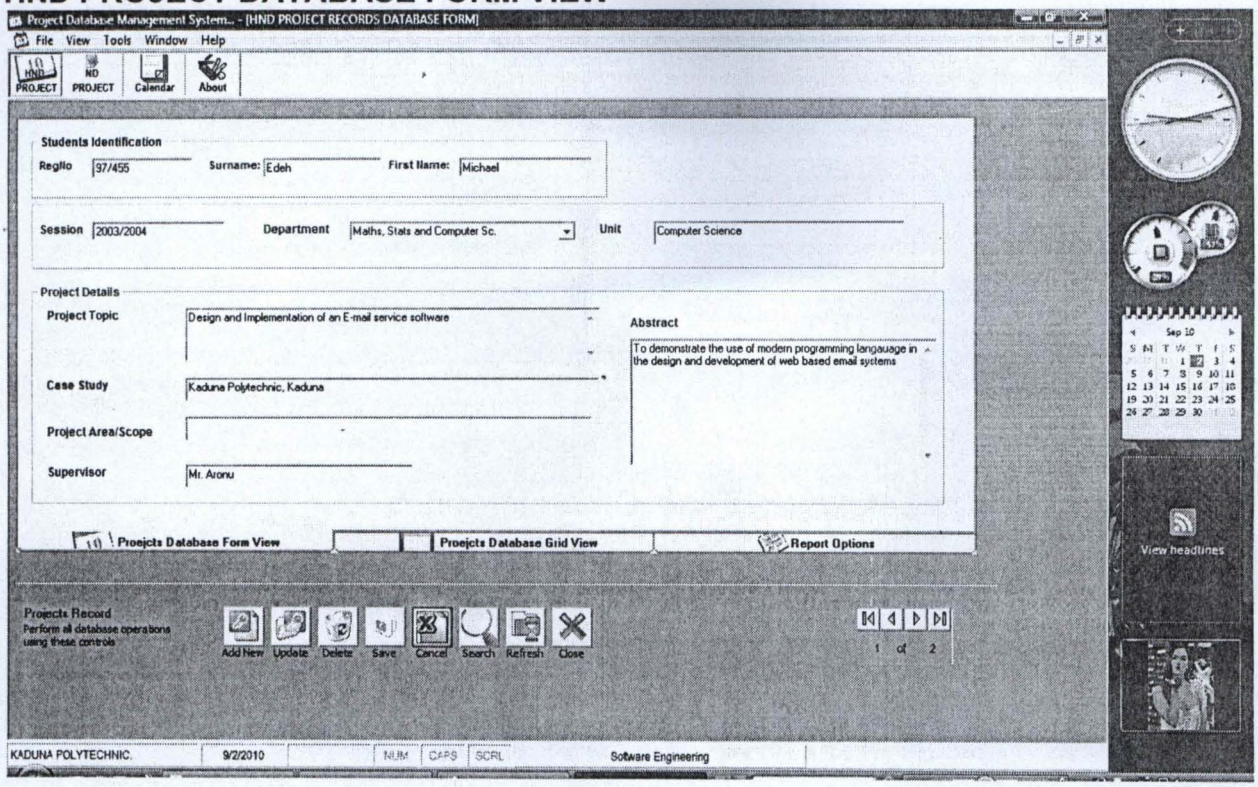


All Rights Reserved (c)2010

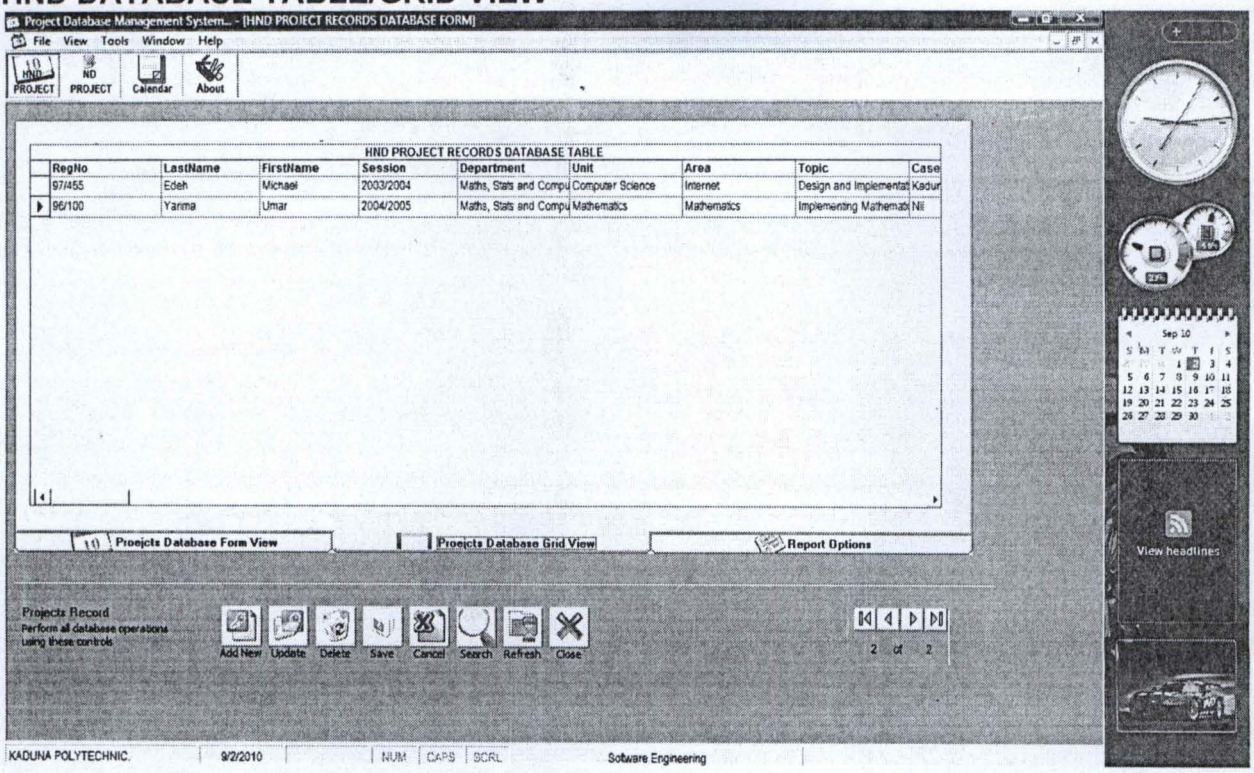
MAIN MENU DISPLAY VIEW



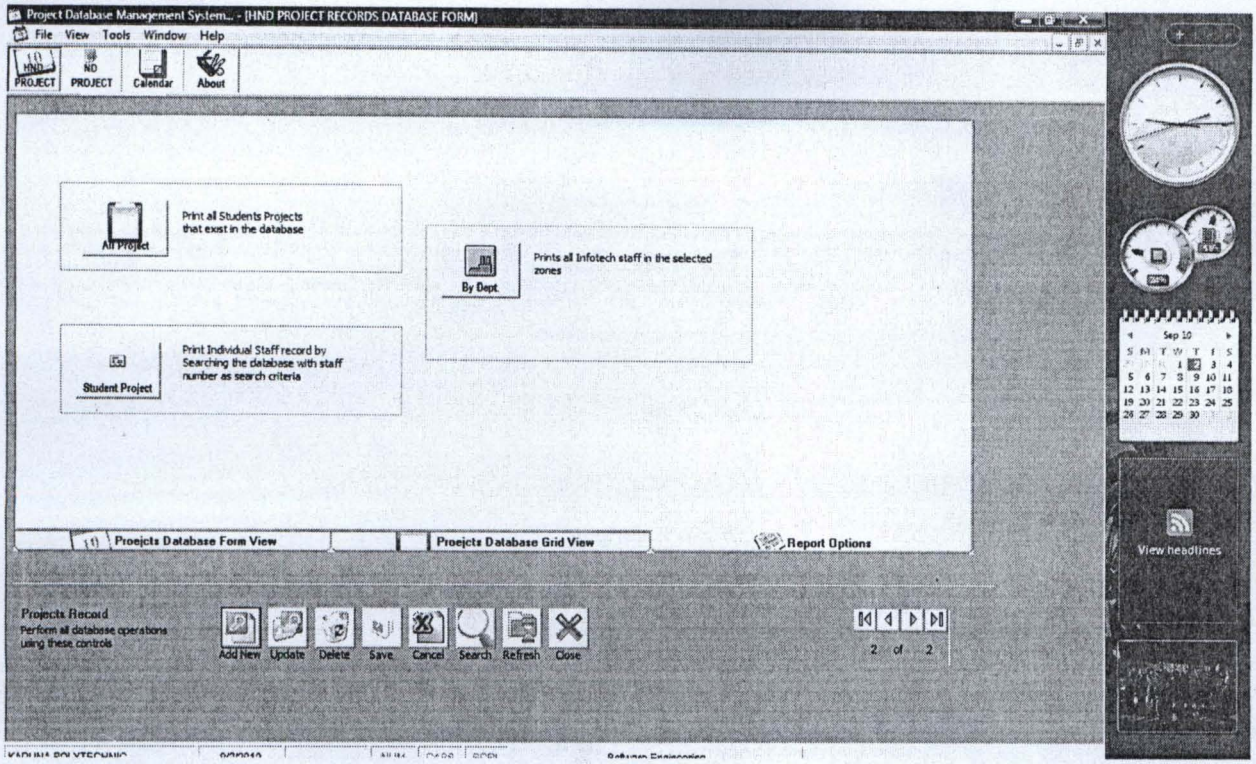
HND PROJECT DATABASE FORM VIEW



HND DATABASE TABLE/GRID VIEW



REPORT VIEW



KADUNA POLYTECNHIC

Project Database Management System

Thursday, September 02, 2010

HND PROJECT REPORT LIST

Reg. No	Surname	First Name	Department	Project Topic	Session
96/100	Yarima	Umar	Maths, Stats and Computer Sc.	Implementing Mathematics Elements	2004/2005
97/455	Edeh	Michael	Maths, Stats and Computer Sc.	Design and Implementation of an E-mail service software	2003/2004

TOTAL NUMBER OF 2

KADUNA POLYTECNHIC

Project Database Management System

Thursday, September 02, 2010

HND PROJECT REPORT

Reg. No	97/455		**Dept.**	Maths, Stats and Computer Sc.
Student Name	Edeh	Michael	**Unit**	Computer Science
			Session	2003/2004

Project Topic Design and Implementation of an E-mail service software

Abstract

To demonstrate the use of modern programming langauage in the design and development of web based email systems

