

INTERNSHIP REPORT

A report submitted in partial fulfilment of the requirements for the

Award of the Degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

by

KUNAL RAJ, GOURAB GHOSH

Roll Nos.: 10300220051, 10300220041

Regd. Nos.: 201030100210088, 201030100210074

Under the Supervision of

Mr. Rahul Sharma

Microsoft & HPE Certified Trainer

Ardent Computech Pvt. Ltd.

Kolkata

(Duration: 17th January, 2023 to 17th February, 2023)



**DEPARTMENT OF INFORMATION TECHNOLOGY
HALDIA INSTITUTE OF TECHNOLOGY**

(An Autonomous Institution)

Approved by AICTE, affiliated to MAKAUT, WB

HALDIA, WEST BENGAL

Industrial Internship Certificate



This certificate is awarded to

KUNAL RAJ

of

HALDIA INSTITUTE OF TECHNOLOGY

for successfully completing the **Industrial Internship** on

FULL STACK DEVELOPMENT USING JAVA

from

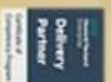
17-01-2023 to 17-02-2023

and implementing the project titled

E-LEARNING PORTAL

Certificate ID: **ARDENT/2023/AD66349**
Issue Date: **17-02-2023**

Microsoft
Technology Associate



Director
Technology Services

Director
Operations



Industrial Training Certificate



This certificate is awarded to

KUNAL RAJ

of

HALDIA INSTITUTE OF TECHNOLOGY

for successfully completing the **Industrial Training** on

FULL STACK DEVELOPMENT USING JAVA

COMPUTECH PVT. LTD. from

High-End Technology Training and Project

17-01-2023 to 17-02-2023

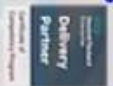
and implementing the project titled

E-LEARNING PORTAL

Certificate ID: ARDENT/2023/AD66366

Issue Date: 17-02-2023

Microsoft
Technology Associate



Director
Technology Services

Director
Operations



Industrial Internship Certificate



This certificate is awarded to

GOURAB GHOSH

of

HALDIA INSTITUTE OF TECHNOLOGY

for successfully completing the **Industrial Internship** on

FULL STACK DEVELOPMENT USING JAVA

from

17-01-2023 to 17-02-2023

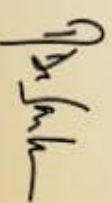
and implementing the project titled


E-LEARNING PORTAL

Certificate ID: ARDENT/2023/AD66348
Issue Date: 17-02-2023

Microsoft
Technology Associate




Director
Technology Services


Director
Operations



Industrial Training Certificate



This certificate is awarded to

GOURAB GHOSH

of

HALDIA INSTITUTE OF TECHNOLOGY

for successfully completing the **Industrial Training** on

FULL STACK DEVELOPMENT USING JAVA

COMPUTECH PVT. LTD.

High-End Technology Training and Project

17-01-2023 to 17-02-2023

and implementing the project titled

E-LEARNING PORTAL

Certificate ID: ARDENT/2023/AD66365

Issue Date: 17-02-2023

Microsoft
Technology Associate



Director
Technology Services

Director
Operations



ACKNOWLEDGEMENT

Success of any project depends largely on the encouragement and guidelines of many others. We take this sincere opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project work.

We would like to show our greatest appreciation to

Rahul Sharma, Technical Trainer at Ardent, Durgapur.

We always feel motivated and encouraged every time by his valuable advice and constant inspiration; without his encouragement and guidance this project would not have materialized. We also want to thank them for sharing their pearls of wisdom with us during this project.

We are also immensely grateful to **Rahul Sir** for his comments on earlier versions of the manuscripts, although any errors are our own and should not tarnish the reputations of these esteemed professionals.

Words are inadequate in offering our thanks to the other trainees, **project assistants** and other members at **Ardent Computech Pvt. Ltd.** for their encouragement and cooperation in carrying out this project work.

The guidance and support received from all the members and who are contributing to this project, was vital for the success of this project.

KUNAL RAJ, GOURAB GHOSH

ABSTRACT

The Teacher-Student Portal for E-Learning is a dynamic and feature-rich web application designed to enhance the educational experience by fostering seamless communication and collaboration between teachers and students. Developed using advanced Java technologies, HTML, CSS, JavaScript, and SQL, this project aims to create a user-friendly and efficient platform that addresses the evolving needs of modern educational environments.

The portal provides distinct yet interconnected modules for teachers and students, offering a range of functionalities to facilitate effective teaching and learning. The user interface is designed with a responsive and intuitive design, ensuring accessibility across various devices and screen sizes.

The **Teacher-Student Portal** for E-Learning not only serves as a centralized hub for educational activities but also promotes a collaborative and engaging learning environment. This project leverages the power of advanced Java technologies and a well-structured database to create a scalable and efficient solution that caters to the diverse needs of educational institutions embracing e-learning methodologies.

We demonstrate the efficiency and effectiveness of our approach through an experimental evaluation using our implemented prototype.

INDEX

S.no.	CONTENTS	Page no.
1.	Introduction	10
1.1.	Modules used	11
2.	System Analysis	12
3.	Requirements specifications	15
4.	Technology	16
4.1.	JAVA EE.....	16
4.2.	HTML	17
4.3.	CSS	17
4.4.	JS	17
4.5.	SQL	17
5.	System Design (with snapshots)	18
6.	Coding	34
7.	Conclusion	36
8.	Future Scope	37
9.	Bibliography & References.....	37

Learning Objectives / Internship Objectives

The basic objective of e-learning education in India is to devise and guide reforms that will transform India into a strong and vibrant knowledge economy. In this context, the focus areas for **TEACHER-STUDENT PORTAL** as a project have been:

- Higher education,
- Professional education, Distance education and
- Continuous and open learning, roughly in that order of preference.
- Increase access to high-quality education for everyone, everywhere
- Enhance teaching and learning on campus and online Advance teaching and learning through research.

Every course on A3 is taught by top instructors from the world's best universities and educational institutions. Courses include recorded video lectures, auto-graded and peer-reviewed assignments, and community discussion.

1. INTRODUCTION

What is E-Learning?

E-Learning is learning utilizing electronic technologies to access educational curriculum outside of a traditional classroom. In most cases, it refers to a course, program or degree delivered completely online. In our project, we have implemented this using a **teacher-student portal**.

There are many terms used to describe learning that is delivered online, via the internet, ranging from Distance Education, to computerized electronic learning, online learning, internet learning and many others. We define eLearning as courses that are specifically delivered via the internet to somewhere other than the classroom where the professor is teaching. It is not a course delivered via a DVD or CD- ROM, video tape or over a television channel. It is interactive in that you can also communicate with your teachers, professors, or other students in class.

E-learning is not just a change of technology. It is part of a redefinition of how we as a species transmit knowledge, skills, and values to younger generations of workers and students. This makes a few predictions of how e-learning and the functions it serves will continue to develop. Learners will have access modules. Some will be Web pages with simple text and graphics. Others may include multimedia simulations. In many fields, e-learning has become the default way to conduct training or to provide education. There are four secrets of e-learning. The first secret is to teach what learners need to learn in the way they most naturally learn. The second secret is to define clear learning objectives. The third secret builds on the first two. It is to focus on the right objectives. The final secret is in the power of testing.

Information and Communication Technology (ICT) has contributed significantly to the efficiency and effectiveness of the industries and organizations. ICT has also empowered the customer and service recipient by ensuring better services and good control over the process of receipt of services.

1.1. Module Description:

Functional requirements:

The modules used in this software are as follows:

- **Registration:** This page contains the user id, name, email id, contact number, address, and password.
- **Login:** This module is for registered users to login. The **ADMIN** has the authority to Add, Delete, and Update etc. The **USER** can only view courses, news updates.
- **Home:** This page contains an overview of highlights from other pages and information about courses, events, contacts, etc
- **Courses:** This page contains all the latest upcoming courses, ongoing courses.
- **About us:** This page displays the details of the website, and how it is helpful to students.
- **Contact:** This page displays the way to contact the admin through website and phone, and comment given from students.

Non-functional requirements:

The modules used in this software are as follows:

- **Usability Requirement:** The system shall allow the users to access the system from any browsers, no special training is required. The system user friendly and the system is written in simple English.
- **Availability Requirement:** The system is available 100% for the user and is used by 24 hours a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.
- **Accuracy:** The system should accurately provide real time information taking into consideration various issues. The system shall provide 100% access reliability.
- **Performance Requirement:** The information is refreshed at regular intervals depending upon whether some updates have occurred or not. The system shall respond the member in less than 2 seconds.
- **Security Requirement:** System will use a secured database and the system will have different users and each user has different types of constraints. Only admins have the rights to update database information of other users.
- **Reliability Requirement:** The system must be 100% reliable due to the importance of data and the damages that can be caused by incorrect data. The system will run 7 days a week and 24 hours a day.

