DESIGN AND IMPLEMENTATION OF AN AUTOMATED REGISTER MANAGEMENT SYSTEM

A CASE STUDY: NABILATUK CATHOLIC PARISH, NAKAPIRIPIRIT DISTRICT

BY

LOKOL PAUL

BIT/10192/81/DU

A PROJECT REPORT SUBMITTED TO KAMPALA INTERNATIONAL UNIVERSITY COLLEGE OF APPLIED SCIENCE AND TECHNOLOGY AS A PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF INFORMATION TECHNOLOGY

MAY, 2013.

DECLARATION

I **LOKOL PAUL** do hereby declare to the best of my knowledge that this project report is my original work and that it has never been submitted to any University or any other Institution for any award.

Signed: .... Date: 16th / 08 / 2013

LOKOL PAUL

BIT/10192/81/DU

APPROVAL

This Graduation project report titled "DESIGN AND IMPLEMENTATION OF AN AUTOMATED REGISTER MANAGEMENT SYSTEM, a case of Nabilatuk Catholic Parish, Nakapiripirit District" has been submitted with the approval of the following supervisor.

Signature.. Date: 17TH / AUG - / 2013

Mr. OKELLO ALFRED

Supervisor

DEDICATION

To my beloved children; Apurio, Angella and Locham; with whose patience and warm encouragement I was able to complete this coursesuccessfully. To my brothers and sisters whose hope has been pegged to the success and completion of this course. And to my greatest friend, ally and caring benefactor, Rev. Fr. Michael Apurio, who has supported me unconditionally to the completion of my studies.

ACKNOWLEDGEMENTS

I acknowledge the support of my friends whose inspiration and great contribution has rendered this project complete. In a special way, I want to appreciate the emotional and academic support of Mr. Opoka, Raymond and Mohamed Gees, whose contributions, insight and advise I won't forget.

I thank my lecturers for the efforts made to shape me in the three years of my course: I also appreciate the good cooperation, and a favorable atmosphere offered by my supervisor, classmates, friends, relatives and the entire administration of Kampala International University. In as much as the finishing has been tougher to me, I have drawn for myself better strategies from the hard experience.

Fr. Michael Apurio takes a big half of my gratitude, for he has seen me through almost my entire education. A good friend is worth a father; and this is what Fr. Michael has been to me. From the charity and care of Fr. Michael, I have learnt to help others and to consider the 'hopeless'

God has been good to me and Holy is His name; what He has done, I cannot tell it all. Thanks be to God.

TABLE OF CONTENTS

LIST OF ABREVIATIONS

MIS	Management information systems
RAD	Rapid applications development
JAD	Joint applications development
I/O	Input/output
DFD	Data flow diagrams
ERD	Entity relationship diagrams
VB	Visual Basic
VBA	Visual Basic Application

ABSTRACT

Catholic Parishes manage a range of data and information books. Most important, are registers where records about the sacramental information are kept as a demand by Mother Church. Much of the registration and register management duties are meant to be carried by the Parish Priest, but due to the work overload and the nature of the traditional book-keeping system, the Parish Priest employs, and is most often assisted by, the seminarians and catechists. The inconsistencies of the current system have rendered the registration process unproductive, thus raising the need for the designing and implementation of the computerized system. The purpose of this project was to design and implement an automated register management system for Nabilatuk Catholic Parish. It involved looking at the concepts and information already made available on: Database Management Systems, Information Systems; information management system, system development approaches, unified modeling language, and the recent developments relating to the proposed system. The importance of this was to relate the proposed system to what other researchers have written about similar systems so as to inform the research and system development process. The system developed has made a starting point where other researchers can obtain insight necessary to make a more robust system. This report presents the architectural, logical and physical designs of the system, implementation and recommendations for further improvement of the new system

CHAPTER ONE

INTRODUCTION

1.0 Introduction

This chapter discusses background information to the project, problem statement, justification, main objective and specific Objectives, and the benefits of the project.

1.1 Background information to the project problem

Nabilatuk Catholic Parish is a mission station of the Roman Catholic Church. It was the third station to be established in Karamoja by the Comboni Missionaries in 1959. In 1989, the parish was handed over to the Apostles of Jesus missionaries (a congregation of the African religious priests and brothers); and in 1997, Nabilatuk was handed to the local diocesan priests of Moroto diocese who have manned it up to date. Nabilatuk was, from its inception, created to serve the spiritual and social needs of the Pokot and the Pian people of South Karamoja but gradually, as the need for services increased, the parish gave rise to the other three parishes of Namalu, Nakapiripirit and Amudat. Currently, Nabilatuk Parish administers its services to the two sub-counties of Lolachat and Nabilatuk in Nakapiripirit District. Nabilatuk Parish station is located in Nabilatuk Sub-County.

Catholic Parishes manage a range of data and information books. Most important, are registers where records about the sacramental information are kept. These records are kept as a demand by the mother church, since they manifest the membership of the church (Dr. Ludwig Ott, 1952). The more frequently visited and used registers in, particularly in Nabilatuk Parish, include: Baptism Register, First Holy Communion Register, Deaths and Burials Register, and the Confirmation Register.

Much of the register management duties are meant to be carried by the Parish Priest, but due to the work overload and the nature of the traditional book-keeping system, the Parish Priest employs, and is most often assisted by, the seminarians and catechists. This puts the data accuracy into question, because afore mentioned novices appear illiterate and not to have undertaken the necessary training for the maintenance of archival information as opposed to the priests.

In 1993, Nabilatuk Parish presbytery was burnt down by fire outbreak, and this led to the loss of several documents. In the process of recovery of the lost registers information, distortion and harm was made on the accuracy and credibility of sacramental register. Looking closely at the registers, you realize that the current management system is incompetent in that, the numbering system is not developed and is discontinuous; in many entries you discover that the numbers have either been skipped or repeated, and yet the serial numbers must be unique since they are used for reference and form the primary key to every kept record. As a rule in the Roman Catholic record maintenance, every member must have a unique Baptism number, Confirmation number and communion number, to mention but a few.

The inconsistencies of the current system have rendered the registration process incompetent and unproductive, thus raising the need for the designing and implementation of the computerized/automated system.

1.2 Problem statement

The dawn of computerized and automated system has brought with it an excitement that has rendered working in bulky books irrelevant and unpleasant. More so, security issues are cropping in the management and control of the use and access of Sacramental registers and the other sensitive information of the parish. The parish has also in many occasions spent reasonable amounts of money for purchase of registers and facilitation of the many data clerks employed to sort, enter or retrieve data in registers. With the current system, it is very difficult to retrieve data since it takes clerks a great deal of time running fingers through the lines of the various books in order to locate a record required by the client. Besides, Registration books make the office space very untidy and crowded and the data in registers is disorderly, inconsistent and inaccurate. The numbering sometimes confounds, especially when semi-trained clerks repeat or skip the numbering.

The aforementioned concerns about the current system therefore raise the need to improve data processing and to protect the register information from abuse and misuse by design and implementation of an automated register management system.

1.3 Purpose of the study

The purpose of this project is to design and implement an automated computer-based registration and register management system for Nabilatuk Catholic Parish.

1.4 Objective of the study

1.4.1 General objective

To design and implement an automated computer-based registration and register management system for Nabilatuk Catholic Parish

1.4.2 Specific Objectives

i. To carry out an analysis of the requirements for an automated register management system of a Catholic Parish.

ii. To design an automated register management system for Nabilatuk Catholic Parish.

iii. To implement the automated register management system for Nabilatuk Catholic Parish.

1.5 Justification

There have been attempts made by the current Parish Priest of Nabilatuk to computerize the sacramental registers and other information systems of the parish; however, the priest has always used methods and applications that require the user to feed the information and carry any computing processes manually.

The information entered in the excel sheets and word documents for instance, can still be (in terms of inconsistencies) comparable to the undesired Manual/Book-keeping system.

Once designed and implemented, the automated registration and register management system will deliver accurate information, guard against abuse of information by unauthorized users and will motivate the priests to enjoy managing registers. The system will arrange the records in order of their entry and will outwit duplication of records and sharing of serial numbers which is a typical of the current manual book keeping system.

The project will also reduce the cost of purchase of registers and will ensure safety and availability of information in case of any disasters, like fire outbreaks. The registration and management system will make back-ups, registration and records retrieval quick, efficient and effective.