Area/Scope Internet

Case Study Kaduna Polytechnic, Kaduna

Supervisor Mr. Aronu

KADUNA POLYTECNHIC

Project Database Management System

Thursday, September 02, 2010

HND PROJECT REPORT

Reg. No	97/455		**Dept.**	Maths, Stats and Computer Sc.
Student Name	Edeh	Michael	**Unit**	Computer Science
			Session	2003/2004

Project Topic Design and Implementation of an E-mail service software

Abstract

To demonstrate the use of modern programming langauage in the design and development of web based email systems

Area/Scope Internet

Case Study Kaduna Polytechnic, Kaduna

Supervisor Mr. Aronu

KADUNA POLYTECNHIC

Project Database Management System

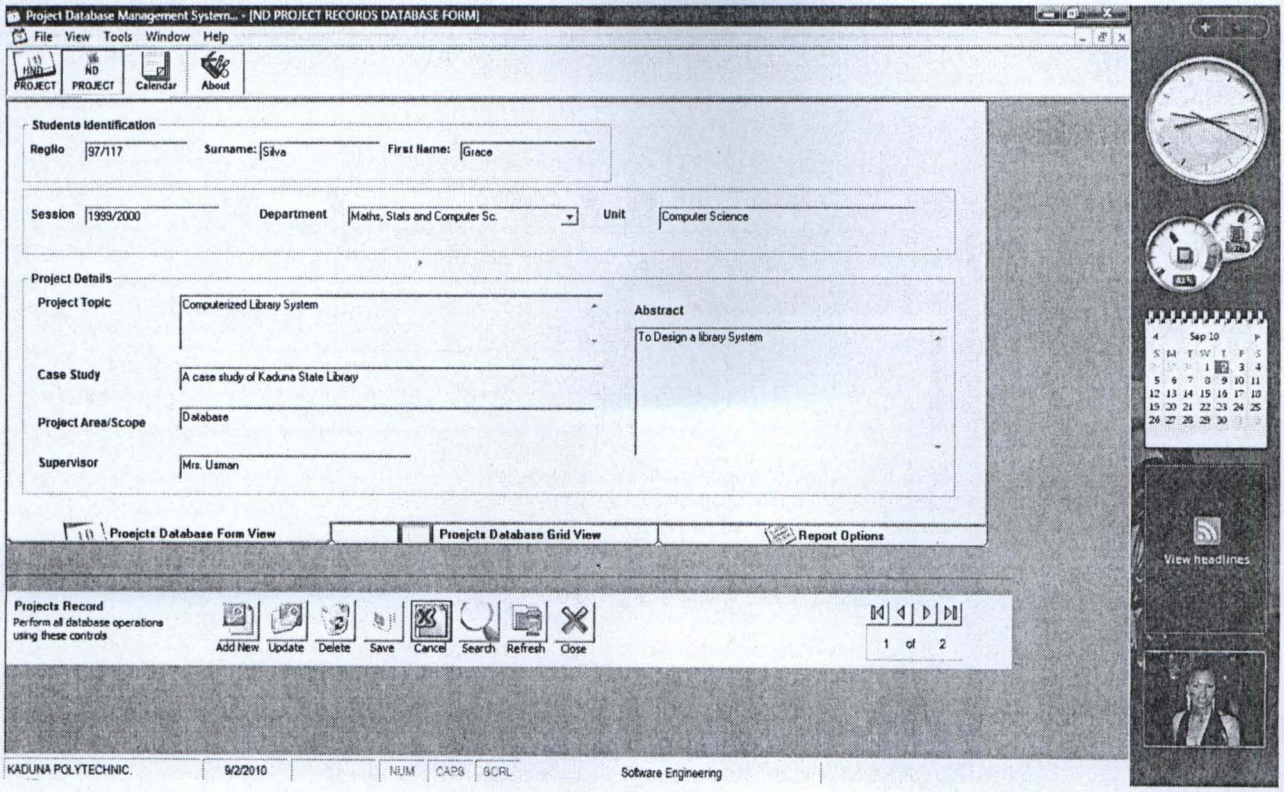
Thursday, September 02, 2010

HND PROJECT REPORT BY DEPARTMENT

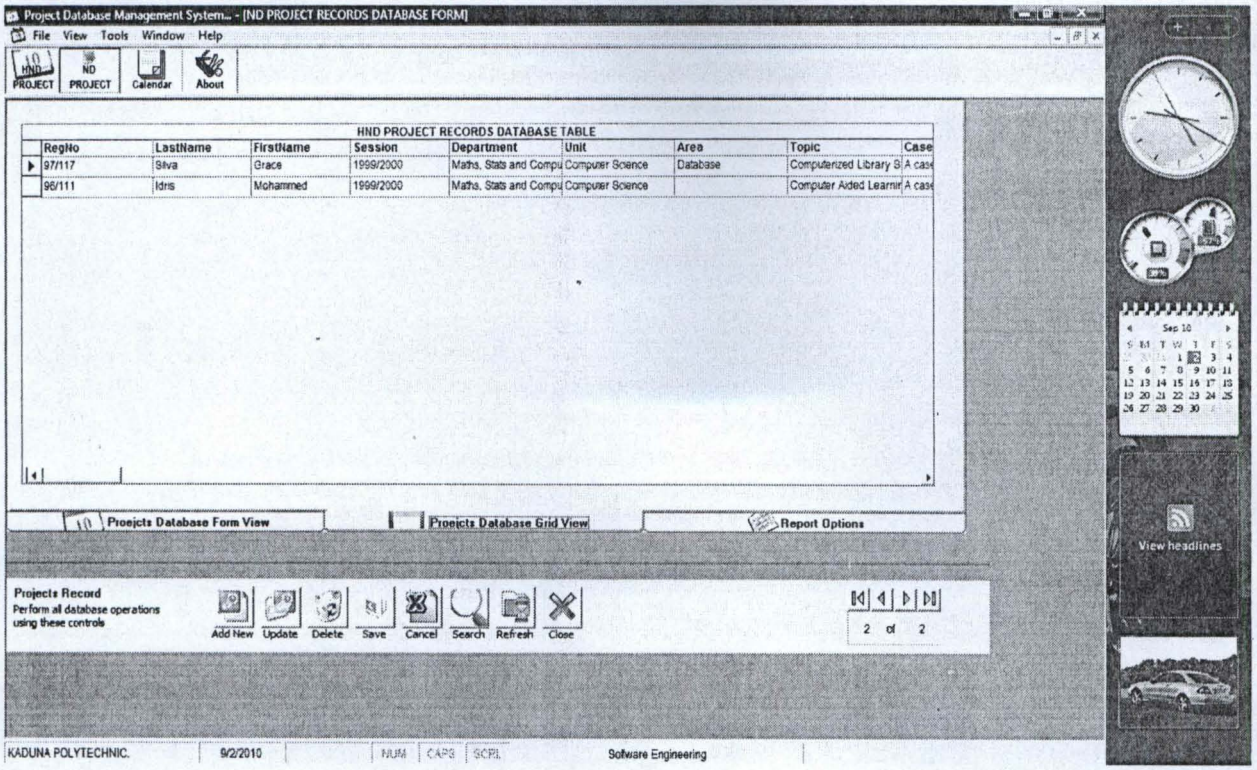
Reg. No	Surname	First Name	Department	Project Topic	Session
97/155	Edeh	Michael	Maths, Stats and Computer Sc.	Design and Implementation of an E mail service software	2003/2004
96/100	Yakima	Umar	Maths, Stats and Computer Sc.	Implementing Mathematics Elements	2004/2005

TOTAL NUMBER OF PROJECTS: 2

ND PROJECT DATABASE FORM



ND PROJECT TABLE/GRID VIEW



KADUNA POLYTECNHIC

Project Database Management System

Thursday, September 02, 2010

ND PROJECT REPORT LIST

Reg. No	Surname	First Name	Department	Project Topic	Session
96/111	Idris	Mohammed	Maths, Stats and Computer Sc.	Computer Aided Learning	1999/2000
97/117	Silva	Grace	Maths, Stats and Computer Sc.	Computerized Library System	1999/2000

TOTAL NUMBER OF PROJECTS: 2

KADUNA POLYTECNHIC

Project Database Management System

Thursday, September 02, 2010

ND PROJECT REPORT

Reg. No	97/117		**Dept.**	Maths, Stats and Computer Sc.
Student Name	Silva	Grace	**Unit**	Computer Science
			Session	1999/2000