2. SYSTEM ANALYSIS

2.1 Identification of Needs

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem-solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The System is viewed as a whole and the input to the system are identified. The outputs from the organization are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and Decisional variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A **detailed study** of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem area are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with proposal.

2.2 Feasibility Study

Feasibility study is made to see if the project on completion will serve the purpose the organization for completion of work.

Effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it

normally goes through a feasibility study before it is approved for development. The document provides the feasibility of the project that is being designed and lists various area that were considered very carefully during the feasibility study of this project such as Technical, Economic, and operational feasibilities.

2.3 Work Flow

This Document plays a vital role in the development life cycle (SDLC) as it describes the complete requirement of the system. It is meant for use by the developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process.

WATER FALL MODEL was being chosen because all requirements were known beforehand and the objective of our software development is the computerization/automation of an already existing manual working system

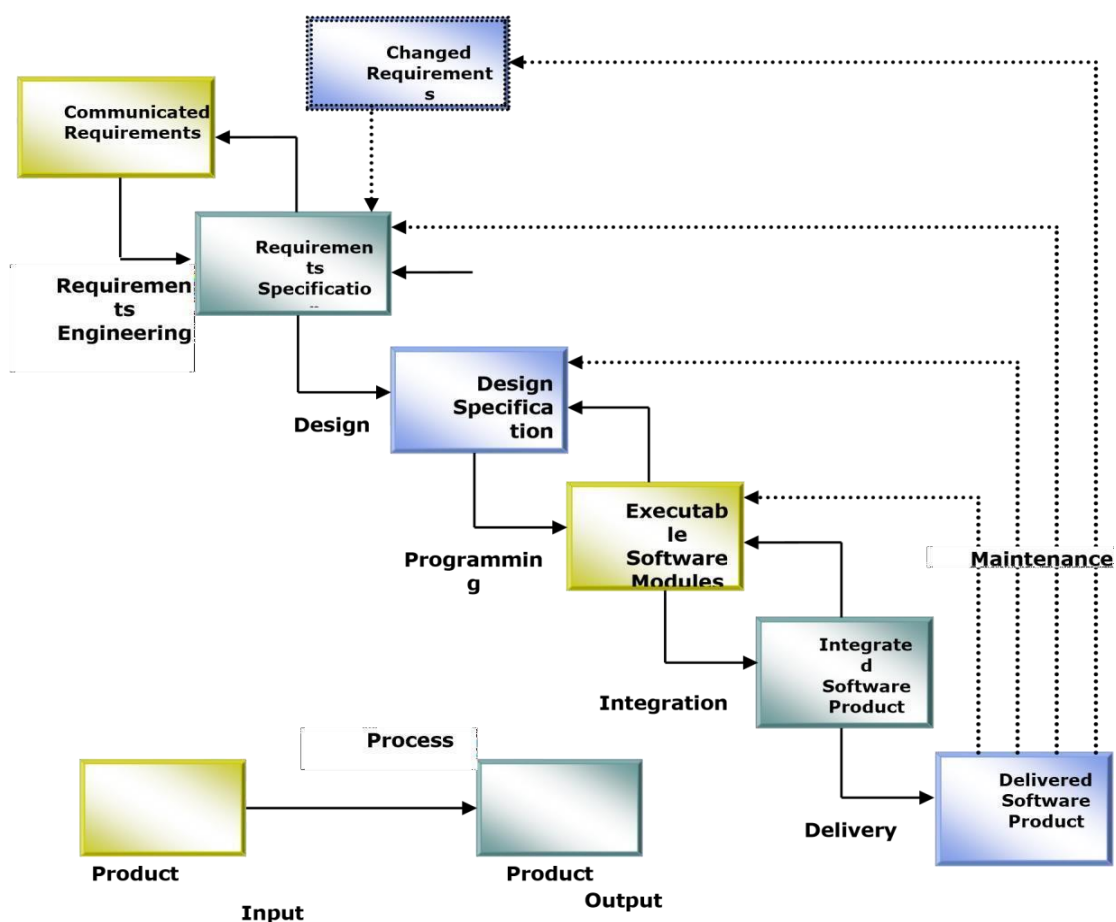


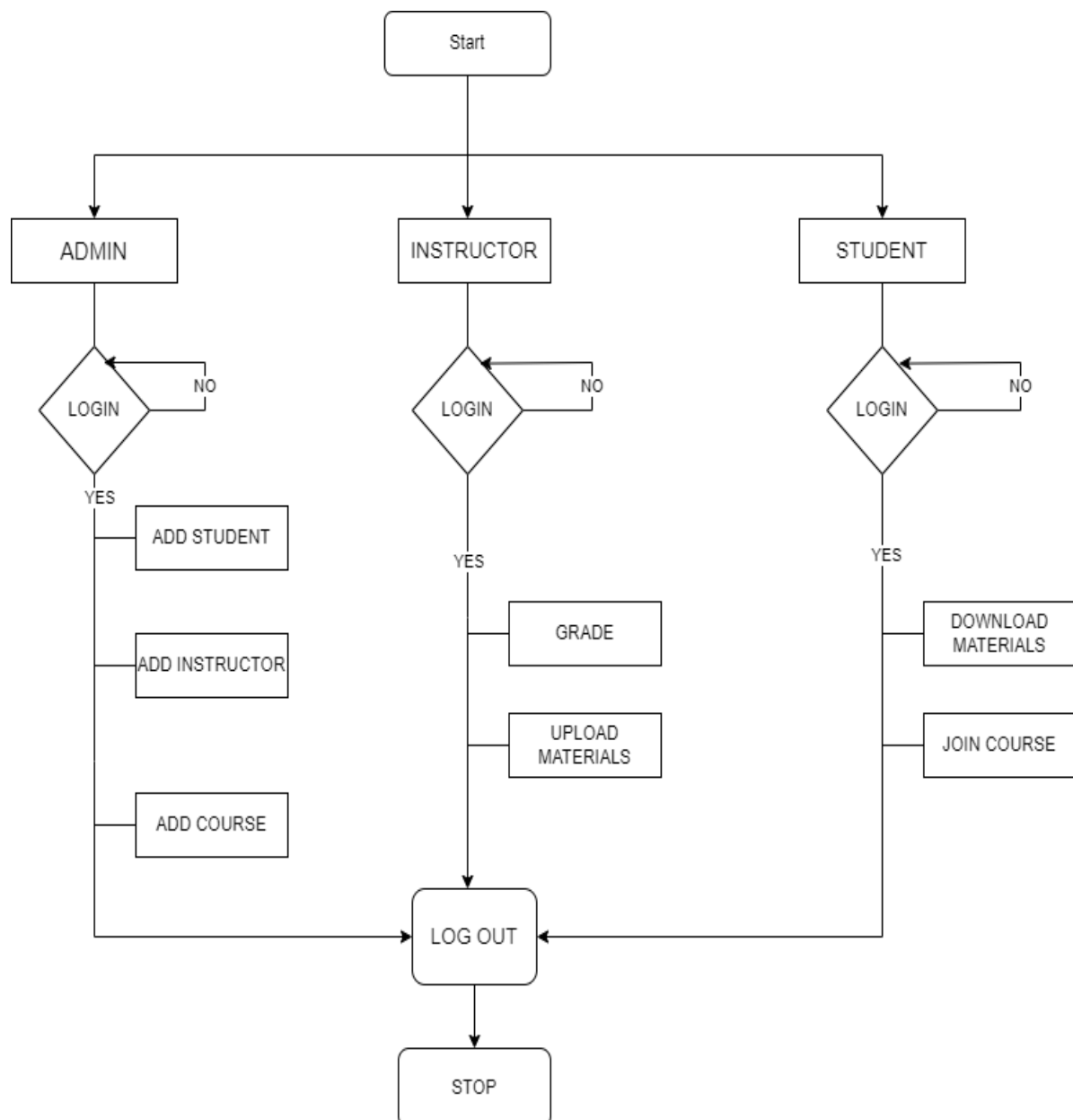
Fig. Waterfall Model

The developer is responsible for:

Developing the system, which meets the SRS and solving all the requirements of the system? Demonstrating the system and installing the system at client's location after the acceptance testing is successful.

Submitting the required user manual describing the system interfaces to work on it and the documents of the system.

Conducting any user training that might be needed for using the system and maintaining the system for a period of one year after installation.