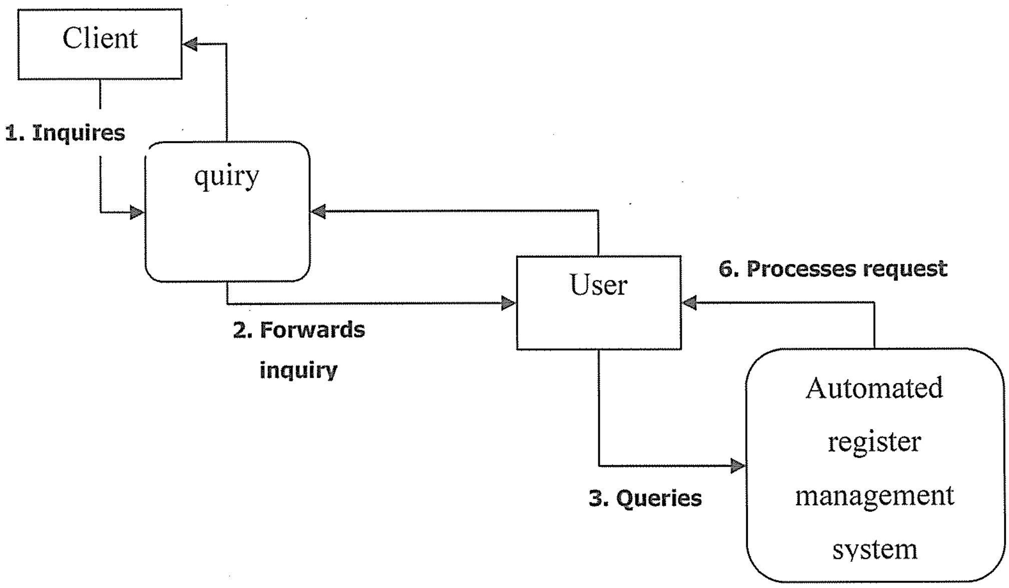


Figure 1 A context diagram for the proposed automated register management system

1.6 Conceptual Framework

1.6.1 Description of the context diagram

The client (catechumen, a child's parent, a priest or the Bishop's office) initiates action by sending a request which they fill in a inquiry form. Upon receipt of the request, the user of the system (Parish Priest, Catechist or Seminarian) logs in and operates the system to service the request of the client; the system then automatically checks/updates the database and retrieves the information to the forms or reports (depending on the query/command of the user) and then prints a hard copy for the client.

CHAPTER TWO

LITERATURE REVIEW

2.0 Introduction

This chapter looks at the concept of database management systems, information systems; information management system, system development approaches, unified modeling language, and the recent developments relating to the proposed system. The importance of this section is to relate the proposed system to what other researchers have written about similar systems and to inform this research and systems development.

2.1 Register system of the Roman Catholic Church

Unlike its other Christian counterparts, the Roman Catholic Church recognizes all the six sacraments instituted by the council of Nicea (Baptism, Confirmation, Holy Eucharist/Communion, Anointing of the sick, Holy Orders, and Matrimony) Information about the reception of this sacrament is kept into special registers that form the church's register systems. Whereas a Parish is obliged to keep her registers, the office of the Bishops periodically request for either the state or information of registers through formally prescribed requests. All the registers, except that of Holy Orders are kept in the Parish archives.

In Ireland, surviving Roman Catholic baptism records usually record the date of baptism, the child's name, the father's name in full, the mother's first name and maiden surname, the name of any godparents (sponsors) and the residence of the parents. Unfortunately, this latter element does not always appear (Irish genealogy toolkit, 2011).

Unlike the Irelands church registers, the current Catholic baptism registers today have included Registration number and Date of birth.

A marriage entry typically includes the first name, surname, age, father's name and occupation, and place of residence for each of bride and groom. In addition, the address of the church where the ceremony took place is provided, as is the name of the officiating priest, and the names of two witnesses. The latter are often a brother or best friend of the groom and a sister or best friend of the bride but this is not always the case.

The place of residence was sometimes omitted in earlier registers but after the 1860s this became rarer because priests were provided with new registers which included a section for addresses (Irish genealogy toolkit, 2011).

Death and Burial registers are, at observation, the most neglected. They rarely contain sufficient information to prove their worth. The researcher only dims designing this registers necessary because he hopes there will be concern for them in this century where death and birth certificates have gained a meaningful value. It is hard to carry analysis of the burial registers because formally, where they did survived, Catholic burial registers contained only the name of the deceased and the date of burial. However, a look at the recent Death and burial registers sold at the Pauline Bookshop-Kampala, suggests a great improvement made for fields required.

The Confirmation and Holy Communion registers entirely depend on the entries of the Baptism registers, except that they go ahead to add "Date" of reception and specify the "Parish of Baptism" and the priest or bishop who administer the sacrament.

A challenge normally one would meet in redesigning the Catholic register is the use of the Latin literature. For instance, a typical full-form Latin entry in a Roman Catholic baptism register would read: "BaptisaviMichaeli, filiumlegitimumPatricus Daly etEllena Driscoll de Courtmacsherry. Sponsoribus Johannes Doyle, Marian Shea." Sometimes: this might be abbreviated to BaptMichaeli, flPatricus Daly etEllena Driscoll, Courtmacsherry. Sp John Doyle, Marian Shea. (Irish genealogy toolkit, 2011).

The translation is: I baptized Michael, legitimate son of Patrick Daly and Ellen Driscoll of Courtmacsherry. Godparents John Doyle and Mary Shea.

2.2 Information Systems

An information system is defined as a well-coordinated collection of resources that gather and transform data into information products and services that help the enterprise to perform its designed functions. (David Harris, 1996). David Harris further argues that the most distinctive way to define information systems is to focus on what they do.

An information system has also been defined as a system that collects data, stores and computes business transaction data and presents the result of processing to the management in an organization in form of information for decision making. This collected information can be used for carrying out statistical reports (Cornell 2005). Cornell further states that building a comprehensive information system is time consuming and requires significant financial and labor resources. Collecting appropriate data sets, analyzing data, and organizing these data are challenging tasks and require significant effort. However, the benefits of having a comprehensive information management system greatly outweigh the difficulties.

An information system has been referred to as arrangement of people, data, processes, information presentation and information technology that interact to support and improve day-to-day operations in a business as well as support the problem solving and decision making needs of management and users. (Jeffrey L. Et al, 2000).

2.3 Management Information Systems

Management Information Systems (MIS) is the term given to the discipline focused on the integration of computer systems with the aims and objectives on an organization. The development and management of information technology tools assists executives and the general workforce in performing any tasks related to the processing of information.

When information systems are designed to provide information needed for effective decision making by managers, they are called management information systems. "MIS is an information system application that provides for management-oriented reporting. The reports are usually generated on a predetermined schedule and appear in a prearranged format". (Jeffrey L. Et al, 2000).

A management information system has been defined as an integrated machine system that provides information to support the planning and control function of managers in an organization. The output of an MIS is information that serves managerial functions. When a system provides information to persons who are not managers, then it will not be considered as part of an MIS. Generally, MIS deals with information that is systematically and routinely collected in accordance with a well-defined set of rules (www.management-hub.com, 2011).

2.4 Database Management Systems

Keneth and Laudon (2003) define a database management system as "simply a software that permits an organization to centralize data, manage them efficiently, and provide access to the stored data by applications program". The database management system relieves the end user from the task of understanding where and how data are actually stored, by separating the physical and the logical views of data.

2.5 System development approaches

2.5.1 Prototyping

Prototyping has been described as a design-development methodology that facilitates continued user involvement in the project. Prototyping has been proposed as a method to use for systems that are not overly complex. A system designed with prototyping is much easier to change because it was designed to be modified from the start (Gerald V & David A.).

In as much as prototypes are treated as working models of the proposed system, some prototyping occurs naturally as the design element for I/O and processing are prepared through joint application design activities; such informal prototypes might involve pen-and-paperwork. Prototyping is often treated as an integral part of the system design process, where it is believed to reduce project risk and cost. (David Harris, 1999)

More prototypes are made in a process of iterative and incremental development where each prototype is influenced by the performance of previous designs. In this way, problems or deficiencies in design can be corrected. 'When the prototype is sufficiently refined and meets the functionality, robustness, manufacturability and other design goals, the product is ready for production (En.wikipedia.org, 2011).

Early visibility of the prototype gives users an idea of what the final system looks like, increases system development speed, assists to identify any problems with the efficacy of earlier design, requirements analysis and coding activities and helps to refine the potential risks associated with the delivery of the system being developed. However, a producer might produce a system inadequate for overall organization needs and there is a possibility of causing systems to be left unfinished (En.wikipedia.org2011).

For that reason, therefore, the above approach cannot be chosen in isolation for the proposed system since the Nabilatuk Parish registers needs are essential to develop the system.

2.5.2 Water fall model

Waterfall is a sequential software development model in which development is seen as flowing steadily downwards through the phases of requirements analysis, design, implementation, testing (validation), integration and maintenance (Bronzite M, 2000).

However, the waterfall model involves the inflexible division of a project into separate stages, so that commitments are made early on, and it is difficult to react to changes in requirements. Iterations are expensive. This means that the waterfall model is likely to be unsuitable if requirements are not well understood or are likely to change in the course of the project. Therefore, this approach, too, cannot be chosen to develop the proposed system because of the above reasons.

2.5.3 Incremental Model

The incremental model has been said to be an intuitive approach to the waterfall model. Multiple development cycles take place here, making the life cycle a "multi-waterfall" cycle. Cycles are divided up into smaller, more easily managed iterations. Each iteration passes through the requirements, design, implementation and testing phases. A working version of software is produced during the first iteration, so you have working software early on during the software life cycle. Subsequent iterations build on the initial software produced during the first iteration.

The incremental model generates working software quickly and early during the software life cycle, More flexible and Easier to test and debug during a smaller iteration. However, problems may arise pertaining to system architecture because not all requirements are gathered up front for the entire software life cycle (Raymond, L. 2005).

This approach, just like the waterfall model, is not appropriate because it requires obtaining all system requirements once before development.

2.5.4 Spiral Model

The spiral model has been said to be similar to the incremental model, with more phases placed on risk analysis. The spiral model has four phases: Planning, Risk analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). Requirements are gathered during the planning phase. In the risk analysis phase, a process is undertaken to identify risks and alternate solutions. A prototype is produced at the end of the risk analysis phase. Software is produced in the engineering phase, along with testing at the end of the phase. The evaluation phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral (McConnell. S, 2006).

The spiral model is used most often in large projects and for this reason, therefore, it may not produce good results for small projects like the proposed project.

