

**COMPUTERIZATION OF STUDENTS' RECORDS
(A CASE STUDY OF GOVERNMENT GIRLS'
SECONDARY SCHOOL BIDA)**

by

GANNA, Janet

PGD /MCS/2008/1245

**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**

JULY, 2010

CERTIFICATION

This project titled "Computerisation of students' records (A case study of Government Girls' Secondary School, Bida)", by Gana Janet (PGD/MCS/2008/1245) meets the regulations governing the award of Postgraduate Diploma in Computer Science of Federal University of Technology, Minna.

Mr. Sirajo Abdulrahman
Supervisor

Date

Prof. N. I. Akinwande
Head of Department

Date

DEDICATION

This work is dedicated to the Author of success who is JESUS for His Divine protection and provision for me throughout the course of this study.

ACKNOWLEDGEMENT

I am most grateful to Almighty God for keeping me and availing me the opportunity to undertake this programme.

I wish to express my profound gratitude to my Supervisor, Mr. Sirajo Abdulrahman for his useful suggestion and guidance through the course of this study.

My gratitude also goes to all the staff of Department of Mathematics / Computer Science of the Federal University of Technology, Minna, who contributed greatly to the success of my PGD programme.

I am grateful to my Husband Dr. Gana, Andrew Kolo and children for their patience, love and encouragement.

I wish to thank all the members of Deeper Life Bible Church, Bida, for their love, prayer and encouragement.

ABSTRACT

This project work is based on developing a Computerized School Record. It is an attempt to replace the existing manual system with an effective and efficient Computerized - based System for processing school record. Government Girls' Secondary School, Bida is taken as case study.

TABLE OF CONTENTS

Title page	i
Certification	ii
Dedication	iii
Acknowledgement	iv
Abstract	v
Table of content	vi
CHAPTER ONE	
1.1 Background of the study	1
1.2 Objective of the study	2
1.3 Significance of the study	3
1.4 Scope and limitation of the study	3
1.5 Definition of some terms	3
CHAPTER TWO	
LITRATURE REVIEW	
2.1 Historical perspective of computerization of records	5
2.2 Method of collecting records	8
2.3 Problems of manual students record keeping	8
2.4 Advantage of computerization of students record	9
2.5 Disadvantage of computerization of students records	10
CHAPTER THREE	
3.1 System Design	11
3.2 Files Designing	12
3.2.1 Students admission or registration files	12
3.3 Programming languages	13
3.4 Materials to be used	14

CHAPTER FOUR

SYSTEM IMPLEMENTATION

4.1 Introduction	15
4.2 Operation Manual for the new system	15

CHAPTER FIVE

5.0 SUMMARY, RECOMMENDATION AND CONCLUSION

5.1 Summary	21
5.2 Recommendation	21
5.3 Conclusion	22
Reference	23
Appendix	25

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

Student's record is an important document to the staff, students, school and even the ministry of education. This is made evident in the way it is been in some schools and the security provided to ensure its safety.

A good student's record must be accurate, complete and legible for prompt decision making. This is best provided by a computer which is defined by Larry (1988) as an electro-mechanical device that can process information at a faster rate, with a very high degree of accuracy and at a high level of reliability. It can be used to store and search massive files of data and can solve wide variety of problems.

In the early 1950's, there was rigorous campaign to foster Girls' Secondary School among Nigerian youths, this gave rise to the opening of the Government Girls' Secondary School (GGSS) Bida in 1958 to be one of the purely girls school in Niger State. The school operate in accordance with the aims and objectives of the Federal Ministry of Education. Drawing her students and staff from all parts of the Federation, the students graduate in both science and arts subjects. Over the years the management of the school has been adopting the manual approach to all its processing activities. This manual approach has been characterised by many set backs which led to many discouraging reports about the students records. Some of the problems encountered are:

- i. The system has low efficiency
- ii. In terms of accuracy, it is relatively low compared with a Computer based system.
- iii. The system is strenuous for school record officers.
- iv. In term of storage, since register books are used in the manual system it occupy a lot of space and even after some years, it is

discarded to create room for new records which leads to lost information for future references.

- v. Insecurity or lack of privacy to students records, these and many other problems in the manual system call for an improved system, computerized system to be specific.

Therefore, this integral and important organ of the government needs a computer based information processing system. This information system will speed up the data processing and information handling activities of the school as a whole. It will also be able to generate appreciable and timely information which will serve as a basis for effective management decision making.

The design and implementation of a comprehensive computerized information system is solely what this project work is out to do.

1.2 Objective of the Study

This project work has many objective to achieve, all of which are geared towards computerized information system for the Government Girls Secondary School Bida's record.

These objectives are itemised as follows:

- i. To carry out a detailed investigation into the existing manual information processing system.
- ii. Identify the area(s) that need to be computerized.
- iii. Design new information processing system that will replace the short comings in the existing manual system.
- iv. Design and develop suitable computer programmes (database oriented) with the accompanying documents which when put to use will bring about a realization of the computer - based information system.
- v. Develop computerized system fast and easy preparation of statistical report for efficient managerial decision.

- vi. Have controlled and monitor access to database through the use of passwords so that only authorized personnel can gain access to the system.

1.3 Significance of the Study

The project is immensely justified or significant in the following areas.

- i. The school management will benefit a lot since it will help to have better and quicker access to information, that is storage and retrieval will be with ease.
- ii. Government will benefit since it will reduce number of workers that need to be paid every months.
- iii. It brings about an efficient, accurate, fast automated system for school record.
- iv. Students will also benefit since better access to information will directly or indirectly hasten the execution of their different results.

1.4 Scope and Limitation of the Study

The case study of this project is limited to Government Girls Secondary School Bida. The project work will be based on information gathered from student's records.

1.5 Definition of Some Terms

Below are the definition of the commonly used terms:

Computer: This is an electro - mechanical device, which is capable of processing data that is inputted by following a set of instructions called program to generate output, regarded as information.

Computerization: Is the conversion of manual procedure of processing data to get output [information] into an automated system.

Personal Record: A personal record can be defined as a document that gives information and other detail about a person.

Program: These are the sequential steps of how the Computer will execute instruction that is used to solve human problems.

Records: This is defined as an entire collection of related field. A record can contain personal data, academic data, office information etc.

System: System can be defined as a set of related component that work together in a particular environment to perform whatever function required to achieve the system objectives.

Software: Software is a program written for the proper functioning of the hardware. Without the software the hardware is useless, there are two types of software - Application software and system software e.g. Microsoft word, Microsoft excel, Fortran, Basic, Windows 95, 98, 2000 etc.

CHAPTER TWO

LITERATURE REVIEW

2.1 Historic Perspective of Computerization of Records

Shodipe (1994) defines record as any thing such as a document or a phonography or a photograph providing permanent evidence of information about past events. He further explained that a written document preserves knowledge of facts or events.

Larry (1988) described record as the process of assembling items of data into a meaningful form e.g. a payroll or a financial statement. Record can range from a simple report required by top management to make strategic decision.

Aronu (1999) in his book title "Computer Record System" refers to record as facts and opinion provided and received during the course of daily life activities. He further emphasized that one obtains record directly from another human beings, mass-media, electronic data banks and from all other sorts of observable phenomenon surrounding the environment. A person using such facts generates more information, some of which is communicated to others during discussion by instructions, in letters and via other media.

In the project write-up put together by Abubakar (2005), he defined a record system as a collection of machines, people, ideas and activities that process data in a manner that will meet the information need of an organisation. The machines are the technologies that accept instructions in many forms and from many sources. The people are the clerks, operators, analysts and programmers that work directly with the departments and division which supply the data or use the information contained in the processed report.

Badmus (2002) hinted that the curiosity of man to communication with not only his fellow human beings but also even with other creatures

like animals led to the use of SIGN LANGUAGES but this was difficult as interpretation is not easy. The use of speech became the method. People then started talking to themselves with various languages they were conversant with. This method also came with its enormous complexities as the varying languages pose a big problem. The difficulty involved in learning languages to enable communications, led to the use of another method i.e. writing. This entails that the problems of having to be close to the listener before communication through speech is solved.

Thus, the difficulty of having to write down voluminous speeches by different people at different times meant that many copies could be produced for onward transfer to the places of need. Also, the stress of carry along voluminous copies of documents produced for record transfer led to the advent of the adoption of telecommunication, telephones, telegrams, fax etc. were used and this to a very great extent meant a very big technological break through in information technology.

However, in an attempt to have proper information processing, storage and retrieval led to the addition of the Computer system and this is what is today called INFORMATION TECHNOLOGY.

Badmus (2002) finally stated that based on the whole national and international community and the way organization are run highlights the fact that modern society is heavily dependent on the communication, processing and storage of information. It is claimed by some that we are moving towards "an information society" in which the majority of the labour force will be engaged in information processing and the use of information technology.

