A Project Report on

“COMPUTERIZED EXAMINATION SYSTEM”

Submitted in Partial Fulfilment of the Requirements For the award of the degree

Bachelor of Computer Applications
OF

BENGALURU CITY UNIVERSITY

SUBMITTED BY

STUDENT NAME - RAHUL BABU **Reg No - R1911677**

GAIN MORE KNOWLEDGE
REACH GREATER HIEGHT
PRESIDENCY COLLEGE

DEPARTMENT OF COMPUTER APPLICATIONS
Bachelor of Computer Applications

PRESIDENCY COLLEGE,

KEMPAPURA, HEBBAL
Bangalore-560 024

Session: OCT 2021-FEB 2022

PRESIDENCY COLLEGE

DEPARTMENT OF COMPUTER APPLICATIONS

Bachelor of Computer

Applications

(Affiliated to Bengaluru City University)

Kempapura, Hebbal, Bangaluru-

560024

GAIN MORE KNOWLEDGE
REACH GREATER HIEGHT
PRESIDENCY COLLEGE

CERTIFICATE

This is to certify that ***RAHUL BABU*** *with Register No.* ***R1911677*** *has satisfactorily completed the Fifth Semester BCA Project titled "Computerized Examination System", as a partial fulfillment of the requirements for the award of the Degree in* ***Bachelor of Computer Applications****,* ***Bengaluru City University****, during the Academic Year* ***2021– 22****.*

Project guide

Head of Department
(Department of computer Application)

Examiners

Reg No: --------------------------------------

1. --------------------------------

Examination Centre: -------------------

2. --------------------------------

Date of the exam: --------------------------

Declaration

The project titled "**COMPUTERIZED EXAMINATION SYSTEM**" was developed by me in partial fulfillment of Bengaluru City University. It is a systematic work carried by us under the guidance Mr Pachayappan R, Assistant Professor in Computer Science Department, Presidency College, Bangaluru-24.

I, declare that this same project has not been submitted to any degree or diploma to Bengaluru University or any other University.

Name of the student: - **RAHUL BABU**

Date:-

Signature

Acknowledgement

The development of software is generally a bit complex and time-consuming task. The goal of developing the project "**COMPUTERIZED EXAMINATION SYSTEM**" could not be archived without the encouragement of kindly helpful and supportive people. Here we convey our sincere thanks for all of them.

I take this opportunity to express my gratitude to people who had been instrumental in the successful completion of this project.

I am thankful to our management trustee for providing us an opportunity to work and complete the project successfully.

I wish to express my thanks to our Principal for his support to the project work. I would like to acknowledge my gratitude to our HOD of Bachelor of Computer Applications **Dr ALLI** for her encouragement and support. Without her encouragement and guidance, this project would not have materialized.

The guidance and support received from our Internal Guide **Mr Pachayappan R** who contributed to this project were vital for the success of the project. We are grateful for their constant support and help.

INDEX

Synopsis

Department of Computer Applications

BCA5 : MINI PROJECT - SYNOPSIS

TITLE OF THE PROJECT: COMPUTERIZED EXAMINATION SYSTEM

OBJECTIVE AND INTRODUCTION:

The Computerized Examination system is application software that is used by the schools and colleges in this pandemic to conduct the Examination virtually and also to enhance the skill of the students. It records the marks and performance of the student

The main reason behind designing and developing this project is to enable transact efficiently, and it also maintains the details about the students

and the performance in their academics.

This project handles the complete automation regarding the Examination. It also enables us to add records regarding new students, teachers, staff, new students, questions, and details about them.

PRESENT AND PROPOSED SYSTEM:

The existing system is limited with few modules and not easy to access the student details and details of the Examination. But our purpose system provides the proper student details and results of the Student, Adding questions and appropriately updating them.

Major Modules

INPUT:

- Adding student details,
- staff details,
- teacher’s details
- view of various modules

MAJOR MODULES:

- Student
- Teacher
- Result
- Question

Functionalities of Modules

- **Student:**

  This gives the summarized details about the student suits or Student ID, Student name, address, class, qualification. Computerized examinations are a new and exciting experience for students. There will be no lengthy answers for students to write but only multiple-choice questions. So students can't blindly memorize their syllabus but have the major role to understand each concept and should also change the method of studying. Computerized exams give the student value .extraordinary thinking, self-assessment, overcoming failures, filling them with positivity to improve the quality of education.

- **Teacher:**

  This module summarizes the details of Teachers such as adding the New questions, Adding up new students, calculating the result, and all. They track the information of students and they deal with monitoring the information and transactions of the exam. It has the role of moderator or the validator who has access to modify /edit the question bank for the specific subjects assigned to this role. A computerized examination system is an easy way for teachers to operate.

- **Result:**

  This report summarizes the details about the student marks. To increase the efficiency of managing the results, it provides filter reports on exams, questions, students. It generates the report on results, teachers, paper. A computerized examination system can be ensured fair marks when it comes to results. As soon as the student is done with exams, it generates results automatically without any delay. It also helps students to see where exactly they have gone wrong

- **Questions:**

  This allows to enter the details of new questions to the Question and also delete and update the questions. It can control the amount of input. The computerized examination system is based on the existing implementation infrastructure and also modeled along with the traditional mode of examination and the major role of design is a strategical organization of ideas, materials, and processes to achieve a goal. This model involves the ideas, processes, and deliverables. The design model gives a good test that is highly reliable.

REQUIREMENTS SPECIFICATION

SOFTWARE REQUIREMENTS:

- Client Side Technologies - VB19.0
- Database - MS SQL Server
- Connection Technology - ADODB / ADO.Net

HARDWARE REQUIREMENTS:

- Processor - Intel Core i5-5300U
- RAM - 4GB
- HDD - 500GB

Objective

Introduction to the System:

The computerized Examination System is a user-friendly, easy-to-use desktop software project where admin can add and update questions. Students attempt the exam and get more knowledge about the subject. A timer runs for the given time and within that period one has to answer all those questions, some buttons are presented at the bottom of the window like the start button, previous, next, finish. It's an easy way for students to attend exams rather than writing a lengthy answer in the booklet.

Objective:

- It provides a less expensive as well as an effective method of performing examinations.
- It provides the best facilities for the students to answer the question only with a single click.
- It lessens the student's and teachers' frustration.
- It helps the students to get their best knowledge about the subjects.
- They can take up their exams within a specified time and improve their speed.
- Results will automatically upgrade themselves.
- It is more secure and flexible
- The instant result is available

Tool description

FRONT END

VB.NET 19

Visual Basic is a tool that is used by more developers than any other tool. Visual Basic has been the choice of developers for various good reasons. So many small and big companies use Visual Basic for developing various types of applications. One of the key factors that contributed to the success of Visual Basic is its ease of use.

When Windows Programming (writing programs that run on Windows OS) was very tough and was confined only to a few people who were good in C and C++, Visual Basic hit the market and changed the way one would look at windows programming. Visual Basic made windows programming so simple, even a novice started writing one or two programs for windows in Visual Basic.

What you can do with Visual Basic 19?
Since then Visual Basic has grown considerably. The following are the areas that are supported by Visual Basic 19

Standard Windows Application

Simple to complex windows applications can be developed in Visual Basic. Visual Basic provides access to all system components such as printers and clipboards. Visual Basic allows the developer to use reusable components, such as ActiveX controls. And developers have always cheered it.
Visual Basic creates a standard. EXE file that you can distribute and deploy on any machine. VB allows the developer to access windows API, which gives all the power of windows.

Database Application

Visual Basic allows you to create a front-end portion of Client/Server applications, and application servers in three-tier client/server applications. You can access any database using ODBC and OLEDB interfaces.

ActiveX Component

Visual Basic allows you to create reusable software components based on ActiveX technology.

Internet Application

Visual Basic allows you to develop an application that can run on the Internet and Intranet. Support for Internet application has been enhanced in Visual Basic 19 by adding two new project types – DHTML application and IIS application.Next, we will understand various editions of Visual Basic.

Visual Basic Editions

Visual Basic is available in three versions, each geared to meet a specific set of development requirements. The following are the three editions and what they provide to developers. The editions are discussed in the order of features.

Learning Edition

Allows programmers to easily create powerful applications for Microsoft Windows and Windows NT. It includes all intrinsic controls, plus grid, tab, and data-bound controls.

Professional Edition

Provides a full-featured set of tools for developing solutions for others. It includes all the features of the Learning edition, plus additional ActiveX controls, the Internet Information Server Application Designer, integrated Visual Database Tools, and Data Environment, Active Data Objects, and the Dynamic HTML Page Designer.

Enterprise edition

Allows professionals to create robust distributed applications in a team setting. It includes all the features of the Professional edition, plus Back Office tools such as SQL Server, Microsoft
Transaction Server, Internet Information Server, Visual SourceSafe, SNA Server, and more.

Note: Visual Basic 19 is a part of Visual Studio 19

Starting Visual Basic IDE

Visual Basic provides IDE (Integrated Development Environment) which provides developers with all the tools they need to develop applications.

To start Visual Basic from Windows:

1. Click Start on the Taskbar.
2. Select Programs.
3. Select Microsoft Visual Studio 19 and then Microsoft Visual Basic 19

When you start Visual Basic IDE, you are prompted to select the type of project - more on this later in this chapter.

Components of IDE

Visual Basic's IDE has a collection of components. Each component has a specific task. For example, Project Explorer is used to displaying the components of the project. And properties window allows you to view & change properties.

The following are the components available in Visual Basic IDE.

Menu Bar

Displays the commands you use to work with Visual Basic. Besides the standard File, Edit, View, Window, and Help menus, menus are provided to access functions specific to programmings such as Project, Format, or Debug.

BACK-END

Microsoft SQL Server

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft SQL Server aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users. Its primary query languages are T-SQL and ANSI SQL.

Enterprise

SQL Server Enterprise Edition includes both the core database engine and add-on services, with a range of tools for creating and managing a SQL Server cluster. It can manage databases as large as 524 petabytes and address 12 terabytes of memory and supports 640 logical processors (CPU cores).

Standard

SQL Server Standard edition includes the core database engine, along with the stand-alone services. It differs from Enterprise edition in that it supports fewer active instances (number of nodes in a cluster) and does not include some high-availability functions such as hot-add memory (allowing memory to be added while the server is still running), and parallel indexes.

Web

SQL Server Web Edition is a low-TCO option for Web hosting.

Business Intelligence

Introduced in SQL Server 2012 and focusing on Self Service and Corporate Business Intelligence. It includes the Standard Edition capabilities and Business Intelligence tools: PowerPivot, Power View, the BI Semantic Model, Master Data Services, Data Quality Services, and x Velocity in-memory analytics.

SQL Server is a relational database management system, or RDBMS, developed and marketed by Microsoft. Similar to other RDBMS software, SQL Server is built on top of SQL, a standard programming language for interacting with relational databases. SQL Server is tied to Transact-SQL, or T-SQL, Microsoft's implementation of SQL that adds a set of proprietary programming constructs.

SQL Server works exclusively on the Windows environment for more than 20 years. In 2016, Microsoft made it available on Linux. SQL Server 2017 became generally available in October 2016 that ran on both Windows and Linux.

MS SQL Server for OS/2 began as a project to port Sybase SQL Server onto OS/2 in 1989, by Sybase, Ashton-Tate, and Microsoft.

SQL Server 4.2 for NT is released in 1993, marking the entry onto Windows NT.

SQL Server 6.0 is released in 1995, marking the end of the collaboration with Sybase; Sybase would continue developing their variant of SQL Server, Sybase Adaptive Server Enterprise, independently of Microsoft.

SQL Server 7.0 is released in 1998, marking the conversion of the source code from C to C++.

SQL Server 2005, released in 2005, finishes the complete revision of the old Sybase code into Microsoft code.

SQL Server 2012, released in 2012, adds columnar in-memory storage aka xVelocity. SQL Server 2017, released in 2017, adds Linux support for these Linux platforms: Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Ubuntu & Docker Engine.

System analysis

Feasibility Study

A feasibility study is a measure of how beneficial or practical the development of an information system will be to an organization. The Feasibility analysis is across life cycle activity and should be continuously performed throughout the system life cycle. A feasibility study lets the developer foresee the future of the project and its usefulness. The study on feasibility is done based on a few factors. They are:

Operational feasibility:

The computerized Exam Management System is a user-friendly, easy-to-use desktop software project where admin can add and update questions. Students attempt the exam and can check their course knowledge. A timer runs for the given time and within that period one has to answer all those questions, some buttons are presented at the bottom of the window like the start button, previous, next, finish. Considering all these factors we can conclude that all the users and end-users will be satisfied by the system.

Technical feasibility:

The system must be evaluated from a technical point of view. The assessment of this feasibility must be based on an outline design of the system requirement in terms of input, output, programs, and procedures. For the design and development of the system, several software products have been accommodated.
Database design – MY SQLServer This software has enough efficiency in producing the system Therefore the project is technically feasible

Schedule feasibility:

The duration of time required for the project has been planned appropriately and it is the same as the duration of time expected by the customer. Therefore the product can be delivered to the customer within the expected time duration, satisfying the customer. Hence the project is feasible in scheduling.

Economic feasibility:

According to the resources available and the project scheduling process it is estimated that the expenses allocated for the software to be developed, by the customer are sufficient enough. Hence the economical factor has been considered feasible.

Behavioral Feasibility:

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

This project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

SYSTEM DESIGN

E-R diagram

Introduction

An Entity-Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects, or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education, and research. Also known as

ERDs or ER Models, use a defined set of symbols such as rectangles, diamonds, ovals, and a Diagram is a type of flowchart that illustrates how “entities” such as people, objects, or concepts relate to each other within a system.

ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education, and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals, and connecting lines to depict the interconnectedness of entities, relationships, and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

ERD example

ER diagrams are related to data structure diagrams (DSDs), which focus on the relationships of elements within entities instead of relationships between entities themselves. ER diagrams also are often used in conjunction with data flow diagrams (DFDs), which map out the flow of information for processes or systems.

History of ER models

Peter Chen (a.k.a. Peter Pin-Shan Chen), currently a faculty member at Carnegie-Mellon University in Pittsburgh is credited with developing ER modeling for database design in the 1970s. While serving as an assistant professor at MIT’s Sloan School of Management, he published a seminal paper in 1976 titled “The Entity-Relationship Model: Toward a Unified View of Data.”

In a broader sense, the depiction of the interconnectedness of things dates back to least ancient Greece, with the works of Aristotle, Socrates, and Plato. It’s seen more recently in the 19th and 20th Century works of philosopher-logicians like Charles Sanders Peirce and Got Frege.

