A
Project
On

"College administration management system"

Submitted to

Shiksha Mandal's

G.S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR

(AUTONOMOUS)

In the Partial Fulfillment of

B.Com. (Computer Application) Final Year

Submitted by
Tanu R. kalaskar

Khushi R. yadav

Under the Guidance of

Pravin J.Yadao

Shiksha Mandal's

G. S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR

(AUTONOMOUS)

AUTONOMOUS

2022-2023

Shiksha Mandal's

G. S. COLLEGSE OF COMMERCE & ECONOMICS, NAGPUR (AUTONOMOUS)

CERTIFICATE

(2022 - 2023)

This is to certify that Miss .Tanu kalaskar & Miss .Khushi yadav has completed their project on the topic of College administration management system Services prescribed by G. S. College of Commerce & Economics, Nagpur (Autonomous) for B.Com. (Computer Application) – Semester-VI.

Date:

Place: Nagpur

Pravin J. Yadao

Project Guide

External Examiner **Internal Examiner**

ACKNOWLEDGEMENT

We take this opportunity to express our deep gratitude and whole hearted thanks to project guide Prof. Pravin Yadao, Coordinator for his guidance throughout this work. We are very much thankful to him for his constant encouragement, support and kindness.

We are also grateful to our teachers Prof. Rahul Tiwari, Prof. Sushma Gawande, Prof. Preeti Rangari, Prof. Prajakta Deshpande and Prof. Haresh Naringe for their encouragement, help and support from time to time.

We also wish to express our sincere thanks to Principal Dr. Swati S.Kathaley for providing us wide range of opportunities, facilities and inspiration to gather professional knowledge and material without which this project could not have been complete

Date

Place: Nagpur

(Miss. Tanu R. Kalaskar)

(Miss. Khushi R. Yadav)

DECLARATION

We (**Miss. Tanu R. Kalaskar & Miss Khushi R. Yadav**) hereby honestly declare that the work entitled "**College administration management system**" submitted by us at **G. S. College of Commerce & Economics, Nagpur (Autonomous)** in partial fulfillment of requirement for the award of **B.Com. (Computer Application)** degree by **Rashtrasant Tukadoji Maharaj, Nagpur University, Nagpur** has not been submitted elsewhere for the award of any degree, during the academic session 2022-2023.

The project has been developed and completed by us independently under the supervision of the subject teacher and project guide.

Date

Place: Nagpur

(Miss. Tanu R. Kalaskar)

(Miss. Khushi R. Yadav)

SYNOPSIS

1.Introduction: (Write 4 to 5 lines)

", computers are better solution to us. Now a day the concept of College Management has changed a lot from where it has started. The concept has become more professional and modern. Till date large numbers of software have been made for various departments to reduce the time consumption of lengthy processes and to accelerate the process of doing work. This project is about "College Management".

2. Objectives of the project: (Write only 5 points)

- Main objective of college management administration system project is to record various details of activities of the college.
- It will simplify college work
- This system can maintain huge number of college records including student, course, attendance, employee etc.
- This system can maintain huge number of college records including student, course, attendance, employee etc.

3. Project Category: informative Website.

4. Tools/ Platform/ Languages to be used: My SQL(Backend) or VB.Net (Front-end).

5. Scope of future application: (Write 4 to 5 points)

- To increase the accuracy and efficiency in work.
- To reduce the calculation work, which was earlier carried out manually.
- To eliminate the time required to search, delete or to modify particular record.
- To reduce overall cost of installing a particular system.

Submitted by,
Khushi Yadav.
Tanu Kalaskar.

Approved by,

Prof. Pravin Yadao
Project Guide

INDEX

INDEX

INTRODUCTION

INTRODUCTION

In the current scenario where everything is being computerized due to the large and wide spread benefits of computers in every sector. Today, hardly any corner of life is left untouched by computers. In Companies, College, Banks, Grocery shops, Jumbo Sized Department Stores, Hospitals, Post Officers, Government Offices, Reservation Counters of Travel Companies, Hostels and everywhere we find the Computers. In current scenario, right decision in the right time must be made and to take correct decision on right time and computer has a great role to play.

Till date large numbers of software have been made for various departments to reduce the time consumption of lengthy processes and to accelerate the process of doing work. This project is about "College Management System". I have tried my level best to make project more efficient for practical use.

As our line of work is "College Management Administration System", computers are better solution to us. Now a day the concept of College Management has changed a lot from where it has started. The concept has become more professional and modern. Till date large numbers of software have been made for various departments to reduce the time consumption of lengthy processes and to accelerate the process of doing work. This project is about "College Management". I have tried my level best to make project more efficient for practical use. In a College Management, we need to maintain records of all the students & teachers including their salaries records and other information's such as their student's results, their improvements and teachers salary, etc.

OBJECTIVE

OBJECTIVE

Main objective of college management administration system project is to record various details of activities of the college. It will simplify college work. This system can maintain huge number of college records including student, course, attendance, employee etc. College management administration system project is designed to remove all the deficiency of the current system.

Functions of college management administration system project: -

- This project contains various sections. One section is for managing master record. It includes department, course, designation, qualification etc.
- Other section is for managing student admission, employee attendance and exam.
- This system generates various reports. That makes monitoring process very easy. Now admin can consume more time for monitoring the college for better administration.
- Paper work is reduced with the use of this application.
- Each user can access the system with his/her own user id and password.
- College Management System Project will work on college area network. Under this College Management System Project students as well as college management can get all the information by using their valid registration id and password.
- College Management System Project will also help the college administrator. Administrator will able to get all the information regarding any faculty members and students. Admin panel will also able to edit the login display panel of teachers, staff members, students etc.
- Admin can add person, delete person, can give special permission. Admin can schedule any type of updation.
- For student section, Student can login in the system using login id and password if he she is new user then they can register themselves. They can check any type of information regarding college.

PRELIMINARY SYSTEM ANALYSIS

PRELIMINARY SYSTEM ANALYSIS

PRELIMINARY INVESTIGATION

The preliminary investigation phase is the first phase of the classic systems development process. In other methodologies this might be called the initial study phase, survey phase or planning phase. The preliminary investigation answers the question, "Is this project worth looking at?" To answer this question, the preliminary investigation must define the scope of the project and the perceived problems, opportunities, and directives that triggered the project. Assuming the project is deemed worth considering, the survey phase must also establish the project planning terms of scope, development strategy or "route", schedule, resource requirements, and budget.

The preliminary investigation phase is intended to be quick. The entire phase should not exceed two or three days for most projects. The preliminary investigation typically includes the following tasks: -

1. List problems, opportunity and directives.

2. Negotiate preliminary scope.

3. Assess project worth.

4. Plan the project.

5. Present the project and plan.

FLAW IN PRESENT SYSTEM

Every system contains some flaws. In this system, when the user inputs wrong data the system provides wrong output. The main flaw of the system is it is very expensive. The estimate of purchase and installation goes very high. Following are some Advantages & Dis-Advantages of the System.

Advantages:

- A computerized system saves time and it is easy to check records.
- These systems provide accuracy that means data quality & content remain constant.
- With the help of this system we can work fast with more accuracy. Searching the information by code is very easy and understandable.
- Without any human intervention the chances of data destruction is very low.
- It arrange the records systematically **Dis-Advantages:**
- Computerized system is very expensive.
- Computerized system is not easy to understand by anyone.
- We have to supply correct data and information. If we supplied wrong data or information, system cannot correct it and we get wrong output.
- We can delete any record easily.

NEED OF NEW SYSTEM

- 1.To increase the accuracy and efficiency in work.

- 2 To reduce the calculation work, which was earlier carried out manually.

- 3. To eliminate the time required to search, delete or to modify particular record.

- 4. To reduce paper work.

- 5. To reduce overall cost of installing a particular system.

- 6. The facility of report generation is also possible. Any combination can be done to generate the report in desired manner. This helps in giving it a systematic & clean look.

FEASIBILITY STUDY

- Feasibility of a system means whether the system is practically possible to build or not. It also evaluates the benefits of the new system.
- Feasibility is evaluated from developers and customer's point of view.
- The software/system, which is to be, is analyzed in sufficient detail and it is found that the system, which is to be developed, is Technically, Operationally and economically feasible.

➢ Technical feasibility: -

Technical Feasibility centers around the existing computer system i.e.... the availability of the required hardware, software and operating system. This project is developed in VB.NET, it is a highly structured language.

➢ Operational feasibility:-

The proposed system/software is user friendly in nature. This system will be helpful for the user to perform all necessary operations efficiently and effectively. The Graphic user interface is self-explanatory as a result there is no need of providing any kind of training to the user about the software provided the user is computer literate. Demonstration of the system/software once to the user will definitely move the user operate the software efficiently.

➢ Economic feasibility: -

The cost of installing this project is not high. Once installed, the project gives long-term benefit. Cost of the Hardware and the Software that is required to build the system is very much within the reach of the organization. So, it is possible to go ahead with the proposed system from economic point of view.

PROJECT CATEGORY

Database Technology: -

Database technology is one of the most rapidly growing areas of computer or information science. The total amount of data submitted to database could be measured in millions of bytes. A database system is little more than computer based record-keeping system. It is a collection of interrelated data stored together without harmful or unnecessary redundancy.

A database is a respiratory system for stored data. It can be thought of as a unification of a several distinct data files, with redundancy, if any, in those fields, partially or fully eliminated. Also the individual pieces of the data in the database may be shared among several users. Each of these users may have access to the same piece of data. Data are one of the major components of the system. Other components are hardware, software and user. Hardware consists of secondary storage devices like disk, drums, taps etc.

Software pack of database system is responsible for a layer between the physical database itself and the user of the system. Thus software is called as DBMS (Database Management System). All requests from the users for access to the database system are handled by DBMS. The database users never need to know about the hardware level details of data.

The project i.e. "College Management Administration System" is categorized and developed by using the following:

- VB.NET (Front-end)
- My SQL (Backend)

VB.NET

VB.NET stands for Visual Basic.NET, and it is a computer programming language developed by Microsoft. It was first released in 2002 to replace Visual Basic 6. VB.NET is an object-oriented programming language. This means that it supports the features of object-oriented programming which include encapsulation, polymorphism, abstraction, and inheritance.

Visual Basic .ASP NET runs on the .NET framework, which means that it has full access to the .NET libraries. It is a very productive tool for rapid creation of a wide range of Web, Windows, Office, and Mobile applications that have been built on the .NET framework.

History of Vb.Net:-

- VB.NET is a multi-paradigm programming language developed by Microsoft on the .NET framework. It was launched in 2002 as a successor to the Visual Basic language. This was the first version of VB.NET (VB.NET 7.0) and it relied on .NET version 1.0.

- In 2003, the second version of VB.NET, VB.NET 7.1, was released. This one relied on .NET version 1.1. This version came with a number of improvements including support for .NET Compact Framework and an improved reliability and performance of the .NET IDE. VB.NET 2003 was also made available in the academic edition of Visual Studio.NET and distributed to various scholars from different countries for free.

- In 2005, VB.NET 8.0 was released. The .NET core portion was dropped from its name so as to distinguish it from the classical Visual Basic language. This version was named Visual Basic 2005. This version came with many features since Microsoft wanted this language to be used for rapid application developers.
- In 2008, VB 9.0 was introduced. This was released together with .NET 3.5. Some of the features added to this release of VB.NET included anonymous types, true conditional operator, LINQ support, XML literals, Lambda expressions, extension methods, and type inference.

VB.NET FEATURES

- VB.NET is not case sensitive like other languages such as C++ and Java.
- It is an object-oriented programming language. It treats everything as an object.
- Automatic code formatting, XML designer, improved object browser etc.
- Garbage collection is automated.
- Support for Boolean conditions for decision making.
- Simple multithreading, allowing your apps to deal with multiple tasks simultaneously.
- Simple generics.
- A standard library.
- Events management.
- References. You should reference an external object that is to be used in a VB.NET application.
- Attributes, which are tags for providing additional information regarding elements that have been defined within a program.
- Windows Forms- you can inherit your form from an already existing form.

Using In Making GUI Program

The Windows Forms library is most commonly used to create GUI interfaces in Visual Basic. All visual elements in the Windows Forms class library derive from the Control class. This provides the minimal functionality of a user interface element such as location, size, color, font, text, as well as common events like click and drag/drop. The Control class also has docking support to let a control rearrange its position under its parent.

Forms are typically designed in the Visual Studio IDE. In Visual Studio, forms are created using drag-and-drop techniques. A tool is used to place controls (e.g., text boxes, buttons, etc.) on the form (window). Controls have attributes and event handlers associated with them. Default values are provided when the control is created, but may be changed by the programmer. Many attribute values can be modified during run time based on user actions or changes in the environment, providing a dynamic application. For example, code can be inserted into the form resize event handler to reposition a control so that it remains centered on the form, expands to fill up the form, etc. By inserting code into the event handler for a keypress in a text box, the program can automatically translate the case of the text being entered, or even prevent certain characters from being inserted.

MySQL

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB

MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often, MySQL is used with other programs to implement applications that need relational database capability. MySQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress. MySQL is also used by many including Facebook, Flickr,[12] Media Wiki Twitter and YouTube. popular websites,

MySQL is offered under two different editions: the open source MySQL Community Server and the proprietary Enterprise Server MySQL Enterprise Server is differentiated by a series of proprietary extensions which install as server plugins, but otherwise shares the version numbering system and is built from the same code base.

Major features as available in MySQL 5.6:

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support

- Stored procedures, using a procedural language that closely adheres to SQL/PSM
- Triggers
- Cursors
- Updatable views
- Online Data Definition Language (DDL) when using the InnoDB Storage Engine.
- Information schema
- Performance Schema that collects and aggregates statistics about server execution and query performance for monitoring purposes.[80]
- A set of SQL Mode options to control runtime behavior, including a strict
- mode to better adhere to SQL standards.
- X/Open XA distributed transaction processing (DTP) support; two phase
- commit as part of this, using the default InnoDB storage engine Transactions with savepoints when using the default InnoDB Storage Engine.
- The NDB Cluster Storage Engine also supports transactions.
- ACID compliance when using InnoDB and NDB Cluster Storage Engines
- SSL support
- Query caching

- Sub-SELECTs (i.e. nested SELECTs)
- Built-in replication support
  Asynchronous replication: master-slave from one master to many slaves or many masters to one slave
  - Semi synchronous replication: Master to slave replication where the master waits on replication
  - Synchronous replication: Multi-master replication is provided MySQL Cluster
  - Virtual Synchronous: Self managed groups of MySQL servers with multi master support can be done using: Galera Cluster or the built in Group Replication plugin[89]

- . Full-text indexing and searching[b]

- Embedded database library

- Unicode support[a]

- Partitioned tables with pruning of partitions in optimizer

- Shared-nothing clustering through MySQL Cluster

- . Multiple storage engines, allowing one to choose the one that is most effective for each table in the application.[c]

- Native storage engines InnoDB, MyISAM, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, NDB Cluster.

- . Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.

- The developers release minor updates of the MySQL Server approximately every two months. The sources can be obtained from MySQL's website or from MySQL's GitHub repository, both under the GPL license.

HARDWARE AND SOFTWARE REQUIREMNT SPECIFICATION

HARDWARE AND SOFTWARE REQUIREMNT SPECIFICATION

Hardware:

hardware specifications are technical descriptions are technical descriptions of the computer components and capabilities. Processor speed, model and manufacturer. Processor speed is typically indicated in giga, moshertz(GHz).

The higher number, the faster the computer. Random Access Memory

(RAM), This is typically indicated in gigabytes(GB).

The more RAM in a computer the more it can do simultaneously. This is typically indicated in gigabytes (GB) and refers generally to the amount of information (like documents, music and other data)your computer can hold. Other specifications might include network adapters or audio and video capabilities.

i. RAM : 8.00 GB
ii. PROCESSOR: 11th Gen
iii. HARDISK
iv. KEYBOARD
v. MOUSE

Software:

A software is abbreviated as SW and S/W, software is a collection of instructions that enable the user to interact with a computer, its hardware, or perform tasks. Without software, most computers would be useful.

Operating system

- Windows 11
- Microsoft 3

NUMBER OF MODULES AND PROCESS LOGIC

Modules are used for Global Declaration of the variables and functions whose scope are global. The main advantage of module in a project is that it needs not to declare again and again, second advantage of module is that we can pass values throughout the program wherever necessary.

In College management administration System project, there is only one standard module, used to declare all variables globally which are going to use in this project. In the project, there are six form modules also, for the efficient coding of the project. Also, module has provided the coding a short way and systematic

Fint of all, we store all the detailed information of students and other

All these tables hold the core data of their respective function and the data for reports are retrieved from these tables only as and when required

MODULES

Main modules of College Management Administration System project are

STUDENT FORM:- It helps to store all the student related information about their studies and personal data. It helps to access student data easily. Following are some student related information which can be stored using **student form.**

1.Student From

2.Fees Deposit From

3.Department From

4.Attendance Form

EMPLOYEE FORM:- It helps to store all the Staff related information about their designation, post and personal data. It also helps to access the data easily. Following are some Staff related information which can be stored using **Teaching Staff form.**

1.Teaching Staff

2.Salary From

OTHER FORM:- It helps to store the the information of the students who have taken part in extracurricular activities organised by the college. Following are some student related information who have taken part in extracurricular activities which can be stored using **other form.**

1.NCC form

2.Bonafide Form

3.Library Form

REPORTS: **-** It helps to show the data stored by the college administration at a time. As soon as we store the data or delete the data the result can be seen in reports. Following are some reports which are generated by this project.

