Chapter-1

Introduction

"COLLEGE MANAGEMENT SYSTEM" is provides the detailed structure of The college campus and its departments. College Management System Synchronizes the working of all the departments. It looks on all aspects of a college, its student's, facilities, departments, marks and other activities. College Management System is the easiest way to manage all functionalities of a college. The entire knowledge regarding the college, teachers, students, faculties etc. Here we gain the recent knowledge regarding the students and teachers. During the process of project progression, there is no chance of data mishandling. The system **"COLLEGE MANAGEMENT SYSTEM"** can be used to manage the data of all type of educational institutes.

1.1 OVERVIEW OF PROJECT

The COLLEGE MANAGEMENT SYSTEM can be used to manage the data of all type of educational institutes. It will support both stand alone and also networking environment. The system uses .NET Technology. The main form involved in this system are:

1. Login
2. Form
3. **Login: -** It is used to check whether the user is an authorized person to use the system or not. The user should give the correct user name and password.

Form: - The different types of forms are:

1. Teacher information form
2. Student information form
3. Teacher salary details form

Chapter-2

SYSTEM REQUIREMENT SPECIFICATION

2.1 PRELIMINARY INVESTIGATION

2.1.1 OBJECTIVE

The software aims at automating the existing manual systems and providing secure, cheaper and quicker communication within an organization. As this is a well designed and easy-to-use communication system will certainly give the organization an edge over the other organizations with outdated and manual communication system, with manpower doing the bulk of data transfer in the form of files and paper documents.

This software turns out to be a very suitable one for any type of organization which requires an efficient and secure communication among their teacher. It provides centralized information transfer system with any number of client systems, to cater to any number of users.

.

The Main objectives of this project are:

- Reduction of paper work
- Automation of existing manual information systems.
- Reduction of manual processing

- Increase in processing and transfer speeds of information over the network.
- Decrease in processing time
- Fast retrieval of all type of information
- Good efficiency and response time
- More consistent data handling
- A user-friendly system which do not require any special training or expertise of computer

Chapter-3

Methodology

3.1 INTRODUCTION

Methodology is a systematic description of the sequence of activities required to solve a problem. In other ward, Methodology is a collection of procedure, technique, tools and documentation aid which helps the system developers in their effort to implement a new information system.

For the system development, I have used structured system development - Software Development Process. In system development I used the Structured Analysis methodology, which is based on process modeling. Here I used Fact Finding Techniques for requirement analysis. For process modeling Data Flow Diagram (DFD) was used.

3.2 SYSTEM DEVELOPMENT PROCESS

I have used the traditional system development technique:

- Preliminary Investigation
- System Analysis
- System Design
- Coding
- Testing

3.2.1 PRELIMINARY INVESTIGATION

At this Stage we had to perceive the problem and opportunities. We went to the Organization to review existing system. We had to find out initially what are the Organization activities and problem have been, and whether those could be resolved through available techniques known to me. Then we established the scope, preliminary requirement and constraints of the proposed system and project schedule. We found out that, such proposed system would be possible to make within given scope and it might be a feasible solution for the business organization.

3.2.2 SYSTEM ANALYSIS

At this stage of System Development Process we studied the existing system and tried to find out the requirements of the proposed system in details. we found out what is the purpose of the education department's activity, what are the steps performed in every department, how, where, how long and how often are they performed, how great the volume of transaction, who use the resulting information, what are the constraints, what is the input of each process and what are the outputs. We found out whether the proposed system would be feasible or not, what will be user requirements, record requirements.

There are following methods and techniques

The software was built using Visual Basic and it can run on computers that have the .NET Framework installed. We divided the whole project into five main modules: employee data, payroll records, rates, ledger and help.When a data is saved, all the data of the employee is stored, along with each month’s payroll data for the employee. Getting any data from the file will require the data to first be desterilized before it can be used.

3.2.3 SYSTEM DESIGN

After finishing the analysis part, started the designing phase of System Development

Process. In this part I used both logical and physical design. For logical design - we used process modeling by DFD (Data Flow Diagram).

Here I tried to implement my findings, which I got in System Analysis phase by DFD. How the system would work and how the output will be given was the main concern at this level. In logical design part-for the-we used MS Access for the database design and for front-end design, we used visual basic.net

3.2.4 CODING

Coding is the heart of software development. After the designing phase, we got up to coding. For Development of my project we choose VB.NET, MS Access, as it is very much suitable for building database applications quickly and effectively.

3.2.5 TESTING

After the development of the software had been finished, it went for testing. There are two types of testing- White Box and Black Box testing. I had done only Black Box testing for my software. After testing we found a test report and later on, we fixed those errors according to report. Some new requirements were imposed and enhancement request was made. The development process was stopped after testing.

Chapter-4
System Analysis

System analysis is the systematic analysis of a proposed system and the identification of the requirements that it should meet. It is a process of gathering and interpreting facts, diagnosing problem, defining the problem and recognizing causes. System Analysis is the starting point for system design. The term is most commonly used in the context of commercial programming, where software developers are often classed as either system analyst or programmers. The systems analysts are responsible for identifying requirements (i.e. systems analysis) and producing a design. The programmers are then responsible for implementing it. The System Analysis is performed in three different steps-

4.1 PROBLEM ANALYSIS

Organization maintains the entire student’s, fees, and employee’s related information. At present much education system uses a paper-based system to store and analyze that information. Some software like Ms. Word, MS Access, VB.NET etc. used on a limited basis.

To manage all these information a paper based-manual system is not efficient in education system. There are several activities and each activity has their own data to process. Therefore huge data collection is going all the day and managing all these information in a paper based system is very difficult, time consuming and enormous. So the challenge was to build software to overcome these problems.

Chapter-5

FEASIBILITY CONSIDERATIONS

There are three essential considerations that are involved in feasibility analysis:

- Technical Feasibility
- Economic feasibility
- Behavioral feasibility

Technical feasibility:-Technical feasibility centers on the existing computer system (hardware, software etc.) and to what extent. It can support the proposed addition for example, if the current ids operating 80% capacity and arbitrary ceiling then running another application could overload the system or required additional hardware. This involves financial consideration to accommodate technical enhancement .If the budget is a serious constraint, when the project is not feasible. The purpose system was found to be technically feasible in all respect.

Economical Feasibility:-Economic analysis is most frequent method used for evaluating the effectiveness of candidate system. More commonly known as cost and benefit analysis, the procedures is to determine the benefits and saving that are expected from a candidate system.

Compare then with the cost. The purposed system is economically feasible and fulfill all the paper work will be reduced and time consumed will become minimum. In short it will over weigh the existing system in cost compared to benefit.

Behavioral feasibility:-People are inherently resistant to change, and computers have been now to facilitate change .An estimate should be made be how strong is user staff is likely to have towards the development of computerized system.

5.1 Steps in feasibility analysis:

- It involves the seven steps:-

1. From a project team appoints a project leader.

2. Prepare system flow chart.
3. Enumerate potential candidate system.
4. Describe the identity characteristics of candidate system.
5. Determine the evaluate performance and cost effectiveness of each candidate system.
6. Weigh system performance and cost data.
7. Select the best candidate system.

5.2 SYSTEM ANALYSIS

Before we start studying the system we are to know with the term system. What is system? The term system is derived from Greek word system a, which means an organized relationship among functioning units or components. A system exits because it is designed to achieve one or more objectives. The term system refers to an orderly grouping of interdependent components linked together according to plan to achieve a specific objective the idea of system become more practical and necessary in conceptualizing the interrelationship and integration of operation especially when using computer. A study of system concept has three basic implications: A system must be designed to achieve a predetermined objective. Interrelationship and interdependence must exist among the component. The objective of organization as a whole has higher priority than the objective of its sub system in the present study system is an integrated collection of data files. The combination of all these programs and database made this system. The process of designing a system is not done in one phase. It consist of many phases can be seen comprising of four phases.

Analysis is detail study of various operations performed by the system and their relationship within and outside the system, outside factor also place major role in the system like govt. vendor, customer etc.

During analysis, data are collected on available files, decision points and transactions entered by the present system.

5.3 SYSTEM DESIGN

System Design is a solution, how to approach to creation of a new system. This important phase is composed of several steps. It provides the understanding and procedural details for implementing the system recommended infeasibility study. Stress in on translating performance requirement into design specification design goes through logical physical stages of development. Logical design reviews the present physical, prepare input and output specification. These steps are as follow:

1. Problem definition.
2. Input output specification.
3. Data based designed.
4. Modular program design.
5. Preparation of source code.
6. Testing and debug

5.4 TESTING

Testing is the major quality control measure employed during software development. Testing is the process of executing with intent of finding the error. No peace of code is completely ready unless it has been fully tested.

NEED FOR TESTING:-Testing is the vital to the success of the system. Testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved in adequate testing and non-testing leads to error that may not appear until month later. Finally, testing leads to software reliability. Identifying and removing faults, during testing can make the software more reliable.

The basic level of testing is:

- Unit testing
- Integration testing
- System testing
- Acceptance testing

Chapter-6

Hardware and Software Requirements

6.1 HARDWARE REQUIREMENTS

- Minimum Pentium IV Processor
- 256 MB RAM
- 2 GB of Hard Disk space
- Windows 7 or Windows XP Professional
- Combo Drive
- SVGA COLOR monitor
- 1.44" FDD, CD-ROM 52 X secondary device
- Mouse or any other pointing device

6.2 SOFTWARE USED

FRONTEND: VISUAL BASIC.NET

- Visual Basic was designed to be easy to learn and use.
- The language not only allows programmers to easily create simple GUI

applications, but also has the flexibility to develop fairly complex applications as well.

- Programming in VB is a combination of visually arranging components or controls on a form, specifying attributes and actions of those components, and writing additional lines of code for more functionality.
- Forms are created using drag and drop techniques. A tool is used to place controls (e.g., text boxes, buttons, etc.) on the form (window). Controls have attributes and event handlers associated with them.
- Default values are provided when the control is created, but may be changed by the programmer. Many attribute values can be modified during run time based on user actions or changes in the environment, providing a dynamic application.
- Visual Basic can create executables (EXE files), ActiveX controls, DLL files, but is primarily used to develop Windows applications and to interface web database systems. Dialog boxes with less functionality (e.g., no maximize/minimize control) can be used to provide pop-up capabilities.
- Alternatively, a Visual Basic component can have no user interface, and instead provide ActiveX objects to other programs via Component Object Model (COM). This allows for server-side processing or an add-in module.

TOOLBOX

The Tool box contains the tools which are used to place various controls on your forms. It displays all the standard Visual Basic controls in addition to any custom controls and objects you have added to your project with the Custom Controls dialog box. There are various tools in the Toolbox:

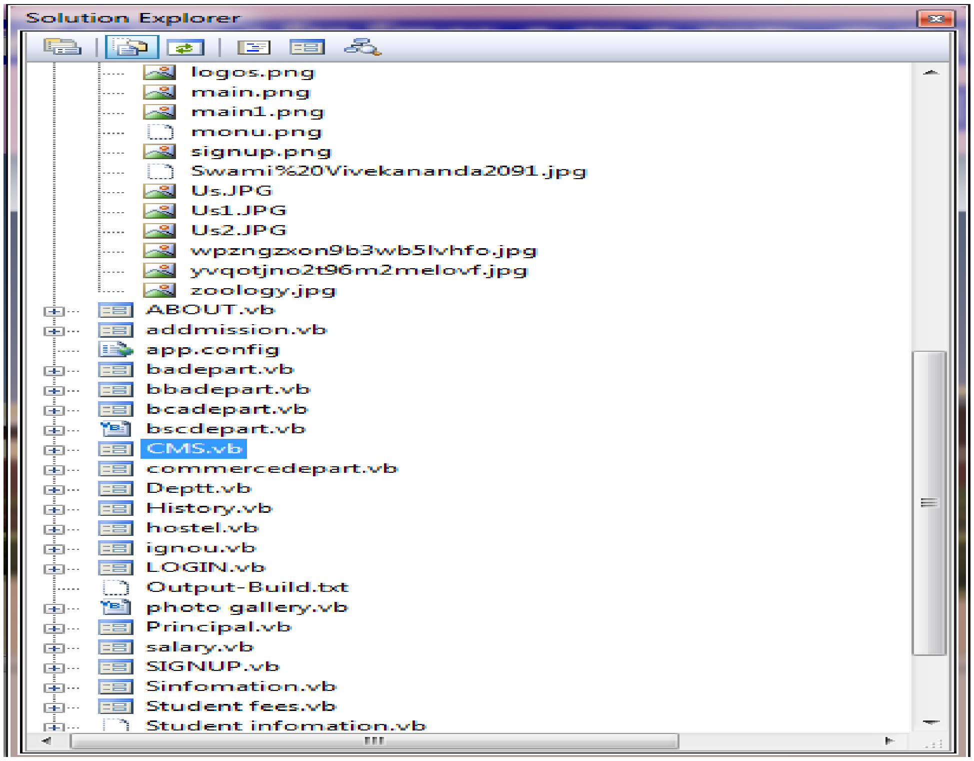
- **Label**: Used for text that you don't want the user to change, such as a caption under a graphic.
- **Command Button**: Used to create a button that the user can choose to carry out a

command.

- **Text Box**: Used to hold text that the user can either enter or change.
- **Main Menu**: Adds menus under the titles bar of the form i.e. File, Edit etc
- **Check Box**: Used to create a box that the user can easily choose to indicate if something is true or false, or to display multiple choices when the user can choose more than one.
- **Radio Button**: Used in a group of option buttons to display multiple choices from which the user can choose only one.
- **Picture Box**: Used to display graphical images (either decorative or active), as a container that receives output from graphics methods, or as a container for other controls.
- **Data grid**: Allows users see and edit multiple rows of data simultaneously, also useful for rapid entry of large amounts of data.
- **List Box**: Used to display a list of items from which the user can choose one. The list can be scrolled if it has more items than can be displayed at one time.
- **Checked List Box**: Allows the user to select multiple items in a list by providing a check box for each item.
- **Combo Box**: Used to draw a combination list box and text box. The user can either choose an item from the list or enter a value in the text box.
- **Horizontal Scroll Bar:** Used to provide a graphical tool for quickly navigating through a long list of items or a large amount of information, for indicating the current position on a scale, or as an input device or indicator of speed or quantity.
- **Vertical Scroll Bar:** Used to provide a graphical tool for quickly navigating through a long list of items or a large amount of information, for indicating the current position on a scale, or as an input device or indicator of speed or quantity.
- **Timer:** Used to generate timer events at set intervals. This control is invisible at run time.

- **Image List:** Just like it sounds a list of images. The list can be navigated by a button that scrolls through the images.
- **Help Provider:** Brings up the help window.
- **Tool Tip:** Used to display text when the user moves the mouse over a control.
- **Context Menu:** Displays a menu of frequently used commands associated with the selected item.
- **Tool Bar:** Allows the programmer to add a toolbar to his form.
- **Status Bar:** Used to allow an application to display various information about its status.
- **Print Preview Control:** Opens print preview with the default settings.
- **Print Document:** Prints the current document with the default printer settings

SOLUTION EXPLORER:



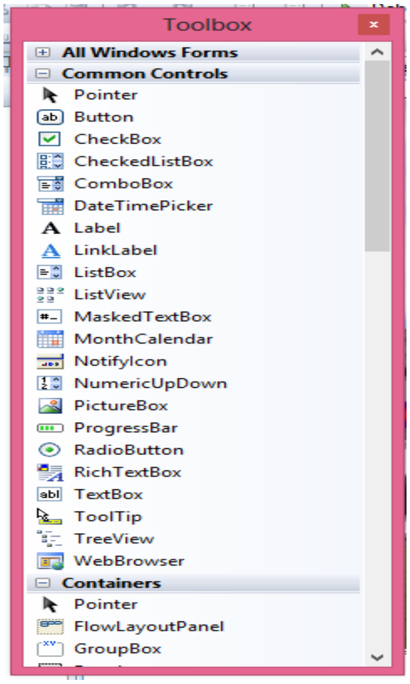
```
logos.png
main.png
main1.png
monu.png
signup.png
Swami%20Vivekananda2091.jpg
Us.JPG
Us1.JPG
Us2.JPG
wpzngzxon9b3wb5lvhfo.jpg
yvqotjno2t96m2melovf.jpg
zoology.jpg
ABOUT.vb
addmission.vb
app.config
badepart.vb
bbadepart.vb
bcadepart.vb
bscdepart.vb
CMS.vb
commercedepart.vb
Deptt.vb
History.vb
hostel.vb
ignou.vb
LOGIN.vb
Output-Build.txt
photo gallery.vb
Principal.vb
salary.vb
SIGNUP.vb
Sinfomation.vb
Student fees.vb
Student infomation.vb
```

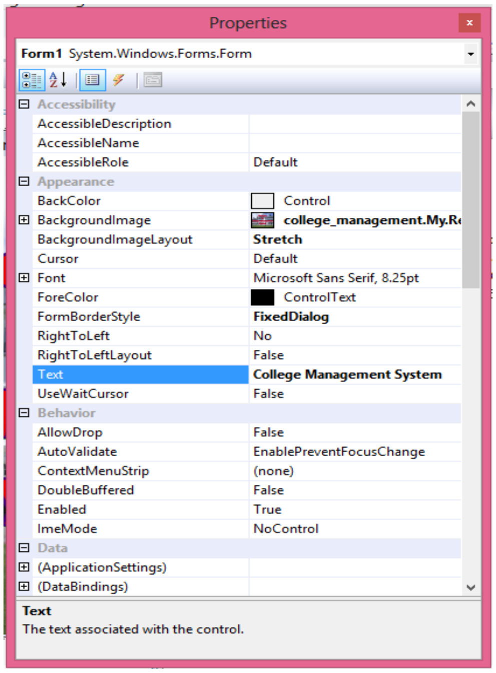
Fig. 1.1

This is the window that allows coordinating the parts of the program into folders for easy manipulations as all the parts of that project appear in review. To pick the required part of the project, find that from the project explorer and double click on it, you will get that. Items such as forms can be added or removed from in the project explorer. The buttons at the top allows the user to switch between the different views. The left button displays the object code window, the middle button displays the object itself and the right button toggles the folder open and closed in the project explorer. So, project explorer gives us a valuable overview of our entire project, which is very useful when a project gets too large and contains many components.

PROPERTIES WINDOW:

This is the window where the properties of an object can be set. When any object is selected in visual basic using mouse, its properties are displayed in the properties window. Properties of an object can be changed in two ways- either at design time or runtime. Properties appear in the properties window are set at design time.





Form1 System.Windows.Forms.Form

⊟ Accessibility	
AccessibleDescription	
AccessibleName	
AccessibleRole	Default
⊟ Appearance	
BackColor	□ Control
⊞ BackgroundImage	**college_management.My.R**
BackgroundImageLayout	**Stretch**
Cursor	Default
⊞ Font	Microsoft Sans Serif, 8.25pt
ForeColor	■ ControlText
FormBorderStyle	**FixedDialog**
RightToLeft	No
RightToLeftLayout	False
Text	**College Management System**
UseWaitCursor	False
⊟ Behavior	
AllowDrop	False
AutoValidate	EnablePreventFocusChange
ContextMenuStrip	(none)
DoubleBuffered	False
Enabled	True
ImeMode	NoControl
⊟ Data	
⊞ (ApplicationSettings)	
⊞ (DataBindings)	

Text
The text associated with the control.