By the 1960s and 1970s, Charles Bachman (above) and A.P.G. Brown were working with close predecessors of Chen’s approach. Bachman developed a type of Data Structure Diagram, named after him as the Bachman Diagram. Brown published works on real-world systems modeling. James Martin added ERD refinements.

The work of Chen, Bachman,
Brown, Martin, and others also contributed to the development of Unified Modeling Language (UML), widely used in software design.

Uses of entity-relationship diagrams
Database design:
ER diagrams are used to model and design relational databases, in terms of logic and business rules (in a logical data model) and in terms of the specific technology to be implemented (in a physical data model.) In software engineering, an ER diagram is often an initial step in determining requirements for an information systems project. It's also later used to model a particular database or databases. A relational database has an equivalent relational table and can potentially be expressed that way as needed.

Database troubleshooting:

ER diagrams are used to analyze existing databases to find and resolve problems of entity-relationship diagrams

Database design:

ER diagrams are used to model and design relational databases, in terms of logic and business rules (in a logical data model) and in terms of the specific technology to be implemented (in a physical data model.) In software engineering, an ER diagram is often an initial step in determining requirements for an information systems project. It's also later used to model a particular database or databases. A relational database has an equivalent relational table and can potentially be expressed that way as needed.

Database troubleshooting:

ER diagrams are used to analyze existing databases to find and resolve problems in logic or deployment. Drawing the diagram should reveal where it's going wrong.

Business information systems:

The diagrams are used to design or analyze relational databases used in business processes.
Any business process that uses fielded data involving entities, actions, and interplay can potentially benefit from a relational database. It can streamline processes, uncover information more easily and improve results.

Business process re-engineering (BPR):

ER diagrams help in analyzing databases used in business process re-engineering and in modeling a new database setup.

Education:

Databases are today's method of storing relational information for educational purposes and later retrieval, so ER Diagrams can be valuable in planning those data structures.

Research:

Since so much research focuses on structured data, ER diagrams can play a key role in setting up useful databases to analyze the data.

The components and features of an ER diagram

ER Diagrams are composed of entities, relationships, and attributes. They also depict cardinality, which defines relationships in terms of numbers. Here's a glossary:

Entity

A definable thing—such as a person, object, concept, or event—can have data stored about it.

Think of entities as nouns. Examples: a customer, student, car, or product attributes.

They also depict cardinality, which defines relationships in terms of numbers. Here's a glossary:

Entity type:

A group of definable things, such as students or athletes, whereas the entity would be the specific student or athlete. Other examples: customers, cars, or products.

Entity set:

Same as an entity type, but defined at a particular point in time, such as students enrolled in a class on the first day. Other examples: Customers who purchased last month, cars currently registered in Florida. A related term is an instance, in which the other or are associated with each other. Think of relationships as verbs.

For example, the named student might register for a course. The two entities would be the student and the course, and the relationship depicted is the act of enrolling, connecting the two entities in that way.

Relationships are typically shown as diamonds or labels directly on the connecting lines.

Recursive relationship:

The same entity participates more than once in the relationship.

Attribute

A property or characteristic of an entity. Often shown as an oval or circle.

Descriptive attribute:

A property or characteristic of a relationship (versus of an entity.)

Attribute categories:

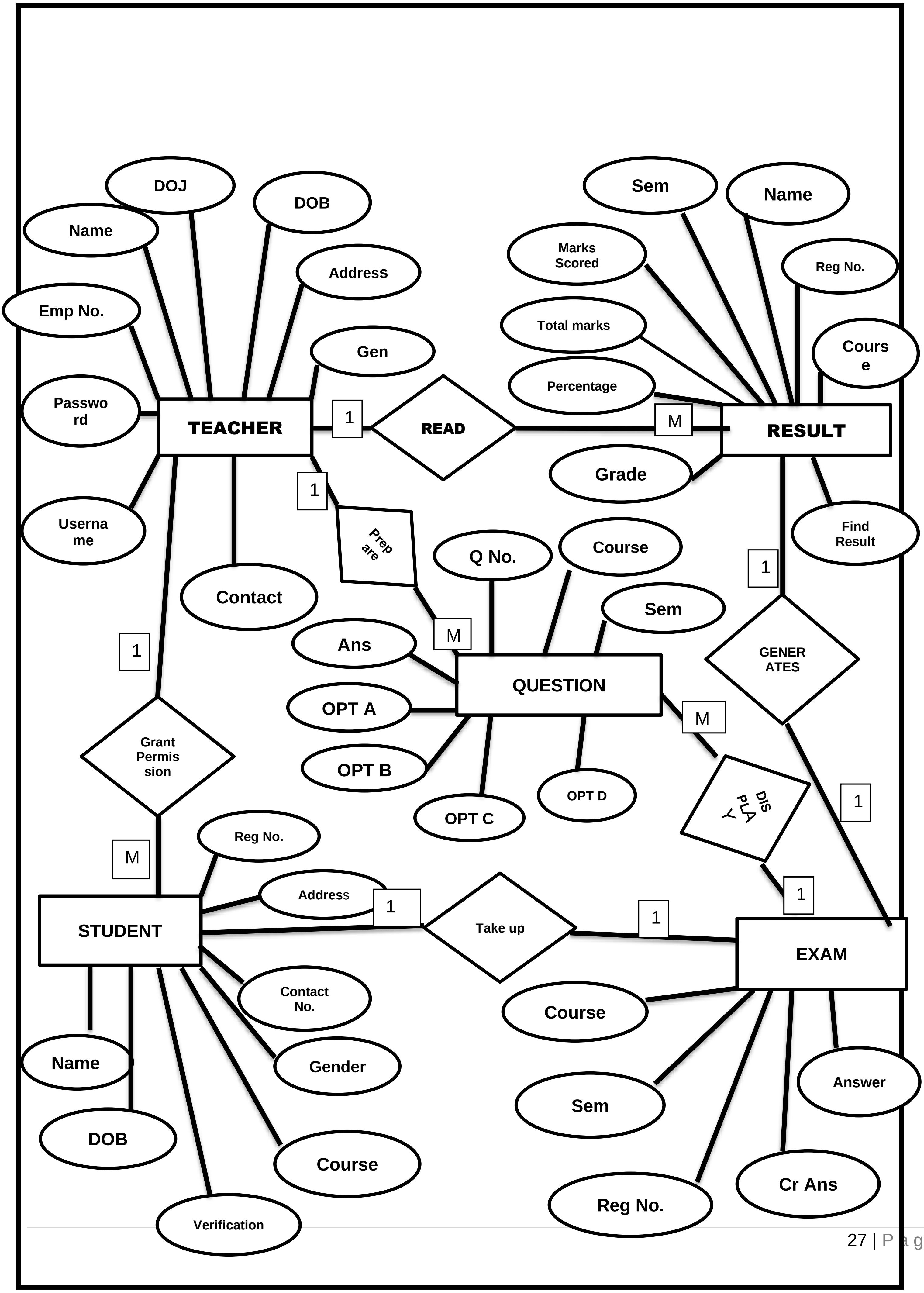
Attributes are categorized as simple, composite, derived, as well as single-value or multi-valued

Cardinality views:

Cardinality can be shown as look-across or same-side, depending on where the symbols are shown.

Cardinality constraints:

The minimum or maximum numbers apply to a relationship.


DOJ
DOB
Name
Address
Emp No.
Gen
Passwo rd
TEACHER
1
READ
M
RESULT
Sem
Name
Marks Scored
Reg No.
Total marks
Cours e
Percentage
Grade
Find Result
Userna me
1
Prep are
Q No.
Course
Contact
Sem
M
Ans
1
QUESTION
GENER ATES
1
OPT A
M
Grant Permis sion
OPT B
OPT D
OPT C
DIS PLA Y
1
M
Reg No.
1
Address
1
Take up
1
STUDENT
EXAM
Course
Contact No.
Name
Gender
Sem
Answer
DOB
Course
Cr Ans
Reg No.
Verification

Data flow diagram

DATA FLOW DIAGRAM

Data Flow diagram:

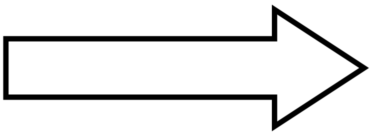
A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured design).
On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process.
A DFD provides no information about the timing or ordering of processes, or about whether processes
will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD)
The idea behind the explosion of a process into more process is that understanding at one level of details is exploded into greater detail at the next level. This is done until further explosion is necessary and an adequate amount of detail is described for the analyst to understand the process.
Larry Constantine first developed the DFD as a way of expressing system requirements in a graphical form, this led to the modular design.
A DFD is known as a “bubble chart” has the purpose of clarifying system requirements and identifying major transformations that will become a program in system design. So it is the starting point of the design to the lowest level of details. A DFD consists of a series of bubbles joined by data flows in the system.

DFD Symbols

In DFD, there are four Symbols

1. A square defines a source or destination system data
2. An arrow identified data flow. It is the pipeline through which the information flow
3. A circle or a bubble represents a process that transforms
4. Incoming data flow into outgoing data flows
5. An open rectangle is a data source, data at rest, or a temporary data

i.DATAFLOW :

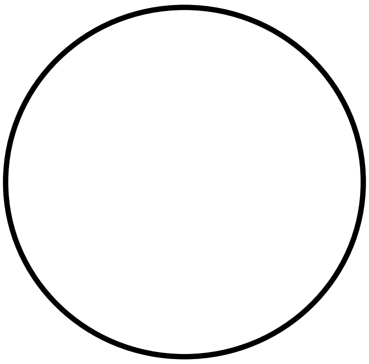


Arrows showing direction of flow

It’s a way of representing a flow of data through a process or a system.

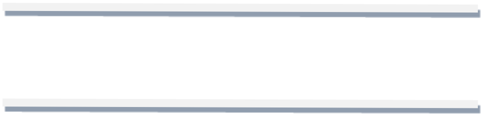
ii. PROCESS :

Circles



Data flow diagrams represent systems by use of a spare number of symbols. Systems processes are symbolized by circles , entities, external to the system which interact with the system .

i.FILE :



