COLLEGE ADMINISTARTION SYSTEM

A project report submitted to the Bharathiar University in the partial fulfillment

of the requirements for the award of the degree of

BACHELOR OF COMPUTER APPLICATIONS

Submitted by

ALEN P SHYJU

(REG.NO: 20UGCA006)

Under the Guidance of

Ms. RAYNUKAAZHAKARSAMY

MCA, M. Phil.(CS), M. Tech.(CSE), M. Sc.(Maths), Ph.D.(CS)

NEHRU ARTS AND SCIENCE COLLEGE

(Autonomous)

(Reaccredited by NAAC with “A” Grade, ISO 9001-2015 & ISO 14001: 2004 Certified)

Recognized by UGC & Affiliated to Bharathiar University

“NEHRU GARDENS”, T.M. Palayam , Coimbatore - 641 105.

March – 2023

DECLARATION

I, **ALEN P SHYJU** (**20UGCA006**) hereby declare that the project entitled **“COLLEGE ADMINISTRATION SYSTEM”** submitted to Bharathiar University in partial fulfillment for the award of the Bachelor Degree of Computer Applications is an independent project report done by me during the project duration of the period of study in Nehru Arts and Science College, Coimbatore (Recognized by UGC &Affiliated to Bharathiar University) under the guidance of **Dr. RAYNUKAAZHAKARSAMY** during the academic year 2020-23.

PLACE :

DATE : Signature of the Student

ACKNOWLEDGEMENT

I solemnly take this opportunity to all the helping hands that made me accomplish this project. First and foremost, I thank the Almighty who is the source of knowledge and one who guided me for completing the project successfully.

I sincerely thank our respected Principal **Dr. B. ANIRUDHAN M.A., B.Ed., M.Phil., Ph.D.,** Nehru Arts and Science College for permitting me to undertake this project as a part of curriculum and for giving me the best facilities and infrastructure for the completion of the courseand project.

I solemnly thank our Dean, **Dr. Saraswathi. S, MCA., M.Phil., Ph.D**. Dean of Academics for her support.

My immense gratitude for my Head of the Department **Mr. P. BOOPATHI MCA M.Phil.,** for his consultant help and creative ideas over the period of the project work.

I owe a great deal of gratitude to my faculty guide **Dr. RAYNUKAAZHAKARSAMY MCA., M. Phil (CS)., M. Tech.(CSE), M. Sc.(Maths), Ph.D.(CS)** Assistant Professor, Department of Computer Applications, for his consultant help and creative ideas over the period of project work. I express my sincere words gratitude to other faculty members and staff members of the Department of Computer Application.

I extend my sincere thanks to my parents and I would like to thank all my friends for theirmoral support rendered to me to complete this project in grand success.

NEHRU ARTS AND SCIENCE COLLEGE

(Reaccredited by NAAC with “A” Grade, ISO 9001-2015 & ISO 14001: 2004 Certified)
Recognized by UGC & Affiliated to Bharathiar University
“NEHRU GARDENS”, T.M. Palayam, Coimbatore - 641 105.

DEPARTMENT OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project report entitled **“COLLEGE ADMINISTRATION SYSTEM”**, is a bonafied work done by **ALEN P SHYJU (20UGCA006)** in partial fulfillment of the requirement of the award of the degree of Bachelor of Computer Applications, Bharathiar University, Coimbatore during the academic year 2023.

Internal Guide   Head of the Department

Certify that we examined the Candidate in the Project Work / Viva-Voce Examination held at Nehru arts and science college on ________________

Internal Examiner   External Examiner

ceritificate

SYNOPSIS

This project entitled “College Administration System” is a software application designed to manage various administrative tasks within a college or university. It serves as a centralized platform for students, faculty, and staff to access and manage academic and administrative information.

The system typically includes modules for managing student admissions, registration, as well as tracking academic progress and managing course schedules. It may also include modules for managing faculty information. Other key features of a college administration system may include financial management, such as managing tuition payments and financial aid, as well as managing campus facilities and resources such as library, labs, and equipment.

Overall, a college administration system is designed to streamline administrative activities, which improve communication and collaboration across departments and stakeholders, and provide a more efficient and effective system for managing the many aspects of running a college or university.

S. NO	CONTENT	PAGE NO
1	**INTRODUCTION** 1.1 About the project 1.2 Organization Profile	1 1 2
2	**SYSTEM ANALYSIS** 2.1 Existing System 2.1.1 Disadvantages of existing system 2.2 Proposed System 2.2.1Advantages of proposed system 2.2.2 System Study 2.3 System Specification 2.3.1 Hardware Specification 2.3.2 Software Specification 2.3.3 About the Software 2.3.4 Module description	3 3 3 4 4 4 6 6 7 10
3	**SYSTEM DESIGN** 3.1 Design Notations 3.1.1 Data Flow Diagram 3.1.2 System Flow Diagram 3.1.3 ER Diagram 3.2 Design Process 3.2.1 Input Design 3.2.2 Database Design 3.2.3 Table Design 3.2.4 Output Design 3.2.5 System Development	12 12 17 18 19 19 19 20 22 23
4	**SYSTEM TESTING AND IMPLEMENTATION**	24

	4.1 Testing Methodologies 4.1.1 Unit testing 4.1.2 Integration testing 4.1.3 Validation testing 4.1.4 Output testing 4.2 System Implementation 4.3 System Maintenance	24 24 25 25 26 26
5	**CONCLUSION**	27
6	**FUTURE ENHANCEMENTS**	28
7	**BIBLIOGRAPHY**	29
8	**APPENDIX** A. Sample Screenshot B. Sample Coding	30 51

1. INTRODUCTION

1.1 ABOUT THE PROJECT

COLLEGE ADMINISTRATION SYSTEM

This system is used to maintain students, staffs, subject and fees details. In this system includes paper works made by human so the times taken to maintain all these details are very long and also tedious to implement. which is implemented in automated manner. It has four modules, which are Student, Staffs, and Subjects, section Details. In student section includes student's Registration, marks, and attendance details. In staffs section includes registration and details of the staffs. In an office section includes student's fees details and staffs salary details.

At first the new student names are registered in the register form of the students. Once you register in this form, which automatically included in mark list form and attendance form. After you registration you can maintain the marks and the attendance details through their corresponding form. Similarly you can also register for new staffs and you can maintain other all relevant details.

This puts the user firmly in control of how the program flows and therefore increases user satisfaction with the program. The key portions of a visual basic project are the forms, controls, and code modules. A form or control has properties that you can alter. The program's code is contained primarily in events mentioned below in various forms in order to keep track of the system.

1.2 ORGANIZATION PROFILE

The **Black code technologies** are one of the best software development companies in Coimbatore. We offer software development services like customized ERP development, CRM development and HRM development. We are also one of the top web development companies in Coimbatore. Our web development services include Word Press development, Magneto development, E-Commerce development and Static Web development.

Delivering Quality services at an affordable cost has been the motto of The Bigcode technologies ever since its inception. We are experts in analyzing your requirements and presenting the software solutions you need. Our company has so far catered software services to over 1000 Clients in the last 15 years. Also, we are one of the very few software companies in Coimbatore to provide round the clock support to our clients.

Our Mission

Our mission is to build long term relationship with our clients through a combination of innovative, high quality, cost effective solutions and services.

We ensure client's satisfaction as our top most priority by consistently meeting and exceeding customer requirements. Black code Technologies will continue to keep ourselves abreast of cutting edge technologies to support clients grow their business and to constantly outperform our peers.

Our Vision

We passionately dedicate ourselves to the cause of creating high end, innovative, quality services and solutions to clients

We want to reach out worldwide organizations with preeminent services and leading edge technologies to provide excellent offshore development services. We strive to be acknowledged as a leader and preferred partner in offshore development

2. SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

In the system is maintained in manually. The fees details, the enrolling details and the students, staffs details are maintained manually. The study of the system revealed that the system has several drawbacks. Student needs to take long time to learn and explore. Besides that, home pages of the system also not user friendly. The existing manual system for the college information system is done by manual application, which is very difficult to handle. This is very typical and time consuming job to maintain manually all these tasks required by the organization. So it is not a reliable and efficient approach to do work manually.

2.1.1 DISADVANTAGES OF EXISTING SYSTEM

- The existing system has no security measure against logging in and no checks are made for authorized users.
- The end user has to remember a lot of command to make efficient use of the system.
- The system does not have any descriptive reports and thus did not help administration in decision-making.
- The registration information per day is unable to find.
- Enormous amount of time is consumed.

2.2 PROPOSED SYSTEM

The proposed system for university management system supports very efficiently and able to solve the problems occurred in the existing system. Easy to generate the details of each and every student. Stored all data can be for a longer period. At first the new student names are registered in the register form of the Students. Once you register in this form, which automatically included in mark list form and attendance form. After you registration you can maintain the marks and the attendance details through their corresponding form.

2.2.1 ADVANTAGES OF PROPOSED SYSTEM

- The user can enter only if the username and the password are correct.
- The process of planning will be easy since every process is computerized.
- Time Saving.
- The Fees information per day and per month can be known.
- The details of the all saved information can be viewed.
- The data can be accessed easily whenever needed and so the manual work can be reduced.

2.2.2 SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- OPERATIONAL FEASIBILITY
- SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

OPERATIONAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

2.3 SYSTEM SPECIFICATION

2.3.1 HARDWARE SPECIFICATION

System	:	HCL/HP Any
Processor	:	Intel core2 dual
Hard disk capacity	:	500 GB
Internal memory capacity	:	4 GB

2.3.2 SOFTWARE SPECIFICATION

Operating system	:	Windows 10
Front end	:	VB.NET
Back end	:	SQL SERVER

2.3.3 ABOUT THE SOFTWARE

VISUAL BASIC.NET

Visual Baic.Net has revolutionized windows programming windows programming and with an object based, event driven approach to software designs. Visual basic.Net applications act as a front end to the database. Visual basic.Net application provides the interface between the user and the database. Sophisticated features that make the language truly object oriented and interfaces it with the latest in the database technology.

.NET provides a new, object-oriented API as a set of classes that will be accessible from any programming language. This book describes this framework of classes and provides a reference to what is available and how you can use this framework to write Windows applications in the brave new world of .NET.

Microsoft .NET Framework is a computing platform for developing distributed applications for the Internet. Following are the design goals of Microsoft .NET Framework:

- To provide a very high degree of language interoperability
- To provide a runtime environment that completely manages code execution
- To provide a very simple software deployment and versioning model
- To provide high-level code security through code access security and strong type checking
- To provide a consistent object-oriented programming model
- To facilitate application communication by using industry standards such as SOAP and XML.
- simplify Web application development

Visual basic.net lets the user to add menus, text boxes, command buttons, option buttons, check boxes, list boxes, scroll bars, and file directory boxes to blank windows. Visual basic.net has many different tools.

The Common Language Runtime

The CLR is the mechanism through which .NET code is executed. It is built upon a single, common language—IL—into which source languages are compiled and includes mechanisms for executing the compiled code. This includes code verification and just-in-time

(JIT) compilation, garbage collection and enforcement of security policies, and the provision of profiling and debugging services.

The CLR provides a lot of added value to the programs it supports. Because it controls how a .NET program executes and sits between the program and the operating system, it can implement security, versioning support, automatic memory management through garbage collection, and provide transparent access to system services.

The user interface is the part of the program that responds to the key press and mouse clicks. The action is referred to as events of the form and controls in the form. The form is a visual object where a set of object controls is placed to communicate with backend databases and validation checks. VB.NET provides a vast properties and methods for each controls, which helps to utilize all them, functions for record manipulations.

IMPORTANT FEATURES

- The application is a graphical user interface.
- Client-Server architecture benefits picture and image box can be easily handled using bit mapped files and icons.
- Bit mapped files and icons are used as simple debugging tools.
- With the advent of .NET, Microsoft has introduced many new technologies that make writing component-based distributed systems easier, more flexible, and more powerful than ever before.
- It is now easier than it has ever been to write components in any programming language that can interoperate with components on other machines, which may not be Windows-based at all.

SQL SERVER 2005

The purpose of this document is to help you migrate your applications when you are migrating the underlying database from Microsoft SQL Server to Some other Database. Most of the issues encountered when migrating applications to use an SQL Server Database are related to database incompatibility. This paper presents these incompatibilities and provides solutions for many issues. It familiarizes yourself with this document prior to migrating your databases. The choices made about how to migrate your application affect how you migrate the underlying database from Microsoft SQL Server to some other Database.

If the application uses the Microsoft OLE DB Provider for SQL Server or a third-party OLE DB Provider to connect the Microsoft SQL Server database, use the SQL Server Provider for OLE DB (SQLOLEDB) to connect to the migrated Database.

There are quite a few reasons, the first being that SQL is a feature rich program that can handle any database related task you have. You can create places to store your data build tools that make it easy to read and modify your database contents, and ask questions of your data. SQL is a relational database, a database that stores information about related objects. In MS SQL that database means a collection of tables that hold data. It collectively stores all the other related objects such as queries, forms and reports that are used to implement function effectively.

SQL Server Security

- Login Authentication.
- Windows NT Authentication
- SQL Server Authentication
- Permissions validation on user database.
- T-SQL statements sent to SQL Server.
- SQL server checks user permissions on receipt of T-SQL statements

FEATURES

- Created by Microsoft and Sybase in the 80s.
- Is SQL Compliant - Uses ANSI SQL
- Supports SQL – 92 standards - Uses T-SQL
- Stores data in a central location and delivers it to clients on request
- New Server Architecture
- Graphic Administration Tools
- Maintains ANSI standards and 6.x Compatibility
- Data integrity means reliability and accuracy of data.
- Integrity rules keep data consistent.
- Supports Client/Server model.
- Request response dialog.

- Workload is split between the client and the server.
- Operating System compatibility.
- Runs on Win 95/98 NT, Netware, UNIX, OS/2, AppleTalk, Banyan VINES.
- SQL Server must have Service Pack 4 (SP4) to run on Windows NT 4.0.
- Multiple protocol compatibility.

2.3.4 MODULE DESCRIPTION

Maintenance Module

- Registration Details
- Staffs Details
- Library Books Details
- Course Details

Students Module

- Students Details
- Attendance Details
- Marks Details
- Library Entry
- Lab Entry

Office Module

- Students Fees Details

Reports module

- Registration
- Attendance report
- Course report

- Fee
- Library
- Marks report

Students Module

It is used to Enter Student Details who comes to join in our college. It contains a Student Register Number, Name, Address, Qualification, Department of Study and other necessary details. Add, modify, delete and view the records from the database as possible. The user login section only views the student's details.

Staffs Module

It is used to Enter Staff Details who comes to join in our concern. It contains a Staff Code, Staff, Name, Address, Contact No, Designation and other necessary details. Add, modify, delete and view the records from the database as possible only by admin Section. The User only views the staff's details. In this module add the staff pay slip details.

Marks Details

In this module the student marks are stored in to the database. The marks are classified by semester wise, internal marks and class tests. Here the parent's wants mark means from the system we have to send them.

Attendance Details

In this module all the students' attendance details is maintained. Here the students attendance and staff attendance maintained in separate table.

Library Details

Library details are maintain in this module book id, name, number of books, author name and date etc.

Fees details

Fees details over all maintain the student semester fees details and hostel fees details etc. It contains student id, student name, fees per semester, hostel number and cost etc.