Power (1984) proved that the role of computer information technology is immense and as such cannot be over emphasized. Computers are special in that they automate many methods of information processing. Thus, it is the technology which supports the activities

involving the creation, storage, manipulation, communication of information together with their related methods, management and application, therefore, information technology may be seen as the broadly based technology to support information system.

Blumenthal (1969) noted that the wide spread use of Computer in the non-scientific area was characterized by their use as adjuncts to or more efficient replacement for the regular clerical process. Of course, from the early days of computing, there were specialist developments that are full of the computer's potentials.

Oliver *et al* (1992) explains that stores maintains records of purchase so that they can bill customer correctly and maintain a stock of goods. Library records the loan of a book together with the name and address of the borrower. Physicians maintain files containing medical information about individual patients. The names of registered voters are stored within a file for retrieval on voting day. Banks record each transaction by holders of check and saving account. He further explained that record fall into three categories discussed below. Master records contain data that are relatively permanent for example, the name; address, social security number, sex and birthday of a student would appear on his master record. A file of master records is a master file. Detail records contain data corresponding to a single transaction or event for example the name, the student ID number and grade in a particular course would appear on the detail record of a student. Summary records are those that summarise a number of detail records. For example one could have a summary class records giving the names, ID number and grade of all the students in a given class.

2.2 Method of Collecting Records

According to the project write-up of Fasasi (1999) the following methods are used in collecting records:

Interviews: This involved meeting officers in charge who work in the section of the school where student's files are kept.

Existing records: This involved going through the existing records, so as to see how the data is managed, the data formats and how the data is processed within the section.

Mode of operation: This involved studying the mode of operation of the existing system so as to be able to automate it to serve the same purpose efficiently. Interacting with students to know which information they considered personal to them and the staff of the school too, to know the advantages of computerising the students personal data.

2.3 Problem of Manual Students Record Keeping

1. Keeping students record at G.G.S.S. Bida in manual form is characterised by problems enumerated below:

- a. Retrieval of information takes a very long time
- b. The security of information or data is questionable
- c. Correctness of some information cannot be relied on because certain information cannot easily be removed, since some files are kept on the floor of the admin. block of the school.

2. Brightman (1992) stated that it has been recently estimated that in the U.S.A. there are about 200 million sheets of papers store in offices. These are being added to at the rate of one million per day. This shows that storing information on paper causes two problems.

- a. Where to store it all; the files have to be protected from fire and other destructive insects.
- b. It is costly and paper consuming.

3. It encourages bribery and corruption; files can be kept away by manager and demanding money from the owner if it is urgently needed.
4. There is slow response enquiry about students performance academically.
5. Files are easily misplaced due to laziness or carelessness of the admin. officer incharge and it takes longer time to recover information lost when files are misplaced.

2.4 Advantages of Computerization of Student Record

The importance of using Computer to keep records cannot be over emphasised. Honourable Minister of Education (1987) inaugurated a committee to prepare the National Policy on Computer Education, education on a pilot scale in selected secondary where the needed facilities can be provided.

According to Larry (1988) the following are the advantages of computerization of students' record.

1. **Speed:** The most obvious benefit of using Computer is speed. The Computer can perform calculations and data processing more quickly than alternative methods. Work that might take months or even years to be completed manually may be accomplished in hours or at most days by Computer. For example, some Computer can do hundreds of thousands or even million of arithmetic operations per second.
2. **Accuracy:** The computerized personnel information system with accurate data, will do the intended work with a very high degree of accuracy. The Computer does exactly what the program tells it to do. In addition, the Computer does not get bored or fatigue thus avoiding the errors human might make under the same circumstances.
3. **Reliability:** The Computer can work almost twenty four hours a day (with a little time out equipment everyday of the year), it still operate

reliably, modern electronic Computer perform at high level of reliability and equipment failure are very few.

4. **Retention:** The Computer can store and search massive files of data and programs, the content of the files does not fade or get lost, and it can be used any time.
5. **Economy:** The advantage of speed and accuracy in terms of cost of processing data or doing computation by Computers is considerably lower than the manual type.
6. **Wide Applicability:** A Computer can be used to solve a wide variety of problems that arise in student records. The boundaries of what the Computer can accomplish are limited only by the ability and imagination of its users.

According to Larry (1988) the following are the disadvantages of computerisation of school records:

2.5 Disadvantages of Computerization of Student Record

1. Installing the Computer requires an expensive task of system analysis and design. There is a scarcity of Computer professionals to do this.
2. Presentation of wrong record about a student: Computer produces exactly what is been fed into it, so if there are mistakes in the data fed into it, definitely, there would be error in that students record.
3. The need to obtain stand by facilities in the event of breakdown of any computerized systems. Manual systems, though slow, are vastly flexible. If the format of a report for example has to be changed, it can be readily done communicating the necessary instruction to the concerned staff whereas in a computerized system, this would have to go through all the stages of system analysis and design before it is effected.

CHAPTER THREE

SYSTEM DESIGN

3.1 Introduction

The proposed system would contain the following database to carry out its function and from which query shall be done.

The databases are:

- a. Set up database
- b. Register database
- c. Completed database

a. **Set up database:** The set up database would contain all the students personal data which the system would recognise and work with.

b. **Register database:** The register database would contain names of all the students personal data which exist in the set up database. It would contain the following field of information.

Date

File Number

Name

Class

c. **Completed Database:** The completed database would contain records of all the students admitted up to date. It would contain the following field of information.

Year of admission

Year of graduation

Name

File Number

3.2 Files Designing

This is one of the major aspect of the system design once data design is completed the next step is to make final organisation into two:

- a. Students admission files
- b. Students academics files

3.2.1 Students Admission or Registration File

This file will be only one for the whole school. It will be a permanent file. It will contain the following information; Students Name, Admission Number, Date of Birth, Age, Sex, Address, Parent Names, Parents Occupation and the students academic file, this would be one per class. It will serve as class register file as well as academic file, apart from student personal data, it will also contain summary of students academic record.

Data Structure for Student Admission File

F/No	DESCRIPTION	F/NAME	TYPE
1	ADMISSION	ADM	CHARACTER
2	NAME	NAME	CHARACTER
3	DATE OF BIRTH	DATE	DATE
4	AGE	AGE	CHARACTER
5	SEX	SEX	CHARACTER
6	PLACE OF BIRTH	P/BIRTH	CHARACTER
7	STATE	STATE	CHARACTER
8	RELIGION	REL	CHARACTER
9	FATHER'S NAME	F/NAME	CHARACTER
10	FATHER'S ADDRESS	F.ADD	CHARACTER

3.2.2 Data Structure for Student Academic File

F/No	DESCRIPTION	F/NAME	TYPE
1	NAME	NAME	CHARACTER
2	ADMISSION NUMBER	N. ADM	CHARACTER
3	DATE OF ADMISSION	D. ADM	DATE
4	SS1 PROMOTION RESULT	SS1	CHARACTER
5	SS2 PROMOTION RESULT	SS2	CHARACTER
6	SSCE RESULT	SSCE	CHARACTER
7	YEAR OF GRADUATION	G. YEAR	CHARACTER

3.3 Programming Language

The programmer selects visual basic which is one of the many version of Basic programming language because of the following reasons:

1. It has faster compiler
2. It allows database integration with wide variety of application.
3. It has additional internet capabilities

Steps in Developing Application

There are three primary steps involved in building a Visual Basic Application

1. Draw the user interface
2. Assign properties to controls
3. Attach code to control

Visual Basic Operates in three modes which are:

1. Design mode - used to build application
2. Run mode - used to run the application
3. Break mode - application halted and debugger is available

3.4 Materials to be Used

1. **Hardware:** The hardware of a Computer refers to the various mechanised, magnetic, electrical and electronic parts of the computer, the most powerful feature of the hardware component, the better the utilization of the design system. For the hardware requirement a Computer with the following specification will be ideal: COMPATIBLE (Comprising of intel/processor HARD-DISK, Floppy disk drive, VGA Adaptor, VGA Monitor, English keyboard (101 keys).

2. Software Requirement

The software of a Computer system is a collection of programs that are responsible for the controlling of the activities of the Computer. The software requirements that are essential for the operation of the proposed system are as follows:

- a. Disk operating system
- b. Database Management System Package

3. Peripheral Equipment Requirement

a. **Printer:-** A printer is essential in obtaining hard copies of reports, In view of this printing mechanism mentioned below will satisfy the requirements.

- A fast D: T Matrix printer
- UPS - Uninterrupted Power Supply

In order to avoid unnecessary interruption of power supply which may eventually lead to loss of information an uninterrupted power supply of about 500 VA will suit the purpose of this system.

Acquisition of backup equipment is recommended to this effect.

4. Personnel

A Computer personnel is needed to be able to operate the Computer, as the Computer is nothing but an electronic robot.