Project Topic Computerized Library System

Abstract

To Design a library System

Area/Scope Database

Case Study A case study of Kaduna State Library

Supervisor Mrs. Usman

KADUNA POLYTECNHIC

Project Database Management System

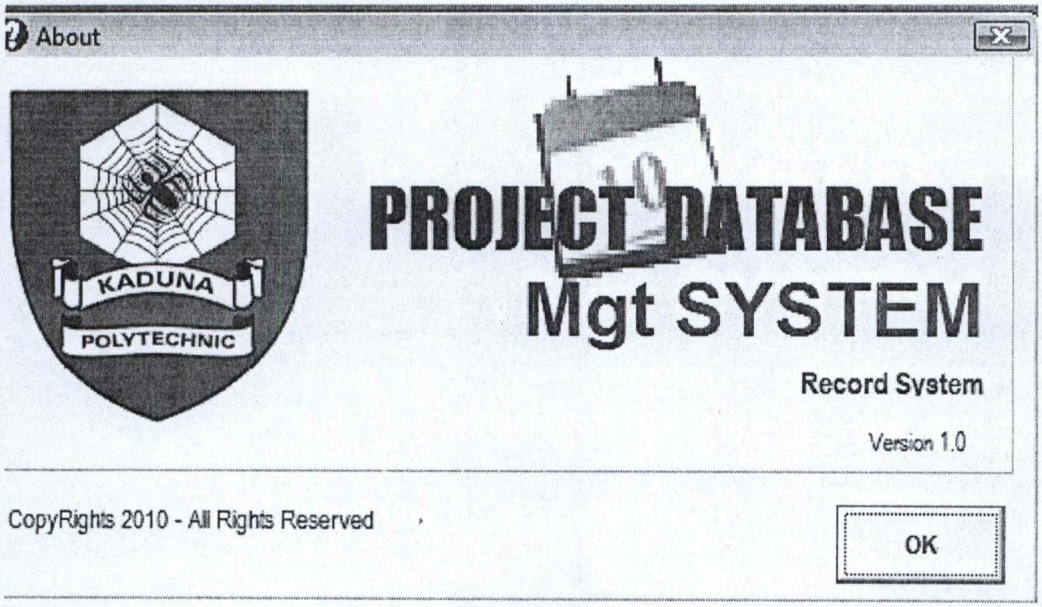
Thursday, September 02, 2010

ND PROJECT REPORT BY DEPARTMENT

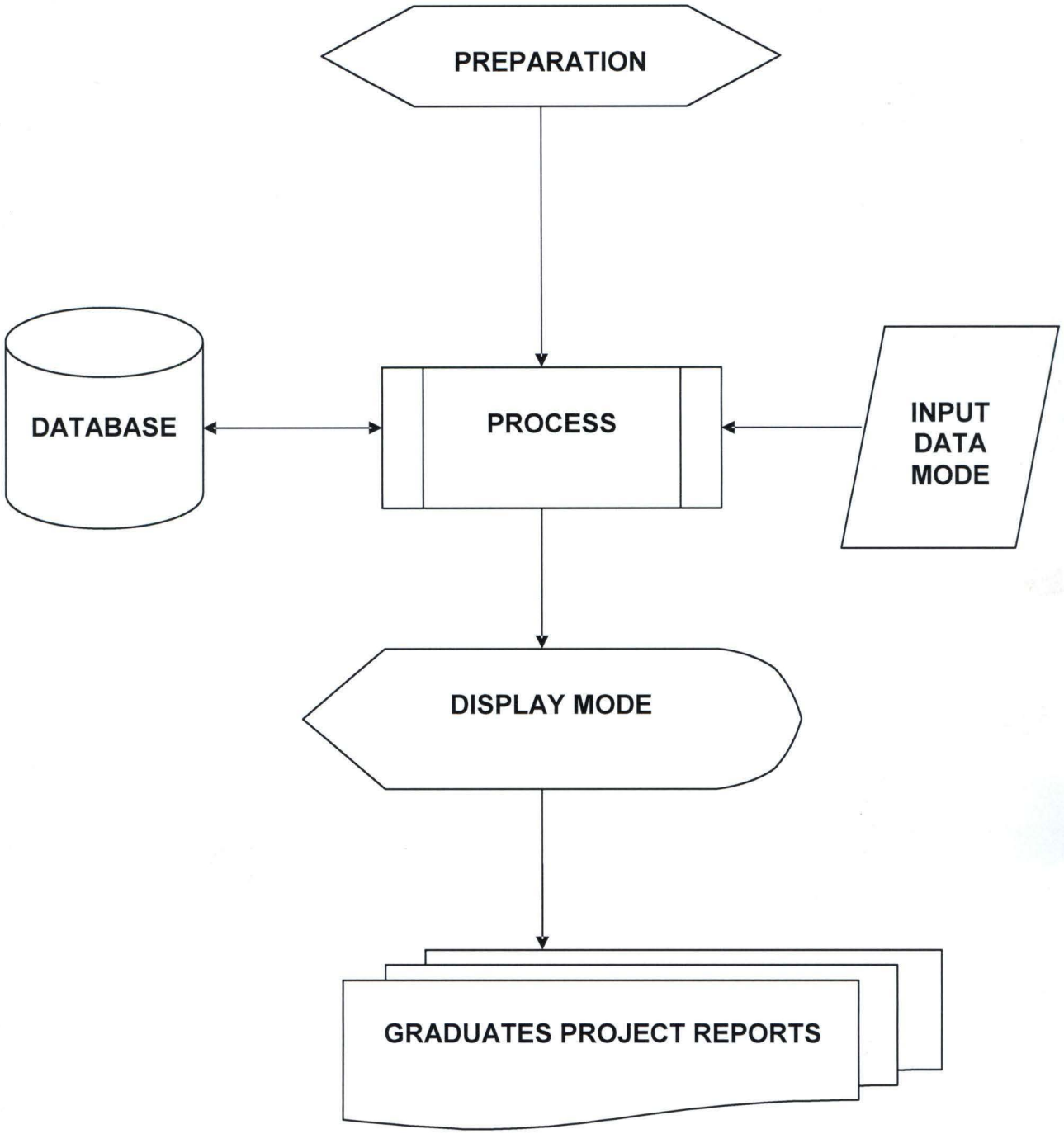
Reg. No	Surname	First Name	Department	Project Topic	Session
97/117	Silva	Grace	Maths, Stats and Computer Sc.	Computerized Library System	1999/2000
96/111	Idris	Mohammed	Maths, Stats and Computer Sc.	Computer Aided Learning	1999/2000