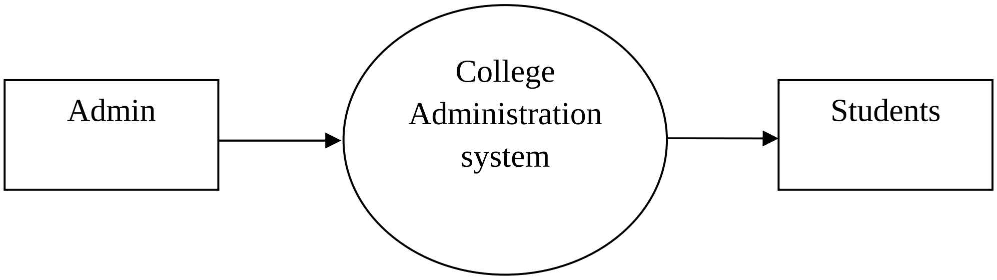
3. SYSTEM DESIGN

3.1 DESIGN NOTATION

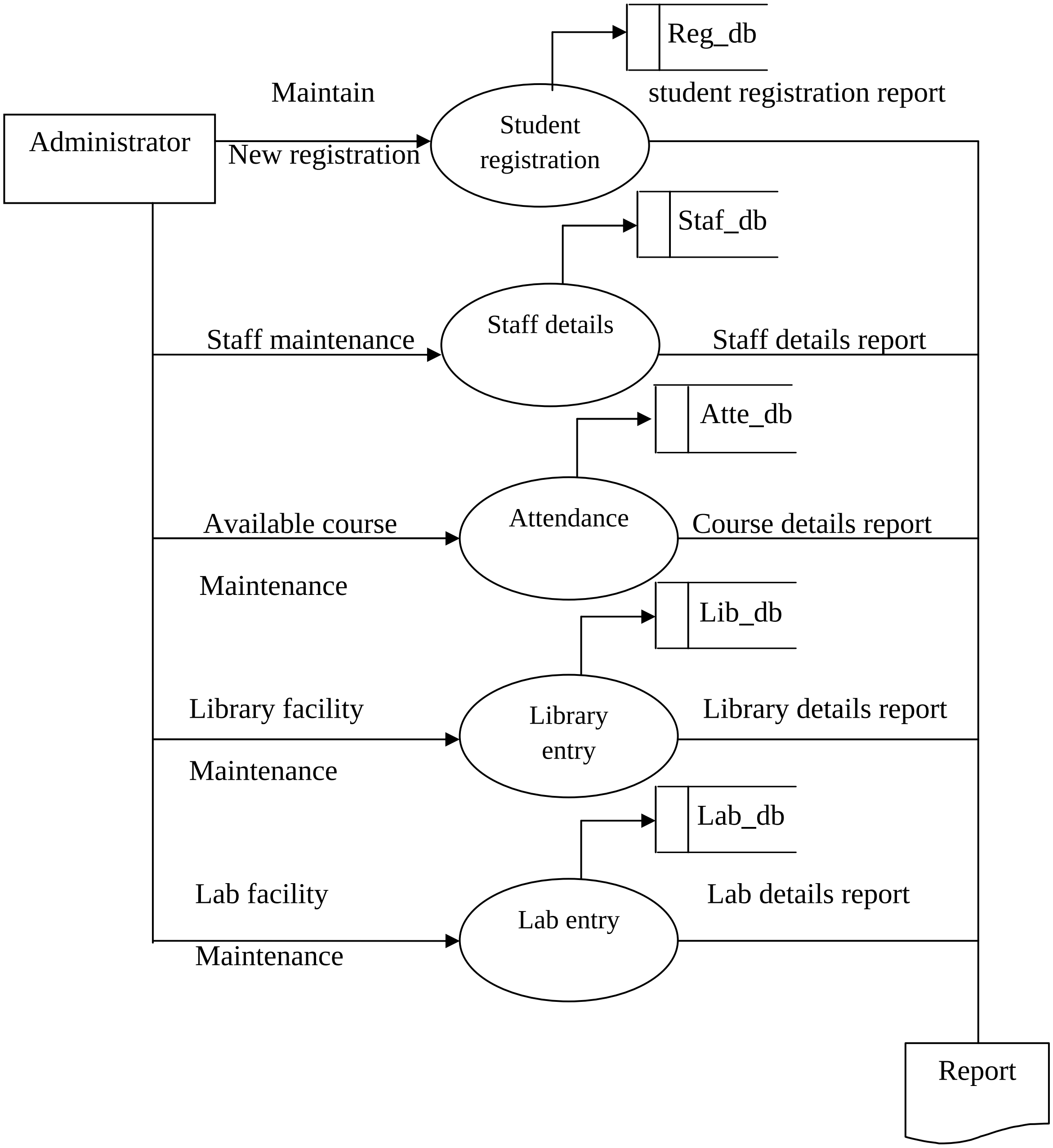
System design is the process of planning a new system to complement or altogether replace the old system. The purpose of the design phase is the first step in moving from the problem domain to the solution domain. The design of the system is the critical aspect that affects the quality of the software. System design is also called top-level design. The design phase translates the logical aspects of the system into physical aspects of the system.

3.1.1 DATA FLOW DIAGRAM

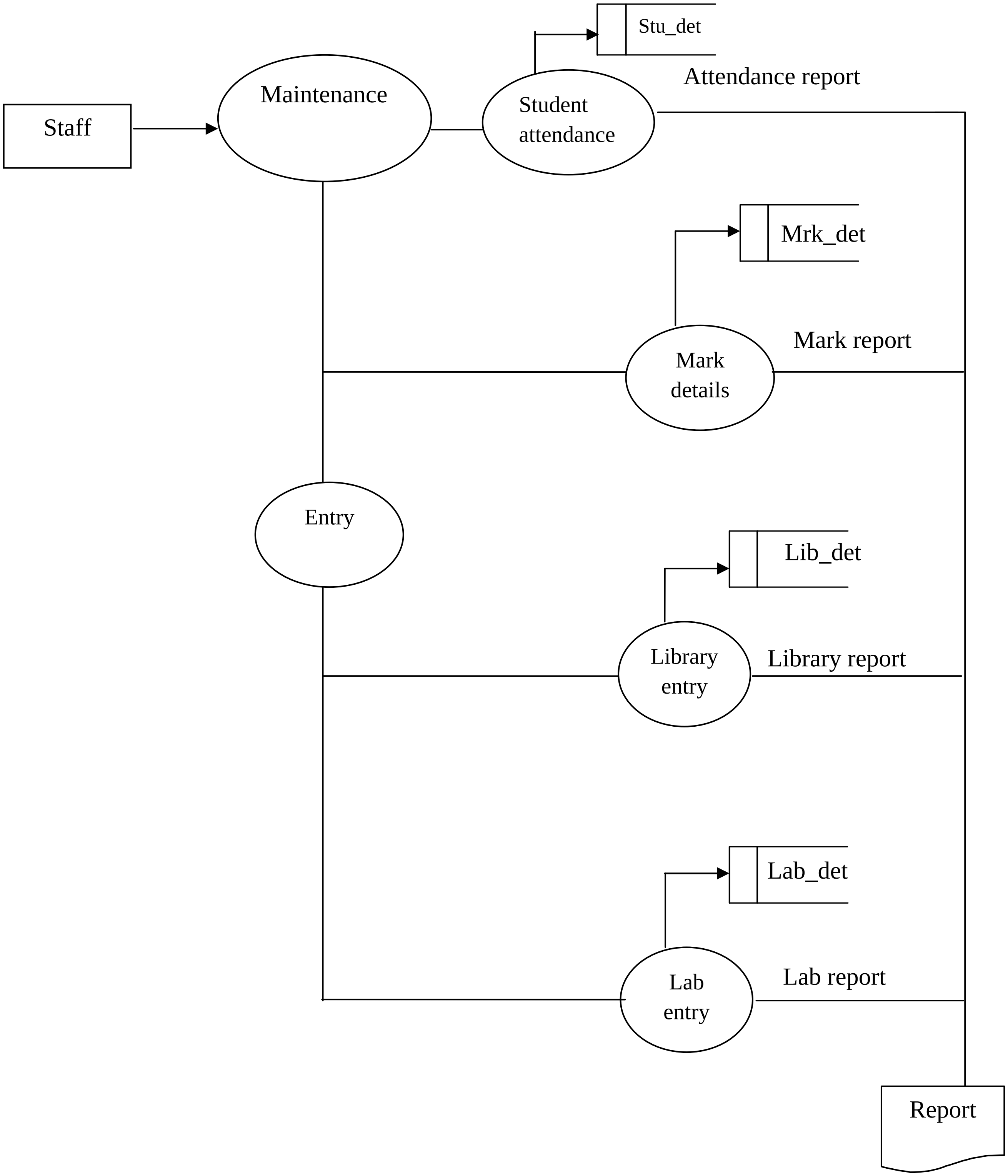
Level 0



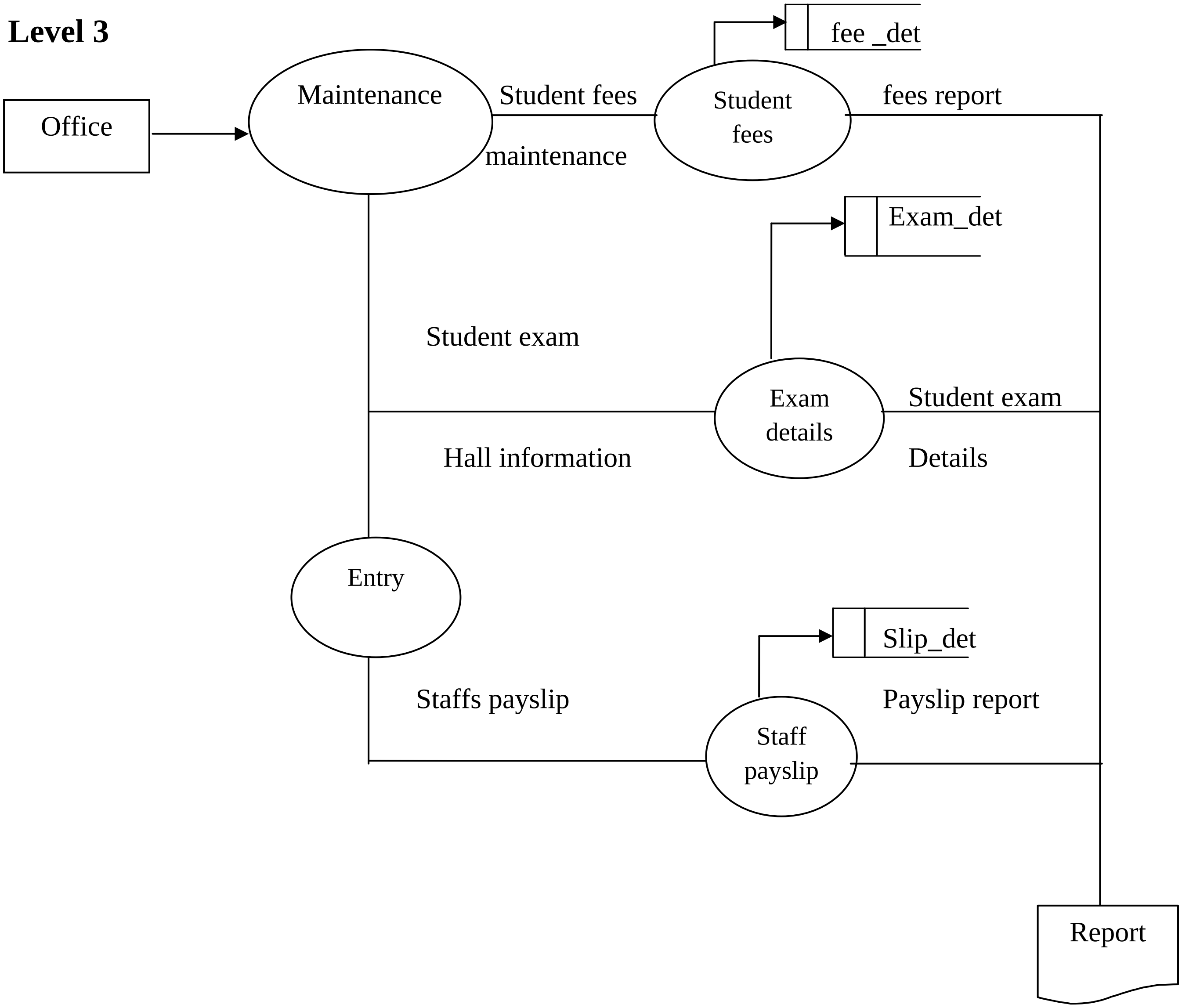
Level 1


Reg_db
Maintain
student registration report
Administrator
New registration
Student registration
Staf_db
Staff maintenance
Staff details
Staff details report
Atte_db
Available course
Maintenance
Attendance
Course details report
Lib_db
Library facility
Maintenance
Library entry
Library details report
Lab_db
Lab facility
Maintenance
Lab entry
Lab details report
Report

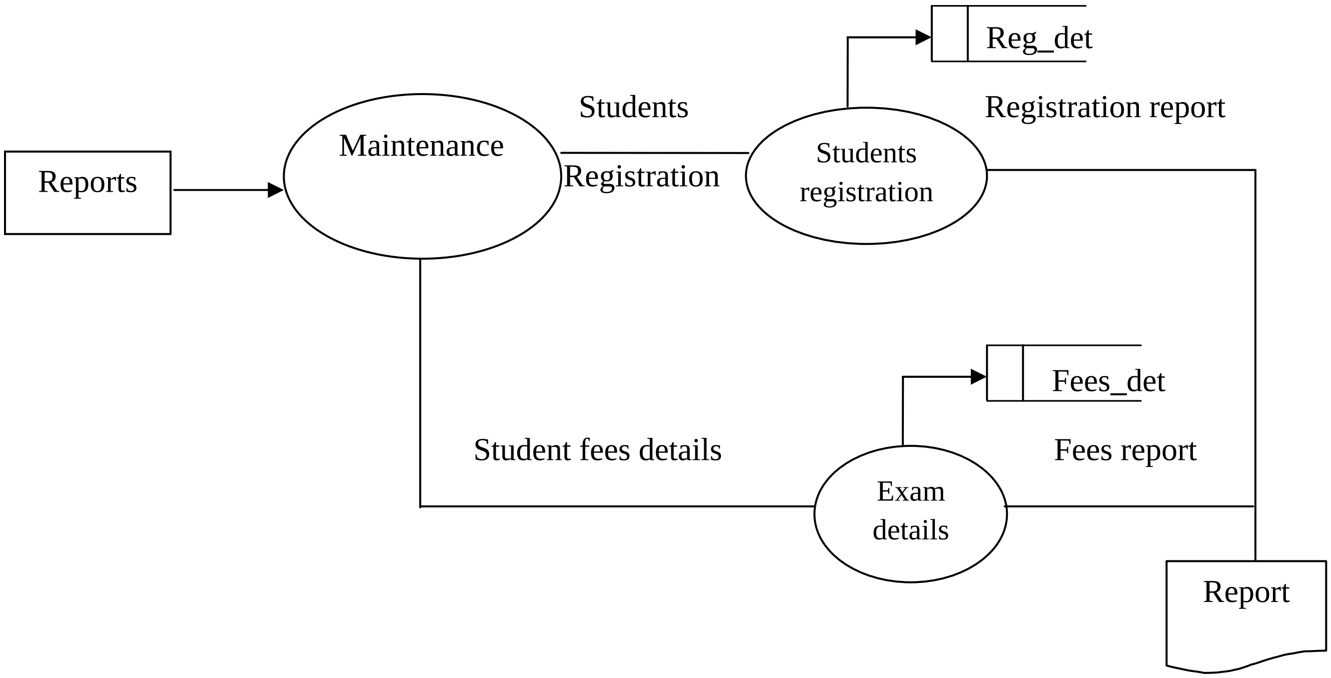
Level 2


Staff
Maintenance
Student attendance
Stu_det
Attendance report
Mrk_det
Mark details
Mark report
Entry
Lib_det
Library entry
Library report
Lab_det
Lab entry
Lab report
Report