CHAPTER FOUR

SYSTEM IMPLEMENTATION

4.1 Introduction

System implementation is the coordination of the users and data processing department in getting the new system into operation. System implementation is aimed at putting theory into practice. The design states what should be done while the implementation does it. The implementation of this system is basically the programming and testing and the programme is well documented, because it helps in putting the system into operation and allows the user to operate the system correctly. Arrangement should be made to get the right kind of person to handle the programming and operations aspect of the computer. It is necessary that the user of the new system be trained before the new system is introduced.

4.2 Operation Manual for the new System

The source code of this program is made to run on MS Access 2003 and all its associated database files. The program and the associated files are stored in directory called Mrs. A. K. Gana. To run the program, the user should proceed as follows.

1. Install the program from the provided flash CD ROM, if the program is not already installed.
2. Click start.
3. Click all programme
4. Click on Microsoft visual basic 2005 express edition

5. After few seconds Fig1. will be displaced

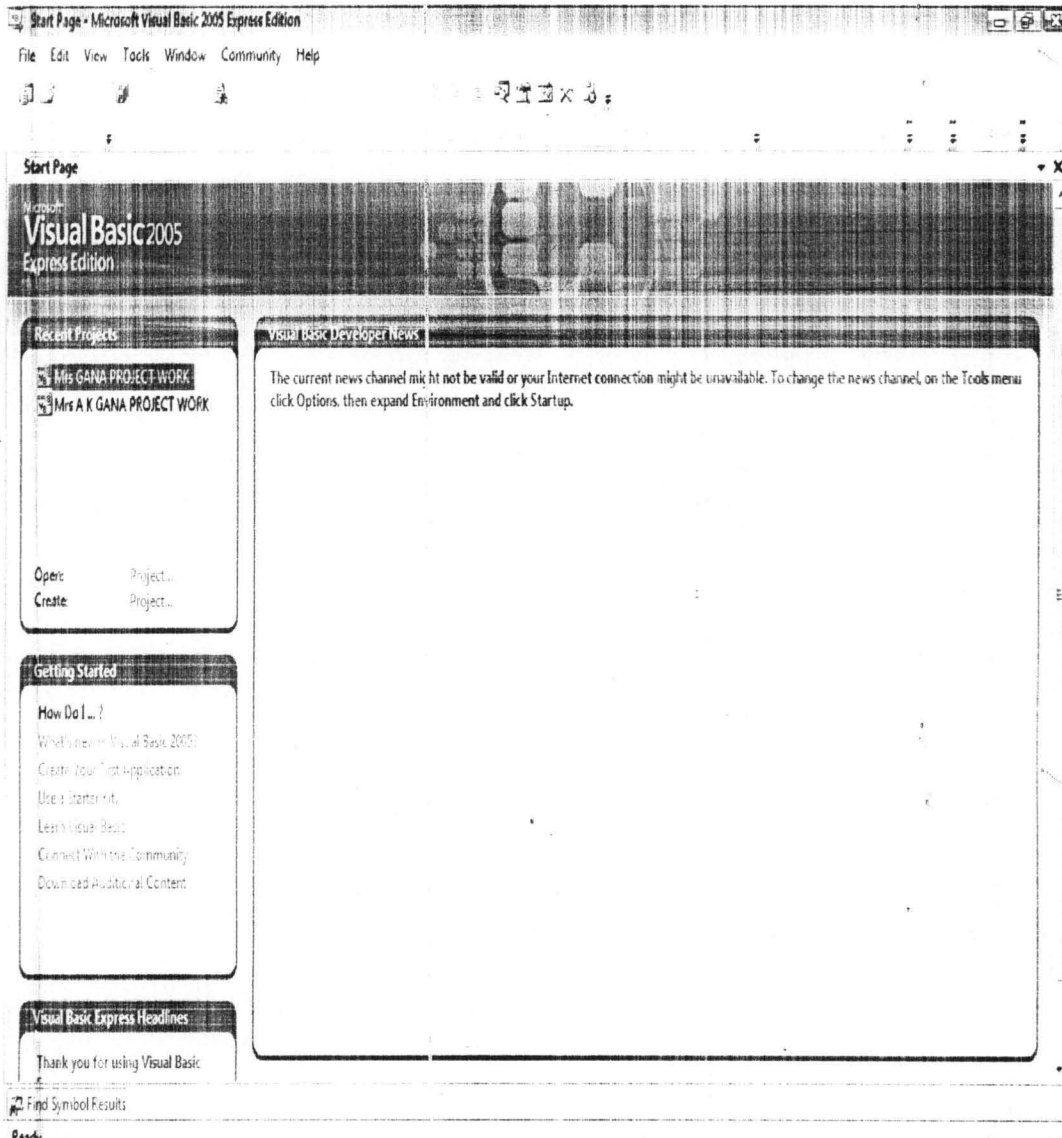


Fig 1

6. Go to file menu
7. Click on open project
8. Double click Mrs. Gana project work folder
9. Double click Mrs. Gana project work file
10. Press F5 from your key board or click start debugging from the standard tool bar

Fig 2 is displaced

Mrs GANA PROJECT WORK

1 cd7

WELCOME TO GOVERNMENT GIRLS SECONDARY SCHOOL BIDA STUDENTS RECORD

STUDENTS PERSONAL RECORDS

FIRST NAME: SURNAME: YISA REGISTRATION NUMBER: 4540 YEAR OF GRADUATION: 2004 AGE: 15

DATE OF BIRTH: Tuesday, December 05, 1995 PLACE OF BIRTH: DOKO STATE OF ORIGIN: NIGER P. RELIGION: CHRISTIANITY

SEX: FEMALE FATHER'S NAME: Mr ANDREW YISA

FATHER'S ADDRESS: Mr ANDREW YISA, ARMY DAY SECONDARY SCHOOL BIDA.

ACADEMIC RESULT PHASE ONE	ACADEMIC RESULT PHASE TWO	ACADEMIC RESULT PHASE THREE
ACCOUNT 1:	ACCOUNT 2:	ACCOUNT:
AGRICULTURAL SCIENCE 1: 56	AGRICULTURAL SCIENCE 2: 56	AGRICULTURAL SCIENCE: C4
BIOLOGY 1: 67	BIOLOGY 2: 84	BIOLOGY: C5
CHEMISTRY 1: 56	CHEMISTRY 2: 34	CHEMISTRY: F
COMMERCE 1:	COMMERCE 2:	COMMERCE:
COMPUTER SCIENCE 1:	COMPUTER SCIENCE 2:	COMPUTER SCIENCE:
ENGLISH 1: 43	ENGLISH 2: 67	ENGLISH: B2
GEOGRAPHY 1: 57	GEOGRAPHY 2: 65	GEOGRAPHY: A1
HAUSA 1:	HAUSA 2:	HAUSA:
MATHEMATICS 1: 74	MATHEMATICS 2: 34	MATHEMATICS: C4
PHYSICS 1: 45	PHYSICS 2: 67	PHYSICS: C5
RELIGION 1: 38	RELIGION 2:	RELIGION: C4