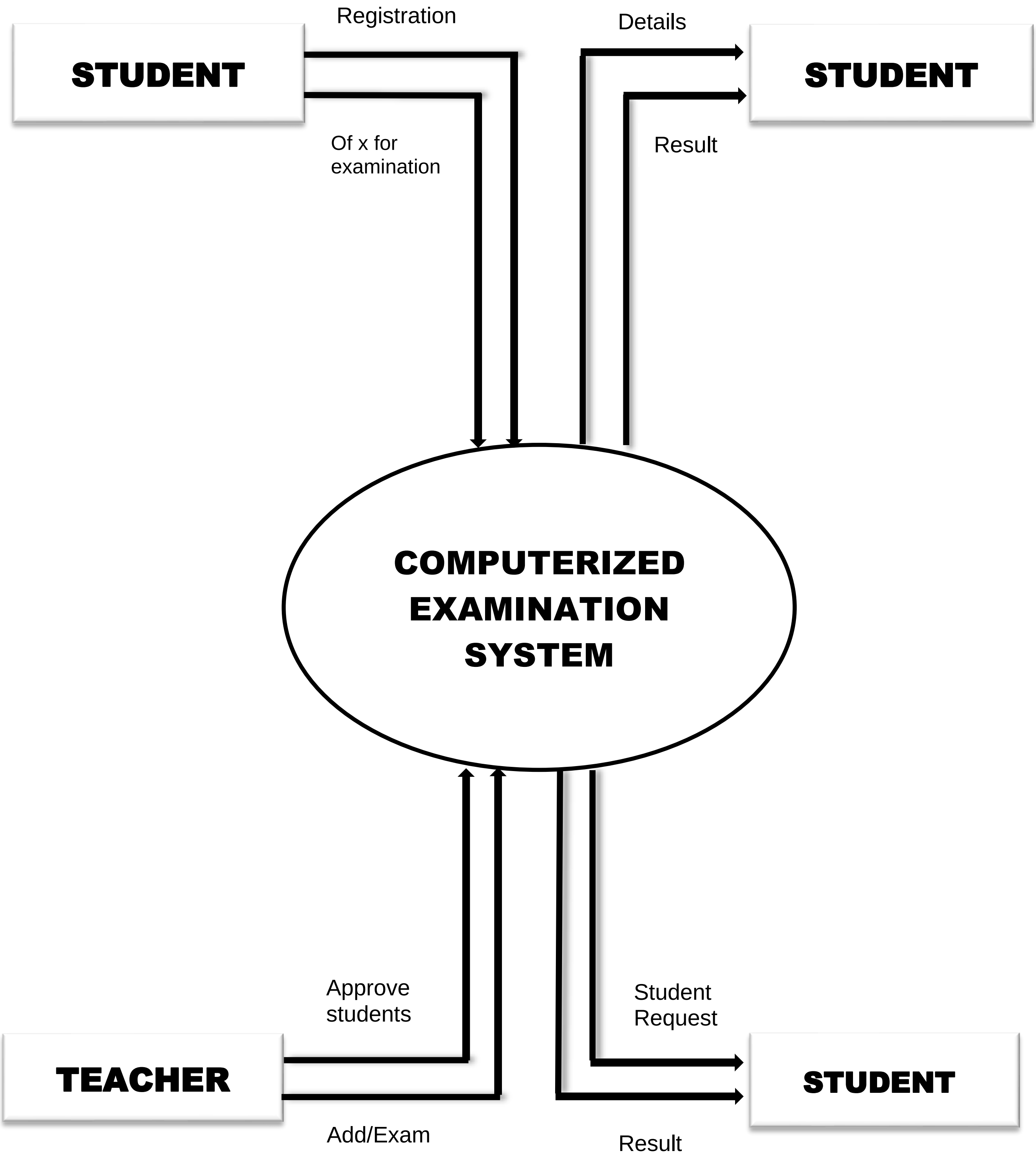
Horizontal pair of lines

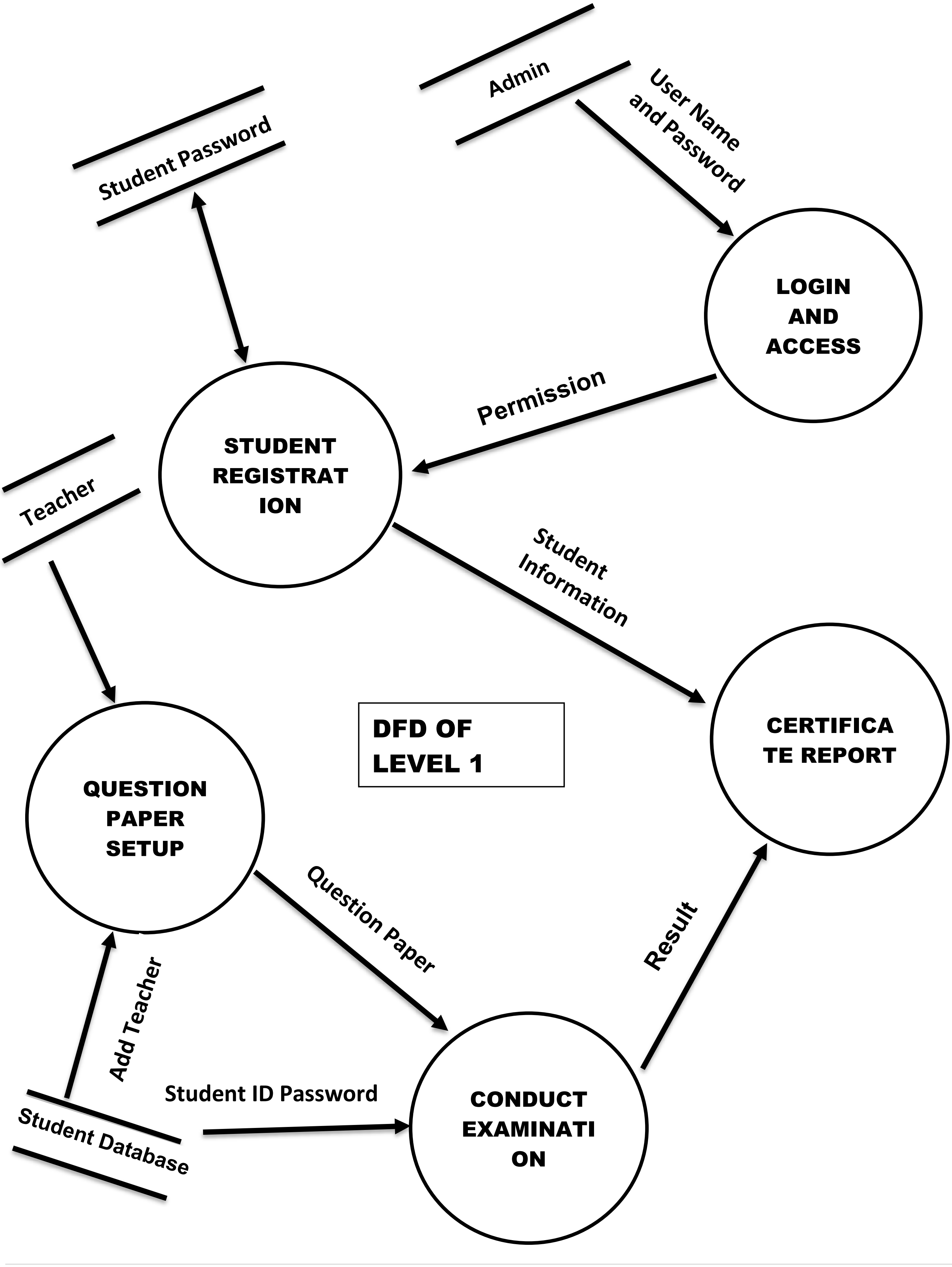
Data flow diagram maps out the flow of information for any process or the system.

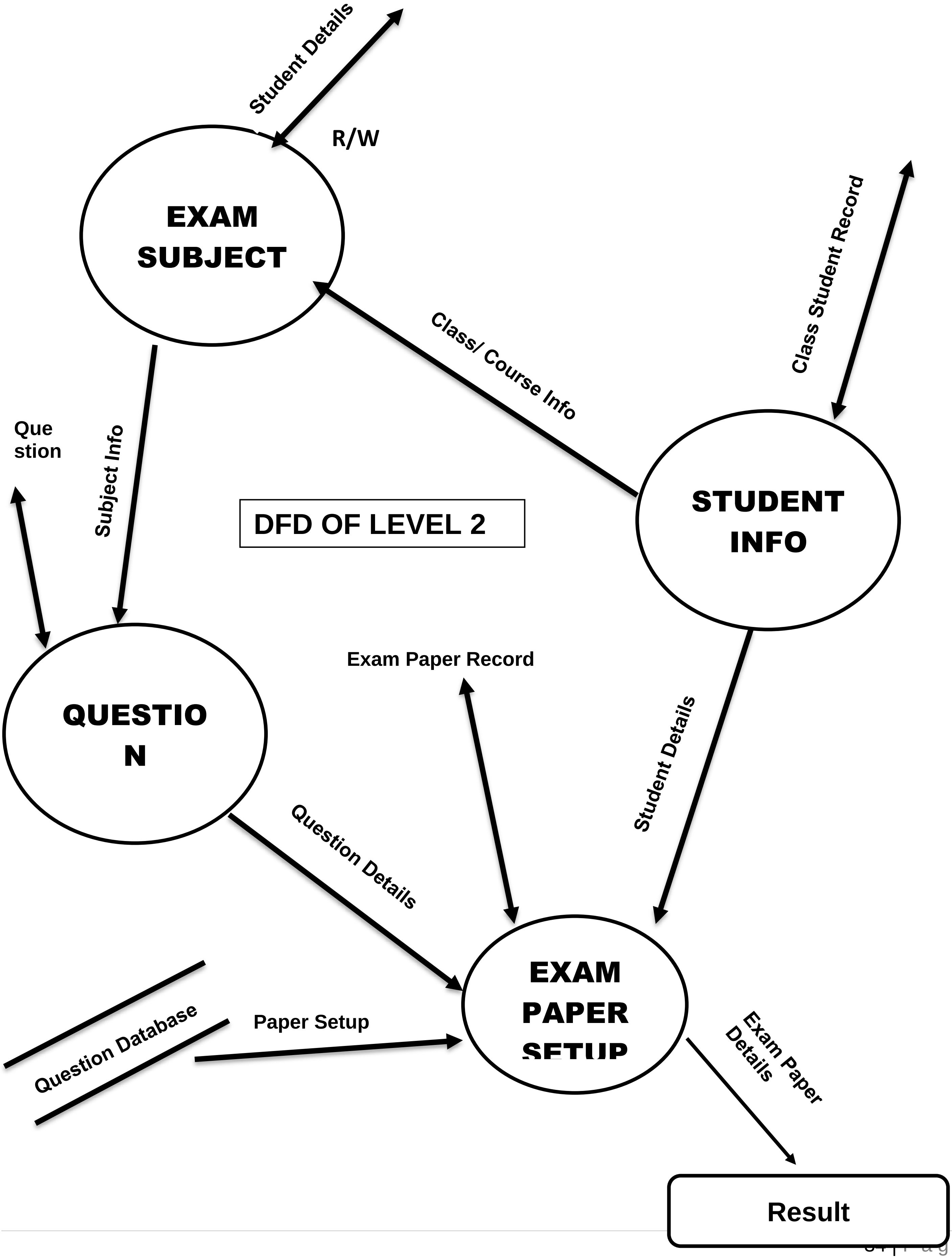
ii. DATA – SOURCE, SINK :

A data source connection specifies the parameters needed to connect to a database ,such as the location of the database and the timeout duration. These parameters form a connection string for the data source.

DFD OF LEVEL 0




Admin
User Name and Password
Student Password
LOGIN AND ACCESS
Permission
STUDENT REGISTRATION
Teacher
Student Information
DFD OF LEVEL 1
CERTIFICATE REPORT
QUESTION PAPER SETUP
Question Paper
Result
Add Teacher
Student ID Password
Student Database
CONDUCT EXAMINATION


DFD OF LEVEL 2

Schema

distrib

Database Schema

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It’s the database designers who design the schema to help programmers understand the database and make it useful.

A database schema can be divided broadly into two categories −

Physical Database Schema − This schema pertains to the actual storage of data and its form of storage like files, indices, etc. It defines how the data will be stored in a secondary storage.

Logical Database Schema − This schema defines all the logical constraints that need to be applied on the data stored. It defines tables, views, and integrity constraints.

Database Instance

It is important that we distinguish these two terms individually. Database schema is the skeleton of database. It is designed when the database doesn't exist at all. Once the database is operational, it is very difficult to make any changes to it. A database schema does not contain any data or information.

A database instance is a state of operational database with data at any given time. It contains a snapshot of the database. Database instances tend to change with time. A DBMS ensures that its every instance (state) is in a valid state, by diligently following all the validations, constraints, and conditions that the database designers have imposed.

A database schema is the logical representation of a database, which shows how the data is stored logically in the entire database. It contains list of attributes and instruction that informs the database engine that how the data is organized and how the elements are related to each other.

A database schema contains schema objects that may include tables, fields, packages, views, relationships, primary key, foreign key,

In actual, the data is physically stored in files that may be in unstructured form, but

to retrieve it and use it, we need to put it in a structured form. To do this, a database schema is used. It provides knowledge about how the data is organized in a database
and how it is associated with other data.

The schema does not physically contain the data itself; instead, it gives information about the shape of data and how it can be related to other tables or models.
A database schema object includes the following:
Consistent formatting for all data entries.
Database objects and unique keys for all data entries.
Tables with multiple columns, and each column contains its name and datatype.

The complexity & the size of the schema vary as per the size of the project. It helps developers to easily manage and structure the database before coding it.

The given diagram is an example of a database schema. It contains three tables, their data types. This also represents the relationships between the tables and primary keys as well as foreign keys.
The database schema is divided into three types, which are:

- Logical Schema
- Physical Schema
- View Schema
- Database Schema

1. Physical Database Schema

A physical database schema specifies how the data is stored physically on a storage system or disk storage in the form of Files and Indices. Designing a database at the physical level is called a physical schema.

2.Logical Database Schema

The Logical database schema specifies all the logical constraints that need to be applied to the stored data. It defines the views, integrity constraints, and table. Here the term integrity constraints define the set of rules that are used by DBMS (Database Management System) to maintain the quality for insertion & update the data. The logical schema represents how the data is stored in the form of tables and how the attributes of a table are linked together.

At this level, programmers and administrators work, and the implementation of the data structure is hidden at this level.

Various tools are used to create a logical database schema, and these tools demonstrate the relationships between the component of your data; this process is called ER modelling.

The ER modelling stands for entity-relationship modelling, which specifies the relationships between different entities

We can understand it with an example of a basic commerce application. Below is the schema diagram, the simple ER model representing the logical flow of transaction in a commerce application.

3.Database Schema

In the given example, the Ids are given in each circle, and these Ids are primary key & foreign keys.

The primary key is used to uniquely identify the entry in a document or record. The Ids of the upper three circles are the primary keys.

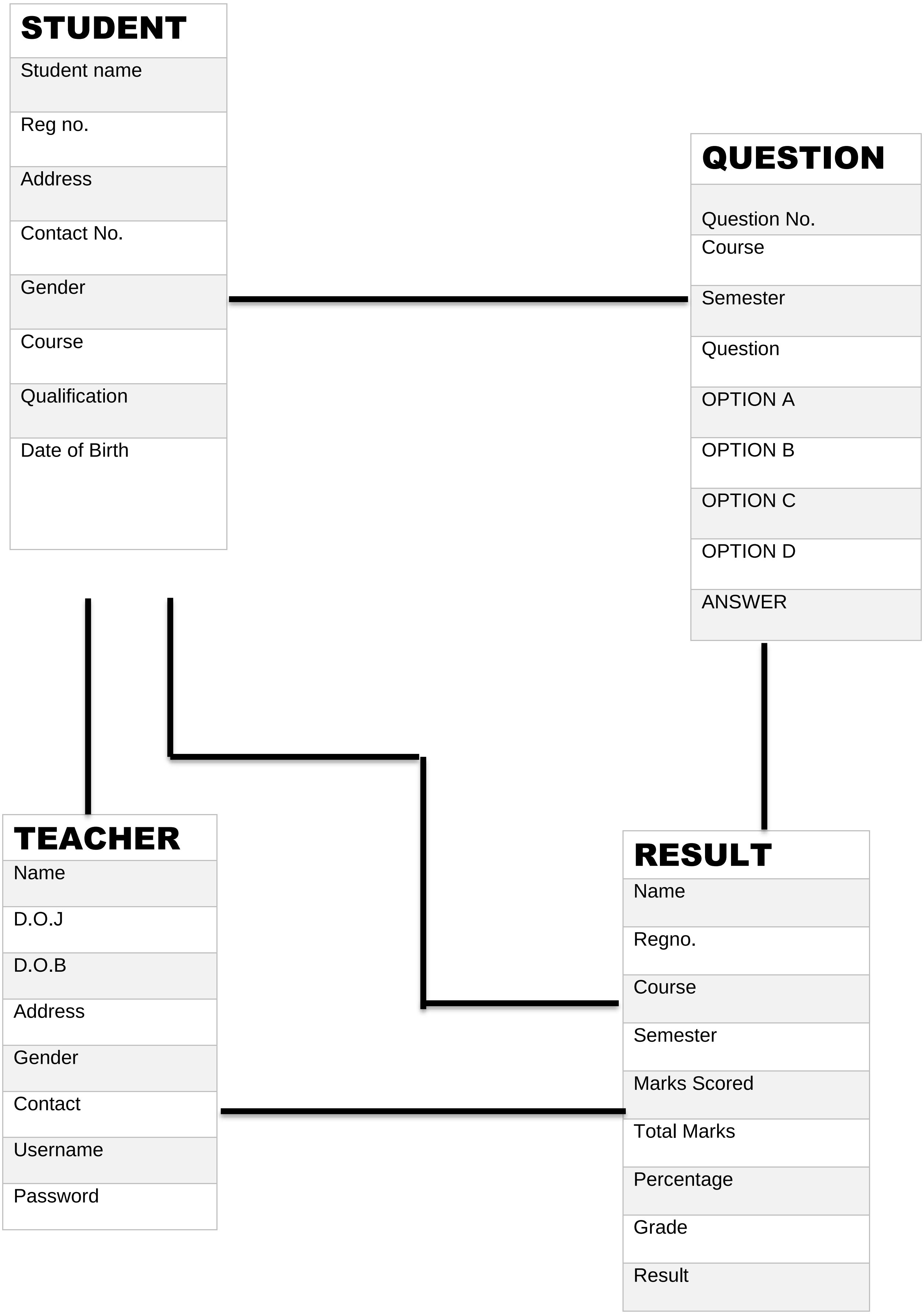
The Foreign key is used as the primary key for other tables. The FK represent the foreign key in the diagram. It relates one table to another table.