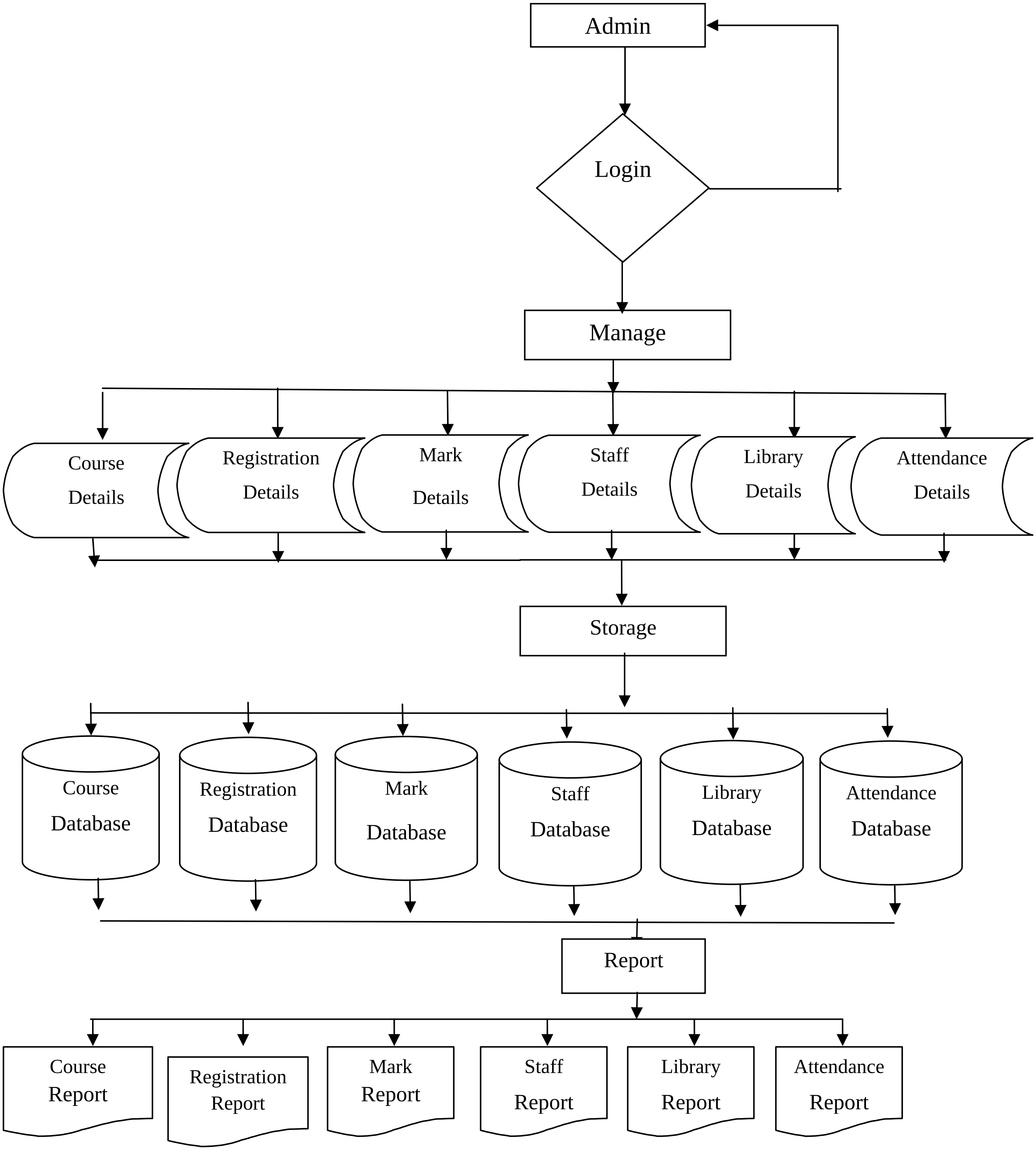
Level 3


fee _det
Office
Maintenance
Student fees
Student
fees
fees report
maintenance
Exam_det
Student exam
Exam
details
Student exam
Hall information
Details
Entry
Slip_det
Staffs payslip
Payslip report
Staff
payslip
Report

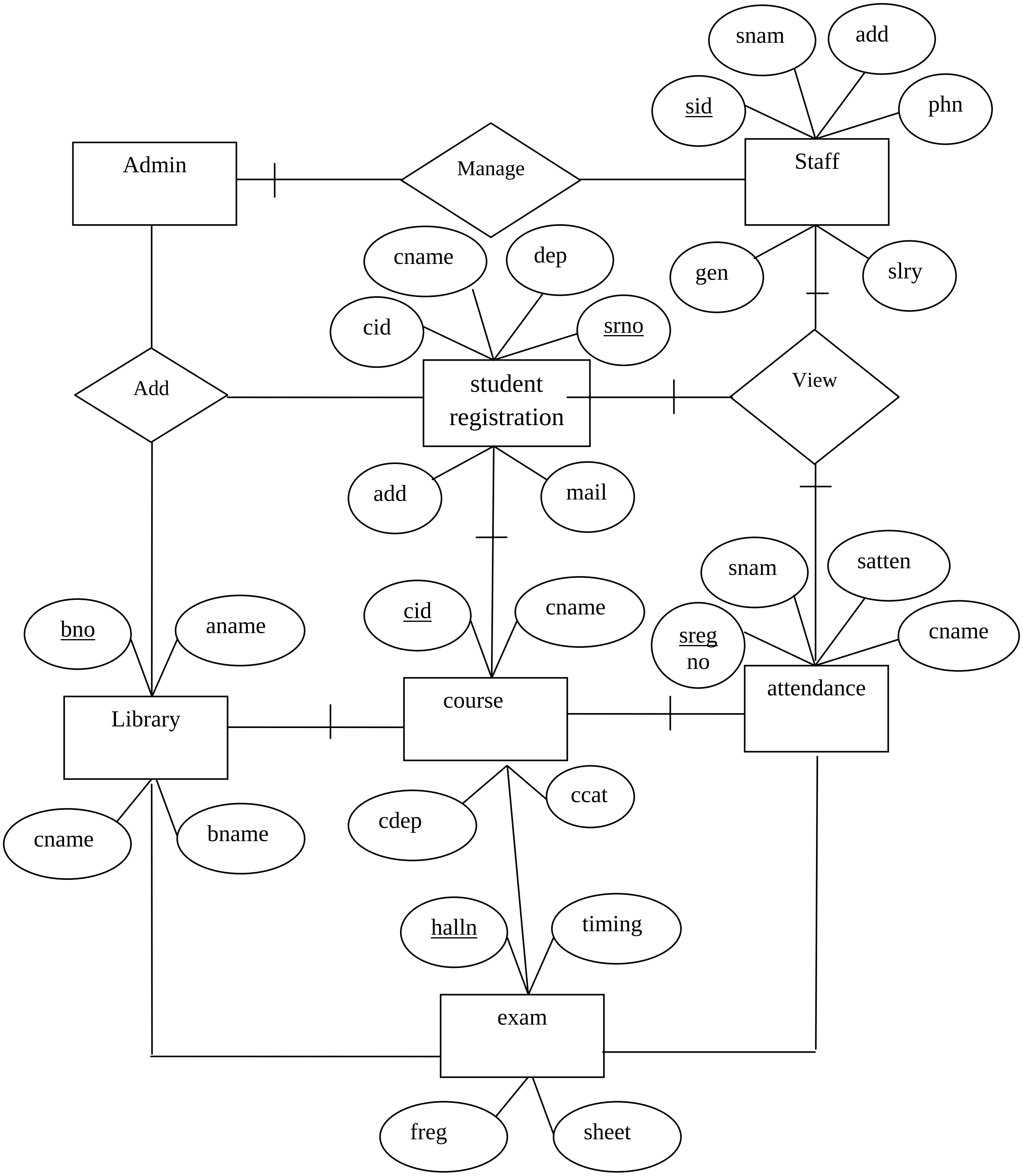
Level 4


Reg_det
Students
Registration report
Maintenance
Reports
Registration
Students registration
Fees_det
Student fees details
Fees report
Exam details
Report

3.1.2 SYSTEM FLOW DIAGRAM



3.1.3 ER DIAGRAM



3.2 DESIGN PROCESS

3.2.1 INPUT DESIGN

Input design is the process of converting the user-oriented. Input to a computer based format. The goal of the input design is to make the data entry easier, logical and free error. Errors in the input data are controlled by the input design. The quality of the input determines the quality of the system output.

All the data entry screen are interactive in nature, so that the user can directly enter into data according to the prompted messages. The user are also can directly enter into data according to the prompted messages. The users are also provided with option of selecting an appropriate input from a list of values. This will reduce the number of error, which are otherwise likely to arise if they were to be entered by the user itself.

3.2.2 DATABASE DESIGN

A database should provide integration, Integrity and a data independence table in a database contains information pertaining to a specific entity. To maintain the tables in an effective way, it should be normalized to ensure that the number of tables does no exceed the optimum level unless it is mandatory.

To prevent unauthorized access, security measures have been provided. This may prevent unauthorized persons using data that is private. The normalization techniques have been used to design the table such that the use of all the tables is made easy.

The various relations between different tables, the number of fields in each table and the type, width of each field were analyses. The names of the fields and tables where so chosen that the users would not face any problem in identifying the table structure.

TABLE DESIGN

**Table name : ** Student Registration

Primary key : srno

Column name	**Data type**	**Width**	**Description**
srno	integer	10	Student register number
sname	char	15	Student name
gen	varchar	20	Gender
dob	integer	15	Date of birth
fname	char	10	Father name
foccu	char	15	Father occupation
speradd	varchar	20	Student permanent address
phno	integer	10	Phone number
cid	integer	10	Course id
cname	char	15	Course name
dep	varchar	20	Department
dur	varchar	25	Duration
cate	varchar	15	Category

**Table name : ** Course

Primary key : cid

Column name	**Data type**	**Width**	**Description**
cid	integer	10	Course id

cname	char	15	Course name
cdur	varchar	20	Course duration
cdep	char	15	Course department
ccat	char	10	Course category

Table name : Exam

Primary key : hallno

Column name	**Data type**	**Width**	**Description**
hallno	integer	10	Hall number
freg	integer	10	From register
treg	char	15	To register
noon	char	15	After noon
timing	char	10	Timing
sheet	integer	05	Sheet

Table name : Staff

Primary key : sid

Column name	**Data type**	**Width**	**Description**
sid	integer	10	Staff id
sname	integer	10	Staff name
gen	char	15	Gender
dob	char	15	Date of birth
peradd	char	10	Permanent address
phno	integer	10	Phone number
quali	char	10	Qualification

dep	varchar	15	Department
cate	varchar	20	Category

**Table name : **Library

Primary key : bno

Column name	Data type	Width	Description
bno	integer	10	Book number
aname	char	10	Author name
bname	char	10	Book name
cname	char	10	Course name

**Table name : **Attendance

Primary key : sregno

Column name	Data type	Width	Description
sregno	integer	10	Student register number
sname	char	10	Student name
satten	varchar	15	Student attendance
cname	char	10	Course name

3.2.3 OUTPUT DESIGN

Output design is very important concept in the computerized system, without reliable output the user may feel the entire system is unnecessary and avoids using it. The proper output design is important in any system and facilitates effective decision-making. The output design of this system includes various reports.

Computer output is the most important and direct source of information the user. Efficient, intelligible output design should improve the system's relationships with the user and help in decision making. A major form of output is the hardcopy from the printer.

Output requirements are designed during system analysis. A good starting point for the output design is the data flow diagram. Human factors reduce issues for design involved addressing internal controls to ensure readability.

3.2.5 SYSTEM DEVELOPMENT

The Relation Model (RM) is the formal model of a database that was developed for IBM in the early 1970s by Dr. E.F. Codd. It is largely based on set theory, which makes it both powerful and easy to implement in computers. All modern relational databases are based on this model. We will use it to represent information that does not (and should not) appear in the UML model but is needed for us to build functioning databases. The table model is an informal set of terms for relational model objects. These are the terms used most often by database developers.

The Structured Query Language (SQL, pronounced "sequel" or "ess-que-ell") is used to build and manipulate relational databases. It is based on relational algebra, but provides additional capabilities that are needed in commercial systems. It is a declarative, rather than a procedural, programming language. There is a standard for this language, but products vary in how closely they implement it

4. SYSTEM TESTING AND IMPLEMENTATION

4.1 TESTING METHODOLOGIES

Testing is a series of different tests that whose primary purpose is to fully exercise the computer based system. Although each test has a different purpose, all work should verify that all system element have been properly integrated and performed allocated function. Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system. The philosophy behind testing is to find the errors. A good test is one that has a high probability of finding an undiscovered error.

TYPES OF TESTING

- Unit testing
- Integration testing
- Validation testing
- Output testing

4.1.1UNIT TESTING

The unit test results are recorded for further references. During unit testing the functions of the program unit validation and the limitations are tested. Unit testing is testing changes made in a existing or new program this test is carried out during the programming and each module is found to be working satisfactorily. For example in the registration form after entering all the fields we click the submit button.

4.1.2 INTEGRATION TESTING

The entire project was split into small program; each of this single programs gives a frame as an output. These programs were tested individually; at last all these programs where combined together by creating another program where all these constructors were used. It give a lot of problem by not functioning is an integrated manner.

4.1.3VALIDATION TESTING

At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of test i.e., Validation succeeds when the software function in a manner that can be reasonably
accepted by the customer.

4.1.4 OUTPUT TESTING

After performing the validation testing the next step is output testing of the proposed system. Since the system cannot be useful if it does not produce the required output. Asking the user about the format in which the system is required tests the output displayed or generated by the system under consideration. Here the output format is considered in two ways. one is on screen and another one is printed format. The output format on the screen is found to be corrected as the format was designed in the system phase according to the user needs.

4.2 SYSTEM IMPLEMETATION

The System implementation is stage in the project where the theoretical design is turned into the working system. The most crucial stage is giving the users confidence that the new system will work effectively and efficiently. The performance of reliability of the system is tested and it gained acceptance. The system was implemented successfully. Implementation is a process that means converting a new system in to operation. Visual Basic .NET (VB.NET) is a multi-paradigm, object-oriented programming language, implemented on the .NET Framework. Microsoft launched VB.NET in 2002 as the successor to its original Visual Basic language.

4.3 SYSTEM MAINTENANCE

Maintenance is actually implementation of the review plan as important as it is programmers and analyst is to perform or identify with him or herself with the maintenance. There are psychologically personality and professional reasons for this. Analyst and programmers spend fair more time maintaining programmer then they do writing them Maintenances account for 50-80% of total system development. Maintenance is expensive.