2.5.5 Rapid Applications Development (RAD)

RAD is a term originally used to describe a software development process. The methodology involves iterative development and the construction of prototypes. RAD approaches may entail compromises in functionality and performance in exchange for enabling faster development and facilitating application maintenance (James, M 1991).

Rapid application development was a response to non-agile processes developed in the 1970s and 1980s, such as the structured systems analysis and design method and other waterfall models. One problem with previous methodologies was that applications took so long to build that requirements had changed before the system was complete, resulting in inadequate or even unusable systems. Another problem was the assumption that a methodical requirements analysis phase alone would identify all the critical requirements. Ample evidence attests to the fact that this is seldom the case, even for projects with highly experienced professionals at all levels (James, M 1991).

This is the approach that this study will follow because it is an iterative and incremental process, which allows the refining of user requirements and the prototypes developed enable faster development and facilitation of application requirements definition.

2.6 System Development Tools

2.6.1 Unified Modeling Language (UML)

The UML is a way of representing the various requirements, relationships and other software development concepts in a way that can be used to help model a system. It is a methodology to model the processes, states and classes in a software project. There are many different ways in UML of representing relationships and data flow. Among these are Class, Use Case, State Machine, Collaboration and Sequence diagrams. It can also be used to model other things besides software; it is sometimes used to model business logic. One of the advantages of using a standardized methodology like UML is that if developers need to communicate ideas it is easier as all the concepts are there to be used. Many software tools are available which make the processes in UML easier to implement (Dmwiki, 2005).

2.6.2 Visual Basic (VB)

VB is a RAD system for normal windows applications, re-usable components, database applications and internet applications (Jurgen Bayer, 2002). Visual basic is one of the modern programming environments where you don't need to create the interface for an application in the program code, but instead make use of the readymade forms and controls. The advantage with using VB is the fact that any programming you will need to do will focus on solving a problem and not the shape or color of the interface; and you can also include the other objects like ActiveX components to VB, for as long as they have source code (Jurgen Bayer, 2002).

VBA, the programming Language used in VB guards against some errors that are common in other programming languages like C++ or Delphi, which can lead to long nights of frustrating searches of errors (Jurgen Buyer, 2002); instead VB makes programming easy by offering the programmer with the several operations to a given class at the coding. That is to say, VB preempts the programmer so as to help him/her choose from alternatives for as long as the object invoked is existent.

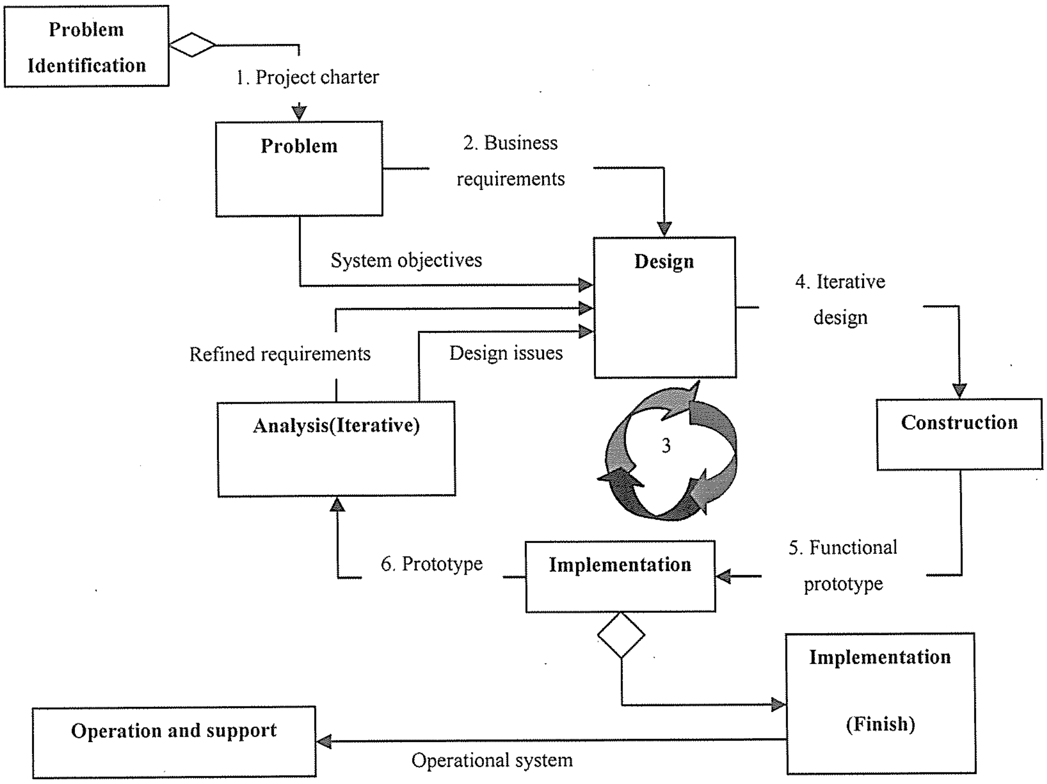
ADODC engine found in VB offers a unified object model and some excellent features. It allows access to a database system from within the windows application.

CHAPTER THREE

METHODOLOGY

3.0 Introduction

This chapter discusses the approaches to data collection, techniques for data analysis that will be used for designing and implementation of the proposed system. This section comprises of the research design which describes the tools, instruments, approaches, processes and techniques that will be employed in the research study, requirements collection, analysis, design, logical flow implementation, testing and validation as detailed below. The system will be developed through the following stages:



3.1 Requirements definition

A critical study of the existing system will be carried out and evaluation will be done to identify inefficiencies that the system seeks to address. The following fact-finding methods will be used to establish the requirements of the developed system.

3.1.1 Reading documents about the existing system

The researcher will review existing documents of Nabilatuk Parish such as reports, registers and any canonical writs regarding registration and keeping of archives to establish the inputs required for the development of the automated registration and register management system.

3.1.2 Interviewing relevant authorities

An interview is defined as a formal face-to-face meeting for instance, a conversation, as one conducted by a reporter, in which facts, or statements, are obtained. The advantage with this technique is that it is a flexible and a better tool than a questionnaire for the evaluation of the validity of the information that is being gathered avoiding misunderstandings and carefully evaluating the responses (martymodell.com 2007).

As per this proposed project, interviews will be conducted for members of staff, such as the Parish priest, catechist and seminarians to avail the researcher with information concerning the current system and the difficulties encountered.

3.2 Data analysis

The requirements obtained will be analyzed and further categorized into user, system, functional or non-functional requirements so as to differentiate, simplify, and refine requirements when changes occur.

3.3 Design tools

The information obtained will then be structured into diagrams like, data flow diagram (DFDs), entity relationship diagrams (ERDs) and data dictionary to model and map the flow of data among entities. The design of Nabilatuk Parish automated register management system will involve conceptual, logical and physical designs so as to produce a complete and detailed specification of the system components and of the database to be maintained by the system. To aid the faster development of designs, Essential Business Modeler will be used to generate Data flow diagrams, Entity Relationship diagrams, and process communication (transaction) diagrams.

3.4 Implementation

Here, individual system components will be built and tested. This involves the use of SQL to support and provide back-end programming of the system database, and VB 6.0 for developing the graphical user interface and for communication between the graphical user interface and the database.

3.5 Testing and Validation

This will cover the functionality of the system and its usability. It involved presenting the system to stakeholders and interacting with them to get their views on the system, consultations with a team of experts and use of test data to ensure that the user gets the expected output. In the case of the proposed system, a copy of the packaged system will be distributed to the supervisor, school of computer studies and to Nabilatuk Parish; the recommendations of these stakeholders will be analyzed and considered for implementation by the researcher.

CHAPTER FOUR

SYSTEM DESIGN

4.0 Introduction

This chapter presents the conceptual, logical and physical design of the system. It handles the preliminary design then the detailed design. It as well includes diagrams which will facilitate the users' understanding of the new system. Entity relationship diagram, class Diagram and Data Flow Diagrams were put into consideration. The diagrams will serve to facilitate the users understanding of the new system. The purpose of this chapter was to develop a design of the intended system.

4.1 Entity Relationship Diagram of the new system.

An entity relationship model is part of system development methodology that provides an understanding of the logical data requirement of a system independently of the systems' organization and process. It also reflects a static view of the relationship between different entities.