TOOL BOX PROPERTY WINDOW

Fig. 1.2 Fig. 1.3

Features of Visual Basic.Net:

Writing a graphical user interface program is much easier in Visual Basic. Visual Basic has a very friendly environment, which helps to create forms, add contacts to the form and write code behind the form, very quickly and easily. Visual basic also has an online help system. This Online help contains thousands of programming

Example. There are many other features in Visual Basic which are discussed as follows:

Event–driven programming language:

Everything in Visual Basic is event-driven. Unlike earlier programming languages like COBAL and Pascal in which control was with the application, in Visual Basic user is in the control of the application. Every time the user clicks a command buttons or presses the mouse, an event stream is generated and coded that has been written behind the event is executed. After the execution of the code the control again comes back to the user.

GUI / Window Environment :

An application developed in Visual Basic has the looks and feel of windows application development system. It behaves like any other Windows program to which any user is accustomed, so, for any user it is not new and he feels very comfortable while using an application developed in Visual Basic.

Internet Based Application:

Visual Basic can be used to write Internet programs. Visual basic can be extended to other application by the use of Active X controls, Dynamically Linked Libraries (DLL's), and Add-Ins.

VISUAL BASIC.NET (FRONT END):

The start in VISUAL STUDIO has following in steps are:-

1. Click the start menu
2. Click the Programs
3. Click the MS Visual Studio 2008

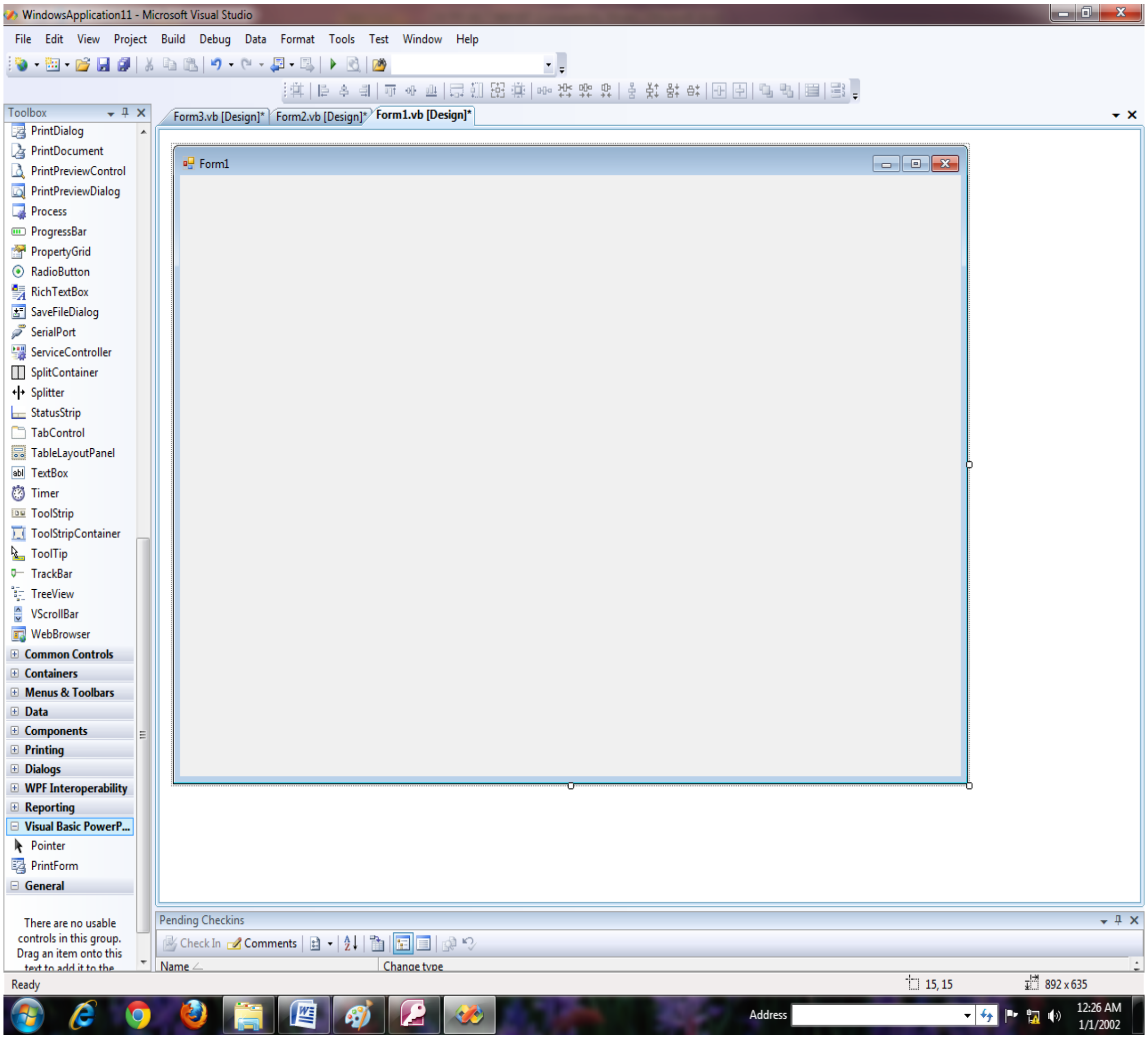


Fig. 1.4

In this project I use visual basic.NET as programming language which is an application development tool as well as a relational data base management system which use client/ server architecture I have taken front end tools as visual basic 2008 and ms-access has taken as back end tools to store data in the form of relational the visual basic makes the project interactive , thus helping the operator to use the Application programmer conveniently visual basic is window based language for making advance window programs. As the more suggest a big portion of the programming is analyses visually .This means during design this one could be able to see how the programmed will look during design time. This is great advantage over the other programming languages because we are able to change and practice with our design, color, size, screen etc. which are included in programmatic follows the standard syntax of basic except that some new language feature have also been added to give flexibility to the programmer. VISUAL STUDIO realize very heavily on the window user interface. Therefore the only prerequisite to the work in the visual basic environment to have firm understanding of how to use mouse, windows, pull down menus and dialog boxes.

6.3 Back End: MS Access

Microsoft access is a Relational Database Management System (RDBMS) used to store and manipulate large collection of information of any kind. Here RDBMS refers to the organization of data in series of rows and columns in such a manner that any specific piece of information is available with the click of mouse and few keystrokes. MS-Access has tools, which are easy to use and provide powerful development environment, making it an appropriate choice for novices as well as professionals. For example, MS access can be used to enter and maintain student records, telephone numbers. Once the records are stored any type of queries can be asked, reports can be created and data entry forms can be designed. At an advanced level, access can be used for developed custom database application by employing access and a distribution kit for compiling applications.

Hardware and software requirements Access

Access, a part of Microsoft office, is windows based application software. It can run on MS Window 95/98/ME/XP/NT/2000. A Computer may have 80386, 80486 or Pentium based processor with minimum 8 MB RAM, VGA graphics display adapter and a mouse.

FEATURES OF MS-ACCESS:

MS-Access provides a database engine and stores all the application data tables, indexes, forms reports in MS database file (.mdb) file. The engine also provides advanced capabilities, such as data validation, concurrently control using locks, query optimization.

It provides a programming language called access basic. It can be used for developing custom database applications.

It provides the database developer with hyperlinks as a native data type, extending the functionality of database with the ability to share information on internet.

It provides the facility to specify the adhoc queries graphically over an oracle database stored in UNIX server .Also the forms and reports can be made graphically.

It provides the facility use active data objects (ado) to access and manipulate data in database server through any OLE database provider .It supports multi-user operations by making an application available on a network server. For concurrent updates, locking is provided. MS-ACCESS maintains an LDB file that contains current locking information .The locking property can be set for a given form .It helps to input and export from the access database to other

Database software. For example –access can import table from oracle.

COMPONENTS OF MS ACCESS:

The components of MS-Access are the objects, which can be seen in database window. The Database window is opened whenever we create a new database or open an existing database. The various objects shown in a database window are Tables, Forms, Reports, Macro, Modules etc .All object are managed through the database window and all objects of a database window and all objects of a database are stored in a single file.

DATABASE:

A database is a collection of information related to a particular subject or purpose such as tracking customer orders or maintaining a music collection. Using Microsoft access, you can manage all your information from a single database file. With the file, data is divided into separate storage containers called tables, views. One may add, and update table data using online forms, find and retrieve just the data you want using queries, and analyze or data in a specific layout using reports.

To store your data, create one table for each type of information you track. To bring the data from multiple tables together in a query, from, or report, you have defined relationship between the tables. To find and retrieve just the data the meets conditions you specify, including data from multiple tables, create a query. A query can also update or delete multiple records at the same time, and perform built- in or custom calculations on your data,

To easily view, enter and change data directly in a table create a form. When you open a form, Microsoft access retrieves the data from one or more table

And tables and displays it on screen using the layout chosen in the form wizard or using a layout that you created from scratch.

ADVANTAGE OF DATABASE:

The stored information can be viewed, stored manipulated, retrieved and printed as per requirement.

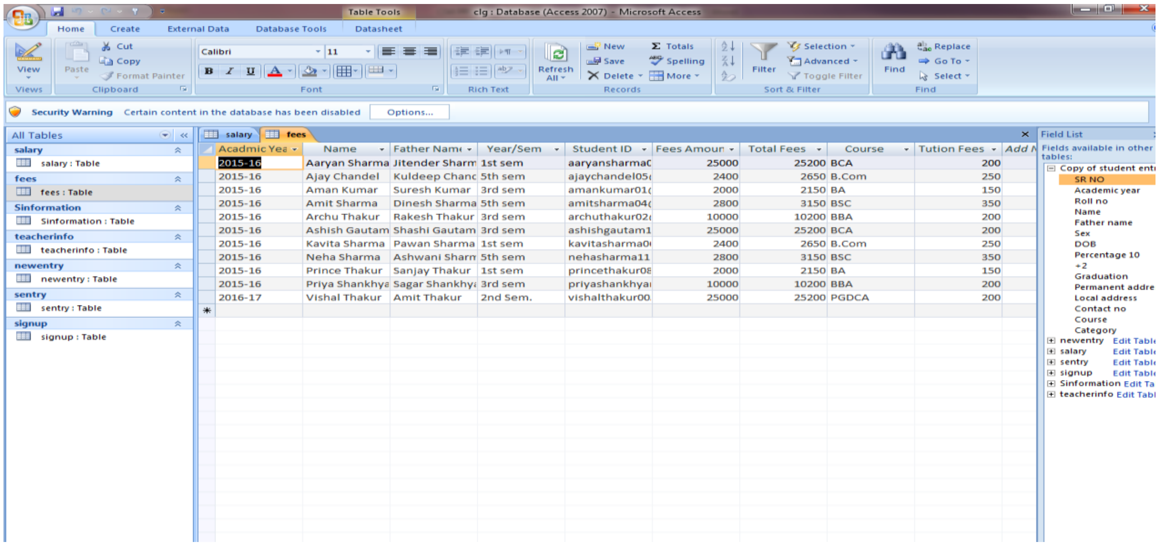
The data can be obtained in multiple formats.

The information in Access if more and interrelated can be retrieved quickly and accurately.

DATABASE:

The Database use in table the names are following as :-

1. STUDENT
2. STAFF
3. WORKER



Acadmic Yea	Name	Father Nam	Year/Sem	Student ID	Fees Amour	Total Fees	Course	Tution Fees	Add N
2015-16	Aaryan Sharma	Jitender Sharm	1st sem	aaryansharma0	25000	25200	BCA	200	
2015-16	Ajay Chandel	Kuldeep Chanc	5th sem	ajaychandel05	2400	2650	B.Com	250	
2015-16	Aman Kumar	Suresh Kumar	3rd sem	amankumar01	2000	2150	BA	150	
2015-16	Amit Sharma	Dinesh Sharma	5th sem	amitsharma04	2800	3150	BSC	350	
2015-16	Archu Thakur	Rakesh Thakur	3rd sem	archuthakur02	10000	10200	BBA	200	
2015-16	Ashish Gautam	Shashi Gautam	3rd sem	ashishgautam1	25000	25200	BCA	200	
2015-16	Kavita Sharma	Pawan Sharma	1st sem	kavitasharma0	2400	2650	B.Com	250	
2015-16	Neha Sharma	Ashwani Sharn	5th sem	nehasharma11	2800	3150	BSC	350	
2015-16	Prince Thakur	Sanjay Thakur	1st sem	princethakur08	2000	2150	BA	150	
2015-16	Priya Shankhya	Sagar Shankhy	3rd sem	priyashankhya	10000	10200	BBA	200	
2016-17	Vishal Thakur	Amit Thakur	2nd Sem.	vishalthakur00	25000	25200	PGDCA	200	

Fig. 1.5

TABLES OF PROJECT:-

STUDENTS INFORMATION:-

Field Name	**Data Type**	**Size**
Name	Text	30
DOB	Text	30
Permanent address	Text	30
Local address	Text	30
Contact no	Text	30
Percentage10th	Number	Long integer
+2	Number	Long integer
Graduation	Number	Long integer
Course	Text	30

Roll no	Number	Long integer
Father's name	Text	30
Category	Text	30
Year/Sem	Text	30
Academic Year	Text	30
Email	Text	30

TEACHERS INFORMATION:-

Field Name	**Data Type**	**Size**
Name	Text	25
Father's name	Text	25
DOB	Text	25
Permanent address	Text	25
Local address	Text	25
Contact No	Text	25
Teacher Id	Text	25
Experience	Text	25
E-mail	Text	25
Married Status	Text	25
Qualification	Text	25

TEACHER NEW ENTRY:-

Field Name	**Data Type**	**Size**
SR NO	Text	20
DOB	Text	20

Qualification	Text	20
Experience	Text	20
Permanent address	Text	20
Local address	Text	20
Contact no	Text	20
Material Status	Text	20
Acadmic Year	Text	20
Basic Salary	Number	Long integer
DA	Number	Long integer
HRA	Number	Long integer
Gross Pay	Number	Long integer
Other Diction	Number	Long integer
PF	Number	Long integer
Total Salary	Number	Long integer

STUDENT NEW ENTERY:-

Field Name	**Data Type**	**Size**
Academic year	Text	20
Roll no	Number	Long integer
Name	Text	20
Father name	Text	20
Gender	Text	20
DOB	Text	20
Percentage 10	Number	Long integer
+2	Number	Long integer
Graduation	Number	Long integer

Permanent address	Text	20
Local address	Text	20
Contact no	Text	20
Course	Text	20
Category	Text	20
Tuition Fee	Number	Long integer
Fees Amount	Number	Long integer
Total Fees	Number	Long integer
E-mail	Text	20
Year/Sem	Text	20

TEACHER SALARY DETAIL:-

Field Name	**Data Type**	**Size**
Teacher ID	Text	20
Name	Text	20
Phone no	Text	20
Basic Salary	Number	Long integer
DA	Number	Long integer
HRA	Number	Long integer
PF	Number	Long integer
Net Pay	Number	Long integer
Gross Salary	Number	Long integer
Total Salary	Number	Long integer

STUDENT FEES DETAIL:-

Field Name	Data Type	Size
Academic Year	Text	20
Name	Text	20
Father Name	Text	20
Course	Text	20
Year/Sem	Text	20
Student ID	Text	20
Tuition Fees	Number	Long integer
Fees Amount	Number	Long integer
Total Fees	Number	Long integer

LOGIN ACCOUNT:-

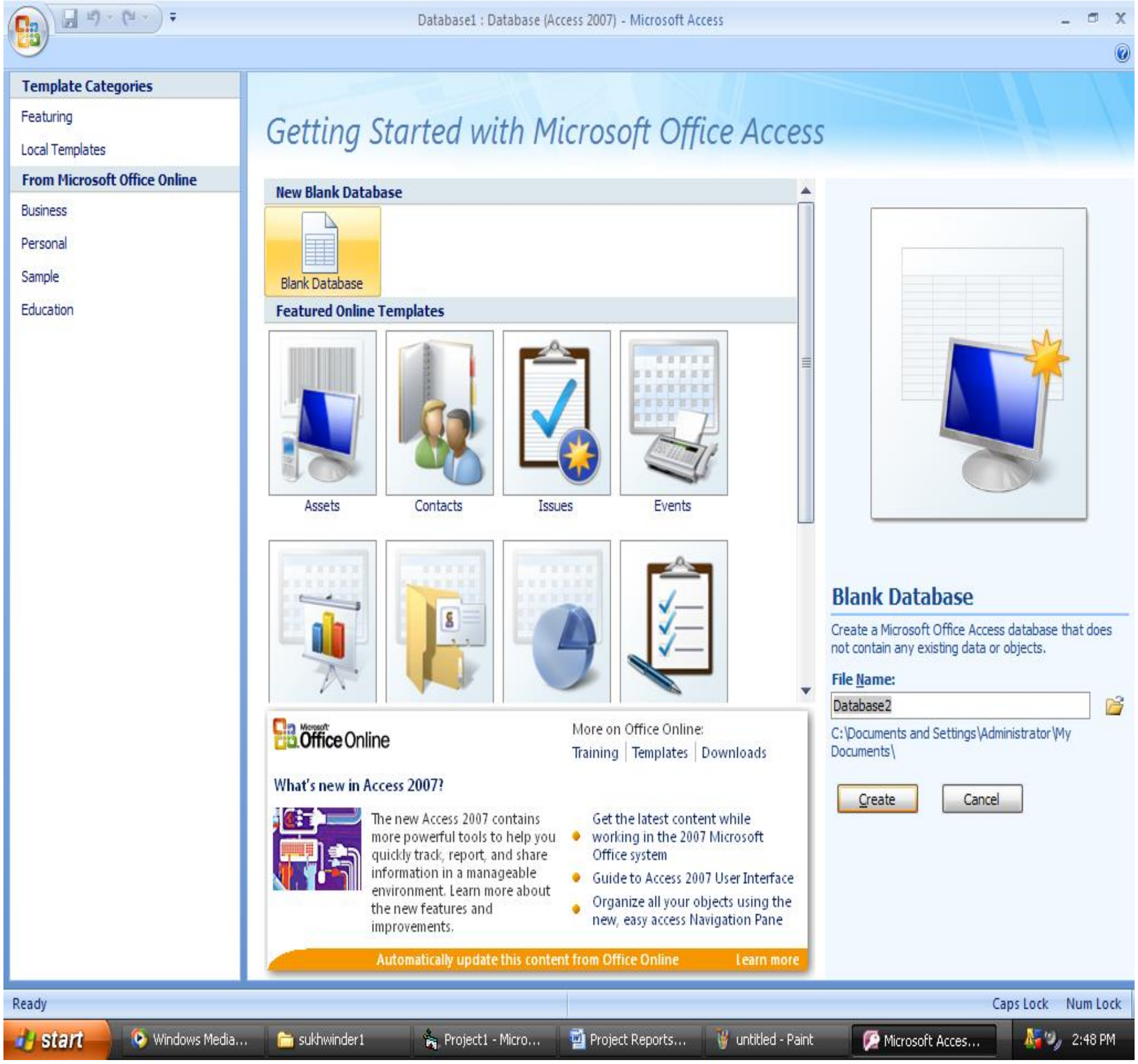
Field Name	Data Type	Size
User Name	Text	10
Password	Text	9

Database Design

Designing a database is much like designing anything else a building, a car, or a roadway through a city. Care must to take to design a plan. Many approaches can be taken to explain database design. It can be debated indefinitely what to include, what not be include, and in what order to present the material.