CLOSE

Fig 2

To check for any student's record click on current position box


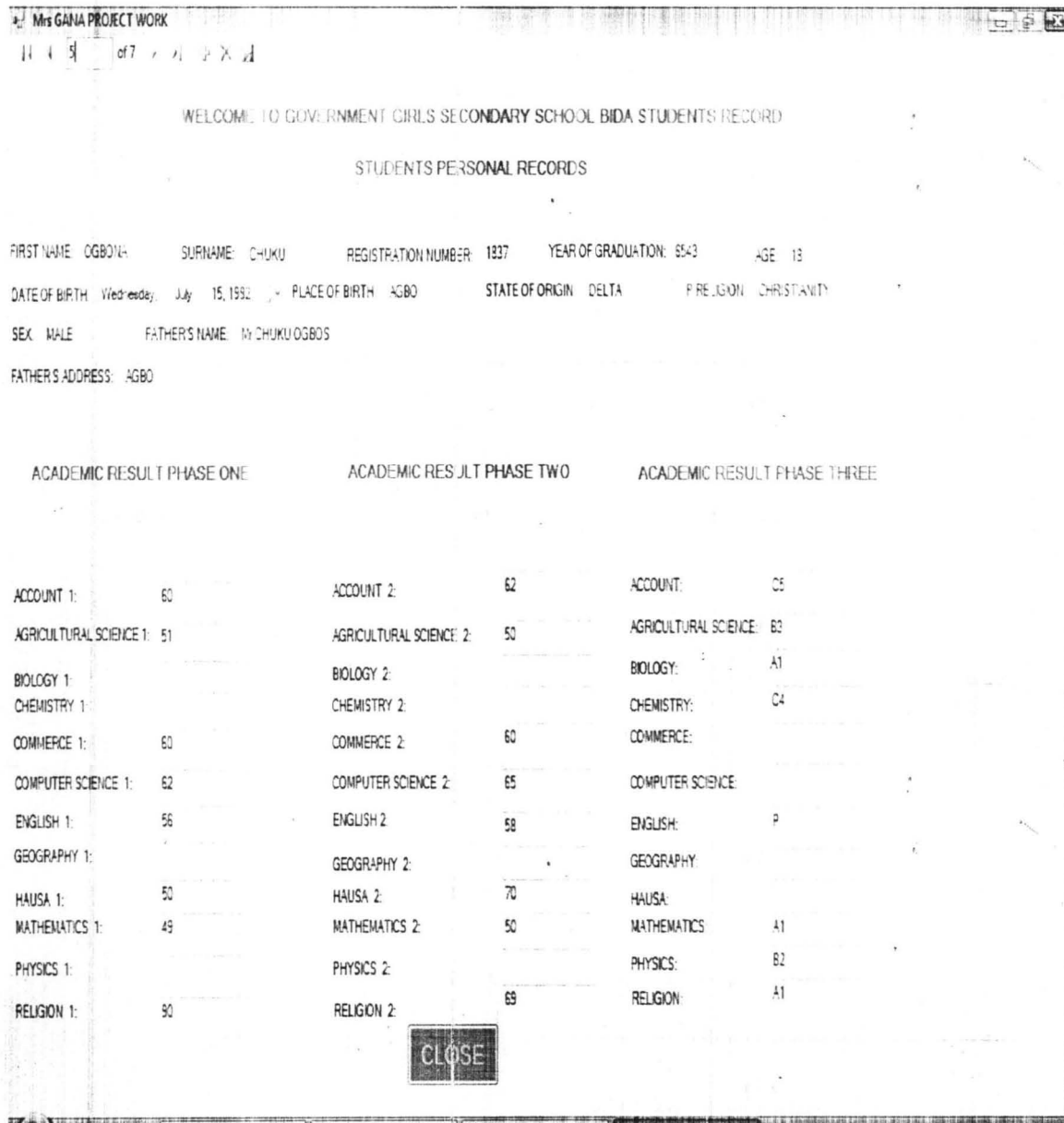
() and type the student's Registration number and press enter on the key board.

Fig 3 will be displaced.



Mrs GANA PROJECT WORK

WELCOME TO GOVERNMENT GIRLS SECONDARY SCHOOL BIDA STUDENTS RECORD

STUDENTS PERSONAL RECORDS

FIRST NAME: OGBONIA SURNAME: CHUKU REGISTRATION NUMBER: 1937 YEAR OF GRADUATION: 95-93 AGE: 13

DATE OF BIRTH: Wednesday, July 15, 1992 PLACE OF BIRTH: AGBO STATE OF ORIGIN: DELTA RELIGION: CHRISTIANITY

SEX: MALE FATHER'S NAME: M CHUKU OGBOS

FATHER'S ADDRESS: AGBO

ACADEMIC RESULT PHASE ONE	ACADEMIC RESULT PHASE TWO	ACADEMIC RESULT PHASE THREE
ACCOUNT 1: 60	ACCOUNT 2: 62	ACCOUNT: C5
AGRICULTURAL SCIENCE 1: 51	AGRICULTURAL SCIENCE 2: 50	AGRICULTURAL SCIENCE: B3
BIOLOGY 1:	BIOLOGY 2:	BIOLOGY: A1
CHEMISTRY 1:	CHEMISTRY 2:	CHEMISTRY: C4
COMMERCE 1: 60	COMMERCE 2: 60	COMMERCE:
COMPUTER SCIENCE 1: 62	COMPUTER SCIENCE 2: 65	COMPUTER SCIENCE:
ENGLISH 1: 56	ENGLISH 2: 58	ENGLISH: P
GEOGRAPHY 1:	GEOGRAPHY 2:	GEOGRAPHY:
HAUSA 1: 50	HAUSA 2: 70	HAUSA:
MATHEMATICS 1: 49	MATHEMATICS 2: 50	MATHEMATICS: A1
PHYSICS 1:	PHYSICS 2:	PHYSICS: B2
RELIGION 1: 90	RELIGION 2: 69	RELIGION: A1

CLOSE

Fig 3

To add new record click on add new (+)

Fig 4 will be displaced, then enter the students information

Ms GANA PROJECT WORK

11 4 8 of 8

WELCOME TO GOVERNMENT GIRLS SECONDARY SCHOOL BIDA STUDENTS RECORD

STUDENTS PERSONAL RECORDS

FIRST NAME: _____ SURNAME: _____ REGISTRATION NUMBER: _____ YEAR OF GRADUATION: _____ AGE: _____

DATE OF BIRTH: Wednesday, July 15, 1992 [] PLACE OF BIRTH: _____ STATE OF ORIGIN: _____ P. RELIGION: _____


SEX: _____ FATHER'S NAME: _____

FATHER'S ADDRESS: _____

ACADEMIC RESULT PHASE ONE	ACADEMIC RESULT PHASE TWO	ACADEMIC RESULT PHASE THREE
ACCOUNT 1: _____	ACCOUNT 2: _____	ACCOUNT: _____
AGRICULTURAL SCIENCE 1: _____	AGRICULTURAL SCIENCE 2: _____	AGRICULTURAL SCIENCE: _____
BIOLOGY 1: _____	BIOLOGY 2: _____	BIOLOGY: _____
CHEMISTRY 1: _____	CHEMISTRY 2: _____	CHEMISTRY: _____
COMMERCE 1: _____	COMMERCE 2: _____	COMMERCE: _____
COMPUTER SCIENCE 1: _____	COMPUTER SCIENCE 2: _____	COMPUTER SCIENCE: _____
ENGLISH 1: _____	ENGLISH 2: _____	ENGLISH: _____
GEOGRAPHY 1: _____	GEOGRAPHY 2: _____	GEOGRAPHY: _____
HAUSA 1: _____	HAUSA 2: _____	HAUSA: _____
MATHEMATICS 1: _____	MATHEMATICS 2: _____	MATHEMATICS: _____
PHYSICS 1: _____	PHYSICS 2: _____	PHYSICS: _____
RELIGION 1: _____	RELIGION 2: _____	RELIGION: _____

CLOSE

Fig 4

To save the information, press save data ()

To delete any student's data or record enter the student's Registration number and click delete (X) and the record will be deleted

To close the form, click on close button then Fig 5 will be displaced.