Below is a figure of the ERD of the system:

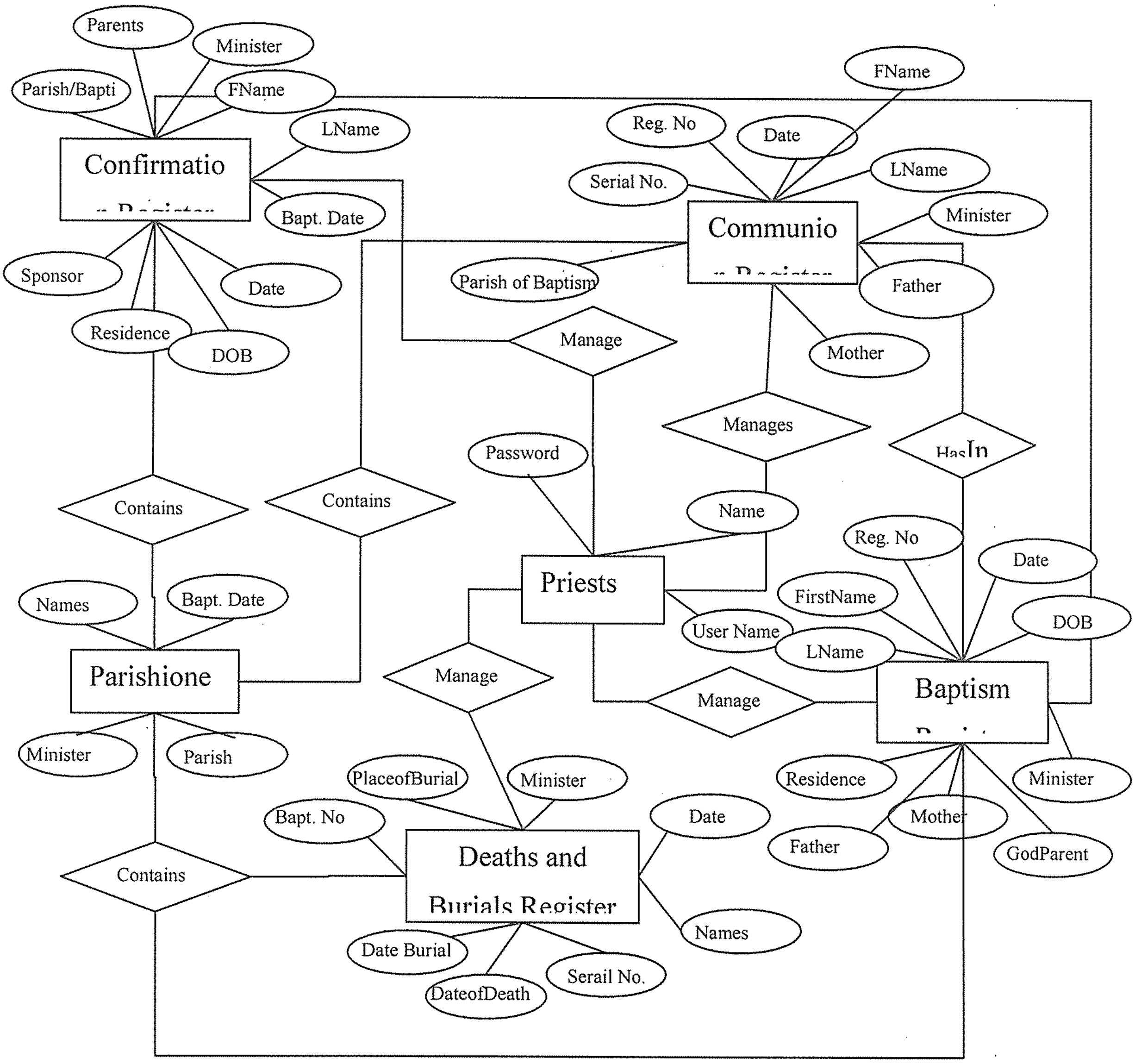


Figure 4.3 Entity Relationship Diagram

4.2 Class Diagram

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or) methods and the relationships between the classes.

Figure 4.3 below illustrates the relationship between the classes of the new system

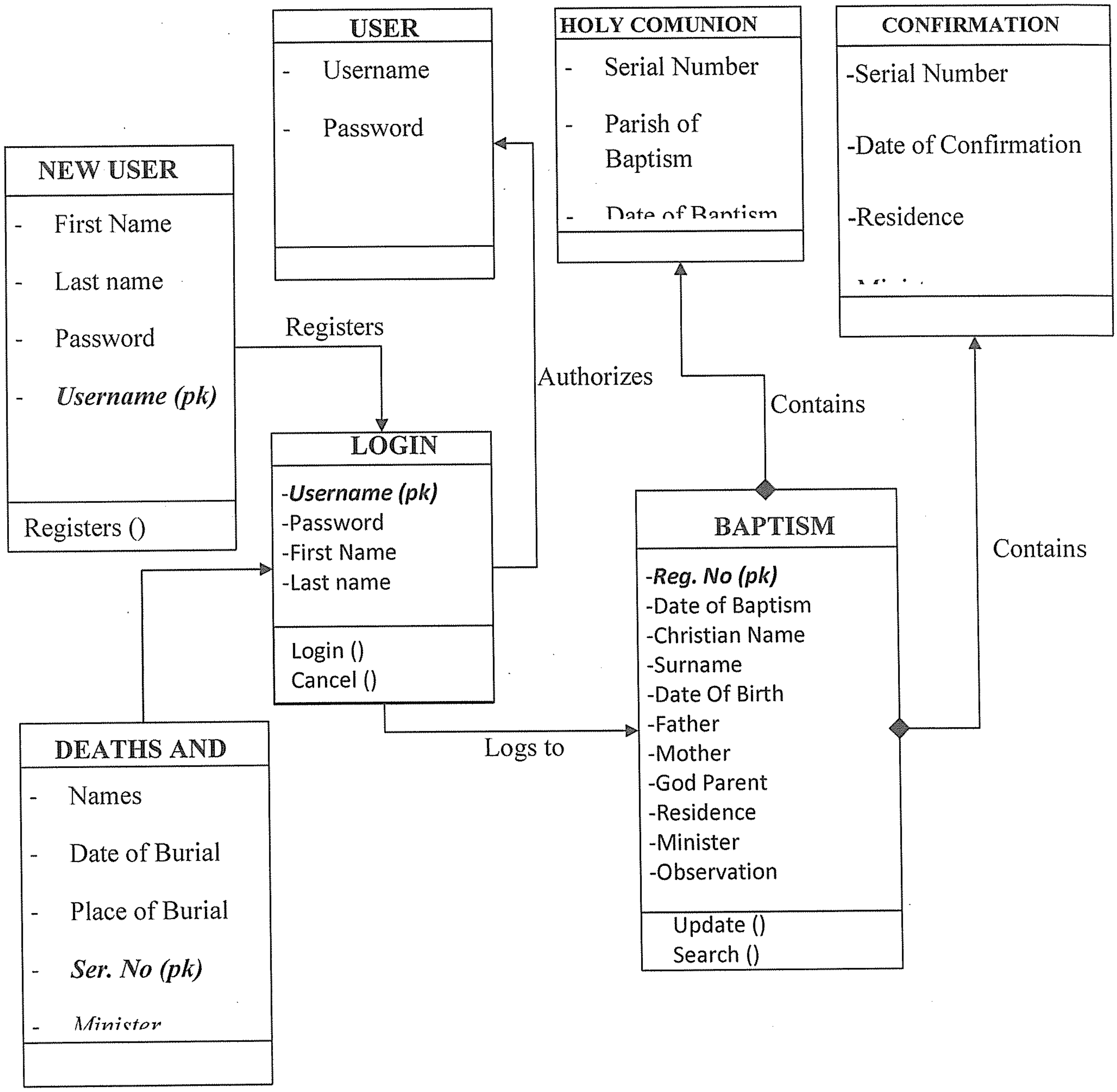


Figure 4.4 Class diagram for the new system

4.3 Data Flow Diagrams

A **data flow diagram (DFD)** is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing

On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process.