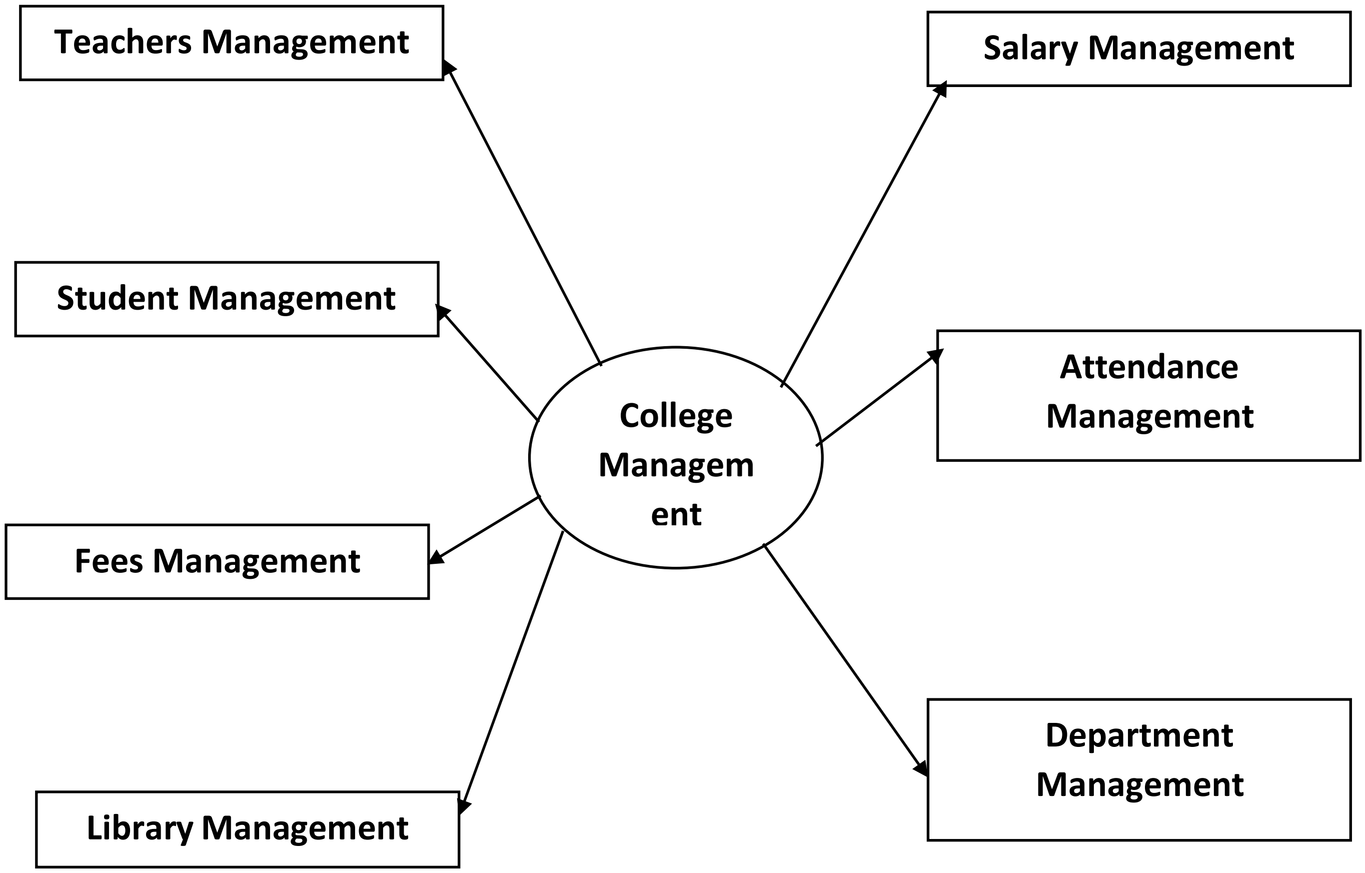
1.Salary Report

2.Teaching Staff Report

3.Fees Report

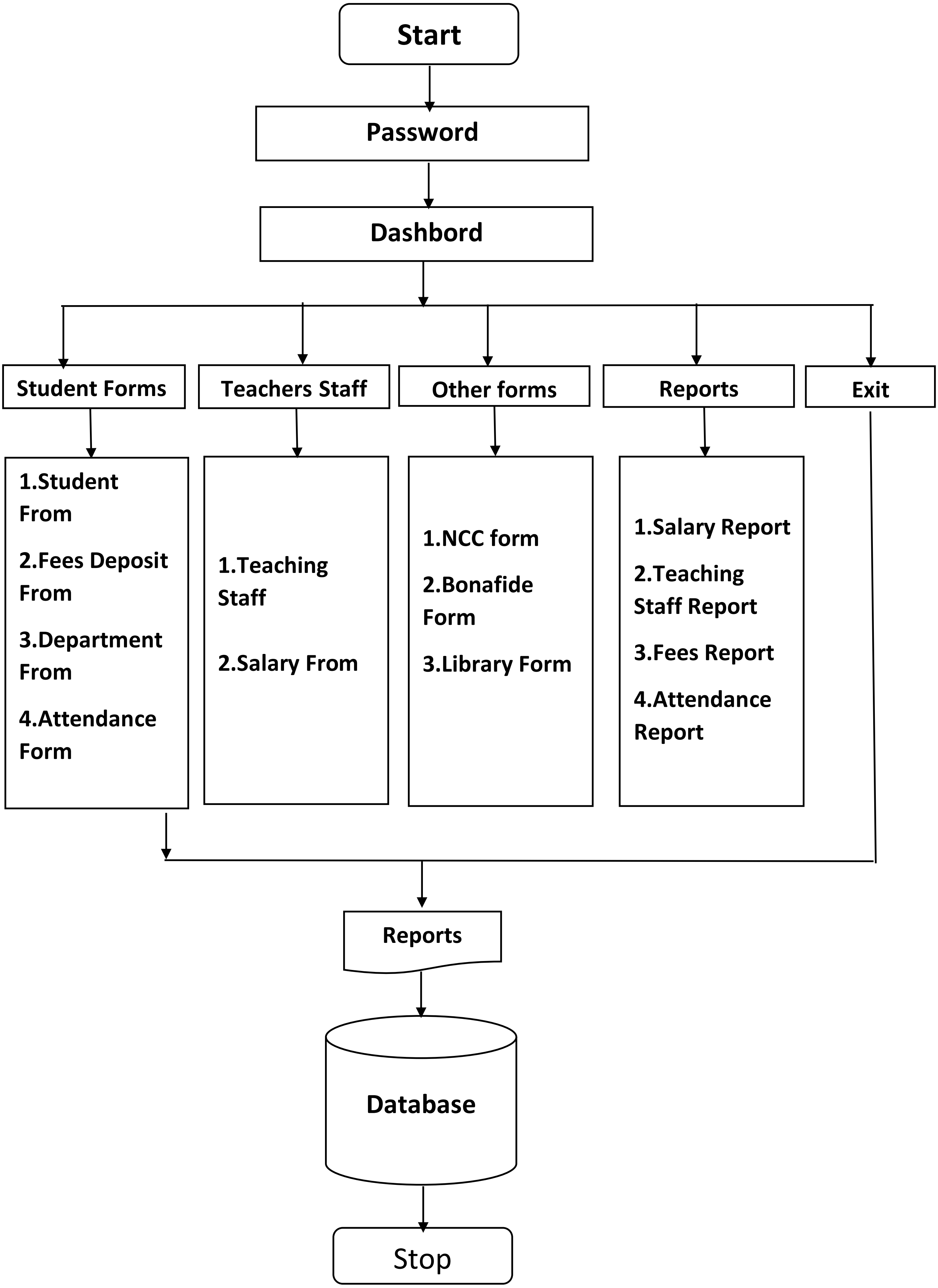
4.Attendance Report

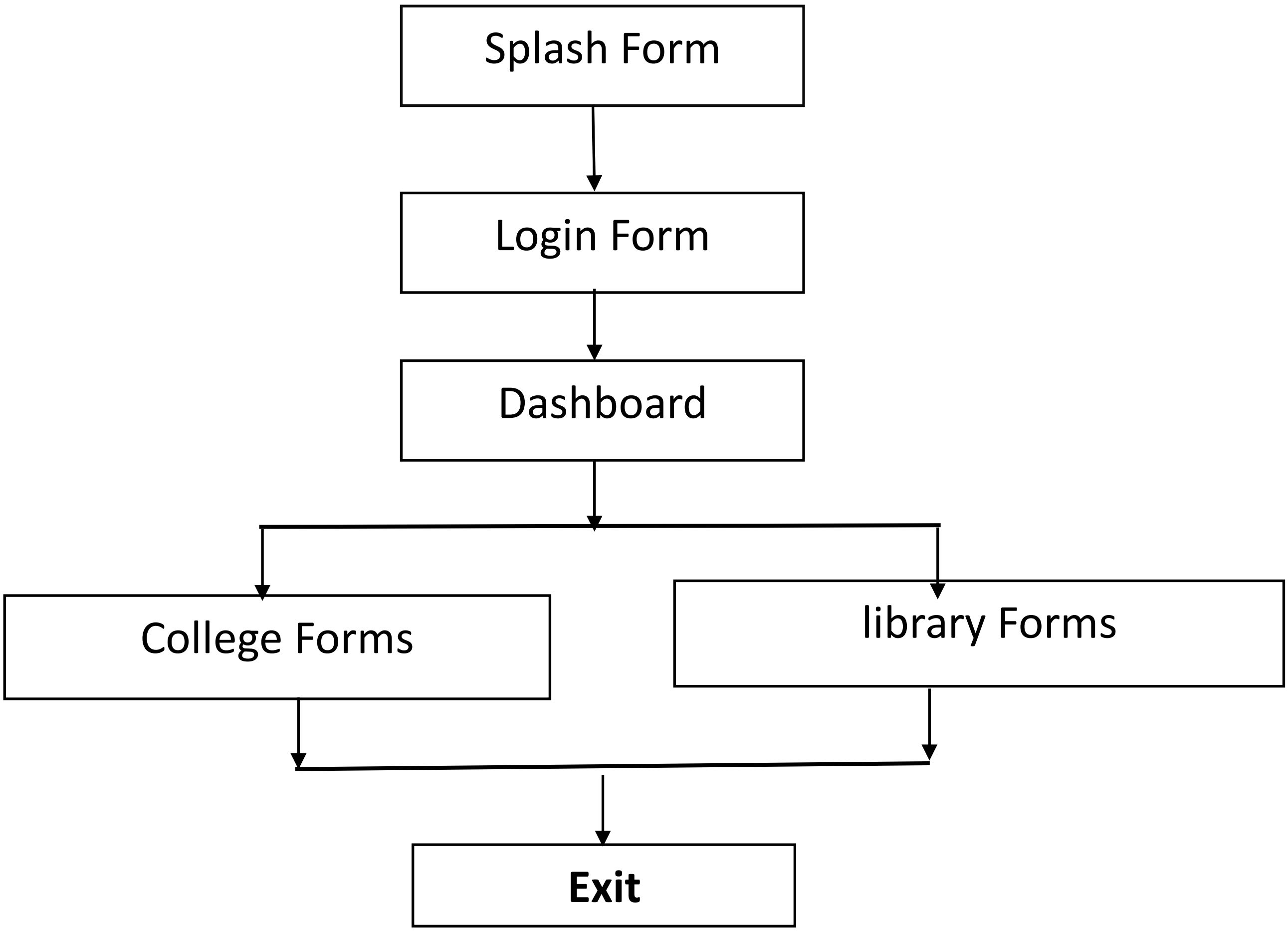
SYSTEM/MODULAR CHART



DETAILED SYSTEM ANALYSIS

DATA FLOW DIAGRAM




Splash Form
Login Form
Dashboard
College Forms
library Forms
Exit

NUMBER OF MODULES AND PROCESS LOGIC

Modules are used for Global Declaration of the variables and functions whose scope are global. The main advantage of module in a project is that it needs not to declare again and again, second advantage of module is that we can pass values throughout the program wherever necessary.

In **College management administration System project**, there is only one standard module, used to declare all variables globally which are going to use in this project. In the project, there are six form modules also, for the efficient coding of the project. Also, module has provided the coding a short way and systematic

1. Teacher Form
2. Student Form
3. Fees Form
4. Department Form
5. Attendance Form
6. Salary Form
7. Library Form
8. NSS Form
9. Bonafede certificate Form

DATA STRUCTURES AND TABLES

The College Management Administration System project records are stored in the database named “CollegeSystem.mdb”, which consists of fifteen tables. The tables in the database are as follows:

1. Teacher Table

2. Student Table

3. Fees Table

4. Department Table

5. Attendance Table

6. Salary Table

7. Library Table

8. NSS Table

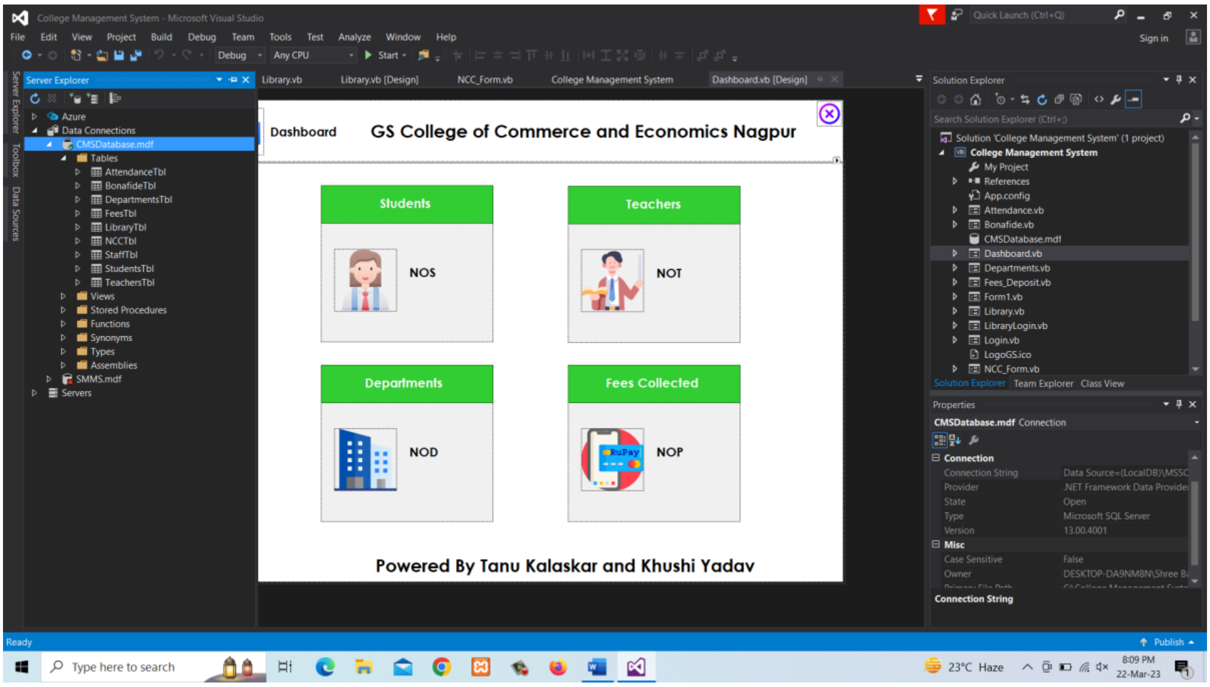
9. Bonafede certificate Table

Fint of all, we store all the detailed information of students and other details.

All these tables hold the core data of their respective function and the data for reports are retrieved from these tables only as and when required

SQL Server Data Tool

Database



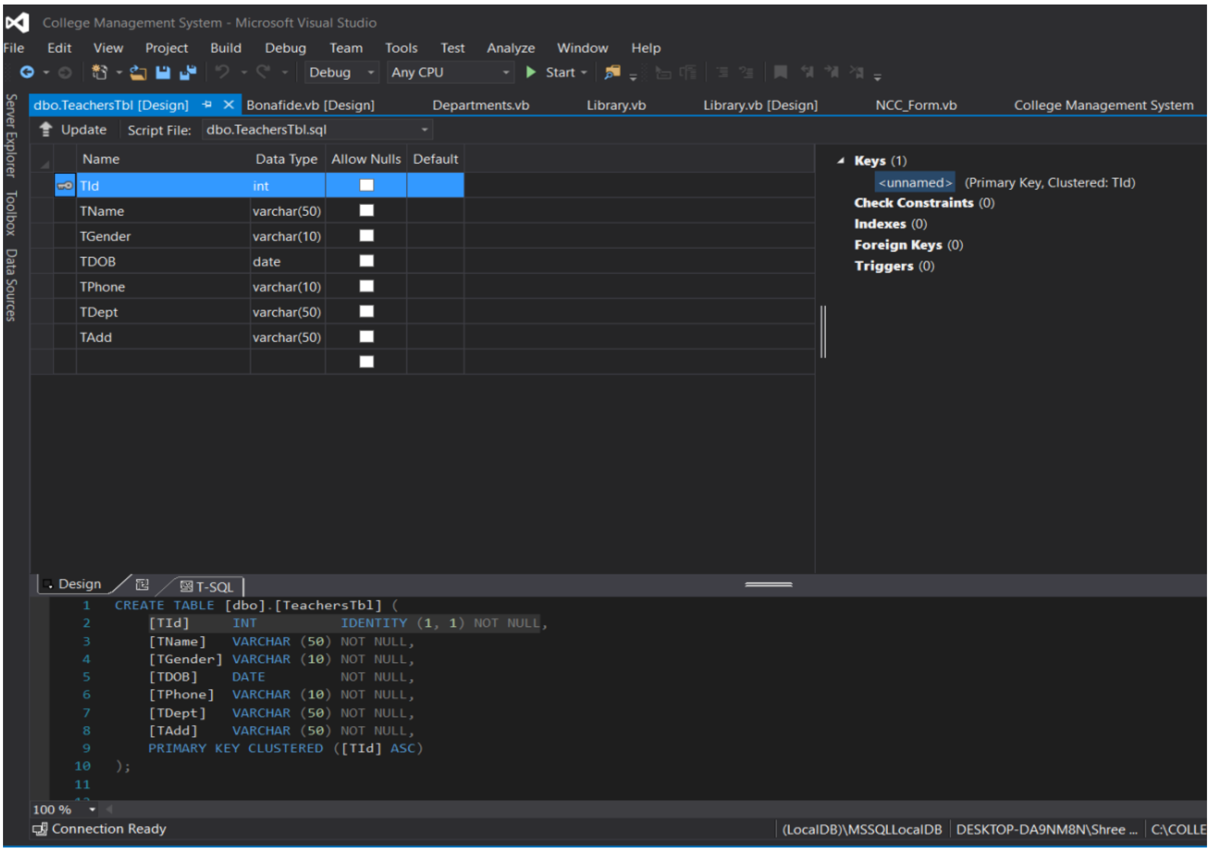
Properties

CMSDatabase.mdf Connection

Connection	
Connection String	Data Source=(LocalDB)\MSSC
Provider	.NET Framework Data Provide
State	Open
Type	Microsoft SQL Server
Version	13.00.4001
Misc	
Case Sensitive	False
Owner	DESKTOP-DA9NM8N\Shree B

Connection String

Teachers Table


College Management System - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
Start
dbo.TeachersTbl [Design]
Bonafide.vb [Design]
Departments.vb
Library.vb
Library.vb [Design]
NCC_Form.vb
College Management System
Update
Design
T-SQL
100 %

Script File: dbo.TeachersTbl.sql

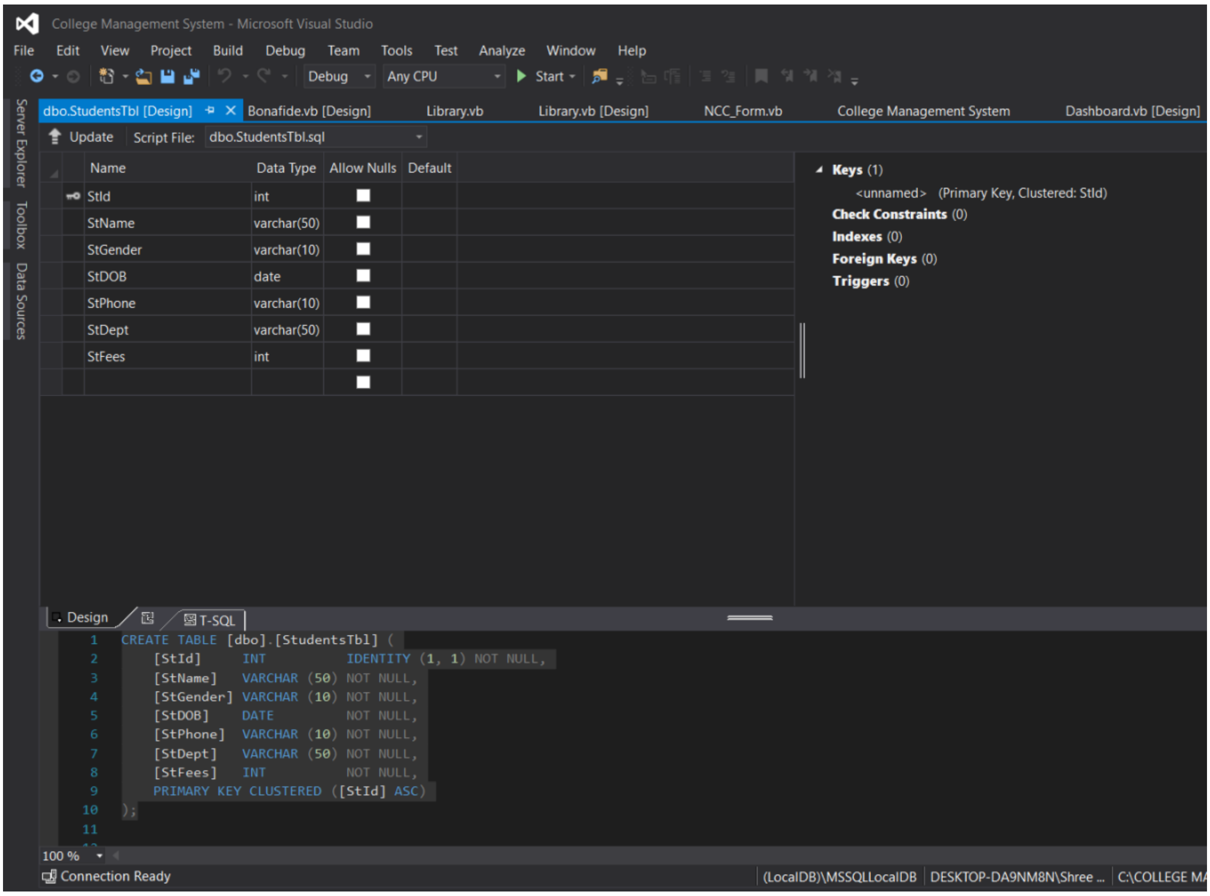
Name	Data Type	Allow Nulls	Default
TId	int	□	
TName	varchar(50)	□	
TGender	varchar(10)	□	
TDOB	date	□	
TPhone	varchar(10)	□	
TDept	varchar(50)	□	
TAdd	varchar(50)	□	
		□	

Keys (1)
<unnamed> (Primary Key, Clustered: TId)
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

```
CREATE TABLE [dbo].[TeachersTbl] (
    [TId]      INT          IDENTITY (1, 1) NOT NULL,
    [TName]    VARCHAR (50) NOT NULL,
    [TGender]  VARCHAR (10) NOT NULL,
    [TDOB]     DATE         NOT NULL,
    [TPhone]   VARCHAR (10) NOT NULL,
    [TDept]    VARCHAR (50) NOT NULL,
    [TAdd]     VARCHAR (50) NOT NULL,
    PRIMARY KEY CLUSTERED ([TId] ASC)
);
```

Connection Ready | (LocalDB)\MSSQLLocalDB | DESKTOP-DA9NM8N\Shree ... | C:\COLLE

Student Table

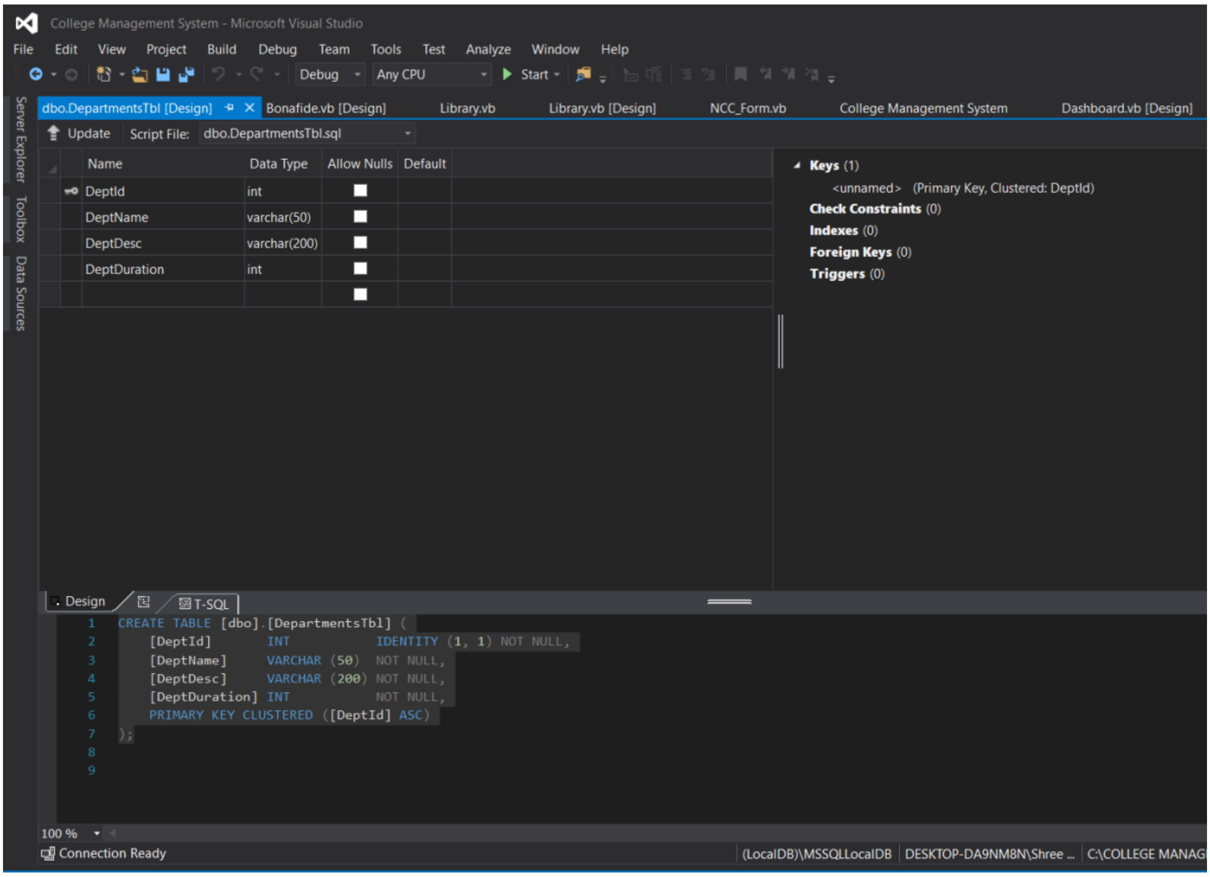


Name	Data Type	Allow Nulls	Default
StId	int	□	
StName	varchar(50)	□	
StGender	varchar(10)	□	
StDOB	date	□	
StPhone	varchar(10)	□	
StDept	varchar(50)	□	
StFees	int	□	
		□	