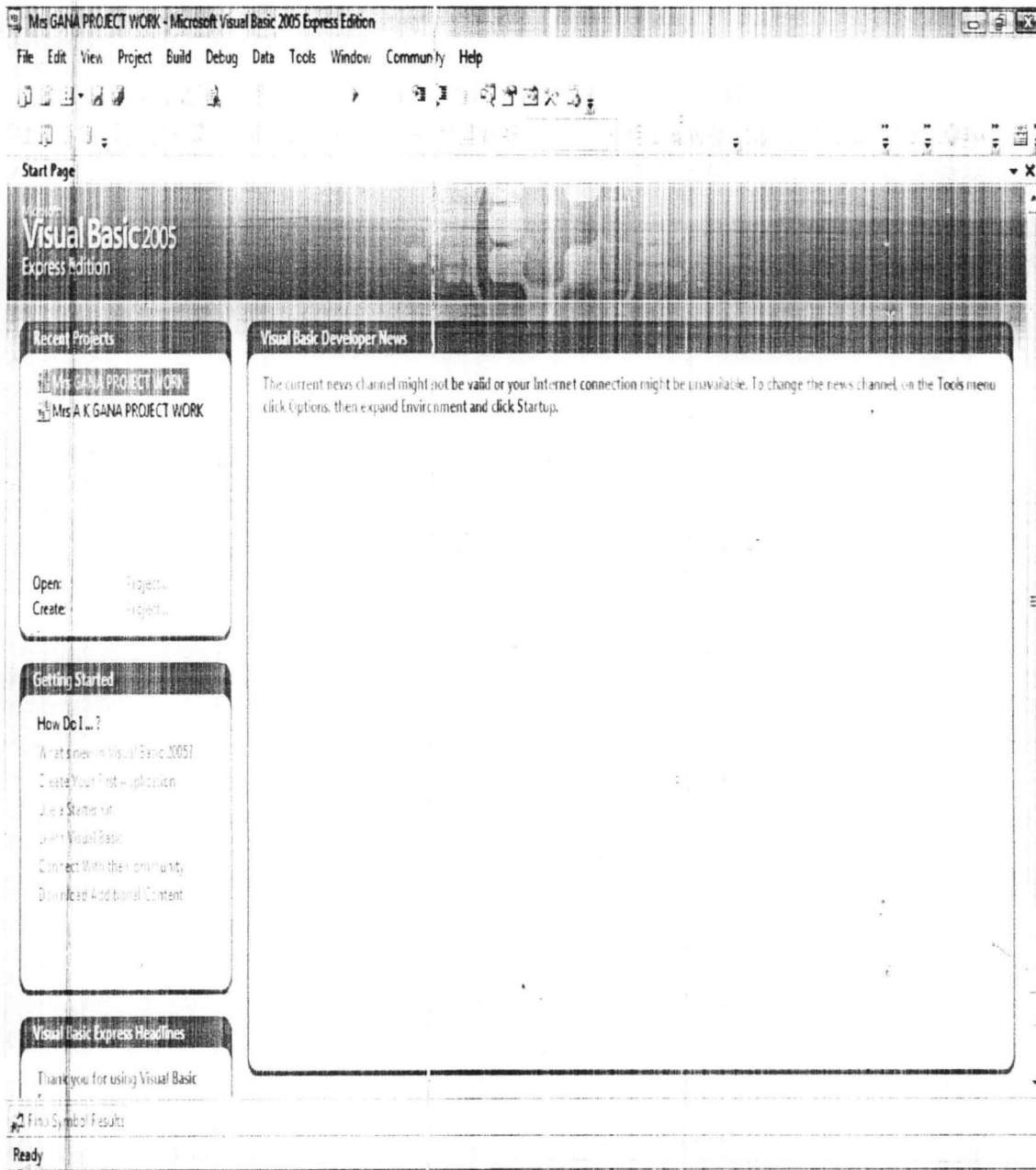


Fig 5

CHAPTER FIVE

SUMMARY, RECOMMENDATION AND CONCLUSION

5.1 Summary

The proposed computerized student record system is designed to involve the use of computer in establishment division. The mechanization of record keeping and retrieval system is not only time saving, it is economical in terms of space, it is also easier and faster for keeping and retrieval of students records.

In order to enable the staff in the establishment division carry out their job accurately, thoroughly, unbiased and to actually contribute to its future well being. The new automated system is designed to make the retrieving and addition of new students easier.

The program has also been coded to actually carry out some processing and the expected result has been achieved.

This has formed the basis of the project and obviously a report has been produced to that effect.

5.2 Recommendation

During the feasibility study of this project work at Government Girls' Secondary School Bida. It was observed that to keep and process students record was difficult and cumbersome, for this we therefore recommend that the school should use computer in order to boost the easy processing of students record.

In making the new system effective, the school should have a standby generator and Uninterrupted Power Supply (UPS) in order to have an easy flow of information without failure, so as to overcome (NEPA) power failure.

We also recommended that the centre should introduce Network to their system (Interconnected set of two or more computer).

5.3 Conclusion

This project has been able to analyzed, designed and completed how the computer system can be more beneficial to Government Girls' Secondary School Bida particularly in the area of keeping students record.

When this project is fully implemented by the institution, it will go along way in making record keeping easier and faster.

This does not means that human effort is no longer needed, computer is a slave to man, it can not do things on its own, unless instructed by human being.

Reference

- Aronu (1999); Information System Techniques with Internet Operation Kaduna - Nigeria.
- Abubakar U.Y. (2005): PGD Lecture Note on System Analysis and Design (Unpublished)
- Badmus (2002); Introduction to Computer System (Lecture Notes), FUT, Minna.
- Blumelhal (1969); Management Information System: A Frame Work for planning and control. U.S.A.
- Brightman R. W. *et al* (1992); Computer is an Information Age Delmar Publisher Inc.
- Fasasi (1999), An Automated Student Information System. A case study of Kaduna Polytechnic, Kaduna. Student project PGD project, FUT Minna.
- Larry Long (1988); Introduction to Computer and Information. 2nd Edition, U.S.A.
- Oliver E.C., Chapman R.J. and French C.S. (1992); Data Processing and Information Technology, Ninth Edition DP Publication.
- Power J. Michael (1984); Computer Information System Developments: Analysis and Design U.S.A.
- Shodipe (1994); Computerization of Staff Record. A Case Study of National Population Commission, Minna, Student Project PGD Project, FUT Minna.

APPENDIX

```
Global.System.ComponentModel.DesignerCategoryAttribute("code"), _
Global.System.ComponentModel.ToolboxItem(true), _

Global.System.Xml.Serialization.XmlSchemaProviderAttribute("GetTypedDataSetSch
ema"), _

Global.System.Xml.Serialization.XmlRootAttribute("Mrs_GANA_DATABASDataSet")
Global.System.ComponentModel.Design.HelpKeywordAttribute("vs.data.DataSet")>
_
Partial Public Class Mrs_GANA_DATABASDataSet
    Inherits Global.System.Data.DataSet
    Private tablePERSONAL_RECORD As PERSONAL_RECORDDataTable
        Private tableS_S_12 As S_S_12DataTable
        Private tableS_S_2 As S_S_2DataTable
        Private tableS_S_3 As S_S_3DataTable
        Private tableTB As TBDataTable
        Private relationPERSONAL_RECORDS_S_12 As
Global.System.Data.DataRelation
        Private relationPERSONAL_RECORDS_S_2 As
Global.System.Data.DataRelation
        Private relationPERSONAL_RECORDS_S_3 As
Global.System.Data.DataRelation
        Private relationPERSONAL_RECORDTB As Global.System.Data.DataRelation
        Private _schemaSerializationMode As
Global.System.Data.SchemaSerializationMode =
Global.System.Data.SchemaSerializationMode.IncludeSchema
        <Global.System.Diagnostics.DebuggerNonUserCodeAttribute()> _
    Public Sub New()
        MyBase.New
        Me.BeginInit
        Me.InitClass
        Dim schemaChangedHandler As
Global.System.ComponentModel.CollectionChangeEventHandler = AddressOf
Me.SchemaChanged
        AddHandler MyBase.Tables.CollectionChanged, schemaChangedHandler
        AddHandler MyBase.Relations.CollectionChanged, schemaChangedHandler
        Me.EndInit
    End Sub
```

```

Protected Sub New(ByVal info As
Global.System.Runtime.Serialization.SerializationInfo, ByVal context As
Global.System.Runtime.Serialization.StreamingContext)
    MyBase.New(info, context, false)
    If (Me.IsBinarySerialized(info, context) = true) Then
        Me.InitVars(false)
        Dim schemaChangedHandler1 As
Global.System.ComponentModel.CollectionChangeEventHandler = AddressOf
Me.SchemaChanged
        AddHandler Me.Tables.CollectionChanged, schemaChangedHandler1
        AddHandler Me.Relations.CollectionChanged, schemaChangedHandler1
        Return
    End If
    Dim strSchema As String = CType(info.GetValue("XmlSchema",
GetType(String)),String)
    If (Me.DetermineSchemaSerializationMode(info, context) =
Global.System.Data.SchemaSerializationMode.IncludeSchema) Then
        Dim ds As Global.System.Data.DataSet = New Global.System.Data.DataSet
        ds.ReadXmlSchema(New Global.System.Xml.XmlTextReader(New
Global.System.IO.StringReader(strSchema)))
        If (Not (ds.Tables("PERSONAL RECORD")) Is Nothing) Then
            MyBase.Tables.Add(New
PERSONAL_RECORDDataTable(ds.Tables("PERSONAL RECORD")))
        End If
        If (Not (ds.Tables("S S 12")) Is Nothing) Then
            MyBase.Tables.Add(New S_S_12DataTable(ds.Tables("S S 12")))
        End If
        If (Not (ds.Tables("S S 2")) Is Nothing) Then
            MyBase.Tables.Add(New S_S_2DataTable(ds.Tables("S S 2")))
        End If
        If (Not (ds.Tables("S S 3")) Is Nothing) Then
            MyBase.Tables.Add(New S_S_3DataTable(ds.Tables("S S 3")))
        End If
        If (Not (ds.Tables("TB")) Is Nothing) Then
            MyBase.Tables.Add(New TBDataTable(ds.Tables("TB")))
        End If
        Me.DataSetName = ds.DataSetName
        Me.Prefix = ds.Prefix
        Me.Namespace = ds.Namespace
        Me.Locale = ds.Locale
        Me.CaseSensitive = ds.CaseSensitive
        Me.EnforceConstraints = ds.EnforceConstraints
        Me.Merge(ds, false, Global.System.Data.MissingSchemaAction.Add)
        Me.InitVars
    End Sub

```

```

Else
    Me.ReadXmlSchema(New Global.System.Xml.XmlTextReader(New
Global.System.IO.StringReader(strSchema)))
End If
Me.GetSerializationData(info, context)
Dim schemaChangedHandler As
Global.System.ComponentModel.CollectionChangeEventHandler = AddressOf
Me.SchemaChanged
AddHandler MyBase.Tables.CollectionChanged, schemaChangedHandler
AddHandler Me.Relations.CollectionChanged, schemaChangedHandler
End Sub
Global.System.ComponentModel.DesignerSerializationVisibility(Global.System.Comp
onentModel.DesignerSerializationVisibility.Content)> _
Public ReadOnly Property PERSONAL_RECORD() As
PERSONAL_RECORDDataTable
Get
    Return Me.tablePERSONAL_RECORD
End Get
End Property
Global.System.ComponentModel.DesignerSerializationVisibility(Global.System.Comp
onentModel.DesignerSerializationVisibility.Content)> _
Public ReadOnly Property S_S_12() As S_S_12DataTable
Get
    Return Me.tableS_S_12
End Get
End Property
Public ReadOnly Property S_S_2() As S_S_2DataTable
Get
    Return Me.tableS_S_2
End Get
End Property
Public ReadOnly Property S_S_3() As S_S_3DataTable
Get
    Return Me.tableS_S_3
End Get
End Property

Public ReadOnly Property TB() As TBDataTable
Get
    Return Me.tableTB
End Get
End Property
Public Overrides Property SchemaSerializationMode() As
Global.System.Data.SchemaSerializationMode