4.View Schema

The view level design of a database is known as view schema. This schema generally describes the end-user interaction with the database systems.The terms database schema and database instances are related to each other & sometimes confusing to be used as the same thing. But both are different from each other. Database Schema is a representation of a planned database and does not actually containthe data. ensures that every database instance complies with the constraints imposed by the database designers in the database schema.

The database schema is its structure described in a formal language supported by the database management system (DBMS). The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database.[citation needed]

These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realizationof the database language.[1] The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modeled in the database.



STUDENT

Student name

Reg no.

Address

Contact No.

Gender

Course

Qualification

Date of Birth

QUESTION

Question No.

Course

Semester

Question

OPTION A

OPTION B

OPTION C

OPTION D

ANSWER

TEACHER

Name

D.O.J

D.O.B

Address

Gender

Contact

Username

Password

RESULT

Name

Regno.

Course

Semester

Marks Scored

Total Marks

Percentage

Grade

Result

Testing

TESTING

The project is been tested during the execution of the programs and modules. Validation and verification is completely tested during different stages of the project in order to determine whether the project is executing as expected. This also includes different types of testing:

Unit testing:

This testing includes the complete flow of execution of modules and objects used in the project. Each unit of project is been tested independently

System testing:

This is an integrated form of testing, which focuses on functionality and interface between units and team in a controlled environment does it.

Back end testing:

The project is evaluated and tested whether the data is been stored in the database and also the respective output is been displayed. The database used in the project is Oracle 19c which have been evaluated which involves connectivity and data stored in the project.

Graphical user interface (GUI) testing:

The objective of GUI testing is to validate the GUI as per the user requirement. It has been validated under different stages of GUI testing.

Module Testing:This is an optional form of testing, which is done only for large system, which has a large number of modules.

Performance testing:

Performance Testing is done to check whether the system meets the performance requirements. Different performance and load tools have been used to check the performance testing in the project.

Implementation:

The proposed project is validated by implementing in other computer system with the change and revised design of the project. The project can be implemented on computer system but not in modified application. The other aspects of post implementation of software and maintenance.

Maintenance:

-After the project is implemented in other system, maintenance is done by checking the following conditions:

Adaptive maintenance:

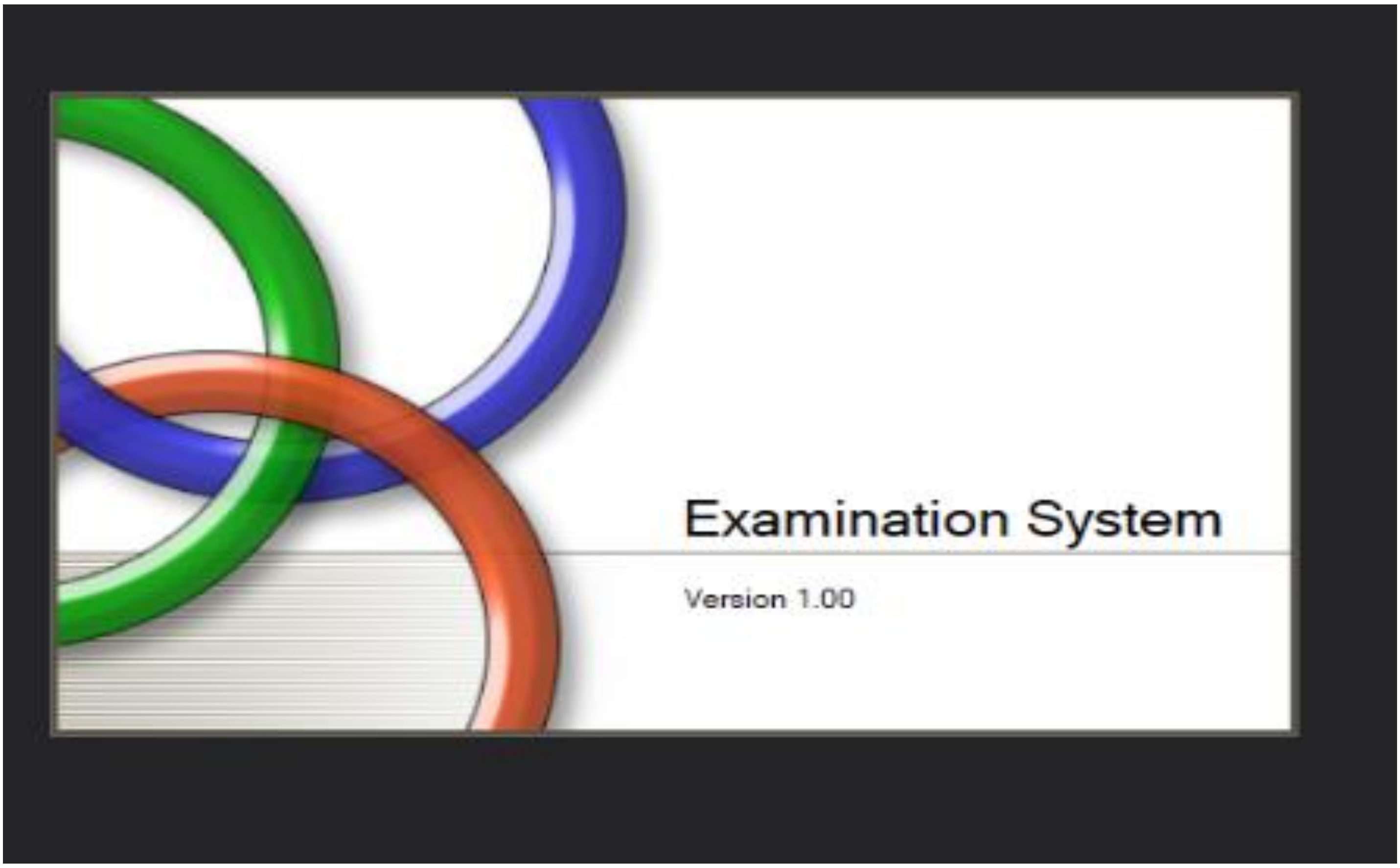
This includes modifications and validation when the users need the product to run on new platforms, on new operating systems, or when they need the product to interface with new hardware and software.

Perfective maintenance:A software product needs maintenance to support the new features that the users want or to change different types of functionalities of the system according to the user demands.

Preventive maintenance: This type of maintenance includes modifications and updation to prevent future problems of the software. It goals to attend problems, which are not significant at this moment but may cause serious issues in future.