- **Keys** (1)
  - <unnamed> (Primary Key, Clustered: StId)
- **Check Constraints** (0)
- **Indexes** (0)
- **Foreign Keys** (0)
- **Triggers** (0)

```
CREATE TABLE [dbo].[StudentsTbl] (
    [StId]     INT          IDENTITY (1, 1) NOT NULL,
    [StName]   VARCHAR (50) NOT NULL,
    [StGender] VARCHAR (10) NOT NULL,
    [StDOB]    DATE         NOT NULL,
    [StPhone]  VARCHAR (10) NOT NULL,
    [StDept]   VARCHAR (50) NOT NULL,
    [StFees]   INT          NOT NULL,
    PRIMARY KEY CLUSTERED ([StId] ASC)
);
```

Departments Table

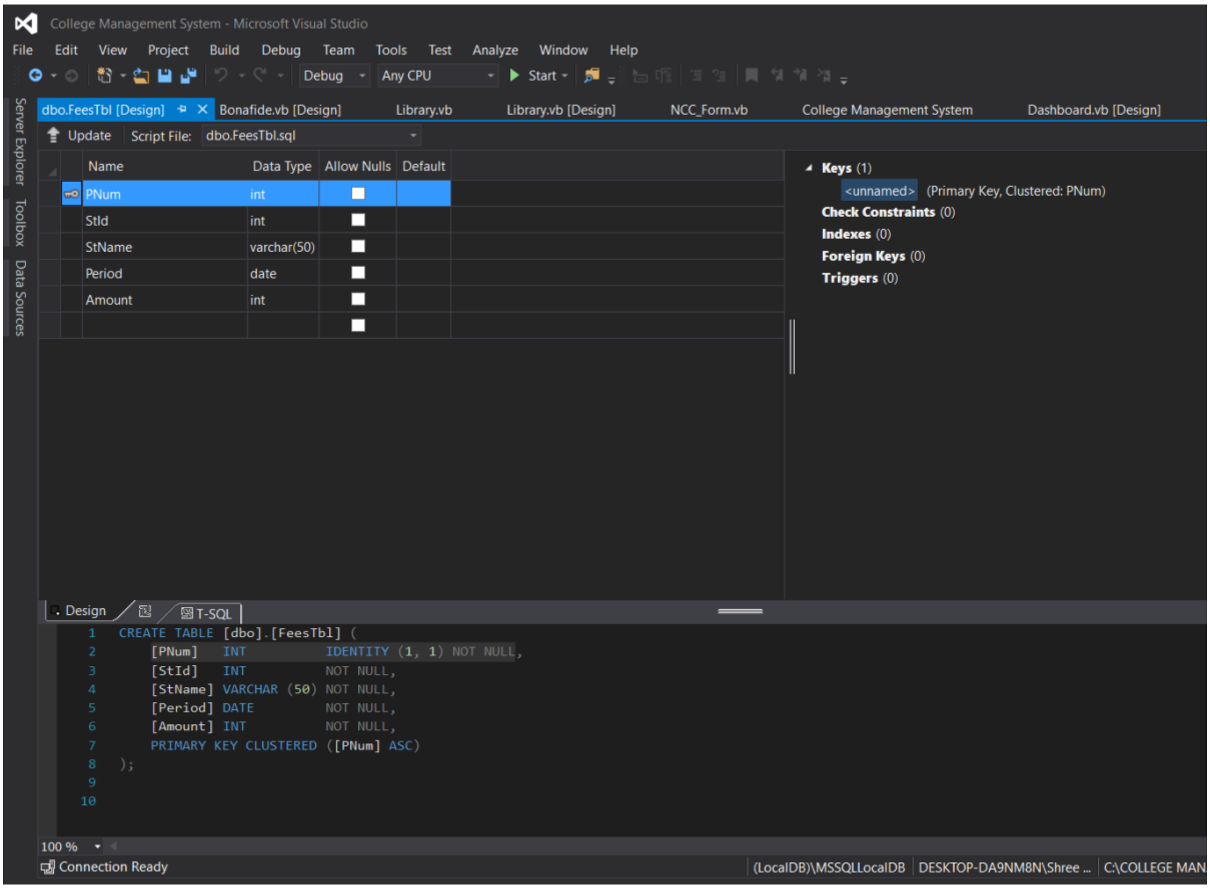

College Management System - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
Start
dbo.DepartmentsTbl [Design]
Bonafide.vb [Design]
Library.vb
Library.vb [Design]
NCC_Form.vb
College Management System
Dashboard.vb [Design]
Server Explorer
Toolbox
Data Sources
Update
Script File: dbo.DepartmentsTbl.sql
Design
T-SQL
100 %
Connection Ready
(LocalDB)\MSSQLLocalDB
DESKTOP-DA9NM8N\Shree ...
C:\COLLEGE MANAG

Name	Data Type	Allow Nulls	Default
DeptId	int	□	
DeptName	varchar(50)	□	
DeptDesc	varchar(200)	□	
DeptDuration	int	□	
		□	

- **Keys** (1)
  - <unnamed> (Primary Key, Clustered: DeptId)
- **Check Constraints** (0)
- **Indexes** (0)
- **Foreign Keys** (0)
- **Triggers** (0)

```
CREATE TABLE [dbo].[DepartmentsTbl] (
    [DeptId]       INT           IDENTITY (1, 1) NOT NULL,
    [DeptName]     VARCHAR (50)  NOT NULL,
    [DeptDesc]     VARCHAR (200) NOT NULL,
    [DeptDuration] INT           NOT NULL,
    PRIMARY KEY CLUSTERED ([DeptId] ASC)
);
```

Fees Table

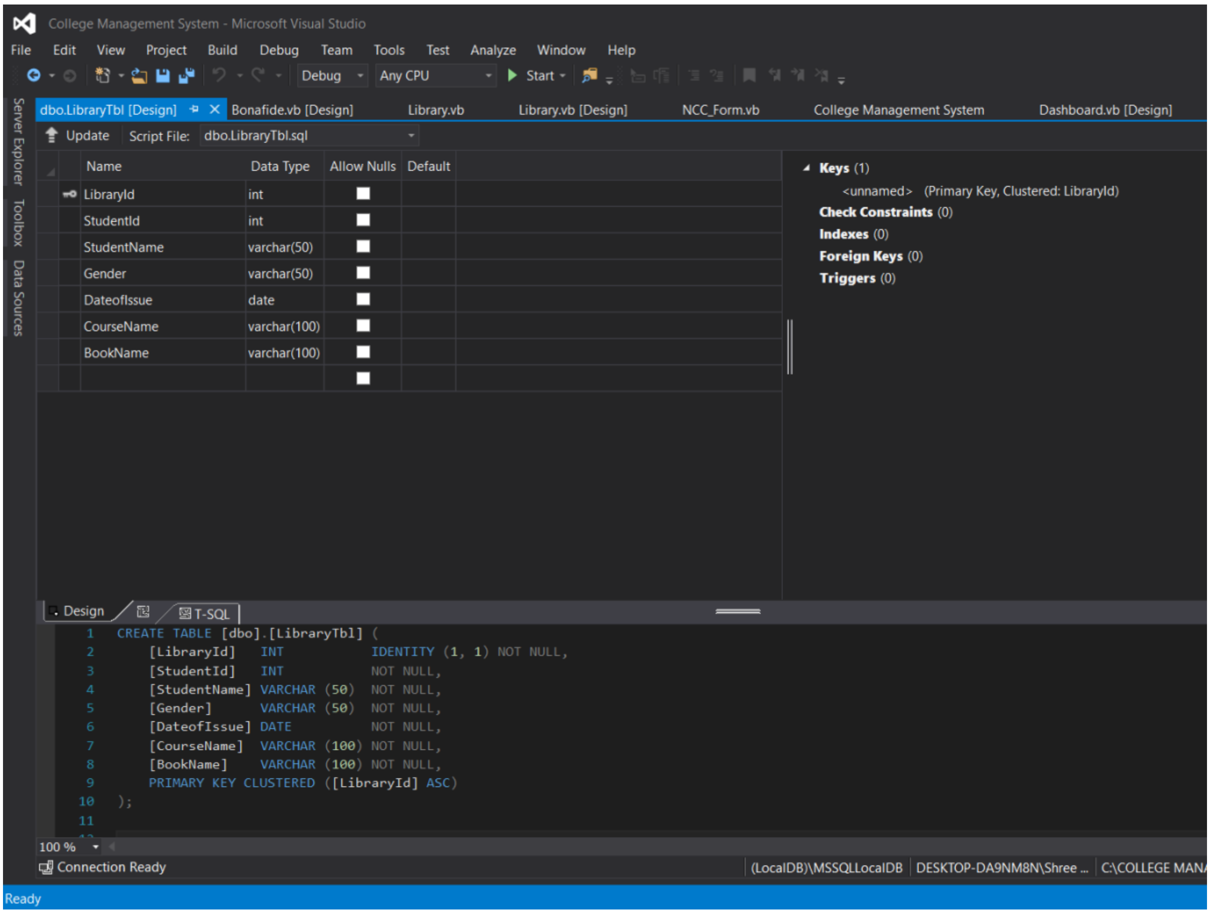


	Name	Data Type	Allow Nulls	Default
🔑	PNum	int	□	
	StId	int	□	
	StName	varchar(50)	□	
	Period	date	□	
	Amount	int	□	
			□	

- Keys (1)
  - <unnamed> (Primary Key, Clustered: PNum)
- Check Constraints (0)
- Indexes (0)
- Foreign Keys (0)
- Triggers (0)

```
CREATE TABLE [dbo].[FeesTbl] (
    [PNum]   INT          IDENTITY (1, 1) NOT NULL,
    [StId]   INT          NOT NULL,
    [StName] VARCHAR (50) NOT NULL,
    [Period] DATE         NOT NULL,
    [Amount] INT          NOT NULL,
    PRIMARY KEY CLUSTERED ([PNum] ASC)
);
```

Library Database Table

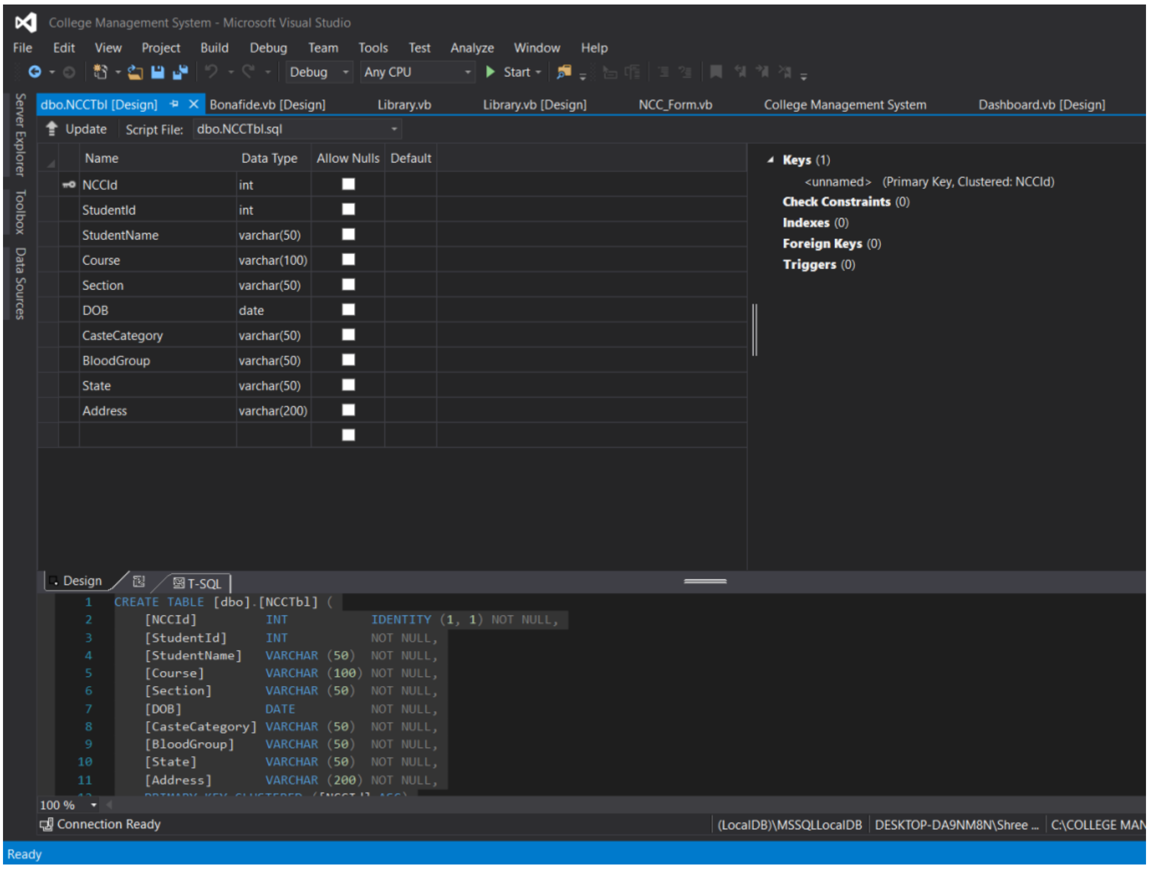


	Name	Data Type	Allow Nulls	Default
🔑	LibraryId	int	□	
	StudentId	int	□	
	StudentName	varchar(50)	□	
	Gender	varchar(50)	□	
	DateofIssue	date	□	
	CourseName	varchar(100)	□	
	BookName	varchar(100)	□	
			□	

- Keys (1)
  - <unnamed> (Primary Key, Clustered: LibraryId)
- Check Constraints (0)
- Indexes (0)
- Foreign Keys (0)
- Triggers (0)

```
CREATE TABLE [dbo].[LibraryTbl] (
    [LibraryId]   INT           IDENTITY (1, 1) NOT NULL,
    [StudentId]   INT           NOT NULL,
    [StudentName] VARCHAR (50)  NOT NULL,
    [Gender]      VARCHAR (50)  NOT NULL,
    [DateofIssue] DATE          NOT NULL,
    [CourseName]  VARCHAR (100) NOT NULL,
    [BookName]    VARCHAR (100) NOT NULL,
    PRIMARY KEY CLUSTERED ([LibraryId] ASC)
);
```

NCC Table

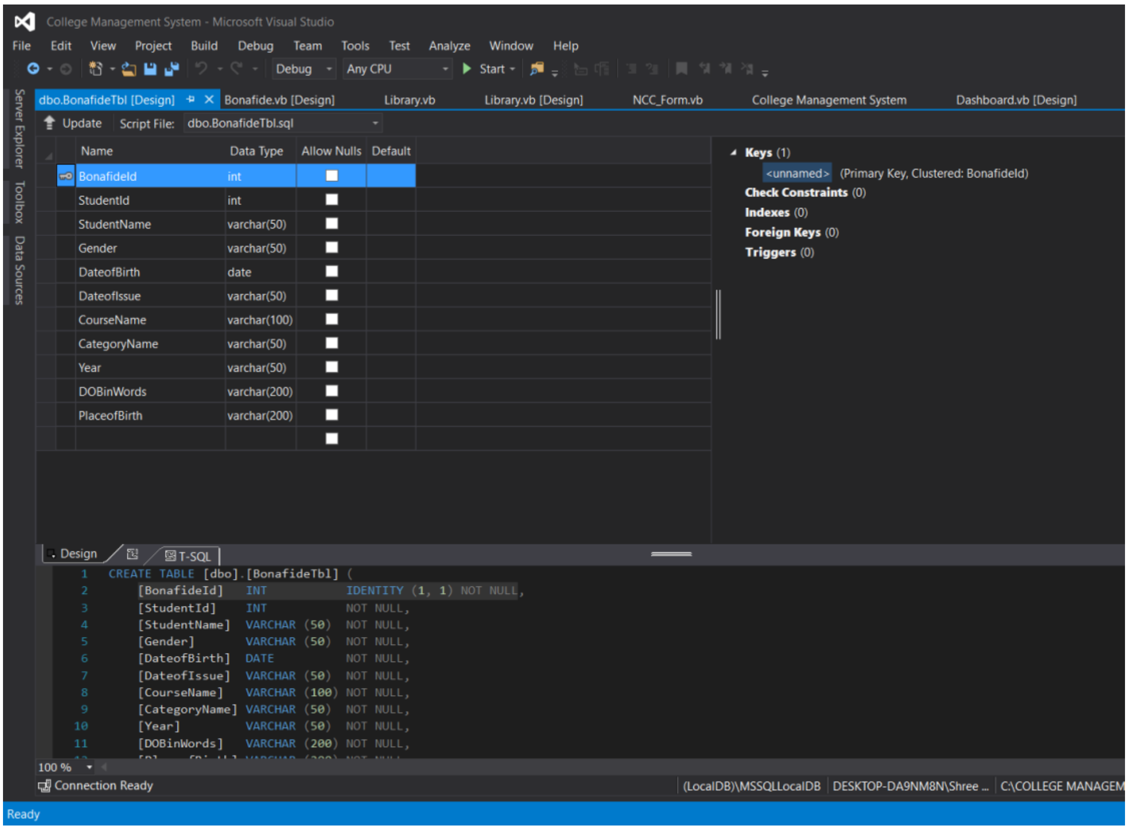


Name	Data Type	Allow Nulls	Default
NCCId	int	□	
StudentId	int	□	
StudentName	varchar(50)	□	
Course	varchar(100)	□	
Section	varchar(50)	□	
DOB	date	□	
CasteCategory	varchar(50)	□	
BloodGroup	varchar(50)	□	
State	varchar(50)	□	
Address	varchar(200)	□	
		□	

Keys (1)
<unnamed> (Primary Key, Clustered: NCCId)
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

```
CREATE TABLE [dbo].[NCCTbl] (
    [NCCId]         INT           IDENTITY (1, 1) NOT NULL,
    [StudentId]     INT           NOT NULL,
    [StudentName]   VARCHAR (50)  NOT NULL,
    [Course]        VARCHAR (100) NOT NULL,
    [Section]       VARCHAR (50)  NOT NULL,
    [DOB]           DATE          NOT NULL,
    [CasteCategory] VARCHAR (50)  NOT NULL,
    [BloodGroup]    VARCHAR (50)  NOT NULL,
    [State]         VARCHAR (50)  NOT NULL,
    [Address]       VARCHAR (200) NOT NULL,
    PRIMARY KEY CLUSTERED ([NCCId] ASC)
```

Bonafide Table

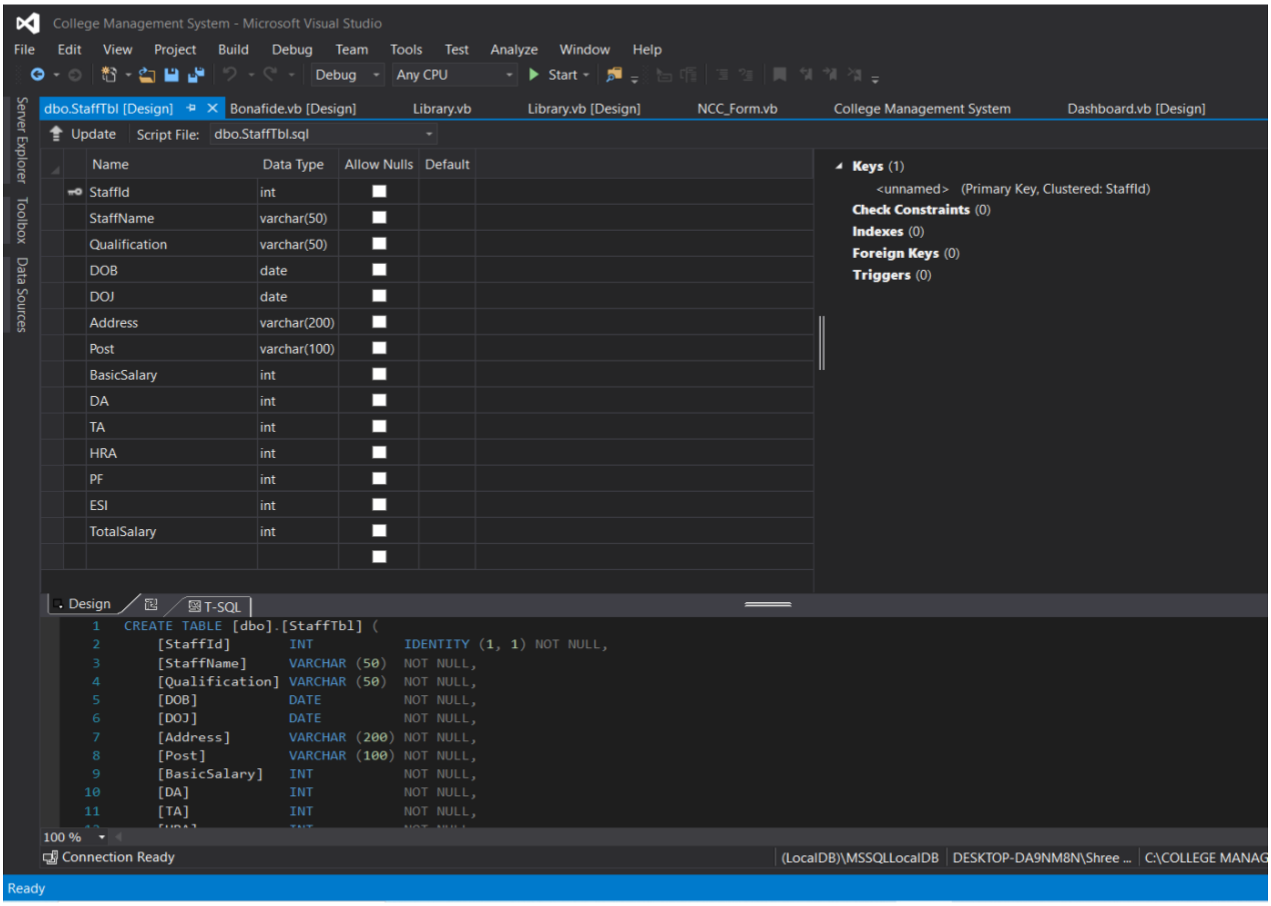


Name	Data Type	Allow Nulls	Default
BonafideId	int	□	
StudentId	int	□	
StudentName	varchar(50)	□	
Gender	varchar(50)	□	
DateofBirth	date	□	
DateofIssue	varchar(50)	□	
CourseName	varchar(100)	□	
CategoryName	varchar(50)	□	
Year	varchar(50)	□	
DOBinWords	varchar(200)	□	
PlaceofBirth	varchar(200)	□	
		□	