```

```

Get
    Return Me._schemaSerializationMode
End Get
Set
    Me._schemaSerializationMode = value
End Set
End Property
Public Shadows ReadOnly Property Tables() As
Global.System.Data.DataTableCollection
    Get
        Return MyBase.Tables
    End Get
End Property
Public Shadows ReadOnly Property Relations() As
Global.System.Data.DataRelationCollection
    Get
        Return MyBase.Relations
    End Get
End Property
Protected Overrides Sub InitializeDerivedDataSet()
    Me.BeginInit
    Me.InitClass
    Me.EndInit
End Sub
Public Overrides Function Clone() As Global.System.Data.DataSet
    Dim cln As Mrs_GANA_DATABASEDataSet =
CType(MyBase.Clone,Mrs_GANA_DATABASEDataSet)
    cln.InitVars
    cln.SchemaSerializationMode = Me.SchemaSerializationMode
    Return cln
End Function
    Protected Overrides Function ShouldSerializeTables() As Boolean
    Return false
End Function
    Protected Overrides Function ShouldSerializeRelations() As Boolean
    Return false
End Function
    If (Me.DetermineSchemaSerializationMode(reader) =
Global.System.Data.SchemaSerializationMode.IncludeSchema) Then
        Me.Reset
        Dim ds As Global.System.Data.DataSet = New Global.System.Data.DataSet
        ds.ReadXml(reader)
        If (Not (ds.Tables("PERSONAL RECORD")) Is Nothing) Then

```

```

        MyBase.Tables.Add(New
PERSONAL_RECORDDataTable(ds.Tables("PERSONAL RECORD")))
    End If
    If (Not (ds.Tables("S S 12")) Is Nothing) Then
        MyBase.Tables.Add(New S_S_12DataTable(ds.Tables("S S 12")))
    End If
    If (Not (ds.Tables("S S 2")) Is Nothing) Then
        MyBase.Tables.Add(New S_S_2DataTable(ds.Tables("S S 2")))
    End If
    If (Not (ds.Tables("S S 3")) Is Nothing) Then
        MyBase.Tables.Add(New S_S_3DataTable(ds.Tables("S S 3")))
    End If
    If (Not (ds.Tables("TB")) Is Nothing) Then
        MyBase.Tables.Add(New TBDataTable(ds.Tables("TB")))
    End If
    Me.DataSetName = ds.DataSetName
    Me.Prefix = ds.Prefix
    Me.Namespace = ds.Namespace
    Me.Locale = ds.Locale
    Me.CaseSensitive = ds.CaseSensitive
    Me.EnforceConstraints = ds.EnforceConstraints
    Me.Merge(ds, false, Global.System.Data.MissingSchemaAction.Add)
    Me.InitVars
Else
    Me.ReadXml(reader)
    Me.InitVars
End If
End Sub
Protected Overrides Function GetSchemaSerializable() As
Global.System.Xml.Schema.XmlSchema
    Dim stream As Global.System.IO.MemoryStream = New
Global.System.IO.MemoryStream
    Me.WriteXmlSchema(New Global.System.Xml.XmlTextWriter(stream, Nothing))
    stream.Position = 0
    Return Global.System.Xml.Schema.XmlSchema.Read(New
Global.System.Xml.XmlTextReader(stream), Nothing)
End Function
Friend Overloads Sub InitVars()
    Me.InitVars(true)
End Sub
Friend Overloads Sub InitVars(ByVal initTable As Boolean)
    Me.tablePERSONAL_RECORD = CType(MyBase.Tables("PERSONAL
RECORD"),PERSONAL_RECORDDataTable)
    If (initTable = true) Then

```

```

    If (Not (Me.tablePERSONAL_RECORD) Is Nothing) Then
        Me.tablePERSONAL_RECORD.InitVars
    End If
End If
Me.tableS_S_12 = CType(MyBase.Tables("S S 12"),S_S_12DataTable)
If (initTable = true) Then
    If (Not (Me.tableS_S_12) Is Nothing) Then
        Me.tableS_S_12.InitVars
    End If
End If
Me.tableS_S_2 = CType(MyBase.Tables("S S 2"),S_S_2DataTable)
If (initTable = true) Then
    If (Not (Me.tableS_S_2) Is Nothing) Then
        Me.tableS_S_2.InitVars
    End If
End If
Me.tableS_S_3 = CType(MyBase.Tables("S S 3"),S_S_3DataTable)
If (initTable = true) Then
    If (Not (Me.tableS_S_3) Is Nothing) Then
        Me.tableS_S_3.InitVars
    End If
End If
Me.tableTB = CType(MyBase.Tables("TB"),TBDataTable)
If (initTable = true) Then
    If (Not (Me.tableTB) Is Nothing) Then
        Me.tableTB.InitVars
    End If
End If
Me.relationPERSONAL_RECORDS_S_12 = Me.Relations("PERSONAL
RECORDS S 12")
Me.relationPERSONAL_RECORDS_S_2 = Me.Relations("PERSONAL
RECORDS S 2")
Me.relationPERSONAL_RECORDS_S_3 = Me.Relations("PERSONAL
RECORDS S 3")
Me.relationPERSONAL_RECORDDTB = Me.Relations("PERSONAL
RECORDDTB")
End Sub
Private Sub InitClass()
    Me.DataSetName = "Mrs_GANA_DATABASEDataSet"
    Me.Prefix = ""
    Me.Namespace = "http://tempuri.org/Mrs_GANA_DATABASEDataSet.xsd"
    Me.EnforceConstraints = true
    Me.SchemaSerializationMode =
Global.System.Data.SchemaSerializationMode.IncludeSchema

```



```

Me.tablePERSONAL_RECORD = New PERSONAL_RECORDDataTable
MyBase.Tables.Add(Me.tablePERSONAL_RECORD)
Me.tableS_S_12 = New S_S_12DataTable
MyBase.Tables.Add(Me.tableS_S_12)
Me.tableS_S_2 = New S_S_2DataTable
MyBase.Tables.Add(Me.tableS_S_2)
Me.tableS_S_3 = New S_S_3DataTable
MyBase.Tables.Add(Me.tableS_S_3)
Me.tableTB = New TBDataTable
MyBase.Tables.Add(Me.tableTB)
Me.relationPERSONAL_RECORDS_S_12 = New
Global.System.Data.DataRelation("PERSONAL RECORDS S 12", New
Global.System.Data.DataColumn()
{Me.tablePERSONAL_RECORD.REGISTRATION_NUMBERColumn}, New
Global.System.Data.DataColumn()
{Me.tableS_S_12.REGISTRATION_NUMBERColumn}, false)
    Me.Relations.Add(Me.relationPERSONAL_RECORDS_S_12)
    Me.relationPERSONAL_RECORDS_S_2 = New
Global.System.Data.DataRelation("PERSONAL RECORDS S 2", New
Global.System.Data.DataColumn()
{Me.tablePERSONAL_RECORD.REGISTRATION_NUMBERColumn}, New
Global.System.Data.DataColumn()
{Me.tableS_S_2.REGISTRATION_NUMBERColumn}, false)
    Me.Relations.Add(Me.relationPERSONAL_RECORDS_S_2)
    Me.relationPERSONAL_RECORDS_S_3 = New
Global.System.Data.DataRelation("PERSONAL RECORDS S 3", New
Global.System.Data.DataColumn()
{Me.tablePERSONAL_RECORD.REGISTRATION_NUMBERColumn}, New
Global.System.Data.DataColumn()
{Me.tableS_S_3.REGISTRATION_NUMBERColumn}, false)
    Me.Relations.Add(Me.relationPERSONAL_RECORDS_S_3)
    Me.relationPERSONAL_RECORDTB = New
Global.System.Data.DataRelation("PERSONAL RECORDTB", New
Global.System.Data.DataColumn()
{Me.tablePERSONAL_RECORD.REGISTRATION_NUMBERColumn}, New
Global.System.Data.DataColumn() {Me.tableTB.IDColumn}, false)
    Me.Relations.Add(Me.relationPERSONAL_RECORDTB)
End Sub
Private Function ShouldSerializePERSONAL_RECORD() As Boolean
    Return false
End Function
Private Function ShouldSerializeS_S_12() As Boolean
    Return false
End Function