TOTAL NUMBER OF PROJECTS: 2

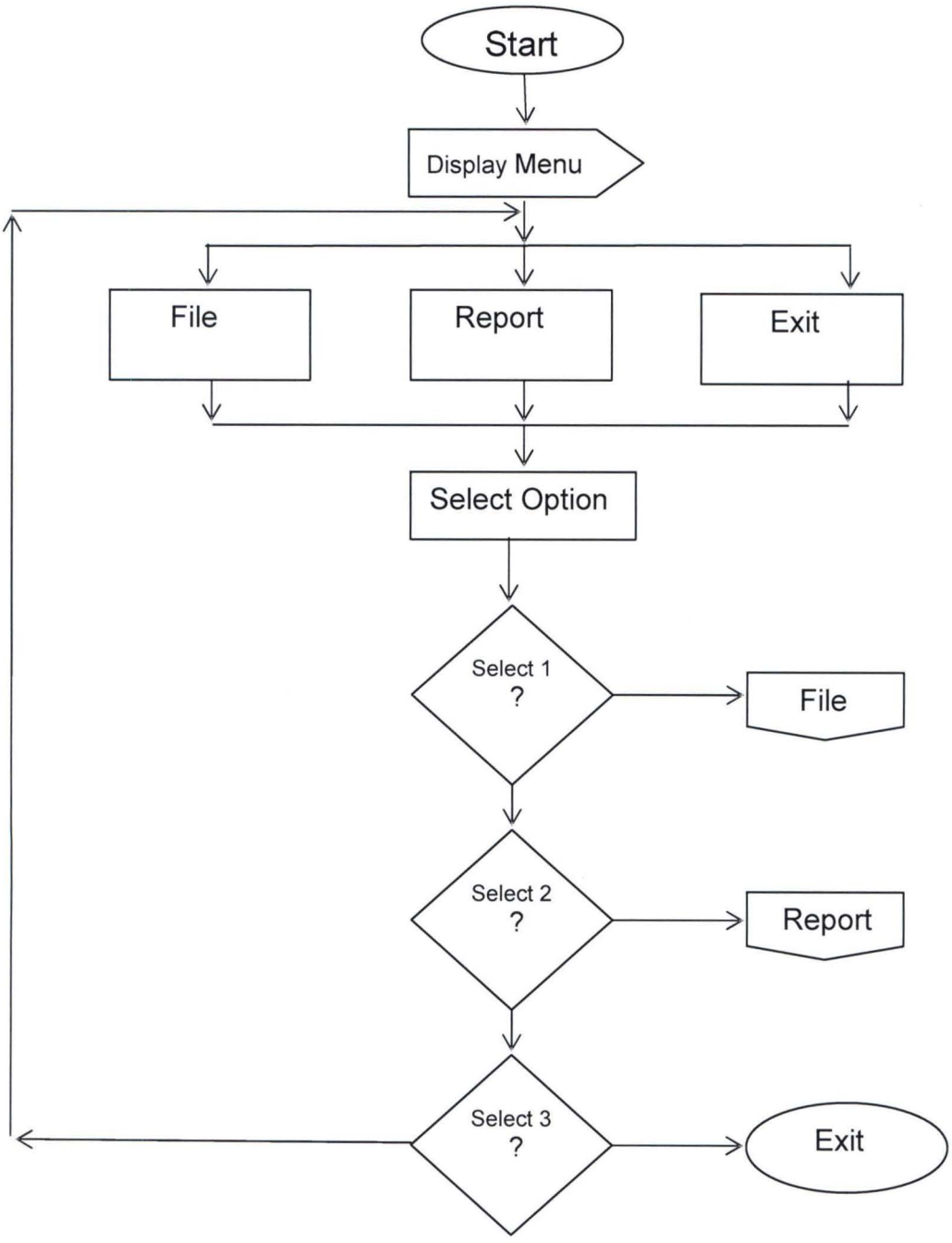
ABOUT PROGRAM VIEW

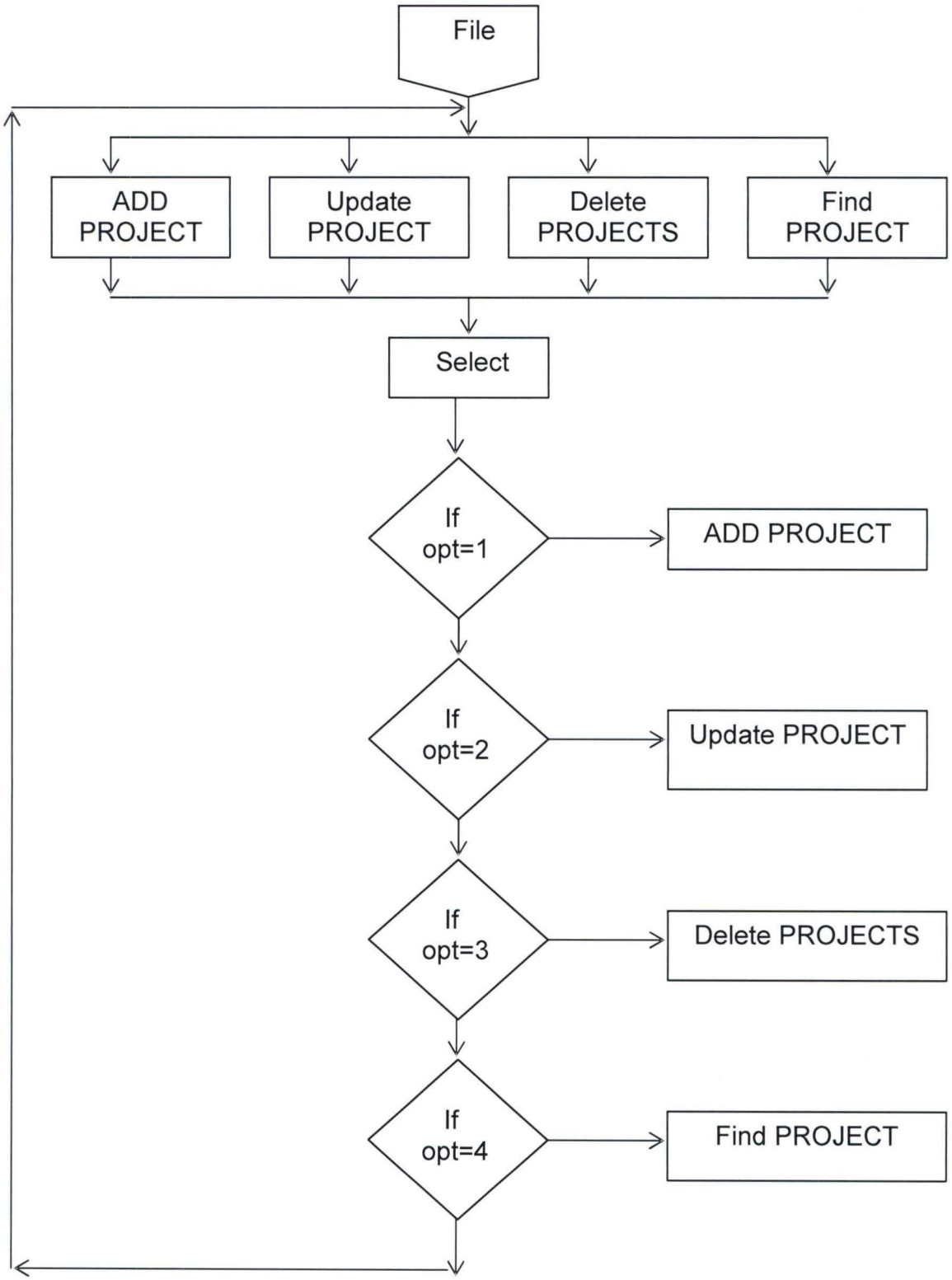


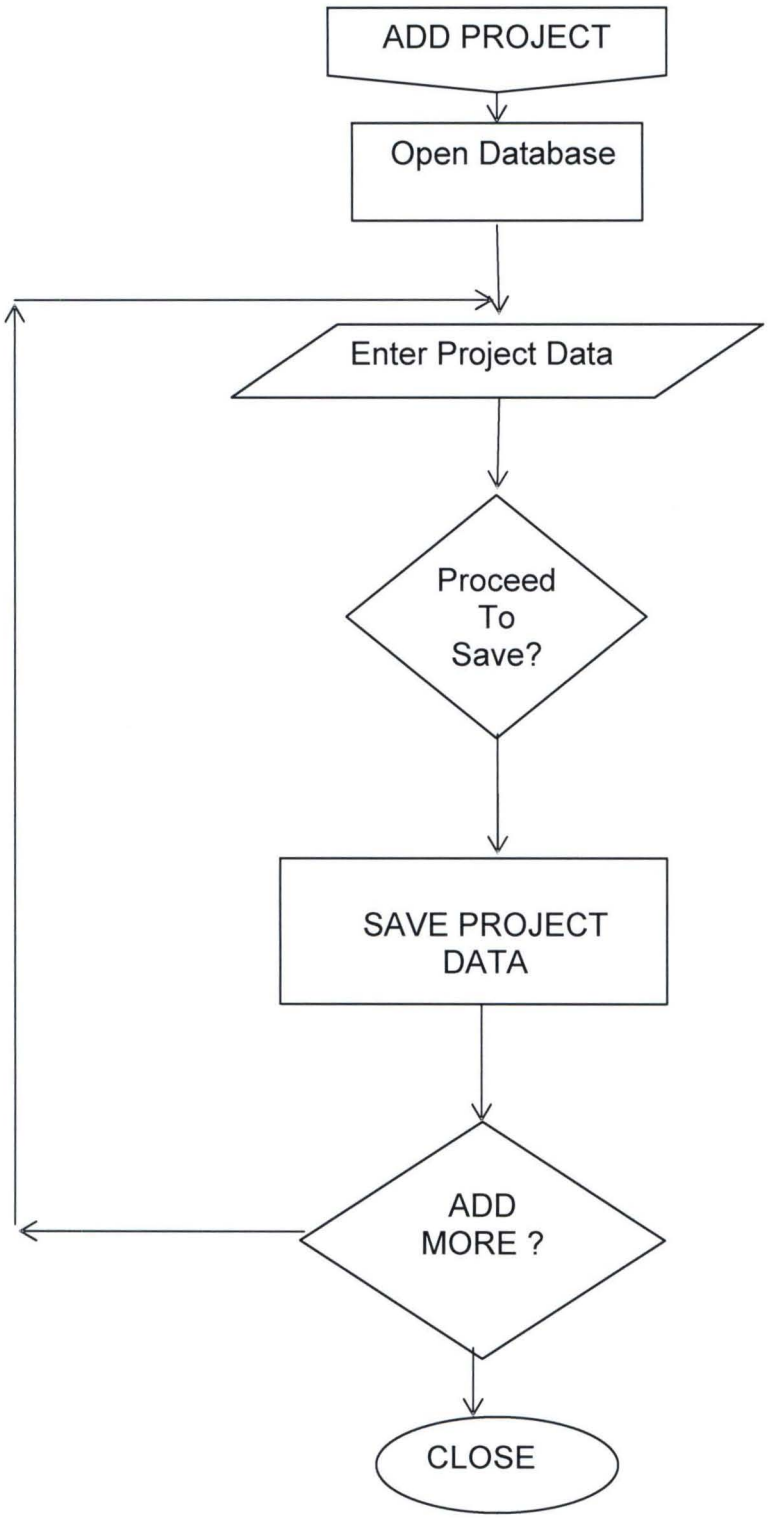
4.3 SYSTEM FLOWCHART

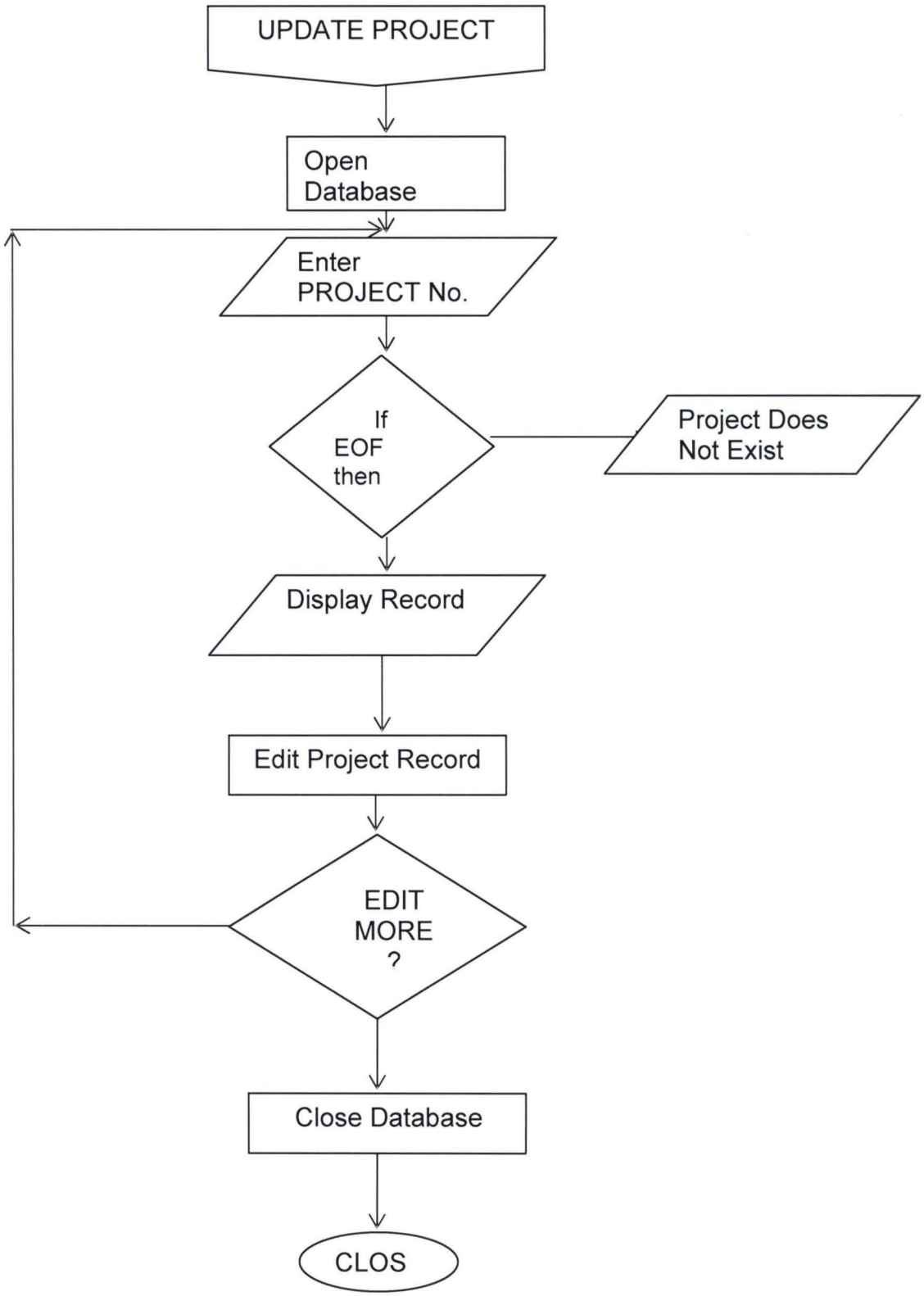


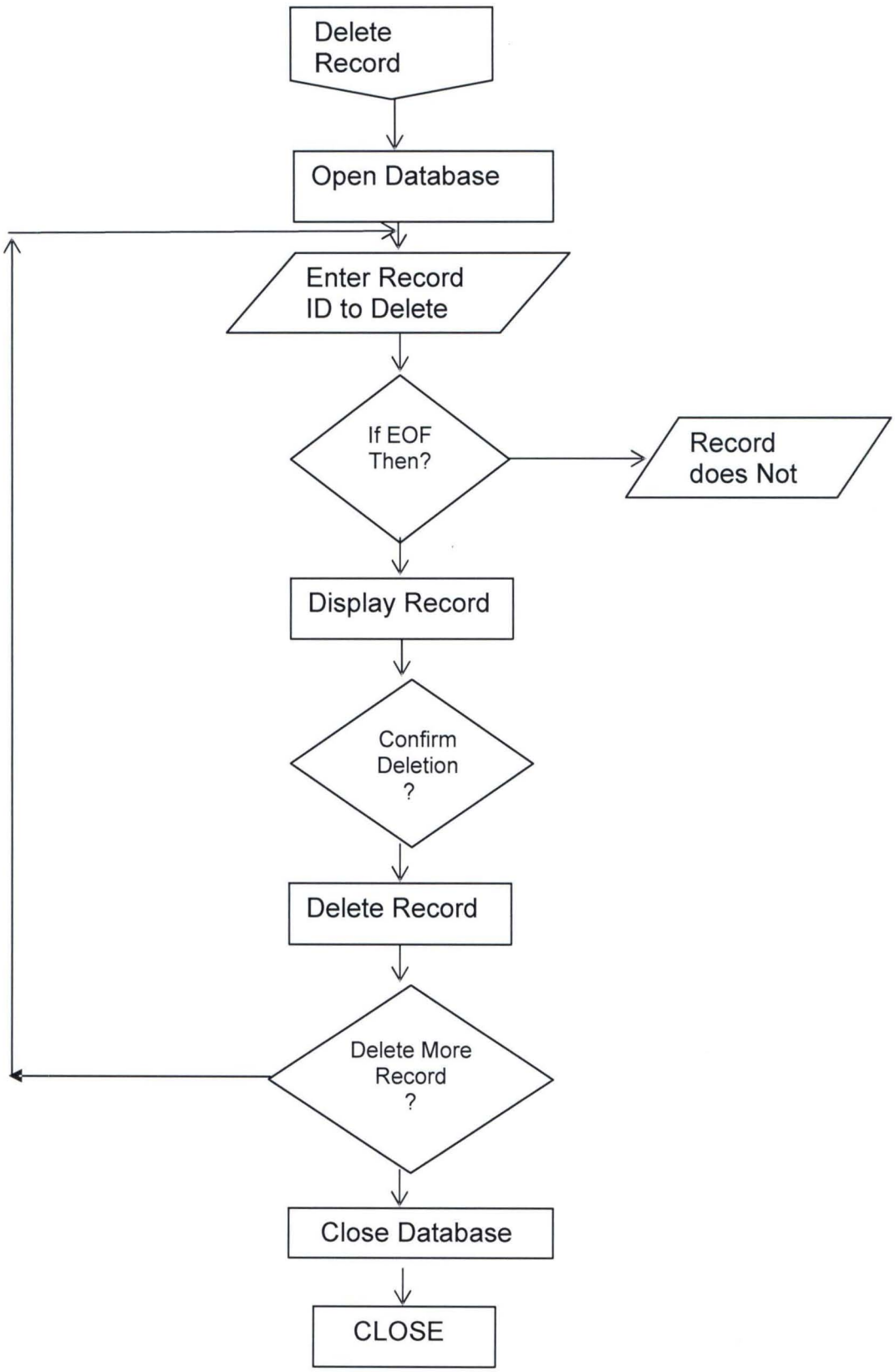
4.4 PROGRAM FLOWCHART

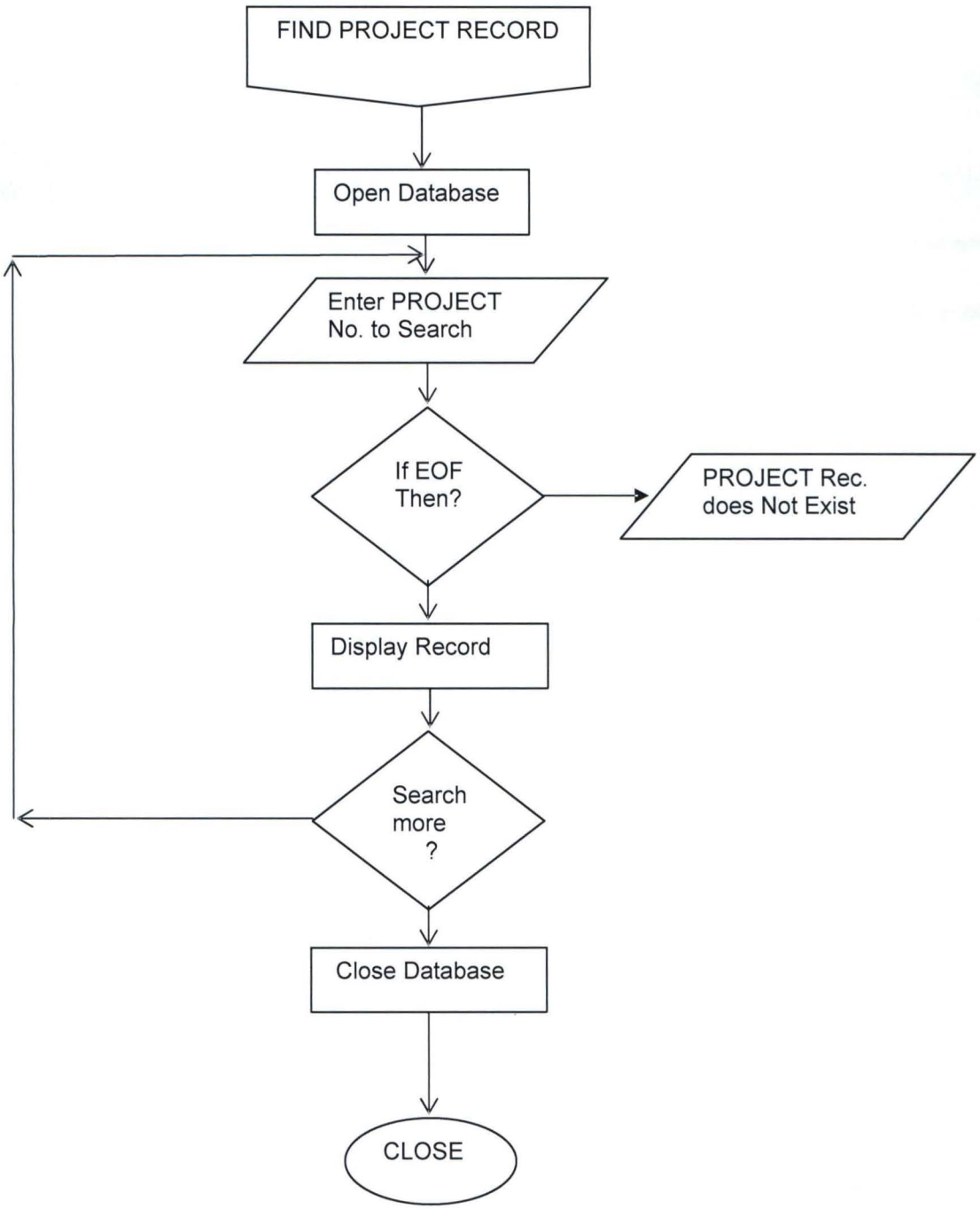


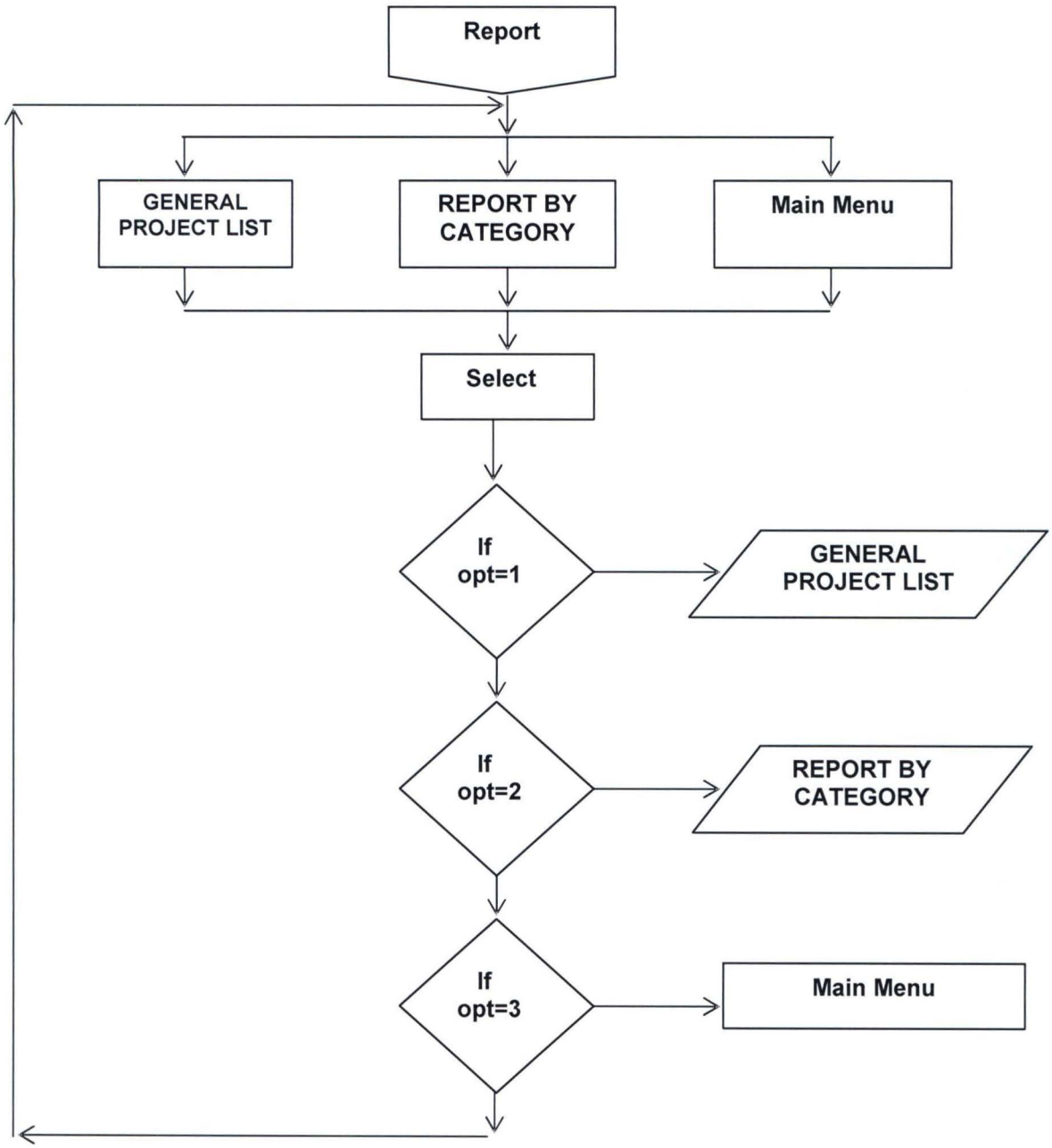

File
ADD PROJECT
Update PROJECT
Delete PROJECTS
Find PROJECT
Select
If opt=1
ADD PROJECT
If opt=2
Update PROJECT
If opt=3
Delete PROJECTS
If opt=4
Find PROJECT


ADD PROJECT
Open Database
Enter Project Data
Proceed To Save?
SAVE PROJECT DATA
ADD MORE ?
CLOSE


UPDATE PROJECT
Open Database
Enter PROJECT No.
If EOF then
Project Does Not Exist
Display Record
Edit Project Record
EDIT MORE ?
Close Database
CLOS


Delete Record
Open Database
Enter Record ID to Delete
If EOF Then?
Record does Not
Display Record
Confirm Deletion ?
Delete Record
Delete More Record ?
Close Database
CLOSE


FIND PROJECT RECORD
Open Database
Enter PROJECT No. to Search
If EOF Then?
PROJECT Rec. does Not Exist
Display Record
Search more ?
Close Database
CLOSE


Report
GENERAL PROJECT LIST
REPORT BY CATEGORY
Main Menu
Select
If opt=1
GENERAL PROJECT LIST
If opt=2
REPORT BY CATEGORY
If opt=3
Main Menu

4.5 OPERATIONAL MANUAL

The program was designed using Microsoft Visual Basic 6.0 and has been enabled to run on windows operating system ranging from Windows 98 edition all through to the most recent OS.