Keys (1)
<unnamed> (Primary Key, Clustered: BonafideId)
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

```
CREATE TABLE [dbo].[BonafideTbl] (
    [BonafideId]   INT           IDENTITY (1, 1) NOT NULL,
    [StudentId]    INT           NOT NULL,
    [StudentName]  VARCHAR (50)  NOT NULL,
    [Gender]       VARCHAR (50)  NOT NULL,
    [DateofBirth]  DATE          NOT NULL,
    [DateofIssue]  VARCHAR (50)  NOT NULL,
    [CourseName]   VARCHAR (100) NOT NULL,
    [CategoryName] VARCHAR (50)  NOT NULL,
    [Year]         VARCHAR (50)  NOT NULL,
    [DOBinWords]   VARCHAR (200) NOT NULL,
    [PlaceofBirth] VARCHAR (200) NOT NULL,
```

Staff Salary Table



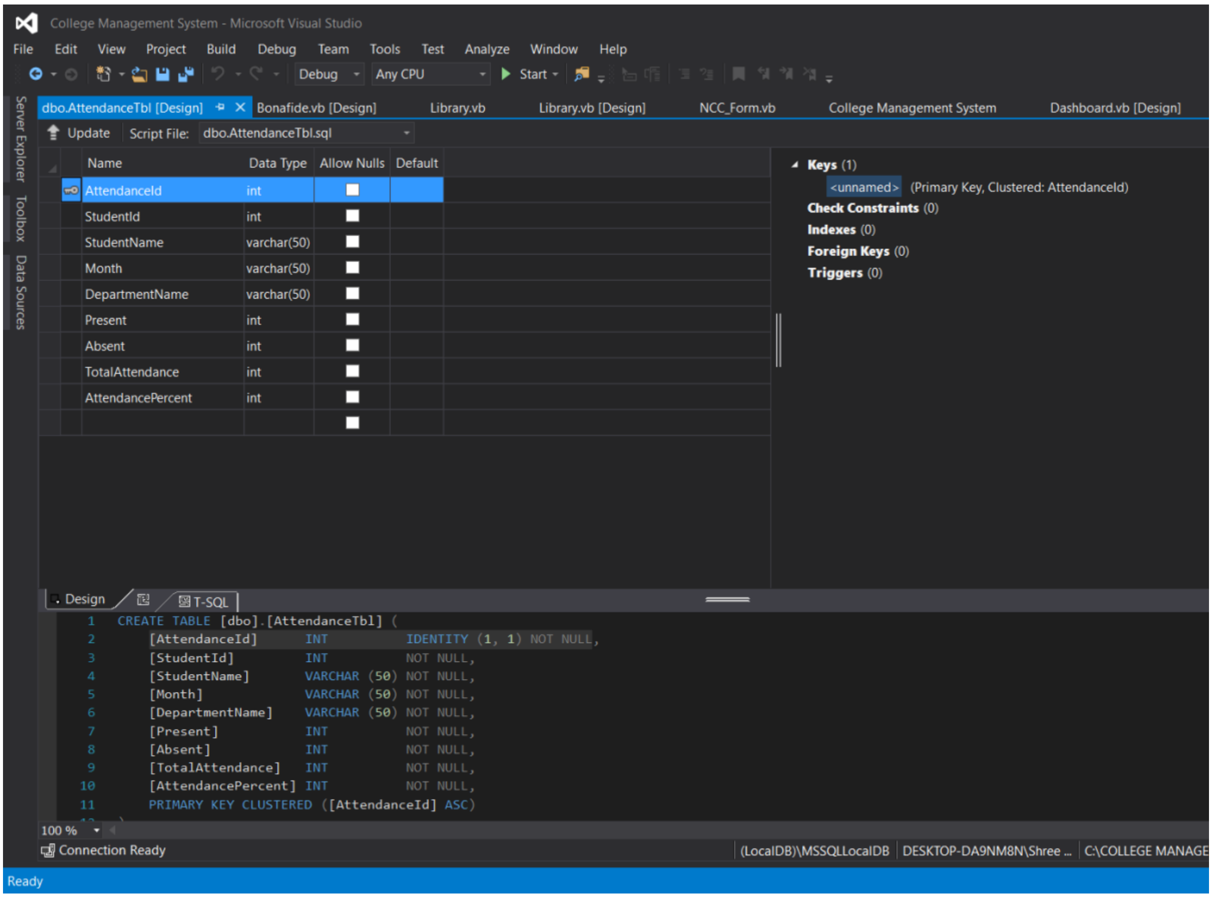
Script File: dbo.StaffTbl.sql

Name	Data Type	Allow Nulls	Default
StaffId	int	□	
StaffName	varchar(50)	□	
Qualification	varchar(50)	□	
DOB	date	□	
DOJ	date	□	
Address	varchar(200)	□	
Post	varchar(100)	□	
BasicSalary	int	□	
DA	int	□	
TA	int	□	
HRA	int	□	
PF	int	□	
ESI	int	□	
TotalSalary	int	□	
		□	

Keys (1)
<unnamed> (Primary Key, Clustered: StaffId)
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

```
CREATE TABLE [dbo].[StaffTbl] (
    [StaffId]       INT           IDENTITY (1, 1) NOT NULL,
    [StaffName]     VARCHAR (50)  NOT NULL,
    [Qualification] VARCHAR (50)  NOT NULL,
    [DOB]           DATE          NOT NULL,
    [DOJ]           DATE          NOT NULL,
    [Address]       VARCHAR (200) NOT NULL,
    [Post]          VARCHAR (100) NOT NULL,
    [BasicSalary]   INT           NOT NULL,
    [DA]            INT           NOT NULL,
    [TA]            INT           NOT NULL,
```

Attendance Table



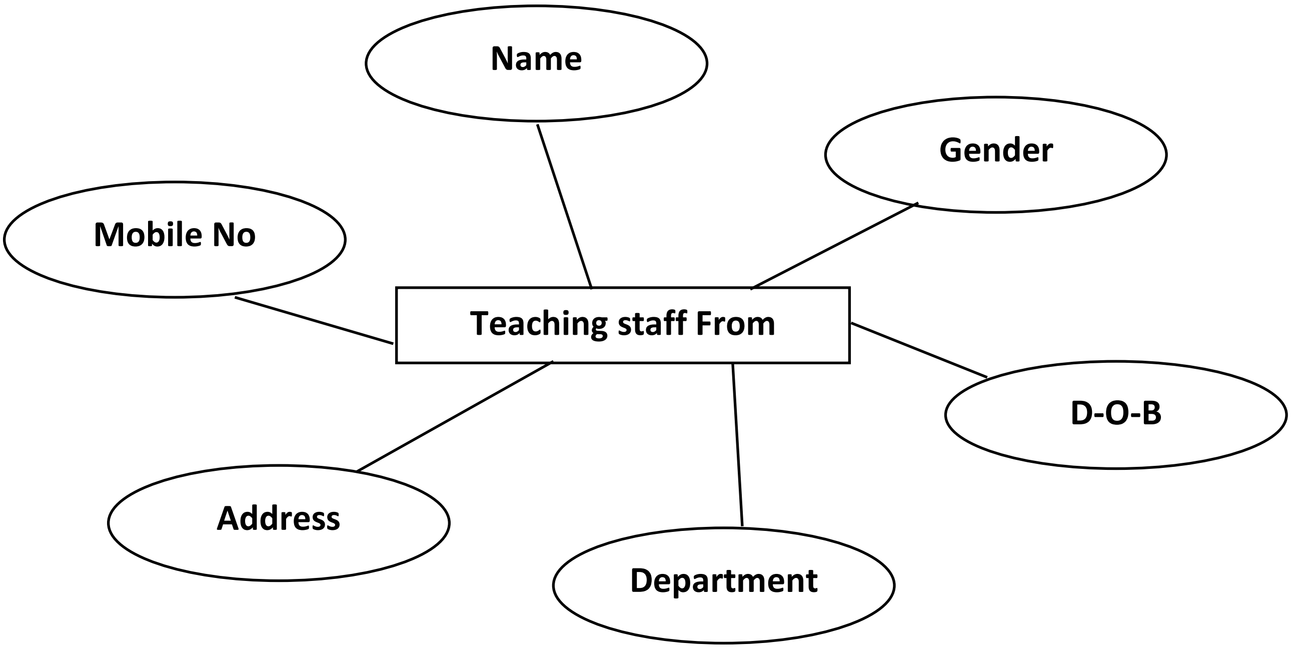
Script File: dbo.AttendanceTbl.sql

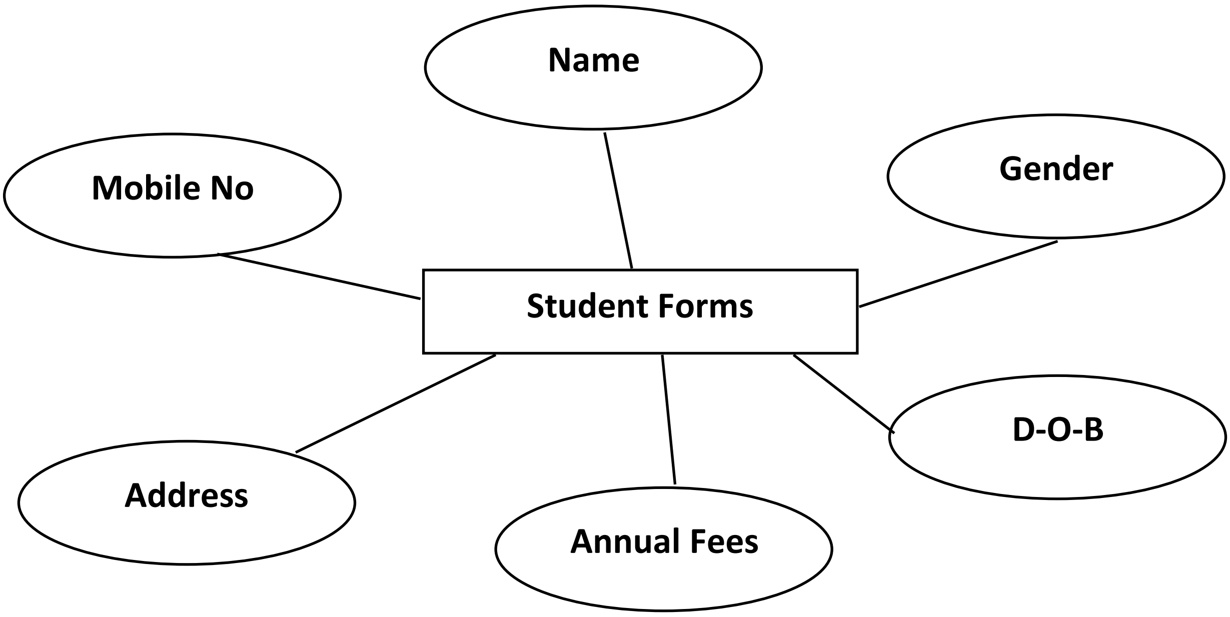
Name	Data Type	Allow Nulls	Default
AttendanceId	int	□	
StudentId	int	□	
StudentName	varchar(50)	□	
Month	varchar(50)	□	
DepartmentName	varchar(50)	□	
Present	int	□	
Absent	int	□	
TotalAttendance	int	□	
AttendancePercent	int	□	
		□	

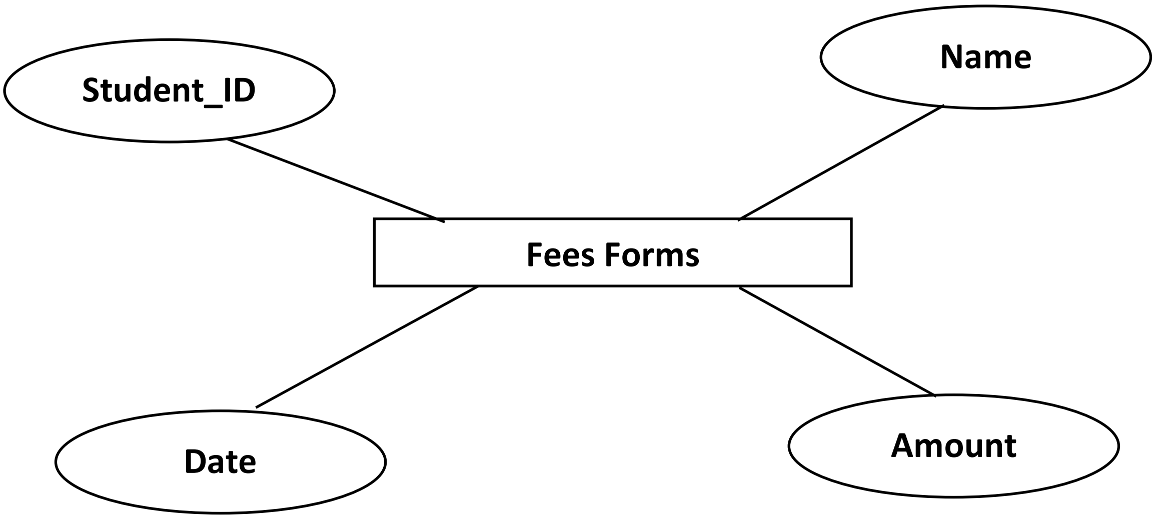
Keys (1)
<unnamed> (Primary Key, Clustered: AttendanceId)
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

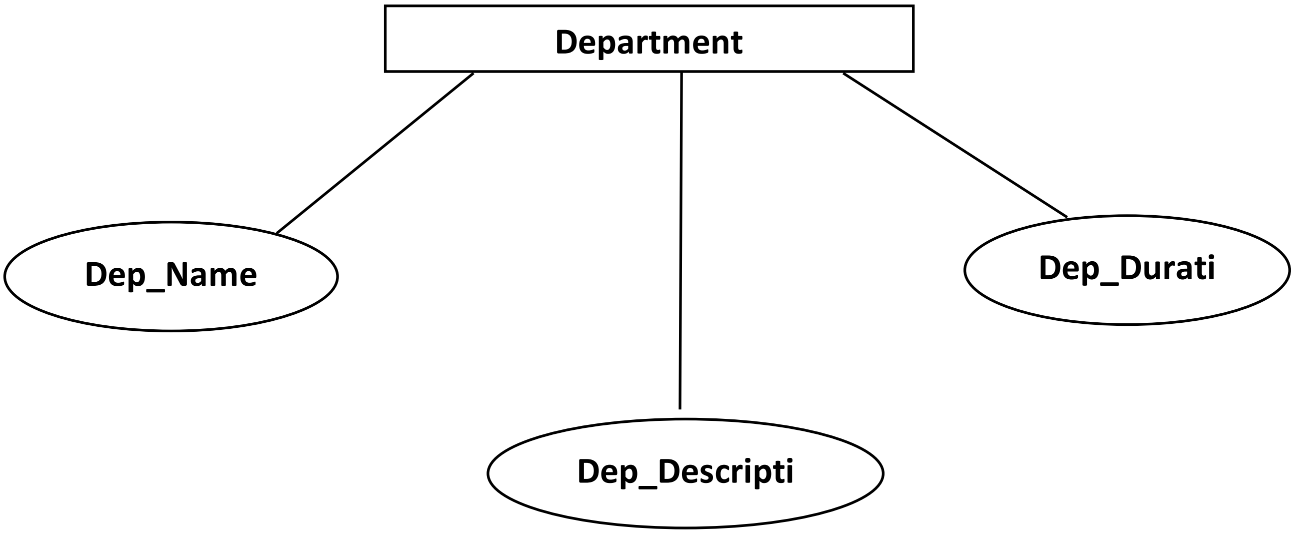
```
CREATE TABLE [dbo].[AttendanceTbl] (
    [AttendanceId]      INT          IDENTITY (1, 1) NOT NULL,
    [StudentId]         INT          NOT NULL,
    [StudentName]       VARCHAR (50) NOT NULL,
    [Month]             VARCHAR (50) NOT NULL,
    [DepartmentName]    VARCHAR (50) NOT NULL,
    [Present]           INT          NOT NULL,
    [Absent]            INT          NOT NULL,
    [TotalAttendance]   INT          NOT NULL,
    [AttendancePercent] INT          NOT NULL,
    PRIMARY KEY CLUSTERED ([AttendanceId] ASC)
```

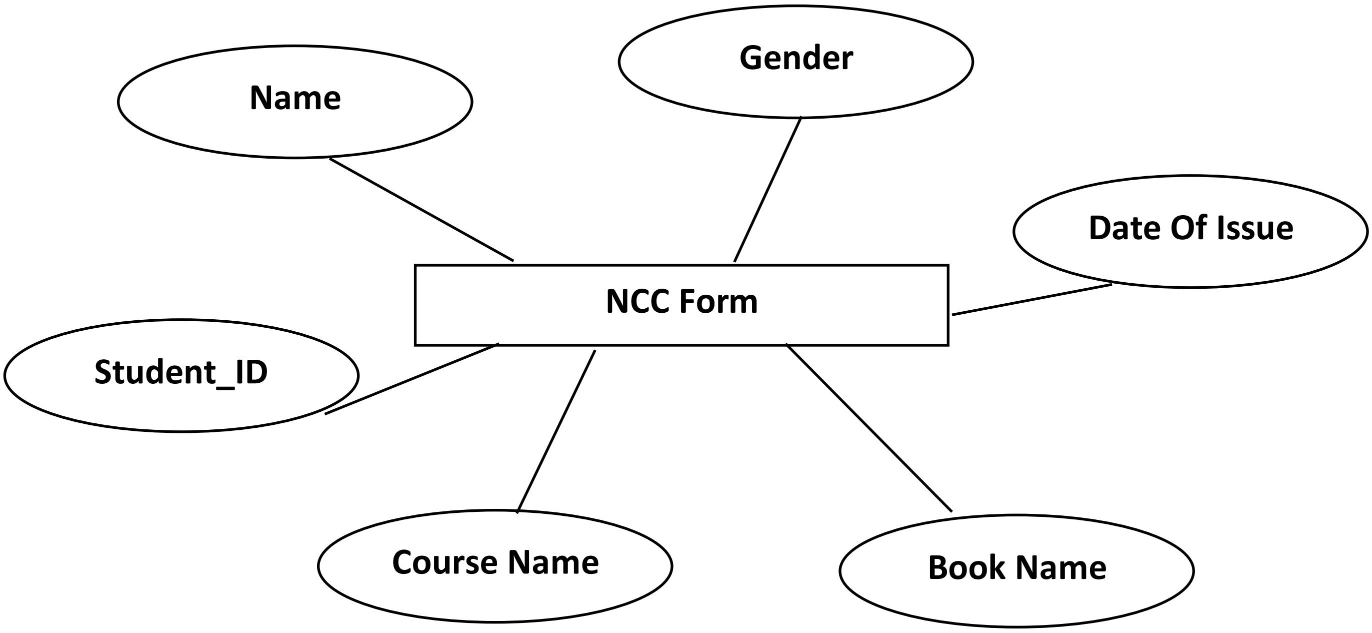
ENTITY RELATIONSHIP DIAGRAM






Student_ID
Name
Fees Forms
Date
Amount


Department
Dep_Name
Dep_Durati
Dep_Descripti


Gender
Name
Date Of Issue
NCC Form
Student_ID
Course Name
Book Name

SYSTEM DESING

SOURES CODE:-

SPLASH FORM :-

```
Public Class Form1
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles
Timer1.Tick
        ProgressBar1.Increment(1)
        If ProgressBar1.Value = 100 Then
            Me.Hide()
            Dim log = New Login
            log.Show()
            Timer1.Enabled = False
        End If
    End Sub

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Timer1.Start()
    End Sub

    Private Sub btnLogo_Click(sender As Object, e As EventArgs) Handles
btnLogo.Click

    End Sub
End Class
```

COLLEGE GOGIN FORM :-

```
Public Class Login
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub btnLogin_Click(sender As Object, e As EventArgs) Handles
btnLogin.Click
        If txtUsername.Text = "" Or txtPassword.Text = "" Then
            MsgBox("Please Enter Username & Password")
        ElseIf txtUsername.Text = "College" And txtPassword.Text = "Password"
Then
            Dim Obj = New Dashboard
            Obj.Show()
            Me.Hide()
        Else
            MsgBox("Wrong Username & Password")
            txtUsername.Text = ""
            txtPassword.Text = ""
        End If
    End Sub

    Private Sub btnLibrary_Click(sender As Object, e As EventArgs) Handles
btnLibrary.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub txtUsername_OnValueChanged(sender As Object, e As EventArgs)
Handles txtUsername.OnValueChanged

    End Sub

    Private Sub btnLogo_Click(sender As Object, e As EventArgs) Handles
btnLogo.Click

    End Sub
End class
```

Dashboard:-

```
Imports System.Data.SqlClient
Public Class Dashboard
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub CountStudents()
        Dim StNum As Integer
        Con.Open()
        Dim sql = "select COUNT(*) from StudentsTbl"
        Dim cmd As SqlCommand
        cmd = New SqlCommand(sql, Con)
        StNum = cmd.ExecuteScalar
        LblNOS.Text = StNum
        Con.Close()
    End Sub
    Private Sub CountTeachers()
        Dim TNum As Integer
        Con.Open()
        Dim sql = "select COUNT(*) from TeachersTbl"
        Dim cmd As SqlCommand
        cmd = New SqlCommand(sql, Con)
        TNum = cmd.ExecuteScalar
        LblNOT.Text = TNum
        Con.Close()
    End Sub
    Private Sub CountDepartments()
        Dim DeptNum As Integer
        Con.Open()
        Dim sql = "select COUNT(*) from DepartmentsTbl"
        Dim cmd As SqlCommand
        cmd = New SqlCommand(sql, Con)
        DeptNum = cmd.ExecuteScalar
        LblNOD.Text = DeptNum
        Con.Close()
    End Sub
    Private Sub SumFees()
        Dim FeesAmount As Integer
        Con.Open()
        Dim sql = "select Sum(Amount) from FeesTbl"
        Dim cmd As SqlCommand
        cmd = New SqlCommand(sql, Con)
        FeesAmount = cmd.ExecuteScalar
        LblNOP.Text = FeesAmount
        Con.Close()
    End Sub

    Private Sub Dashboard_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        CountStudents()
        CountTeachers()
        CountDepartments()
        SumFees()
    End Sub

    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub btnTeachers_Click(sender As Object, e As EventArgs) Handles
btnTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblTeachers_Click(sender As Object, e As EventArgs) Handles
lblTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnStudents_Click(sender As Object, e As EventArgs) Handles
btnStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblStudents_Click(sender As Object, e As EventArgs) Handles
lblStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnFeesDeposit_Click(sender As Object, e As EventArgs) Handles
btnFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblFeesDeposit_Click(sender As Object, e As EventArgs) Handles
lblFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDepartments_Click(sender As Object, e As EventArgs) Handles
btnDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDepartments_Click(sender As Object, e As EventArgs) Handles
lblDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub Panel4_Paint(sender As Object, e As PaintEventArgs) Handles
Panel4.Paint

    End Sub
End Class
```