5. CONCLUSION

The **“COLLEGE ADMINISTRATION SYSTEM”** has been developed to satisfy all proposed requirements. This is highly scalable and user friendly. Almost all the system objectives have been met. The system has been tested under all criteria. The system minimizes the problem arising in the existing manual system and it eliminates the human errors to zero level. The software executes successfully by fulfilling the objectives of the project. Further extensions to this system can be made required with minor modifications.

6. FUTURE ENHANCEMENT

This system can be implemented in college information, In computer hardware, software, or in combinations of them. Apparatus of can be implemented in a computer program product storage device for execution by a programmable processor; and method steps of performed by a programmable processor executing a program of instructions to perform functions of operating on input data and generating output.

BIBLIOGRAPHY

BOOK REFERENCES

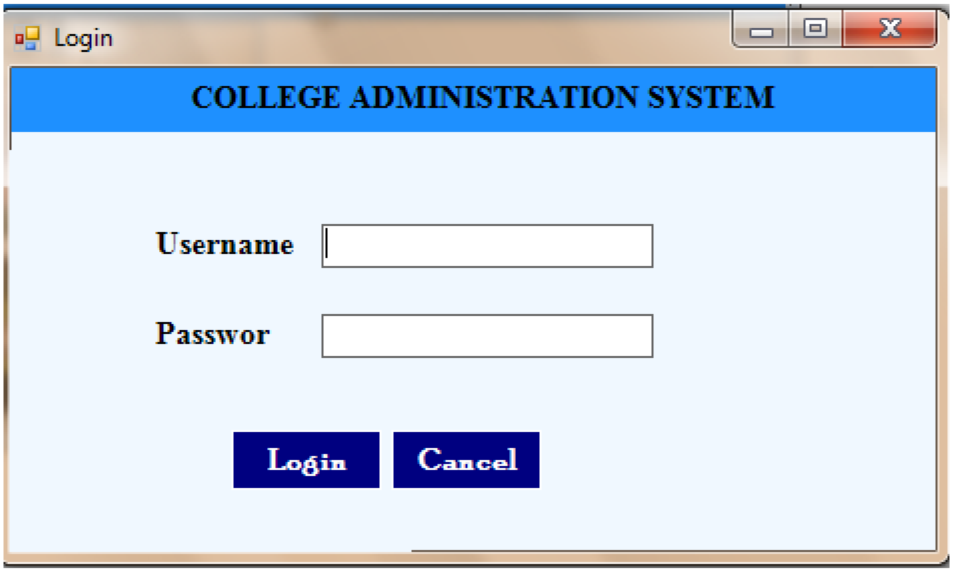
1. Elias Awath, “SYSTEM ANALYSIS AND DESIGN”, Tata Mc Graw Hill Publication, Sixth Edition,2003
2. S. Ramachandran, “COMPUTER AIDED DESIGN”, Air Walk Publication, Third Edition,2003
3. Richard Fairley,”SOFTWARE ENGINEERING CONCEPTS”, Tata Mc Graw Hill Publication, Second Edition,1997
4. Programming VB.NET: A Guide for Experienced Programmers by Gary Cornell, Jonathan Morrison
5. Learning VB.NET Through Applications by Clayton Crooks II
6. VB .NET How to Program (2nd Edition) by Harvey M. Deitel, Paul J. Deitel, Tem R. Nieto.

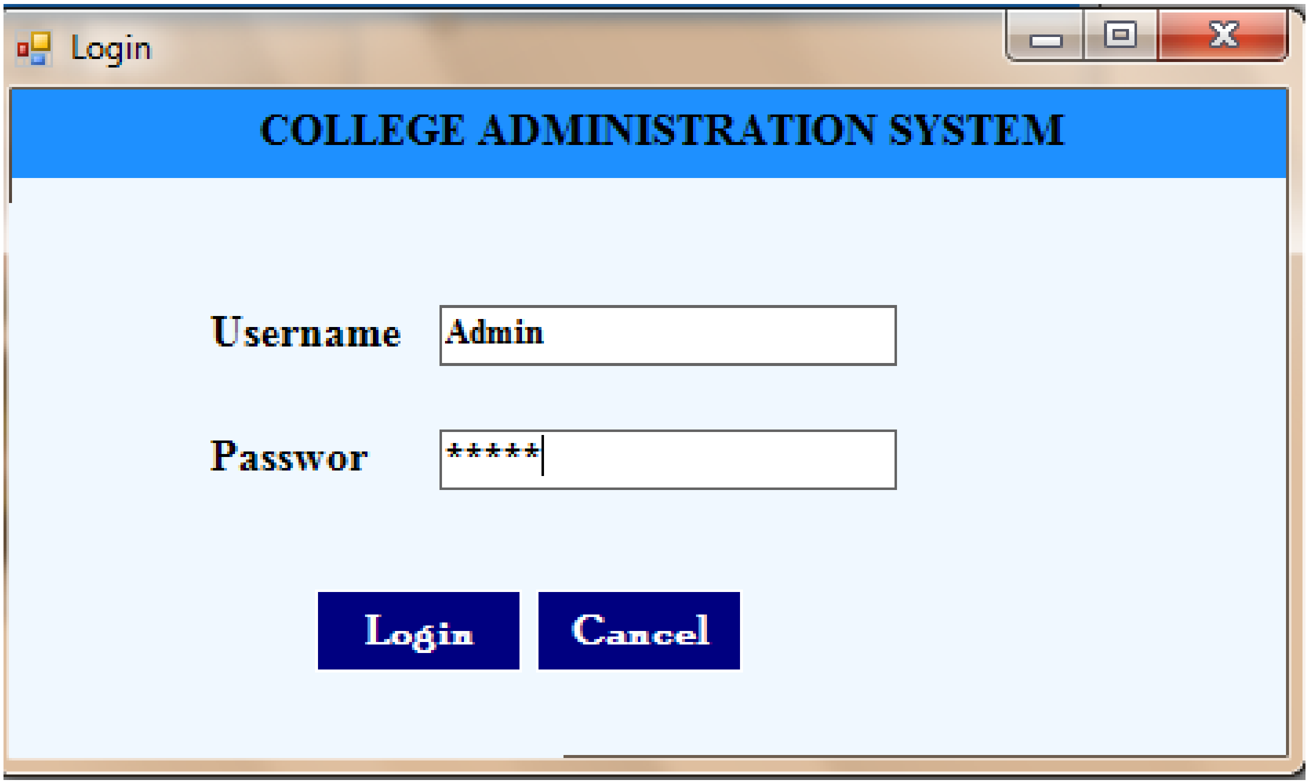
ONLINE REFERENCES

1. www.msdn.microsoft.com
2. www.microsoftdotnet.net
3. www.dotnet-tricks.com
4. www.sqlmag.com

Appendix

Admin login



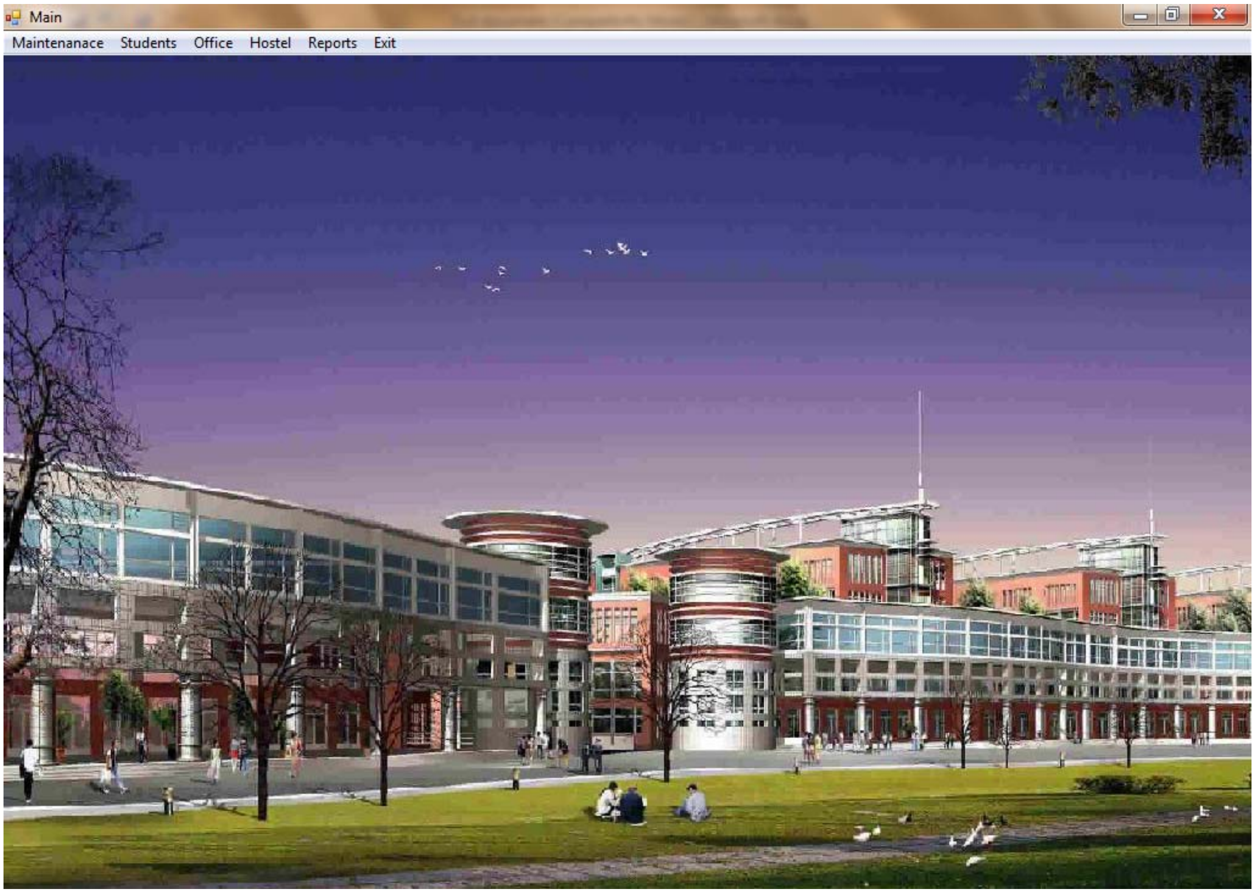


COLLEGE ADMINISTRATION SYSTEM

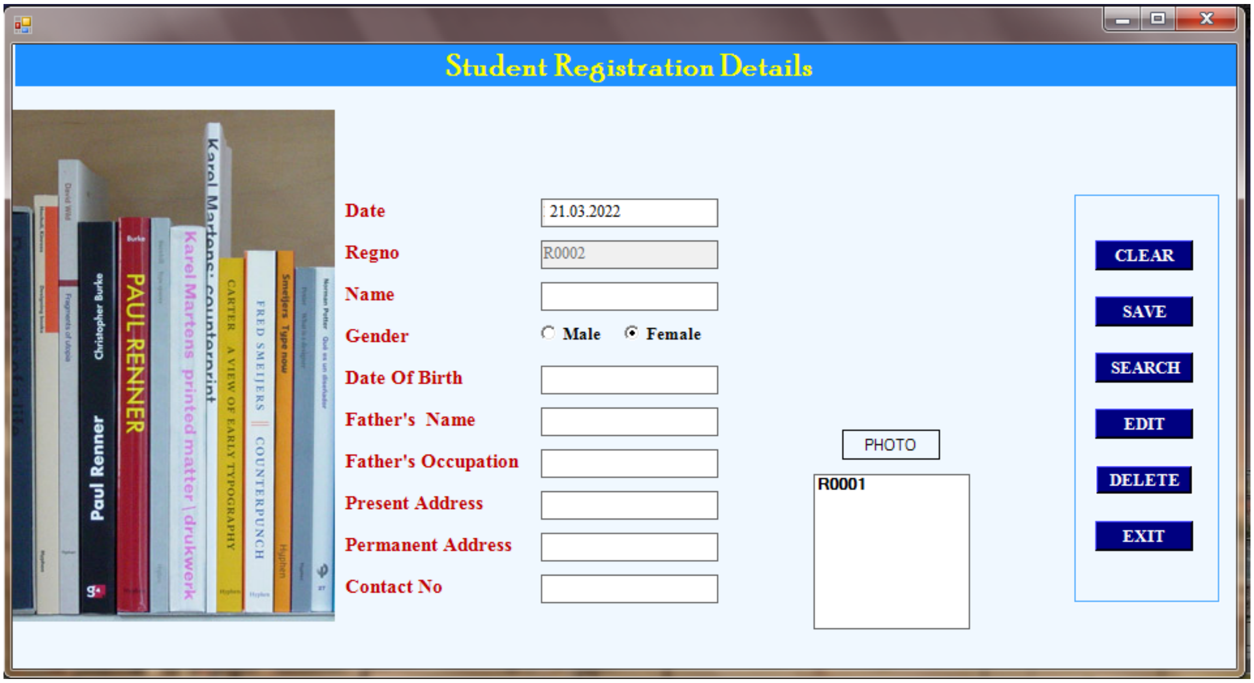
Username Admin

Passwor *****

Main form



Student registration



Student Registration Details

Date: 21.03.2022

Regno: R0002

Name:

Gender: ○ Male ● Female

Date Of Birth:

Father's Name:

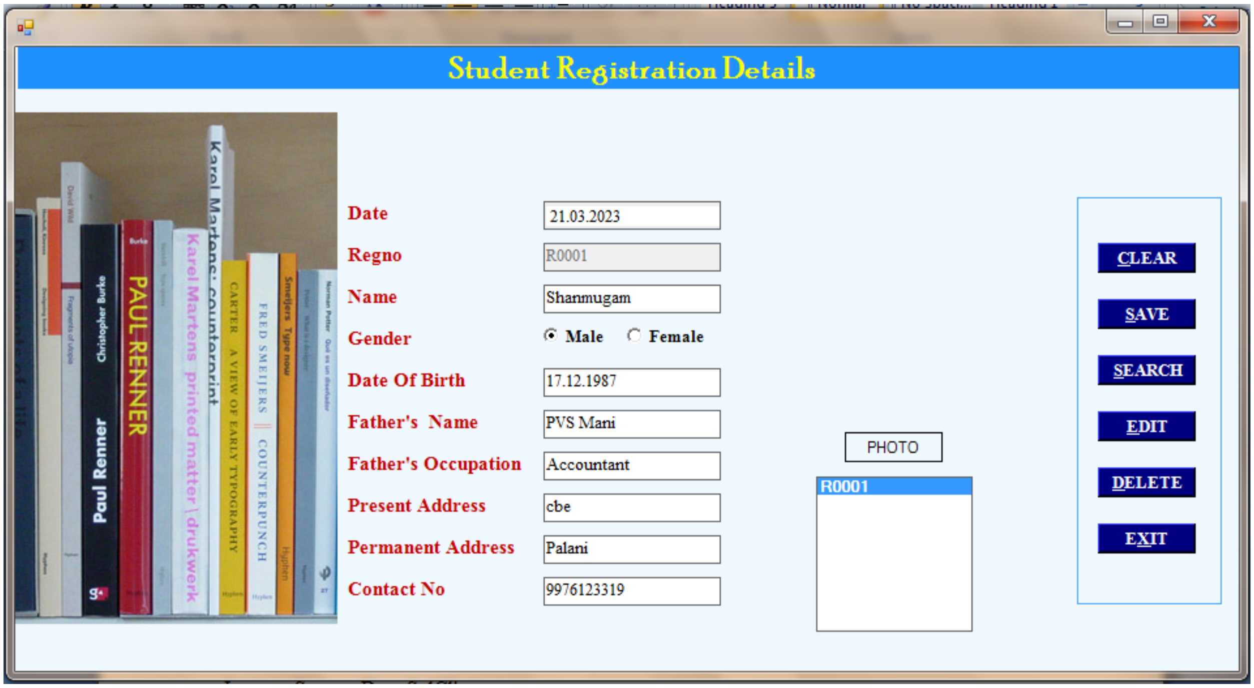
Father's Occupation:

Present Address:

Permanent Address:

Contact No:

R0001

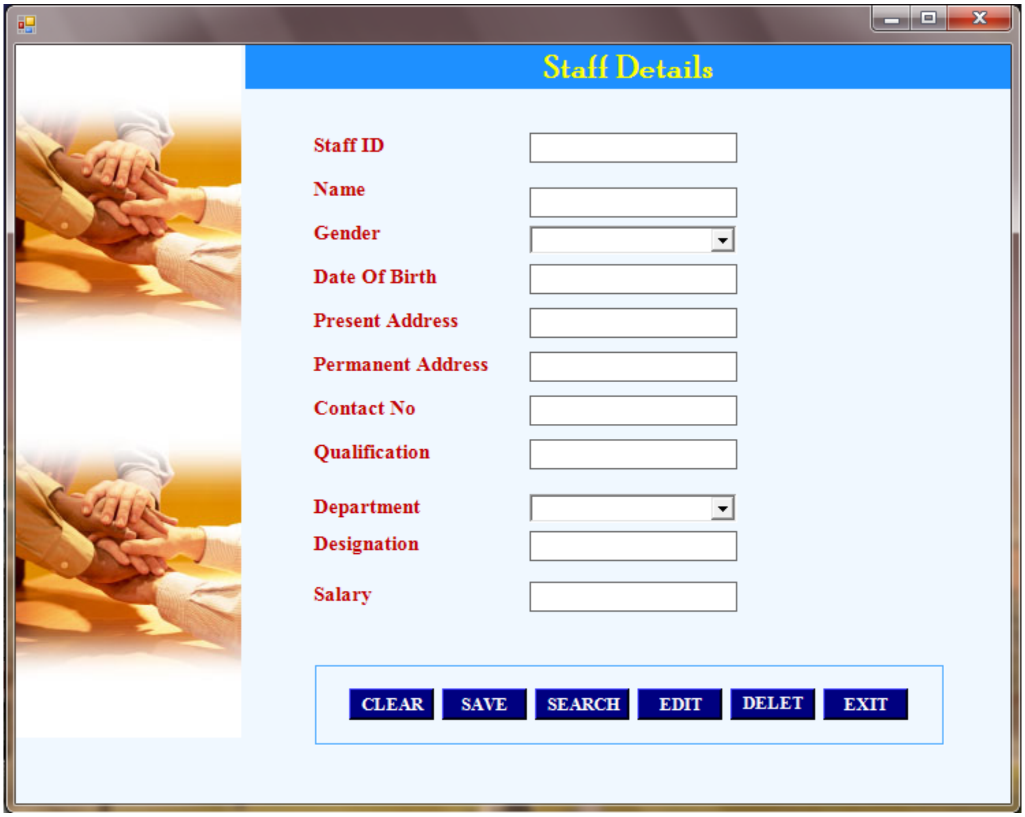


Student Registration Details

Date	21.03.2023
Regno	R0001
Name	Shanmugam
Gender	● Male ○ Female
Date Of Birth	17.12.1987
Father's Name	PVS Mani
Father's Occupation	Accountant
Present Address	cbe
Permanent Address	Palani
Contact No	9976123319

R0001

Staff details



Staff Details

Staff ID

Name

Gender

Date Of Birth

Present Address

Permanent Address

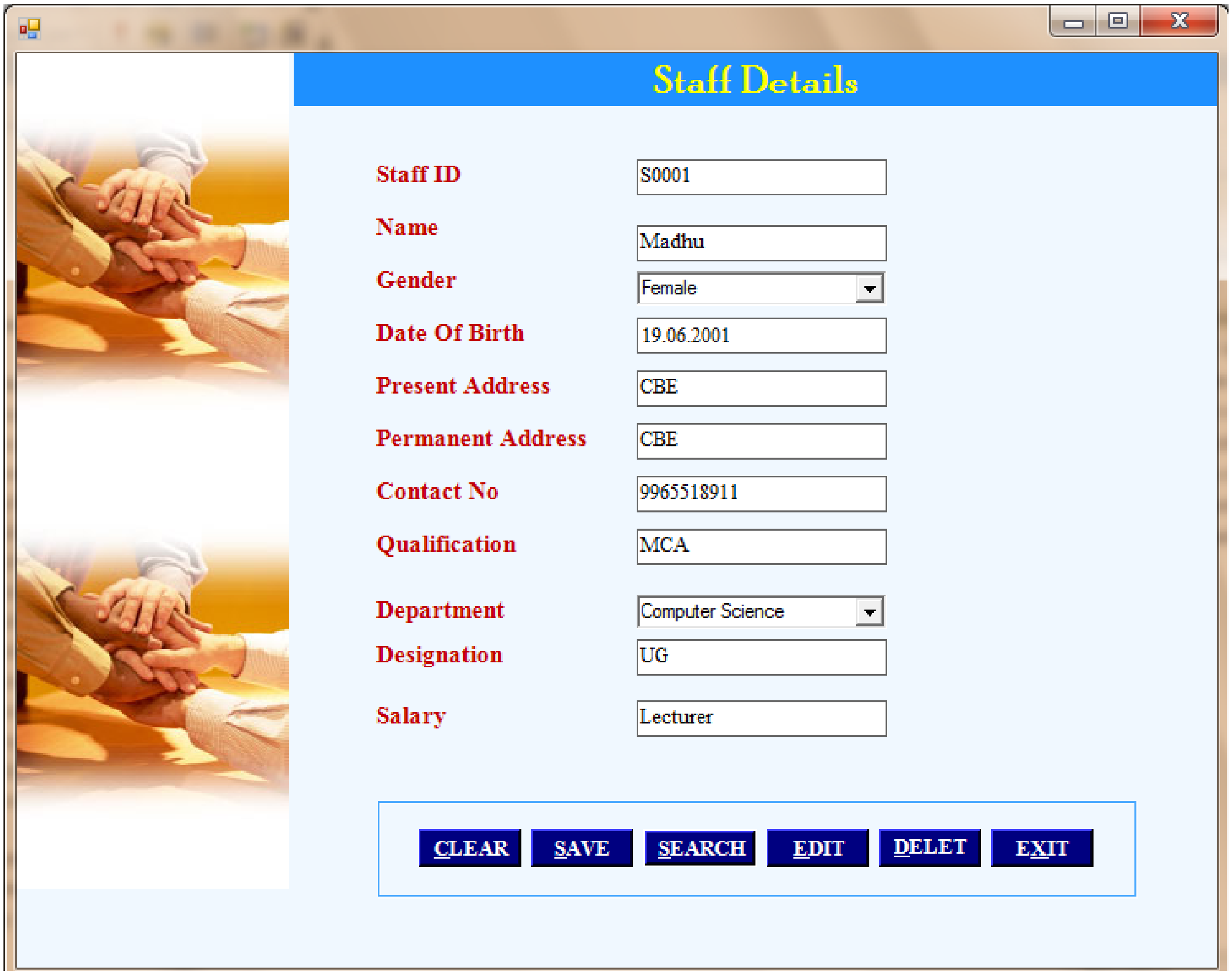
Contact No

Qualification

Department

Designation

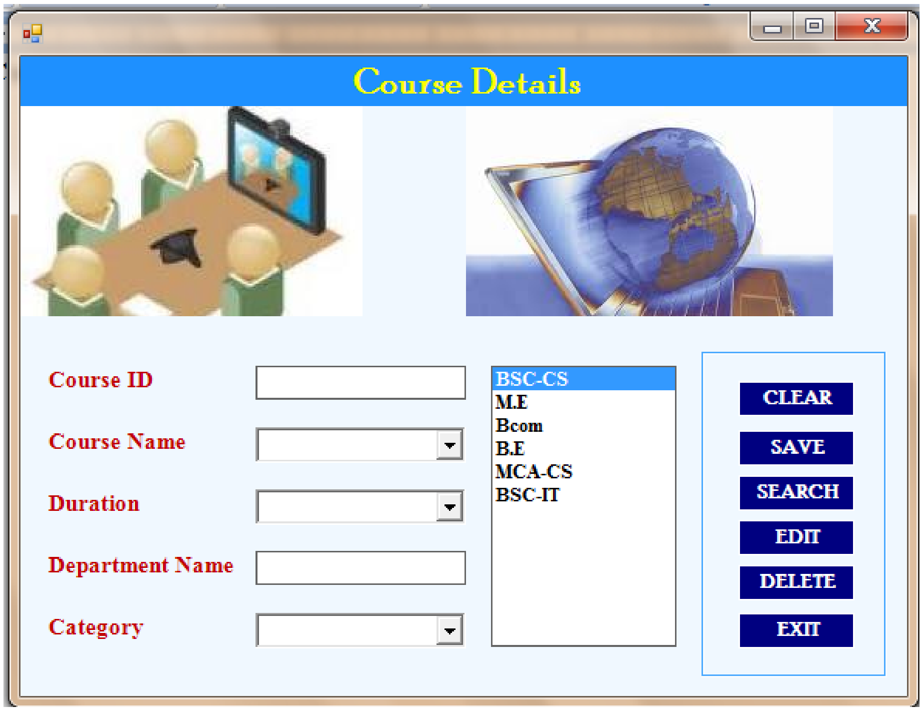
Salary



Staff Details

Staff ID	S0001
Name	Madhu
Gender	Female
Date Of Birth	19.06.2001
Present Address	CBE
Permanent Address	CBE
Contact No	9965518911
Qualification	MCA
Department	Computer Science
Designation	UG
Salary	Lecturer