3. REQUIREMENTS SPECIFICATION

Hardware Requirements:

- **System:** Standard computer with at least i3 processor
- **RAM:** 2 GB minimum
- **Hard Disk:** Minimum 100GB of free space required
- **Connectivity:** Active Internet Connectivity with good bandwidth

Software Requirements:

- **Operating System:** Windows 7 Ultimate
- **Database:** SQL Server 2008
- **Coding Languages:** Java 8 or above, HTML, CSS, JS
- **IDE used:** Eclipse
- **Web Application Server:** Apache Tomcat web server 9.0
- **Web Application Tester:** XAMPP
- **Word Application:** Ms Office
- **Drawing Application:** Draw.io

4. TECHNOLOGY

4.1. JAVA EE:

Java EE, or Java Platform, Enterprise Edition, is a set of specifications that extend the Java SE (Standard Edition) to provide a robust and scalable platform for developing enterprise-level applications. It offers a collection of APIs (Application Programming Interfaces) and runtime environments for building large-scale, distributed, and multi-tiered applications.

Here are brief explanations of some key components in Java EE which are used in the project:

1. Servlets:

- a. Servlets are Java classes that extend the functionality of servers to handle requests and generate dynamic responses.
- b. They are typically used to develop web applications and provide a server-side processing logic.

2. JSP (Java Server Pages):

- a. JSP is a technology that enables the creation of dynamic, platform-independent web content.
- b. It simplifies the development of web pages by allowing the embedding of Java code in HTML pages, making it easier to generate dynamic content.

3. JavaBeans:

- a. JavaBeans are reusable software components written in Java that encapsulate many objects into a single, reusable component.
- b. They follow a naming convention and are used to simplify the development and maintenance of Java applications.

4. JDBC (Java Database Connectivity):

- a. JDBC is a Java-based API that provides a standard interface for connecting Java applications to relational databases.
- b. It allows developers to interact with databases, execute SQL queries, and retrieve or update data from Java applications.

Java EE is a comprehensive platform for building enterprise-level applications, and the mentioned components (Servlets, JSP, JavaBeans, JDBC) play specific roles in enabling the development of scalable, distributed, and dynamic web applications. Servlets handle requests, JSP simplifies web page development, JavaBeans encapsulate reusable components, and JDBC facilitates database connectivity for Java applications.

4.2. HTML:

HTML is the standard markup language used to create and design documents on the World Wide Web. It provides a structure for web content by using a set of tags and attributes that define the elements on a web page.

HTML serves as the foundation for web development, providing a standardized way to structure content, create links, integrate multimedia, and capture user input. Its simplicity and versatility make it an essential language for building the backbone of virtually every website on the internet.

4.3. CSS:

CSS is a style sheet language used for describing the presentation of a document written in HTML or XML. It enables web developers to control the layout, appearance, and formatting of multiple web pages simultaneously. CSS separates the structure and content of a web page from its presentation, allowing for a more efficient and flexible web development process.

In essence, CSS is a powerful tool that enhances the aesthetics and user experience of web pages by providing a standardized and efficient way to manage their presentation. It simplifies the process of creating visually appealing, consistent, and responsive websites across various devices and platforms.

4.4. JS:

JavaScript (JS) is a high-level, interpreted programming language primarily known for its role in web development. It is widely used to create dynamic and interactive content on websites.

JavaScript is a versatile programming language that plays a crucial role in web development, offering the ability to create dynamic, interactive, and responsive user experiences both on the client and, with the rise of Node.js, on the server side as well.

4.5. SQL:

SQL is a domain-specific language used for managing and manipulating relational databases. It provides a standardized way to interact with databases, allowing users to define, query, update, and manage data. SQL is widely used in various applications and industries due to its simplicity and effectiveness in handling relational databases.

SQL is a powerful language for managing relational databases, providing a standardized and efficient means of querying, updating, and controlling access to data. Its uses span across various applications, from simple data retrieval to complex database management tasks in enterprise-level systems.

5. SYSTEM DESIGN & SNAPSHOTS

3.1 DATA FLOW DIAGRAM:

A **Data Flow Diagram (DFD)** is a diagram that describes the flow of data and the processes that change data throughout a system. A structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process-oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, and specifies the implementation plan etc. Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

DATA FLOW DIAGRAM SYMBOLS:



Source or Destination of Data



Process



Data Flow



Storage

STEPS TO CONSTRUCT DATA FLOW DIAGRAM:

Four Steps are generally used to construct a DFD:

- Process should be named and referred for easy reference. Each name should be representative of the reference.
- The destination of flow is from top to bottom and from left to right.
- When a process is distributed into lower-level details they are numbered.
- The names of data stores, sources and destinations are written in capital letters.

RULES FOR CONSTRUCTING A DATA FLOW DIAGRAM:

- Arrows should not cross each other.
- Squares, Circles, Files must bear a name.
- Decomposed data flow squares and circles can have same names.
- Draw all data flow around the outside of the diagram.