TEACHERS FORM:-

```
Imports System.Data.SqlClient
Public Class Teachers
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub FillDepartment()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        TDepartmentsCb.DataSource = Tbl
        TDepartmentsCb.DisplayMember = "DeptName"
        TDepartmentsCb.ValueMember = "DeptName"
        Con.Close()
    End Sub
    Private Sub DisplayTeachers()
        Con.Open()
        Dim query = "select * from TeachersTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        TeachersDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        TNameTb.Text = ""
        TGenderCb.SelectedIndex = 0
        TMobileNoTb.Text = ""
        TDepartmentsCb.SelectedIndex = 0
        TAddressTb.Text = ""
    End Sub

    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If TNameTb.Text = "" Or TAddressTb.Text = "" Or TMobileNoTb.Text = ""
Or TGenderCb.SelectedIndex = -1 Or TDepartmentsCb.SelectedIndex = -1 Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into TeachersTbl values('" & TNameTb.Text &
"','" & TGenderCb.SelectedItem.ToString() & "','" & TDOB.Value.Date & "','" &
TMobileNoTb.Text & "','" & TDepartmentsCb.SelectedValue.ToString() & "','" &
TAddressTb.Text & "')"
```

```
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Teacher saved Successfully")
                Con.Close()
                DisplayTeachers()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If

    End Sub

    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub Teachers_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        DisplayTeachers()
        FillDepartment()
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If TNameTb.Text = "" Or TAddressTb.Text = "" Or TMobileNoTb.Text = ""
Or TGenderCb.SelectedIndex = -1 Or TDepartmentsCb.SelectedIndex = -1 Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update TeachersTbl set TName='" & TNameTb.Text &
"',TGender='" & TGenderCb.SelectedItem.ToString() & "',TDOB='" & TDOB.Text &
"',TPhone='" & TMobileNoTb.Text & "',TDept='" &
TDepartmentsCb.SelectedValue.ToString() & "',TAdd='" & TAddressTb.Text & "'
where TId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Teacher updated Successfully")
                Con.Close()
                DisplayTeachers()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
```

```
            MsgBox("Please Select the Teacher to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from TeachersTbl where TId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Teacher Deleted Successfully")
                Con.Close()
                DisplayTeachers()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub
    Dim Key = 0
    Private Sub TeachersDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles TeachersDGV.CellContentClick
        Dim row As DataGridViewRow = TeachersDGV.Rows(e.RowIndex)
        TNameTb.Text = row.Cells(1).Value.ToString
        TGenderCb.SelectedItem = row.Cells(2).Value.ToString
        TDOB.Text = row.Cells(3).Value.ToString
        TMobileNoTb.Text = row.Cells(4).Value.ToString
        TDepartmentsCb.SelectedValue = row.Cells(5).Value.ToString
        TAddressTb.Text = row.Cells(6).Value.ToString
        If TNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If

    End Sub

    Private Sub btnStudents_Click(sender As Object, e As EventArgs) Handles
btnStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblStudents_Click(sender As Object, e As EventArgs) Handles
lblStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnFeesDeposit_Click(sender As Object, e As EventArgs) Handles
btnFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblFeesDeposit_Click(sender As Object, e As EventArgs) Handles
lblFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDepartments_Click(sender As Object, e As EventArgs) Handles
btnDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDepartments_Click(sender As Object, e As EventArgs) Handles
lblDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblDashboard_Click(sender As Object, e As EventArgs) Handles
lblDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub
End Class
```

STUDNT FORMS:-

```
Imports System.Data.SqlClient
Public Class Students
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub Students_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        FillDepartment()
        DisplayStudents()
    End Sub

    Private Sub BunifuImageButton1_Click(sender As Object, e As EventArgs)
Handles btnClose.Click
        Application.Exit()
    End Sub
    Private Sub FillDepartment()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StDepartmentCb.DataSource = Tbl
        StDepartmentCb.DisplayMember = "DeptName"
        StDepartmentCb.ValueMember = "DeptName"
        Con.Close()
    End Sub
    Private Sub DisplayStudents()
        Con.Open()
        Dim query = "select * from StudentsTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        StudentsDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub NoDueList()
        Con.Open()
        Dim query = "select * from StudentsTbl where StFees < 100000"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        StudentsDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        StNameTb.Text = ""
        StFeesTb.Text = ""
        StMobileNoTb.Text = ""
        StGenderCb.SelectedIndex = 0
        StDepartmentCb.SelectedIndex = 0
    End Sub

    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If StNameTb.Text = "" Or StFeesTb.Text = "" Or StMobileNoTb.Text = ""
Or StGenderCb.SelectedIndex = -1 Or StDepartmentCb.SelectedIndex = -1 Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into StudentsTbl values('" & StNameTb.Text
& "','" & StGenderCb.SelectedItem.ToString() & "','" & StDOB.Value.Date & "','"
& StMobileNoTb.Text & "','" & StDepartmentCb.SelectedValue.ToString() & "'," &
StFeesTb.Text & ")"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student saved Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
    Dim Key = 0
    Private Sub StudentsDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles StudentsDGV.CellContentClick
        Dim row As DataGridViewRow = StudentsDGV.Rows(e.RowIndex)
        StNameTb.Text = row.Cells(1).Value.ToString
        StGenderCb.SelectedItem = row.Cells(2).Value.ToString
        StDOB.Text = row.Cells(3).Value.ToString
        StMobileNoTb.Text = row.Cells(4).Value.ToString
        StDepartmentCb.SelectedValue = row.Cells(5).Value.ToString
        StFeesTb.Text = row.Cells(6).Value.ToString
        If StNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If StNameTb.Text = "" Or StFeesTb.Text = "" Or StMobileNoTb.Text = ""
Or StGenderCb.SelectedIndex = -1 Or StDepartmentCb.SelectedIndex = -1 Then
```

```
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update StudentsTbl set StName='" & StNameTb.Text &
"',StGender='" & StGenderCb.SelectedItem.ToString() & "',StDOB='" & StDOB.Text
& "',StPhone='" & StMobileNoTb.Text & "',StDept='" &
StDepartmentCb.SelectedValue.ToString() & "',StFees=" & StFeesTb.Text & " where
StId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student updated Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
            MsgBox("Please Select the Student to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from StudentsTbl where StId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Deleted Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub

    Private Sub btnReload_Click(sender As Object, e As EventArgs) Handles
btnReload.Click
        DisplayStudents()
    End Sub
    Private Sub btnNoDueList_Click(sender As Object, e As EventArgs) Handles
btnNoDueList.Click
        NoDueList()
    End Sub

    Private Sub StDepartmentCb_SelectionChangeCommitted(sender As Object, e As
EventArgs) Handles StDepartmentCb.SelectionChangeCommitted

    End Sub

    Private Sub btnTeachers_Click(sender As Object, e As EventArgs) Handles
btnTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblTeachers_Click(sender As Object, e As EventArgs) Handles
lblTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnFeesDeposit_Click(sender As Object, e As EventArgs) Handles
btnFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblFeesDeposit_Click(sender As Object, e As EventArgs) Handles
lblFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDepartments_Click(sender As Object, e As EventArgs) Handles
btnDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDepartments_Click(sender As Object, e As EventArgs) Handles
lblDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
```

```
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDashboard_Click(sender As Object, e As EventArgs) Handles
lblDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
 End Sub
End Class
```

FEES DEPOSIT:-

```
Imports System.Data.SqlClient
Public Class Fees_Deposit
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub Fees_Deposit_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        DisplayPayments()
        FillStudents()
    End Sub

    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub
    Private Sub FillStudents()
        Con.Open()
        Dim query = "select * from StudentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StIdCb.DataSource = Tbl
        StIdCb.DisplayMember = "StId"
        StIdCb.ValueMember = "StId"
        Con.Close()
    End Sub
    Private Sub DisplayPayments()
        Con.Open()
        Dim query = "select * from FeesTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        FeesDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        AmountTb.Text = ""
        StNameTb.Text = ""
        StIdCb.SelectedIndex = -1
    End Sub
    Private Sub UpdateStudents()
        Try
            Con.Open()
            Dim query = "update StudentsTbl set StFees=" & AmountTb.Text & "
where StId=" & StIdCb.SelectedValue.ToString() & ""
            Dim cmd As SqlCommand
            cmd = New SqlCommand(query, Con)
            cmd.ExecuteNonQuery()
            MsgBox("Fees updated Successfully")
            Con.Close()
            DisplayPayments()
            Reset()
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub GetStName()
        Con.Open()
        Dim query = "select * from StudentsTbl where StId=" &
StIdCb.SelectedValue.ToString() & ""
        Dim cmd As New SqlCommand(query, Con)
        Dim dt As New DataTable
        Dim reader As SqlDataReader
        reader = cmd.ExecuteReader()
        While reader.Read
            StNameTb.Text = reader(1).ToString()
        End While
        Con.Close()
    End Sub
    Private Sub btnPay_Click(sender As Object, e As EventArgs) Handles
btnPay.Click
        If StNameTb.Text = "" Or AmountTb.Text = "" Then
            MsgBox("Missing Information .... !")
        ElseIf Convert.ToInt32(AmountTb.Text) > 100000 Or
Convert.ToInt32(AmountTb.Text) < 1 Then
            MsgBox("Wrong Amount")
        Else
            Try
                Con.Open()
                Dim query = "insert into FeesTbl values(" &
StIdCb.SelectedValue.ToString() & ",'" & StNameTb.Text & "','" &
Period.Value.Date & "'," & AmountTb.Text & ")"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Fees Paid Successfully")
                Con.Close()
                DisplayPayments()
                UpdateStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub StIdCb_SelectionChangeCommitted(sender As Object, e As
EventArgs) Handles StIdCb.SelectionChangeCommitted
        GetStName()
    End Sub
    Private Sub btnTeachers_Click(sender As Object, e As EventArgs) Handles
btnTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblTeachers_Click(sender As Object, e As EventArgs) Handles
lblTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnStudents_Click(sender As Object, e As EventArgs) Handles
btnStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblStudents_Click(sender As Object, e As EventArgs) Handles
lblStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDepartments_Click(sender As Object, e As EventArgs) Handles
btnDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDepartments_Click(sender As Object, e As EventArgs) Handles
lblDepartments.Click
        Dim Obj = New Departments()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDashboard_Click(sender As Object, e As EventArgs) Handles
lblDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub
```

DEPARTMENTS FORM:-

```
Imports System.Data.SqlClient
Public Class Departments
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub Departments_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        DisplayDepartments()
    End Sub
    Private Sub DisplayDepartments()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        DepartmentsDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        DeptNameTb.Text = ""
        DeptDescTb.Text = ""
        DeptDurationTb.Text = ""
    End Sub
    Private Sub BunifuImageButton1_Click(sender As Object, e As EventArgs)
Handles BunifuImageButton1.Click
        Application.Exit()
    End Sub

    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If DeptNameTb.Text = "" Or DeptDescTb.Text = "" Or DeptDurationTb.Text
= "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into DepartmentsTbl values('" &
DeptNameTb.Text & "','" & DeptDescTb.Text & "'," & DeptDurationTb.Text & ")"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Deparment saved Successfully")
                Con.Close()
                DisplayDepartments()
                Reset()
            Catch ex As Exception
```

```
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If DeptNameTb.Text = "" Or DeptDescTb.Text = "" Or DeptDurationTb.Text
= "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update DepartmentsTbl set DeptName='" &
DeptNameTb.Text & "',DeptDesc='" & DeptDescTb.Text & "',DeptDuration=" &
DeptDurationTb.Text & " where DeptId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Deparment updated Successfully")
                Con.Close()
                DisplayDepartments()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
    Dim Key = 0
    Private Sub DepartmentsDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles DepartmentsDGV.CellContentClick
        Dim row As DataGridViewRow = DepartmentsDGV.Rows(e.RowIndex)
        DeptNameTb.Text = row.Cells(1).Value.ToString
        DeptDescTb.Text = row.Cells(2).Value.ToString
        DeptDurationTb.Text = row.Cells(3).Value.ToString
        If DeptNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
            MsgBox("Please Select the Department to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from DepartmentsTbl where DeptId=" & Key &
""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Deparment Deleted Successfully")
                Con.Close()
                DisplayDepartments()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub

    Private Sub btnTeachers_Click(sender As Object, e As EventArgs) Handles
btnTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblTeachers_Click(sender As Object, e As EventArgs) Handles
lblTeachers.Click
        Dim Obj = New Teachers()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnStudents_Click(sender As Object, e As EventArgs) Handles
btnStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblStudents_Click(sender As Object, e As EventArgs) Handles
lblStudents.Click
        Dim Obj = New Students()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnFeesDeposit_Click(sender As Object, e As EventArgs) Handles
btnFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblFeesDeposit_Click(sender As Object, e As EventArgs) Handles
lblFeesDeposit.Click
        Dim Obj = New Fees_Deposit()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDashboard_Click(sender As Object, e As EventArgs) Handles
lblDashboard.Click
        Dim Obj = New Dashboard()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub
End Class
```

LIBRARY LOGIN FORM:-

```
Public Class LibraryLogin
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub btnLogin_Click(sender As Object, e As EventArgs) Handles
btnLogin.Click
        If txtUsername.Text = "" Or txtPassword.Text = "" Then
            MsgBox("Please Enter Username & Password")
        ElseIf txtUsername.Text = "Library" And txtPassword.Text = "Password"
Then
            Dim Obj = New Library
            Obj.Show()
            Me.Hide()
        Else
            MsgBox("Wrong Username & Password")
            txtUsername.Text = ""
            txtPassword.Text = ""
        End If
    End Sub

    Private Sub btnCollege_Click(sender As Object, e As EventArgs) Handles
btnCollege.Click
        Dim Obj = New Login()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub txtUsername_OnValueChanged(sender As Object, e As EventArgs)
Handles txtUsername.OnValueChanged

    End Sub
End Class
```

NCC FORM:-

```
Imports System.Data.SqlClient
Public Class NCC_Form
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDashboard_Click(sender As Object, e As EventArgs) Handles
lblDashboard.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblBonafide_Click(sender As Object, e As EventArgs) Handles
lblBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnBonafide_Click(sender As Object, e As EventArgs) Handles
btnBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblAttendance_Click(sender As Object, e As EventArgs) Handles
lblAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnAttendance_Click(sender As Object, e As EventArgs) Handles
btnAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblSalary_Click(sender As Object, e As EventArgs) Handles
lblSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnSalary_Click(sender As Object, e As EventArgs) Handles
btnSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub FillDepartment()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        CourseCb.DataSource = Tbl
        CourseCb.DisplayMember = "DeptName"
        CourseCb.ValueMember = "DeptName"
        Con.Close()
    End Sub
    Private Sub DisplayStudents()
        Con.Open()
        Dim query = "select * from NCCTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        NCCFormDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        StNameTb.Text = ""
        CourseCb.SelectedIndex = 0
        SectionCb.SelectedIndex = 0
        CasteCategoryCb.SelectedIndex = 0
        BloodGroupCb.SelectedIndex = 0
        StateTb.Text = ""
        PermanantAddressTb.Text = ""
    End Sub
    Private Sub GetStName()
        Con.Open()
        Dim query = "select * from StudentsTbl where StId=" &
StRegNo.SelectedValue.ToString() & ""
        Dim cmd As New SqlCommand(query, Con)
        Dim dt As New DataTable
        Dim reader As SqlDataReader
        reader = cmd.ExecuteReader()
        While reader.Read
            StNameTb.Text = reader(1).ToString()
        End While
        Con.Close()
    End Sub
    Private Sub FillStudents()
        Con.Open()
        Dim query = "select * from StudentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StRegNo.DataSource = Tbl
        StRegNo.DisplayMember = "StId"
        StRegNo.ValueMember = "StId"
        Con.Close()
    End Sub
    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If StNameTb.Text = "" Or CourseCb.SelectedIndex = -1 Or
SectionCb.SelectedIndex = -1 Or CasteCategoryCb.SelectedIndex = -1 Or
BloodGroupCb.SelectedIndex = -1 Or StateTb.Text = "" Or PermanantAddressTb.Text
= "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into NCCTbl values(" &
StRegNo.SelectedValue.ToString() & ",'" & StNameTb.Text & "','" &
CourseCb.SelectedValue.ToString() & "','" & SectionCb.SelectedItem.ToString() &
"','" & DOB.Value.Date & "','" & CasteCategoryCb.SelectedItem.ToString() &
"','" & BloodGroupCb.SelectedItem.ToString() & "','" & StateTb.Text & "','" &
PermanantAddressTb.Text & "')"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record saved Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub NCC_Form_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        FillDepartment()
        DisplayStudents()
        FillStudents()
    End Sub

    Private Sub StRegNo_SelectionChangeCommitted(sender As Object, e As
EventArgs) Handles StRegNo.SelectionChangeCommitted
        GetStName()
    End Sub
    Dim Key = 0
    Private Sub NCCFormDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles NCCFormDGV.CellContentClick
        Dim row As DataGridViewRow = NCCFormDGV.Rows(e.RowIndex)
        StRegNo.Text = row.Cells(1).Value.ToString
        StNameTb.Text = row.Cells(2).Value.ToString
        CourseCb.SelectedItem = row.Cells(3).Value.ToString
        SectionCb.SelectedItem = row.Cells(4).Value.ToString
        DOB.Text = row.Cells(5).Value.ToString
        CasteCategoryCb.SelectedValue = row.Cells(6).Value.ToString
        BloodGroupCb.SelectedValue = row.Cells(7).Value.ToString
        StateTb.Text = row.Cells(8).Value.ToString
        PermanantAddressTb.Text = row.Cells(9).Value.ToString
        If StNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
            MsgBox("Please Select the Student to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from NCCTbl where NCCId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record Deleted Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If StNameTb.Text = "" Or CourseCb.SelectedIndex = -1 Or
SectionCb.SelectedIndex = -1 Or CasteCategoryCb.SelectedIndex = -1 Or
BloodGroupCb.SelectedIndex = -1 Or StateTb.Text = "" Or PermanantAddressTb.Text
= "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update NCCTbl set StudentName='" & StNameTb.Text &
"',Course='" & CourseCb.SelectedValue.ToString() & "',Section='" &
SectionCb.SelectedItem.ToString() & "',DOB='" & DOB.Text & "',CasteCategory='"
& CasteCategoryCb.SelectedItem.ToString() & "',BloodGroup='" &
BloodGroupCb.SelectedItem.ToString() & "',State='" & StateTb.Text &
"',Address='" & PermanantAddressTb.Text & "' where NCCId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record updated Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub
End Class
```

SALARY FORM:-

```
Imports System.Data.SqlClient
Public Class Salary
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub
    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnLibrary_Click(sender As Object, e As EventArgs) Handles
btnLibrary.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblLibrary_Click(sender As Object, e As EventArgs) Handles
lblLibrary.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblBonafide_Click(sender As Object, e As EventArgs) Handles
lblBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnBonafide_Click(sender As Object, e As EventArgs) Handles
btnBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblAttendance_Click(sender As Object, e As EventArgs) Handles
lblAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnAttendance_Click(sender As Object, e As EventArgs) Handles
btnAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblNCC_Click(sender As Object, e As EventArgs) Handles
lblNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnNCC_Click(sender As Object, e As EventArgs) Handles
btnNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub DisplayStaff()
        Con.Open()
        Dim query = "select * from StaffTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        StaffDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        StaffNameTb.Text = ""
        QualificationCb.SelectedIndex = 0
        AddressTb.Text = ""
        PostCb.SelectedIndex = 0
        BasicSalaryTb.Text = ""
        DATb.Text = ""
        TATb.Text = ""
        HRATb.Text = ""
        PFTb.Text = ""
        ESITb.Text = ""
        TotalSalaryTb.Text = ""
    End Sub
    Private Sub Salary_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        DisplayStaff()
    End Sub
    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If StaffNameTb.Text = "" Or QualificationCb.SelectedIndex = -1 Or
AddressTb.Text = "" Or PostCb.SelectedIndex = -1 Or BasicSalaryTb.Text = "" Or
```

```
DATb.Text = "" Or TATb.Text = "" Or HRATb.Text = "" Or PFTb.Text = "" Or
ESITb.Text = "" Or TotalSalaryTb.Text = "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into StaffTbl values('" & StaffNameTb.Text
& "','" & QualificationCb.SelectedItem.ToString() & "','" & StaffDOB.Value.Date
& "','" & StaffDOJ.Value.Date & "','" & AddressTb.Text & "','" &
PostCb.SelectedItem.ToString() & "','" & BasicSalaryTb.Text & "','" & DATb.Text
& "','" & TATb.Text & "','" & HRATb.Text & "','" & PFTb.Text & "','" &
ESITb.Text & "'," & TotalSalaryTb.Text & ")"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Staff saved Successfully")
                Con.Close()
                DisplayStaff()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub
    Dim Key = 0
    Private Sub StaffDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles StaffDGV.CellContentClick
        Dim row As DataGridViewRow = StaffDGV.Rows(e.RowIndex)
        StaffNameTb.Text = row.Cells(1).Value.ToString
        QualificationCb.SelectedItem = row.Cells(2).Value.ToString
        StaffDOB.Text = row.Cells(3).Value.ToString
        StaffDOB.Text = row.Cells(4).Value.ToString
        AddressTb.Text = row.Cells(5).Value.ToString
        PostCb.SelectedItem = row.Cells(6).Value.ToString
        BasicSalaryTb.Text = row.Cells(7).Value.ToString
        DATb.Text = row.Cells(8).Value.ToString
        TATb.Text = row.Cells(9).Value.ToString
        HRATb.Text = row.Cells(10).Value.ToString
        PFTb.Text = row.Cells(11).Value.ToString
        ESITb.Text = row.Cells(12).Value.ToString
        TotalSalaryTb.Text = row.Cells(13).Value.ToString
        If StaffNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub
    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
```

```
            MsgBox("Please Select the Staff to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from StaffTbl where StaffId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Staff Deleted Successfully")
                Con.Close()
                DisplayStaff()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If StaffNameTb.Text = "" Or QualificationCb.SelectedIndex = -1 Or
AddressTb.Text = "" Or PostCb.SelectedIndex = -1 Or BasicSalaryTb.Text = "" Or
DATb.Text = "" Or TATb.Text = "" Or HRATb.Text = "" Or PFTb.Text = "" Or
ESITb.Text = "" Or TotalSalaryTb.Text = "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update StaffTbl set StaffName='" &
StaffNameTb.Text & "',Qualification='" &
QualificationCb.SelectedItem.ToString() & "',DOB='" & StaffDOB.Text & "',DOJ='"
& StaffDOJ.Text & "',Address='" & AddressTb.Text & "',Post='" &
PostCb.SelectedItem.ToString() & "',BasicSalary='" & BasicSalaryTb.Text &
"',DA='" & DATb.Text & "',TA='" & TATb.Text & "',HRA='" & HRATb.Text & "',PF='"
& PFTb.Text & "',ESI='" & ESITb.Text & "',TotalSalary=" & TotalSalaryTb.Text &
" where StaffId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Staff updated Successfully")
                Con.Close()
                DisplayStaff()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
End Class
```