Snapshots

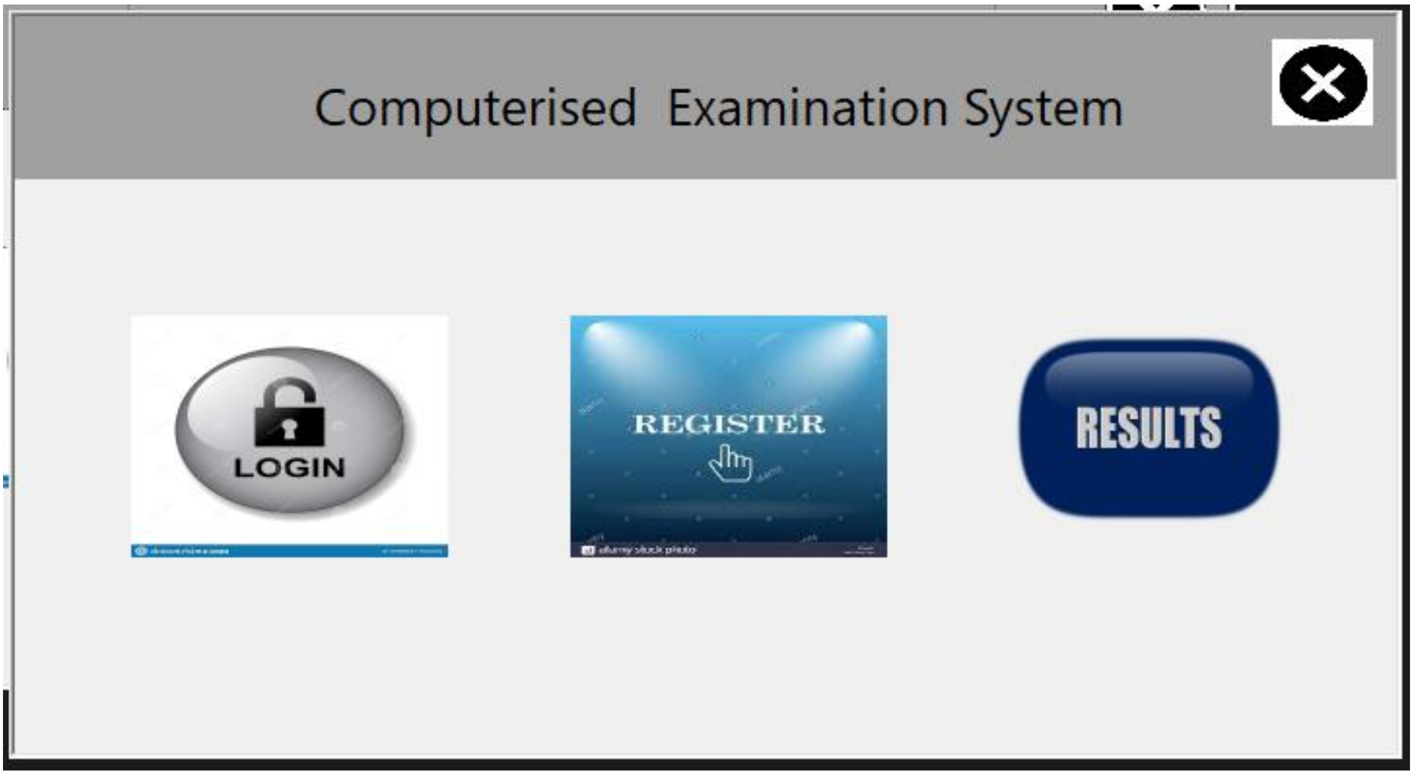
SPLASHSCREEN FRAME



Examination System

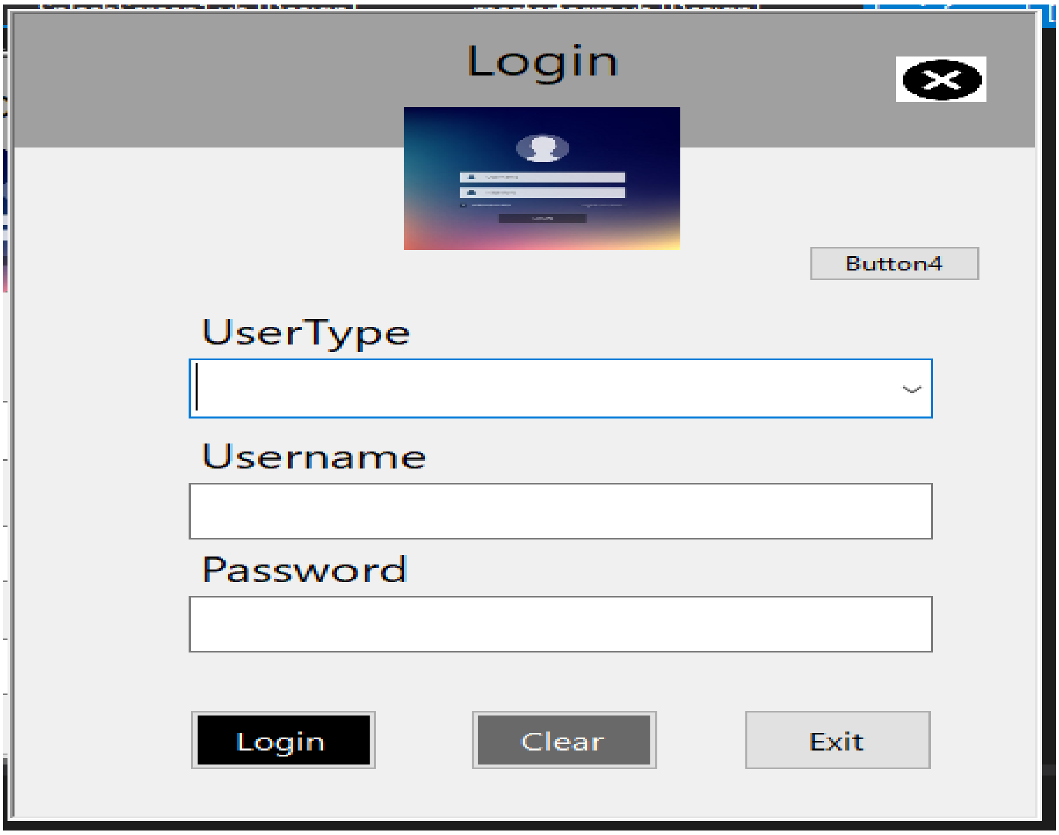
Version 1.00

STARTING PAGE



Computerised Examination System

Login page



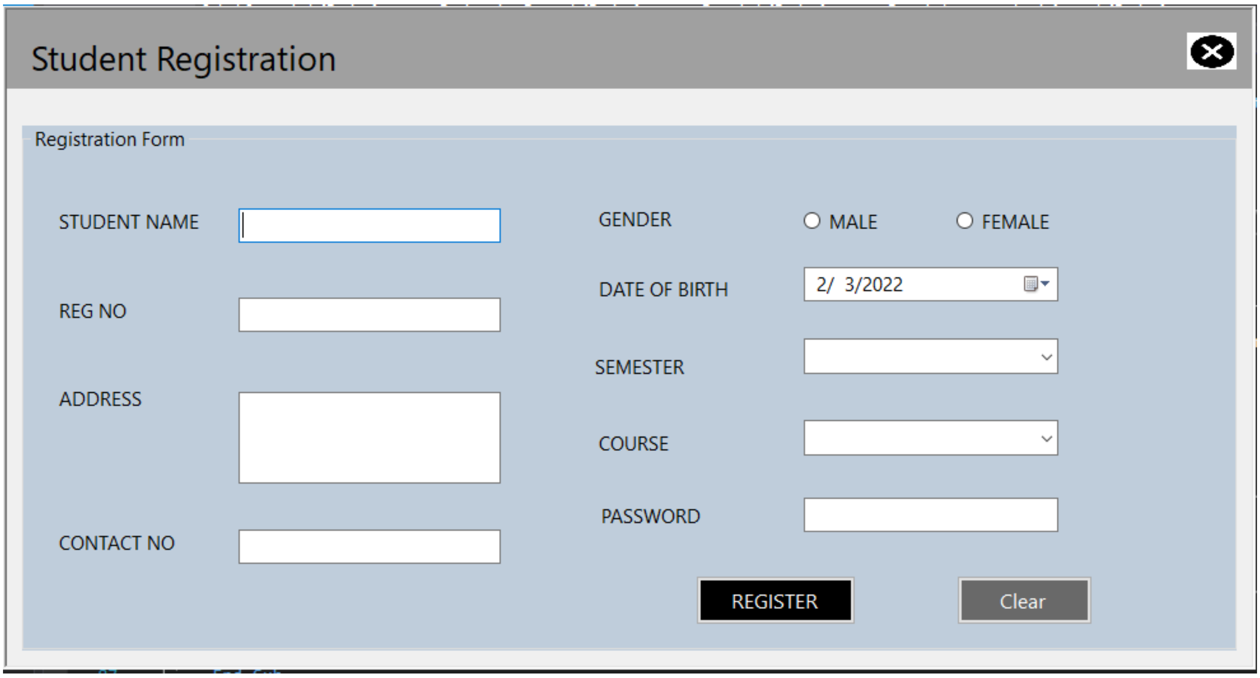
Login

UserType

Username

Password

Registration From For Student



Student Registration

Registration Form

STUDENT NAME

REG NO

ADDRESS

CONTACT NO

GENDER ○ MALE ○ FEMALE

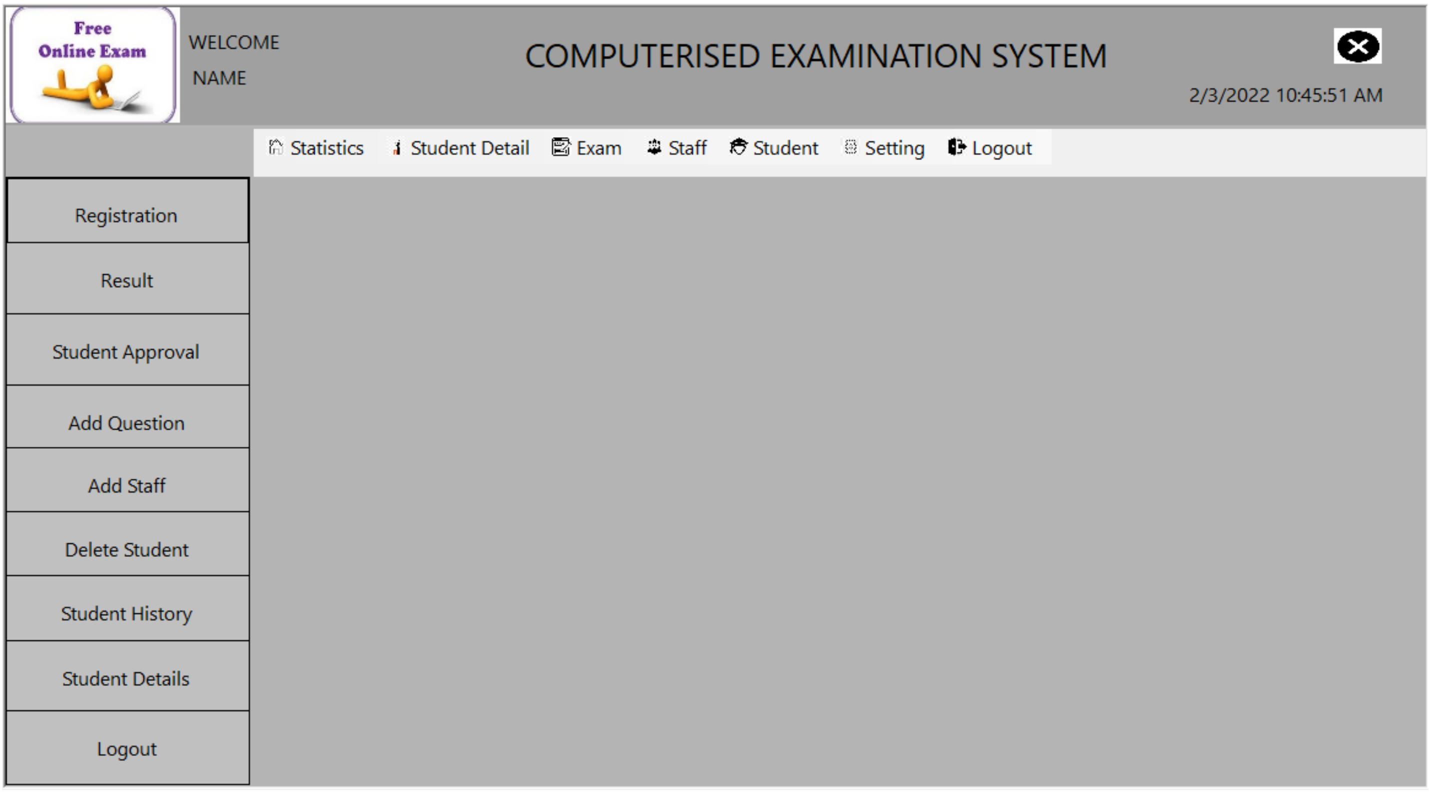
DATE OF BIRTH 2/ 3/2022

SEMESTER

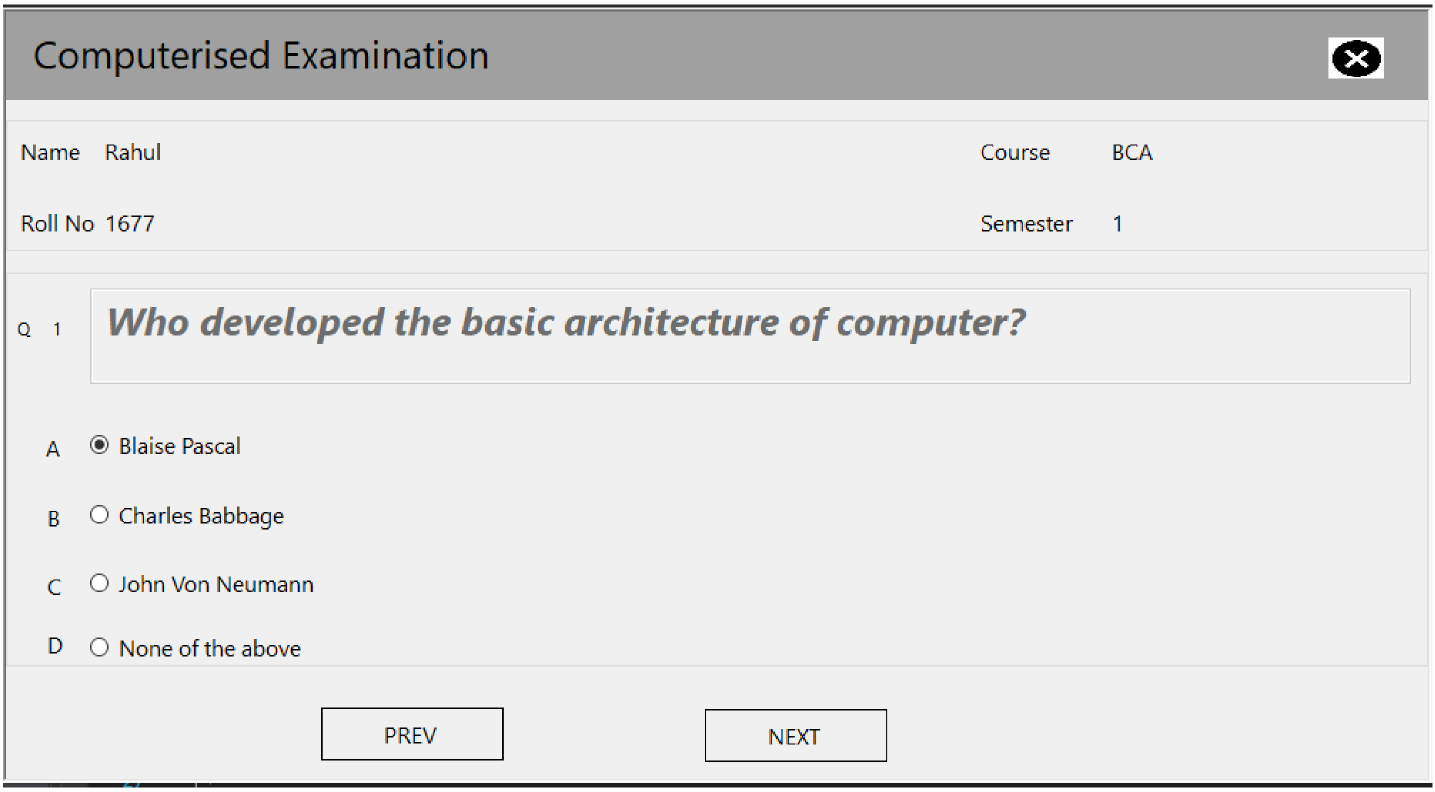
COURSE

PASSWORD

Dashboard For Teacher



Student Question Dispalay


Computerised Examination

Name Rahul

Course BCA

Roll No 1677

Semester 1

Q 1 *Who developed the basic architecture of computer?*

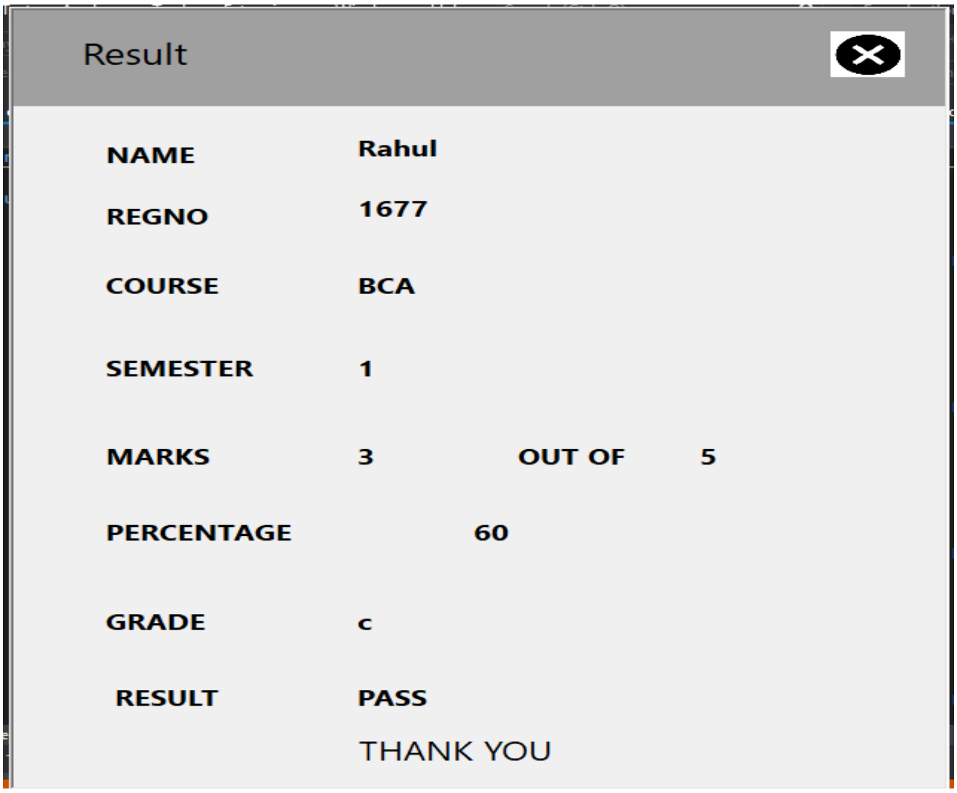
A ● Blaise Pascal

B ○ Charles Babbage

C ○ John Von Neumann

D ○ None of the above

Display Of Result

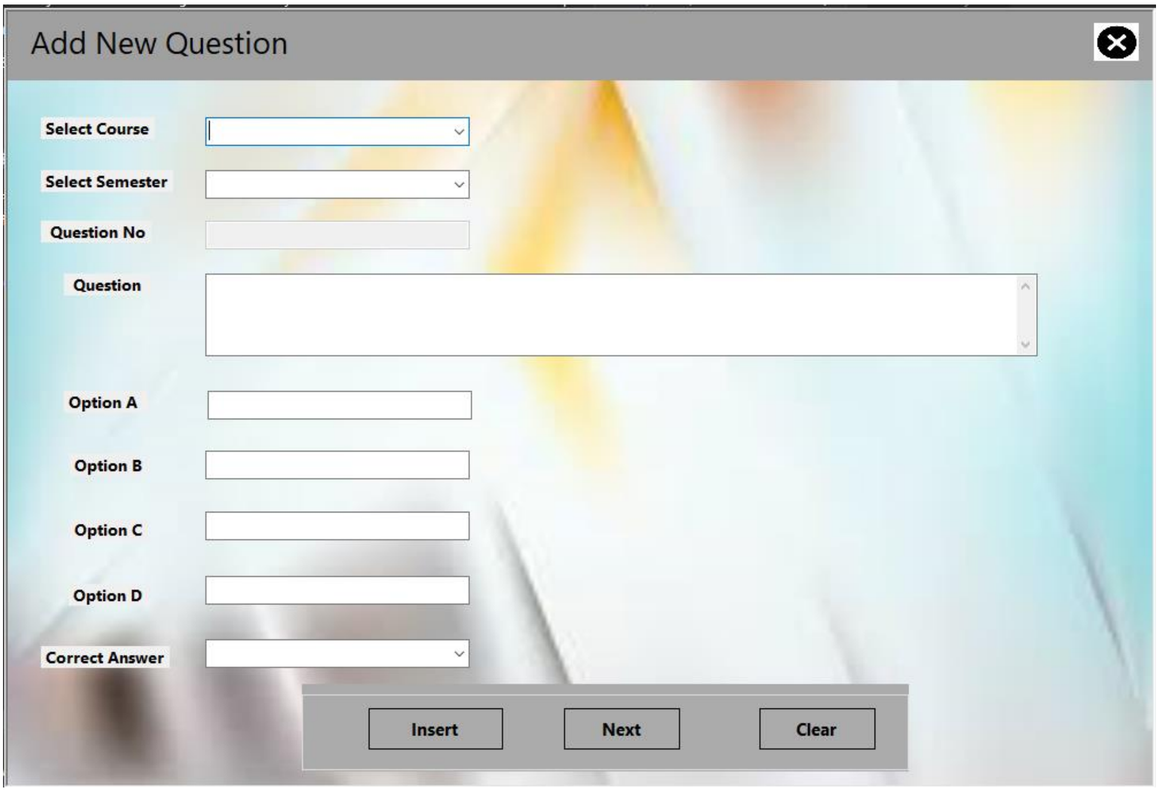


Result

NAME	Rahul		
REGNO	1677		
COURSE	BCA		
SEMESTER	1		
MARKS	3	OUT OF	5
PERCENTAGE	60		
GRADE	c		
RESULT	PASS		

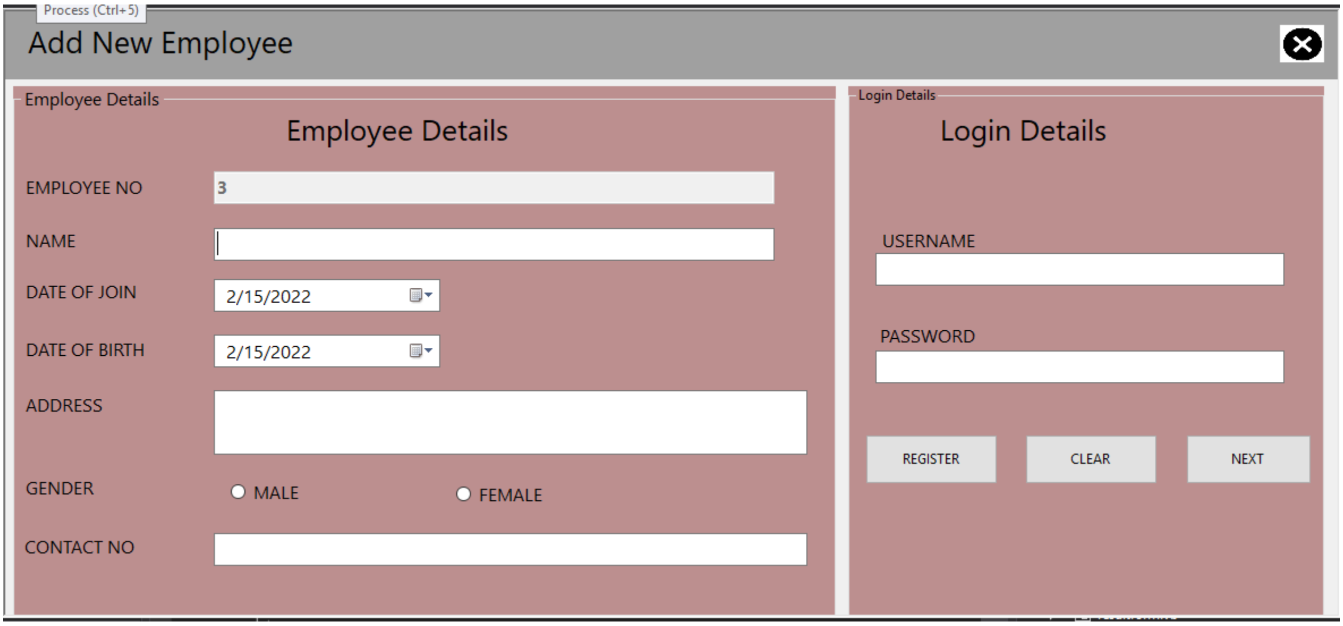
THANK YOU

Adding New Question



Add New Question

Adding New Employee



Add New Employee

Employee Details

Employee Details

EMPLOYEE NO 3

NAME

DATE OF JOIN 2/15/2022

DATE OF BIRTH 2/15/2022

ADDRESS

GENDER ○ MALE ○ FEMALE

CONTACT NO

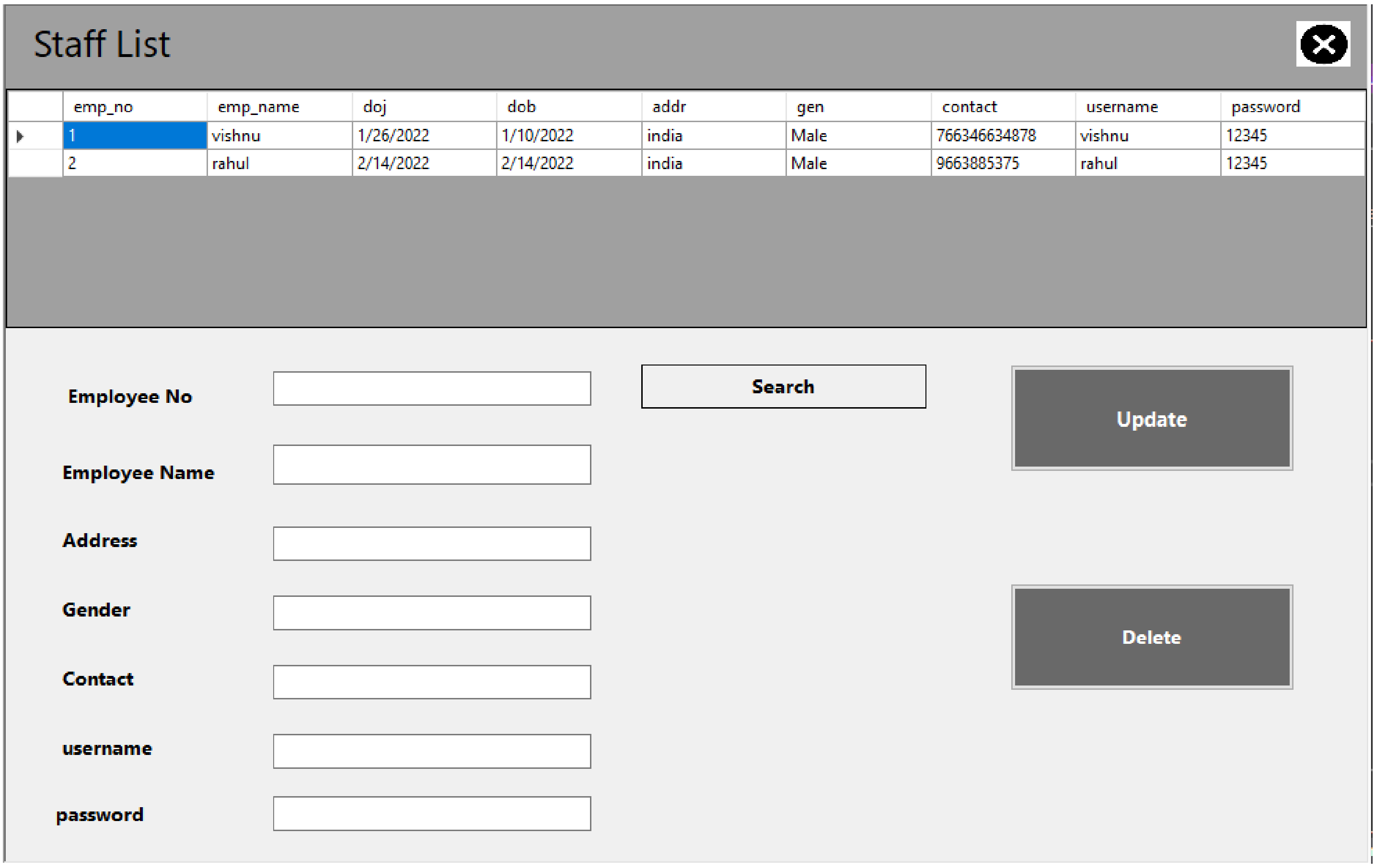
Login Details

Login Details

USERNAME

PASSWORD

View Staff list and Updation



Staff List

	emp_no	emp_name	doj	dob	addr	gen	contact	username	password
▸	1	vishnu	1/26/2022	1/10/2022	india	Male	766346634878	vishnu	12345
	2	rahul	2/14/2022	2/14/2022	india	Male	9663885375	rahul	12345

Employee No

Employee Name

Address

Gender

Contact

username

password

Source Code’

PLASH SCREEN FRAME CODING

```
Public NotInheritable Class SplashScreen1

    'TODO: This form can easily be set as the splash screen for the application
by going to the "Application" tab
    '  of the Project Designer ("Properties" under the "Project" menu).


    Private Sub SplashScreen1_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
        'Set up the dialog text at runtime according to the application's
assembly information.

        'TODO: Customize the application's assembly information in the
"Application" pane of the project
        '  properties dialog (under the "Project" menu).

        'Application title
        If My.Application.Info.Title <> "" Then
            ApplicationTitle.Text = My.Application.Info.Title
        Else
            'If the application title is missing, use the application name,
without the extension
            ApplicationTitle.Text =
System.IO.Path.GetFileNameWithoutExtension(My.Application.Info.AssemblyName)
        End If

        Version.Text = System.String.Format(Version.Text,
My.Application.Info.Version.Major, My.Application.Info.Version.Minor)

        'Copyright info
        Copyright.Text = My.Application.Info.Copyright
```

Source Code of First Page

```
Public Class Form1

    Private Sub PictureBox5_Click(sender As Object, e As EventArgs) Handles
PictureBox5.Click

        Me.Close()

    End Sub

    Private Sub PictureBox2_Click(sender As Object, e As EventArgs) Handles
PictureBox2.Click
        Loginform.Show()
        Me.Hide()


    End Sub

    Private Sub PictureBox3_Click(sender As Object, e As EventArgs) Handles
PictureBox3.Click
        Registration_Form.Show()
        Me.Hide()

    End Sub

    Private Sub PictureBox4_Click(sender As Object, e As EventArgs) Handles
PictureBox4.Click
        result2form.Show()
        Me.Hide()
    End Sub
End Class
```

Source Code of login page

```
Imports System.Data.SqlClient

Public Class Loginform
    Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")
    Dim sqlstr As String

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        TextBox1.Text = ""
        TextBox2.Text = ""

    End Sub
    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
        Me.Close()
        Form1.Show()

    End Sub

    Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
        Me.Close()
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        masterform.writetextboxtolabel(TextBox1.Text)
        If Len(Trim(TextBox2.Text)) = 0 Then
            MessageBox.Show("Please Enter the Password Correctly", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox1.Focus()
            Exit Sub
        End If
        valid()

        sqlstr = "select * from password where u_rname ='" & TextBox1.Text & "'
And password='" & TextBox2.Text & "' And type='" & ComboBox1.SelectedItem & "'"
        con.Open()
        Dim cmd As SqlCommand = New SqlCommand(sqlstr, con)
        Dim dr1 As SqlDataReader = cmd.ExecuteReader
        If dr1.Read Then
            Me.Hide()
            If ComboBox1.SelectedItem = "Admin" Then
```

```
                masterform.Show()
            ElseIf ComboBox1.SelectedItem = "Student" Then
                stu_details.Show()
            ElseIf ComboBox1.SelectedItem = "Teacher" Then
                masterform.Show()
            End If
        Else
            MessageBox.Show("Entered Username or password is incorrect",
"Warning", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End If
        con.Close()
    End Sub
    Public Sub valid()
        Dim pattern As String = "^[a-zA-Z][\w\.-]*[a-zA-Z0-9]@[a-zA-Z0-9][\w\.-
]*[a-zAZ0-9]\.[a-zA-Z][a-zA-Z\.]*[a-zA-Z]$"
        If TextBox1.Text = pattern Then
            MessageBox.Show("Invalid Character", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox1.Focus()

        End If
    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles
Button4.Click
        masterform.Show()
    End Sub

    Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs)