4.3.1 Data Flow Diagram for registering new user

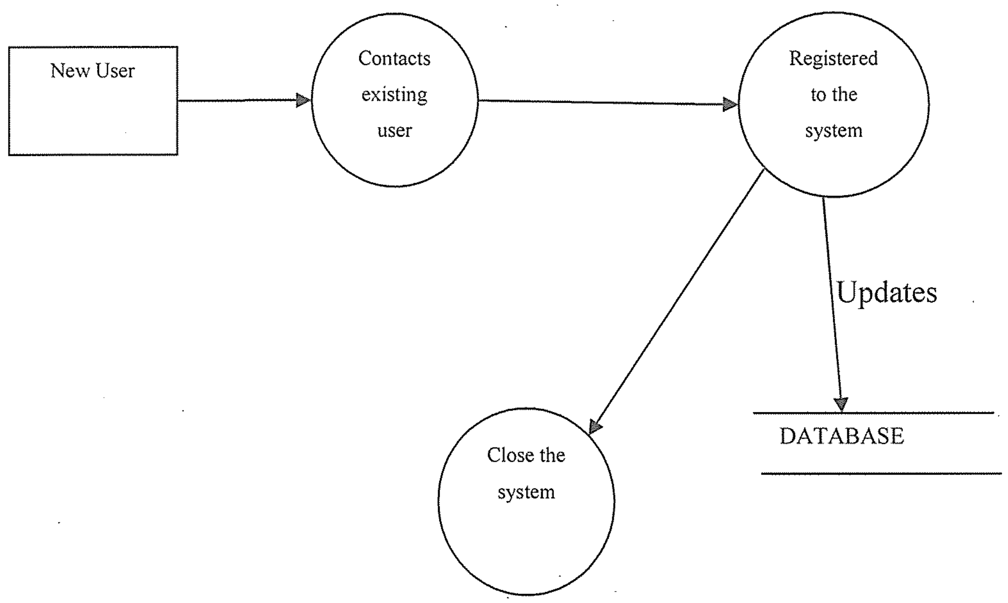


Figure 4.5Data flow diagram for registering new user

4.3.2 Data Flow Diagram for Logging into the system

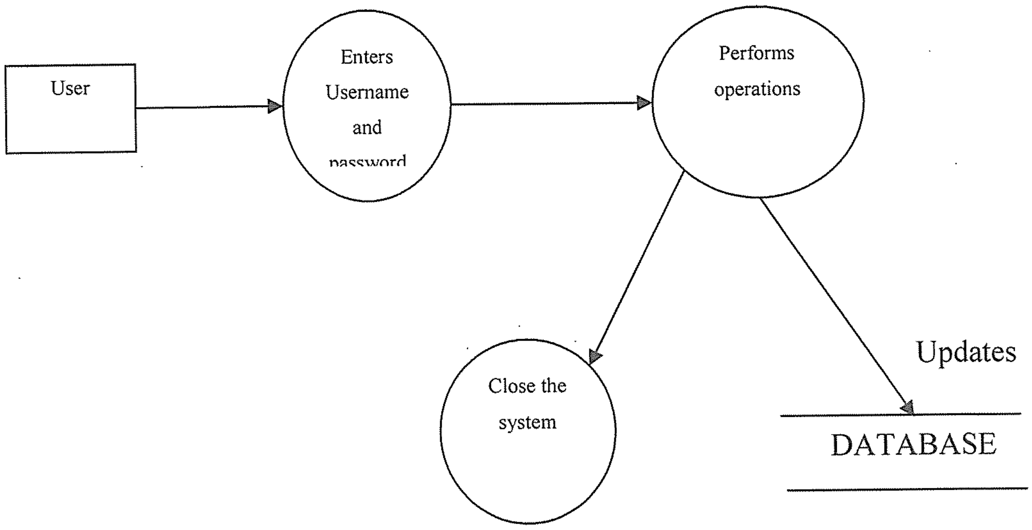


Figure 4.6 Data Flow Diagram for logging into the system

.3.3Data Flow Diagram for Searching details

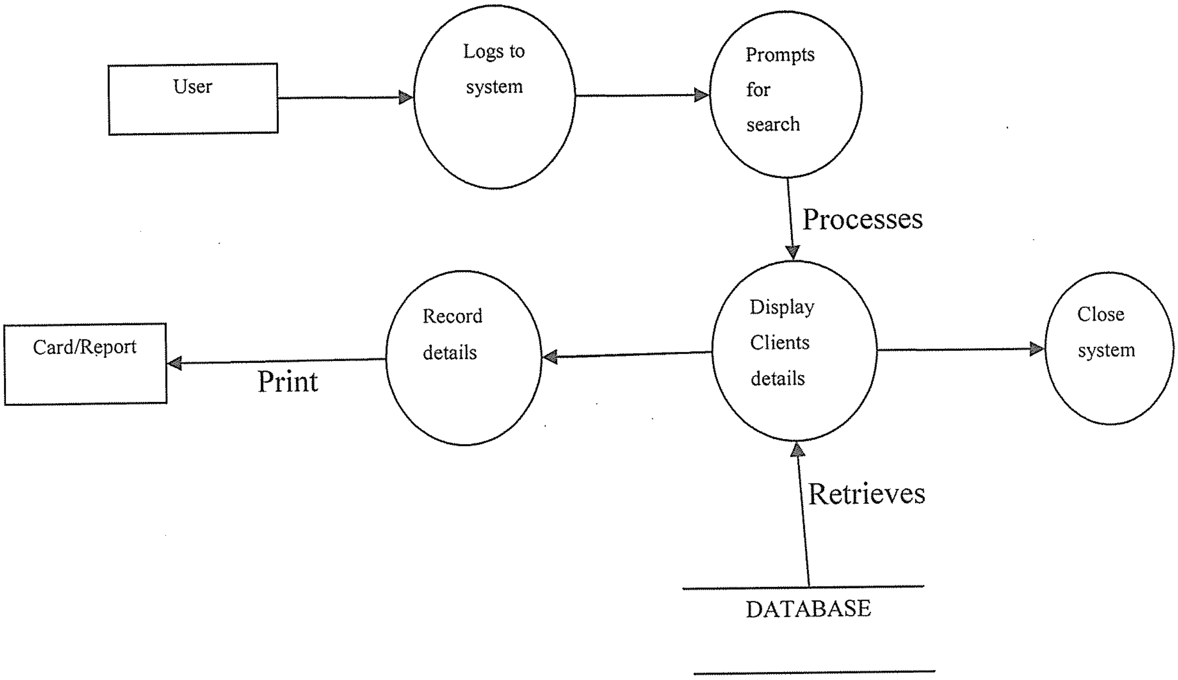


Figure 4.7 Data Flow Diagram for Searching details

4.4 Database Design

File management of the system will use an access database as its core driving force. All files will be managed in a single database. Reports will draw data from the various views. This will eliminate inconsistency as well as redundancy. However, tables will form the basic database structure.

4.4.1 Table Structures

Field Name	Data type	Data size
Reg.Number (pk)	Text	255
DateOfBaptism	Date/Time	Short Date
ChristianName	Text	255
Surname	Text	255
DateOfBirth	Date/Time	Short Date
Father	Text	255
Mother	Text	255
GodParent	Text	255

Residence1	Text	Medium Date
Minister1	Text	255
Observation	Memo	
SerialNum	Number	Long Integer
DateOfCommunion	Date	Short Date
ParishOfBaptism	Text	255
SerialNumber	Number	Long Integer
DateOfConfirmation	Date/Time	Short Date
Residece2	Text	255
Sponsor2	Text	255
Minister2	Text	255

Table 1 Sacraments of Christian initiation

Field Name	**Data type**	**Data Size**
SerialNumber(PK)	Number	Long Integer
DateOfEntry	Date/Time	Short Date
Names	Text	255
DateOfDeath	Date/Time	Short date
Baptism Number	Text	255
PlaceOfBurial	Text	255
DateOfBurial	Date/Time	Short Date
Minister	Text	255

Table 2 Deaths and Burials

Field Name	**Data type**	**Data Size**
UserName (PK)	Text	255
Password	Text	255
Firstname	Text	255
LastName	Text	255

Table 3 Login Details

4.4.2 Data Input Design

All the Data/command is entered using a mouse and keyboard by users. A user will input text to textboxes and make commands by clicking user-friendly buttons provided in the graphical user interface of the system.

4.4.3 Data Output Design

The system produces Forms and reports which are viewed via the computer screen and can be printed for use by the relevant people. The reports are majorly in forms of registers.

4.5 Conclusion

The chapter basically covered the physical, logical, conceptual design of the system with all the diagrams that support each design level. The next chapter will look into the implementation of the new system.

CHAPTER FIVE

SYSTEM IMPLEMENTATION

5.0 Introduction

This chapter deals with how the new system was implemented. It includes how the system operates and supports the users. The chapter also covers the different ways in which the old system is converted to embrace changes brought by the new system developed.

5.1 Program Testing

This was intended to ensure that the system is consistent and conforms to its specification and that the system meets the expectations of the users. The testing process was proceeded in stages where testing was carried out incrementally with system implementation. The following were the stages to be followed:

Unit testing: Individual components were tested to ensure that they operate correctly. Each component was tested independently without other system components. The database units were also tested in this stage.

Module testing: A module encapsulates related components, thus tested without other system modules. A module was tested independently to check for bugs and efficiency.

System testing: - This process was with validating that the system meets its functional and non-functional requirements and testing the developing system properties.

5.2 User Training and Documentation

The implementation of the new system involves training individuals who will use the final system and developing a documentation to aid the system users. It includes an audit to gauge the success of the completed project.

This stage was not achieved as desired, since the researcher did not have time to complete the exercise.

5.3 System implementation

In this stage, the new system was installed and data loaded to the new database. Conversion to the new system is a significant milestone. After conversion, the ownership of the system officially transfers from the researcher to the end-users. The researcher completed this task by carefully carrying out the conversion plan.. The system owners provided feedback regarding the new system that has been placed into operation. The system users provided valuable feedback pertaining actual use of the new system. They were the source of the majority of the feedback used to measure the system's acceptance.

5.4 Structural overview of the new system

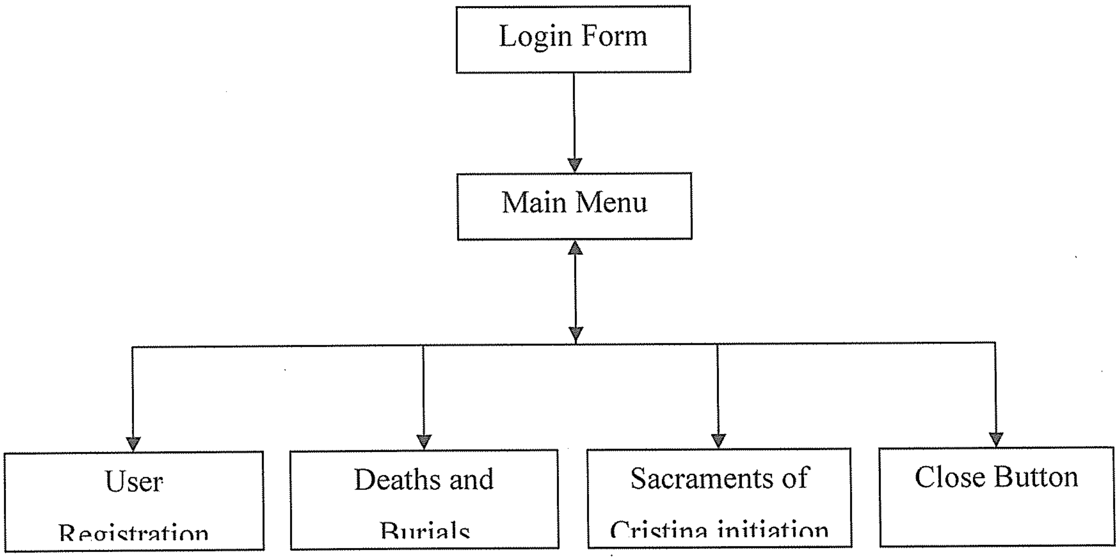


Figure 8 Structural overview of the new system

5.5 User interface

User interface will help the user in the implementation process. Following are a series of steps and procedures on how to gain access and operate the new system.