Before a design effort can be precede full speed ahead, the designer must first take time to understand the business. Understanding the business involves understanding the entities, data and rules within an organization and then converting these attributes of business into a business model .Then the designer must have a solid comprehension of proposed database model. Finally, the designer will convert the business model into a database model, using a design methodology, whether automated or a manual process.

MS ACCESS (BACK END)



Create a Microsoft Office Access database that does not contain any existing data or objects.

File Name: Database2

C:\Documents and Settings\Administrator\My Documents\

The new Access 2007 contains more powerful tools to help you quickly track, report, and share information in a manageable environment. Learn more about the new features and improvements.

- Get the latest content while working in the 2007 Microsoft Office system
- Guide to Access 2007 User Interface
- Organize all your objects using the new, easy access Navigation Pane

Fig. 1.6

Chapter-7

DESIGN

WELCOME FORM

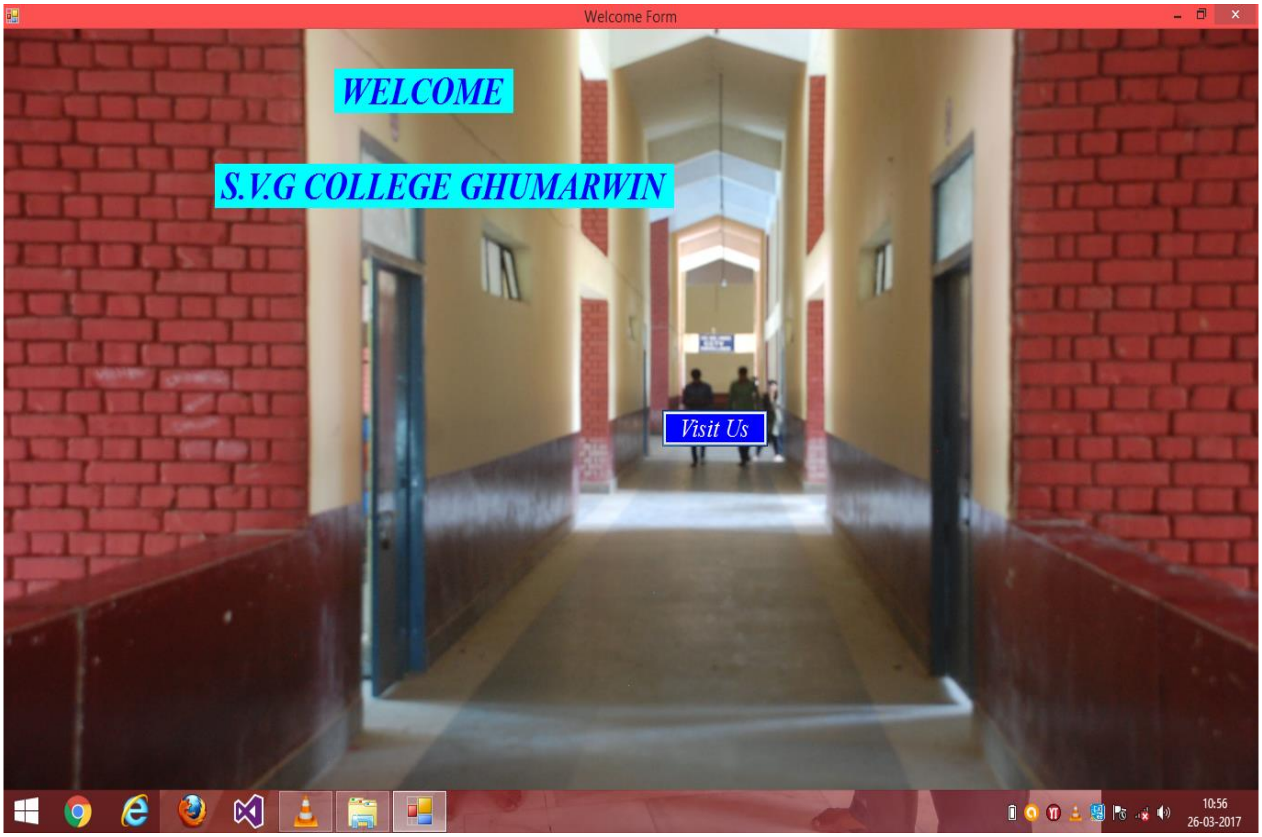


Fig. 1.7The Welcome form picture

LOGIN FORM

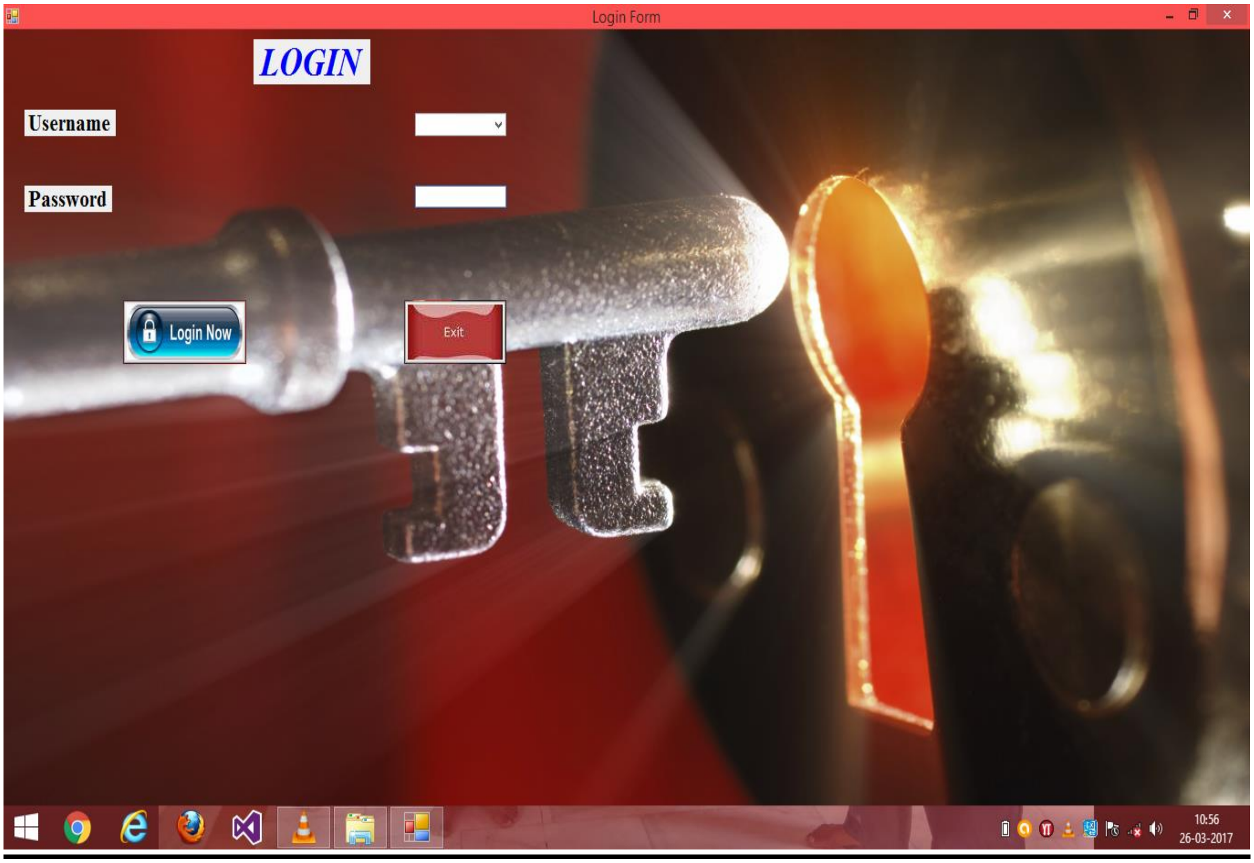
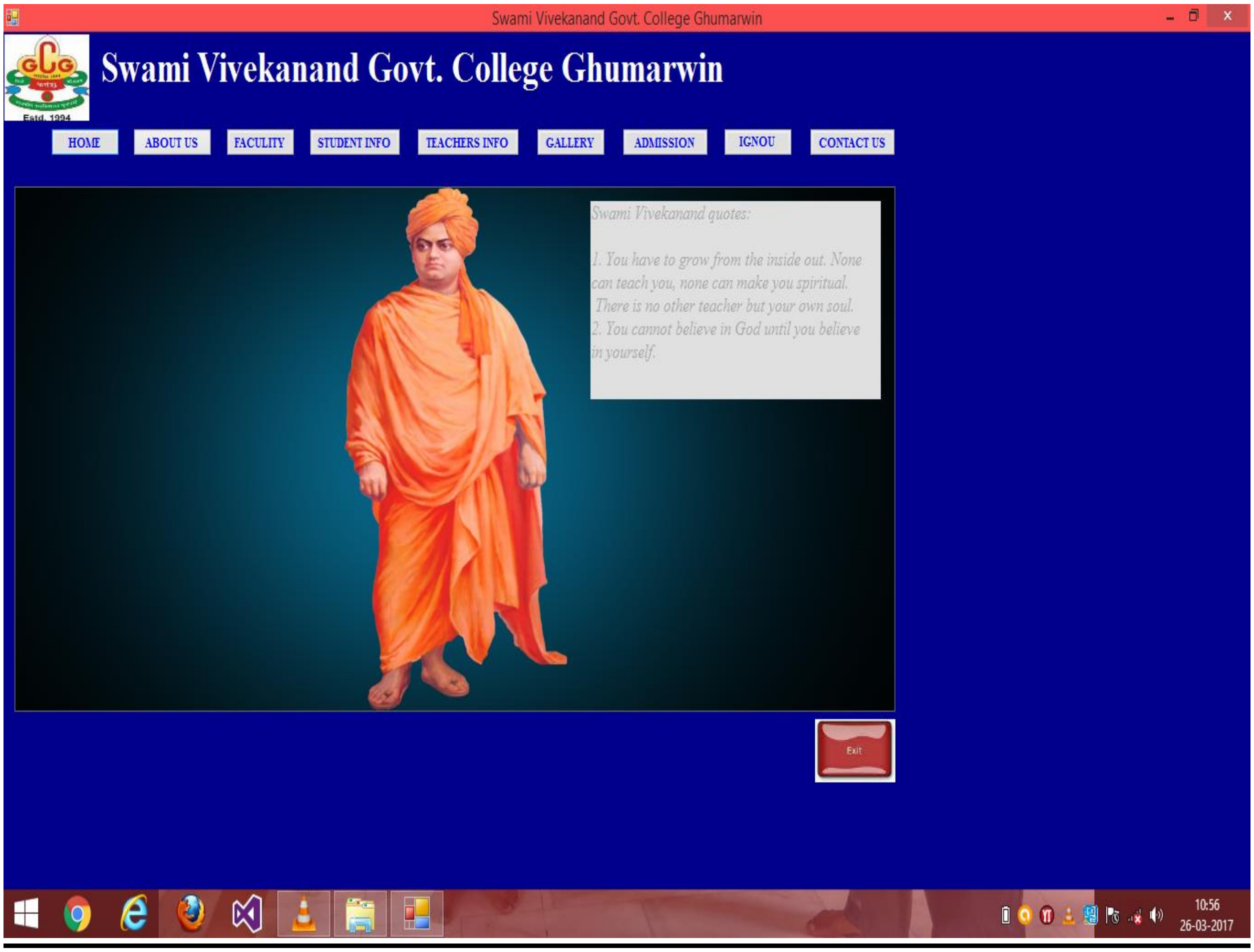


Fig. 1.8 The Login form picture

- This form is an entry point of the project.
- In this form user can enter the user name and password.
- And user can access the Information update.

MDI FORM



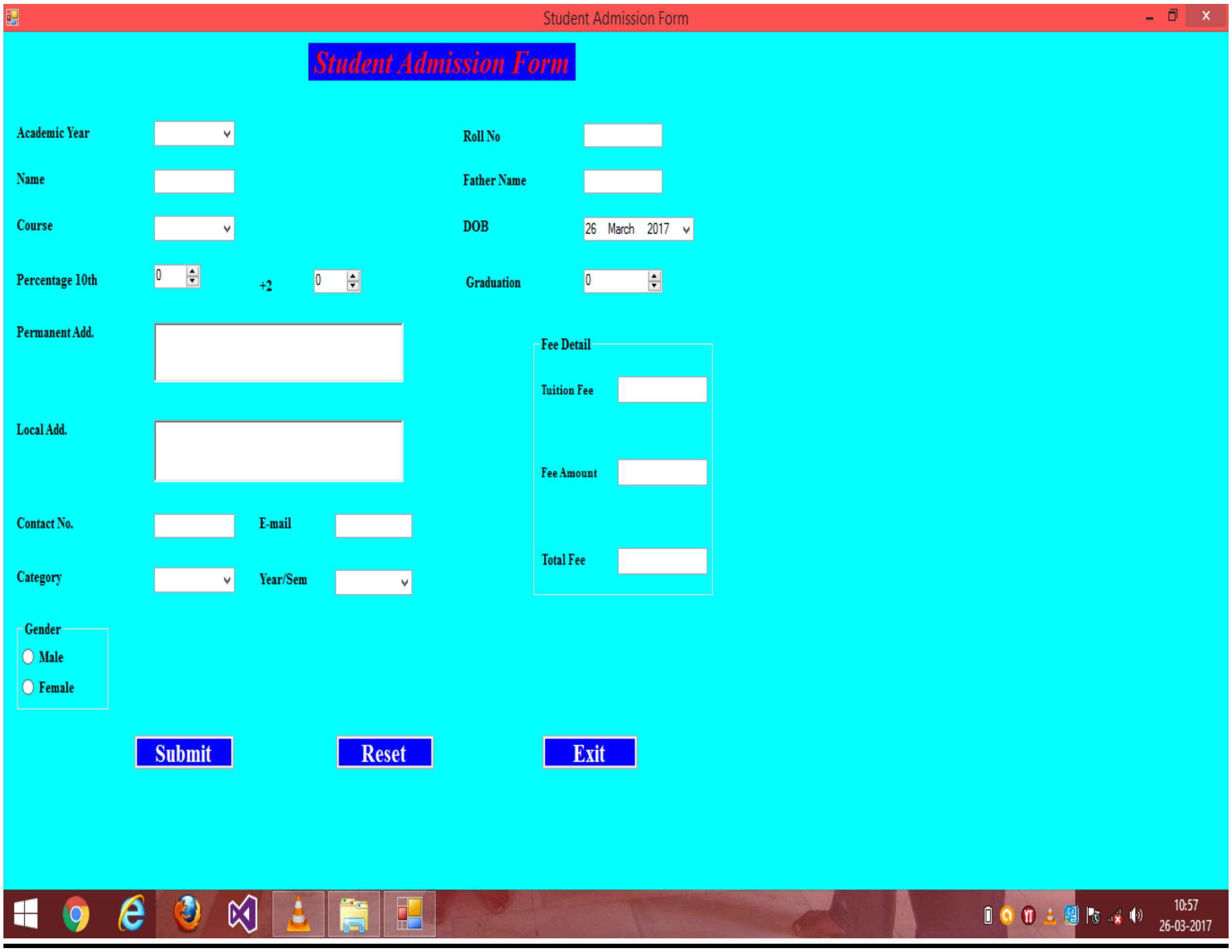
Swami Vivekanand quotes:

1. You have to grow from the inside out. None can teach you, none can make you spiritual. There is no other teacher but your own soul.
2. You cannot believe in God until you believe in yourself.

Fig. 1.9 The MDI form

- ➢ In this form user can access all the fields.
- ➢ It contains all the information.
- ➢ In this form user can access all the information