ATTENDANCE FORM:-

```
Imports System.Data.SqlClient
Public Class Attendance
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnLibrary_Click(sender As Object, e As EventArgs) Handles
btnLibrary.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLibrary_Click(sender As Object, e As EventArgs) Handles
lblLibrary.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblBonafide_Click(sender As Object, e As EventArgs) Handles
lblBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnBonafide_Click(sender As Object, e As EventArgs) Handles
btnBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub btnSalary_Click(sender As Object, e As EventArgs) Handles
btnSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblSalary_Click(sender As Object, e As EventArgs) Handles
lblSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnNCC_Click(sender As Object, e As EventArgs) Handles
btnNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblNCC_Click(sender As Object, e As EventArgs) Handles
lblNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub FillDepartment()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StDepartmentCb.DataSource = Tbl
        StDepartmentCb.DisplayMember = "DeptName"
        StDepartmentCb.ValueMember = "DeptName"
        Con.Close()
    End Sub
    Private Sub DisplayStudents()
        Con.Open()
        Dim query = "select * from AttendanceTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        AttendanceDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        StNameTb.Text = ""
        StMonthCb.SelectedIndex = 0
        StDepartmentCb.SelectedIndex = 0
        PresentTb.Text = ""
        AbsentTb.Text = ""
        TotalAttendanceTb.Text = ""
        AttendancePercentTb.Text = ""
    End Sub
    Private Sub GetStName()
        Con.Open()
        Dim query = "select * from StudentsTbl where StId=" &
StRollNo.SelectedValue.ToString() & ""
        Dim cmd As New SqlCommand(query, Con)
        Dim dt As New DataTable
        Dim reader As SqlDataReader
        reader = cmd.ExecuteReader()
        While reader.Read
            StNameTb.Text = reader(1).ToString()
        End While
        Con.Close()
    End Sub
    Private Sub FillStudents()
        Con.Open()
        Dim query = "select * from StudentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StRollNo.DataSource = Tbl
        StRollNo.DisplayMember = "StId"
        StRollNo.ValueMember = "StId"
        Con.Close()
    End Sub
    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If StNameTb.Text = "" Or StMonthCb.SelectedIndex = -1 Or
StDepartmentCb.SelectedIndex = -1 Or PresentTb.Text = "" Or AbsentTb.Text = ""
Or TotalAttendanceTb.Text = "" Or AttendancePercentTb.Text = "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into AttendanceTbl values(" &
StRollNo.SelectedValue.ToString() & ",'" & StNameTb.Text & "','" &
StMonthCb.SelectedItem.ToString() & "','" &
StDepartmentCb.SelectedValue.ToString() & "','" & PresentTb.Text & "','" &
AbsentTb.Text & "','" & TotalAttendanceTb.Text & "','" &
AttendancePercentTb.Text & "')"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record saved Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub Attendance_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        FillDepartment()
        DisplayStudents()
        FillStudents()
    End Sub

    Private Sub StRollNo_SelectionChangeCommitted(sender As Object, e As
EventArgs) Handles StRollNo.SelectionChangeCommitted
        GetStName()
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If StNameTb.Text = "" Or StMonthCb.SelectedIndex = -1 Or
StDepartmentCb.SelectedIndex = -1 Or PresentTb.Text = "" Or AbsentTb.Text = ""
Or TotalAttendanceTb.Text = "" Or AttendancePercentTb.Text = "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update AttendanceTbl set StudentName='" &
StNameTb.Text & "',Month='" & StMonthCb.SelectedItem.ToString() &
"',DepartmentName='" & StDepartmentCb.SelectedValue.ToString() & "',Present='"
& PresentTb.Text & "',Absent='" & AbsentTb.Text & "',TotalAttendance='" &
TotalAttendanceTb.Text & "',AttendancePercent='" & AttendancePercentTb.Text &
"' where AttendanceId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record updated Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
```

```
            MsgBox("Please Select the Student to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from AttendanceTbl where AttendanceId=" &
Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record Deleted Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
    Dim Key = 0
    Private Sub AttendanceDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles AttendanceDGV.CellContentClick
        Dim row As DataGridViewRow = AttendanceDGV.Rows(e.RowIndex)
        StRollNo.Text = row.Cells(1).Value.ToString
        StNameTb.Text = row.Cells(2).Value.ToString
        StMonthCb.SelectedItem = row.Cells(3).Value.ToString
        StDepartmentCb.SelectedValue = row.Cells(4).Value.ToString
        PresentTb.Text = row.Cells(5).Value.ToString
        AbsentTb.Text = row.Cells(6).Value.ToString
        TotalAttendanceTb.Text = row.Cells(7).Value.ToString
        AttendancePercentTb.Text = row.Cells(8).Value.ToString
        If StNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub
End Class
```

BONAFIDE FORM:-

```
Imports System.Data.SqlClient
Public Class Bonafide
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()

    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblDashboard_Click(sender As Object, e As EventArgs) Handles
lblDashboard.Click
        Dim Obj = New Library()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnAttendance_Click(sender As Object, e As EventArgs) Handles
btnAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblAttendance_Click(sender As Object, e As EventArgs) Handles
lblAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub lblSalary_Click(sender As Object, e As EventArgs) Handles
lblSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnSalary_Click(sender As Object, e As EventArgs) Handles
btnSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblNCC_Click(sender As Object, e As EventArgs) Handles
lblNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnNCC_Click(sender As Object, e As EventArgs) Handles
btnNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub FillDepartment()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        CourseNameCb.DataSource = Tbl
        CourseNameCb.DisplayMember = "DeptName"
        CourseNameCb.ValueMember = "DeptName"
        Con.Close()
    End Sub
    Private Sub DisplayStudents()
        Con.Open()
        Dim query = "select * from BonafideTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        BonafideDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
```

```
        StNameTb.Text = ""
        StGenderCb.SelectedIndex = 0
        CourseNameCb.SelectedIndex = 0
        CategoryNameCb.SelectedIndex = 0
        YearCb.SelectedIndex = 0
        DOBinWordsTb.Text = ""
        PlaceofBirthTb.Text = ""
    End Sub
    Private Sub GetStName()
        Con.Open()
        Dim query = "select * from StudentsTbl where StId=" &
StudentID.SelectedValue.ToString() & ""
        Dim cmd As New SqlCommand(query, Con)
        Dim dt As New DataTable
        Dim reader As SqlDataReader
        reader = cmd.ExecuteReader()
        While reader.Read
            StNameTb.Text = reader(1).ToString()
        End While
        Con.Close()
    End Sub
    Private Sub FillStudents()
        Con.Open()
        Dim query = "select * from StudentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StudentID.DataSource = Tbl
        StudentID.DisplayMember = "StId"
        StudentID.ValueMember = "StId"
        Con.Close()
    End Sub
    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If StNameTb.Text = "" Or StGenderCb.SelectedIndex = -1 Or
CourseNameCb.SelectedIndex = -1 Or CategoryNameCb.SelectedIndex = -1 Or
YearCb.SelectedIndex = -1 Or DOBinWordsTb.Text = "" Or PlaceofBirthTb.Text = ""
Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into BonafideTbl values(" &
StudentID.SelectedValue.ToString() & ",'" & StNameTb.Text & "','" &
StGenderCb.SelectedItem.ToString() & "','" & StDOB.Value.Date & "','" &
StDOI.Value.Date & "','" & CourseNameCb.SelectedValue.ToString() & "','" &
CategoryNameCb.SelectedItem.ToString() & "','" & YearCb.SelectedItem.ToString()
& "','" & DOBinWordsTb.Text & "','" & PlaceofBirthTb.Text & "')"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record saved Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub Bonafide_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        FillDepartment()
        DisplayStudents()
        FillStudents()
    End Sub

    Private Sub StudentID_SelectionChangeCommitted(sender As Object, e As
EventArgs) Handles StudentID.SelectionChangeCommitted
        GetStName()
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If StNameTb.Text = "" Or StGenderCb.SelectedIndex = -1 Or
CourseNameCb.SelectedIndex = -1 Or CategoryNameCb.SelectedIndex = -1 Or
YearCb.SelectedIndex = -1 Or DOBinWordsTb.Text = "" Or PlaceofBirthTb.Text = ""
Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update BonafideTbl set StudentName='" &
StNameTb.Text & "',Gender='" & StGenderCb.SelectedItem.ToString() &
"',DateofBirth='" & StDOB.Text & "',DateofIssue='" & StDOI.Text &
"',CourseName='" & CourseNameCb.SelectedValue.ToString() & "',CategoryName='" &
CategoryNameCb.SelectedItem.ToString() & "',Year='" &
YearCb.SelectedItem.ToString() & "',DOBinWords='" & DOBinWordsTb.Text &
"',PlaceofBirth='" & PlaceofBirthTb.Text & "' where AttendanceId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record updated Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
            MsgBox("Please Select the Student to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from BonafideTbl where BonafideId=" & Key &
""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record Deleted Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub
    Dim Key = 0
    Private Sub BonafideDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles BonafideDGV.CellContentClick
        Dim row As DataGridViewRow = BonafideDGV.Rows(e.RowIndex)
        StudentID.Text = row.Cells(1).Value.ToString
        StNameTb.Text = row.Cells(2).Value.ToString
        StGenderCb.SelectedItem = row.Cells(3).Value.ToString
        StDOB.Text = row.Cells(4).Value.ToString
        StDOI.Text = row.Cells(5).Value.ToString
        CourseNameCb.SelectedValue = row.Cells(6).Value.ToString
        CategoryNameCb.SelectedItem = row.Cells(7).Value.ToString
        YearCb.SelectedItem = row.Cells(8).Value.ToString
        DOBinWordsTb.Text = row.Cells(9).Value.ToString
        PlaceofBirthTb.Text = row.Cells(10).Value.ToString
        If StNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub

    Private Sub BunifuImageButton1_Click(sender As Object, e As EventArgs)
    End Sub
End Class
```

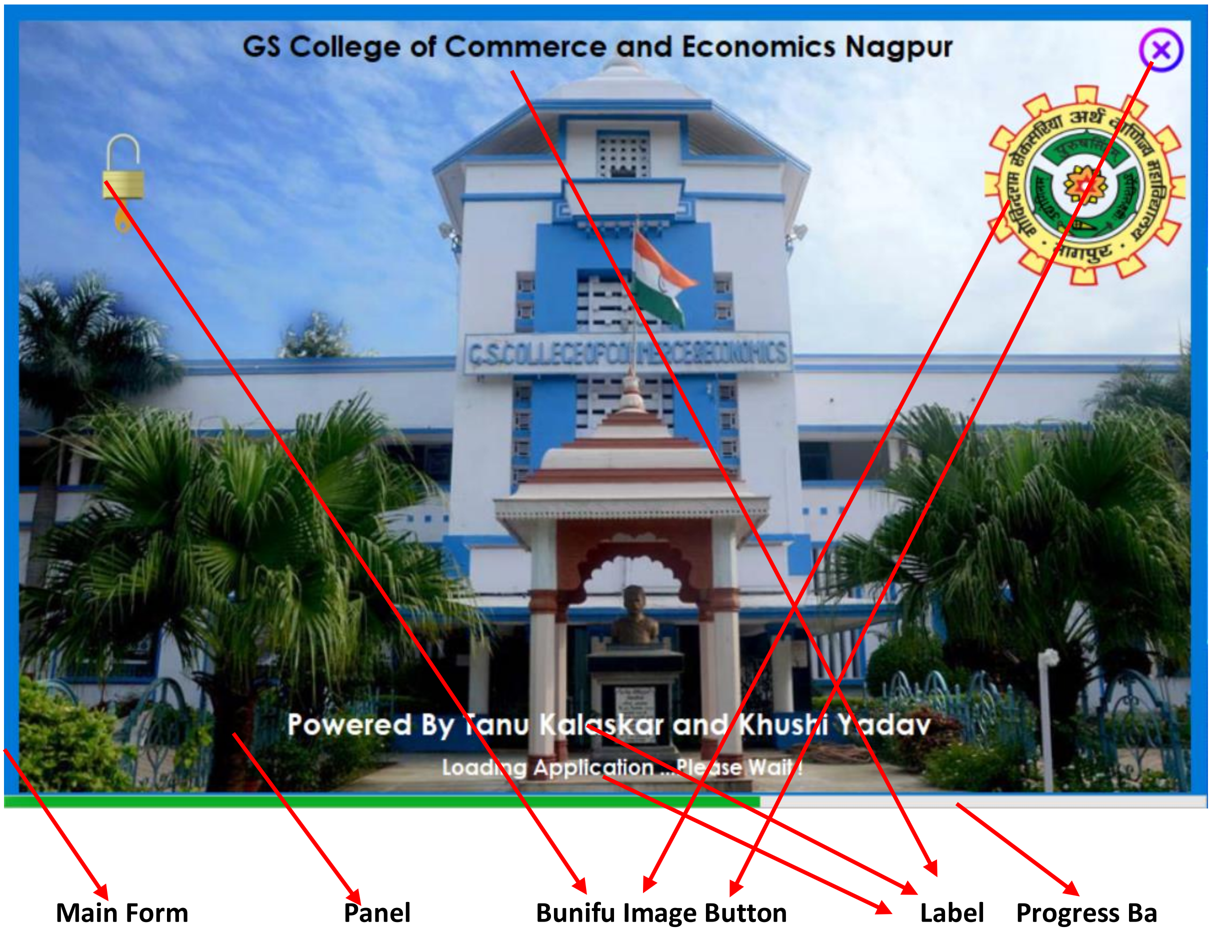
LIBRARY FORM:-

```
Imports System.Data.SqlClient
Public Class Library
    Dim Con = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\College Management
System\College Management System\CMSDatabase.mdf;Integrated Security=True")
    Private Sub btnClose_Click(sender As Object, e As EventArgs) Handles
btnClose.Click
        Application.Exit()
    End Sub

    Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblLogout_Click(sender As Object, e As EventArgs) Handles
lblLogout.Click
        Dim Obj = New LibraryLogin()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnBonafide_Click(sender As Object, e As EventArgs) Handles
btnBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblBonafide_Click(sender As Object, e As EventArgs) Handles
lblBonafide.Click
        Dim Obj = New Bonafide()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnAttendance_Click(sender As Object, e As EventArgs) Handles
btnAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblAttendance_Click(sender As Object, e As EventArgs) Handles
lblAttendance.Click
        Dim Obj = New Attendance()
        Obj.Show()
        Me.Hide()
    End Sub
```

```vbnet
    Private Sub btnSalary_Click(sender As Object, e As EventArgs) Handles
btnSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblSalary_Click(sender As Object, e As EventArgs) Handles
lblSalary.Click
        Dim Obj = New Salary()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub btnNCC_Click(sender As Object, e As EventArgs) Handles
btnNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub

    Private Sub lblNCC_Click(sender As Object, e As EventArgs) Handles
lblNCC.Click
        Dim Obj = New NCC_Form()
        Obj.Show()
        Me.Hide()
    End Sub
    Private Sub FillDepartment()
        Con.Open()
        Dim query = "select * from DepartmentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        CourseNameCb.DataSource = Tbl
        CourseNameCb.DisplayMember = "DeptName"
        CourseNameCb.ValueMember = "DeptName"
        Con.Close()
    End Sub
    Private Sub DisplayStudents()
        Con.Open()
        Dim query = "select * from LibraryTbl"
        Dim adapter As SqlDataAdapter
        Dim cmd = New SqlCommand(query, Con)
        adapter = New SqlDataAdapter(cmd)
        Dim builder = New SqlCommandBuilder(adapter)
        Dim ds As DataSet
        ds = New DataSet
        adapter.Fill(ds)
        LibraryDGV.DataSource = ds.Tables(0)
        Con.Close()
    End Sub
    Private Sub Reset()
        StNameTb.Text = ""
        StGenderCb.SelectedIndex = 0
        CourseNameCb.SelectedIndex = 0
        BookNameTb.Text = ""
    End Sub
    Private Sub GetStName()
        Con.Open()
        Dim query = "select * from StudentsTbl where StId=" &
StudentID.SelectedValue.ToString() & ""
        Dim cmd As New SqlCommand(query, Con)
        Dim dt As New DataTable
        Dim reader As SqlDataReader
        reader = cmd.ExecuteReader()
        While reader.Read
            StNameTb.Text = reader(1).ToString()
        End While
        Con.Close()
    End Sub
    Private Sub FillStudents()
        Con.Open()
        Dim query = "select * from StudentsTbl"
        Dim cmd As New SqlCommand(query, Con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim Tbl As New DataTable()
        adapter.Fill(Tbl)
        StudentID.DataSource = Tbl
        StudentID.DisplayMember = "StId"
        StudentID.ValueMember = "StId"
        Con.Close()
    End Sub
    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If StNameTb.Text = "" Or StGenderCb.SelectedIndex = -1 Or
CourseNameCb.SelectedIndex = -1 Or BookNameTb.Text = "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "insert into LibraryTbl values(" &
StudentID.SelectedValue.ToString() & ",'" & StNameTb.Text & "','" &
StGenderCb.SelectedItem.ToString() & "','" & StDOI.Value.Date & "','" &
CourseNameCb.SelectedValue.ToString() & "','" & BookNameTb.Text & "')"
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record saved Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub
    Private Sub Library_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        FillDepartment()
        DisplayStudents()
        FillStudents()
    End Sub

    Private Sub StudentID_SelectionChangeCommitted(sender As Object, e As
EventArgs) Handles StudentID.SelectionChangeCommitted
        GetStName()
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If StNameTb.Text = "" Or StGenderCb.SelectedIndex = -1 Or
CourseNameCb.SelectedIndex = -1 Or BookNameTb.Text = "" Then
            MsgBox("Missing Information .... !")
        Else
            Try
                Con.Open()
                Dim query = "update LibraryTbl set StudentName='" &
StNameTb.Text & "',Gender='" & StGenderCb.SelectedItem.ToString() &
"',DateofIssue='" & StDOI.Text & "',CourseName='" &
CourseNameCb.SelectedValue.ToString() & "',BookName='" & BookNameTb.Text & "'
where LibraryId=" & Key & ""
                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record updated Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If Key = 0 Then
            MsgBox("Please Select the Student to Delete")
        Else
            Try
                Con.Open()
                Dim query = "delete from LibraryTbl where LibraryId=" & Key &
""

                Dim cmd As SqlCommand
                cmd = New SqlCommand(query, Con)
                cmd.ExecuteNonQuery()
                MsgBox("Student Record Deleted Successfully")
                Con.Close()
                DisplayStudents()
                Reset()
            Catch ex As Exception
                MsgBox(ex.Message)
            End Try
        End If
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles
btnReset.Click
        Reset()
    End Sub
    Dim Key = 0
    Private Sub LibraryDGV_CellContentClick(sender As Object, e As
DataGridViewCellEventArgs) Handles LibraryDGV.CellContentClick
        Dim row As DataGridViewRow = LibraryDGV.Rows(e.RowIndex)
        StudentID.Text = row.Cells(1).Value.ToString
        StNameTb.Text = row.Cells(2).Value.ToString
        StGenderCb.SelectedItem = row.Cells(3).Value.ToString
        StDOI.Text = row.Cells(4).Value.ToString
        CourseNameCb.SelectedValue = row.Cells(5).Value.ToString
        BookNameTb.Text = row.Cells(6).Value.ToString
        If StNameTb.Text = "" Then
            Key = 0
        Else
            Key = Convert.ToInt32(row.Cells(0).Value.ToString)
        End If
    End Sub

    Private Sub StudentID_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles StudentID.SelectedIndexChanged

    End Sub
End Class
```

INPUT SCREENS & OUTPUT SCREENS

Splash Form (Loading Form)

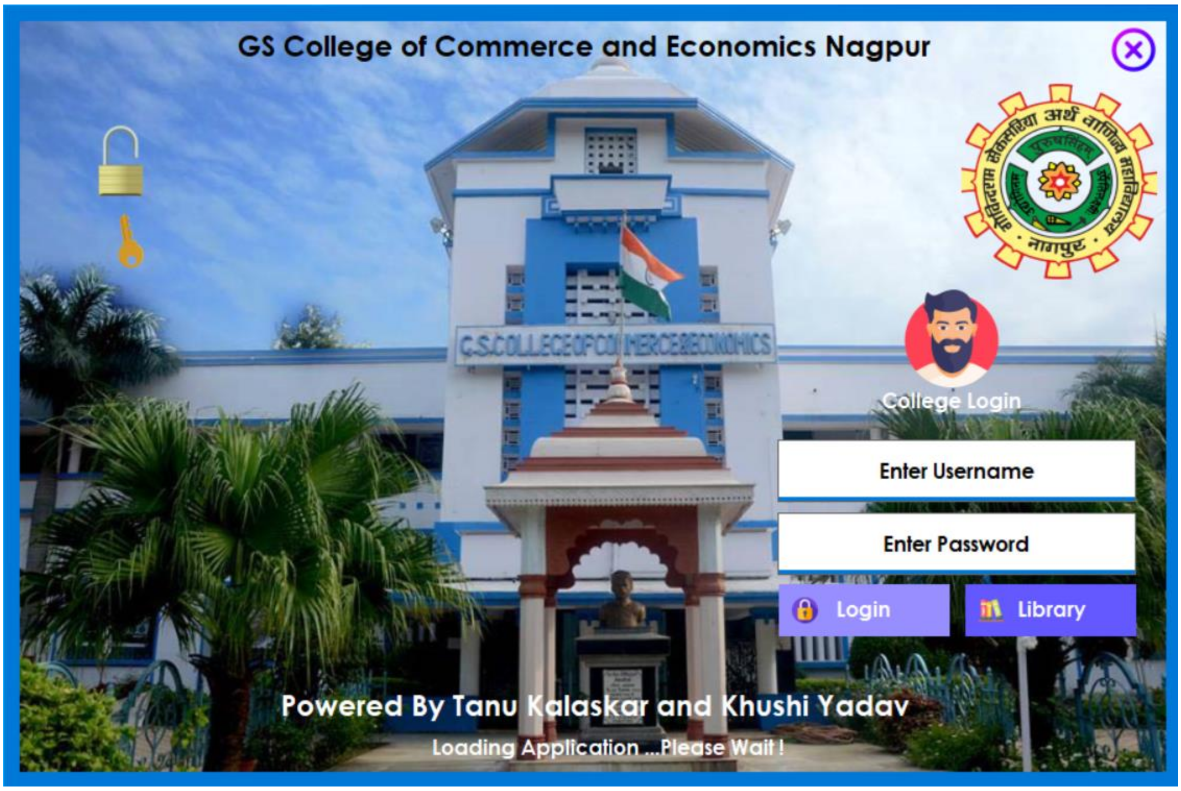


GS College of Commerce and Economics Nagpur

Powered By Tanu Kalaskar and Khushi Yadav

Loading Application ...Please Wait !