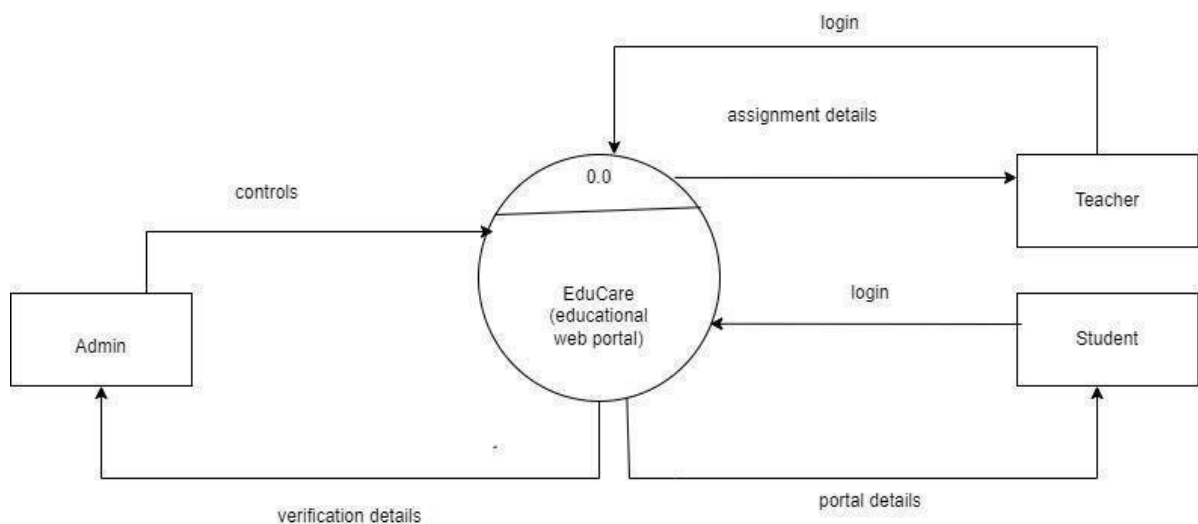
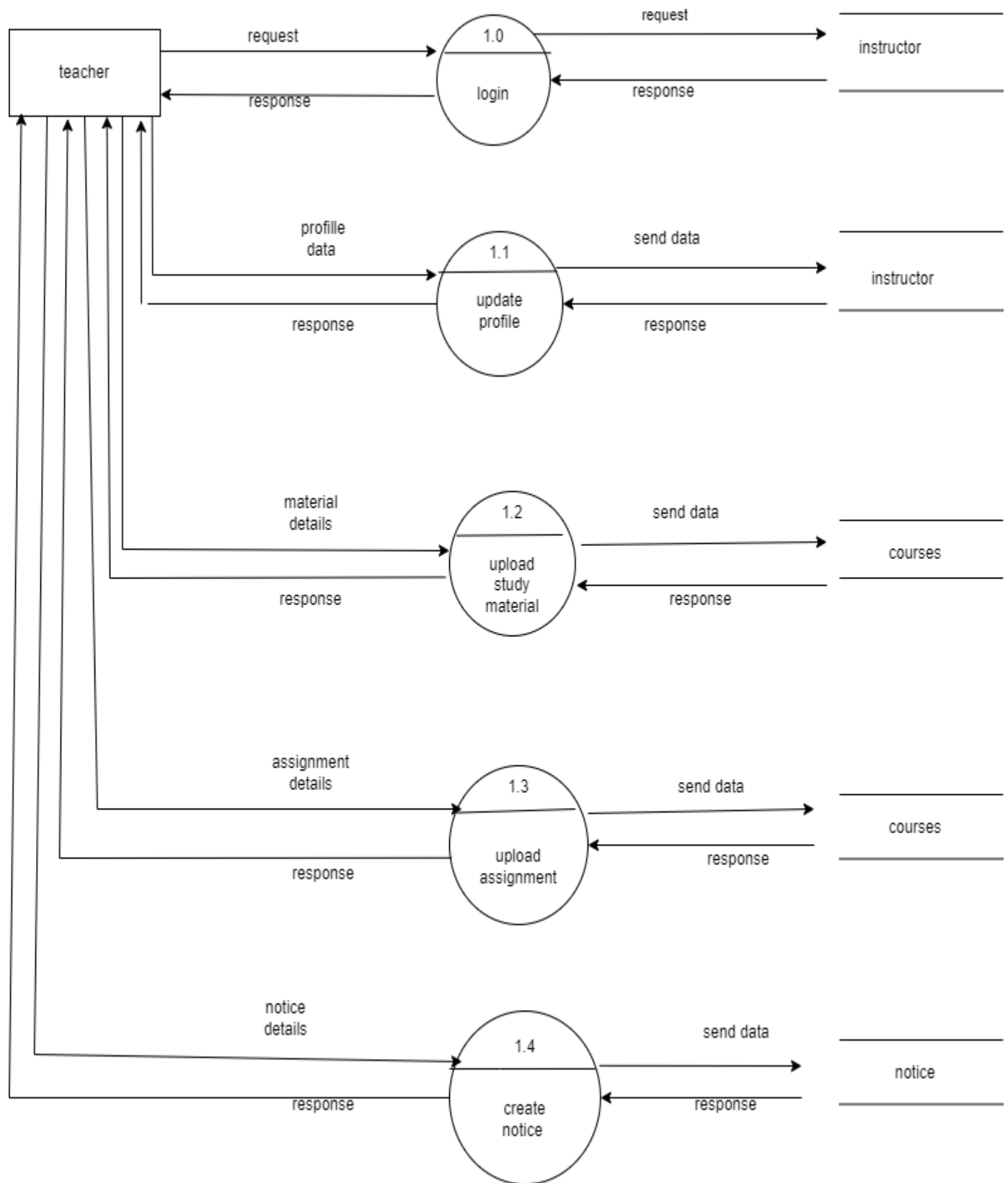
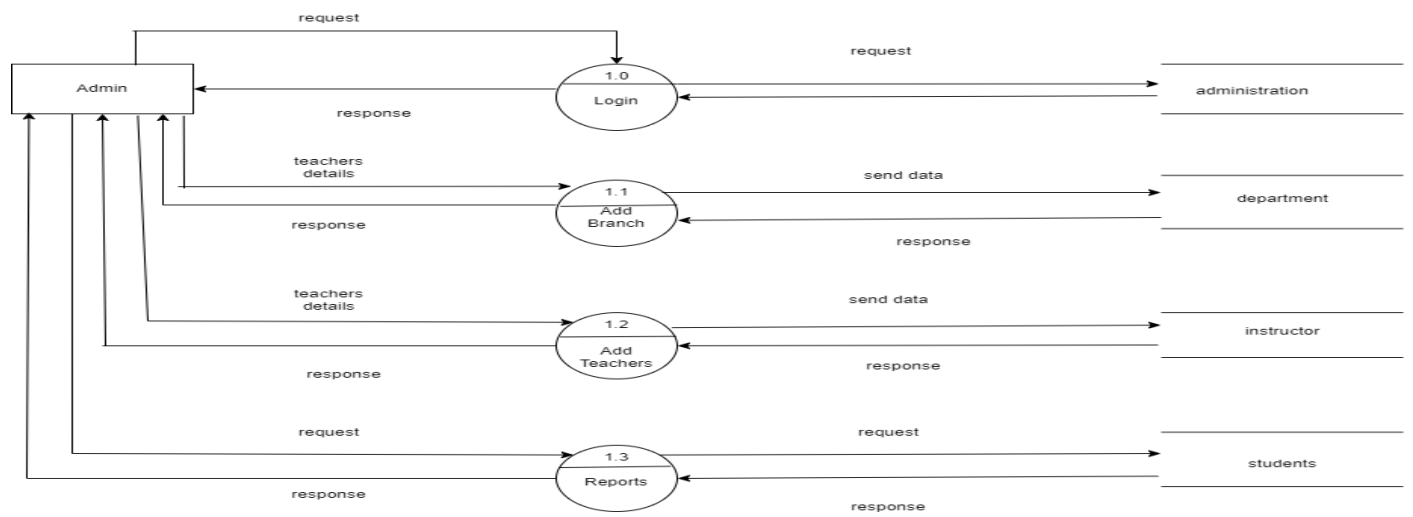


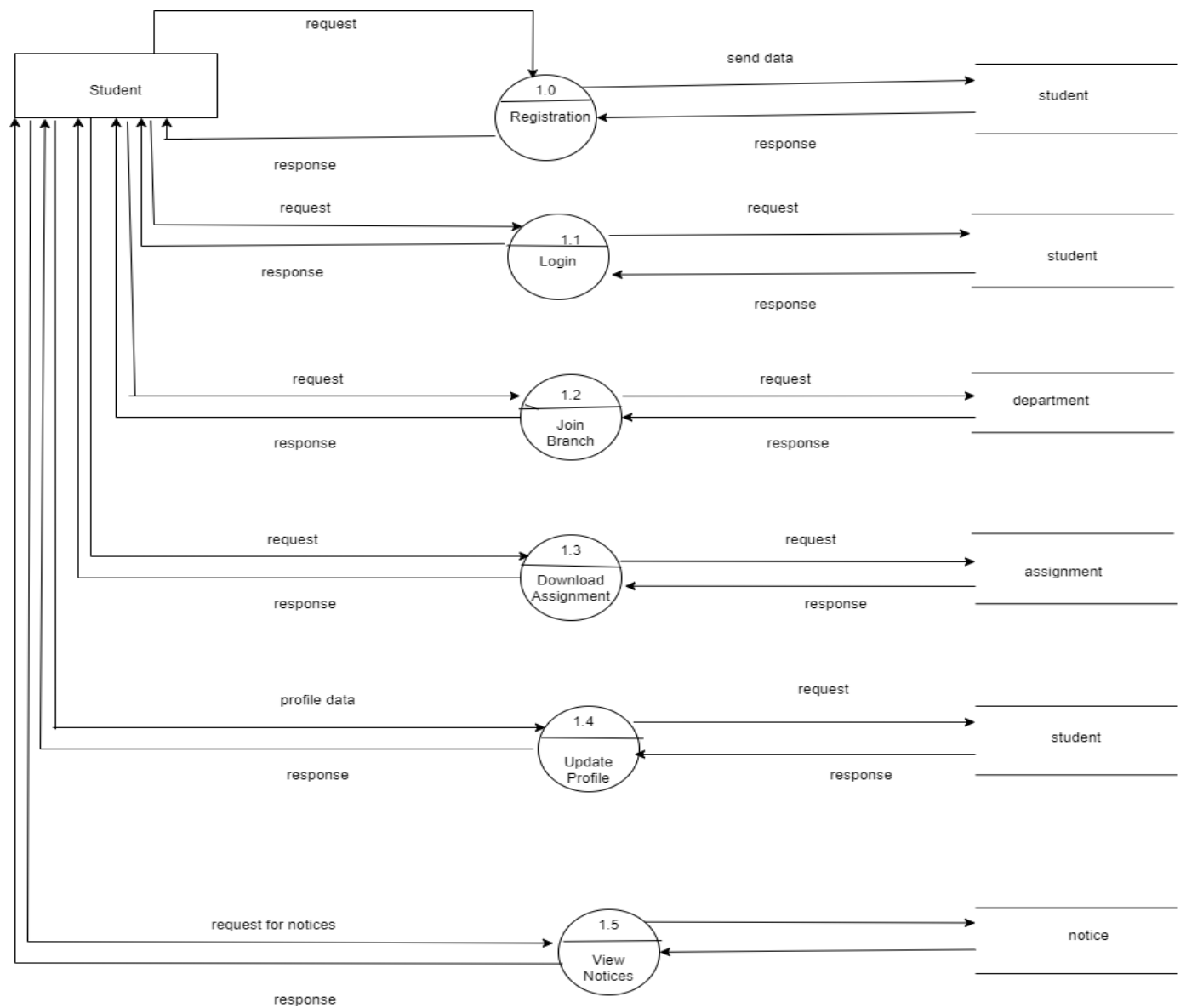
Fig. Level 0 DFD



LEVEL 1 DFD FOR TEACHERS



LEVEL 1 DFD FOR ADMIN



3.2 ENTITY-RELATIONSHIP DIAGRAM:

In software engineering, an **entity–relationship model (ER model)** is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them. However, variants of the idea existed previously, and have been devised subsequently such as super type and subtype data entities and commonality relationships.

An entity–relationship model is a systematic way of describing and defining a business process. The process is modeled as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into zero or more apartments, but one apartment can only be in one building. Entities may have various properties (attributes) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity–relationship diagrams.