```

```

Private Function ShouldSerializeS_S_2() As Boolean
    Return false
End Function
Private Function ShouldSerializeS_S_3() As Boolean
    Return false
End Function
Private Function ShouldSerializeTB() As Boolean
    Return false
End Function
Private Sub SchemaChanged(ByVal sender As Object, ByVal e As
Global.System.ComponentModel.CollectionChangeEventArgs)
    If (e.Action =
Global.System.ComponentModel.CollectionChangeAction.Remove) Then
        Me.InitVars
    End If
End Sub
Public Shared Function GetTypedDataSetSchema(ByVal xs As
Global.System.Xml.Schema.XmlSchemaSet) As
Global.System.Xml.Schema.XmlSchemaComplexType
    Dim ds As Mrs_GANA_DATABASEDataSet = New
Mrs_GANA_DATABASEDataSet
    Dim type As Global.System.Xml.Schema.XmlSchemaComplexType = New
Global.System.Xml.Schema.XmlSchemaComplexType
    Dim sequence As Global.System.Xml.Schema.XmlSchemaSequence = New
Global.System.Xml.Schema.XmlSchemaSequence
    Dim any As Global.System.Xml.Schema.XmlSchemaAny = New
Global.System.Xml.Schema.XmlSchemaAny
    any.Namespace = ds.Namespace
    sequence.Items.Add(any)
    type.Particle = sequence
    Dim dsSchema As Global.System.Xml.Schema.XmlSchema =
ds.GetSchemaSerializable
    If xs.Contains(dsSchema.TargetNamespace) Then
        Dim s1 As Global.System.IO.MemoryStream = New
Global.System.IO.MemoryStream
        Dim s2 As Global.System.IO.MemoryStream = New
Global.System.IO.MemoryStream
        Try
            Dim schema As Global.System.Xml.Schema.XmlSchema = Nothing
            dsSchema.Write(s1)
            Dim schemas As Global.System.Collections.IEnumerator =
xs.Schemas(dsSchema.TargetNamespace).GetEnumerator
            Do While schemas.MoveNext

```

```

    schema =
CType(schemas.Current,Global.System.Xml.Schema.XmlSchema)
    s2.SetLength(0)
    schema.Write(s2)
    If (s1.Length = s2.Length) Then
        s1.Position = 0
        s2.Position = 0
        Do While ((s1.Position < s1.Length) _
            AndAlso (s1.ReadByte = s2.ReadByte))

            Loop
            If (s1.Position = s1.Length) Then
                Return type
            End If
        End If
    Loop
Finally
    If (Not (s1) Is Nothing) Then
        s1.Close
    End If
    If (Not (s2) Is Nothing) Then
        s2.Close
    End If
End Try
End If
xs.Add(dsSchema)
Return type
End Function

```

```

End Set
End Property
Public Property SURNAME() As String
Get
Try
Return CType(Me.tablePERSONAL_RECORD.SURNAMEColumn),String)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column
'SURNAME' in table 'PERSONAL RECORD' is DBNull.", e)
End Try
End Get
Set
Me(Me.tablePERSONAL_RECORD.SURNAMEColumn) = value
End Set
End Property
Public Property REGISTRATION_NUMBER() As String
Get
Return
CType(Me(Me.tablePERSONAL_RECORD.REGISTRATION_NUMBERColumn),String)
End Get
Set
Me(Me.tablePERSONAL_RECORD.REGISTRATION_NUMBERColumn) = value
End Set
End Property
Public Property YEAR_OF_GRADUATION() As Integer
Get
Try
Return
CType(Me(Me.tablePERSONAL_RECORD.YEAR_OF_GRADUATIONColumn),Integer)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column 'YEAR OF
GRADUATION' in table 'PERSONAL RECORD' is DBNull.", e)
End Try
End Get
Set
Me(Me.tablePERSONAL_RECORD.YEAR_OF_GRADUATIONColumn) = value
End Set
End Property
Public Property AGE() As Integer
Get
Try
Return CType(Me(Me.tablePERSONAL_RECORD.AGEColumn),Integer)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column 'AGE' in
table 'PERSONAL RECORD' is DBNull.", e)
End Try
End Get
Set
Me(Me.tablePERSONAL_RECORD.AGEColumn) = value
End Set
End Property
Public Property DATE_OF_BIRTH() As Date
Get
Try
Return CType(Me(Me.tablePERSONAL_RECORD.DATE_OF_BIRTHColumn),Date)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column 'DATE OF
BIRTH' in table 'PERSONAL RECORD' is DBNull.", e)
End Try
End Get

```

```

Set
    Me(Me.tablePERSONAL_RECORD.DATE_OF_BIRTHColumn) = value
End Set
End Property
Public Property PLACE_OF_BIRTH() As String
Get
    Try
        Return CType(Me(Me.tablePERSONAL_RECORD.PLACE_OF_BIRTHColumn),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column 'PLACE
OF BIRTH' in table 'PERSONAL_RECORD' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tablePERSONAL_RECORD.PLACE_OF_BIRTHColumn) = value
End Set
End Property
Public Property STATE_OF_ORIGIN() As String
Get
    Try
        Return CType(Me(Me.tablePERSONAL_RECORD.STATE_OF_ORIGINColumn),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column 'STATE
OF ORIGIN' in table 'PERSONAL_RECORD' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tablePERSONAL_RECORD.STATE_OF_ORIGINColumn) = value
End Set
End Property
Public Property P_RELIGION() As String
Get
    Try
        Return CType(Me(Me.tablePERSONAL_RECORD.P_RELIGIONColumn),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column 'P
RELIGION' in table 'PERSONAL_RECORD' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tablePERSONAL_RECORD.P_RELIGIONColumn) = value
End Set
End Property
Public Property SEX() As String
Get
    Try
        Return CType(Me(Me.tablePERSONAL_RECORD.SEXColumn),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column 'SEX' in
table 'PERSONAL_RECORD' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tablePERSONAL_RECORD.SEXColumn) = value
End Set
End Property
Public Property FATHER_S_NAME() As String
Get
    Try
        Return CType(Me(Me.tablePERSONAL_RECORD.FATHER_S_NAMEColumn),String)

```

```

        Catch e As Global.System.InvalidCastException
            Throw New Global.System.Data.StrongTypingException("The value for column
'FATHER'S NAME' in table 'PERSONAL_RECORD' is DBNull.", e)
        End Try
    End Get
    Set
        Me(Me.tablePERSONAL_RECORD.FATHER_S_NAMEColumn) = value
    End Set
End Property
Public Property FATHER_S_ADDRESS() As String
Get
    Try
        Return
CType(Me(Me.tablePERSONAL_RECORD.FATHER_S_ADDRESSColumn),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column
'FATHER'S ADDRESS' in table 'PERSONAL_RECORD' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tablePERSONAL_RECORD.FATHER_S_ADDRESSColumn) = value
End Set
End Property
Public Function IsFIRST_NAMENull() As Boolean
Return Me.IsNull(Me.tablePERSONAL_RECORD.FIRST_NAMEColumn)
End Function
Public Sub SetFIRST_NAMENull()
Me(Me.tablePERSONAL_RECORD.FIRST_NAMEColumn) = Global.System.Convert.DBNull
End Sub
Public Function IsSURNAMENull() As Boolean
Return Me.IsNull(Me.tablePERSONAL_RECORD.SURNAMEColumn)
End Function
Public Sub SetSURNAMENull()
Me(Me.tablePERSONAL_RECORD.SURNAMEColumn) = Global.System.Convert.DBNull
End Sub
Public Function IsYEAR_OF_GRADUATIONNull() As Boolean
Return Me.IsNull(Me.tablePERSONAL_RECORD.YEAR_OF_GRADUATIONColumn)
End Function
Public Sub SetYEAR_OF_GRADUATIONNull()
Me(Me.tablePERSONAL_RECORD.YEAR_OF_GRADUATIONColumn) =
Global.System.Convert.DBNull
End Sub
Public Function IsAGENull() As Boolean
Return Me.IsNull(Me.tablePERSONAL_RECORD.AGEColumn)
End Function
Public Sub SetAGENull()
Me(Me.tablePERSONAL_RECORD.AGEColumn) = Global.System.Convert.DBNull
End Sub
Public Function IsDATE_OF_BIRTHNull() As Boolean
Return Me.IsNull(Me.tablePERSONAL_RECORD.DATE_OF_BIRTHColumn)
End Function
Public Sub SetDATE_OF_BIRTHNull()
Me(Me.tablePERSONAL_RECORD.DATE_OF_BIRTHColumn) =
Global.System.Convert.DBNull
End Sub
Public Function IsPLACE_OF_BIRTHNull() As Boolean
Return Me.IsNull(Me.tablePERSONAL_RECORD.PLACE_OF_BIRTHColumn)
End Function
Public Sub SetPLACE_OF_BIRTHNull()