STUDENT ADMISSION FORM



DOB: 26 March 2017

Percentage 10th: 0

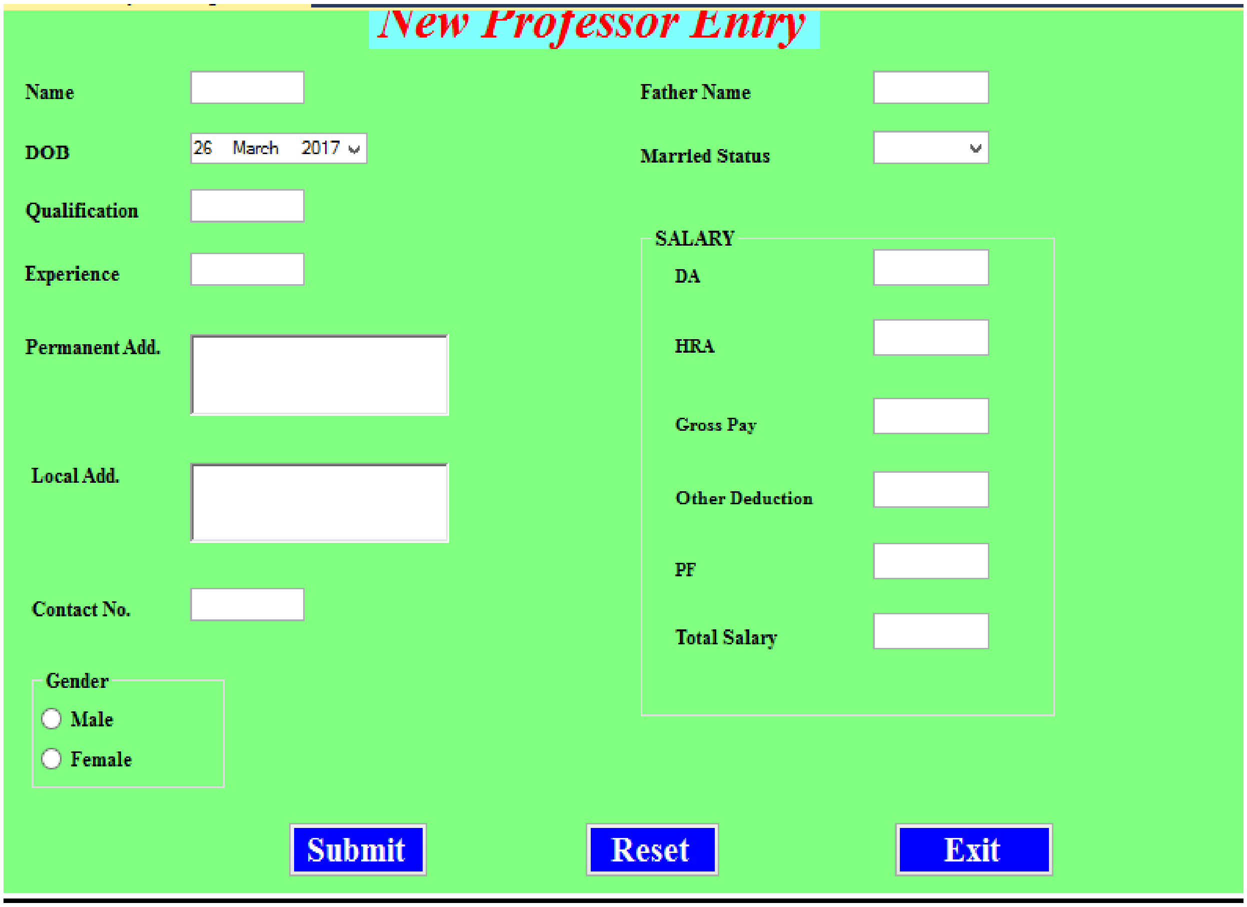
+2: 0

Graduation: 0

Fig. 1.10 The student form

- User can add the new student information

THE TEACHERS FORM



DOB 26 March 2017

Fig. 1.11 The teachers form

- ➢ User can add the new professor information

STUDENT INFORMATION FORM

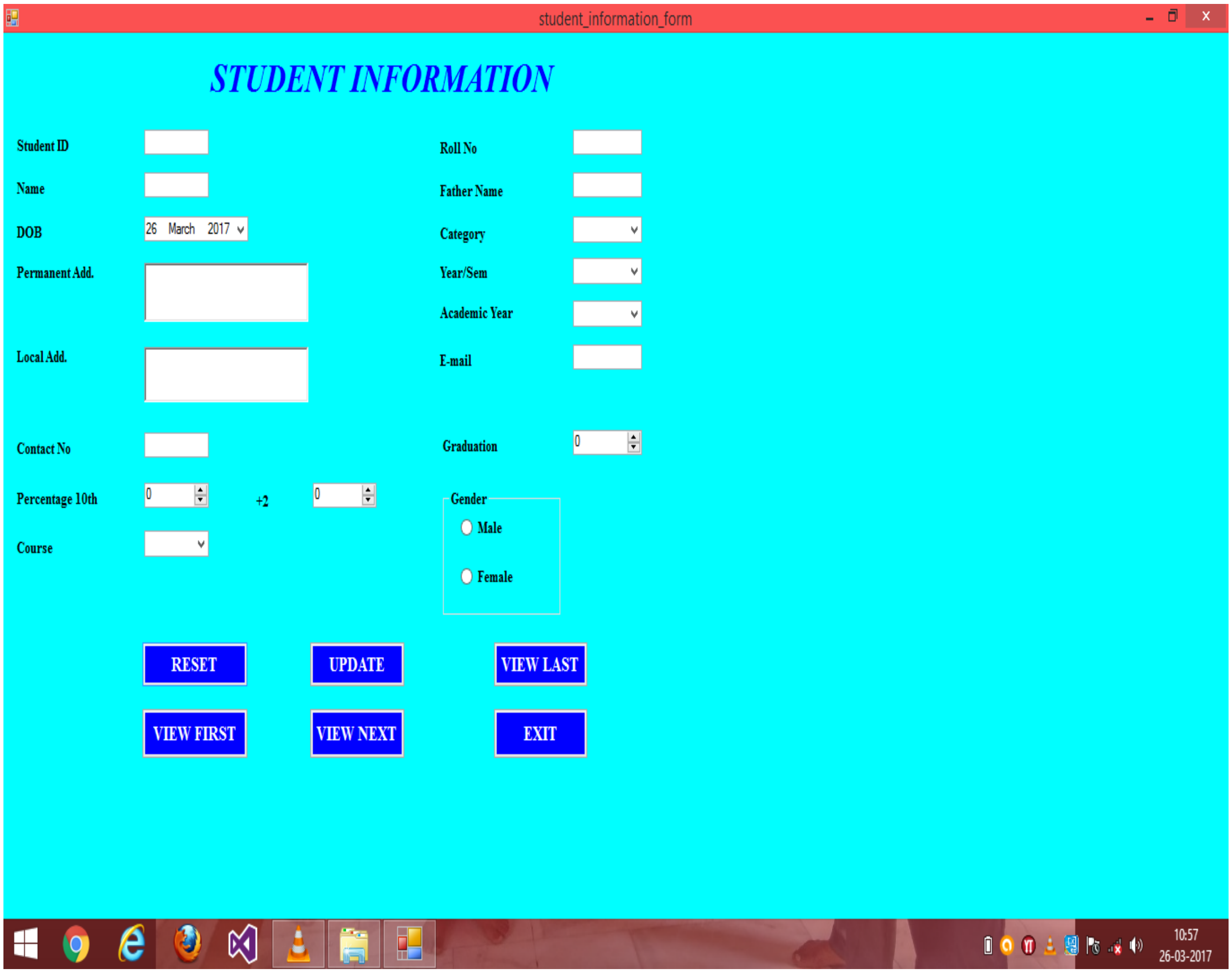
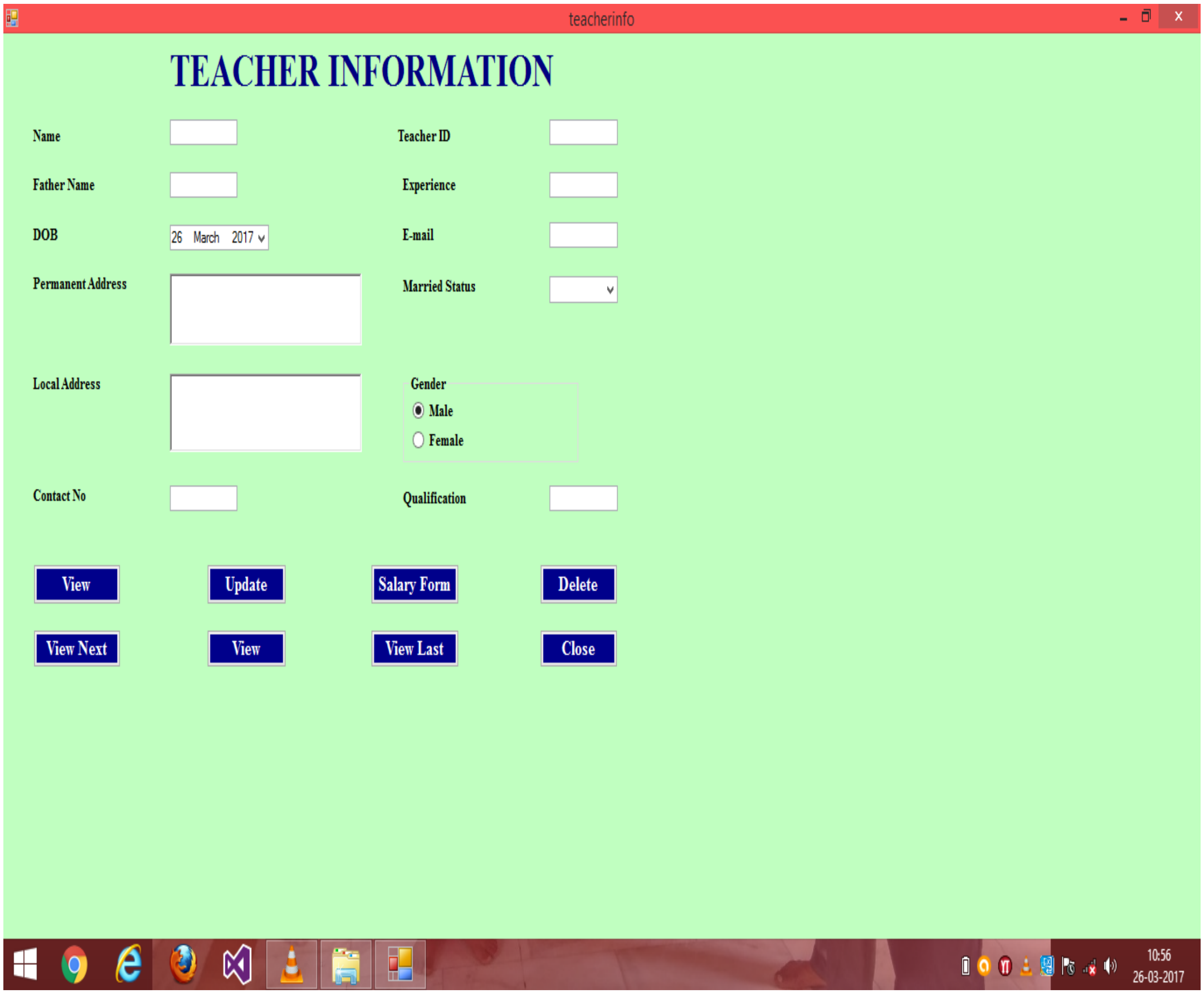


Fig. 1.12 The student form

- User can update the Student information.
- User can view the Student information.

TEACHER INFORMATION FORM



DOB 26 March 2017

Gender
● Male
○ Female

Fig. 1.13 The teacher form

- ➢ User can update the Teacher information.
- ➢ User can view the Teacher information.

STUDENT FEES DETAIL FORM

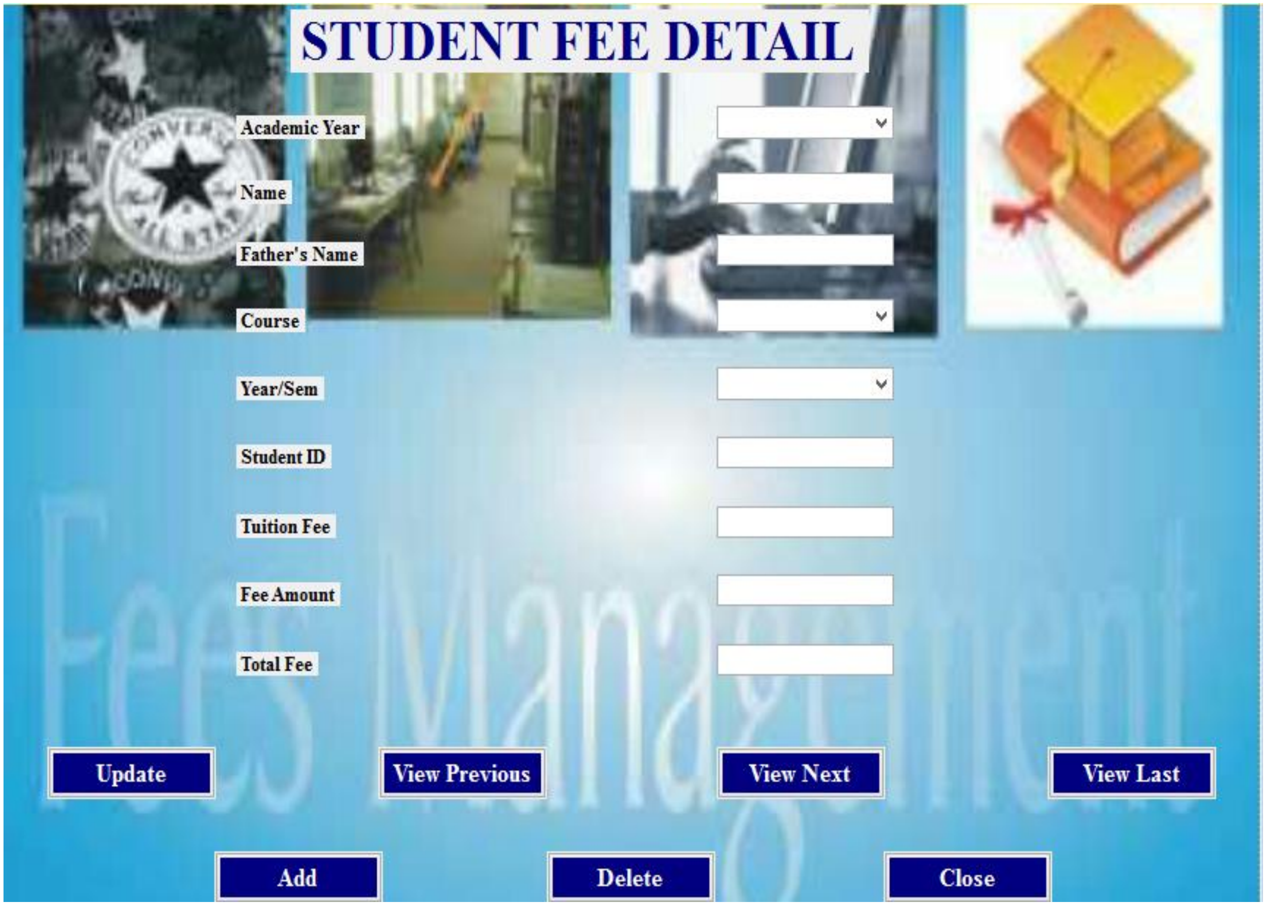


Fig. 1.14 The fees detail form

- ➢ User can update the student fees detail.
- ➢ User can add & delete the students fees detail

TEACHER SALARY DETAIL FORM

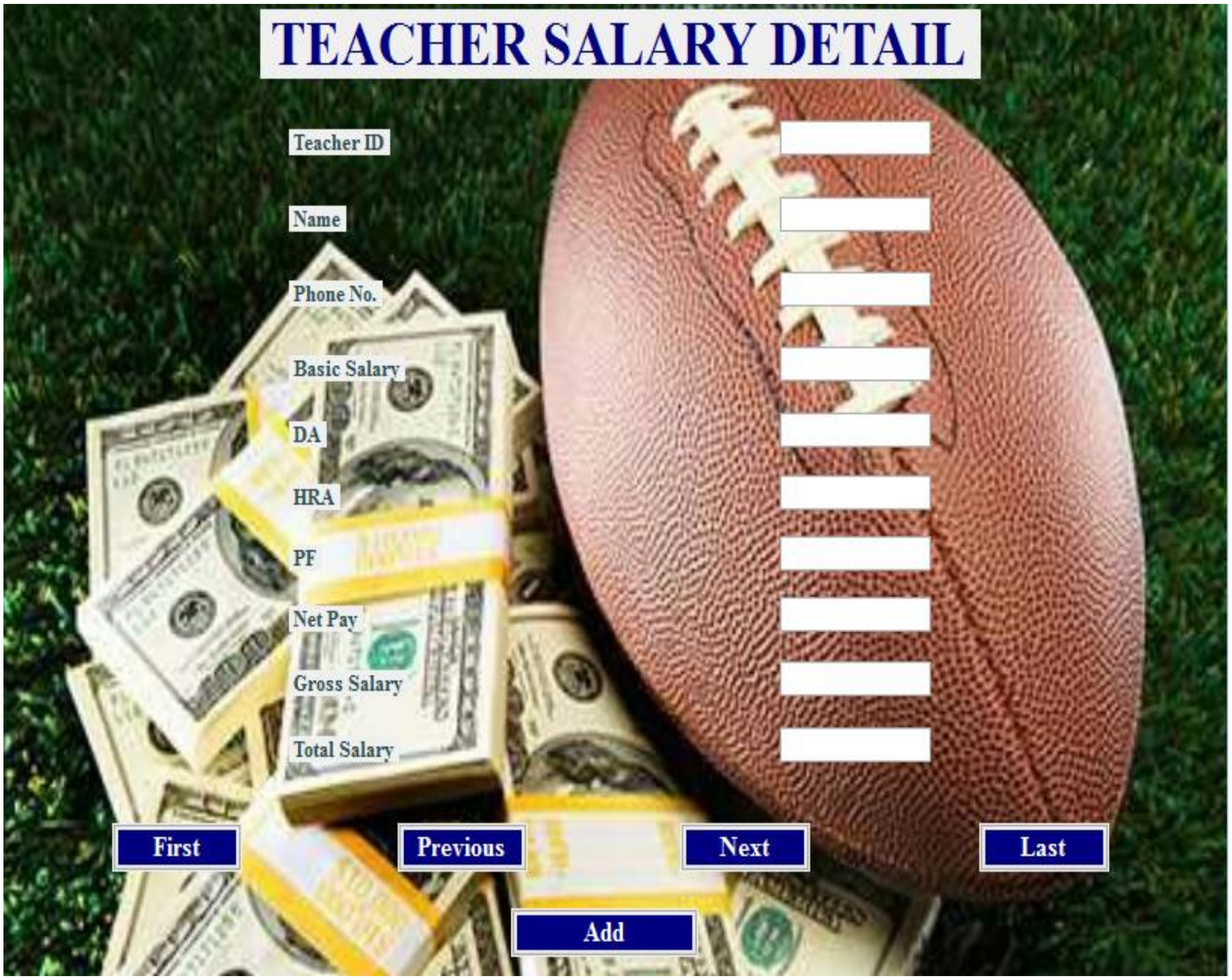
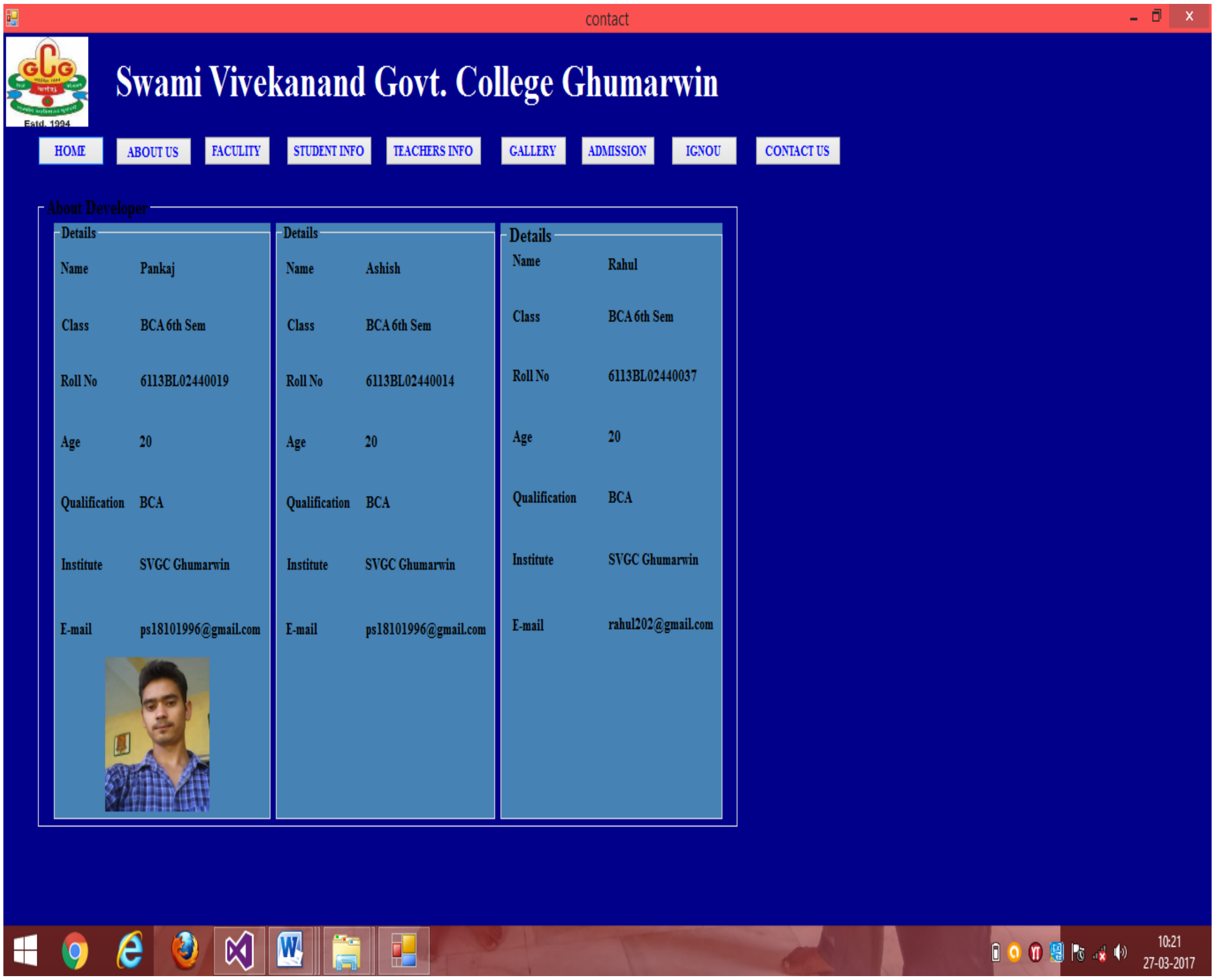


Fig. 1.15 The salary detail form

- ➢ User can update the Teacher salary.
- ➢ User can add & delete the Teacher salary.
- ➢ User can calculate the Teacher salary.