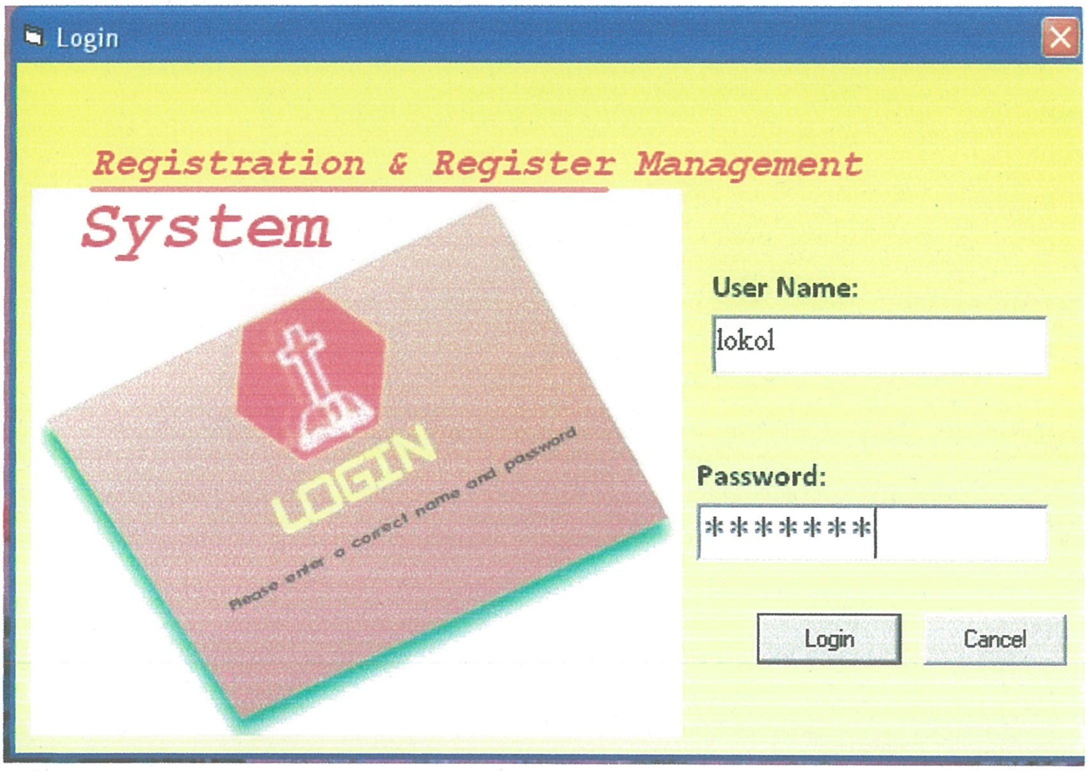
The system starts with a Login Form which leads to the Main Menu where a user will then choose a desired form.. The Login interface is to authenticate a user; and it requires a user to enter the authentic username and password.

Once a user has chosen/clicked a desired form, operations can be made by either entering, viewing or acting on (saving, deleting or editing) the information/records displayed on the form. The user shall use the buttons or menus to perform any desired actions on the system.

Note that the **close/exit** and the **delete** buttons will generate a confirmation message for in avoidance of lose of data. The user is required to click a save button whenever the system is about to be close. This ensures that any corrections are maintained to the system.

Security Requirement: All the users have to first log in with a correct username and password to gain access as shown below

5.5.1 Login Form


Login
Registration & Register Management System
LOGIN
Please enter a correct name and password
Login
Cancel

User Name: lokol

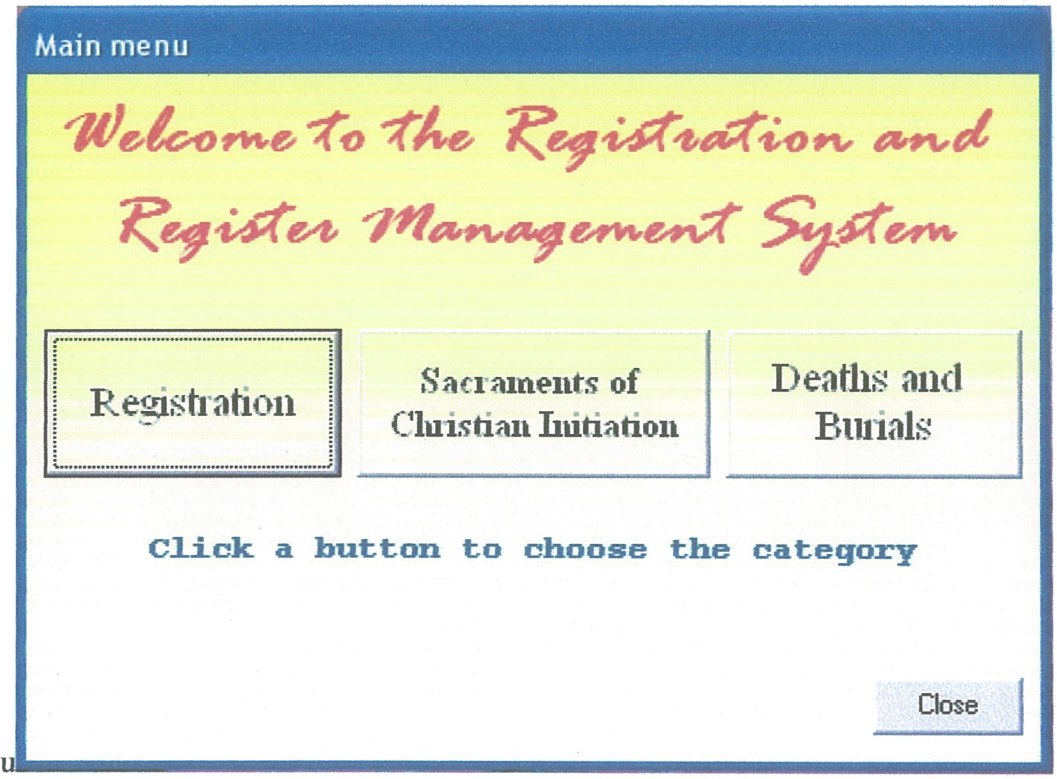
Password: *******

Figure 9 Login Form User interface

The Login form provides two text boxes and buttons. The text boxes allow the user to enter in the username and password. Upon clicking the **Login** button, the user will access the other forms only if he/she is already registered; else, the access will be denied.

A user can only try logging in three times upon which the system can close if the user does not provide right credentials.

5.5.2 Main Menu



Welcome to the Registration and Register Management System

Click a button to choose the category

Figure 10 Main menu interface of the system

The main menu form buttons include:

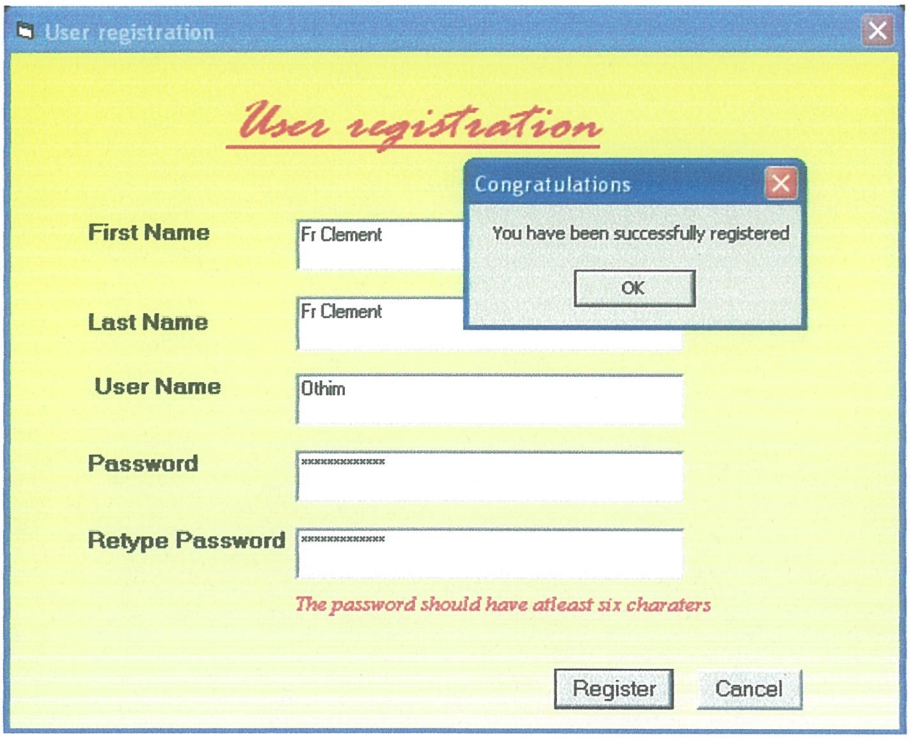
Registration: This button is used to access the registration form; where new system users are registered to acquire authentication and authorization rights.

Sacraments of Christian Initiation: This button is used to open the registration form where the Baptism, 1st Holy Communion and Confirmation can be accessed and updated.

Deaths and Burials: The Deaths and Burials button is used to open the Deaths and Burials Form where users can update Deaths and burial records.