Login Form



GS College of Commerce and Economics Nagpur

College Login

Enter Username

Enter Password

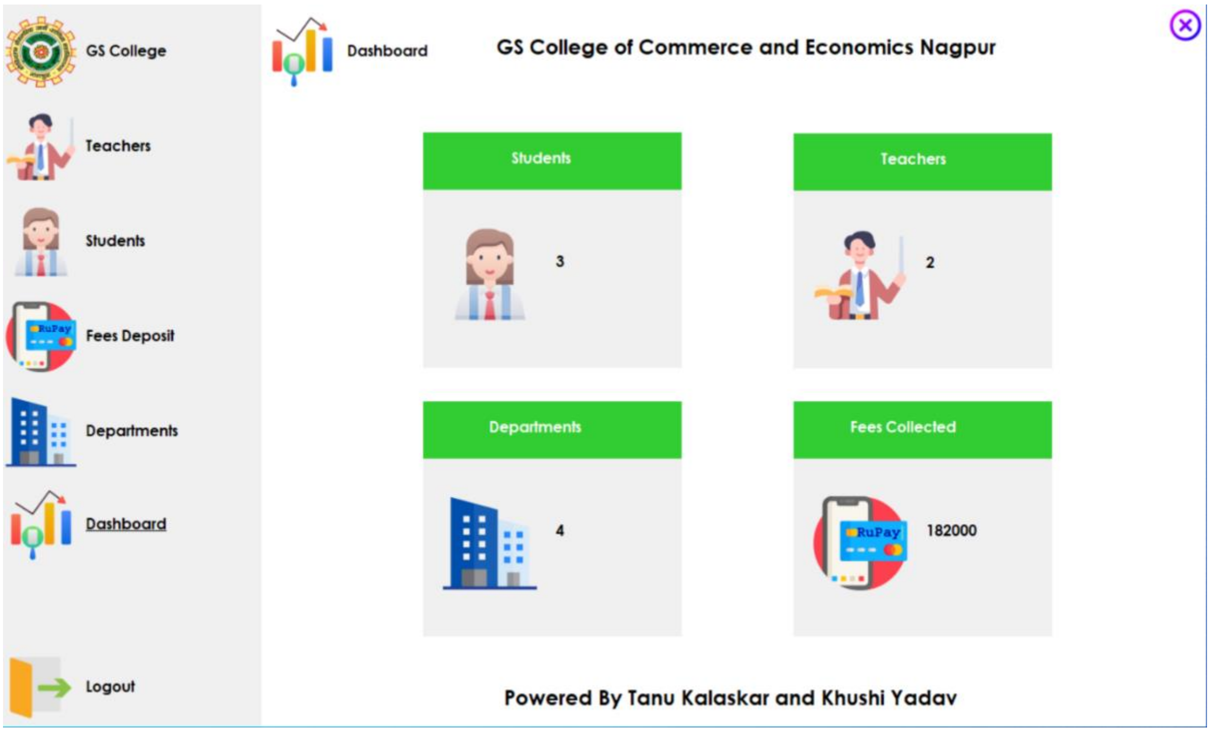
Login

Library

Powered By Tanu Kalaskar and Khushi Yadav

Loading Application ...Please Wait !

Dashboard



GS College

Teachers

Students

Fees Deposit

Departments

Dashboard

Logout

Dashboard

GS College of Commerce and Economics Nagpur

Students 3

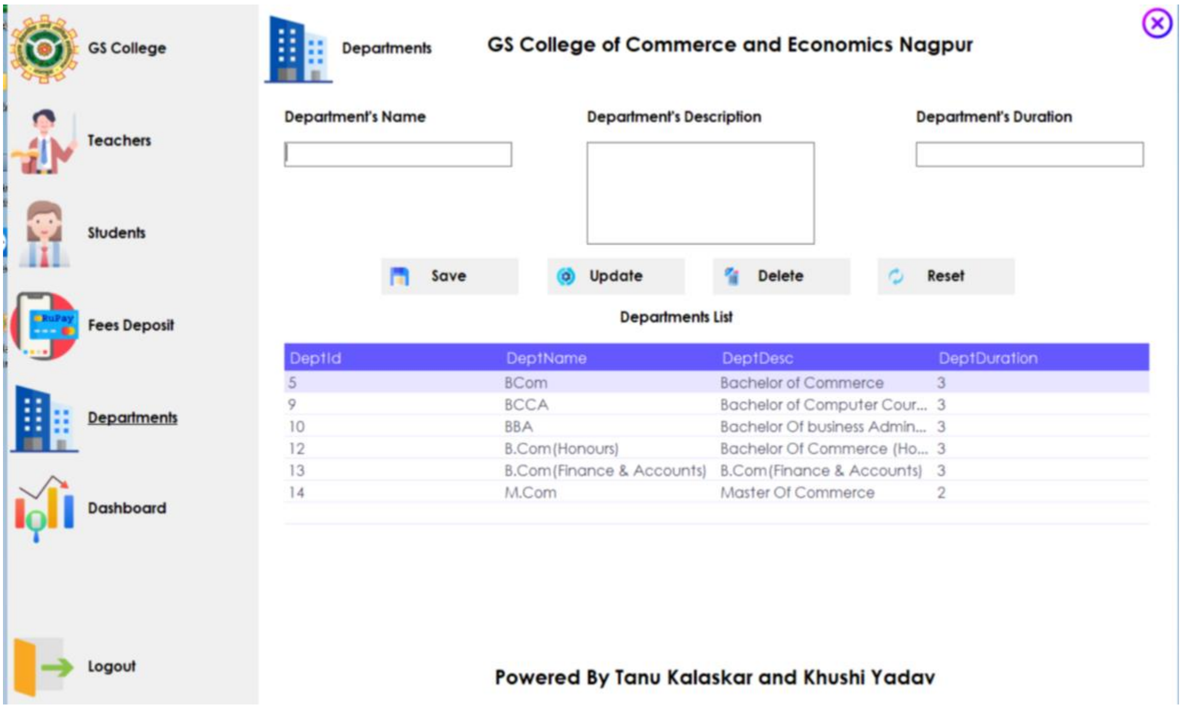
Teachers 2

Departments 4

Fees Collected 182000

Powered By Tanu Kalaskar and Khushi Yadav

Departments



GS College of Commerce and Economics Nagpur

Department's Name

Department's Description

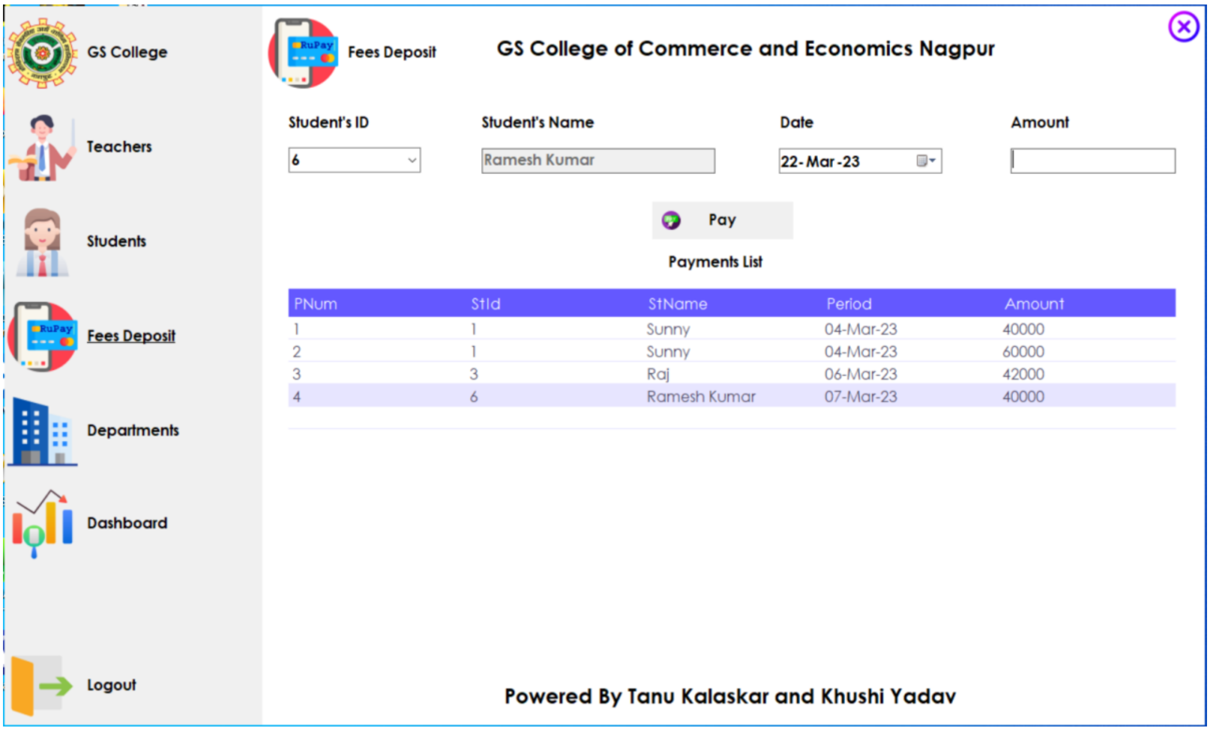
Department's Duration

Departments List

DeptId	DeptName	DeptDesc	DeptDuration
5	BCom	Bachelor of Commerce	3
9	BCCA	Bachelor of Computer Cour...	3
10	BBA	Bachelor Of business Admin...	3
12	B.Com(Honours)	Bachelor Of Commerce (Ho...	3
13	B.Com(Finance & Accounts)	B.Com(Finance & Accounts)	3
14	M.Com	Master Of Commerce	2

Powered By Tanu Kalaskar and Khushi Yadav

Fee Deposit



GS College of Commerce and Economics Nagpur

Student's ID: 6

Student's Name: Ramesh Kumar

Date: 22-Mar-23

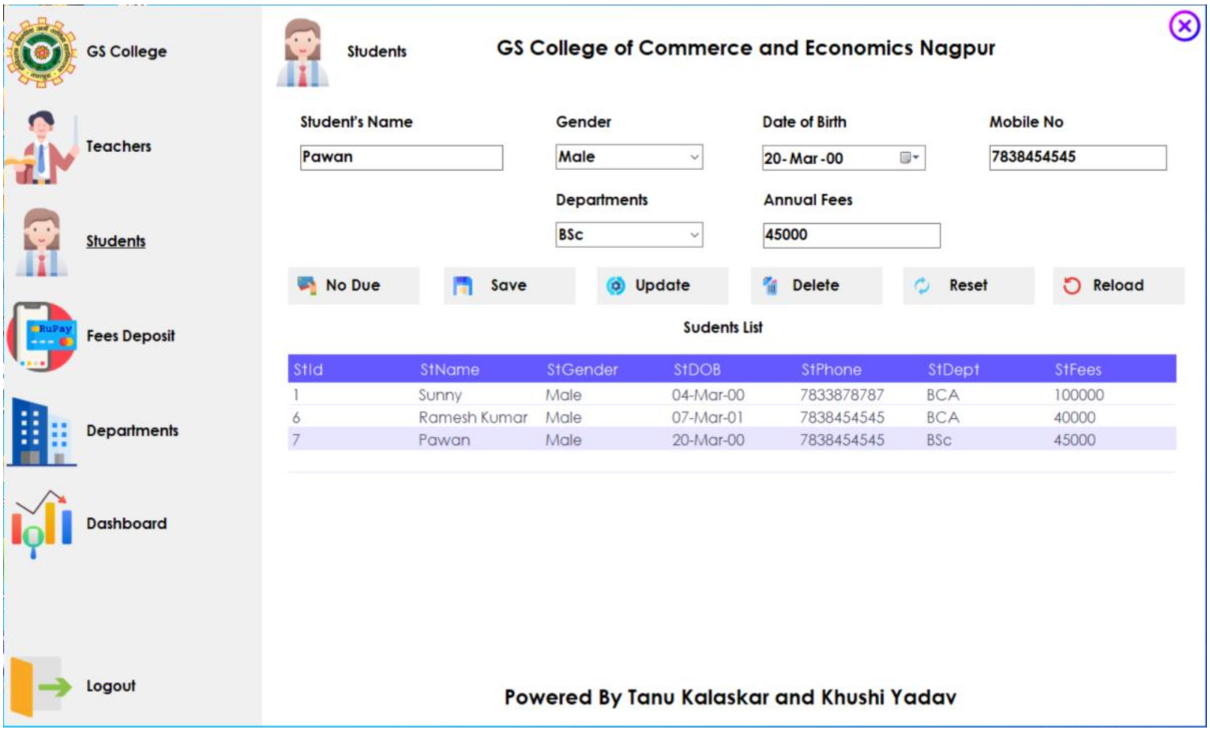
Amount

Payments List

PNum	StId	StName	Period	Amount
1	1	Sunny	04-Mar-23	40000
2	1	Sunny	04-Mar-23	60000
3	3	Raj	06-Mar-23	42000
4	6	Ramesh Kumar	07-Mar-23	40000

Powered By Tanu Kalaskar and Khushi Yadav

Students



GS College of Commerce and Economics Nagpur

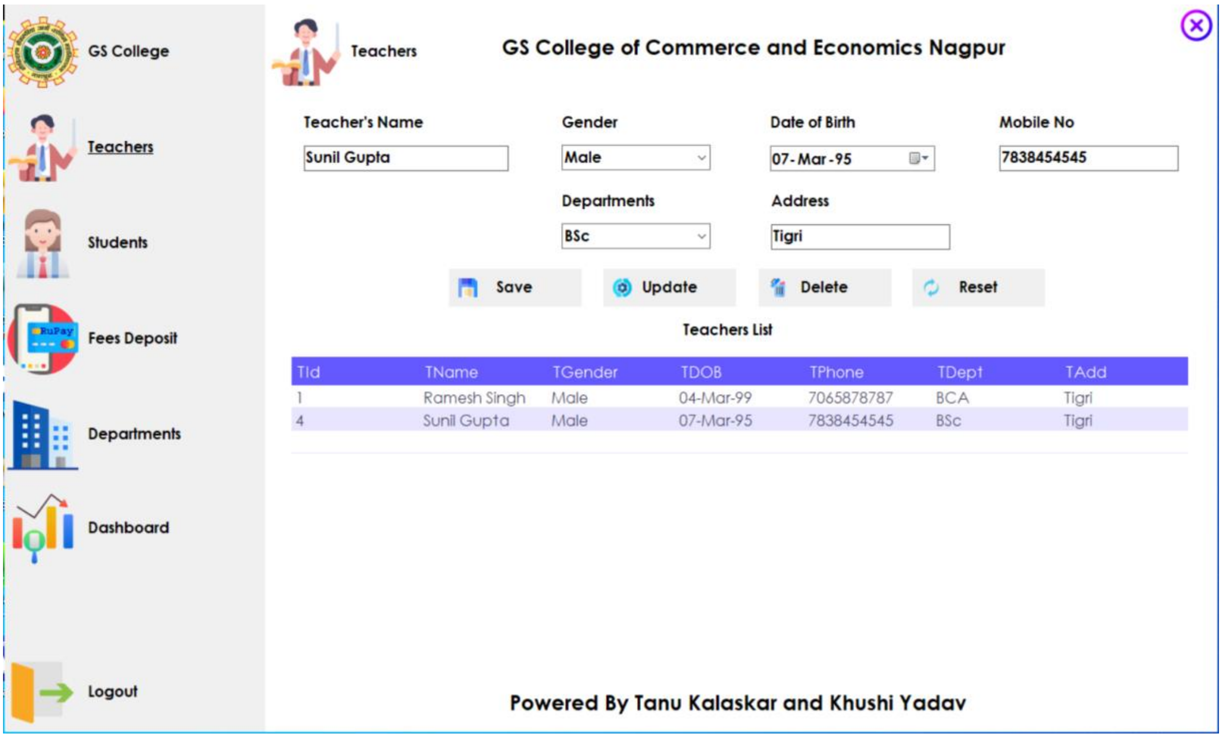
Student's Name	Pawan
Gender	Male
Date of Birth	20-Mar-00
Mobile No	7838454545
Departments	BSc
Annual Fees	45000

Sudents List

StId	StName	StGender	StDOB	StPhone	StDept	StFees
1	Sunny	Male	04-Mar-00	7833878787	BCA	100000
6	Ramesh Kumar	Male	07-Mar-01	7838454545	BCA	40000
7	Pawan	Male	20-Mar-00	7838454545	BSc	45000

Powered By Tanu Kalaskar and Khushi Yadav

Teachers



GS College of Commerce and Economics Nagpur

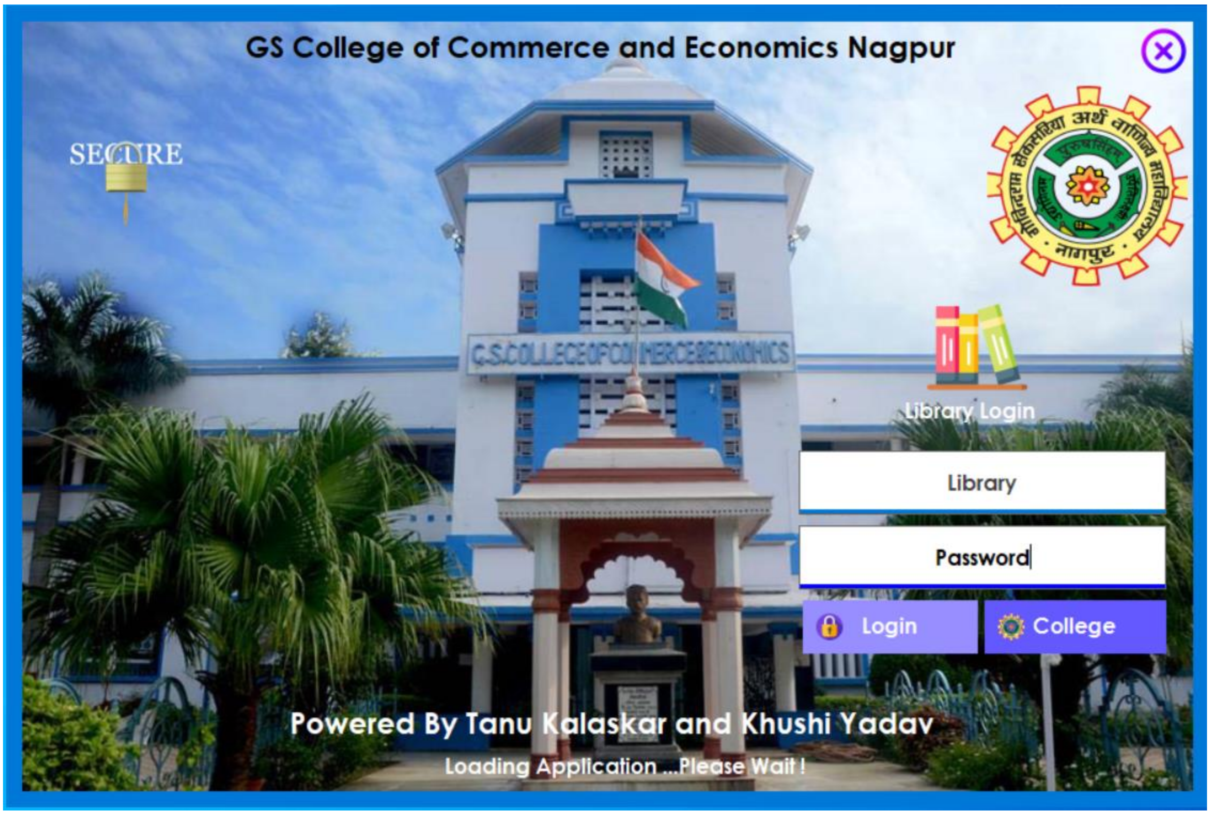
Teacher's Name	Sunil Gupta
Gender	Male
Date of Birth	07-Mar-95
Mobile No	7838454545
Departments	BSc
Address	Tigri

Teachers List

TId	TName	TGender	TDOB	TPhone	TDept	TAdd
1	Ramesh Singh	Male	04-Mar-99	7065878787	BCA	Tigri
4	Sunil Gupta	Male	07-Mar-95	7838454545	BSc	Tigri

Powered By Tanu Kalaskar and Khushi Yadav

GS College of Commerce and Economics Nagpur (Library)



GS College of Commerce and Economics Nagpur

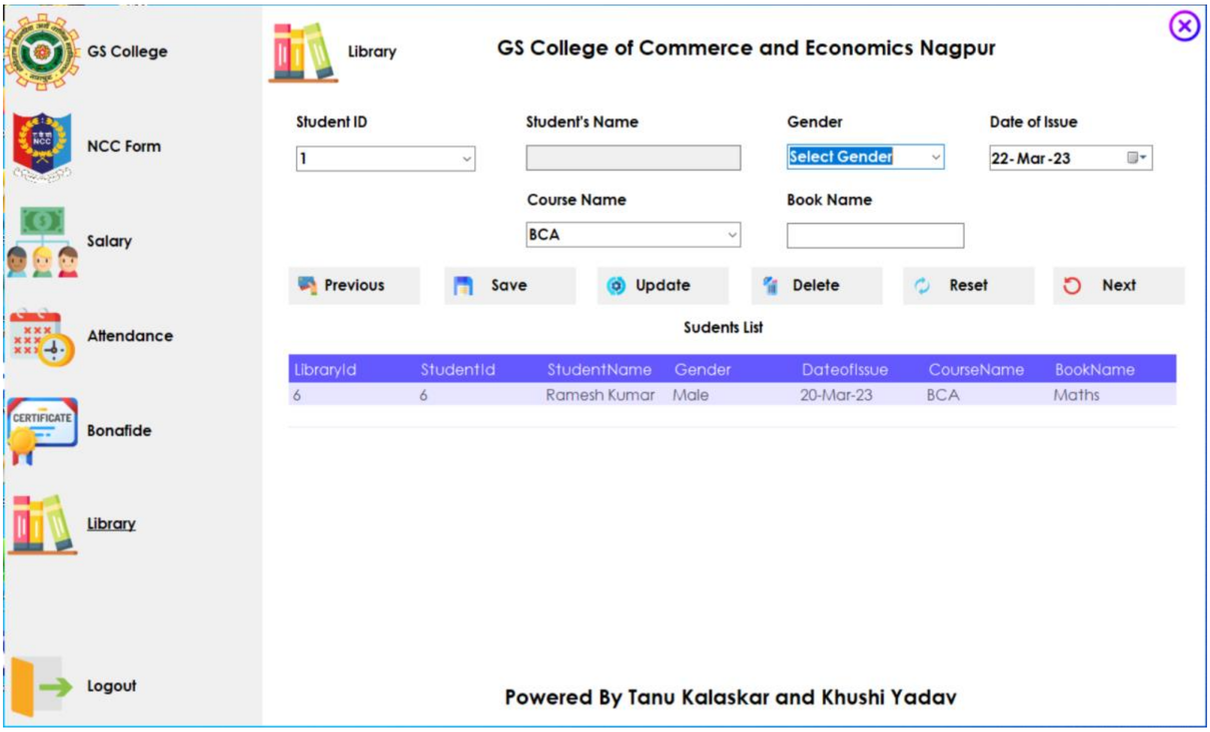
Library

Password

Powered By Tanu Kalaskar and Khushi Yadav

Loading Application ...Please Wait !