Course details



Course Details

Course ID

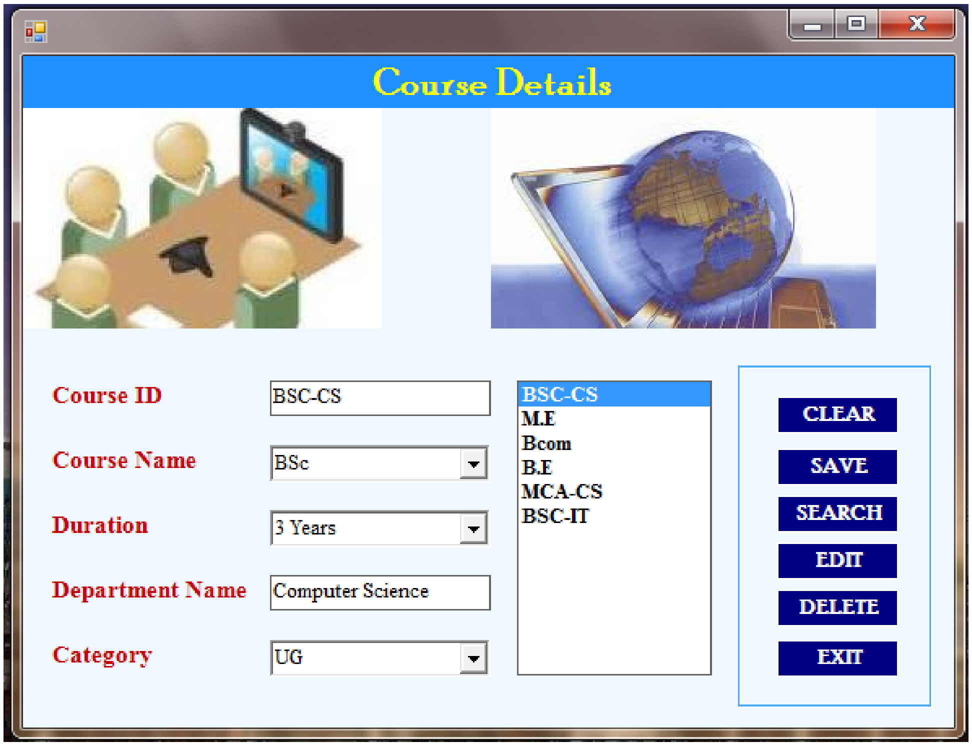
Course Name

Duration

Department Name

Category

BSC-CS
M.E
Bcom
B.E
MCA-CS
BSC-IT



Course Details

Course ID: BSC-CS

Course Name: BSc

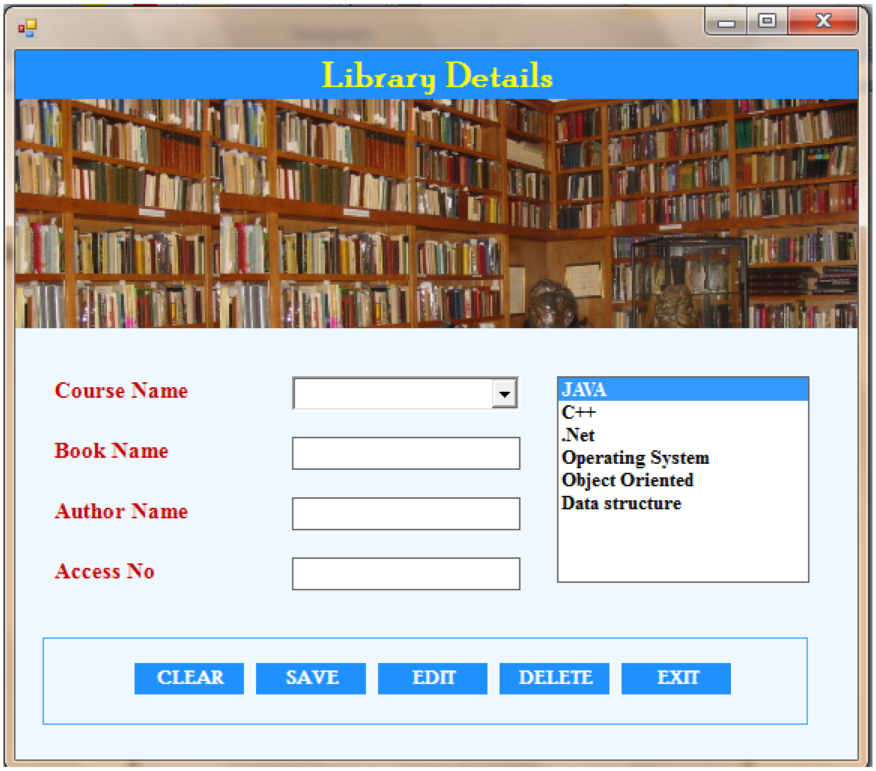
Duration: 3 Years

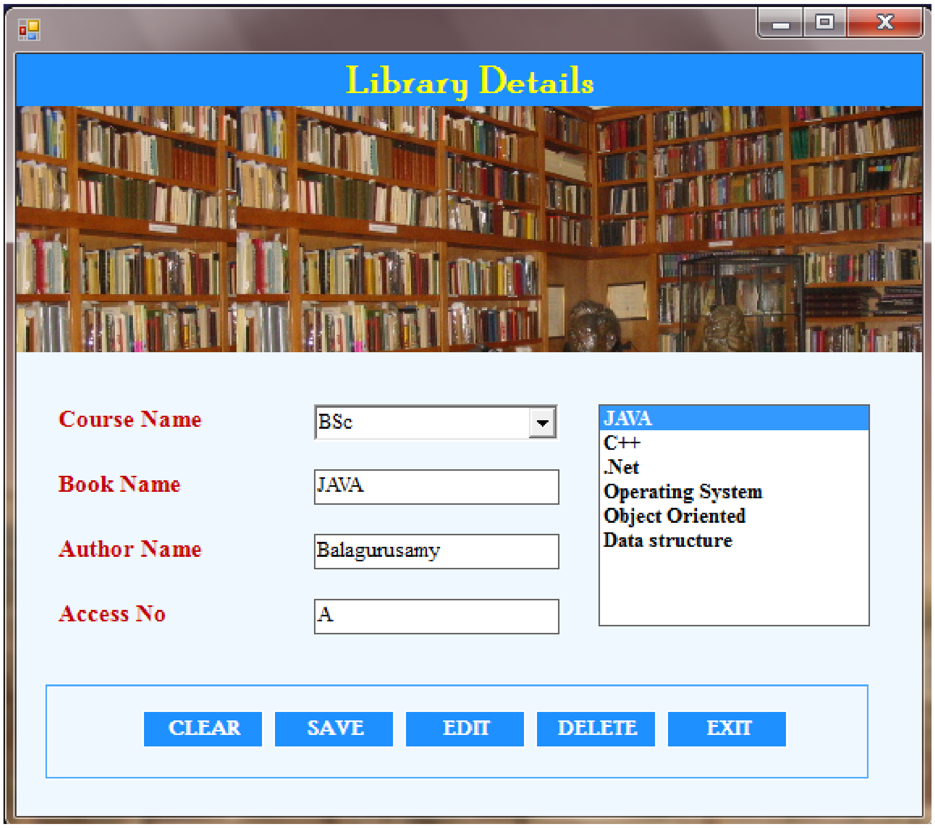
Department Name: Computer Science

Category: UG

BSC-CS
M.E
Bcom
B.E
MCA-CS
BSC-IT

Library details





Library Details

Course Name: BSc

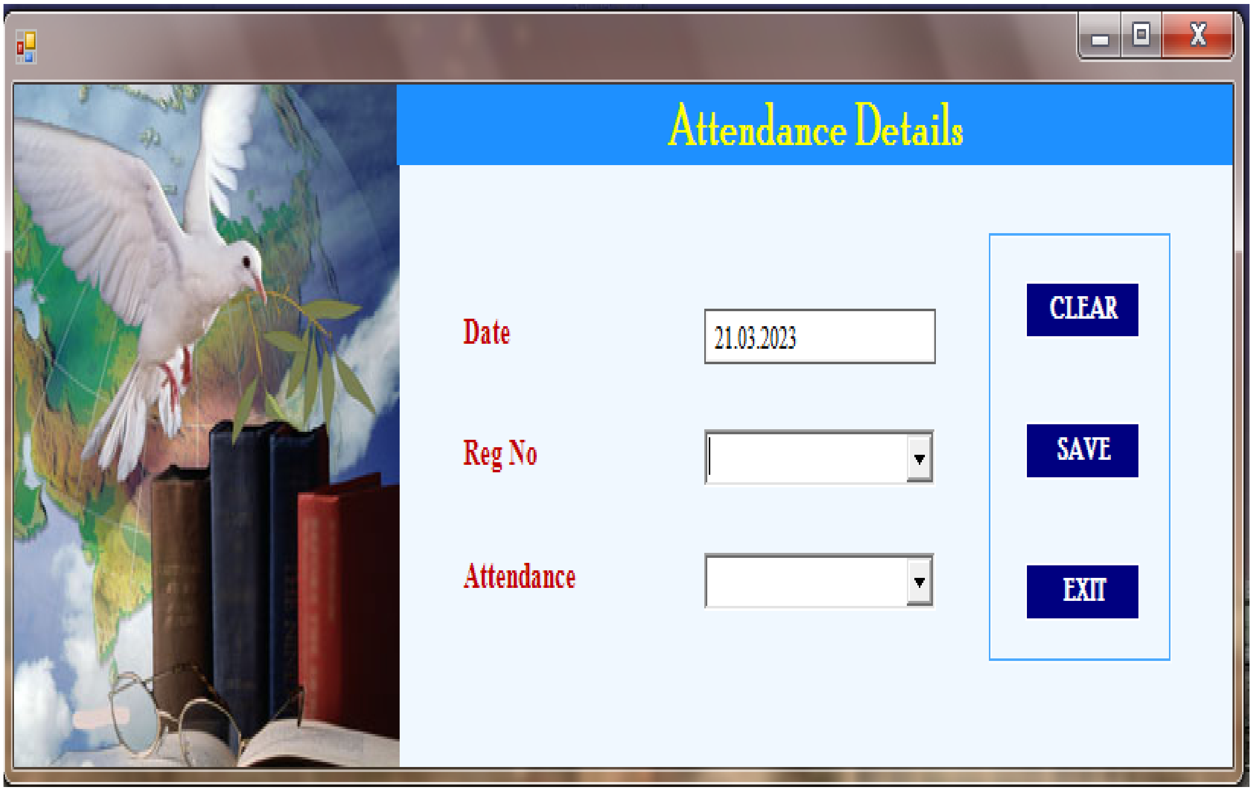
Book Name: JAVA

Author Name: Balagurusamy

Access No: A

JAVA
C++
.Net
Operating System
Object Oriented
Data structure

Attendance details

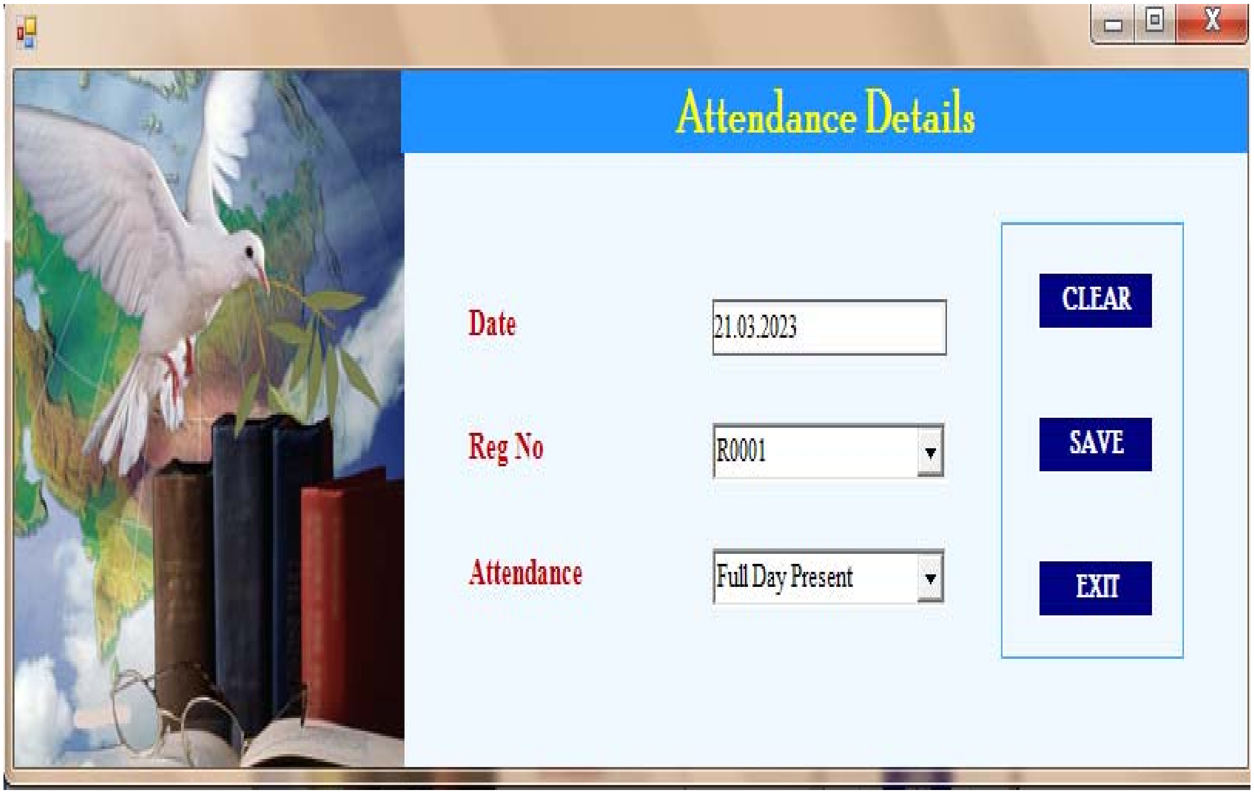