TO RUN THE PROGRAM

1. Copy the program folder "**STUDENT PROJECT DATABASE**" to any drive or location on your computer.
2. Open the Program folder.
3. Double click on the .exe file "**Projects Database Mgt System**"
4. Enter Username and Password (Open the README file to find login details)
5. Click Ok

CHAPTER FIVE

5.0 SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 INTRODUCTION

This chapter presents a summary and conclusion of the entire project carried out so far. Also the recommendations made by the researcher to enable the proposed system work efficiently.

5.2 SUMMARY

The research work was carried out for the purpose of developing a all-inclusive database for the management of the graduates' projects that are carried out in the Institution.

Therefore, the research work encompassed all activities involved in graduates' projects record management, which includes creating, updating, storing, filing and retrieving of projects/research records for the institution's library. The mode of operations was looked at and the problems associated with the manual system were stated.

Importance of the system were enumerated which would ease all file creation, modification and report generation of graduates record efficiently and effectively.

In carrying out the detailed investigation, both primary and secondary source of data were used.

The findings made after the analysis and interpretation of data are: -

(i.) The library does not have an up-to-date record of graduates, which is one of the basic problems of the current method of record keeping.

(ii.) The library does not have all relevant and comprehensive records of past and present graduates' projects information.

(iii.) There was a general opinion by students and staff of the institution that, there is need to change the method of operation into a computerized one.

According to the above findings, the developed database and application program will aid update, retrieval, querying and printing of reports for library's management consumption as regarding graduates projects information.

5.3 CONCLUSION

This research has been directed towards the design and development of application software to aid graduates record keeping in institution's library.

Conclusively, the newly developed system would speed up processes, enhance accuracy and integrity of data files, security would be re-enforced, there would be easy accessibility and retrieval of past and present graduates' records.

5.4 RECOMMENDATION

Based on the findings during the research work, the following recommendations have been made to the management of the institution:-

1. The current system of managing and keeping records of projects records needs to be changed in order to provide an effective way of maintaining such vital records.

2. In addition training programmes, seminars and workshops for staff in the library and institution at large should be introduced to make them knowledgeable and literate in computer in order to enable an easy implementation and application of the Graduates Projects.

3. Conclusively, it is highly recommended that all aspect of information processing and management operations in the institution should be computerized to meet up with the growing trends in modern technological applications.

REFERENCES

1. Beker M., Haberfellner R., and Liebetrau G. (1989), *Electronic Data Processing in Practice*, Ellis Horwood Limited, England

2. Cascio, W.F. (1989), *Managing Human Resources, Productivity, Quality of Work, Life, Profits*. McGraw Hill Inc. U.S.A., Second Edition

3. Dennis L., Michael S., and William C., (1995), *Information Security (Dictionary of Concepts Standards and Terms)*, McMillan Publishers Limited, U.S.A.

4. Evangelos Petroutsos (2000), Mastering Database Programming with Visual Basic. Sybex inc. Marina village parkway, CA. USA.

5. Gupta C.B., (1973), *An Introduction to Statistical Methods*, Vikas Publishing House PVT LTD, 9th Edition.

6. Michael S., and William C., (1995), "*Information Security*" (Dictionary of Concepts Standards and Terms), McMillan Publishers Limited, U.S.A.

7. Murdrick, R.G. and Ross J.B., "*Information System for Modern Management*". Prentice-Hall, Eagle Wood Cliff, New Jersey, U.S.A.

8. Murdrick, R.G. and Ross J.B., *Information System for Modern Management.* Prentice-Hall, Eagle Wood Cliff, New Jersey, .S.A.