An ER model is typically implemented as a database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables point to indexes in other tables; such pointers represent the relationships.

The three schema approach to software engineering uses three levels of ER models that may be developed.

An entity may be defined as a thing capable of an independent existence that can be uniquely identified. An entity is an abstraction from the complexities of a domain. When we speak of an entity, we normally speak of some aspect of the real world that can be distinguished from other aspects of the real world.

A **relationship** captures how entities are related to one another. Relationships can be thought of as verbs, linking two or more nouns.

Cardinality constraints are expressed as follows:

- A double line indicates a participation constraint, totality or subjectivity : all entities in the entity set must participate in at least one relationship in the relationship set;
- An arrow from entity set to relationship set indicates a key constraint, i.e. infectivity: each entity of the entity set can participate in at most one relationship in the relationship set;
- A thick line indicates both, i.e. objectivity: each entity in the entity set is involved in exactly one relationship.
- An underlined name of an attribute indicates that it is a key: two different entities or relationships with this attribute always have different values for this attribute.

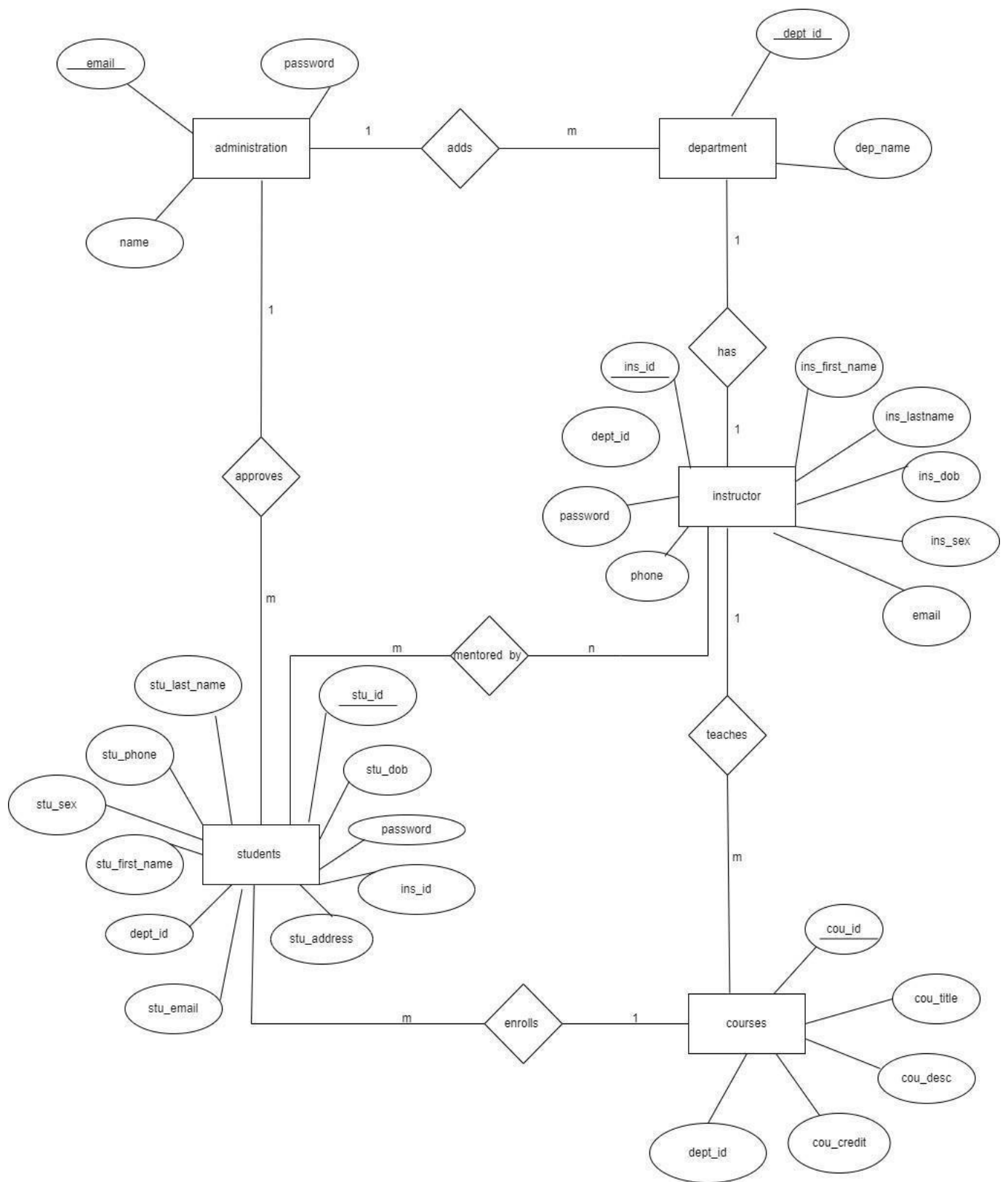


Fig. Entity-Relationship Diagram

3.3 USE-CASE DIAGRAM:

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different **use cases** in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. So only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we must discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. So use case diagrams are consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. So to model the entire system numbers of use case diagrams are used. The purpose of use case diagram is to capture the dynamic aspect of a system. But this definition is too generic to describe the purpose. Because other four diagrams (activity, sequence, collaboration, and State chart) are also having the same purpose. So, we will investigate some specific purpose which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So, when a system is analyzed to gather its functionalities use cases are prepared and actors are identified.

Now when the initial task is complete use case diagrams are modeled to present the outside view. So, in brief, the purposes of use case diagrams can be as follows:

- Used to gather requirements of a system.
- Used to get an outside view of a system.
- Identify external and internal factors influencing the system.
- Show the interacting among the requirements are actors.

How to draw Use Case Diagram?

Use case diagrams are considered for high level requirement analysis of a system.

So, when the requirements of a system are analyzed, the functionalities are captured in use cases. So, we can say that use cases are nothing but the system functionalities written in an organized manner. Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system.

The actors can be human user, some internal applications or may be some external applications. So, in a brief when we are planning to draw a use case diagram, we should have the following items identified.

- Functionalities to be represented as a use case
- Actors
- Relationships among the use cases and actors.

Use case diagrams are drawn to capture the functional requirements of a system.

So, after identifying the above items we must follow the following guidelines to draw an efficient use case diagram.

- The name of a use case is very important. So, the name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.
- Do not try to include all types of relationships. Because the main purpose of the diagram is to identify requirements.
- Use note whenever required to clarify some important points.