Close: This button closes the form and the application.

5.5.3 User Registration Form



User registration

User registration

Congratulations

You have been successfully registered

First Name: Fr Clement

Last Name: Fr Clement

User Name: Othim

Password: xxxxxxxxxxxxx

Retype Password: xxxxxxxxxxxxx

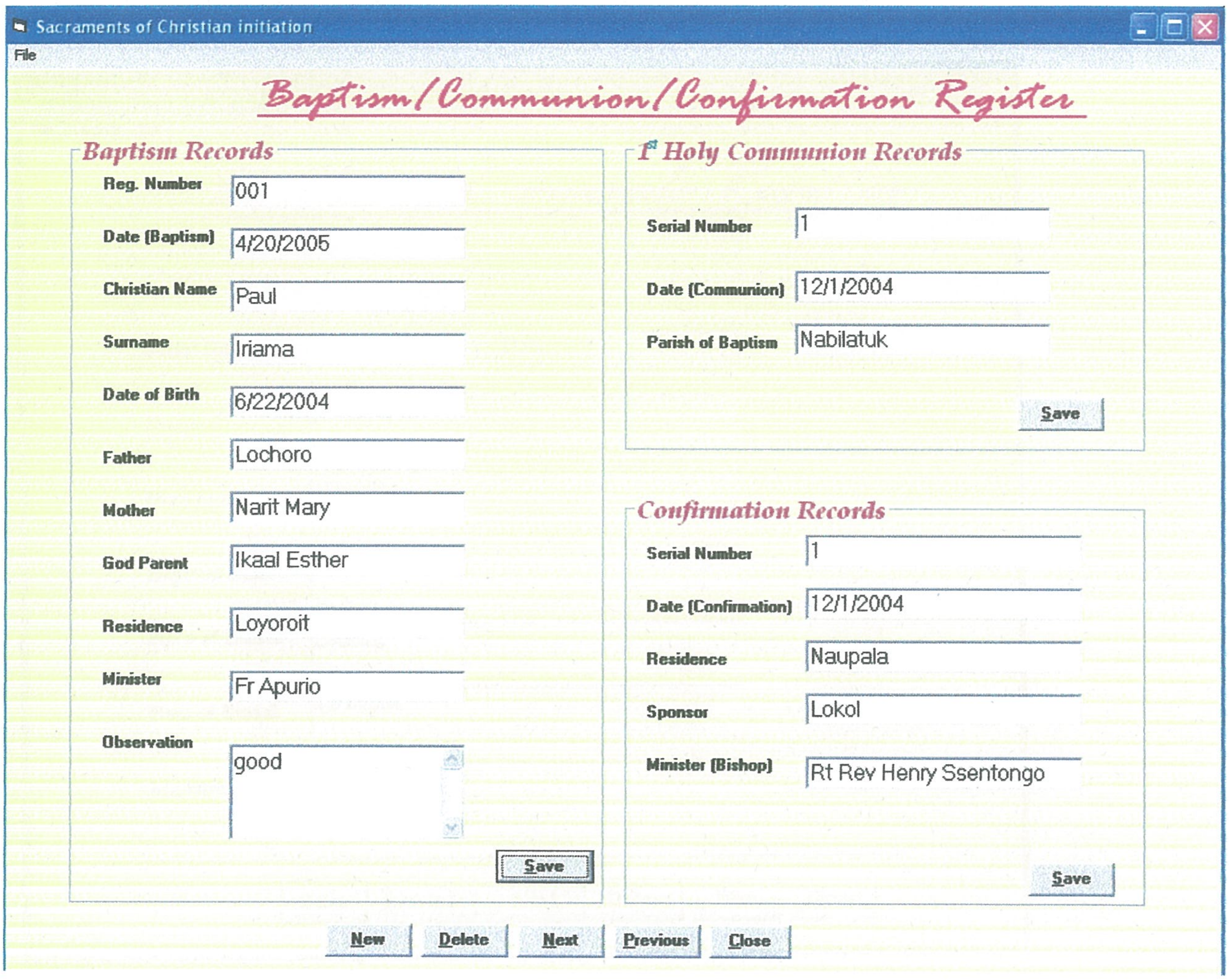
The password should have atleast six charaters

Figure 11 User Registration Form interface

This form updates the Login table in the database to register new users. New users should be registered by the already registered users before they can access the system. If the 'Password' field and the 'Retype Password' do not match, a user will not be registered and will be prompted to retype passwod.

5.5.4 Sacraments of Christian Initiation

Form



Baptism/Communion/Confirmation Register

Baptism Records

Reg. Number	001
Date (Baptism)	4/20/2005
Christian Name	Paul
Surname	Iriama
Date of Birth	6/22/2004
Father	Lochoro
Mother	Narit Mary
God Parent	Ikaal Esther
Residence	Loyoroit
Minister	Fr Apurio
Observation	good

1st Holy Communion Records

Serial Number	1
Date (Communion)	12/1/2004
Parish of Baptism	Nabilatuk

Confirmation Records

Serial Number	1
Date (Confirmation)	12/1/2004
Residence	Naupala
Sponsor	Lokol
Minister (Bishop)	Rt Rev Henry Ssentongo

Figure5.7 Sacraments of Christian Initiation Form

This form Captures detail about Baptism, 1st Holy Communion and Confirmation. It also enables the user to find records in the database easily. Reports of the various registers can be viewed by clicking the **File** menu, then open. The user shall ensure that records are saved in their appearance order.

The deliverable of the system implementation and project implementation is the operational system that will enter the operation and support stage.

Users, who are the most important element of the new system, were trained on how to feed in data, delete and update records. Various functions of the system were also explained to the concerned users. The users were trained on how to generate and print reports when need arises. This was done during the iterative testing processes.

5.7 Adapting the system to new requirements

New requirements may include business problems; new user requirements; new technical problems or new technology requirements which will need to be adapted into the new system.

5.8 Maintenance

Plans are developed for discarding system and making the transition to a new system. After implementing the systemthe Parish should continuously monitor performance of the system to ensure that it is consistent with pre-established user and security requirements, and that needed system modifications are incorporated. Configuration management and control activities should be conducted to document any proposed or actual changes in the security plan of the system.

5.9 Conclusion

When the system is implemented it must be maintained whereby files are updated and unnecessary information is deleted. It should show the workings of the new system and how the user should navigate through the system from login to the generation of reports

CHAPTER SIX

RECOMMENDATIONS AND CONCLUSION

6.0 Introduction

The researcher carried out a detailed study of the case study with the aim of analyzing the current system, designing the alternative automated system to replace the manual book keeping and to enhance implementation of the automated system.

From the research conducted, the researcher came up with the following conclusions and recommendations based on the research objectives for this study, which are stated below;

6.1 Recommendations

The researcher recommends for the system to be run on windows XP operating system, Windows Vista or 7, Hard disk size of at least 40GB and RAM size of at least 512MB.

Functionalities such as those that allow multiple access of the system should be implemented at a later stage to further enhance usability of the system.

The researcher recommends that before the application is put into full use, it should be tested in a sample field to estimate any bugs that may not have been identified at the time of development.

Passwords for users to login must be made long enough to strengthen system security so that unauthorized persons may not gain access into the system.

Once the system is in use, it should be run alongside manual files in order to prevent unexpected embarrassments, that is, parallel conversion should be adopted for the system; until a time when users and owners can trust the outputs of the new system

6.2 Areas for Further Work

The researcher recommends that efforts should be made for the development of a marriage registration form as well.

Other areas that the researcher deems necessary for further work are, but not limited to:

- Program maintenance
- Routine backups
- System recovery and
- Technical support

If opportunity allows, the researcher hopes that the features that have not been implemented in this application but were originally desired features will be taken into consideration in order to improve on the robustness of the system.

6.3 Conclusion

This system has not been pre-tested sufficiently to remove all the bugs that may not have been recognized at the time of developing this application. Exhaustive testing needs to be carried out to isolate these bugs and to make the system more robust.

The parish needs to implement the registration system for baptism, marriage, death and others for future reference so that the data for a particular person that is required for any reason can be access easily, especially with this fast advancing era of technology.

REFERENCE

C, J, DATE. 2000. An introduction to database systems. Addison Welsley (Ed.) Palpargan: India

C. J Date, (2001) An Introduction to Database System, 7th Edition, Pearson Education Publishers Newdelhi, India.

David Harris (1996); Systems analysis and design for small enterprises, Second Edition.

Effy Oz. 2002; Management Information Systems, 3rd Edition

Jeffrey L, Whitten. Lonnie D. and Kevin C. Systems analysis and Design Methods

Kenneth C, Laudon. Jane P, Laudon. 2002. Essentials of management information systems. Organizations and technology in the networked enterprise

Thomas Connolly, Carolyn Begg, Anne Strachan (1997). Database Systems.

http://www.en.wikipedia.or

http://www.irish-geneology-toolkit/Roman-Catholic-baptism.html

http://www.management-hub.com

Date C.J (2000), An introduction to Database Systems, Addison Welsley (Ed.) Palpargan: India

David Harris (1996); Systems analysis and design for small enterprises, Second Edition.

Gerald, V. & David L. Anderson, Management information systems, 2nd Edition).

Hoffer, G., &Valacich(1999). Mordern systems analysis and design, Third Edition

;

Whitten, B Etal (2000). Systems analysis and design methods, 5th Edition.
Irish Genealogy toolkit (2011), [Web page]. Retrieved 28-Mar-2011 from http://www.irsh-geneology-toolkit/Roman-Catholic-baptism.html
http://en.wikipedia.org

APPENDIX A

GLOSSARY

Association This is the relationship between entities.