9. Terry lucey, (1997), Management Information System 3rd Edition.

10. Wayne F. C. and Awad E. M. (1981), *Human Resources Information Systems*, Mc-Graw Hill Books Company, U.S.A.

APPENDIX A

PROGRAM CODE LISTING

PROGRAM MODULE

```
Attribute VB_Name = "Module1"
   Public userid As String

   Public adoCon As New ADODB.Connection
   Public adoRS As New ADODB.Recordset

   Public GL_REPORT As String

Public Sub openconn()

   On Error GoTo checkerr
   Dim conStr As String

   conStr = "Provider=Microsoft.Jet.OLEDB.4.0;"
   'conStr = conStr & "Password=bmsdbpass;"
   'conStr = conStr & "User ID=;"
   conStr = conStr & "Data source=" & App.Path & "/MasterFile.mdb"


   adoCon.ConnectionString = conStr

   adoCon.Open ', "moderator", "bmsdbpass"

   Exit Sub
checkerr:
   MsgBox "Error Occured: " & Err.Number & ": " & Err.Description & " in: " & Err.Source & vbCrLf& vbCrLf & "Please call me and
tell me about this error & how it came", vbCritical
   Call closeconn
   End

End Sub


Public Sub closeconn()

  On Error Resume Next
   adoRS.Close
   Set adoRS = Nothing
   adoCon.Close
   Set adoCon = Nothing


End Sub

Public Sub rs(ByVal sqlstr As String)
   On Error GoTo checkerr2

   adoRS.Open sqlstr, adoCon, adOpenDynamic, adLockOptimistic
   Exit Sub
checkerr2:
   MsgBox "Error Occured: " & Err.Number & ": " & Err.Description & " in: " & Err.Source & vbCrLf& vbCrLf & "Please Inform your
System Administrator about the problem", vbCritical
   Call closeconn
   End
End Sub


Public Function numdisp(ByVal num)
   numdisp = FormatNumber(num, , , vbFalse, vbTrue)
End Function

'*****************************************************
'***************** Encryption/Decryption ***************
'Works for general text encryption/decryption in order to encrypt usernames
'and passwords all text with character string first and numeric
'later, is encrypted properly
```

```
Public Function Crypt(ByVal s_text As String)
    s_text = Replace(s_text, "0", "%r", , , vbBinaryCompare)
    s_text = Replace(s_text, "1", "#g", , , vbBinaryCompare)
    s_text = Replace(s_text, "2", "1$", , , vbBinaryCompare)
    s_text = Replace(s_text, "3", "j~", , , vbBinaryCompare)
    s_text = Replace(s_text, "4", "j#", , , vbBinaryCompare)
    s_text = Replace(s_text, "5", "3?", , , vbBinaryCompare)
    s_text = Replace(s_text, "6", "*t", , , vbBinaryCompare)
    s_text = Replace(s_text, "7", "u@", , , vbBinaryCompare)
    s_text = Replace(s_text, "8", "n!", , , vbBinaryCompare)
    s_text = Replace(s_text, "9", "&x", , , vbBinaryCompare)

    s_text = Replace(s_text, "a", "1!", , , vbBinaryCompare)
    s_text = Replace(s_text, "b", "2@", , , vbBinaryCompare)
    s_text = Replace(s_text, "c", "3#", , , vbBinaryCompare)
    s_text = Replace(s_text, "d", "4$", , , vbBinaryCompare)
    s_text = Replace(s_text, "e", "5$", , , vbBinaryCompare)
    s_text = Replace(s_text, "f", "7*", , , vbBinaryCompare)
    s_text = Replace(s_text, "g", "9#", , , vbBinaryCompare)
    s_text = Replace(s_text, "h", "0#", , , vbBinaryCompare)
    s_text = Replace(s_text, "i", "4@", , , vbBinaryCompare)
    s_text = Replace(s_text, "j", "7#", , , vbBinaryCompare)
    s_text = Replace(s_text, "k", "8^", , , vbBinaryCompare)
    s_text = Replace(s_text, "l", "0^", , , vbBinaryCompare)
    s_text = Replace(s_text, "m", "5%", , , vbBinaryCompare)
    s_text = Replace(s_text, "n", "a%", , , vbBinaryCompare)
    s_text = Replace(s_text, "o", "e$", , , vbBinaryCompare)
    s_text = Replace(s_text, "p", "f5", , , vbBinaryCompare)
    s_text = Replace(s_text, "q", "6$", , , vbBinaryCompare)
    s_text = Replace(s_text, "r", "h&", , , vbBinaryCompare)
    s_text = Replace(s_text, "s", "0.", , , vbBinaryCompare)
    s_text = Replace(s_text, "t", "e`", , , vbBinaryCompare)
    s_text = Replace(s_text, "u", "4r", , , vbBinaryCompare)
    s_text = Replace(s_text, "v", "7@", , , vbBinaryCompare)
    s_text = Replace(s_text, "w", "f^", , , vbBinaryCompare)
    s_text = Replace(s_text, "x", "t%", , , vbBinaryCompare)
    s_text = Replace(s_text, "y", "g@", , , vbBinaryCompare)
    s_text = Replace(s_text, "z", "h0", , , vbBinaryCompare)

    s_text = Replace(s_text, "A", ".2", , , vbBinaryCompare)
    s_text = Replace(s_text, "B", ".3", , , vbBinaryCompare)
    s_text = Replace(s_text, "C", ".4", , , vbBinaryCompare)
    s_text = Replace(s_text, "D", ".5", , , vbBinaryCompare)
    s_text = Replace(s_text, "E", ".6", , , vbBinaryCompare)
    s_text = Replace(s_text, "F", ".7", , , vbBinaryCompare)
    s_text = Replace(s_text, "G", ".8", , , vbBinaryCompare)
    s_text = Replace(s_text, "H", ".9", , , vbBinaryCompare)
    s_text = Replace(s_text, "I", ".0", , , vbBinaryCompare)
    s_text = Replace(s_text, "J", ".1", , , vbBinaryCompare)
    s_text = Replace(s_text, "K", "/3", , , vbBinaryCompare)
    s_text = Replace(s_text, "L", "/5", , , vbBinaryCompare)
    s_text = Replace(s_text, "M", "/7", , , vbBinaryCompare)
    s_text = Replace(s_text, "N", "/9", , , vbBinaryCompare)
    s_text = Replace(s_text, "O", "/1", , , vbBinaryCompare)
    s_text = Replace(s_text, "P", "/0", , , vbBinaryCompare)
    s_text = Replace(s_text, "Q", "/8", , , vbBinaryCompare)
    s_text = Replace(s_text, "R", "/6", , , vbBinaryCompare)
    s_text = Replace(s_text, "S", "/4", , , vbBinaryCompare)
    s_text = Replace(s_text, "T", "/2", , , vbBinaryCompare)
    s_text = Replace(s_text, "U", ";0", , , vbBinaryCompare)
    s_text = Replace(s_text, "V", ";2", , , vbBinaryCompare)
    s_text = Replace(s_text, "W", ";3", , , vbBinaryCompare)
    s_text = Replace(s_text, "X", ";4", , , vbBinaryCompare)
    s_text = Replace(s_text, "Y", ";6", , , vbBinaryCompare)
    s_text = Replace(s_text, "Z", ";7", , , vbBinaryCompare)

    Crypt = s_text
End Function

Public Function Decrypt(ByVal s_text As String)
    s_text = Replace(s_text, "1!", "a", , , vbBinaryCompare)
    s_text = Replace(s_text, "2@", "b", , , vbBinaryCompare)
    s_text = Replace(s_text, "3#", "c", , , vbBinaryCompare)
```

```
    s_text = Replace(s_text, "4$", "d", , , vbBinaryCompare)
    s_text = Replace(s_text, "5$", "e", , , vbBinaryCompare)
    s_text = Replace(s_text, "6#", "d", , , vbBinaryCompare)
    s_text = Replace(s_text, "7*", "f", , , vbBinaryCompare)
    s_text = Replace(s_text, "9#", "g", , , vbBinaryCompare)
    s_text = Replace(s_text, "0#", "h", , , vbBinaryCompare)
    s_text = Replace(s_text, "4@", "i", , , vbBinaryCompare)
    s_text = Replace(s_text, "7#", "j", , , vbBinaryCompare)
    s_text = Replace(s_text, "8^", "k", , , vbBinaryCompare)
    s_text = Replace(s_text, "0^", "l", , , vbBinaryCompare)
    s_text = Replace(s_text, "5%", "m", , , vbBinaryCompare)
    s_text = Replace(s_text, "a%", "n", , , vbBinaryCompare)
    s_text = Replace(s_text, "e$", "o", , , vbBinaryCompare)
    s_text = Replace(s_text, "f5", "p", , , vbBinaryCompare)
    s_text = Replace(s_text, "6$", "q", , , vbBinaryCompare)
    s_text = Replace(s_text, "h&", "r", , , vbBinaryCompare)
    s_text = Replace(s_text, "0.", "s", , , vbBinaryCompare)
    s_text = Replace(s_text, "e`", "t", , , vbBinaryCompare)
    s_text = Replace(s_text, "4r", "u", , , vbBinaryCompare)
    s_text = Replace(s_text, "7@", "v", , , vbBinaryCompare)
    s_text = Replace(s_text, "f^", "w", , , vbBinaryCompare)
    s_text = Replace(s_text, "t%", "x", , , vbBinaryCompare)
    s_text = Replace(s_text, "g@", "y", , , vbBinaryCompare)
    s_text = Replace(s_text, "h0", "z", , , vbBinaryCompare)

    s_text = Replace(s_text, ".2", "A", , , vbBinaryCompare)
    s_text = Replace(s_text, ".3", "B", , , vbBinaryCompare)
    s_text = Replace(s_text, ".4", "C", , , vbBinaryCompare)
    s_text = Replace(s_text, ".5", "D", , , vbBinaryCompare)
    s_text = Replace(s_text, ".6", "E", , , vbBinaryCompare)
    s_text = Replace(s_text, ".7", "F", , , vbBinaryCompare)
    s_text = Replace(s_text, ".8", "G", , , vbBinaryCompare)
    s_text = Replace(s_text, ".9", "H", , , vbBinaryCompare)
    s_text = Replace(s_text, ".0", "I", , , vbBinaryCompare)
    s_text = Replace(s_text, ".1", "J", , , vbBinaryCompare)
    s_text = Replace(s_text, "/3", "K", , , vbBinaryCompare)
    s_text = Replace(s_text, "/5", "L", , , vbBinaryCompare)
    s_text = Replace(s_text, "/7", "M", , , vbBinaryCompare)
    s_text = Replace(s_text, "/9", "N", , , vbBinaryCompare)
    s_text = Replace(s_text, "/1", "O", , , vbBinaryCompare)
    s_text = Replace(s_text, "/0", "P", , , vbBinaryCompare)
    s_text = Replace(s_text, "/8", "Q", , , vbBinaryCompare)
    s_text = Replace(s_text, "/6", "R", , , vbBinaryCompare)
    s_text = Replace(s_text, "/4", "S", , , vbBinaryCompare)
    s_text = Replace(s_text, "/2", "T", , , vbBinaryCompare)
    s_text = Replace(s_text, ";0", "U", , , vbBinaryCompare)
    s_text = Replace(s_text, ";2", "V", , , vbBinaryCompare)
    s_text = Replace(s_text, ";3", "W", , , vbBinaryCompare)
    s_text = Replace(s_text, ";4", "X", , , vbBinaryCompare)
    s_text = Replace(s_text, ";6", "Y", , , vbBinaryCompare)
    s_text = Replace(s_text, ";7", "Z", , , vbBinaryCompare)


    s_text = Replace(s_text, "%r", "0", , , vbBinaryCompare)
    s_text = Replace(s_text, "#g", "1", , , vbBinaryCompare)
    s_text = Replace(s_text, "1$", "2", , , vbBinaryCompare)
    s_text = Replace(s_text, "j~", "3", , , vbBinaryCompare)
    s_text = Replace(s_text, "j#", "4", , , vbBinaryCompare)
    s_text = Replace(s_text, "3?", "5", , , vbBinaryCompare)
    s_text = Replace(s_text, "*t", "6", , , vbBinaryCompare)
    s_text = Replace(s_text, "u@", "7", , , vbBinaryCompare)
    s_text = Replace(s_text, "n!", "8", , , vbBinaryCompare)
    s_text = Replace(s_text, "&x", "9", , , vbBinaryCompare)
    Decrypt = s_text
End Function


' ================ check for valid chracters only .--==============
Public Function validity(ByVal str As TextBox, ByVal field As String)
    If Len(str.Text) > 0 Then
        For i = 1 To Len(str.Text)
        xyz = InStr(1, "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890", Mid(str.Text, i, 1),
vbBinaryCompare)
            If xyz = 0 Then
```

```
            MsgBox "Invalid Character (" & Mid(str.Text, i, 1) & ") Found in " & field & " Field", vbCritical, "Cannot Proceed!"
            invalid = True
          Exit For
        End If
      Next i

      If invalid = True Then
        validity = False
      Else
        validity = True
      End If
    Else
      validity = True
    End If
End Function

' ================ check for valid of ' & " only .--==========
Public Function validity2(ByVal str As TextBox, ByVal field As String)
    If Len(str.Text) > 0 Then
      For i = 1 To Len(str.Text)
      xyz = InStr(1, """", Mid(str.Text, i, 1), vbBinaryCompare)
        If xyz > 0 Then
            MsgBox "Invalid Character (" & Mid(str.Text, i, 1) & ") Found in " & field & " Field", vbCritical, "Cannot Proceed!"
            invalid = True
          Exit For
        End If
      Next i

      If invalid = True Then
        validity2 = False
      Else
        validity2 = True
      End If
    Else
      validity2 = True
    End If
End Function

VERSION 5.00
Object = "{831FDD16-0C5C-11D2-A9FC-0000F8754DA1}#2.0#0"; "mscomctl.ocx"
Begin VB.MDIForm MDIForm1
  BackColor      =  &H00808080&
  Caption        =  "Project Database Management System..."
  ClientHeight   =  7065
  ClientLeft     =  225
  ClientTop      =  825
  ClientWidth    =  7995
  Icon           =  "MDIForm1.frx":0000
  LinkTopic      =  "MDIForm1"
  StartUpPosition =  3  'Windows Default
  WindowState    =  2  'Maximized
  Begin VB.PictureBox PictureFrame
    Align        =  1  'Align Top
    AutoRedraw   =  -1  'True
    Height       =  735