Attendance Details

Date 21.03.2023

Reg No

Attendance



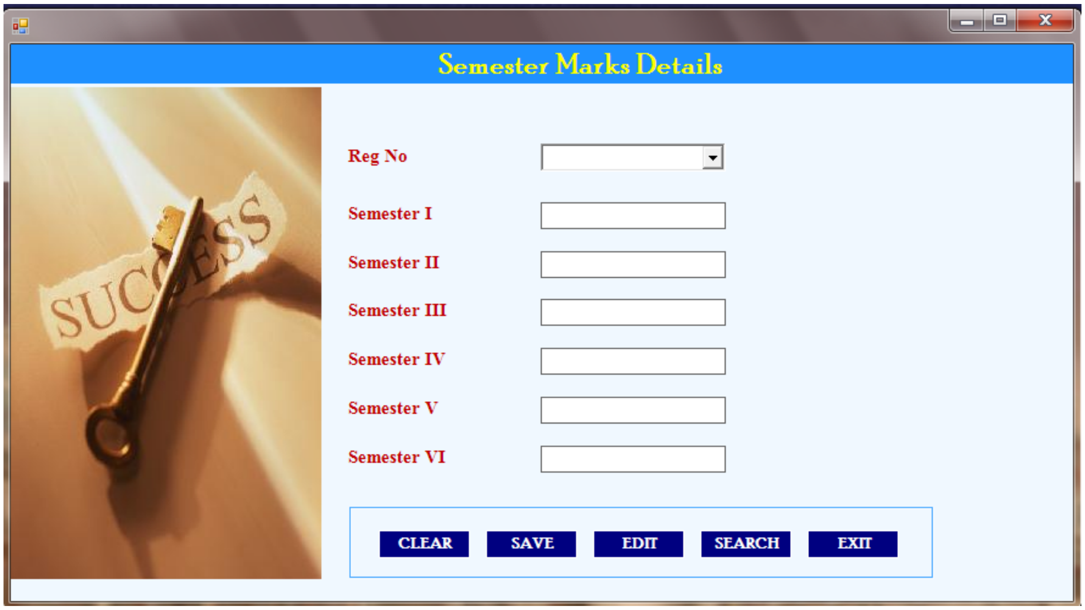
Attendance Details

Date 21.03.2023

Reg No R0001

Attendance Full Day Present

Semester mark details



Semester Marks Details

Reg No

Semester I

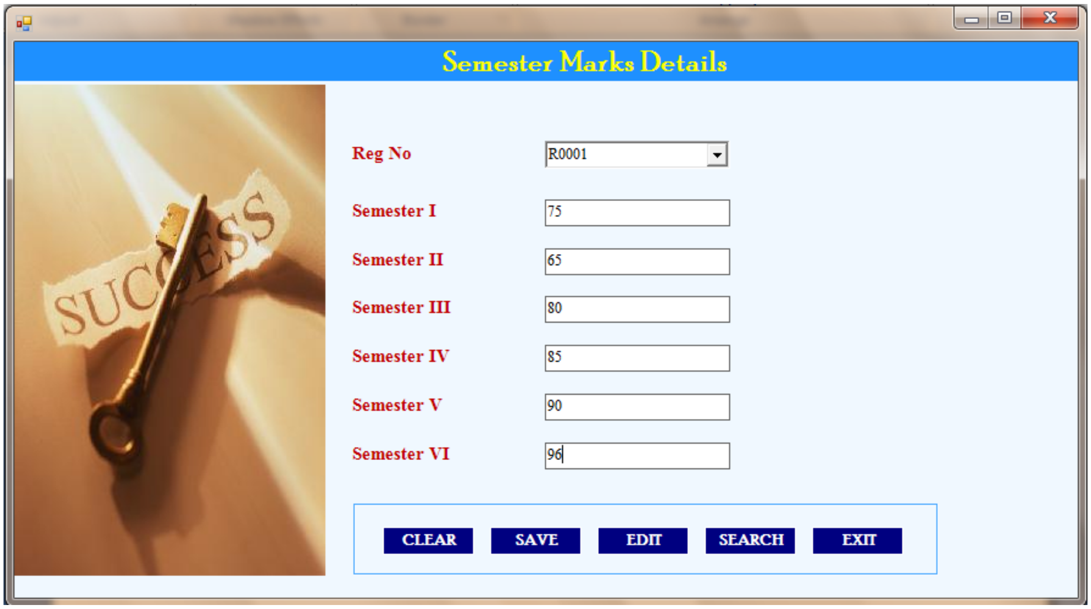
Semester II

Semester III

Semester IV

Semester V

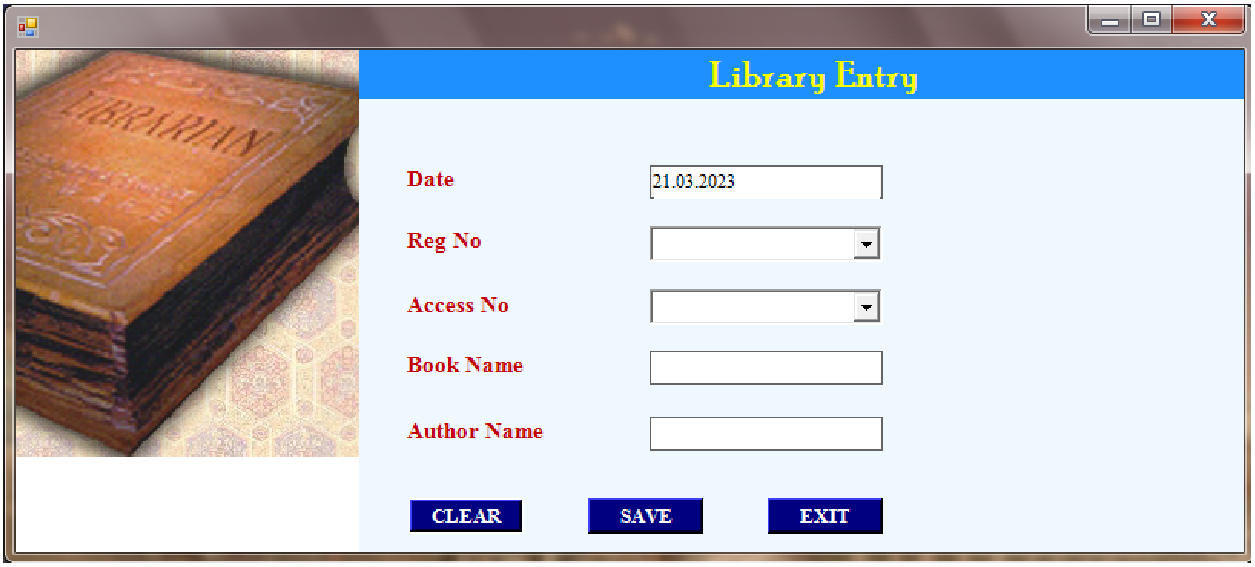
Semester VI



Semester Marks Details

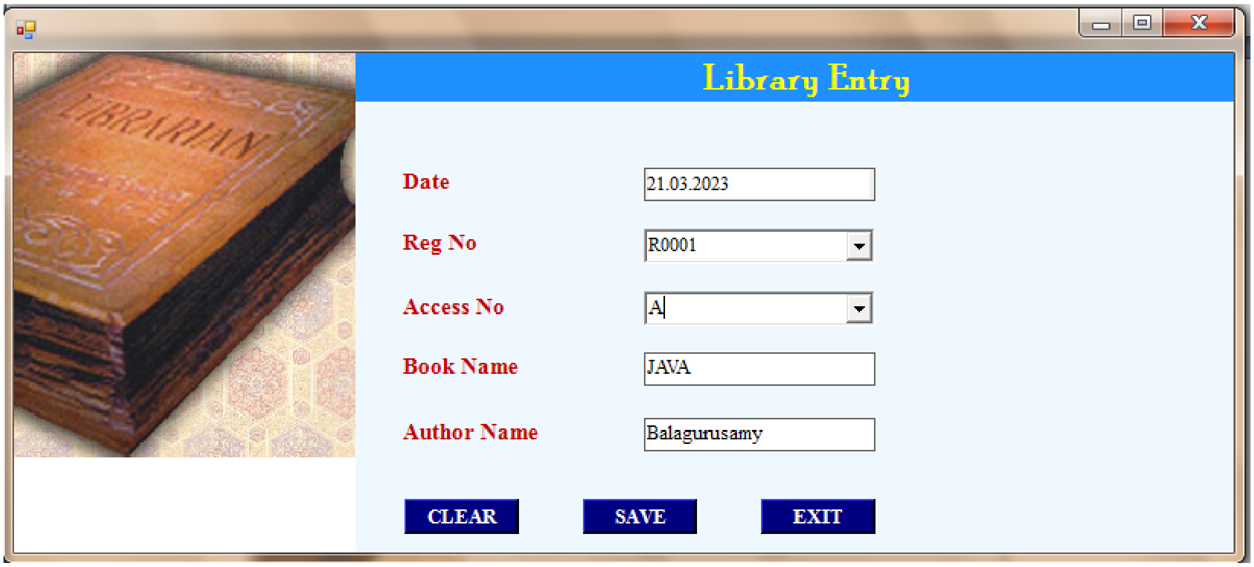
Reg No	R0001
Semester I	75
Semester II	65
Semester III	80
Semester IV	85
Semester V	90
Semester VI	96

Library entry details



Library Entry

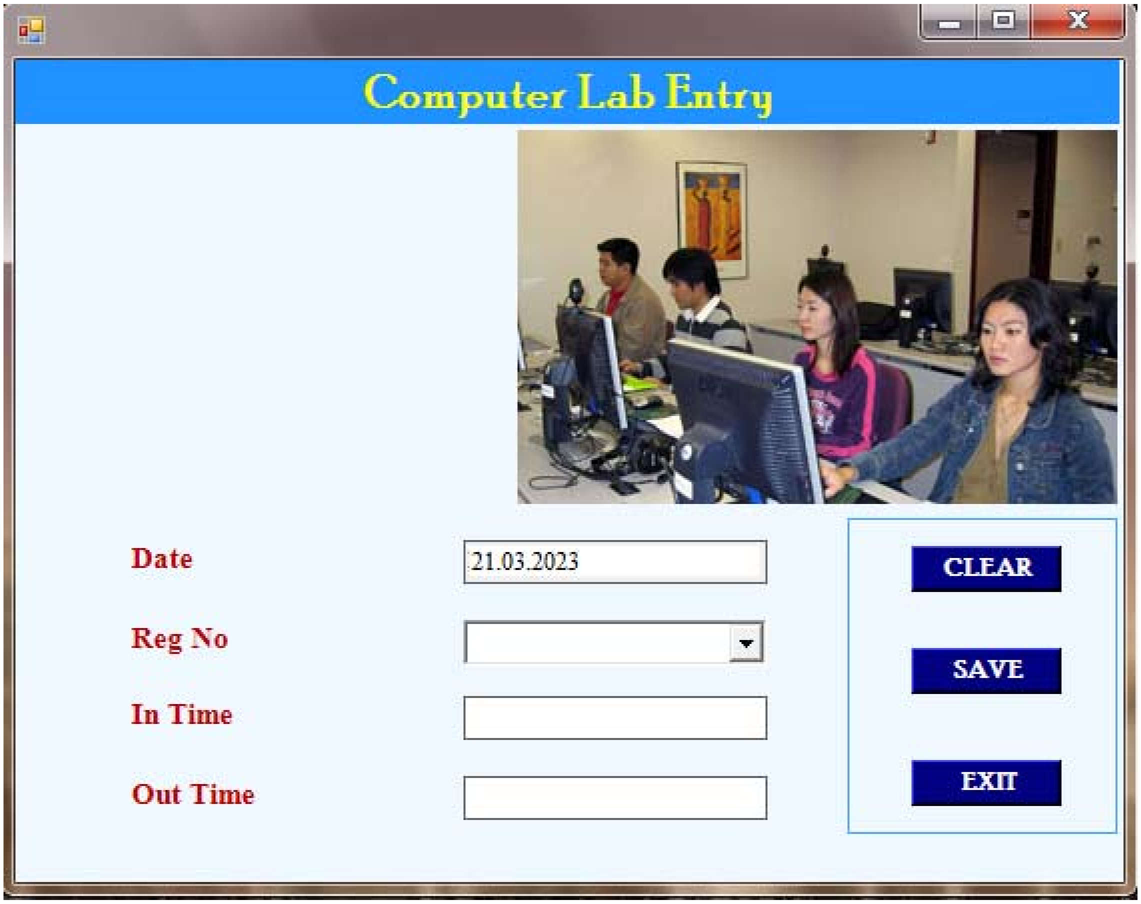
Date	21.03.2023
Reg No	
Access No	
Book Name	
Author Name	