ABOUT US FORM



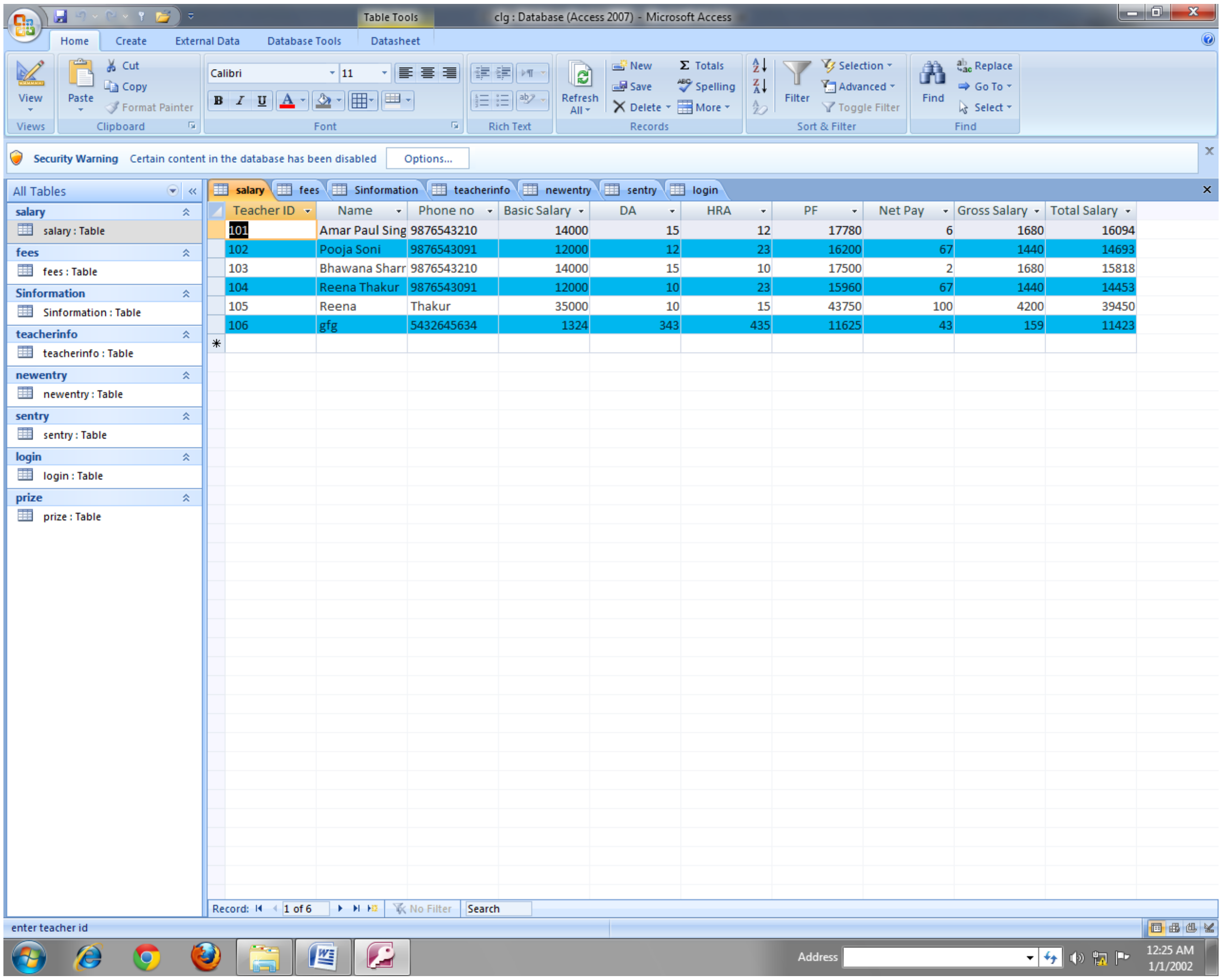
About Developer

Details		Details		Details	
Name	Pankaj	Name	Ashish	Name	Rahul
Class	BCA 6th Sem	Class	BCA 6th Sem	Class	BCA 6th Sem
Roll No	6113BL02440019	Roll No	6113BL02440014	Roll No	6113BL02440037
Age	20	Age	20	Age	20
Qualification	BCA	Qualification	BCA	Qualification	BCA
Institute	SVGC Ghumarwin	Institute	SVGC Ghumarwin	Institute	SVGC Ghumarwin
E-mail	ps18101996@gmail.com	E-mail	ps18101996@gmail.com	E-mail	rahul202@gmail.com

Fig. 1.16 About us form

SNAPSHOT OF TABLES

TEACHER SALARY TABLE



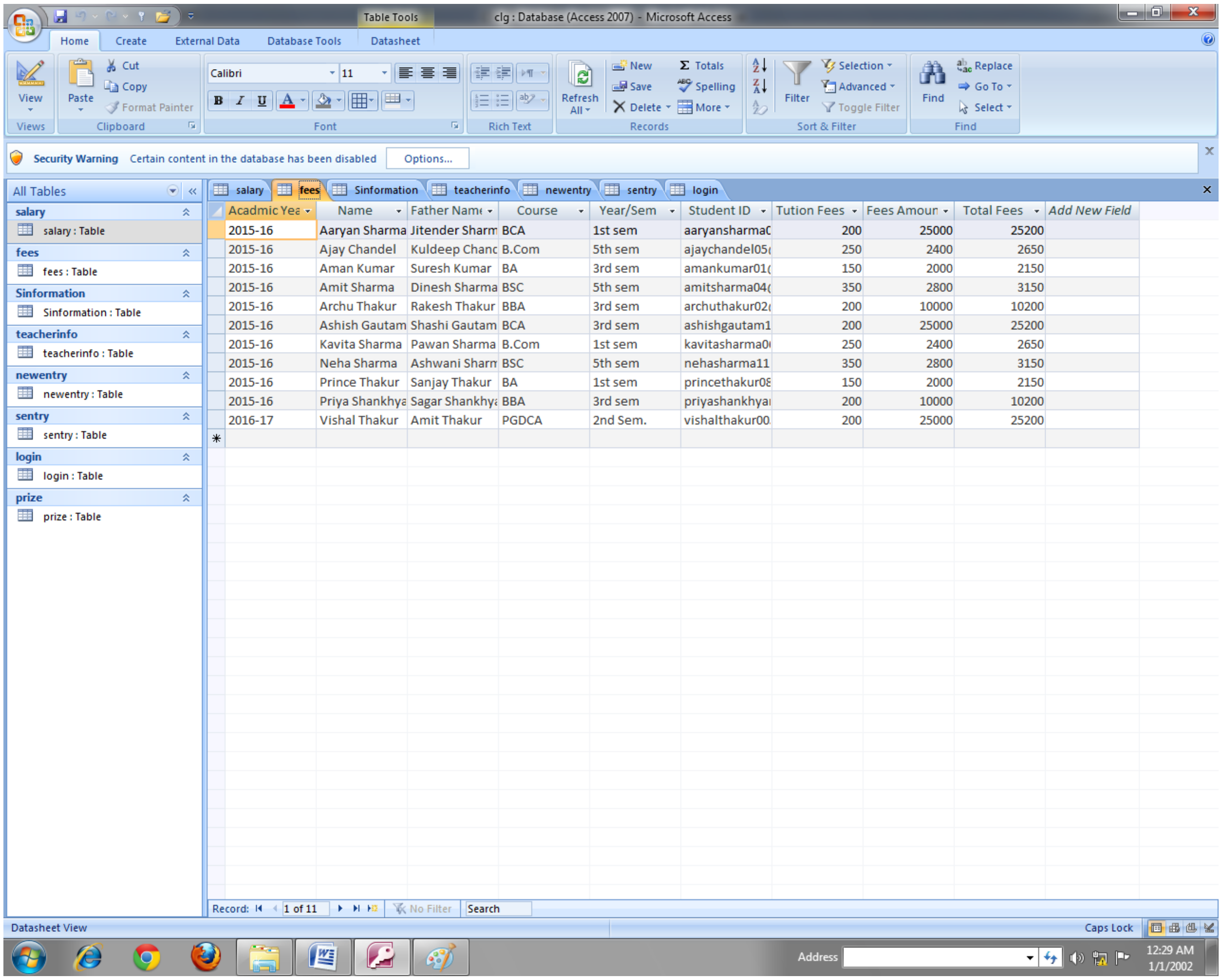
Security Warning Certain content in the database has been disabled Options...

Teacher ID	Name	Phone no	Basic Salary	DA	HRA	PF	Net Pay	Gross Salary	Total Salary
101	Amar Paul Sing	9876543210	14000	15	12	17780	6	1680	16094
102	Pooja Soni	9876543091	12000	12	23	16200	67	1440	14693
103	Bhawana Sharr	9876543210	14000	15	10	17500	2	1680	15818
104	Reena Thakur	9876543091	12000	10	23	15960	67	1440	14453
105	Reena	Thakur	35000	10	15	43750	100	4200	39450
106	gfg	5432645634	1324	343	435	11625	43	159	11423

enter teacher id

Fig. 1.17 TEACHER SALARY TABLE

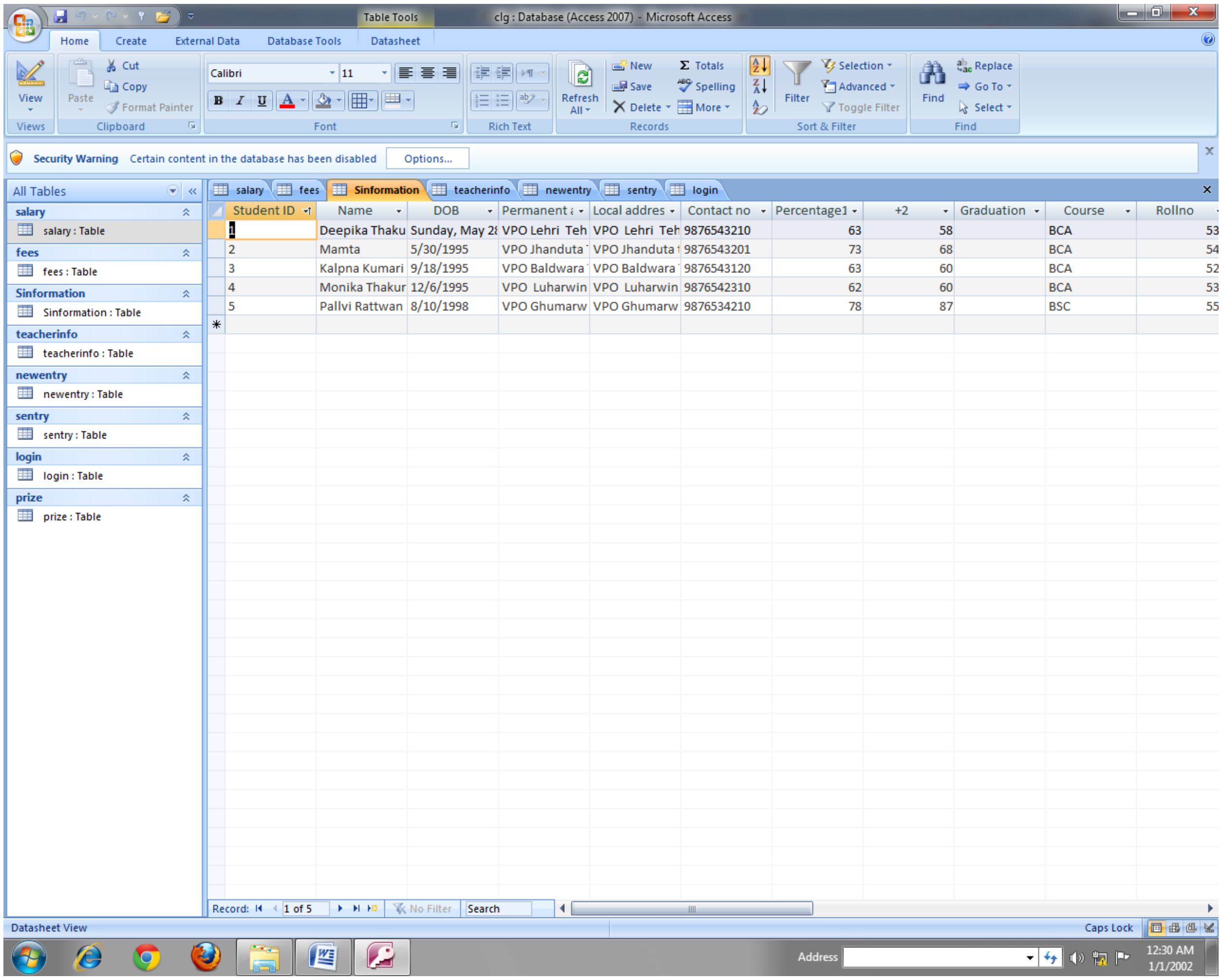
STUDENT FEES DETAIL TABLE



Acadmic Yea	Name	Father Name	Course	Year/Sem	Student ID	Tution Fees	Fees Amour	Total Fees	Add New Field
2015-16	Aaryan Sharma	Jitender Sharm	BCA	1st sem	aaryansharma0	200	25000	25200	
2015-16	Ajay Chandel	Kuldeep Chanc	B.Com	5th sem	ajaychandel05	250	2400	2650	
2015-16	Aman Kumar	Suresh Kumar	BA	3rd sem	amankumar01(	150	2000	2150	
2015-16	Amit Sharma	Dinesh Sharma	BSC	5th sem	amitsharma04(	350	2800	3150	
2015-16	Archu Thakur	Rakesh Thakur	BBA	3rd sem	archuthakur02	200	10000	10200	
2015-16	Ashish Gautam	Shashi Gautam	BCA	3rd sem	ashishgautam1	200	25000	25200	
2015-16	Kavita Sharma	Pawan Sharma	B.Com	1st sem	kavitasharma0	250	2400	2650	
2015-16	Neha Sharma	Ashwani Sharn	BSC	5th sem	nehasharma11	350	2800	3150	
2015-16	Prince Thakur	Sanjay Thakur	BA	1st sem	princethakur08	150	2000	2150	
2015-16	Priya Shankhya	Sagar Shankhya	BBA	3rd sem	priyashankhyai	200	10000	10200	
2016-17	Vishal Thakur	Amit Thakur	PGDCA	2nd Sem.	vishalthakur00	200	25000	25200	
*									

Fig. 1.18 STUDENT FEES DETAIL TABLE

STUDENT INFORMATION TABLE

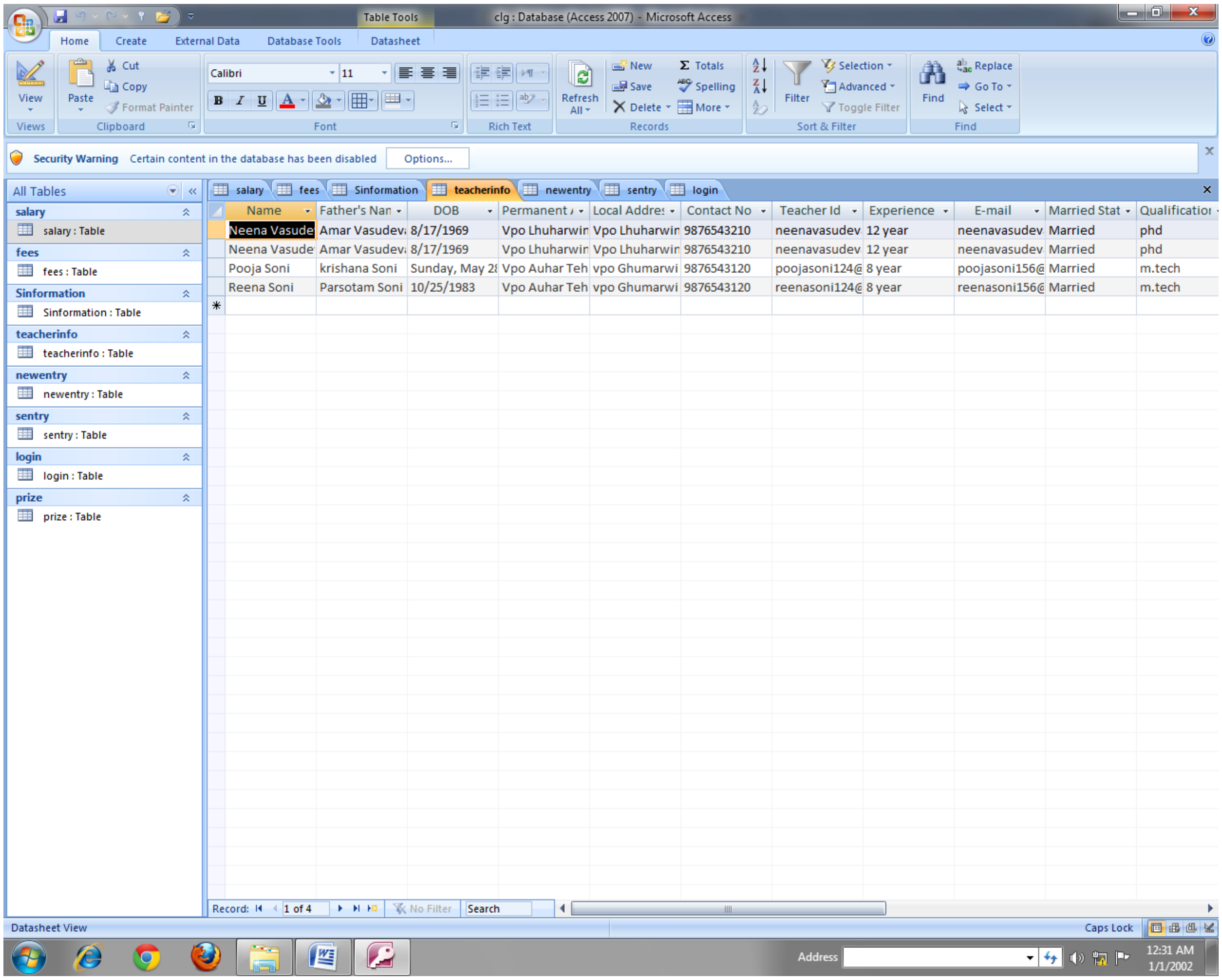


Security Warning Certain content in the database has been disabled Options...