Handles ComboBox1.SelectedIndexChanged
        If ComboBox1.SelectedItem = "Student" Then
            Label2.Text = "Regno"
        Else
            Label2.Text = "Username"
        End If
End Sub
```

Source Code of login page

```
Imports System.Data.SqlClient
Public Class Registration_Form
    Dim con_str As String = "Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30"
    Dim con As New SqlConnection(con_str)
    Dim cmd As New SqlCommand
    Dim adapter As New SqlDataAdapter
    Dim ap As String = ""
    Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
        Me.Hide()
        Form1.Show()
    End Sub
    Private Sub Registration_Form_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        data()
        TextBox2.Focus()
    End Sub

    Sub data()
        con.Open()
        Dim sqlstr As String = "select Max(stdno) + 1 from registration"
        Dim cmd As SqlCommand = New SqlCommand(Sqlstr, con)
        Dim dr1 As SqlDataReader = cmd.ExecuteReader

        If dr1.Read Then
            ap = IIf(IsDBNull(dr1(0)), 1, dr1(0))

        End If

        con.Close()
    End Sub

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


        If Len(Trim(TextBox2.Text)) = 0 Then
```

```
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox2.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox3.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox3.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox4.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox4.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox5.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox6.Focus()
            Exit Sub
        End If

        If Len(Trim(TextBox6.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox6.Focus()
            Exit Sub
        End If
        Dim gen As String = ""
        If RadioButton1.Checked Then
            gen = "Male"
        ElseIf RadioButton2.Checked Then
            gen = "Female"
        End If
        Dim sqlstr As String = "insert into registration values('" & ap & "','" &
TextBox2.Text & "','" & TextBox6.Text & "','" & TextBox3.Text & "','" &
TextBox4.Text & "','" & gen & "','" & ComboBox1.SelectedItem & "','" &
ComboBox2.SelectedItem & "','" & DateTimePicker1.Value & "','" & "No" & "','" & 0
& "','" & TextBox5.Text & "')"
        con.Open()
        Dim cmd1 As SqlCommand = New SqlCommand(sqlstr, con)
        cmd1.ExecuteNonQuery()
        con.Close()
        pass()
        MessageBox.Show("Student Sucessfully Registered ", " Information",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox5.Clear()
        TextBox6.Clear()
        'TextBox1.Clear()
        RadioButton1.Checked = False
        RadioButton2.Checked = False
        ComboBox1.SelectedIndex = -1
        ComboBox2.SelectedIndex = -1

    End Sub
    Public Sub pass()
        Dim s As String = "Student"
        Dim sqlstr As String = "insert into password values('" & s & "','" &
TextBox6.Text & "','" & TextBox5.Text & "')"
        con.Open()
        Dim cmd1 As SqlCommand = New SqlCommand(sqlstr, con)
        cmd1.ExecuteNonQuery()
        con.Close()
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox6.Clear()
        TextBox5.Clear()
        RadioButton1.Checked = False
        RadioButton2.Checked = False
        ComboBox1.SelectedIndex = -1
        ComboBox2.SelectedIndex = -1
    End Sub
    Private Sub TextBox2_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox2.KeyPress
```

```
        If Not (Asc(e.KeyChar) = 8) Then
            Dim allowedChars As String = "abcdefghijklmnopqrstuvwxyz"
            If Not allowedChars.Contains(e.KeyChar.ToString.ToLower) Then
                e.KeyChar = ChrW(0)
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub TextBox3_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox3.KeyPress
        If Not (Asc(e.KeyChar) = 8) Then
            Dim allowedChars As String = "abcdefghijklmnopqrstuvwxyz"
            If Not allowedChars.Contains(e.KeyChar.ToString.ToLower) Then
                e.KeyChar = ChrW(0)
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub TextBox6_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox6.KeyPress
        If Char.IsDigit(e.KeyChar) = False Then
            If e.KeyChar = CChar(ChrW(Keys.Back)) Or e.KeyChar =
CChar(ChrW(Keys.Space)) Then
                e.Handled = False
            Else
                e.Handled = True

            End If
        End If

    End Sub

    Private Sub TextBox4_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox4.KeyPress
        If Char.IsDigit(e.KeyChar) = False Then
            If e.KeyChar = CChar(ChrW(Keys.Back)) Or e.KeyChar =
CChar(ChrW(Keys.Space)) Then
                e.Handled = False
            Else
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub TextBox4_Leave(sender As Object, e As EventArgs) Handles
TextBox4.Leave
        If (TextBox4.TextLength < 10 Or TextBox4.TextLength > 13) Then
            MsgBox("Mobile Number Should Be of 10-13 Digits",
MsgBoxStyle.Exclamation, "Warning")
            TextBox4.Focus()
        End If
    End Sub

    Private Sub TextBox5_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox5.KeyPress
        If Char.IsDigit(e.KeyChar) = False Then
            If e.KeyChar = CChar(ChrW(Keys.Back)) Or e.KeyChar =
CChar(ChrW(Keys.Space)) Then
                e.Handled = False
            Else
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub TextBox5_Leave(sender As Object, e As EventArgs) Handles
TextBox5.Leave
        If (TextBox5.TextLength < 5 Or TextBox5.TextLength > 10) Then
            MsgBox("Password Should Be of 5-10 Digits", MsgBoxStyle.Exclamation,
"Warning")
            TextBox5.Focus()
        End If
    End Sub
End Class
```