    Left         =  0
    ScaleHeight  =  675
    ScaleWidth   =  7935
    TabIndex     =  5
    Top          =  0
    Width        =  7995
    Begin VB.CommandButton Command6
      Caption      =  "About"
      BeginProperty Font
        Name         =  "Arial Narrow"
        Size         =  9
        Charset      =  0
        Weight       =  700
        Underline    =  0  'False
        Italic       =  0  'False
        Strikethrough  =  0  'False
      EndProperty
      Height       =  700
      Left         =  2760
```

```
      Picture         = "MDIForm1.frx":6852
      Style           = 1 'Graphical
      TabIndex        = 10
      Top             = 0
      Width           = 855
    End
    Begin VB.CommandButton Command4
      Caption         = "Calendar"
      BeginProperty Font
        Name          = "Arial Narrow"
        Size          = 8.25
        Charset       = 0
        Weight        = 700
        Underline     = 0 'False
        Italic        = 0 'False
        Strikethrough = 0 'False
      EndProperty
      Height          = 700
      Left            = 1800
      Picture         = "MDIForm1.frx":6B5C
      Style           = 1 'Graphical
      TabIndex        = 9
      Top             = 0
      Width           = 975
    End
    Begin VB.CommandButton Command3
      Caption         = "ND PROJECT"
      BeginProperty Font
        Name          = "Arial Narrow"
        Size          = 8.25
        Charset       = 0
        Weight        = 700
        Underline     = 0 'False
        Italic        = 0 'False
        Strikethrough = 0 'False
      EndProperty
      Height          = 700
      Left            = 840
      Picture         = "MDIForm1.frx":6F9E
      Style           = 1 'Graphical
      TabIndex        = 8
      Top             = 0
      Width           = 975
    End
    Begin VB.CommandButton Command1
      Caption         = "HND PROJECT"
      BeginProperty Font
        Name          = "Arial Narrow"
        Size          = 8.25
        Charset       = 0
        Weight        = 700
        Underline     = 0 'False
        Italic        = 0 'False
        Strikethrough = 0 'False
      EndProperty
      Height          = 700
      Left            = 0
      Picture         = "MDIForm1.frx":7DE0
      Style           = 1 'Graphical
      TabIndex        = 6
      Top             = 0
      Width           = 855
    End
  End
  Begin VB.PictureBox Picture4
    Align           = 2 'Align Bottom
    BackColor       = &H80000010&
    BorderStyle     = 0 'None
    Height          = 10
    Left            = 0
    ScaleHeight     = 15
    ScaleWidth      = 7995
    TabIndex        = 3
    Top             = 7035
```

```
  Width          =  7995
End
Begin VB.PictureBox Picture3
  Align          =  1  'Align Top
  BackColor      =  &H80000005&
  BorderStyle    =  0  'None
  Height         =  10
  Left           =  0
  ScaleHeight    =  15
  ScaleWidth     =  7995
  TabIndex       =  2
  Top            =  750
  Width          =  7995
End
Begin VB.PictureBox Picture2
  Align          =  2  'Align Bottom
  BackColor      =  &H80000005&
  BorderStyle    =  0  'None
  Height         =  10
  Left           =  0
  ScaleHeight    =  15
  ScaleWidth     =  7995
  TabIndex       =  1
  Top            =  7050
  Width          =  7995
End
Begin VB.PictureBox Picture1
  Align          =  1  'Align Top
  BackColor      =  &H80000010&
  BorderStyle    =  0  'None
  Height         =  10
  Left           =  0
  ScaleHeight    =  15
  ScaleWidth     =  7995
  TabIndex       =  0
  Top            =  735
  Width          =  7995
  Begin VB.CommandButton Command2
    Caption        =  "Command2"
    Height         =  195
    Left           =  1200
    TabIndex       =  4
    Top            =  0
    Width          =  1335
  End
End
Begin MSComctlLib.StatusBar StatusBar2
  Align          =  2  'Align Bottom
  Height         =  375
  Left           =  0
  TabIndex       =  7
  Top            =  6660
  Width          =  7995
  _ExtentX       =  14102
  _ExtentY       =  661
  _Version       =  393216
  BeginProperty Panels {8E3867A5-8586-11D1-B16A-00C0F0283628}
    NumPanels      =  7
    BeginProperty Panel1 {8E3867AB-8586-11D1-B16A-00C0F0283628}
      Object.Width         =  5292
      MinWidth       =  5292
      Text           =  "KADUNA POLYTECHNIC."
      TextSave       =  "KADUNA POLYTECHNIC."
    EndProperty
    BeginProperty Panel2 {8E3867AB-8586-11D1-B16A-00C0F0283628}
      Style          =  6
      Alignment      =  1
      Object.Width         =  2293
      MinWidth       =  2293
      TextSave       =  "9/1/2010"
    EndProperty
    BeginProperty Panel3 {8E3867AB-8586-11D1-B16A-00C0F0283628}
      Alignment      =  1
      Object.Width         =  2293
```

```
        MinWidth      = 2293
      EndProperty
      BeginProperty Panel4 {8E3867AB-8586-11D1-B16A-00C0F0283628}
      Caption       = "Mgt SYSTEM"
      BeginProperty Font
        Name          = "Arial"
        Size          = 27.75
        Charset       = 0
        Weight        = 700
        Underline     = 0 'False
        Italic        = 0 'False
        Strikethrough = 0 'False
      EndProperty
      Height        = 735
      Left          = 3360
      TabIndex      = 3
      Top           = 1800
      Width         = 4095
    End
  End
  Begin VB.CommandButton cmdOK
    Cancel        = -1 'True
    Caption       = "OK"
    Default       = -1 'True
    BeginProperty Font
      Name          = "Arial Narrow"
      Size          = 9.75
      Charset       = 0
      Weight        = 700
      Underline     = 0 'False
      Italic        = 0 'False
      Strikethrough = 0 'False
    EndProperty
    Height        = 585
    Left          = 6360
    TabIndex      = 0
    Top           = 3360
    Width         = 1260
  End
  Begin VB.Label Label2
    BackStyle     = 0 'Transparent
    Caption       = "CopyRights 2010 - All Rights Reserved"
    BeginProperty Font
      Name          = "Arial Narrow"
      Size          = 9.75
      Charset       = 0
      Weight        = 400
      Underline     = 0 'False
      Italic        = 0 'False
      Strikethrough = 0 'False
    EndProperty
    ForeColor     = &H00000000&
    Height        = 315
    Left          = 120
    TabIndex      = 1
    Top           = 3360
    Width         = 3135
  End
  Begin VB.Line Line1
    BorderColor   = &H00808080&
    BorderStyle   = 6 'Inside Solid
    Index         = 1
    X1            = 112.686
    X2            = 5337.571
    Y1            = 1905.001
    Y2            = 1905.001
  End
  Begin VB.Line Line1
    BorderColor   = &H00FFFFFF&
    BorderWidth   = 2
    Index         = 0
    X1            = 112.686
    X2            = 5323.485
    Y1            = 1905.001
```

```
    Y2          =  1905.001
  End
End
Attribute VB_Name = "frmAbout"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Private Sub cmdOK_Click()
  Unload Me
End Sub



VERSION 5.00
Object = "{67397AA1-7FB1-11D0-B148-00A0C922E820}#6.0#0"; "MSADODC.OCX"
Object = "{CDE57A40-8B86-11D0-B3C6-00A0C90AEA82}#1.0#0"; "MSDATGRD.OCX"
Object = "{BDC217C8-ED16-11CD-956C-0000C04E4C0A}#1.1#0"; "TABCTL32.OCX"
Begin VB.Form frmHND
   BackColor      =  &H80000010&
   BorderStyle    =  1  'Fixed Single
   Caption        =  "HND PROJECT RECORDS DATABASE FORM"
   ClientHeight   =  8670
   ClientLeft     =  45
   ClientTop      =  330
   ClientWidth    =  15045
   Icon           =  "frmHND.frx":0000
   LinkTopic      =  "Form1"
   MDIChild       =  -1  'True
   ScaleHeight    =  8670
   ScaleWidth     =  15045
   WindowState    =  2  'Maximized
   Begin TabDlg.SSTab SSTabl
      Height         =  6615
      Left           =  120
      TabIndex       =  24
      Top            =  240
      Width          =  14775
      _ExtentX       =  26061
      _ExtentY       =  11668
      _Version       =  393216
      TabOrientation =  1
      TabHeight      =  520
      BeginProperty Font {0BE35203-8F91-11CE-9DE3-00AA004BB851}
         Name           =  "MS Sans Serif"
         Size           =  8.25
         Charset        =  0
         Weight         =  700
         Underline      =  0  'False
         Italic         =  0  'False
         Strikethrough  =  0  'False
      EndProperty
      TabCaption(0)  =  "Proejcts Database Form View"
      TabPicture(0)  =  "frmHND.frx":6852
      Tab(0).ControlEnabled=  -1  'True
      Tab(0).Control(0)=  "Frame1"
      Tab(0).Control(0).Enabled=  0  'False
      Tab(0).Control(1)=  "Frame2"
      Tab(0).Control(1).Enabled=  0  'False
      Tab(0).Control(2)=  "Frame3"
      Tab(0).Control(2).Enabled=  0  'False
      Tab(0).ControlCount=  3
      TabCaption(1)  =  "Proejcts Database Grid View"
      TabPicture(1)  =  "frmHND.frx":D0B4
      Tab(1).ControlEnabled=  0  'False
      Tab(1).Control(0)=  "DataGrid1"
      Tab(1).ControlCount=  1
      TabCaption(2)  =  "Report Options"
      TabPicture(2)  =  "frmHND.frx":13916
      Tab(2).ControlEnabled=  0  'False
      Tab(2).Control(0)=  "Frame7"
      Tab(2).Control(0).Enabled=  0  'False
      Tab(2).Control(1)=  "Command1"
      Tab(2).Control(1).Enabled=  0  'False
```

```
    Tab(2).Control(2)= "Frame8"
    Tab(2).Control(2).Enabled= 0 'False
    Tab(2).Control(3)= "Frame12"
  Begin VB.Label Label6
    BackStyle      = 0 'Transparent
    Caption        = "Create New User"
    BeginProperty Font
      Name         = "Arial Narrow"
      Size         = 9.75
      Charset      = 0
      Weight       = 700
      Underline    = 0 'False
      Italic       = 0 'False
      Strikethrough = 0 'False
    EndProperty
    ForeColor      = &H00000000&
    Height         = 255
    Left           = 3840
    MouseIcon      = "Login.frx":4B0E
    MousePointer   = 99 'Custom
    TabIndex       = 6
    Top            = 4560
    Width          = 1425
  End
  Begin VB.Label Label2
    BackStyle      = 0 'Transparent
    Caption        = "CopyRights 2010 - All Rights Reserved."
    BeginProperty Font
      Name         = "Arial Narrow"
      Size         = 9.75
      Charset      = 0
      Weight       = 400
      Underline    = 0 'False
      Italic       = 0 'False
      Strikethrough = 0 'False
    EndProperty
    ForeColor      = &H00000000&
    Height         = 315
    Left           = 1440
    TabIndex       = 4
    Top            = 3720
    Width          = 2895
  End
End
Attribute VB_Name = "login"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Private Sub cmdcancel_Click()
  Unload Me
End Sub

Private Sub cmdlogin_Click()
  If txtlogin.Text = "" Then
    MsgBox "Enter A Login ID:", vbCritical
    txtlogin.SetFocus
    Exit Sub
  End If

  If Not validity(txtlogin, "Login ID") Then
    txtlogin.SetFocus
    Exit Sub
  End If

  If txtpass.Text = "" Then
    MsgBox "Enter A Password:", vbCritical
    txtpass.SetFocus
    Exit Sub
  End If

  If Not validity(txtpass, "Password") Then
    txtpass.SetFocus
    Exit Sub
```

```
    End If

    Call openconn
    sqlstr = "select * from users where loginid = '" & Crypt(txtlogin.Text) & "' and pass = '" & Crypt(txtpass.Text) & "'"
    Call rs(sqlstr)

    If (adoRS.EOF) Then
        MsgBox "Wrong Login ID or Password! Try Again", vbCritical
        txtlogin.Text = ""
        txtpass.Text = ""
        txtlogin.SetFocus
        Call closeconn
        Exit Sub
    End If

    Call closeconn

    userid = txtlogin.Text

    Unload Me
    SplashScreen.Show

End Sub

Private Sub Label6_Click()
    newuser.Show vbModal
End Sub

Private Sub Label7_Click()
    changepass.Show 'vbModal
End Sub

Private Sub Label9_Click()
    deluser.Show vbModal
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Unload Me
End Sub
```