Student ID	Name	DOB	Permanent a	Local addres	Contact no	Percentage1	+2	Graduation	Course	Rollno
1	Deepika Thaku	Sunday, May 28	VPO Lehri Teh	VPO Lehri Teh	9876543210	63	58		BCA	53
2	Mamta	5/30/1995	VPO Jhanduta	VPO Jhanduta	9876543201	73	68		BCA	54
3	Kalpna Kumari	9/18/1995	VPO Baldwara	VPO Baldwara	9876543120	63	60		BCA	52
4	Monika Thakur	12/6/1995	VPO Luharwin	VPO Luharwin	9876542310	62	60		BCA	53
5	Pallvi Rattwan	8/10/1998	VPO Ghumarw	VPO Ghumarw	9876534210	78	87		BSC	55

Fig. 1.19 STUDENT INFORMATION TABLE

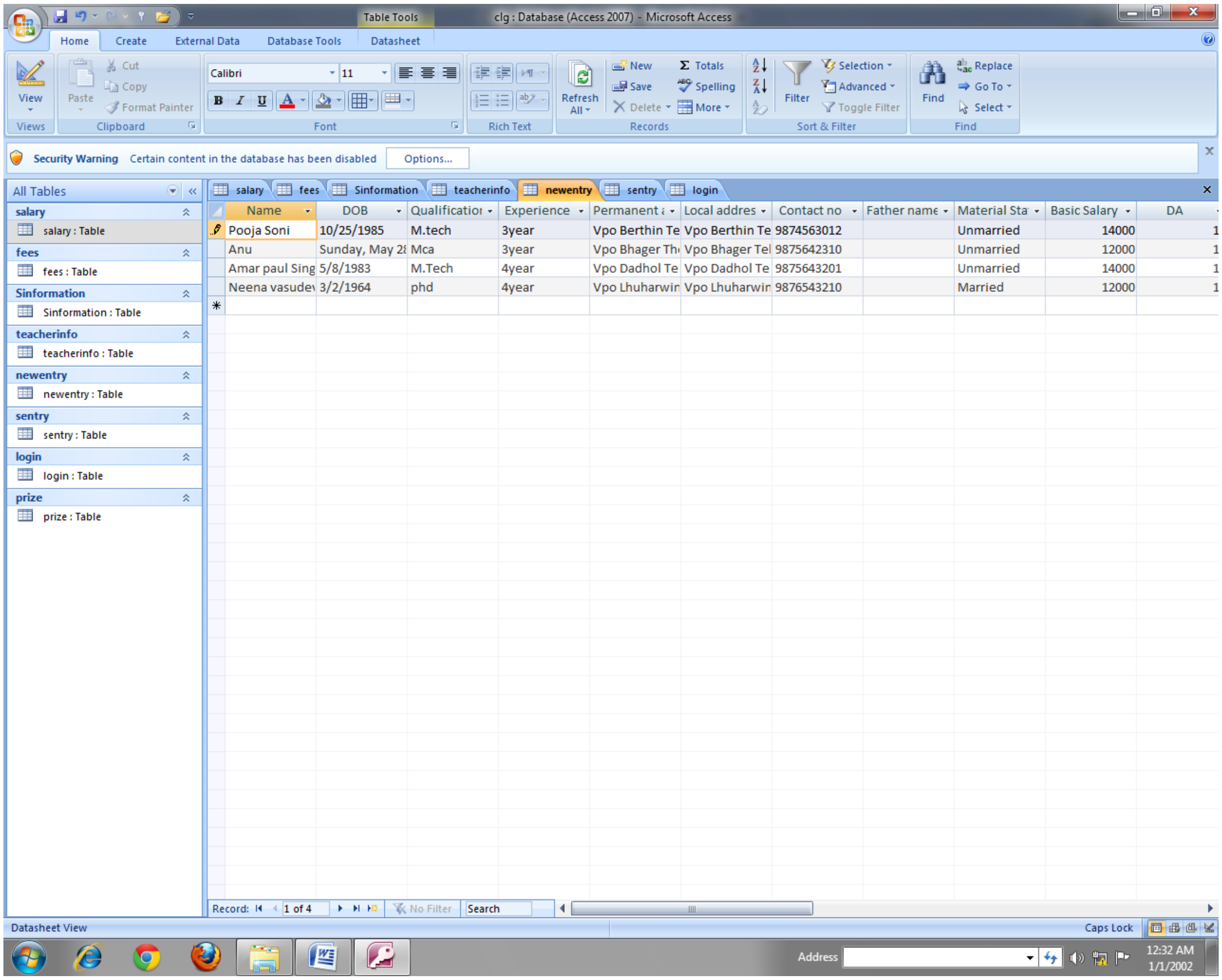
TEACHER INFORMATION TABLE



Name	Father's Nan	DOB	Permanent A	Local Addres	Contact No	Teacher Id	Experience	E-mail	Married Stat	Qualificatior
Neena Vasude	Amar Vasudev:	8/17/1969	Vpo Lhuharwin	Vpo Lhuharwin	9876543210	neenavasudev.	12 year	neenavasudev.	Married	phd
Neena Vasude	Amar Vasudev:	8/17/1969	Vpo Lhuharwin	Vpo Lhuharwin	9876543210	neenavasudev.	12 year	neenavasudev.	Married	phd
Pooja Soni	krishana Soni	Sunday, May 2!	Vpo Auhar Teh	vpo Ghumarwi	9876543120	poojasoni124@	8 year	poojasoni156@	Married	m.tech
Reena Soni	Parsotam Soni	10/25/1983	Vpo Auhar Teh	vpo Ghumarwi	9876543120	reenasoni124@	8 year	reenasoni156@	Married	m.tech

Fig. 1.20 TEACHER INFORMATION TABLE

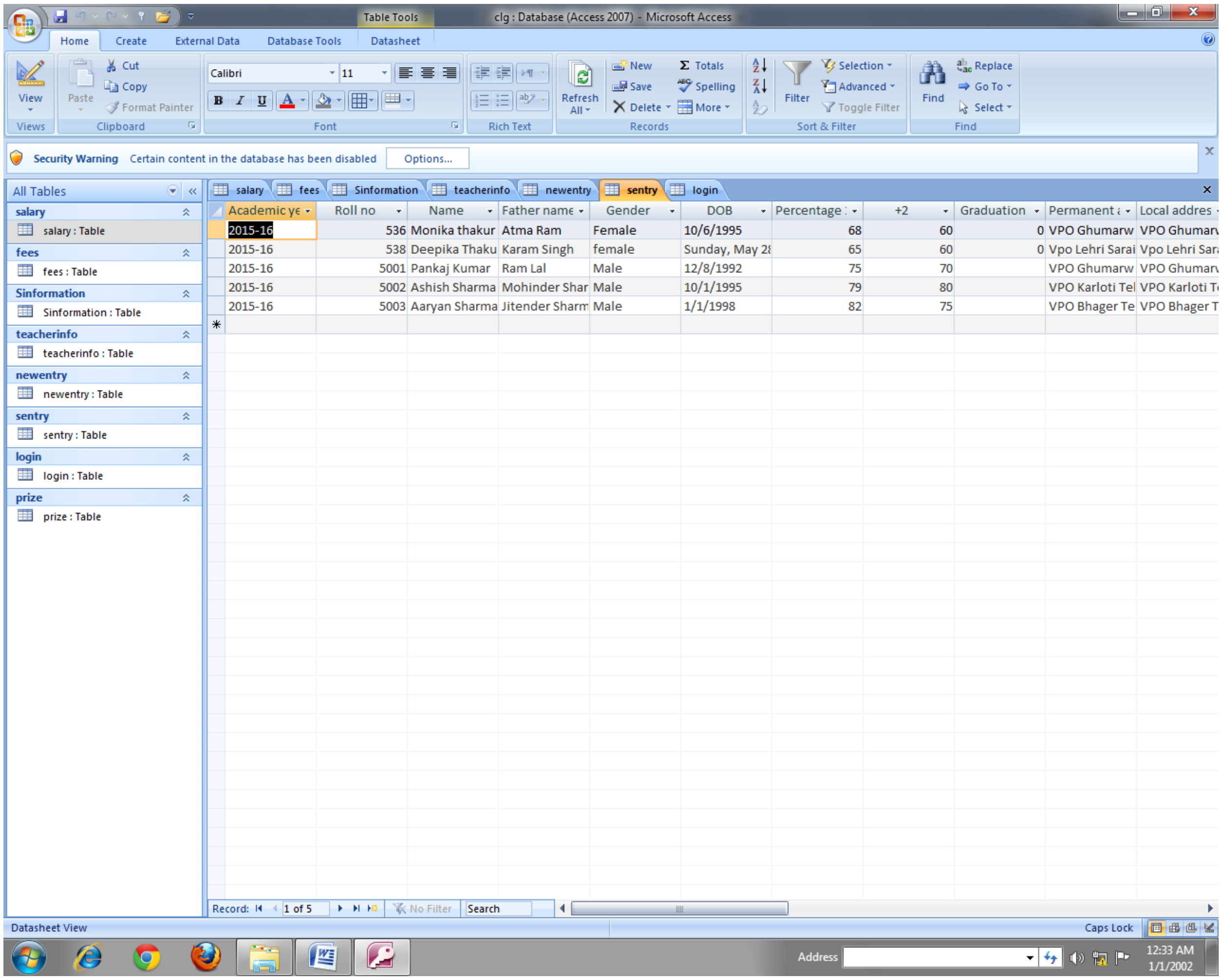
TEACHER ENTRY TABLE



Name	DOB	Qualificatior	Experience	Permanent a	Local addres	Contact no	Father name	Material Sta	Basic Salary	DA
Pooja Soni	10/25/1985	M.tech	3year	Vpo Berthin Te	Vpo Berthin Te	9874563012		Unmarried	14000	1
Anu	Sunday, May 28	Mca	3year	Vpo Bhager Thi	Vpo Bhager Tel	9875642310		Unmarried	12000	1
Amar paul Sing	5/8/1983	M.Tech	4year	Vpo Dadhol Te	Vpo Dadhol Te	9875643201		Unmarried	14000	1
Neena vasudev	3/2/1964	phd	4year	Vpo Lhuharwin	Vpo Lhuharwin	9876543210		Married	12000	1

Fig. 1.21 TEACHER ENTRY TABLE

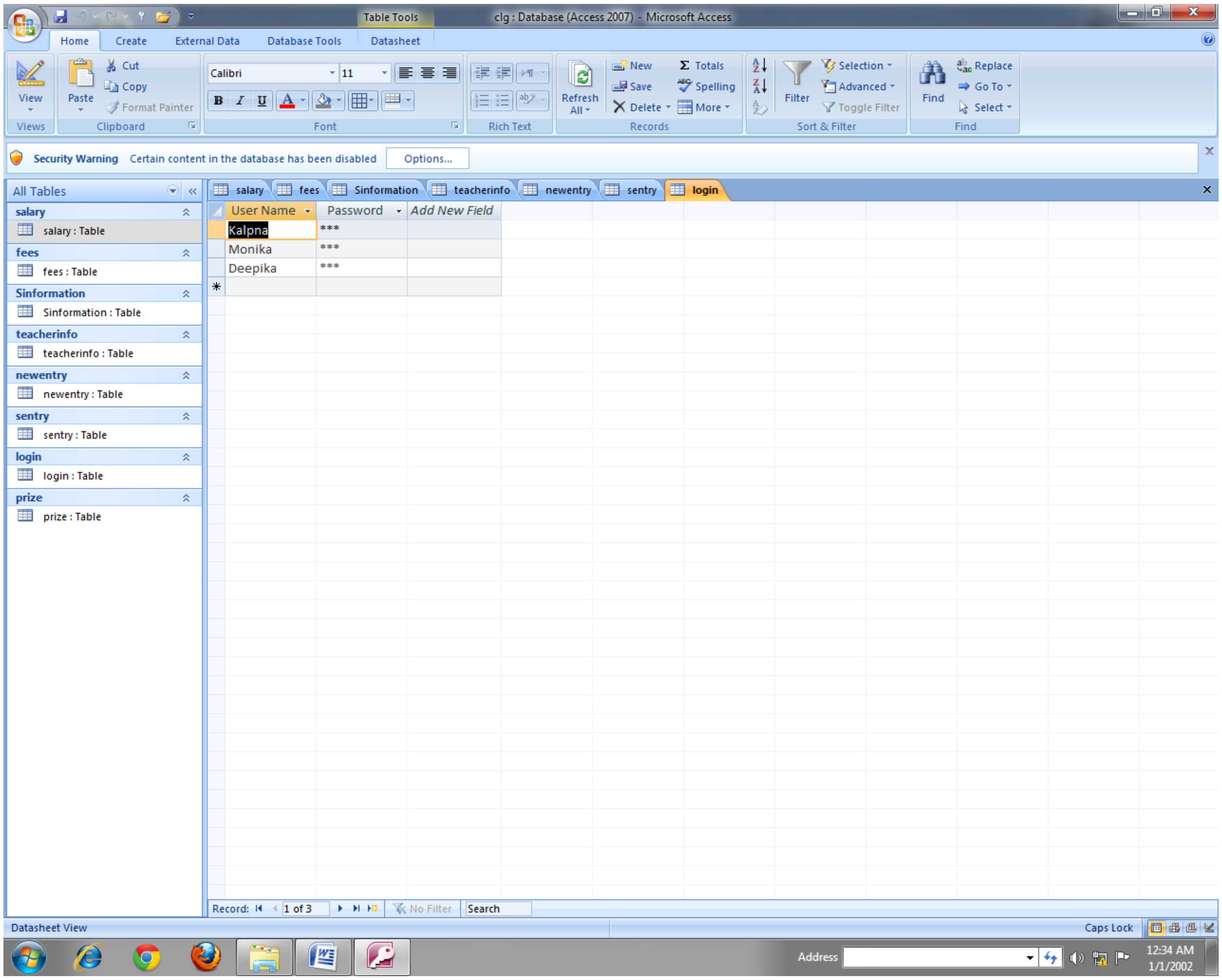
STUDENT ENTRY TABLE



Academic ye	Roll no	Name	Father name	Gender	DOB	Percentage :	+2	Graduation	Permanent a	Local addres
2015-16	536	Monika thakur	Atma Ram	Female	10/6/1995	68	60	0	VPO Ghumarw	VPO Ghumarv
2015-16	538	Deepika Thaku	Karam Singh	female	Sunday, May 28	65	60	0	Vpo Lehri Sarai	Vpo Lehri Sara
2015-16	5001	Pankaj Kumar	Ram Lal	Male	12/8/1992	75	70		VPO Ghumarw	VPO Ghumarv
2015-16	5002	Ashish Sharma	Mohinder Shar	Male	10/1/1995	79	80		VPO Karloti Tel	VPO Karloti Te
2015-16	5003	Aaryan Sharma	Jitender Sharm	Male	1/1/1998	82	75		VPO Bhager Te	VPO Bhager T

Fig. 1.22 STUDENT ENTRY TABLE

LOGIN TABLE



User Name	Password	Add New Field
Kalpna	***	
Monika	***	
Deepika	***	

Fig. 1.23 LOGIN TABLE

Chapter-8

CODING

LOGIN FORM CODING

```
Imports System.Data.OleDb
Public Class Form2
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        If Len(Trim(cmbusername.Text)) = 0 Then
            MessageBox.Show("Please enter user name", "Input Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            cmbusername.Focus()
            Exit Sub
        End If
        If Len(Trim(txtpassword.Text)) = 0 Then
            MessageBox.Show("Please enter password", "Input Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            txtpassword.Focus()
            Exit Sub
        End If
        Try
            Dim myConnection As OleDbConnection
            myConnection = New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;")
            Dim myCommand As OleDbCommand
            myCommand = New OleDbCommand("SELECT login.Username,Password FROM login
where login.Username=login.Username and login.Username = @Username and Password =
@Password", myConnection)

            Dim uName As New OleDbParameter("@Username", SqlDbType.VarChar)
            Dim uPassword As New OleDbParameter("@Password", SqlDbType.VarChar)
            uName.Value = cmbusername.Text
            uPassword.Value = txtpassword.Text
            myCommand.Parameters.Add(uName)

            myCommand.Parameters.Add(uPassword)

            myCommand.Connection.Open()
            Dim myReader As OleDbDataReader =
myCommand.ExecuteReader(CommandBehavior.CloseConnection)

            Dim Login As Object = 0

            If myReader.HasRows Then

                myReader.Read()
                Login = myReader(Login)

            End If

            If Login = Nothing Then

                MsgBox("Login is Failed...Try again !", MsgBoxStyle.Critical, "Login Denied")
                txtpassword.Clear()
            Else
                Me.Hide()
                Form3.Show()
            End If
            myCommand.Dispose()
            myConnection.Close()
        Catch ex As Exception
            MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
        End Try
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        Dim response As MsgBoxResult
        response = MsgBox("Do you want to close form??", MsgBoxStyle.Question +
MsgBoxStyle.YesNo, "confirm")
        If response = MsgBoxResult.Yes Then
            Me.Dispose()
        ElseIf response = MsgBoxResult.No Then
            Exit Sub
        End If

    End Sub
```

HOME FORM CODING

```
Public Class Form3
 Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
    faculity.Show()
    Me.Hide()
  End Sub

  Private Sub Button4_Click(sender As Object, e As EventArgs)
Handles Button4.Click
```

```
    ignou.Show()
    Me.Hide()
  End Sub

  Private Sub PictureBox6_Click(sender As Object, e As EventArgs)
Handles PictureBox6.Click
    Form2.Show()
    Me.Hide()
  End Sub

  Private Sub Button7_Click_1(sender As Object, e As EventArgs)
Handles Button7.Click
    about_us.Show()
    Me.Hide()
  End Sub
 Private Sub Button5_Click(sender As Object, e As EventArgs) Handles
Button5.Click
    Form4.Show()
    Me.Hide()
  End Sub

  Private Sub Button2_Click(sender As Object, e As EventArgs)
Handles Button2.Click
    student_information_form.Show()
    Me.Hide()
  End Sub

  Private Sub Button8_Click(sender As Object, e As EventArgs)
Handles Button8.Click
    contact.Show()
    Me.Hide()
  End Sub

  Private Sub Button9_Click(sender As Object, e As EventArgs)
Handles Button9.Click
    student_admission_form.Show()
    Me.Hide()
  End Sub

  Private Sub Button6_Click(sender As Object, e As EventArgs)
Handles Button6.Click
    teacherinfo.Show()
    Me.Hide()
  End Sub
End Class
```