Teacher Dashboard Source Code

```
Imports System.Data.SqlClient

Public Class masterform
    Dim con As New
SqlConnection("DataSource=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\D
esktop\exam pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")

    Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
        Me.Close()
        Loginform.Show()

    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Registration_Form.Show()

    End Sub
    Public Sub writetextboxtolabel(ByVal txt As String)
        Label3.Text = txt
    End Sub

    Private Sub masterform_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Label4.Text = DateTime.Now.ToString
    End Sub

    Private Sub MenuStrip1_ItemClicked(sender As Object, e As
ToolStripItemClickedEventArgs) Handles MenuStrip1.ItemClicked

    End Sub

    Private Sub ApproveToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles ApproveToolStripMenuItem.Click
        approvalformstu.Show()

    End Sub
    Private Sub HistoryToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles HistoryToolStripMenuItem.Click
        approvalhistory.Show()

    End Sub

    Private Sub LogoutToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles LogoutToolStripMenuItem.Click
        Me.Close()
        Loginform.Show()

    End Sub

    Private Sub ADDQuestionToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles ADDQuestionToolStripMenuItem.Click
        Me.Close()
        addquestionform.Show()

    End Sub

    Private Sub ViewEditToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles ViewEditToolStripMenuItem.Click
        edit_update_question.Show()

    End Sub

    Private Sub AddStaffToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles AddStaffToolStripMenuItem.Click
        AddEmployeeForm.Show()
        Me.Close()

    End Sub

    Private Sub ViewStaffToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles ViewStaffToolStripMenuItem.Click
        view_employee.Show()
        Me.Close()

    End Sub

    Private Sub ExamPasswordToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles ExamPasswordToolStripMenuItem.Click
```

```
        forgetPass.Show()
        Me.Close()

    End Sub

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

    End Sub

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

    End Sub

    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles
Button5.Click
        AddEmployeeForm.Show()
    End Sub

    Private Sub ChangePasswordToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles ChangePasswordToolStripMenuItem.Click
        changepass.Show()
        Me.Close()

    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        result2form.Show()
        Me.Close()

    End Sub
    Sub delete()
        Dim q As String = InputBox("Enter the Registration No ")
        If (MsgBox("Do you want to Delete this Student ", vbYesNo + vbQuestion) =
vbYes) Then
            con.Open()
            cmd = New SqlCommand("delete from registration where regno ='" & q &
"'", con)
```

```
            cmd.ExecuteNonQuery()
            con.Close()
            MsgBox("Student Sucessfully Removed ", vbInformation)
        End If

    End Sub

    Private Sub Button6_Click(sender As Object, e As EventArgs) Handles
Button6.Click
        delete()
    End Sub

    Private Sub MasterToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles MasterToolStripMenuItem.Click
        student_Details.Show()
        'Me.Close()
    End Sub
    Private Sub Button7_Click(sender As Object, e As EventArgs) Handles
Button7.Click
        approvalhistory.Show()
        ' Me.Close()
    End Sub

    Private Sub Button10_Click(sender As Object, e As EventArgs) Handles
Button10.Click
        Loginform.Show()
        Me.Close()
    End Sub
    Private Sub Button8_Click(sender As Object, e As EventArgs) Handles
Button8.Click
        student_Details.Show()
    End Sub
    Private Sub HomeToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles HomeToolStripMenuItem.Click
        Statistics.Show()
    End Sub
End Class
```

Display of Question Source Code

```
Imports System.Data.SqlClient
Public Class examquestion
    Dim table As New DataTable
    Dim index As Integer
    Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")
    Dim d, diff As TimeSpan
    Dim d2 As DateTime

    Private Sub examquestion_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        con.Open()
        Dim sql As String = "select
studname,regno,contact_no,gender,course,qualification,approve from registration
where regno = '" & r & "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()
        Label10.Text = myreader("studname")
        Label6.Text = myreader("regno")
        Label7.Text = myreader("course")
        Label8.Text = myreader("qualification")
        Dim se As Integer = myreader("qualification").ToString
        con.Close()

        con.Open()
        Dim sql1 As String = "select qno,question,sem,opA,opB,opC,opD,ans from
question where sem ='" & se & "' "
        Dim cmd1 As New SqlCommand(sql1, con)
        Dim myreader1 As SqlDataReader
        myreader1 = cmd1.ExecuteReader
        myreader1.Read()
        Label11.Text = myreader1("qno")
        TextBox1.Text = myreader1("question")
        RadioButton1.Text = myreader1("opA")
        RadioButton2.Text = myreader1("opB")
        RadioButton3.Text = myreader1("opC")
        RadioButton4.Text = myreader1("opD")
        qnum = myreader1("qno").ToString
        se = myreader1("sem").ToString
        ans = myreader1("ans").ToString
        con.Close()
        Dim ca As String = ""

        del()
        If Label11.Text = 1 Then
            marks()

        End If
        If RadioButton1.Checked = True Then
            an = "A"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()

        ElseIf RadioButton2.Checked = True Then
            an = "B"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()
        ElseIf RadioButton3.Checked = True Then
            an = "c"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()
        ElseIf RadioButton4.Checked = True Then
            an = "D"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()

        End If

    End Sub

    Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click

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

        'marks()
        If Label11.Text = 1 Then
            marks()
        End If
        Dim qno As Integer = Val(Label11.Text + 1)
        con.Open()
        Dim sq As String = "select max(qno) from question where sem ='" &
Label8.Text & "'"
        Dim cmd As New SqlCommand(sq, con)
        Dim q As String = cmd.ExecuteScalar().ToString
        If qno = q Then
            submitform.Show()
            Me.Close()
        End If
        con.Close()
        If RadioButton1.Checked = False And RadioButton2.Checked = False And
RadioButton3.Checked = False And RadioButton4.Checked = False Then
            MessageBox.Show("Please Select the Answer", "Answer Not Selected ",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
        Else
            con.Open()
            Dim sql1 As String = "select qno,question,sem,opA,opB,opC,opD,ans
from question where sem ='" & sem & "' and qno='" & qno & "' "
            Dim cmd1 As New SqlCommand(sql1, con)
            Dim myreader1 As SqlDataReader
            myreader1 = cmd1.ExecuteReader
            myreader1.Read()
            Label11.Text = myreader1("qno")
            TextBox1.Text = myreader1("question")
            RadioButton1.Text = myreader1("opA")
            RadioButton2.Text = myreader1("opB")
            RadioButton3.Text = myreader1("opC")
            RadioButton4.Text = myreader1("opD")
            ans = myreader1("ans").ToString
            con.Close()
        End If
        If qno = q Then

            Button2.Enabled = False
        End If
        marks()


    End Sub


    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click

        Dim qno As Integer = Val(Label11.Text - 1)
        If qno = 1 Then
            Button1.Enabled = False
            Button2.Enabled = True
        End If
        If RadioButton1.Checked = False And RadioButton2.Checked = False And
RadioButton3.Checked = False And RadioButton4.Checked = False Then
            MessageBox.Show("Please Select the Answer", "Answer Not Selected ",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
        Else
            con.Open()
            Dim sql1 As String = "select qno,question,sem,opA,opB,opC,opD,ans
from question where sem ='" & sem & "' and qno='" & qno & "' "
            Dim cmd1 As New SqlCommand(sql1, con)
            Dim myreader1 As SqlDataReader
            myreader1 = cmd1.ExecuteReader
            myreader1.Read()
            Label11.Text = myreader1("qno")
            TextBox1.Text = myreader1("question")
            RadioButton1.Text = myreader1("opA")
            RadioButton2.Text = myreader1("opB")
            RadioButton3.Text = myreader1("opC")
            RadioButton4.Text = myreader1("opD")
            ans = myreader1("ans").ToString
            con.Close()
        End If


    End Sub
    Public Sub del()
        Dim qry1 As String = "delete from result where regno ='" & Label6.Text &
"''"
        con.Open()
        Dim cmd1 As SqlCommand = New SqlCommand(qry1, con)
        cmd1.ExecuteNonQuery()
        con.Close()
    End Sub

    Public Sub marks()
        If RadioButton1.Checked = True Then
            an = "A"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()
        ElseIf RadioButton2.Checked = True Then
            an = "B"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()
        ElseIf RadioButton3.Checked = True Then
            an = "c"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()
        ElseIf RadioButton4.Checked = True Then
           an = "D"
            Dim qry As String = "insert into result values('" & Label6.Text &
"','" & Label7.Text & "','" & Label8.Text & "','" & Label11.Text & "','" & ans &
"','" & an & "')"
            con.Open()
            Dim cmd11 As SqlCommand = New SqlCommand(qry, con)
            cmd11.ExecuteNonQuery()
            con.Close()

        End If
    End Sub
End Class
```

Display of Result Source Code

```
Imports System.Data.SqlClient
Public Class resultform
    Dim sm, tm As Integer
    Dim nam, regno, course, semester, marks_scored, total_marks, percentage,
grade, result As String

    Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")
    Private Sub resultform_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        rr = InputBox("PLEASE ENTER YOUR REGNO")
        Dim sqlstr As String = "select regno from result where regno ='" & rr &
"'"
        con.Open()
        Dim cmd As SqlCommand = New SqlCommand(sqlstr, con)
        Dim dr1 As SqlDataReader = cmd.ExecuteReader
        If dr1.Read Then
            Me.Hide()
        Else
            MessageBox.Show("Entered Regno is incorrect", "Warning",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            Me.Close()
            End
        End If
        con.Close()
        show_details()
        show_details_1()
        show_details_2()
        show_details_3()
        show_details_4()
        fresult()
    End Sub
    Public Sub show_details()
        con.Open()
        Dim sql As String = "select studname,regno from registration where regno
= '" & rr & "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()
        Label24.Text = myreader("studname")
        nam = myreader("studname").ToString
        Label3.Text = myreader("regno")
        regno = myreader("regno").ToString
        con.Close()
    End Sub
    Public Sub fshow()
        con.Open()
        Dim sql As String = "select
name,regno,course,semester,marks_scored,total_marks,percentage,grade,result from
f_result where regno = '" & rr & "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()


    End Sub
    Public Sub show_details_1()
        con.Open()
        Dim sql As String = "select course,sem from result where regno = '" & rr
& "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()
        Label5.Text = myreader("course")
        course = myreader("course").ToString
        Label7.Text = myreader("sem")
        semester = myreader("sem").ToString
        con.Close()
    End Sub
    Public Sub show_details_2()
        con.Open()
        Dim sql As String = "select count(*)as aa from result where
crc_ans=answer and regno = '" & rr & "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()
        Label9.Text = myreader("aa").ToString
        sm = myreader("aa").ToString
        marks_scored = myreader("aa").ToString
        con.Close()
    End Sub
    Public Sub show_details_3()
        con.Open()
        Dim sql As String = "select count(*)as tq from result where regno = '" &
rr & "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()
        Label11.Text = myreader("tq").ToString
        tm = myreader("tq").ToString
        total_marks = myreader("tq").ToString
        con.Close()
    End Sub
    Public Sub show_details_4()
        Dim avg As Integer
        avg = sm / tm * 100
        Label26.Text = avg
        percentage = Label26.Text

        'Dim grade As String
        If avg > 90 Then
            Label13.Text = "A+"
            grade = "A+"
            Label15.Text = "PASS"
        ElseIf avg >= 90 Then
            Label13.Text = "A"
            grade = "A"
            Label15.Text = "PASS"
        ElseIf avg >= 80 Then
            Label13.Text = "B+"
            grade = "B+"
            Label15.Text = "PASS"
        ElseIf avg >= 70 Then
            Label13.Text = "B"
            grade = "B"
            Label15.Text = "PASS"
        ElseIf avg >= 60 Then
```

```
            Label13.Text = "c"
            grade = "C"
            Label15.Text = "PASS"
        Else
            Label13.Text = "c"
            grade = "C"
            Label15.Text = "FAIL"
        End If
        result = Label15.Text
    End Sub
    Public Sub fresult()
        con.Open()
        Dim qry As String = "insert into f_result values('" & nam & "','" & regno
& "','" & course & "','" & semester & "','" & marks_scored & "','" & total_marks
& "','" & percentage & "','" & grade & "','" & result & "')"
        Dim cmd1 As SqlCommand = New SqlCommand(qry, con)
        cmd1.ExecuteNonQuery()
        con.Close()
    End Sub
    Private Sub PictureBox2_Click(sender As Object, e As EventArgs) Handles
PictureBox2.Click
        Me.Close()
        Form1.Show()
    End Sub
End Class
```