```

```

        Me(Me.tablePERSONAL_RECORD.PLACE_OF_BIRTHColumn) =
Global.System.Convert.DBNull
    End Sub
    Public Function IsSTATE_OF_ORIGINNull() As Boolean
    Return Me.IsNull(Me.tablePERSONAL_RECORD.STATE_OF_ORIGINColumn)
    End Function
    Public Sub SetSTATE_OF_ORIGINNull()
    Me(Me.tablePERSONAL_RECORD.STATE_OF_ORIGINColumn) =
Global.System.Convert.DBNull
    End Sub
    Public Function IsP_RELIGIONNull() As Boolean
    Return Me.IsNull(Me.tablePERSONAL_RECORD.P_RELIGIONColumn)
    End Function
    Public Sub SetP_RELIGIONNull()
    Me(Me.tablePERSONAL_RECORD.P_RELIGIONColumn) = Global.System.Convert.DBNull
    End Sub
    Public Function IsSEXNull() As Boolean
    Return Me.IsNull(Me.tablePERSONAL_RECORD.SEXColumn)
    End Function
    Public Sub SetSEXNull()
    Me(Me.tablePERSONAL_RECORD.SEXColumn) = Global.System.Convert.DBNull
    End Sub
    Public Function Is_FATHER_S_NAMENull() As Boolean
    Return Me.IsNull(Me.tablePERSONAL_RECORD._FATHER_S_NAMEColumn)
    End Function
    Public Sub Set_FATHER_S_NAMENull()
    Me(Me.tablePERSONAL_RECORD._FATHER_S_NAMEColumn) =
Global.System.Convert.DBNull
    End Sub
    Public Function Is_FATHER_S_ADDRESSNull() As Boolean
    Return Me.IsNull(Me.tablePERSONAL_RECORD._FATHER_S_ADDRESSColumn)
    End Function
    Public Sub Set_FATHER_S_ADDRESSNull()
    Me(Me.tablePERSONAL_RECORD._FATHER_S_ADDRESSColumn) =
Global.System.Convert.DBNull
    End Sub
    Public Function GetS_S_12Rows() As S_S_12Row()
    If (Me.Table.ChildRelations("PERSONAL RECORDS S 12") Is Nothing) Then
    Return New S_S_12Row(-1) {}
    Else
    Return CType(MyBase.GetChildRows(Me.Table.ChildRelations("PERSONAL RECORDS S
12")),S_S_12Row())
    End If
    End Function
    Public Function GetS_S_2Rows() As S_S_2Row()
    If (Me.Table.ChildRelations("PERSONAL RECORDS S 2") Is Nothing) Then
    Return New S_S_2Row(-1) {}
    Else
    Return CType(MyBase.GetChildRows(Me.Table.ChildRelations("PERSONAL RECORDS S
2")),S_S_2Row())
    End If
    End Function
    Public Function GetS_S_3Rows() As S_S_3Row()
    If (Me.Table.ChildRelations("PERSONAL RECORDS S 3") Is Nothing) Then
    Return New S_S_3Row(-1) {}
    Else
    Return CType(MyBase.GetChildRows(Me.Table.ChildRelations("PERSONAL RECORDS S
3")),S_S_3Row())
    End If
    End Function

```

```

End Class
Partial Public Class S_S_12Row
Inherits Global.System.Data.DataRow

Private tableS_S_12 As S_S_12DataTable
    Friend Sub New(ByVal rb As Global.System.Data.DataRowBuilder)
        MyBase.New(rb)
        Me.tableS_S_12 = CType(Me.Table,S_S_12DataTable)
    End Sub
    Public Property REGISTRATION_NUMBER() As String
    Get
        Return CType(Me(Me.tableS_S_12.REGISTRATION_NUMBERColumn),String)
    End Get
    Set
        Me(Me.tableS_S_12.REGISTRATION_NUMBERColumn) = value
    End Set
End Property
    Public Property ACCOUNT__1() As String
    Get
        Try
            Return CType(Me(Me.tableS_S_12.ACCOUNT__1Column),String)
        Catch e As Global.System.InvalidCastException
            Throw New Global.System.Data.StrongTypingException("The value for column
'ACCOUNT__1' in table 'S S 12' is DBNull.", e)
        End Try
    End Get
    Set
        Me(Me.tableS_S_12.ACCOUNT__1Column) = value
    End Set
End Property
    Public Property REMARK_FOR_ACCOUNT() As String
    Get
        Try
            Return CType(Me(Me.tableS_S_12.REMARK_FOR_ACCOUNTColumn),String)
        Catch e As Global.System.InvalidCastException
            Throw New Global.System.Data.StrongTypingException("The value for column 'REMARK
FOR ACCOUNT' in table 'S S 12' is DBNull.", e)
        End Try
    End Get
    Set
        Me(Me.tableS_S_12.REMARK_FOR_ACCOUNTColumn) = value
    End Set
End Property
    Public Property AGRICULTURAL_SCIENCE_1() As String
    Get
        Try
            Return CType(Me(Me.tableS_S_12.AGRICULTURAL_SCIENCE_1Column),String)
        Catch e As Global.System.InvalidCastException
            Throw New Global.System.Data.StrongTypingException("The value for column
'AGRICULTURAL_SCIENCE_1' in table 'S S 12' is DBNull.", e)
        End Try
    End Get
    Set
        Me(Me.tableS_S_12.AGRICULTURAL_SCIENCE_1Column) = value
    End Set
End Property
    Public Property REMARK_FOR_AGRICULTURAL_SCIENCE() As String
    Get
        Try

```



```

Return
CType(Me(Me.tableS_S_12.REMARK_FOR_AGRICULTURAL_SCIENCEColumn),String)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column 'REMARK
FOR AGRICULTURAL_SCIENCE' in table 'S S 12' is DBNull."&
"l.", e)
End Try
End Get
Set
Me(Me.tableS_S_12.REMARK_FOR_AGRICULTURAL_SCIENCEColumn) = value
End Set
End Property
Public Property BIOLOGY__1() As String
Get
Try
Return CType(Me(Me.tableS_S_12.BIOLOGY__1Column),String)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column
'BIOLOGY__1' in table 'S S 12' is DBNull.", e)
End Try
End Get
Set
Me(Me.tableS_S_12.BIOLOGY__1Column) = value
End Set
End Property
Public Property REMARK_FOR_BIOLOGY() As String
Get
Try
Return CType(Me(Me.tableS_S_12.REMARK_FOR_BIOLOGYColumn),String)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column 'REMARK
FOR BIOLOGY' in table 'S S 12' is DBNull.", e)
End Try
End Get
Set
Me(Me.tableS_S_12.REMARK_FOR_BIOLOGYColumn) = value
End Set
End Property
Public Property CHEMISTRY__1() As String
Get
Try
Return CType(Me(Me.tableS_S_12.CHEMISTRY__1Column),String)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column
'CHEMISTRY__1' in table 'S S 12' is DBNull.", e)
End Try
End Get
Set
Me(Me.tableS_S_12.CHEMISTRY__1Column) = value
End Set
End Property
Public Property REMARK_FOR_CHEMISTRY() As String
Get
Try
Return CType(Me(Me.tableS_S_12.REMARK_FOR_CHEMISTRYColumn),String)
Catch e As Global.System.InvalidCastException
Throw New Global.System.Data.StrongTypingException("The value for column 'REMARK
FOR CHEMISTRY' in table 'S S 12' is DBNull.", e)
End Try
End Get

```

```

Set
    Me(Me.tableS_S_12.REMARK_FOR_CHEMISTRYColumn) = value
End Set
End Property
Public Property COMMERCE__1() As String
Get
    Try
        Return CType(Me(Me.tableS_S_12.COMMERCE__1Column),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column
'COMMERCE__1' in table 'S S 12' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tableS_S_12.COMMERCE__1Column) = value
End Set
End Property    Public Property REMARK_FOR_AGRICULTURAL_SCIENCE() As String
Get
    Try
        Return
CType(Me(Me.tableS_S_2.REMARK_FOR_AGRICULTURAL_SCIENCEColumn),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column 'REMARK
FOR AGRICULTURAL_SCIENCE' in table 'S S 2' is DBNull"&
        ". ", e)
    End Try
End Get
Set
    Me(Me.tableS_S_2.REMARK_FOR_AGRICULTURAL_SCIENCEColumn) = value
End Set
End Property
Public Property BIOLOGY__2() As String
Get
    Try
        Return CType(Me(Me.tableS_S_2.BIOLOGY__2Column),String)
    Catch e As Global.System.InvalidCastException
        Throw New Global.System.Data.StrongTypingException("The value for column
'BIOLOGY__2' in table 'S S 2' is DBNull.", e)
    End Try
End Get
Set
    Me(Me.tableS_S_2.BIOLOGY__2Column) = value
End Set
End Property
End Function

```