STUDENT ADMISSION FORM CODING

```
Imports System.Data.OleDb
Public Class student_admission_form
    Dim rdr As OleDbDataReader = Nothing
    Dim dtable As DataTable
    Dim con As OleDbConnection = Nothing
    Dim adp As OleDbDataAdapter
    Dim ds As DataSet
    Dim cmd As OleDbCommand = Nothing
    Dim dt As New DataTable
    Dim gender As String
    Dim cs As String = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;"

Private Sub Button1_Click(ByVal sender As Object, ByVal e As
EventArgs) Handles Button1.Click
        Try
            If Len(Trim(cmbacademic.Text)) = 0 Then
                MessageBox.Show("Please enter Academic year", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbacademic.Focus()
                Exit Sub
            End If
            If Len(Trim(txtname.Text)) = 0 Then
                MessageBox.Show("Please enter student name", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtname.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbcourse.Text)) = 0 Then
                MessageBox.Show("Please enter student course", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbcourse.Focus()
                Exit Sub
            End If
            If Len(Trim(nudtenth.Text)) = 0 Then
                MessageBox.Show("Please enter 10th percentage", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                nudtenth.Focus()
                Exit Sub
            End If
            If Len(Trim(nudplustwo.Text)) = 0 Then
                MessageBox.Show("Please enter 12th percentage", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                nudplustwo.Focus()
                Exit Sub
            End If
            If Len(Trim(rtxtpermanent.Text)) = 0 Then
                MessageBox.Show("Please enter student permanent
address", "Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                rtxtpermanent.Focus()
                Exit Sub
            End If
            If Len(Trim(rtxtlocal.Text)) = 0 Then
                MessageBox.Show("Please enter student local address",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                rtxtlocal.Focus()
                Exit Sub
            End If
            If Len(Trim(txtcontact.Text)) = 0 Then
                MessageBox.Show("Please enter student contact number",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtcontact.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbgender.Text)) = 0 Then
                MessageBox.Show("Please enter student contact number",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbgender.Focus()
                Exit Sub
            End If
            If Len(Trim(txtemail.Text)) = 0 Then
                MessageBox.Show("Please enter student e-mail id",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtemail.Focus()
                Exit Sub
            End If

            If Len(Trim(cmbcategory.Text)) = 0 Then
                MessageBox.Show("Please enter student category",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbcategory.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbyear.Text)) = 0 Then
```

```
                MessageBox.Show("Please enter student year or sem",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbyear.Focus()
                Exit Sub
            End If
            If Len(Trim(txtrollno.Text)) = 0 Then
                MessageBox.Show("Please enter student roll no", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtrollno.Focus()
                Exit Sub
            End If
            If Len(Trim(txtfather.Text)) = 0 Then
                MessageBox.Show("Please enter student father name",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtfather.Focus()
                Exit Sub
            End If
            If Len(Trim(dtpdob.Text)) = 0 Then
                MessageBox.Show("Please enter student date of birth",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                dtpdob.Focus()
                Exit Sub
            End If
            If Len(Trim(nudgraduation.Text)) = 0 Then
                MessageBox.Show("Please enter student graduation",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                nudgraduation.Focus()
                Exit Sub
            End If
            If Len(Trim(txttuition.Text)) = 0 Then
                MessageBox.Show("Please enter student tuition fee",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txttuition.Focus()
                Exit Sub
            End If
            If Len(Trim(txtfee.Text)) = 0 Then
                MessageBox.Show("Please enter student fee amount",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtfee.Focus()
                Exit Sub
            End If
            If Len(Trim(txttotal.Text)) = 0 Then
                MessageBox.Show("Please enter total fee amount",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
```

```
                txttotal.Focus()
                Exit Sub
            End If
            con = New OleDbConnection(cs)
            con.Open()
            Dim cb As String = "INSERT INTO sadmission VALUES('" &
cmbacademic.Text & "','" & txtrollno.Text & "','" & txtname.Text &
"','" & txtfather.Text & "','" & cmbgender.Text & "','" & dtpdob.Text
& "','" & nudtenth.Text & "','" & nudplustwo.Text & "','" &
nudgraduation.Text & "','" & rtxtpermanent.Text & "','" &
rtxtlocal.Text & "','" & txtcontact.Text & "','" & cmbcourse.Text &
"','" & cmbcategory.Text & "','" & txttuition.Text & "','" &
txtfee.Text & "','" & txttotal.Text & "','" & txtemail.Text & "','" &
cmbyear.Text & "')"
            cmd = New OleDbCommand(cb)
            cmd.Connection = con
            cmd.ExecuteNonQuery()
            con.Close()
            MessageBox.Show("Successfully saved", " Student Record",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
        End Try

    End Sub

Private Sub Button3_Click(ByVal sender As Object, ByVal e As
EventArgs) Handles btnexit.Click
        Form3.Show()
        Me.Hide()
    End Sub

    Private Sub Button2_Click(ByVal sender As Object, ByVal e As
EventArgs) Handles btnreset.Click
        txtname.Text = " "
        txtcontact.Text = "  "
        txtemail.Text = "  "
        txtrollno.Text = "  "
        txtfather.Text = "  "
        txttuition.Text = "  "
        txtfee.Text = "  "
        txttotal.Text = "  "
        rtxtpermanent.Text = "  "
        rtxtlocal.Text = "  "
        cmbacademic.Text = "  "
        cmbcourse.Text = "  "
        cmbcategory.Text = "  "
        cmbyear.Text = "  "
        cmbgender.Text = "  "
        nudplustwo.TextAlign = "  "
        nudgraduation.TextAlign = "  "
        nudtenth.TextAlign = "  "
    End Sub

Private Sub txtcontact_VALIDATING(sender As Object, e As EventArgs)
Handles txtcontact.Validating
        If (txtcontact.TextLength > 10) Then
            MessageBox.Show("Only 10 digits are allowed", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
            txtcontact.Focus()
        End If
    End Sub

    Private Sub txtemail_VALIDATING(ByVal sender As System.Object,
ByVal e As System.ComponentModel.CancelEventArgs) Handles
txtemail.Validating

        Dim rEMail As New System.Text.RegularExpressions.Regex("^[a-
zA-Z][\w\.-]{2,28}[a-zA-Z0-9]@[a-zA-Z0-9][\w\.-]*[a-zA-Z0-9]\.[a-zA-
Z][a-zA-Z\.]*[a-zA-Z]$")
        If txtemail.Text.Length > 0 Then
            If Not rEMail.IsMatch(txtemail.Text) Then
                MessageBox.Show("invalid email address", "Error",
MessageBoxButtons.OK, MessageBoxIcon.[Error])
                txtemail.SelectAll()
                e.Cancel = True
            End If
        End If
    End Sub
End Class
```

NEW PROFESSOR ENTERYFORM CODING

```
Imports System.Data.OleDb
Imports System.IO
Imports System.Security.Cryptography
Imports System.Text
Public Class Teacher_New_Entry
```

```
    Dim rdr As OleDbDataReader = Nothing
    Dim dtable As DataTable
    Dim con As OleDbConnection = Nothing
    Dim adp As OleDbDataAdapter
    Dim ds As DataSet
    Dim cmd As OleDbCommand = Nothing
    Dim dt As New DataTable
    Dim gender As String
    Dim cs As String = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;"

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Try
            If Len(Trim(txtname.Text)) = 0 Then
                MessageBox.Show("Please enter Teacher Name", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtname.Focus()
                Exit Sub
            End If
            If Len(Trim(txtfather.Text)) = 0 Then
                MessageBox.Show("Please enter Teacher Father name",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtfather.Focus()
                Exit Sub
            End If
            If Len(Trim(dtpdob.Text)) = 0 Then
                MessageBox.Show("Please enter teacher date of birth",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                dtpdob.Focus()
                Exit Sub
            End If
            If Len(Trim(rtxtpermanent.Text)) = 0 Then
                MessageBox.Show("Please enter permanent address",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                rtxtpermanent.Focus()
                Exit Sub
            End If
            If Len(Trim(rtxtlocal.Text)) = 0 Then
                MessageBox.Show("Please enter local address", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                rtxtlocal.Focus()
                Exit Sub
            End If
            If Len(Trim(txtcontact.Text)) = 0 Then
                MessageBox.Show("Please enter contact number", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtcontact.Focus()
                Exit Sub
            End If
            If Len(Trim(txtexperience.Text)) = 0 Then
                MessageBox.Show("Please enter teacher experience",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtexperience.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbmarried.Text)) = 0 Then
                MessageBox.Show("Please enter teacher married status",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbmarried.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbgender.Text)) = 0 Then
                MessageBox.Show("Please enter student gender", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbgender.Focus()
                Exit Sub
            End If
            If Len(Trim(txtqualification.Text)) = 0 Then
                MessageBox.Show("Please enter the teacher's
qualification", "Input Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
                txtqualification.Focus()
                Exit Sub
            End If

            If Len(Trim(txtbasic.Text)) = 0 Then
                MessageBox.Show("Please enter basic salary", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtbasic.Focus()
                Exit Sub
            End If
            If Len(Trim(txtda.Text)) = 0 Then
                MessageBox.Show("Please enter DA", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtda.Focus()
                Exit Sub
            End If
            If Len(Trim(txthra.Text)) = 0 Then
                MessageBox.Show("Please enter HRA", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                txthra.Focus()
                Exit Sub
            End If
            If Len(Trim(txtgross.Text)) = 0 Then
                MessageBox.Show("Please enter GROSS PAY", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtgross.Focus()
                Exit Sub
            End If
            If Len(Trim(txtdiction.Text)) = 0 Then
                MessageBox.Show("Please enter TEACHER OTHER DICTION",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtdiction.Focus()
                Exit Sub
            End If
            If Len(Trim(txtpf.Text)) = 0 Then
                MessageBox.Show("Please enter PF", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtpf.Focus()
                Exit Sub
            End If
            If Len(Trim(txttotal.Text)) = 0 Then
                MessageBox.Show("Please enter TOTAL SALARY", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txttotal.Focus()
                Exit Sub
            End If

            con = New OleDbConnection(cs)
            con.Open()
            Dim cb As String = "INSERT INTO tnerentry VALUES('" &
txtname.Text & "','" & dtpdob.Text & "','" & txtqualification.Text &
"','" & txtexperience.Text & "','" & rtxtpermanent.Text & "','" &
rtxtlocal.Text & "','" & txtcontact.Text & "','" & txtfather.Text &
"','" & cmbmarried.Text & "','" & txtbasic.Text & "','" & txtda.Text &
"','" & txthra.Text & "','" & txtgross.Text & "','" & txtdiction.Text
& "','" & txtpf.Text & "','" & txttotal.Text & "','" & cmbgender.Text
& "')"
            cmd = New OleDbCommand(cb)
            cmd.Connection = con
            cmd.ExecuteNonQuery()
            con.Close()
            MessageBox.Show("Successfully saved", "Teacher record",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
        End Try

    End Sub

    Private Sub btnreset_Click(sender As Object, e As EventArgs)
Handles btnreset.Click
        txtname.Text = " "
        txtcontact.Text = "  "
        dtpdob.Text = "  "
        txtqualification.Text = "  "
        txtfather.Text = "  "
        txtexperience.Text = "  "
        txtbasic.Text = "  "
        txttotal.Text = "  "
        rtxtpermanent.Text = "  "
        rtxtlocal.Text = "  "
        txtda.Text = "  "
        cmbgender.Text = "  "
        cmbmarried.Text = "  "
        txthra.Text = "  "
        txtgross.Text = "  "
        txtdiction.Text = "  "
        txtpf.Text = "  "

    End Sub

    Private Sub btnexit_Click(sender As Object, e As EventArgs)
Handles btnexit.Click
        Form3.Show()
        Me.Hide()
    End Sub
End Class
```

STUDENT INFORMATION FORM CODING

```
Imports System.Data.OleDb
Public Class student_information_form
    Dim con As OleDbConnection
    Dim cmd As OleDbCommand
    Dim dr As OleDbDataReader
    Dim str As String
    Dim cc As Integer
    Dim cs As String = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;"

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        If ComboBox5.Text = "" Then
            MsgBox("Select Student_ID from Student ID field",
MsgBoxStyle.Information, "Edit Student_ID")
        Else
            Try
                con = New OleDbConnection(cs)
                con.Open()
                str = "Update studentinfo Set Contact_No = '" &
TextBox3.Text & "',Local_Address = '" & RichTextBox2.Text & "' where
Student_ID ='" & ComboBox5.Text & "'"
                cmd = New OleDbCommand(str, con)
                cmd.ExecuteNonQuery()
                MsgBox("Record updated Successfully.",
MsgBoxStyle.Information, "Edit Registration")
            Catch
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End If
        con.Close()
    End Sub

    Private Sub Button6_Click(sender As Object, e As EventArgs)
Handles Button6.Click
        Form3.Show()
        Me.Hide()
    End Sub
#Region "CLEARALL"
    Private Sub clearall()
        ComboBox1.Text = ""
        ComboBox2.Text = "   "
        ComboBox3.Text = "  "
        ComboBox4.Text = "  "
        RichTextBox1.Text = "  "
        RichTextBox2.Text = "  "
        DateTimePicker1.Text = "  "
        NumericUpDown1.TextAlign = "  "
        NumericUpDown2.TextAlign = "  "
        NumericUpDown3.TextAlign = "  '"
        ComboBox5.Text = ""
        ComboBox1.Items.Clear()
        TextBox2.Text = ""
        TextBox3.Text = ""
        TextBox4.Text = ""
        TextBox5.Text = "  "
        TextBox6.Text = "  "
        ComboBox5.Refresh()
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select Student_ID From
studentinfo", con)
            dr = cmd.ExecuteReader
            While dr.Read()
                ComboBox5.Items().Add(dr("STUDENT_ID"))
            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub
#End Region

    Private Sub Button1_Click(sender As Object, e As EventArgs)
Handles Button1.Click
        Dim choice As Integer
        choice = MsgBox("Are you sure you want to remove the record",
MsgBoxStyle.OkCancel)
        If (choice = 1) Then
            Try
                con = New OleDbConnection(cs)
                con.Open()
                cmd = New OleDbCommand("Delete * From studentinfo
where Student_ID = '" & ComboBox5.Text & "'", con)
                dr = cmd.ExecuteReader

            Catch
```

```
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
            con.Close()
            clearall()

        Else
            Focus()
        End If
    End Sub

    Private Sub student_information_form_Load(sender As Object, e As
EventArgs) Handles MyBase.Load
        Try
            con = New OleDbConnection(cs)
            con.Open()

            cmd = New OleDbCommand("select Student_ID From
studentinfo", con)
            dr = cmd.ExecuteReader

            While dr.Read()
                ComboBox5.Items().Add(dr("Student_ID"))

            End While

        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try

        con.Close()
    End Sub


    Private Sub ComboBox5_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles ComboBox5.SelectedIndexChanged
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select * From studentinfo where
Student_ID = '" & ComboBox5.Text & "'", con)
            dr = cmd.ExecuteReader
            While dr.Read()
```

```
                TextBox2.Text = dr("Name")
                TextBox3.Text = dr("ContactNo")
                TextBox4.Text = dr("RollNo")
                TextBox5.Text = dr("FatherName")
                TextBox6.Text = dr("e-mail")
                RichTextBox1.Text = dr("PermanentAddress")
                RichTextBox2.Text = dr("LocalAddress")
                DateTimePicker1.Text = dr("DOB")
                NumericUpDown1.Text = dr("Percentage10th")
                NumericUpDown2.Text = dr("Percentage12th")
                NumericUpDown3.Text = dr("Graduation")
                ComboBox1.Text = dr("Course")
                ComboBox2.Text = dr("Category")
                ComboBox3.Text = dr("YeraSem")
                ComboBox4.Text = dr("AcademicYear")
            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub
End Class
```

TEACHER INFORMATION FORM CODING

```
Imports System.Data.OleDb
Public Class teacherinfo
    Dim con As OleDbConnection
    Dim cmd As OleDbCommand
    Dim dr As OleDbDataReader
    Dim str As String
    Dim cc As Integer
    Dim cs As String = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;"

Private Sub Button8_Click(sender As Object, e As EventArgs) Handles
Button8.Click
        Form3.Show()
        Me.Hide()
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs)
Handles Button3.Click
```

```
        teachersalary.Show()
        Me.Hide()
    End Sub

    Private Sub Button9_Click(sender As Object, e As EventArgs)
Handles Button9.Click
        Teacher_New_Entry.Show()
        Me.Hide()
    End Sub

  Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        If ComboBox3.Text = "" Then
            MsgBox("Select Teacher_id from Teacher ID field",
MsgBoxStyle.Information, "Edit Teacher_id")
        Else
            Try
                con = New OleDbConnection(cs)
                con.Open()
                str = "Update teacherinfo Set Contact_No = '" &
TextBox3.Text & "',Local_Address = '" & RichTextBox2.Text & "' where
Teacher_id ='" & ComboBox3.Text & "'"
                cmd = New OleDbCommand(str, con)
                cmd.ExecuteNonQuery()
                MsgBox("Record updated Successfully.",
MsgBoxStyle.Information, "Update Registration")
            Catch
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End If
        con.Close()
    End Sub
#Region "CLEARALL"
    Private Sub clearall()
        ComboBox1.Text = ""
        ComboBox2.Text = "   "
        ComboBox3.Text = "  "
        ComboBox3.Items.Clear()
        RichTextBox1.Text = "  "
        RichTextBox2.Text = "  "
        DateTimePicker1.Text = "  "
        TextBox2.Text = ""
        TextBox3.Text = ""
        TextBox5.Text = "  "
        TextBox6.Text = "  "
        TextBox1.Text = "  "
        TextBox7.Text = "  "
        ComboBox3.Refresh()
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select Teacher_id From
teacherinfo", con)
            dr = cmd.ExecuteReader
            While dr.Read()
                ComboBox3.Items().Add(dr("Teacher_id"))
            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub
#End Region

    Private Sub Button4_Click(sender As Object, e As EventArgs)
Handles Button4.Click
        Dim choice As Integer
        choice = MsgBox("Are you sure you want to remove the record",
MsgBoxStyle.OkCancel)
        If (choice = 1) Then
            Try
                con = New OleDbConnection(cs)
                con.Open()
                cmd = New OleDbCommand("Delete * From teacherinfo
where Teacher_id = '" & ComboBox3.Text & "'", con)
                dr = cmd.ExecuteReader

            Catch
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
            con.Close()
            clearall()

        Else
            Focus()
        End If
    End Sub

    Private Sub teacherinfo_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
        Try
            con = New OleDbConnection(cs)
            con.Open()

            cmd = New OleDbCommand("select Teacher_id From
teacherinfo", con)
            dr = cmd.ExecuteReader

            While dr.Read()
                ComboBox3.Items().Add(dr("Teacher_id"))

            End While

        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try

        con.Close()
    End Sub

    Private Sub ComboBox3_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles ComboBox3.SelectedIndexChanged
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select * From teacherinfo where
Teacher_id = '" & ComboBox3.Text & "'", con)
            dr = cmd.ExecuteReader
            While dr.Read()
                TextBox2.Text = dr("Name")
                TextBox1.Text = dr("father_name")
                TextBox3.Text = dr("Contact_No")
                TextBox5.Text = dr("E-mail")
                TextBox6.Text = dr("experience")
                TextBox7.Text = dr("Qualification")
                RichTextBox1.Text = dr("Permanent_Address")
                RichTextBox2.Text = dr("Local_Address")
                ComboBox1.Text = dr("Married Status")
                ComboBox2.Text = dr("Gender")
                DateTimePicker1.Text = dr("DOB")
            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub

    Private Sub Button6_Click(sender As Object, e As EventArgs)
Handles Button6.Click
        If ComboBox3.Text = "" Then
            MsgBox("Select Teacher_id from Teacher ID field",
MsgBoxStyle.Information, "Edit Teacher_id")
        Else
            Try
                con = New OleDbConnection(cs)
                con.Open()
                str = "Select * from teacherinfo where Teacher_id= '"
& ComboBox3.Text & "'"
                cmd = New OleDbCommand(str, con)
                cmd.ExecuteNonQuery()
                MsgBox("Record show Successfully.",
MsgBoxStyle.Information, "show record")
            Catch
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End If
        con.Close()
    End Sub
End Class
```