Attribute Properties that describe the entity's characteristics. Examples of attributes are First Name, Registration Number

Column A logical group of bytes in a row of relation.

Consistency Two or more concurrent transactions are consistent if the result of their processing is the same as it would have been if they had been processed in the same serial order.

Constraint A rule concerning the allowed values of attributes whose truth can be evaluated.

Database A collection of data stored in a standardized format designed to be shared by multiple users.

Data Definition Language A language that supports the definition or declaration of database objects.

Data Integrity The state of a database in which all constraints are fulfilled; usually refers to inter-table constraints in which the value of a foreign key is required to be present in the table having that foreign key as its primary key.

Database Management System Is a collection of programs that enable users to define, create and maintain a database and provides controlled access to this database.

Data Manipulation Language A language which supports the manipulation or processing of database objects.

Entity An entity is something that can be identified in the users' work environment, something that the users want to track.

Entity Relationship Model Is a relational model whose key elements are entities, attributes, identifiers and relationships.

Field A logical group of bytes in a record used with file processing. A synonym for attribute.

Form A display on a computer screen used to present, enter and modify data. Also called data entry or panel.

Parish Is an administrative unit of the Roman Catholic Church under the control and management of a Priest.

Relationship A relationship is an association between entities.

Row A group of columns in a table. All the columns in a row pertain the same entity. The same as a turple and record.

Schema A complete logical view of the database.

Transaction A record of an event in the business world.

APPENDIX B

WORK PLAN

Activities	**Time Frame**					
	May	*June*	*July*	*Aug*	*Sept*	*Oct*
Requirements collection	▨	▨				
Data analysis and design		▨	▨			
Database design				▨		
Application interface design				▨	▨	
Coding and debugging					▨	
Testing and Validation					▨	▨
Report writing						▨
Packaging and handover						▨

APPENDIX C

BUDGET

No.	Item / Particulars	Quantity	Unit cost	Amount in UGX
1	Laptop Computer	1 PC	1,500,000	1,500,000
2	CD-ROM	8 pieces	500	4,000
3	Printing Paper	1 ream	15,000	15,000
4	Toner Cartridges	2 pieces	70,000	140,000
5	Fuel	10 liters	3,000	30,000
6	Communication	2 persons	50,0000	100,000
7	Transport	2 trips	100,000	200,000
Total				**1,989,000**

APPENDIX D

INTERVIEW GUIDE

i. Introduce yourselves

ii. Get name of respondent

iii. Ask about the importance and use of the current system

iv. Find out the challenges affecting the current system

v. Obtain the attitudes towards use of computerized systems.

vi. Make inquiries on how the current system captures and stores information.

vii. Ask about any challenges faced in operation and maintenance of the current system.

viii. Thank the interviewee and assure him that the information acquired will only be used for the development of a new automated system.

APPENDIX E

Sample Codes used to create the Application interface

Connection Module Code

```
Public cn As New ADODB.Connection

Function connect()

   If cn.State Then cn.Close

cn.open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" &App.Path& "\Project.mdb;Persist
Security Info=False"

End Function

Public Function JustDigits(KeyAscii As Integer)

   If (KeyAscii>= 48 And KeyAscii<= 57) Or KeyAscii = vbKeyBack Then JustDigits =
KeyAscii Else JustDigits = 0

End Function

Public Function JustName(KeyAscii As Integer)

   If (KeyAscii>= vbKeyA + 32 And KeyAscii<= vbKeyZ + 32) Or (KeyAscii>= vbKeyA And
KeyAscii<= vbKeyZ) Or KeyAscii = vbKeyBack Or KeyAscii = vbKeySpace Or KeyAscii = 46
Then JustName = KeyAscii Else JustName = 0

End Function
```

Login Form Code

```
Dim rs As New ADODB.Recordset
```

```
Private Sub cmdCancel_Click()

   Unload Me

End Sub

Private Sub CmdLogin_Click()

   Static count As Integer

   If rs.State Then rs.Close

rs.open "select * from Login", cn

      Do Until rs.EOF

      If txtUserName = rs("Username") And txtPassword = rs("Password") Then

            Unload Me

frmMain.Show

    Exit Sub

    Else

rs.MoveNext

   End If

   Loop

MsgBox "You may have entered wrong credentials, kindly contact the administrator if you aren't
registered", vbCritical, "wrong credentials!"

txtUserName.Text = ""

txtPassword.Text = ""
```

```
    If count = 3 Then End
count = count + 1
End Sub
Private Sub Form_Load()
connect
Me.Top = 2000
Me.Left = 3000
'Me.Height = 5000
'Me.Width = 7000
End Sub
```

Main Menu Form Code

```
Private Sub cmdDeaths_Click()
Me.Hide
frmDeaths.Show
End Sub
Private Sub cmdInitiation_Click()
Me.Hide
frmInitiationSacraments.Show
End Sub
Private Sub cmdCancel_Click()
```

```
End

End Sub

Private Sub cmdRegistration_Click()

Me.Hide

frmRegister.Show

End Sub
```

User Registration Form Code

```
Private Sub cmdCancel_Click()

Unload Me

frmMain.Show

End Sub

Private Sub cmdRegister_Click()

If txtPassword.Text = txtRetype.Text Then 'to confirm the knowledge of password entered

Register1.Recordset.Fields("FirstName") = txtFName.Text

Register1.Recordset.Fields("LastName") = txtFName.Text

Register1.Recordset.Fields("Username") = txtUsername.Text

Register1.Recordset.Fields("Password") = txtPassword.Text

Register1.Recordset.Update

MsgBox "You have been successfully registered", vbOKOnly, "Congratulations" 'to affirm registration
```

```
Else

MsgBox "The password you have retyped does not match.", vbCritical, "Password error" 'when a
theres is password mismatch

End If

Unload Me

frmMain.Show

End Sub

Private Sub Form_Load()

txtFName.Text = ""

txtLName.Text = ""

txtUsername.Text = ""

txtPassword.Text = ""

txtRetype.Text = ""

Register1.Recordset.AddNew

End Sub
```

Sacraments of Christian Initiation Form Code

```
Private Sub cmdClose_Click()

Dim message As Variant

message = MsgBox("You are about to close the application, Would you want to select another
category of records?", vbYesNoCancel, "Confirmation")

If message = vbYes Then
```

```
Unload Me

frmMain.Show

ElseIf message = vbNo Then

Unload Me

End

Else

End If

End Sub

Private Sub cmdDelete_Click()

Dim message As Variant

message = MsgBox("you are about to delete details of" + " " + txtSurname.Text + " " +
txtChristianName.Text + " " + "do you want to continue with the operation", vbYesNoCancel,
"Warning")

If message = vbYes Then

Adodc1.Recordset.Delete

MsgBox "You have succesfully deleted record", , "Confirmation"

Else

MsgBox "The record not deleted", , Notification

End If

End Sub

Private Sub cmdNew_Click()
```

```
Adodc1.Recordset.AddNew
End Sub
Private Sub cmdNext_Click()
If Not Adodc1.Recordset.EOF Then
Adodc1.Recordset.MoveNext
If Adodc1.Recordset.EOF Then
Adodc1.Recordset.MovePrevious
End If
End If
End Sub
Private Sub cmdPrevious_Click()
If Not Adodc1.Recordset.BOF Then
Adodc1.Recordset.MovePrevious
If Adodc1.Recordset.BOF Then
Adodc1.Recordset.MoveNext
End If
End If
End Sub
Private Sub cmdSave_Click()
Adodc1.Recordset.Fields("RegNumber") = txtRegistrationNumber.Text
```

```
Adodc1.Recordset.Fields("DateOfBaptism") = txtDateofBaptism.Text
Adodc1.Recordset.Fields("ChristianName") = txtChristianName.Text
Adodc1.Recordset.Fields("Surname") = txtSurname.Text
Adodc1.Recordset.Fields("DateOfBirth") = txtDateofBirth.Text
Adodc1.Recordset.Fields("Father") = txtFather.Text
Adodc1.Recordset.Fields("Mother") = txtMother.Text
Adodc1.Recordset.Fields("GodParent") = txtGodParent.Text
Adodc1.Recordset.Fields("Residence1") = txtResidence.Text
Adodc1.Recordset.Fields("Minister1") = txtMinister.Text
Adodc1.Recordset.Fields("Observation") = txtObservation.Text
Adodc1.Recordset.Fields("SerialNum") = txtRegistrationNumber.Text
Adodc1.Recordset.Fields("DateOfCommunion") = txtDateofHolyCommunion.Text
Adodc1.Recordset.Fields("ParishOfBaptism") = txtParishofBaptism.Text
Adodc1.Recordset.Fields("SerialNumber") = txtSerial.Text
Adodc1.Recordset.Fields("DateOfConfirmation") = txtDateofConfirmation.Text
Adodc1.Recordset.Fields("Residence2") = txtPlaceofResidence.Text
Adodc1.Recordset.Fields("Sponsor") = txtSponsor.Text
Adodc1.Recordset.Fields("Minister2") = txtBishop.Text
Adodc1.Recordset.Save
MsgBox "The record has been saved", vbOKOnly, "Confirmation"
```

```
End Sub

Private Sub mnubaptismrecords_Click()

rptBaptism.Show

End Sub

Private Sub mnuCommunionrecords_Click()

rptCommunion.Show

End Sub
```