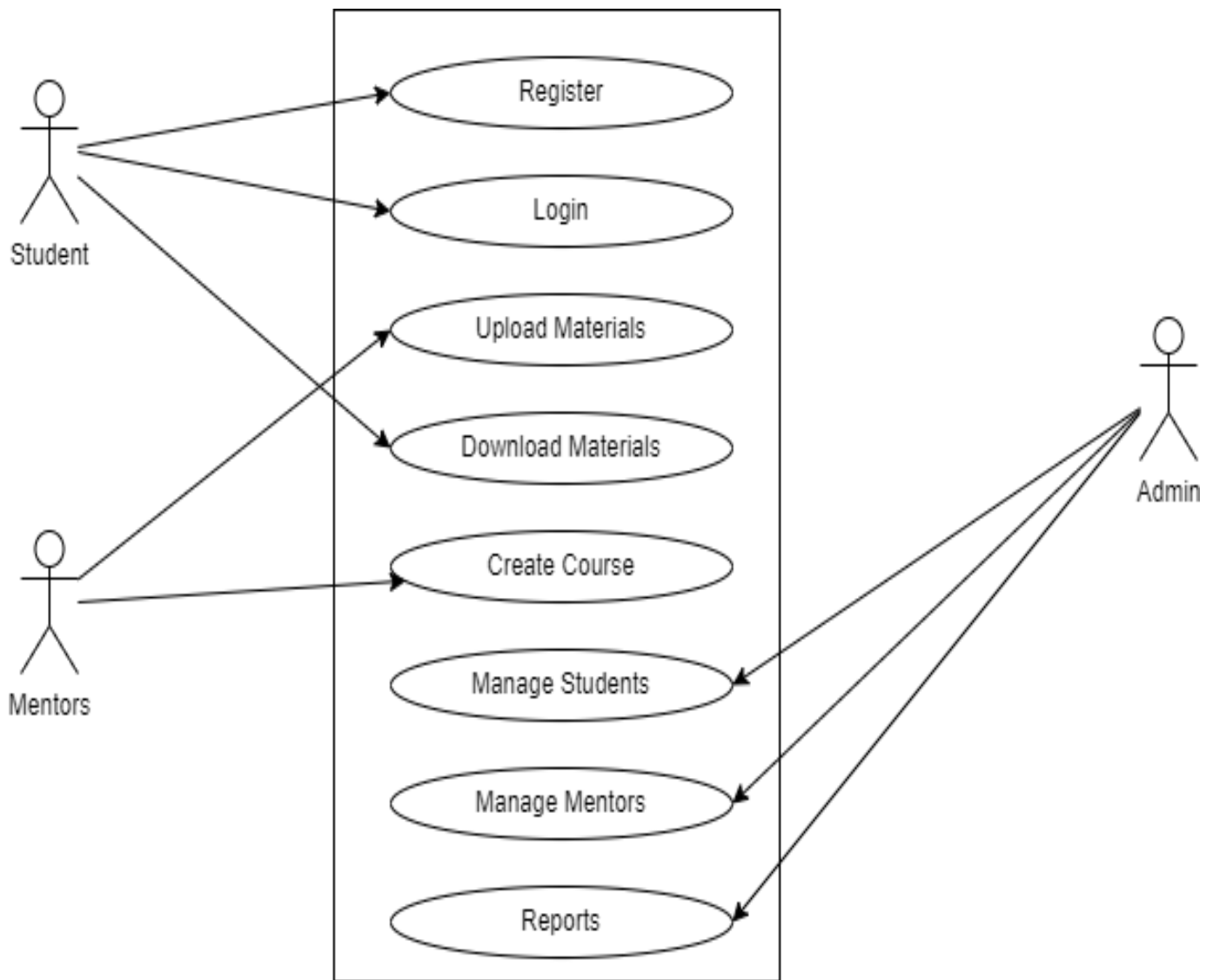


Fig. Use Case Diagram

3.4 CLASS DIAGRAM:

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Purpose of class diagrams:

The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application, however class diagram is a bit different. It is the most popular UML diagram in the coder community.

Where to use class diagrams?

Class diagram is a static diagram and it is used to model the static view of a. The static view describes the vocabulary of the system.

Class diagram is also considered as the foundation for component and deployment diagrams. Class diagrams are not only used to visualize the system static view of the system but they are also used to construct the executable code for forward and reverse engineering of any system.

Generally, UML diagrams are not directly mapped with any object-oriented programming languages but the class diagram is an exception.

Class diagram clearly shows the mapping with object-oriented languages such as Java, C++, etc. From practical experience, class diagram is generally used for construction purpose.

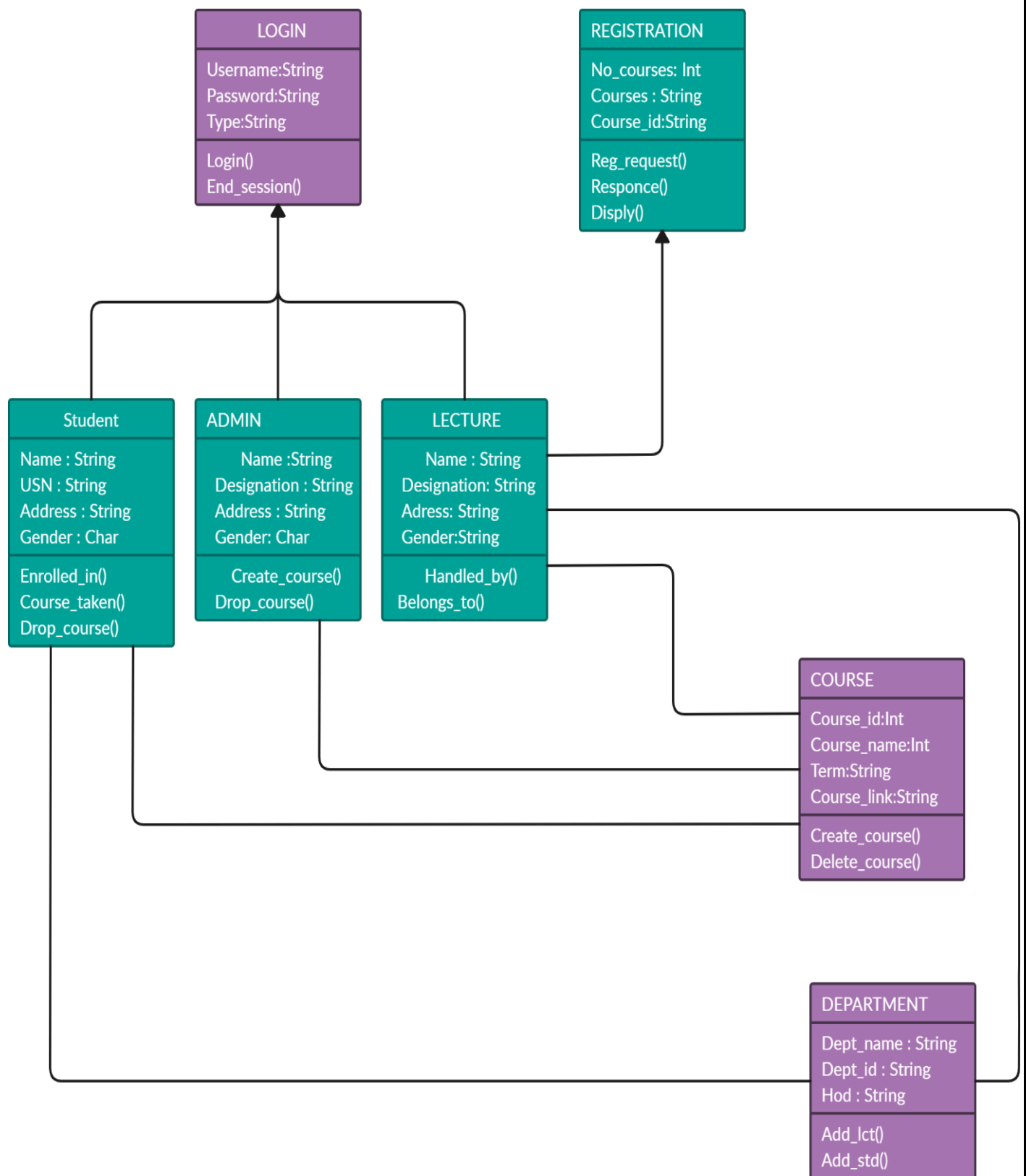


Fig. Class Diagram

3.5 DATA INTEGRITY AND CONSTRAINTS:

- Data integrity is normally enforced in a database system by a series of integrity constraints or rules. Three types of integrity constraints are an inherent part of the relational data model: entity integrity, referential integrity, and domain integrity:
- Entity integrity concerns the concept of a primary key. Entity integrity is an integrity rule which states that every table must have a primary key and that the column or columns chosen to be the primary key should be unique and not null.
- Concerns the concept of a foreign key. The referential integrity rule states that any foreign-key value can only be in one of two states. The usual situation is that the foreign-key value refers to a primary key value of some table in the database. Occasionally, and this will depend on the rules of the data owner, a foreign-key value can be null. In this case we are explicitly saying that either there is no relationship between the objects represented in the database or that this relationship is unknown.
- Domain integrity specifies that all columns in a relational database must be declared upon a defined domain. The primary unit of data in the relational data model is the data item. Such data items are said to be non-decomposable or atomic. A domain is a set of values of the same type.

3.6 DATABASE DESIGN:

A **database** is an organized mechanism that has capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is two level processes. In the first step, user requirements are gathered and a database is designed which will meet these requirements as clearly as possible. This step is called information Level design and it is taken independent of any individual DBMS.

In the following snapshots we display the way we have used MYSQL database for our project and the various entities that have been used along with their table definition and table data.