Library Entry

Date	21.03.2023
Reg No	R0001
Access No	A
Book Name	JAVA
Author Name	Balagurusamy

Lab entry



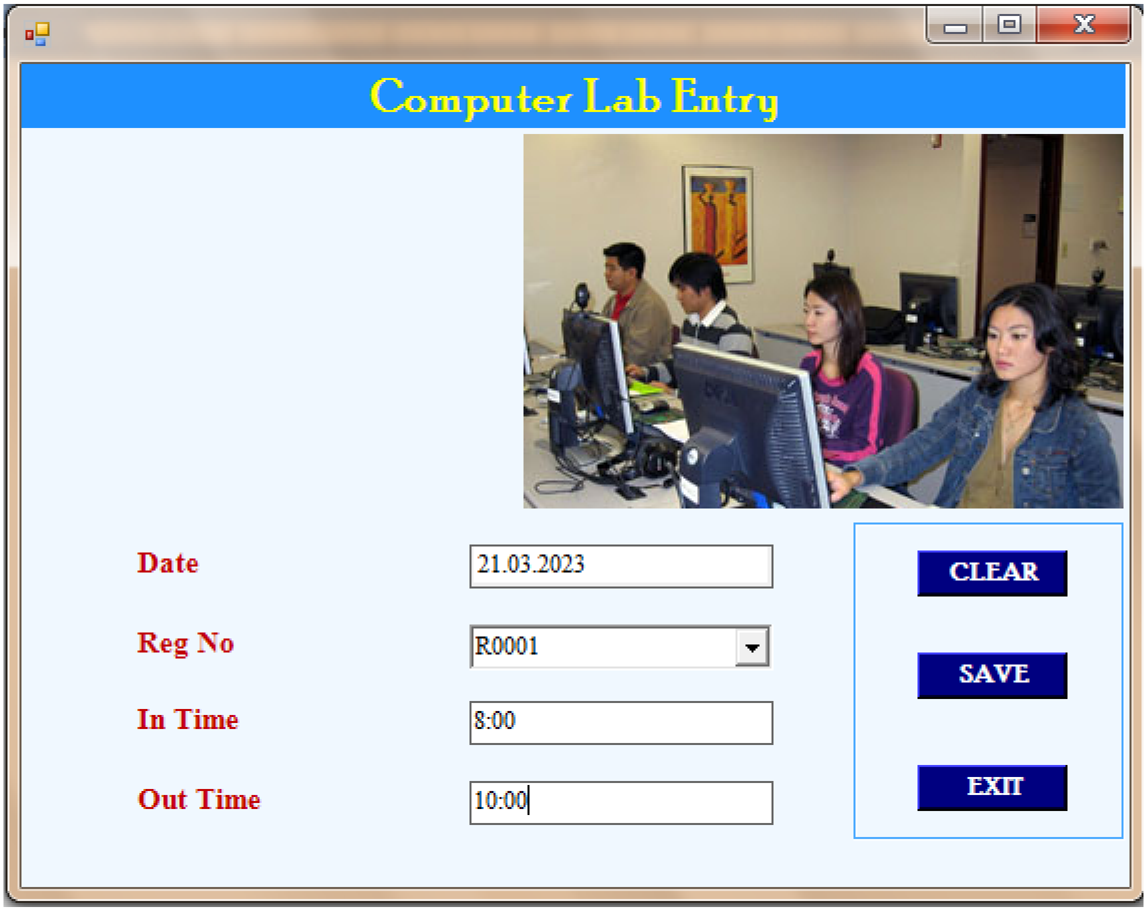
Computer Lab Entry

Date 21.03.2023

Reg No

In Time

Out Time



Computer Lab Entry

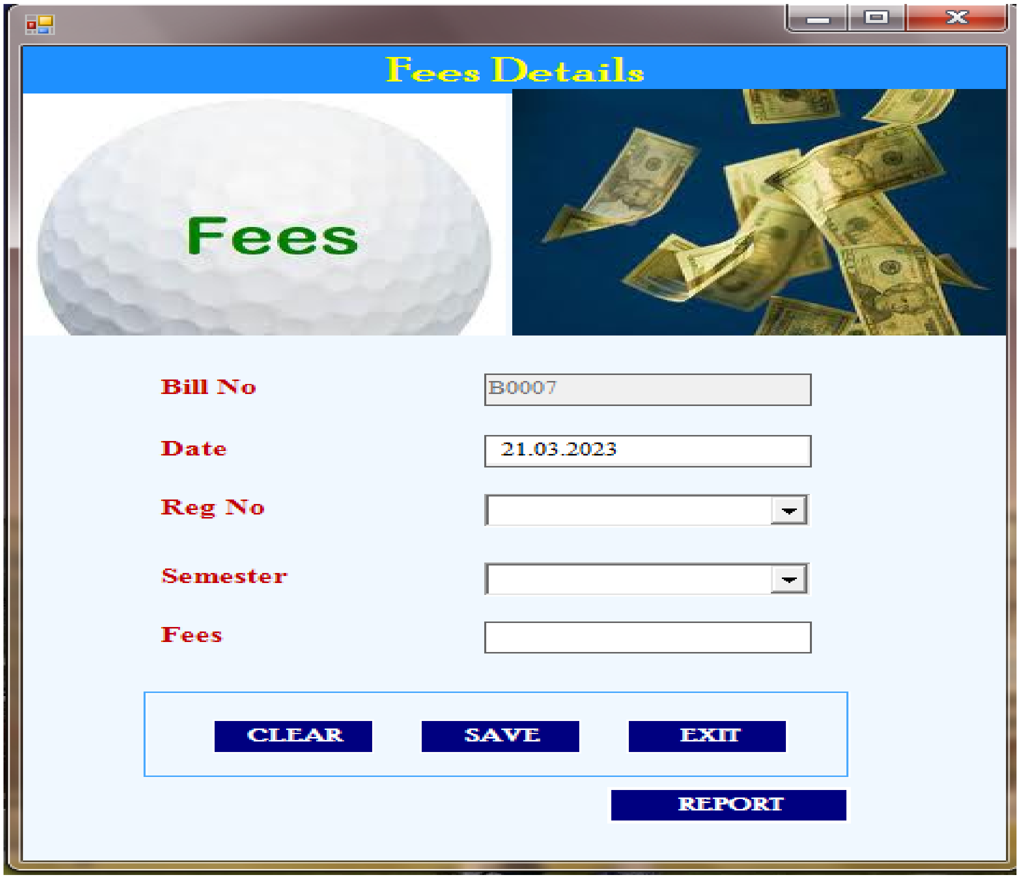
Date 21.03.2023

Reg No R0001

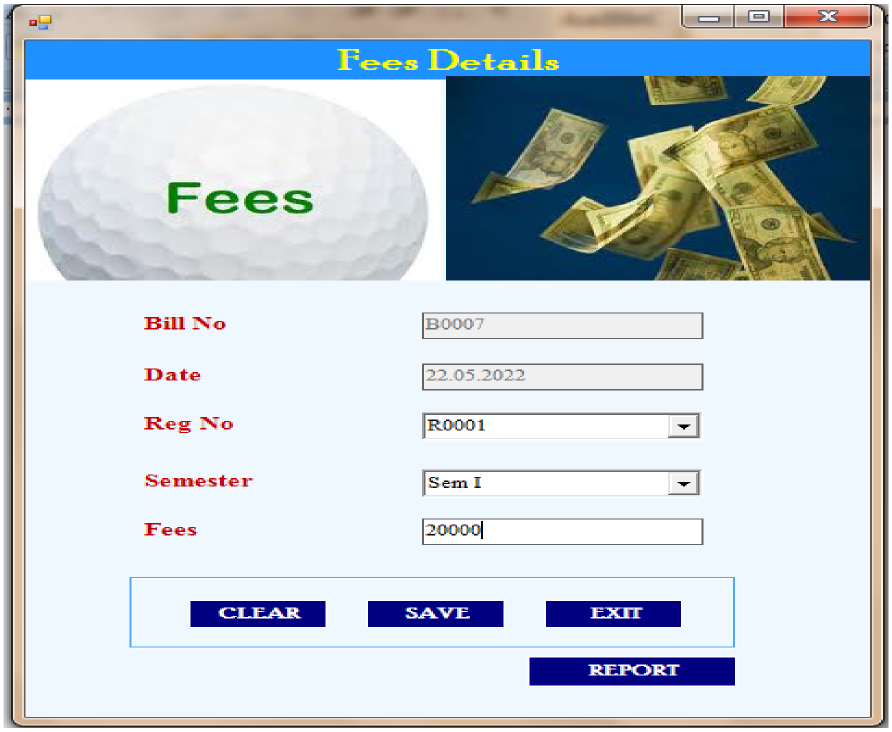
In Time 8:00

Out Time 10:00

Fees details



Bill No	B0007
Date	21.03.2023
Reg No	
Semester	
Fees	



Fees Details

Bill No	B0007
Date	22.05.2022
Reg No	R0001
Semester	Sem I
Fees	20000

SAMPLE CODING

```
Imports System.Data.SqlClient

Public Class course
Inherits System.Windows.Forms.Form
Dim con As SqlConnection = New SqlConnection("Data Source=HELLO-4\SQLEXPRESS;Initial Catalog=campus;Persist Security Info=True;User ID=sa;Password=sql")
Dim com As SqlCommand
Dim qry As String
Dim adp As SqlDataAdapter
Dim ds As DataSet
Dim i As Integer
Friend WithEvents Button1 As System.Windows.Forms.Button
Dim p As String



#Region " Windows Form Designer generated code "

Public Sub New()
MyBase.New()

'This call is required by the Windows Form Designer.
InitializeComponent()

'Add any initialization after the InitializeComponent() call

End Sub
```

```
'Form overrides dispose to clean up the component list.
Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
If disposing Then
If Not (components Is Nothing) Then
components.Dispose()
End If
End If
MyBase.Dispose(disposing)
End Sub

'Required by the Windows Form Designer
Private components As System.ComponentModel.IContainer

'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.
Friend WithEvents PictureBox1 As System.Windows.Forms.PictureBox
Friend WithEvents PictureBox2 As System.Windows.Forms.PictureBox
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents Label7 As System.Windows.Forms.Label
Friend WithEvents Label6 As System.Windows.Forms.Label
Friend WithEvents Label3 As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents Label5 As System.Windows.Forms.Label
Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
Friend WithEvents bclear As System.Windows.Forms.Button
Friend WithEvents bsave As System.Windows.Forms.Button
Friend WithEvents bedit As System.Windows.Forms.Button
Friend WithEvents bdel As System.Windows.Forms.Button
Friend WithEvents bexit As System.Windows.Forms.Button
Friend WithEvents ccat As System.Windows.Forms.ComboBox
Friend WithEvents cid As System.Windows.Forms.TextBox
Friend WithEvents cdept As System.Windows.Forms.TextBox
Friend WithEvents cname As System.Windows.Forms.ComboBox
Friend WithEvents lcid As System.Windows.Forms.ListBox
Friend WithEvents cdur As System.Windows.Forms.ComboBox
<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
Dim resources As System.ComponentModel.ComponentResourceManager = New System.ComponentModel.ComponentResourceManager(GetType(course))
Me.PictureBox1 = New System.Windows.Forms.PictureBox
Me.PictureBox2 = New System.Windows.Forms.PictureBox
Me.Label1 = New System.Windows.Forms.Label
Me.Label7 = New System.Windows.Forms.Label
Me.ccat = New System.Windows.Forms.ComboBox
Me.Label6 = New System.Windows.Forms.Label
Me.lcid = New System.Windows.Forms.ListBox
Me.cid = New System.Windows.Forms.TextBox
Me.Label3 = New System.Windows.Forms.Label
Me.Label2 = New System.Windows.Forms.Label
Me.Label5 = New System.Windows.Forms.Label
Me.GroupBox1 = New System.Windows.Forms.GroupBox
Me.Button1 = New System.Windows.Forms.Button
Me.bclear = New System.Windows.Forms.Button
Me.bsave = New System.Windows.Forms.Button
Me.bedit = New System.Windows.Forms.Button
Me.bdel = New System.Windows.Forms.Button
Me.bexit = New System.Windows.Forms.Button
Me.cdept = New System.Windows.Forms.TextBox
Me.cname = New System.Windows.Forms.ComboBox
Me.cdur = New System.Windows.Forms.ComboBox
CType(Me.PictureBox1, System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.PictureBox2, System.ComponentModel.ISupportInitialize).BeginInit()
Me.GroupBox1.SuspendLayout()
Me.SuspendLayout()
'
'PictureBox1
'
Me.PictureBox1.Image          =          CType(resources.GetObject("PictureBox1.Image"),
System.Drawing.Image)
Me.PictureBox1.Location = New System.Drawing.Point(0, 32)
Me.PictureBox1.Name = "PictureBox1"
Me.PictureBox1.Size = New System.Drawing.Size(288, 136)
Me.PictureBox1.TabIndex = 0
Me.PictureBox1.TabStop = False
'
'PictureBox2
'
Me.PictureBox2.Image          =          CType(resources.GetObject("PictureBox2.Image"),
System.Drawing.Image)
Me.PictureBox2.Location = New System.Drawing.Point(288, 32)
Me.PictureBox2.Name = "PictureBox2"
Me.PictureBox2.Size = New System.Drawing.Size(288, 136)
Me.PictureBox2.TabIndex = 1
Me.PictureBox2.TabStop = False
'
'Label1
'
Me.Label1.BackColor = System.Drawing.Color.DodgerBlue
Me.Label1.Font = New System.Drawing.Font("Poor Richard", 18.0!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label1.ForeColor = System.Drawing.Color.Yellow
Me.Label1.Location = New System.Drawing.Point(0, 0)
Me.Label1.Name = "Label1"
Me.Label1.Size = New System.Drawing.Size(576, 32)
Me.Label1.TabIndex = 3
Me.Label1.Text = "Course Details"
Me.Label1.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
'
'Label7
'
Me.Label7.Font = New System.Drawing.Font("Times New Roman", 11.25!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label7.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer), CType(CType(0, Byte), Integer), CType(CType(0, Byte), Integer))
Me.Label7.Location = New System.Drawing.Point(16, 320)
Me.Label7.Name = "Label7"
Me.Label7.Size = New System.Drawing.Size(128, 23)
Me.Label7.TabIndex = 35
Me.Label7.Text = "Department Name"
'
'ccat
'
Me.ccat.Font = New System.Drawing.Font("Times New Roman", 9.75!, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.ccat.Location = New System.Drawing.Point(152, 360)
Me.ccat.Name = "ccat"
Me.ccat.Size = New System.Drawing.Size(136, 23)
Me.ccat.TabIndex = 34
```

```
'
'Label6
'
Me.Label6.Font = New System.Drawing.Font("Times New Roman", 11.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label6.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(0, Byte), Integer), CType(CType(0, Byte), Integer))
Me.Label6.Location = New System.Drawing.Point(16, 360)
Me.Label6.Name = "Label6"
Me.Label6.Size = New System.Drawing.Size(100, 23)
Me.Label6.TabIndex = 33
Me.Label6.Text = "Category"
'
'lcid
'
Me.lcid.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle
Me.lcid.Font = New System.Drawing.Font("Times New Roman", 9.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.lcid.ItemHeight = 15
Me.lcid.Location = New System.Drawing.Point(304, 200)
Me.lcid.Name = "lcid"
Me.lcid.Size = New System.Drawing.Size(120, 182)
Me.lcid.TabIndex = 27
'
'cid
'
Me.cid.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle
Me.cid.Font = New System.Drawing.Font("Times New Roman", 9.75!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.cid.Location = New System.Drawing.Point(152, 200)
Me.cid.Name = "cid"
Me.cid.Size = New System.Drawing.Size(136, 22)
Me.cid.TabIndex = 23
'
'Label3
'
Me.Label3.Font = New System.Drawing.Font("Times New Roman", 11.25!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label3.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer), CType(CType(0, Byte), Integer), CType(CType(0, Byte), Integer))
Me.Label3.Location = New System.Drawing.Point(16, 280)
Me.Label3.Name = "Label3"
Me.Label3.Size = New System.Drawing.Size(100, 23)
Me.Label3.TabIndex = 21
Me.Label3.Text = "Duration"
'
'Label2
'
Me.Label2.Font = New System.Drawing.Font("Times New Roman", 11.25!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label2.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer), CType(CType(0, Byte), Integer), CType(CType(0, Byte), Integer))
Me.Label2.Location = New System.Drawing.Point(16, 240)
Me.Label2.Name = "Label2"
Me.Label2.Size = New System.Drawing.Size(100, 23)
Me.Label2.TabIndex = 20
Me.Label2.Text = "Course Name"
'
'Label5
'
```

```
Me.Label5.Font = New System.Drawing.Font("Times New Roman", 11.25!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label5.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer), CType(CType(0, Byte), Integer), CType(CType(0, Byte), Integer))
Me.Label5.Location = New System.Drawing.Point(16, 200)
Me.Label5.Name = "Label5"
Me.Label5.Size = New System.Drawing.Size(100, 23)
Me.Label5.TabIndex = 19
Me.Label5.Text = "Course ID"
'
'GroupBox1
'
Me.GroupBox1.Controls.Add(Me.Button1)
Me.GroupBox1.Controls.Add(Me.bclear)
Me.GroupBox1.Controls.Add(Me.bsave)
Me.GroupBox1.Controls.Add(Me.bedit)
Me.GroupBox1.Controls.Add(Me.bdel)
Me.GroupBox1.Controls.Add(Me.bexit)
Me.GroupBox1.Location = New System.Drawing.Point(440, 186)
Me.GroupBox1.Name = "GroupBox1"
Me.GroupBox1.Size = New System.Drawing.Size(120, 216)
Me.GroupBox1.TabIndex = 65
Me.GroupBox1.TabStop = False
'
'Button1
'
Me.Button1.BackColor = System.Drawing.Color.Navy
Me.Button1.FlatStyle = System.Windows.Forms.FlatStyle.Flat
Me.Button1.Font = New System.Drawing.Font("Poor Richard", 9.75!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Button1.ForeColor = System.Drawing.Color.White
Me.Button1.Location = New System.Drawing.Point(24, 85)
Me.Button1.Name = "Button1"
Me.Button1.Size = New System.Drawing.Size(75, 23)
Me.Button1.TabIndex = 33
Me.Button1.Text = "&SEARCH"
Me.Button1.UseVisualStyleBackColor = False
'
'bclear
'
Me.bclear.BackColor = System.Drawing.Color.Navy
Me.bclear.FlatStyle = System.Windows.Forms.FlatStyle.Flat
Me.bclear.Font = New System.Drawing.Font("Poor Richard", 9.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.bclear.ForeColor = System.Drawing.Color.White
Me.bclear.Location = New System.Drawing.Point(24, 24)
Me.bclear.Name = "bclear"
Me.bclear.Size = New System.Drawing.Size(75, 23)
Me.bclear.TabIndex = 28
Me.bclear.Text = "&CLEAR"
Me.bclear.UseVisualStyleBackColor = False
'
'bsave
'
Me.bsave.BackColor = System.Drawing.Color.Navy
Me.bsave.FlatStyle = System.Windows.Forms.FlatStyle.Flat
Me.bsave.Font = New System.Drawing.Font("Poor Richard", 9.75!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.bsave.ForeColor = System.Drawing.Color.White
Me.bsave.Location = New System.Drawing.Point(24, 56)
Me.bsave.Name = "bsave"
Me.bsave.Size = New System.Drawing.Size(75, 23)
Me.bsave.TabIndex = 29
Me.bsave.Text = "&SAVE"
Me.bsave.UseVisualStyleBackColor = False
'
'bedit
'
Me.bedit.BackColor = System.Drawing.Color.Navy
Me.bedit.FlatStyle = System.Windows.Forms.FlatStyle.Flat
Me.bedit.Font = New System.Drawing.Font("Poor Richard", 9.75!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.bedit.ForeColor = System.Drawing.Color.White
Me.bedit.Location = New System.Drawing.Point(24, 114)
Me.bedit.Name = "bedit"
Me.bedit.Size = New System.Drawing.Size(75, 23)
Me.bedit.TabIndex = 30
Me.bedit.Text = "&EDIT"
Me.bedit.UseVisualStyleBackColor = False
'
'bdel
'
Me.bdel.BackColor = System.Drawing.Color.Navy
Me.bdel.FlatStyle = System.Windows.Forms.FlatStyle.Flat
Me.bdel.Font = New System.Drawing.Font("Poor Richard", 9.75!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.bdel.ForeColor = System.Drawing.Color.White
Me.bdel.Location = New System.Drawing.Point(24, 143)
Me.bdel.Name = "bdel"
Me.bdel.Size = New System.Drawing.Size(75, 23)
Me.bdel.TabIndex = 31
Me.bdel.Text = "&DELETE"
Me.bdel.UseVisualStyleBackColor = False
'
'bexit
'
Me.bexit.BackColor = System.Drawing.Color.Navy
Me.bexit.FlatStyle = System.Windows.Forms.FlatStyle.Flat
Me.bexit.Font = New System.Drawing.Font("Poor Richard", 9.75!, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.bexit.ForeColor = System.Drawing.Color.White
Me.bexit.Location = New System.Drawing.Point(24, 174)
Me.bexit.Name = "bexit"
Me.bexit.Size = New System.Drawing.Size(75, 23)
Me.bexit.TabIndex = 32
Me.bexit.Text = "E&XIT"
Me.bexit.UseVisualStyleBackColor = False
'
'cdept
'
Me.cdept.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle
Me.cdept.Font = New System.Drawing.Font("Times New Roman", 9.75!, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.cdept.Location = New System.Drawing.Point(152, 320)
Me.cdept.Name = "cdept"
Me.cdept.Size = New System.Drawing.Size(136, 22)
Me.cdept.TabIndex = 66
'
'cname
'
```

```
Me.cname.Font = New System.Drawing.Font("Times New Roman", 9.75!, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.cname.Location = New System.Drawing.Point(152, 240)
Me.cname.Name = "cname"
Me.cname.Size = New System.Drawing.Size(136, 23)
Me.cname.TabIndex = 67
'
'cdur
'
Me.cdur.Font = New System.Drawing.Font("Times New Roman", 9.75!, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.cdur.Location = New System.Drawing.Point(152, 280)
Me.cdur.Name = "cdur"
Me.cdur.Size = New System.Drawing.Size(136, 23)
Me.cdur.TabIndex = 68
'
'course
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.BackColor = System.Drawing.Color.AliceBlue
Me.ClientSize = New System.Drawing.Size(573, 414)
Me.Controls.Add(Me.cdur)
Me.Controls.Add(Me.cname)
Me.Controls.Add(Me.cdept)
Me.Controls.Add(Me.GroupBox1)
Me.Controls.Add(Me.Label7)
Me.Controls.Add(Me.ccat)
Me.Controls.Add(Me.Label6)
Me.Controls.Add(Me.lcid)
Me.Controls.Add(Me.cid)
Me.Controls.Add(Me.Label3)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label5)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.PictureBox2)
Me.Controls.Add(Me.PictureBox1)
Me.Name = "course"
Me.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen
CType(Me.PictureBox1, System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.PictureBox2, System.ComponentModel.ISupportInitialize).EndInit()
Me.GroupBox1.ResumeLayout(False)
Me.ResumeLayout(False)
Me.PerformLayout()

End Sub

#End Region

Private Sub course_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
cname.Items.Add("ME")
cname.Items.Add("BE")
cname.Items.Add("MCom")
cname.Items.Add("MSc")
cname.Items.Add("MCA")
cname.Items.Add("BCA")
cname.Items.Add("BSc")
cname.Items.Add("BCom")
cname.Items.Add("BCom CA")
ccat.Items.Add("UG")
ccat.Items.Add("PG")
cid.Text = ""
cname.Text = ""
cdur.Text = ""
cdept.Text = ""
ccat.Text = ""

cdur.Items.Add("1 Year")
cdur.Items.Add("2 Years")
cdur.Items.Add("3 Years")
cdur.Items.Add("4 Years")
cdur.Items.Add("5 Years")
Try
lcid.Items.Clear()
qry = "select cid from course"
com = New SqlCommand(qry, con)
adp = New SqlDataAdapter(com)
ds = New DataSet
adp.Fill(ds, "course")
i = ds.Tables("course").Rows.Count
Dim j As Integer
For j = 0 To i - 1
lcid.Items.Add(ds.Tables("course").Rows(j)(0))
Next
Catch ex As Exception

End Try



End Sub
```