Library



GS College of Commerce and Economics Nagpur

Student ID: 1

Student's Name:

Gender: Select Gender

Date of Issue: 22-Mar-23

Course Name: BCA

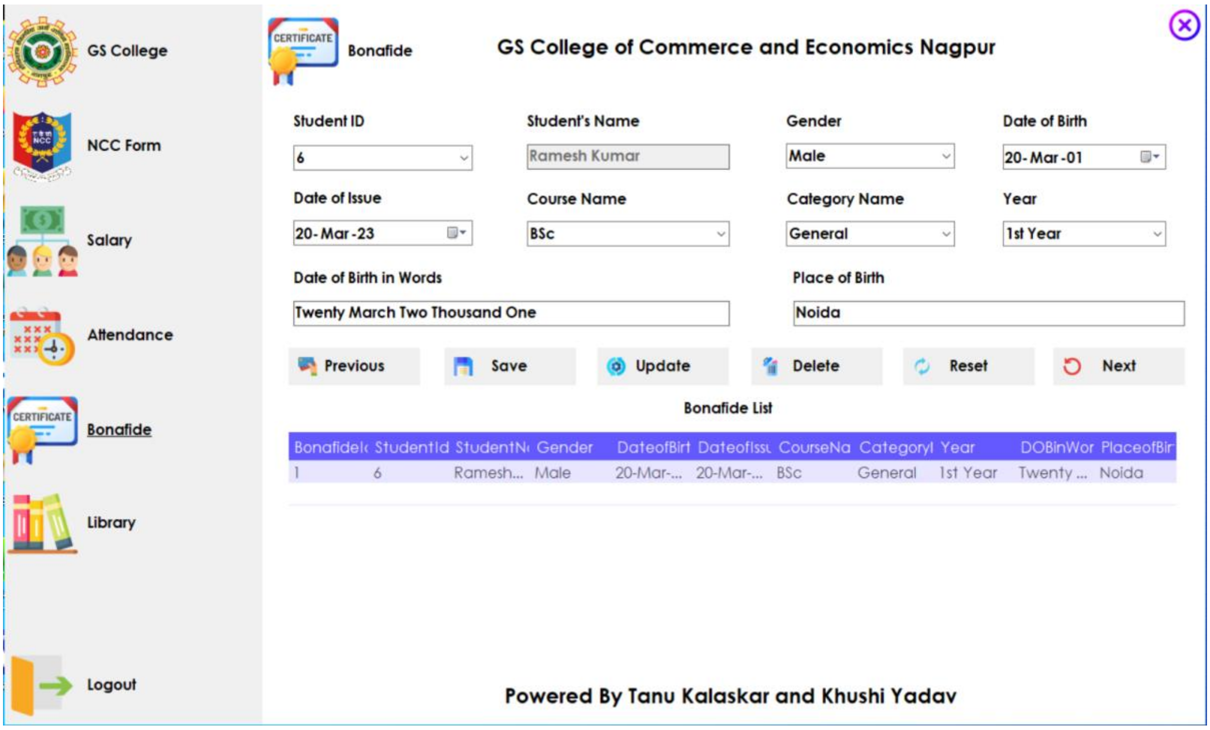
Book Name:

Sudents List

LibraryId	StudentId	StudentName	Gender	DateofIssue	CourseName	BookName
6	6	Ramesh Kumar	Male	20-Mar-23	BCA	Maths

Powered By Tanu Kalaskar and Khushi Yadav

Bonafide



GS College of Commerce and Economics Nagpur

Student ID	Student's Name	Gender	Date of Birth
6	Ramesh Kumar	Male	20-Mar-01
Date of Issue	Course Name	Category Name	Year
20-Mar-23	BSc	General	1st Year

Date of Birth in Words: Twenty March Two Thousand One

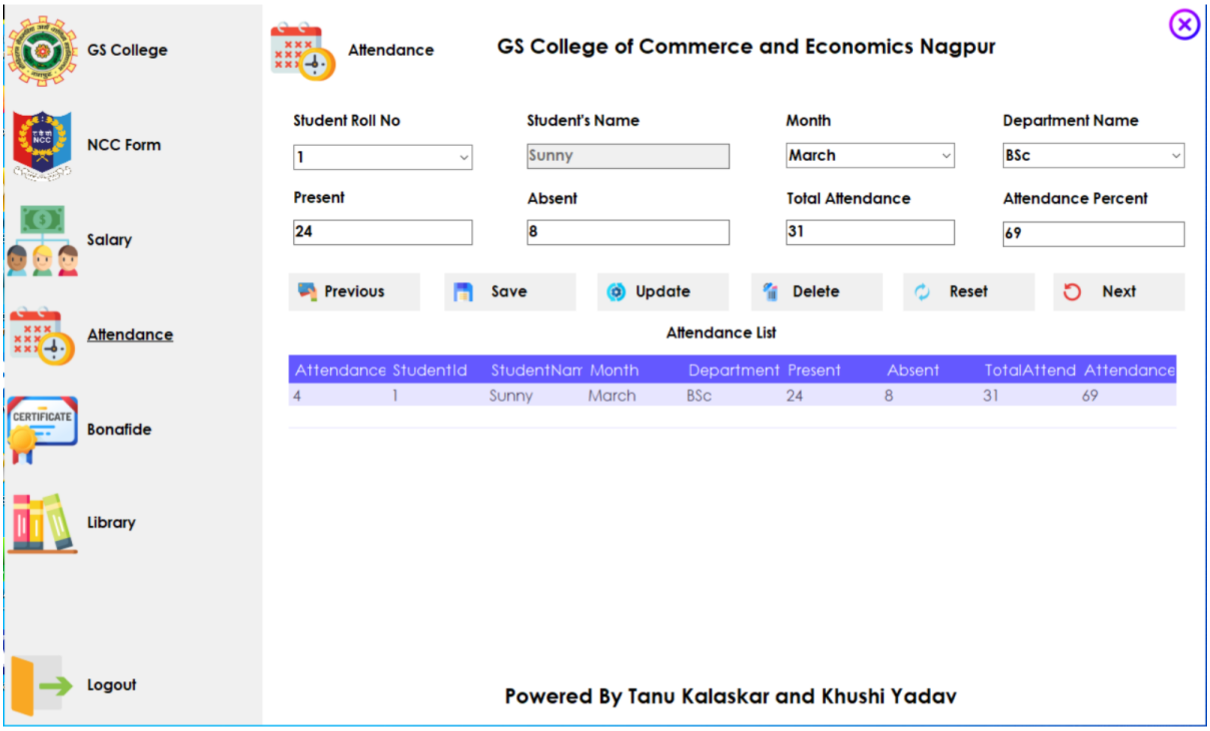
Place of Birth: Noida

Bonafide List

BonafideI	StudentId	StudentN	Gender	DateofBirt	DateofIssu	CourseNa	CategoryI	Year	DOBinWor	PlaceofBir
1	6	Ramesh...	Male	20-Mar-...	20-Mar-...	BSc	General	1st Year	Twenty ...	Noida

Powered By Tanu Kalaskar and Khushi Yadav

Attendance



GS College of Commerce and Economics Nagpur

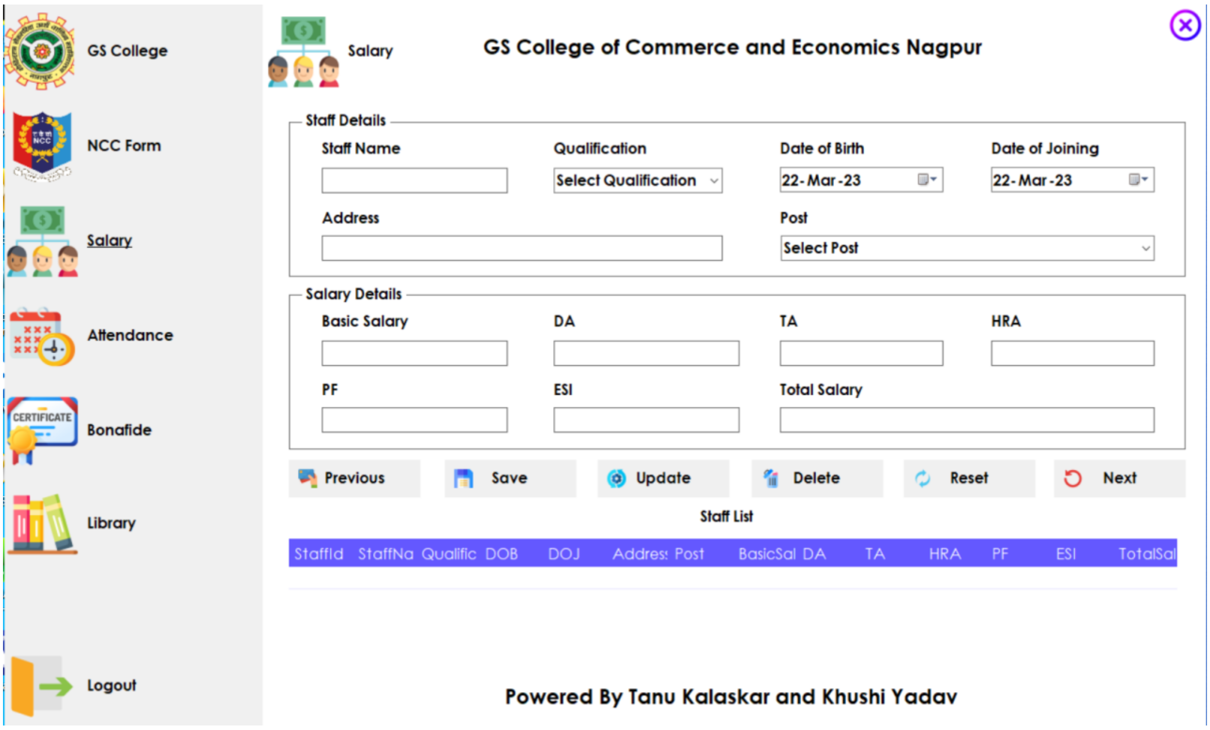
Student Roll No	Student's Name	Month	Department Name
1	Sunny	March	BSc
Present	Absent	Total Attendance	Attendance Percent
24	8	31	69

Attendance List

Attendance	StudentId	StudentNam	Month	Department	Present	Absent	TotalAttend	Attendance
4	1	Sunny	March	BSc	24	8	31	69

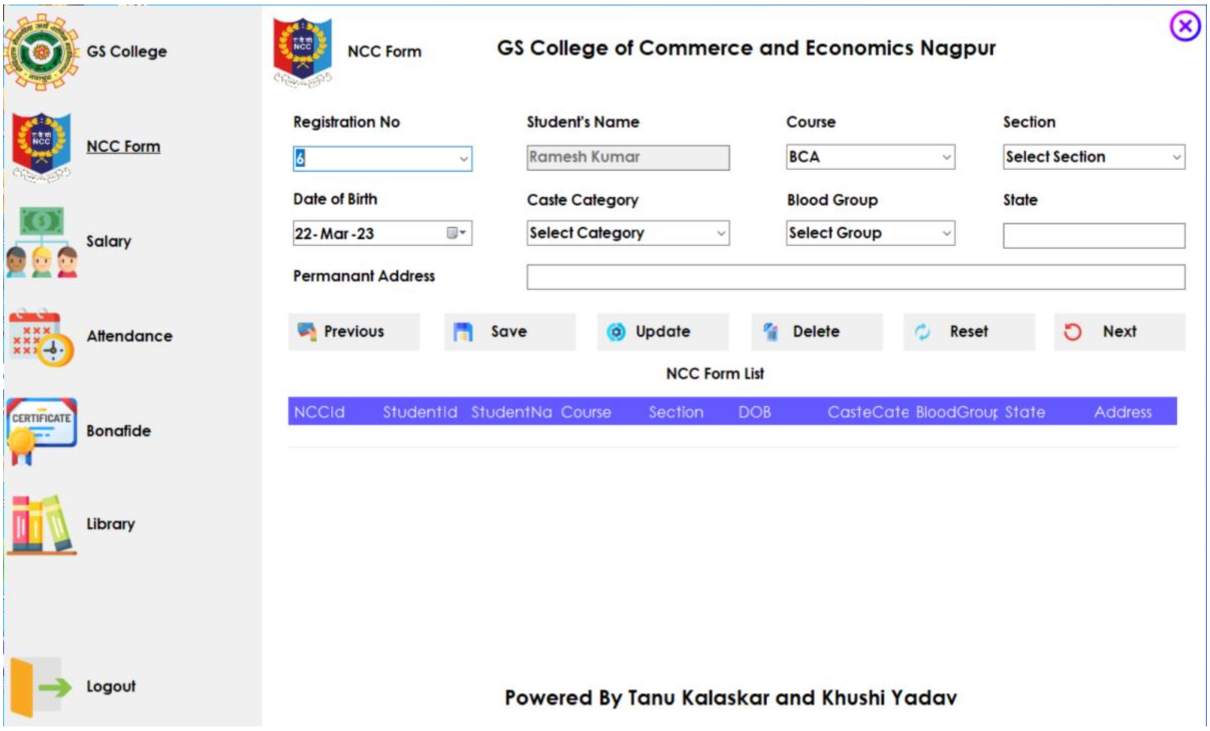
Powered By Tanu Kalaskar and Khushi Yadav

Salary


GS College
NCC Form
Salary
Attendance
Bonafide
Library
Logout
Salary
GS College of Commerce and Economics Nagpur
Staff Details
Salary Details
Previous
Save
Update
Delete
Reset
Next
Staff List
StaffId StaffNa Qualific DOB DOJ Addres: Post BasicSal DA TA HRA PF ESI TotalSal
Powered By Tanu Kalaskar and Khushi Yadav

Staff Name:
Qualification: Select Qualification
Date of Birth: 22-Mar-23
Date of Joining: 22-Mar-23
Address:
Post: Select Post
Basic Salary:
DA:
TA:
HRA:
PF:
ESI:
Total Salary:

NCC Form


GS College
NCC Form
Salary
Attendance
Bonafide
Library
Logout
NCC Form
GS College of Commerce and Economics Nagpur
Previous
Save
Update
Delete
Reset
Next
NCC Form List
NCCId StudentId StudentNa Course Section DOB CasteCate BloodGrouţ State Address
Powered By Tanu Kalaskar and Khushi Yadav

Registration No: 6
Student's Name: Ramesh Kumar
Course: BCA
Section: Select Section
Date of Birth: 22-Mar-23
Caste Category: Select Category
Blood Group: Select Group
State:
Permanant Address:

TESTING AND VALIDATION CHECK

TESTING AND VALIDATION CHECK

TESTING

Software testing is the critical element of software quality assurance and represent the ultimate review of the application, design and coding. The aim of testing process is to identify all defect existing in the software products. Testing provides a practical way of reducing defects in a system and increasing the user’s confidence in a developed system.

TESTING OBJECTIVE

- ✓ Testing is the process of executing the program with a program with the intent of finding out the error.
- ✓ A good test case is one of that has a high probability of finding as a yet undiscovered error.
- ✓ A successful test is one that uncovers as a yet undiscovered error.

VALIDATION CHECKS

Validations are nothing but the security measures taken at the time of execution of any program. It is necessary for the analyst to take validation in his project as it provides more accuracy and systematic flow to the project. Validation not only stops input of false data but also provides information in the form of message to the user to correct his mistake. Hence it plays a role of a guide during input of data.

Validation put its control over the data in both characters as well as integer data type. Whenever wrong data or invalid data is stored by the user it fires the message immediately and warns the user to input correct data type. Validations are classified into two types

Validating input transaction:

Validating input data is largely done through software which is the programmer's responsibility but it is important that system analyst must know what common problem might invalidate a transaction. Business committed to quality will include validation checks as a part of their routine software

Three main problems occur with input transaction.

- Submitting the wrong data to system,
- Submission the data by an unauthorized person or
- Asking the system to perform an unacceptable function.

Validating input data:

It is essential that the input data themselves along with the transactions requested are valid. Several texts can be incorporated into software to ensure this validity. We consider many possible ways to validate input and they are as follows:

- Test for missing data
- Test for correct field length
- Test for class or composition
- Test for range or reasonableness
- Test for comparison with stored data
- Test for invalid values

SYSTEM SECURITIES MEASURES

SYSTEM SECURITIES MEASURES

Security is an important term of any project (system) development. Security provides a protection to the computer, database, and project from unauthorized user. Security of computer facilities stored data and the information generated is a part of successful conversion. Recognition of the need for security is a natural outgrowth of the belief that information is a key organizational resource. It is useful to think of security of the project data and information on an imaginary continuum from totally secure to totally open. Security is the responsibility of all those who come into context with the system. Security has three interrelated aspects i.e. physical, logical and behavioural.

When you click on the "splash screen" very first form of the project, you will get a password frame consisting a text box for password. As soon as frame open and ask for password from the user, user enters the correct password and will move on the next form otherwise move will not be allowed until and unless user types the correct password or until three attempts whichever less. If you enter wrong password, a message box will appear having a message, "Invalid Password"

IMPLEMENTATION

EVALUATION

MAINTENANCE

IMPLEMENTATION

In this last phase of system development, we have implemented the information system. This phase involves training users to handle the system some training is done by users but oversight of training is the responsibility of the system-analyst. In addition, the analyst needs to plan for a smooth conversion from the old system to the new one. This process includes converting files from old formats to zero ones or building a database, installing equipment and bringing the new system into production. It is primary way user training site preparation file conversion when system is link to terminal or remote sides the telecommunication network is also include after implementation. After installation phase is complied and user staff is adjusted and created by system evaluation and maintenance.

EVALUATION

Evaluation is shown as part of this final phase of the System Development Life Cycle (SDLC), mostly for the sake of discussion. Actually, evaluation takes place during every phase. A key criterion that must be satisfied is whether the intended user are indeed using a system throughout at the time of development

We have been evaluating the evolving information system and networks to give feedback for our essential improvement. In recognition the evaluation of information system and network is important. We have taken many evaluation techniques devised. These techniques include cost-benefit analysis models that attempt to estimate the values of a decision based on the effects of revised information using information theory, simulation, user evaluation that emphasize implementation problem and user evaluation involvement and information system utility approaches that examine the properties of information.

Each type of evaluation serves a different purpose and has invented drawbacks. The revised decision evaluation approach present difficulty because all variables involved within the design, development and information of the project cannot be calculated or qualified. The user involvement approach to evaluation can be more comprehensive than others if it is expanded and systematically applied.

MAINTENANCE

After the system is installed it must be maintained. Means the computer programs must be modified and kept up-to-date. Estimates of the time spent by departments on maintenance have ranges from 40-60% of time spent developing projects. Very little time remains for new system development

Maintenance is performed for two reasons. The first of these is to correct software errors. No matters how thoroughly the system is tested, bugs or errors deep into the computer programs. The other reason for performing system maintenance is to enhance the software's capabilities in response to changing organizational needs. Generally involving one of the following two situation

1. Users often request additional features after they become familiar with the computer system and its capabilities

2. Hardware and software are changing at an accelerated phase.

The total cost of maintenance is likely to exceed that of system development. At certain point it becomes more feasible to perform a new information system. Maintenance is an outgoing process over the life cycle of an information system. After the system is installed, maintenance usually takes the form of connecting previously undetected program errors. Once these are connected the system approaches a steady state, providing dependable service to its users.

FUTURE SCOPE OF THE PROJECT

FUTURE SCOPE OF THE PROJECT

Though the project is made by considering the users name there are some limitations of the project, which can be modified in the future. Some of the future modifications and the scope of the project in the future are given as under:

1. In future this project can be modified to incorporate the Internet Feature in it. By doing this we can access student's details, their Records and many other Details at any place anywhere in the world.

2. With other network-based software this software will gives nice result.

3. As it deals with College Management, in future the project can be modified to have connection with any accounting software

4. In future project can be made to support all databases

5. Report Relating to Staff's Tax assessment is to be generated

Through this software, Management of College will be able to have the easy retrieval of any information regarding teacher's personal information, their records, their salary details, etc. This system will definitely reduce the workload of the user and will be helpful in the accurate data maintenance.

CONCLUSION

CONCLUSION

Advantages:

The system has the following advantages:-

- The data can be maintained and updated easily with minimum efforts.
- The complex reports were generated easily and minmum efforts.

Limitations:-

The Developed application satisfies the user's expectation but it has still few limitations, which we cannot ignore. These limitations could have been eliminated, but due to time constraint and the restricted scope of this project still persist. The limitations of this application are discussed as Under:

- ✓ . We have to remember Admission No while referring Student
- ✓ information. No system entry has been provided to keep track of the damaged and
- ✓ lost salary slip from the staff

BIBLIOGRAPHY & REFERENCE

While developing this project, internet was the enternal support.

Following are the website referred by us in developing our project :

- Google
- You Tube
- https://youtu.be/jhFGV_MFsa4