Administrator Table

Server: 127.0.0.1 » Database: student_portal » Table: administrator

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	EMAIL	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 2	PASSWORD	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 3	NAME	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

☐ Check all
 With selected:
 [Browse](#)
 Change
 Drop
 Primary
 Unique
 Index
 Spatial

Fulltext

Course Table

Server: 127.0.0.1 » Database: student_portal » Table: course

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	COU_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 2	COU_TITLE	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 3	COU_DESC	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 4	COU_CREDIT	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 5	DEP_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

☐ Check all
 With selected:
 [Browse](#)
 Change
 Drop
 Primary
 Unique
 Index
 Spatial

Fulltext

Department table

Server: 127.0.0.1 » Database: student_portal » Table: department

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	dep_id	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 2	dep_name	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

☐ Check all
 With selected: [Browse](#) Change Drop Primary Unique Index Spatial

[Fulltext](#)

Grade Table

Server: 127.0.0.1 » Database: student_portal » Table: grade

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	GRADE	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 2	STU_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 3	CLA_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

☐ Check all
 With selected: [Browse](#) Change Drop Primary Unique Index Spatial

Instructor Table

Server: 127.0.0.1 » Database: student_portal » Table: instructor

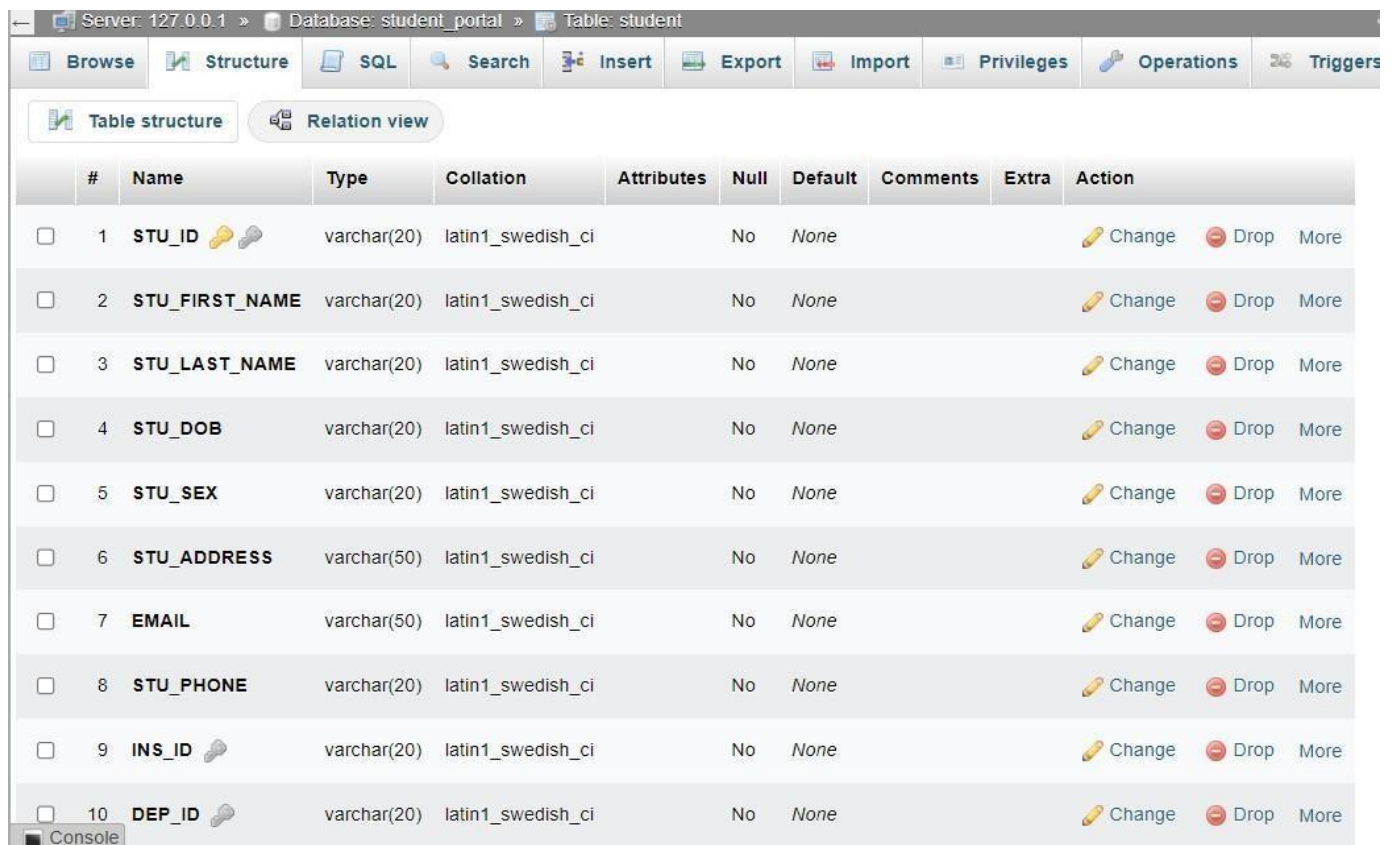
[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	INS_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 2	INS_FIRST_NAME	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 3	INS_LAST_NAME	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 4	INS_DOB	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 5	INS_SEX	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 6	INS_ADDRESS	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 7	EMAIL	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 8	DEP_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 9	PASSWORD	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/> 10	INS_PHONE	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

[Console](#)

Student Table



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	STU_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
2	STU_FIRST_NAME	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
3	STU_LAST_NAME	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
4	STU_DOB	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
5	STU_SEX	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
6	STU_ADDRESS	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
7	EMAIL	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
8	STU_PHONE	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
9	INS_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
10	DEP_ID	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

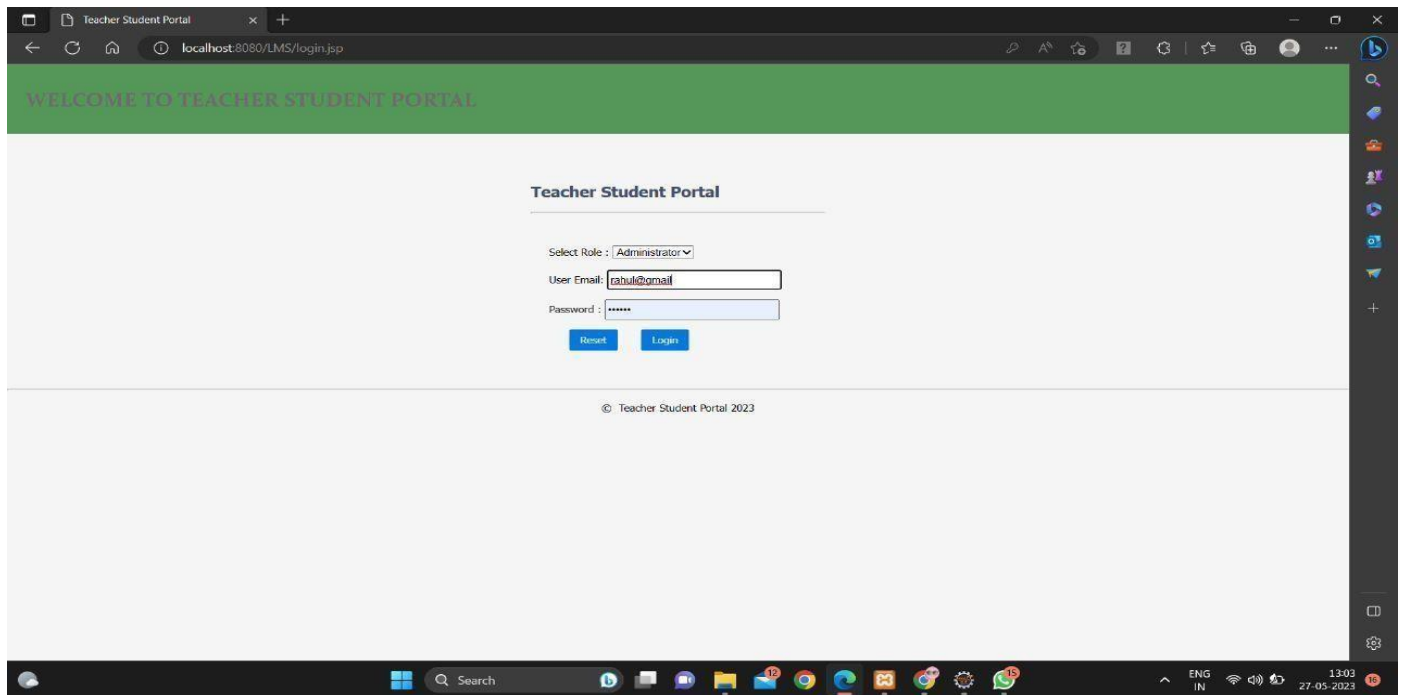
3.7 USER INTERFACE DESIGN:

It is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user centered design).