Adding Question

```
Imports System.Data.SqlClient
Public Class addquestionform
    Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")
    Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
        Me.Close()
        masterform.Show()


    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        If ComboBox1.Text.Trim() = "" Then
            MessageBox.Show("Please Select the Course ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox5.Focus()
            Exit Sub
        End If
        If ComboBox2.Text.Trim() = "" Then
            MessageBox.Show("Please select the Semester ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox5.Focus()
            Exit Sub
        End If

        If ComboBox3.Text.Trim() = "" Then
            MessageBox.Show("Please Select the Correct Answer ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox5.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox1.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox1.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox2.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox2.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox3.Text)) = 0 Then
```

```
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox3.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox4.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox4.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox5.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox5.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox6.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox6.Focus()
            Exit Sub
        End If
        con.Open()
        Dim sc As String = ComboBox1.SelectedItem.ToString
        Dim sem As String = ComboBox2.SelectedItem.ToString
        Dim ans As String = ComboBox3.SelectedItem.ToString
        Dim Sql As String = "insert into question values('" & TextBox1.Text &
"','" & sc & "','" & sem & "','" & TextBox2.Text & "','" & TextBox3.Text & "','"
& TextBox4.Text & "','" & TextBox5.Text & "','" & TextBox6.Text & "','" & ans &
"')"
        Dim cmd As New SqlCommand(Sql, con)
        cmd.ExecuteNonQuery()
        MessageBox.Show("Question Added SucessFully .... ", "SuccesssFull",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        con.Close()
        TextBox1.Text = ""
        TextBox2.Text = ""
        TextBox3.Text = ""
        TextBox4.Text = ""
        TextBox5.Text = ""
        TextBox6.Text = ""
        ComboBox1.SelectedIndex = -1
        ComboBox2.SelectedIndex = -1
        ComboBox3.SelectedIndex = -1
    End Sub
    Public Sub clear()
        TextBox1.Clear()
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox5.Clear()
        TextBox6.Clear()
        ComboBox1.SelectedIndex = -1
        ComboBox2.SelectedIndex = -1
        ComboBox3.SelectedIndex = -1
    End Sub
    Public Sub qstno()
        con.Open()
        Dim sqlstr As String = " select Max(qno) + 1 from question where course
='" & ComboBox1.SelectedItem & "' and sem='" & ComboBox2.SelectedItem & "' "
        Dim cmd As SqlCommand = New SqlCommand(sqlstr, con)
        Dim dr1 As SqlDataReader = cmd.ExecuteReader
        If dr1.Read Then
            TextBox1.Text = IIf(IsDBNull(dr1(0)), 1, dr1(0))
        End If

        con.Close()

    End Sub

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

    End Sub

    Private Sub ComboBox2_SelectedIndexChanged(sender As Object, e As EventArgs)
Handles ComboBox2.SelectedIndexChanged
        qstno()
    End Sub

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

    End Sub
End Class
```

Adding Staffs Code

```
Imports System.Data.SqlClient
Public Class AddEmployeeForm
    Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")
    Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
        Me.Close()
        masterform.Show()
    End Sub
    Public Sub empno()
        con.Open()
        Dim sqlstr As String = " select Max(emp_no) + 1 from employee "
        Dim cmd As SqlCommand = New SqlCommand(sqlstr, con)
        Dim dr1 As SqlDataReader = cmd.ExecuteReader
        If dr1.Read Then
            TextBox1.Text = IIf(IsDBNull(dr1(0)), 1, dr1(0))
        End If

        con.Close()
    End Sub

    Private Sub AddEmployeeForm_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        empno()

    End Sub
    Private Sub TextBox2_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox2.KeyPress
        If Not (Asc(e.KeyChar) = 8) Then
            Dim allowedChars As String = "abcdefghijklmnopqrstuvwxyz"
            If Not allowedChars.Contains(e.KeyChar.ToString.ToLower) Then
                e.KeyChar = ChrW(0)
                e.Handled = True
            End If
        End If
    End Sub
    Private Sub TextBox3_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox3.KeyPress
        If Not (Asc(e.KeyChar) = 8) Then
            Dim allowedChars As String = "abcdefghijklmnopqrstuvwxyz"
            If Not allowedChars.Contains(e.KeyChar.ToString.ToLower) Then
                e.KeyChar = ChrW(0)
                e.Handled = True
            End If
        End If
    End Sub
    Private Sub TextBox4_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox4.KeyPress
        If Char.IsDigit(e.KeyChar) = False Then
            If e.KeyChar = CChar(ChrW(Keys.Back)) Or e.KeyChar =
CChar(ChrW(Keys.Space)) Then
                e.Handled = False
            Else
                e.Handled = True
            End If
        End If
    End Sub

    Private Sub TextBox4_Leave(sender As Object, e As EventArgs) Handles
TextBox4.Leave
        If (TextBox4.TextLength < 10 Or TextBox4.TextLength > 13) Then
            MsgBox("Mobile Number Should Be of 10-13 Digits",
MsgBoxStyle.Exclamation, "Warning")
            TextBox4.Focus()
        End If
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        If Len(Trim(TextBox3.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox3.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox4.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox4.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox5.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox5.Focus()
            Exit Sub
        End If
        If Len(Trim(TextBox6.Text)) = 0 Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox5.Focus()
            Exit Sub
        End If
        If RadioButton1.Checked = False And RadioButton2.Checked = False Then
            MessageBox.Show("Incomplete information ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
        End If
        Dim gen As String = ""
        If RadioButton1.Checked = True Then
            gen = "Male"
        ElseIf RadioButton2.Checked = True Then
            gen = "Female"
        End If
        Dim sqlstr As String = "insert into employee values('" & TextBox1.Text &
"','" & TextBox2.Text & "','" & DateTimePicker1.Value & "','" &
DateTimePicker2.Value & "','" & TextBox3.Text & "','" & gen & "','" &
TextBox4.Text & "','" & TextBox5.Text & "','" & TextBox6.Text & "')"
        con.Open()
        Dim cmd1 As SqlCommand = New SqlCommand(sqlstr, con)
        cmd1.ExecuteNonQuery()
        con.Close()
        pass()
        MessageBox.Show("Sucessfully added New Employee ", "Sucessfully",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        RadioButton1.Checked = False

        RadioButton2.Checked = False
    End Sub
    Public Sub pass()
        Dim s As String = "Admin"
        Dim sqlstr As String = "insert into password values('" & s & "','" &
TextBox5.Text & "','" & TextBox6.Text & "')"
        con.Open()
        Dim cmd1 As SqlCommand = New SqlCommand(sqlstr, con)
        cmd1.ExecuteNonQuery()
        con.Close()

    End Sub

    Private Sub TextBox5_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TextBox5.KeyPress
        If Not (Asc(e.KeyChar) = 8) Then
            Dim allowedChars As String = "abcdefghijklmnopqrstuvwxyz"
            If Not allowedChars.Contains(e.KeyChar.ToString.ToLower) Then
                e.KeyChar = ChrW(0)
                e.Handled = True
            End If
        End If
    End Sub
End Class
```

```
Imports System.Data.SqlClient

Public Class view_employee
    Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\dell\Desktop\exam
pics\Exam_Database.mdf;Integrated Security=True;Connect Timeout=30")
    Private Sub view_employee_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        loadgrid()

    End Sub
    Public Sub loadgrid()
        Dim Sql As String = "Select * from employee"
        Dim cmd As New SqlCommand(Sql, con)
        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable
        da.Fill(dt)
        DataGridView1.DataSource = dt
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
        If Len(Trim(TextBox1.Text)) = 0 Then
            MessageBox.Show("Please Type the Employee Number  ", "Input Error",
MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            TextBox1.Focus()
            Exit Sub
        End If
        con.Open()

        Dim sql As String = "select emp_name,addr,gen,contact,username,password
from employee where emp_no = '" & TextBox1.Text & "'"
        Dim cmd As New SqlCommand(sql, con)
        Dim myreader As SqlDataReader
        myreader = cmd.ExecuteReader
        myreader.Read()
        TextBox2.Text = myreader("emp_name")
        TextBox3.Text = myreader("addr")
        TextBox4.Text = myreader("gen")
        TextBox5.Text = myreader("contact")
        TextBox6.Text = myreader("username")
        TextBox7.Text = myreader("password")
        con.Close()

    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
```

```
        Dim q As Integer = InputBox("Please Enter the Question No ")
        If (MsgBox("Do you want to Delete this Question ", vbYesNo + vbQuestion)
= vbYes) Then
            con.Open()
            cmd = New SqlCommand("delete employee where emp_no ='" & q & "'",
con)
            cmd.ExecuteNonQuery()
            con.Close()
            MsgBox("Employee Sucessfully Removed ", vbInformation)
            loadgrid()
            TextBox1.Text = ""
            TextBox2.Text = ""
            TextBox3.Text = ""
            TextBox4.Text = ""
            TextBox5.Text = ""
            TextBox6.Text = ""
            TextBox7.Text = ""
        End If
    End Sub
End Class
```

Advantages of "Computerized Examination System"

A Computerized examination system has plenty of advantages:

1. **It saves paper.**

You never have to print an exam for your students and hand them out. Saves paper. Saves trees. Everybody is happy.

2. **It saves time.**

You can set up an exam in such a way that it will auto-grade itself. If you only use multiple-choice questions you never have to check an exam again. The computerized exam system will take care of that hassle. Completely automated.

3. **It saves more time.**

The distribution of the exam doesn't take you any time. Just upload the email addresses of your students and send them an invite. And after the exam, they get their result instantly.

4. **It saves you money.**

You don't need to buy any paper. Sending an email is free. On top of that, you save on the logistics: your students don't have to assemble in the classroom to take the exam.

They can do it within a given time frame from their device. You don't have to rent a classroom. You don't have to hire someone to check the students taking the exam.

5. **It saves the student money.**

Students don't have to travel to a specific location to conduct the exam. So even for students from remote areas, it's possible to take the exam.

6. **It's more secure.**

You can make a big question bank with a lot of questions. Every student gets a random selection from that question bank. So it's of little use to share the questions among the exam takers to give them a head start

FUTURE SCOPE :

The existing system is limited with few modules and not easy to access the student details and details of the examination.

But our purpose system provides the student details and results of the student adding of question and appropriately updating them.

The computerized system can be used in private institutions as well as educational institutions. As it is a user-friendly web-based application it can be used anywhere and anytime. Every software may have some cases of bugs. Students may use private equipment – computers to write -in-class examinations.

Computerized examination saves your money in so many ways. Since everything is computerized, there are no printing costs and no logistics costs. Pen-paper examinations require a lot of paper to print question and answer sheets. There is also a lot of waste due to printing errors or over-estimation of learner numbers, not to mention the carbon footprint of the logistics around getting the papers to and from examination locations.

In the past, the more people who took an exam, the bigger the challenge to facilitate it. Computerized examinations make it super easy to scale. Setting up an exam for 1,000 people takes almost the same amount of effort as it is to set up an exam for 10 people. Another advantage of the tech-centric nature of computerized exams is that the more computerized exams people take, the more they get used to the concept and the more comfortable they get with it.

Security has always been a challenge with exams, especially with high-profile exams like bar exams, SATs, college, and university final exams. With online exams, there are fewer chances of leaks since there are no physical papers that can go missing during the printing and logistics process. An added security benefit is that examiners can make use of question banks that select questions at random. This means that almost no two exams are the same, further minimizing the chances of cheating. One of the biggest advantages of computerized examinations is the convenience factor. Examiners can set papers using question banks, and by consulting a database of previous papers, then easily upload it to the examination system. Students can select exam times that suit them best and since the exams are online, do it from almost anywhere (proctored exams may have certain requirements). Students can get their results almost immediately, a big plus. Also, as mentioned above, online examinations scale extremely easily, making it convenient for course administrators to set up exams.

CONCLUSION :

The purpose of the computerized examination system is to test the subject knowledge of the students.Such a system eliminates the logistical hassle and drawbacks in the traditional model of the pen–and–paper examination.As they are convenient, they offer flexibility. They bring the right education to your home. They offer more individual attention. Computerized examination systems help students meet interesting people. It gives students real-world skills. In communication,

The computerized examination is kind of beneficial to the students, tutors and the institution offering. As of now most of the competitive examinations are online, so it is necessary since this method of computerized examination is the only option.

A computerized examination system is a user-friendly system, which is very easy and convenient to use. The system is complete in the sense that it is operational and is tested by entering data and getting the reports in proper order. But there is always a scope for improvement and enhancement.the constant evolution of technology in the digital world has caused a wave of consolidation in the assessment industry, making pen and paper tests redundant. It has bought a patent clarity that online examination conducted with the help of online exam software is the future of assessment methodology. “Future of Computerized Examination System”

The growth of the online industry at a fast pace and obsoleteness of traditional evaluation techniques is making people lose confidence in the current pen-paper examinations.

Today’s article will discuss the future of online examination possible through an online examination system. In this article, we are mentioning the reasons that have dwindled educational institutions' assurance in the conventional evaluation system and the numerous benefits that arise from transitioning to computer-based evaluation.

Before proceeding, let us understand the concept of an online examination system that clothes online computer tests. An online examination software is software that makes conducting of test possible through a computer network and the internet.

educational institutions losing their confidence in pen and paper-based modes of assessment.

Technology has embraced the educational sector in such a manner that it has left no space for pen-paper. The spirit of schools, colleges, universities, and training institutes have withered in the traditional model of assessment due to the following reasons :

● the abundance of entrance examination applications: Universities every year have to incur costs for screening the received abundant application with written tests. Shortlisting facilities to conduct physical tests consumes lots of effort and making space for exam centers disrupts the college life of students.

● the fuss of staff recruitment: Arranging invigilators, and conducting an interview, for choosing teachers to assess the pen-paper examination, is another herculean requirement of this assessment pedagogy.

● improper assessment competencies: The conventional mode of examination is devoid of a definitive way to measure children's ability and personality. This method disables the teachers to precisely tell the students about their behavioral and cognitive competencies. accrued benefits that arise from transitioning to computer-based evaluation: ranging from a school, university, training institute to a corporation running certification programs or assessing training effectiveness.

Computerized examinations provided by the online examination system are being welcomed with open arms by both the exam candidates as well as the organization providing an assessment.

● reduced administrative burden: Printing and circulating question papers and organizing logistics to transport completed scripts to makers is a time-consuming and costly process. On the other hand, organizing and running exams online reduces the organization's administrative burden and also saves time and cost.

BIBLIOGRAPHY

REFERENCES

- www.w3schools.com
- www.guru99.com
- www.roseindia.net