FEES DETAIL FORM CODING

```
Imports System.Data.OleDb
Public Class studentfee
    Dim con As OleDbConnection
    Dim cmd As OleDbCommand
    Dim dr As OleDbDataReader
    Dim str As String
    Dim cc As Integer
    Dim cs As String = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;"

Private Sub studentfee_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Try
            con = New OleDbConnection(cs)
            con.Open()

            cmd = New OleDbCommand("select Student_ID From sfees",
con)
            dr = cmd.ExecuteReader

            While dr.Read()
                cmbid.Items().Add(dr("Student_ID"))

            End While

        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try

        con.Close()
    End Sub

Private Sub Button1_Click_1(sender As Object, e As EventArgs) Handles
Button1.Click
        If cmbid.Text = "" Then
            MsgBox("Select Student_ID from Student ID field",
MsgBoxStyle.Information, "Edit Student_ID")
        Else
            Try
                con = New OleDbConnection(cs)
                con.Open()
                str = "Update sfees Set Tuition_Fee = '" &
txttuition.Text & "',Fee_Amount = '" & cmbfee.Text & "',Total_Fee = '"
& cmbtotal.Text & "' where Student_ID ='" & cmbid.Text & "'"
                cmd = New OleDbCommand(str, con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Fee updated Successfully.",
MsgBoxStyle.Information, "Update Student Fee")
            Catch
```

```
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End If
        con.Close()
    End Sub
#Region "CLEARALL"
    Private Sub clearall()
        cmbacademic.Text = ""
        cmbcourse.Text = "   "
        cmbyear.Text = "  "
        cmbid.Text = "  "
        cmbid.Items.Clear()
        txtfather.Text = ""
        txttuition.Text = "  "
        cmbfee.Text = "  "
        cmbtotal.Text = "  "
        txtname.Text = "  "
        cmbid.Refresh()
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select Student_ID From sfees",
con)
            dr = cmd.ExecuteReader
            While dr.Read()
                cmbid.Items().Add(dr("Student_ID"))
            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub
#End Region
    Private Sub Button6_Click_1(sender As Object, e As EventArgs)
Handles Button6.Click
        Dim choice As Integer
        choice = MsgBox("Are you sure you want to remove the record",
MsgBoxStyle.OkCancel)
        If (choice = 1) Then
            Try
                con = New OleDbConnection(cs)
                con.Open()
        cmd = New OleDbCommand("Delete * From sfees where
Student_ID = '" & cmbid.Text & "'", con)
        dr = cmd.ExecuteReader

      Catch
        MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
      End Try
      con.Close()
      clearall()

    Else
      Focus()
    End If
  End Sub
 Private Sub cmbid_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles cmbid.SelectedIndexChanged
    Try
      con = New OleDbConnection(cs)
      con.Open()
      cmd = New OleDbCommand("select * From sfees where
Student_ID = '" & cmbid.Text & "'", con)
      dr = cmd.ExecuteReader
      While dr.Read()
        txtfather.Text = dr("Name")
        txtname.Text = dr("Father_name")
        cmbfee.Text = dr("Fee_Amount")
        cmbtotal.Text = dr("Total_Fee")
        cmbacademic.Text = dr("Academic_Year")
        cmbcourse.Text = dr("Course")
        cmbyear.Text = dr("Year/Sem")
        txttuition.Text = dr("Tuition_Fee")
      End While
    Catch
      MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
    con.Close()
  End Sub


  Private Sub Label10_Click(sender As Object, e As EventArgs)
Handles Label10.Click
    Dim a, b, c As Integer
```

```
        a = Val(txttuition.Text)
        b = Val(cmbfee.Text)
        c = a + b
        cmbtotal.Text = c
    End Sub

    Private Sub cmbtotal_click(sender As Object, e As EventArgs)
Handles cmbtotal.TextChanged

    End Sub

    Private Sub Button5_Click_1(sender As Object, e As EventArgs)
Handles Button5.Click
        Try
            If Len(Trim(cmbid.Text)) = 0 Then
                MessageBox.Show("Please enter student id", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbid.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbacademic.Text)) = 0 Then
                MessageBox.Show("Please enter academic year", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbacademic.Focus()
                Exit Sub
            End If
            If Len(Trim(txtname.Text)) = 0 Then
                MessageBox.Show("Please enter student name", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtname.Focus()
                Exit Sub
            End If
            If Len(Trim(txtfather.Text)) = 0 Then
                MessageBox.Show("Please enter father name", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtfather.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbcourse.Text)) = 0 Then
                MessageBox.Show("Please enter course", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbcourse.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbyear.Text)) = 0 Then
                MessageBox.Show("Please enter year or sem", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbyear.Focus()
                Exit Sub
            End If
            If Len(Trim(txttuition.Text)) = 0 Then
                MessageBox.Show("Please enter tuition fee", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txttuition.Focus()
                Exit Sub
            End If
            If Len(Trim(cmbfee.Text)) = 0 Then
                MessageBox.Show("Please enter fee amount", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbfee.Focus()
                Exit Sub
            End If
            con = New OleDbConnection(cs)
            con.Open()
            Dim cb As String = "INSERT INTO sfees VALUES('" &
cmbacademic.Text & "','" & txtname.Text & "','" & txtfather.Text &
"','" & cmbcourse.Text & "','" & cmbyear.Text & "','" & cmbid.Text &
"','" & txttuition.Text & "','" & cmbfee.Text & "','" & cmbtotal.Text
& "')"
            cmd = New OleDbCommand(cb)
            cmd.Connection = con
            cmd.ExecuteNonQuery()
            con.Close()
            MessageBox.Show("Successfully saved", " teacher salary
Record", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
        End Try
    End Sub

    Private Sub Button7_Click_1(sender As Object, e As EventArgs)
Handles Button7.Click
        student_information_form.Show()
        Me.Hide()
    End Sub
End Class
```

SALARY DETAIL FORM CODING

```
Imports System.Data.OleDb
Public Class teachersalary
    Dim con As OleDbConnection
    Dim cmd As OleDbCommand
    Dim dr As OleDbDataReader
    Dim str As String
    Dim cc As Integer
    Dim cs As String = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\college.accdb;Persist Security Info=False;"

Private Sub Button9_Click(sender As Object, e As EventArgs) Handles
Button9.Click
        Dim basic, da, hra, pf, netpay, gross, total As Integer
        basic = Val(txtbasic.Text)
        da = Val(txtda.Text)
        hra = Val(txthra.Text)
        netpay = Val(txtnet.Text)
        pf = basic + (basic * (da / 100) + basic * (hra / 100))
        gross = basic * (12 / 100)
        total = pf - (netpay + gross)
        txtpf.Text = pf
        txtgross.Text = gross
        txttotal.Text = total
    End Sub

    Private Sub Button8_Click(sender As Object, e As EventArgs)
Handles Button8.Click
        teacherinfo.Show()
        Me.Hide()
    End Sub

    Private Sub Button6_Click(sender As Object, e As EventArgs)
Handles Button6.Click
        If cmbid.Text = "" Then
            MsgBox("Select teacher_id from teacher ID field",
MsgBoxStyle.Information, "Edit teacher_id")
        Else
            Try
                con = New OleDbConnection(cs)
                con.Open()
                str = "Update tsalary Set Phone_No = '" &
txtphone.Text & "',Basic_Salary = '" & txtbasic.Text & "',DA = '" &
```

```
txtda.Text & "',HRA = '" & txthra.Text & "',PF = '" & txtpf.Text &
"',Net_Pay = '" & txtnet.Text & "',Gross_Salary= '" & txtgross.Text &
"',Total_Salary = '" & txttotal.Text & "' where teacher_id ='" &
cmbid.Text & "'"
                cmd = New OleDbCommand(str, con)
                cmd.ExecuteNonQuery()
                MsgBox("teacher salary updated Successfully.",
MsgBoxStyle.Information, "Update teacher salary")
            Catch
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End If
        con.Close()
    End Sub

    Private Sub teachersalary_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
        Try
            con = New OleDbConnection(cs)
            con.Open()

            cmd = New OleDbCommand("select teacher_id From tsalary",
con)
            dr = cmd.ExecuteReader

            While dr.Read()
                cmbid.Items().Add(dr("teacher_id"))

            End While

        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub

    Private Sub Button7_Click(sender As Object, e As EventArgs)
Handles Button7.Click
        Dim choice As Integer
        choice = MsgBox("Are you sure you want to remove the record",
MsgBoxStyle.OkCancel)
        If (choice = 1) Then
```

```
            Try
                con = New OleDbConnection(cs)
                con.Open()
                cmd = New OleDbCommand("Delete * From tsalary where
teacher_id = '" & cmbid.Text & "'", con)
                dr = cmd.ExecuteReader

            Catch
                MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
            con.Close()
            clearall()

        Else
            Focus()
        End If
    End Sub
#Region "CLEARALL"
    Private Sub clearall()
        txtname.Text = ""
        txtphone.Text = "   "
        txtbasic.Text = "  "
        txtda.Text = "  "
        cmbid.Text = ""
        cmbid.Items.Clear()
        txthra.Text = ""
        txtpf.Text = "  "
        txtnet.Text = "  "
        txtgross.Text = "  "
        txttotal.Text = "  "
        cmbid.Refresh()
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select teacher_id From tsalary",
con)
            dr = cmd.ExecuteReader
            While dr.Read()
                cmbid.Items().Add(dr("teacher_id"))
            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
```

```
        End Try
        con.Close()
    End Sub
#End Region

    Private Sub cmbid_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles cmbid.SelectedIndexChanged
        Try
            con = New OleDbConnection(cs)
            con.Open()
            cmd = New OleDbCommand("select * From tsalary where
teacher_id = '" & cmbid.Text & "'", con)
            dr = cmd.ExecuteReader
            While dr.Read()
                txtname.Text = dr("Name")
                txtphone.Text = dr("phone_no")
                txtbasic.Text = dr("basic_salary")
                txtda.Text = dr("da")
                txthra.Text = dr("hra")
                txtpf.Text = dr("pf")
                txtnet.Text = dr("net_pay")
                txtgross.Text = dr("gross_salary")
                txttotal.Text = dr("total_salary")

            End While
        Catch
            MessageBox.Show(Err.Description, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
        con.Close()
    End Sub

    Private Sub Button5_Click(sender As Object, e As EventArgs)
Handles Button5.Click
        Try
            If Len(Trim(cmbid.Text)) = 0 Then
                MessageBox.Show("Please enter teacher id", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                cmbid.Focus()
                Exit Sub
            End If
            If Len(Trim(txtname.Text)) = 0 Then
                MessageBox.Show("Please enter teacher name", "Input
Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
```

```
                txtname.Focus()
                Exit Sub
            End If
            If Len(Trim(txtphone.Text)) = 0 Then
                MessageBox.Show("Please enter teacher phone_no",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtphone.Focus()
                Exit Sub
            End If
            If Len(Trim(txtbasic.Text)) = 0 Then
                MessageBox.Show("Please enter teacher's basic salary",
"Input Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtbasic.Focus()
                Exit Sub
            End If
            If Len(Trim(txtda.Text)) = 0 Then
                MessageBox.Show("Please enter DA", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtda.Focus()
                Exit Sub
            End If
            If Len(Trim(txthra.Text)) = 0 Then
                MessageBox.Show("Please enter HRA", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                txthra.Focus()
                Exit Sub
            End If
            If Len(Trim(txtnet.Text)) = 0 Then
                MessageBox.Show("Please enter NET PAY", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                txtnet.Focus()
                Exit Sub
            End If
            con = New OleDbConnection(cs)
            con.Open()
            Dim cb As String = "INSERT INTO tsalary VALUES('" &
cmbid.Text & "','" & txtname.Text & "','" & txtphone.Text & "','" &
txtbasic.Text & "','" & txtda.Text & "','" & txthra.Text & "','" &
txtpf.Text & "','" & txtnet.Text & "','" & txtgross.Text & "','" &
txttotal.Text & "')"
            cmd = New OleDbCommand(cb)
            cmd.Connection = con
            cmd.ExecuteNonQuery()
            con.Close()
            MessageBox.Show("Successfully saved", " teacher salary
Record", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
        End Try
    End Sub
End Class
```

Chapter-9

POST IMPLEMENTATION AND MAINTAINENCE

After the installation phase is complete and user staff is adjusted to the changes created by candidate system, evaluation and maintain begins. Like any system there is an aging process that requires periodical of hardware and software. If the new information is in consistent with the design specification when changes have to be made. The important of it is to bring new system to standards. This is the last step in the system development life cycle.

9.1 EXISTING SYSTEM

Whenever we implement new system it is developed to remove the shortcomings of an existing system. The computerized has more
Edge over the manual system. As we are doing a project on
"COLLEGE MANAGEMENT". So firstly we will introduce the existing system, the existing system is based on manual system, which takes lot of time to get performance of the test. It has the following disadvantages:

Wastage of time: -As the test is conducted manually it takes lot of time to attempt question of that particular test.

Inaccuracy: -While conducting the test papers manually it can happen that questions papers are not printed properly, in accuracy can also take place while checking the answer sheets and the teachers may mistake while calculating marks. All the above leads to the problem of inaccuracy.

Late declaration of result:-In the existing system answer sheets are checked manually so it takes a lot of time while checking the answer sheets. So the result is declared late.

Wastage of money: -While checking the different answer sheets different professor is hired. So it requires a lot of money against the checking of answer sheets.

Chances of Partiality:-In the existing system there are chances of partiality while checking the answer sheet of particular student. This is because some professor may be biased towards some student. So there are chances of partiality .Due to all the above problem of manual system the new computerized system is developed.

9.2 PROPOSED SYSTEM

We have developed new system, which is based on computer in which the student used to give test on the computer by simply clicking the answers. It is most reliable & time saving system while conducting some entrances test.

Time Saving:-As the test is conducted on computer it takes very less time for the student to attempt the question by simply clicking on the answer which they thinks to be right.

Accuracy:-Unlike the manual system there is no in accuracy while checking the answer of a question. Since it does not make any calculation mistake while at the same time of checking.

Quick Declaration of Result:-In the computerized system the results are declared in less time i.e. within the fraction of minutes. So we can say that in computer system are more efficient then the existing system from the point of view of declaring the result

Money Saving:- Unlike the existing the computerized system does not require the any professor to check the answer sheet so this saves the lot of money.

Reduces the Chances of Partiality:-As the answer s is checked by the computer itself so there are no chances of partiality while checking the answers.

9.3 STRUCTURED ANALYSIS

Structured analysis is the set of techniques and graphical tool that allow the analyst to develop a new kind of system specification that are easily understandable to the users. It way to on function rather than physical implementation. The traditional approach focuses on cost and benefit analysis, feasibility analysis, project management, hardware and software section and personal consideration in contrast, structured considers new goals and structured tools for analysis. There are different tools structured system analyses; some of them are as follows:

- DATA FLOW DIAGRAM
- DATA DICTIONARY
- STRUCTURED ANALYSIS
- DECISION TREE

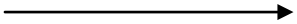
9.4 DATA FLOW DIAGRAM

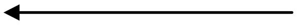
The DFD was first developed by Larry Constantine as a way of expressing system requirement in a graphical form. A DFD also known as bubble chart has a purpose of clarifying system requirement and identifying major transformation that will become the program in the system design.

DFD SYMBOL:-

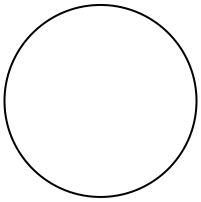
- Square: -A square defines a source or destination of system data.

- Arrow:-An arrow identifies data or data in motion; it is a pipeline through which information flows.

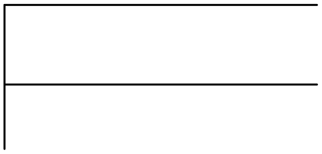




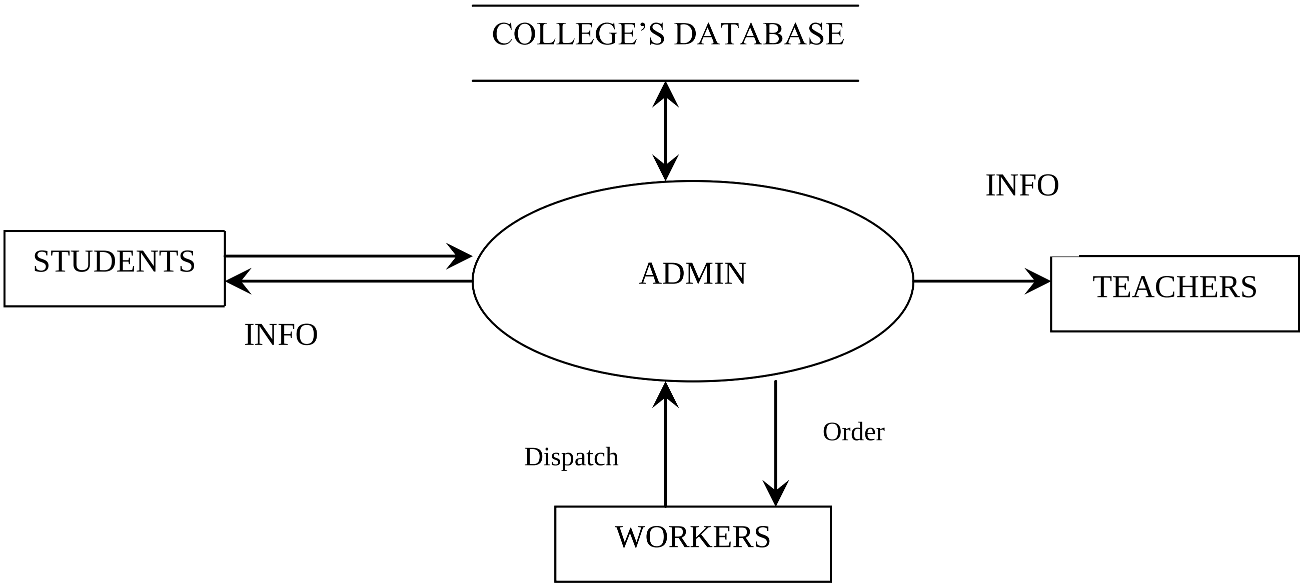
- Circle or Bubble:-It represents a process transforms in coming in data flow in to outgoing data flow.



- Open rectangle:-It is data store or data at rest.



DATA FLOW DIAGRAM OF THE COLLEGE MANAGEMENT PROJECT



Chapter-10

CONCLUSION

A complete review of the application was conducted. The objective for the development of the application was fulfilled. The application was providing services as expected. This application appreciably reduces the time and efforts required. The client was satisfies with the functionality and performance and the reviews and feedbacks were overwhelming.

Earlier the work was being carried out manually. But with the development of the "COLLEGE MANAGEMENT SYSTEM", the whole work is get computerized.

The user of the application can perform the task very easily, by having little knowledge of computer.

No doubt, software development is an ever- processing field. Further improvements and suggestions are always there in terms of more validations, checks, greater simplicity and reliability.

ADVANTAGES OF SYSTEM

Advantages of the proposed system over the earlier traditional system can be organize as follows

1. Provide a solid security to system and vital data.
2. Prevent unauthorized access.
3. Efficient and controlled Booking Handling.
4. Multiple copies of record allow recovery in case of damage or loss to data.
5. User-friendly interface.
6. Quickly information retrieval.
7. Warning and alarm system for personal shortages.
8. Provides a quick view of past information.

9. Also manages the payment record keeping.
10. Provides a list of regular employees.
11. Generate reports to support the management.
12. Minimizes the paperwork.
13. Accurate and timely control of different tasks.
14. Allows 'Optimal utilization' of various resources viz. finances, manpower and machinery.
15. Cost-effective system.

Chapter-11

LIMITATIONS

No endeavor can be considered as 100% accurate and foolproof. In every single task there is a chance of further improvement. As with every new proposed system there are also some limitations and lacks that associates with this system. The limitations of this system are as follows

- Hardware Limitation: Although system works good on a normal configured PC but it might not work well enough on PCs having lower configuration than specified (less than 400 MHZ and 64 MB RAM).
- Software Limitation: In software limitation at least a 32 bit Windows 95 OS is required though a platform of windows 98 is recommended.
- Operational Limitation: It requires a computer literate operational staff to operate on.
- Validation of data at Field-Level as well as at Form-Level is employed which gives accuracy to system but slightly slowed down the processing since code become bulky.

Chapter-12

BIBLOGRAPHY

The following books are being used to fulfill the requirement of the proposed project and are helpful in the understanding, development and the maintenance of the project.

- DATABASE MANAGEMENT SYSTEM (DBMS) **BY Puneet Kumar and Sushil Bhardwaj(Kalyani Publishers)**
- SYSTEM ANALYSIS AND DESIGN (SAD) **BY Elias Award (Kalyani Publishers)**
- ASP .NET **BY Manik Sharma and Rajinder Singh (Kalyani Publishers)**
- VB.NET Black Book **BY Steven Holzner**