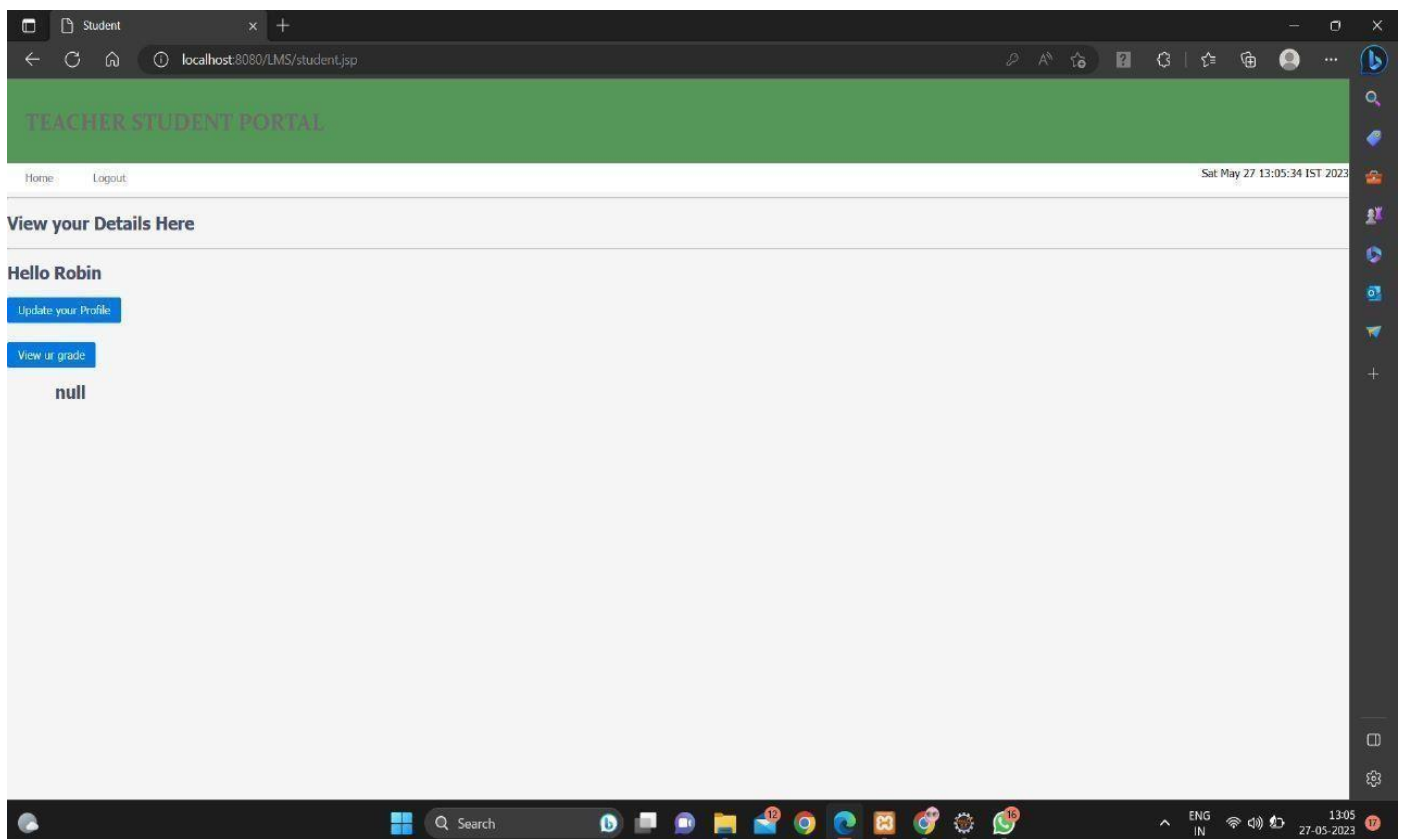
Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. Graphic design and typography are utilized to support its usability, influencing how the user performs certain interactions and improving the aesthetic appeal of the design; design aesthetics may enhance or detract from the ability of users to use the functions of the interface. The design process must balance technical functionality and visual elements (e.g., mental model) to create a system that is not only operational but also usable and adaptable to changing user needs.

Interface design is involved in a wide range of projects from computer systems, to cars, to commercial planes; all of these projects involve much of the same basic human interactions yet also require some unique skills and knowledge. As a result, designers tend to specialize in certain types of projects and have skills centered on their expertise, whether that be software design, user research, web design, or industrial design.

Home Page



Student Portal



Administrator Dashboard

TEACHER STUDENT PORTAL

Hello admin

Instructor Student Course Department Logout

Sat May 27 13:56:25 IST 2023

Instructor Details

Instructor ID :
First Name :
Last Name :
D.O.B :
Sex :
Address :
Email-id :
Department-Id:
Password :
Phone No :

Student

Student ID :
First Name :
Last Name :
D.O.B :
Sex :
Address :
Email-id :
Phone No :
Department-Id:

Course

Course ID :
Course Title :
Course Desc :
Course Credit :
Department ID:

Department

Department ID :
Department Name :

6. CODING

LOGIN PAGE HTML:

```
1  <%@page contentType="text/html" pageEncoding="UTF-8"%>
2  <!DOCTYPE html>
3  <html>
4  <head>
5      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
6      <title>Teacher Student Portal</title>
7      <link href="CSS.css" rel="stylesheet" type="text/css"/>
8      <link href="pure/pure-min.css" rel="stylesheet">
9
10
11     <script type="text/javascript">
12         function validateform()
13         {
14             var name = document.loginform.user_name.value;
15             var password = document.loginform.user_pass.value;
16
17             if (name === null || "") {
18                 alert("Username cannot be left Blank");
19                 form=document.focus;
20                 return false;
21             } else if (password === null || "") {
22                 alert("Please Enter Password");
23                 return false;
24             }
25         }
26     </script>
```

```

53:         <p>
54:             <label>Password :
55:                 <input type="password" name="password" id="user_pass" class="input" value="" size="25" style="height:
56:         </p>
57:         <p>
58:             &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input type="reset" value="Reset" class="pure-button pure-button-primary"/>
59:             &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input type="submit" value="Login" onclick="validateform();" class="pure-but
60:         </p>
61:     </form>
62: </div>
63: <hr>
64: <p align="center">&COPY; &nbsp;&nbsp;&Teacher Student Portal 2023</p>
65: </body>
66: </html>

```

`<div class="users" name="form">`

```

40
41 .btn
42 {
43   display: inline-block;
44   color: #fff;
45   text-align: center;
46   background: #545759;
47   padding: 5px 30px 5px;
48   margin-top: 0px;
49   text-transform: none;
50   font-family: Tahoma;
51   font-size: 12px;
52 }
53
54 table.hoverTable {
55   font-family: verdana,arial,sans-serif;
56   font-size: 11px;
57   color: #333333;
58   border-width: 1px;
59   border-color: #999999;
60   border-collapse: collapse;
61 }
62
63 table.hoverTable th {
64   background-color: #c3dde0;
65   border-width: 1px;
66   padding: 8px;
67   border-style: solid;
68   border-color: #a9c6c9;
69 }
70
71 table.hoverTable tr {
72   background-color: #d4e3e5;
73 }
74
75 table.hoverTable td {
76   border-width: 1px;
77   padding: 8px;
78   border-style: solid;
79   border-color: #a9c6c9;
80 }
81

```

```

1
2 .hoverTable{
3   width:100%;
4   border-collapse:collapse;
5 }
6 .hoverTable td{
7   padding:7px; border: #4e95f4 1px solid;
8 }
9 /* Define the default color for all the table rows */
10 .hoverTable tr{
11   background: #3399ff;
12 }
13 /* Define the hover highlight color for the table row */
14 .hoverTable tr:hover {
15   background-color: #E65C00;
16 }
17
18 th{
19   background: #00FF00;
20 }
21

```

7. CONCLUSION

This project has been appreciated by all the users in the organization. This project has been appreciated by all the users in the organization. It is easy to use, since it uses the GUI provided in the user dialog. User friendly screens are provided. The usage of software increases the efficiency, decreases the effort. It has been efficiently employed as a Site management mechanism. It has been thoroughly tested and implemented.

E-learning is not just a change of technology. It is part of a redefinition of how we as a species transmit knowledge, skills, and values to younger generations of workers and students. This book makes a few predictions of how e-learning and the functions it serves will continue to develop. Learners will have access to millions or billions of

knowledge modules. Some will be Web pages with simple text and graphics. Others may include multimedia simulations. In many fields, e-learning has become the default way to conduct training or to provide education. There are four secrets of e-learning. The first secret is to teach what learners need to learn in the way they most naturally learn. The second secret is to define clear learning objectives. The third secret builds on the first two. It is to focus on the right objectives. The final secret is in the power of testing.

8. FUTURE SCOPE & FURTHER ENHANCEMENT

In future, we would like to keep working on this project and make new additions to provide users with more advanced features and more detailed information. We have set our sights on the following additions in future –

- People expect to be able to work, learn, and study whenever and wherever they want to. The technologies we use are increasingly cloud based and exceptions of IT decentralized support. The world of work is increasingly collaborative, driving changes in the structure of student projects.
- Abundance of resources and relationships are easily accessible via the internet. Education paradigms are shifting to include online learning, hybrid learning, and collaborative models. There is a new emphasis in the classroom and more challenge-based and active learning.

9. BIBLIOGRAPHY

1. <https://www.w3schools.com>
2. <https://www.slideshare.com>
3. <https://www.scribd.com>
4. <https://www.tutorialspoint.com>
5. <https://www.youtube.com>