COURSE MANAGEMENT SYSTEM

Project Report Submitted by

ATHIRA BIJU

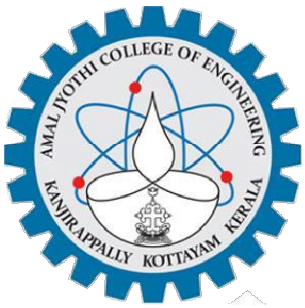
Reg. No.: AJC21MCA-2044

In Partial fulfillment for the Award of the Degree Of

MASTER OF COMPUTER APPLICATIONS

(MCA TWO YEAR)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY



AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala – 686518]

2022-2023

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

CERTIFICATE

This is to certify that the Project report, **"COURSE MANAGEMENT SYSTEM"** is the bona fide work of **ATHIRA BIJU (Regno: AJC21MCA-2044)** in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2022-23.

Ms Sruthimol Kurian
Internal Guide

Meera Rose Mathew
Coordinator

Rev. Fr. Dr. Rubin Thottupurathu Jose

Head of the Department

DECLARATION

I hereby declare that the project report **"COURSE MANAGEMENT SYSTEM"** is a bona fide work done at Amal Jyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2022-2023.

Date: 26/11/2022 **ATHIRA BIJU**

KANJIRAPPALLY **Reg: AJC21MCA-2044**

ACKNOWLEDGEMENT

First and foremost, I thank God almighty for his eternal love and protection throughout the project. I take this opportunity to express my gratitude to all who helped me in completing this project successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our Manager **Rev. Fr. Dr. Mathew Paikatt** and Principal **Dr. Lillykutty Jacob** for providing good faculty for guidance.

I owe a great depth of gratitude towards our Head of the Department **Rev.Fr.Dr. Rubin Thottupurathu Jose** for helping us. I extend my whole hearted thanks to the project coordinator **Meera Rose Mathew** for her valuable suggestions and for overwhelming concern and guidance from the beginning to the end of the project. I would also express sincere gratitude to my guide **Ms Sruthimol Kurian** for her inspiration and helping hand.

I thank our beloved teachers for their cooperation and suggestions that helped me throughout the project. I express my thanks to all my friends and classmates for their interest, dedication, and encouragement shown towards the project. I convey my hearty thanks to my family for the moral support, suggestions, and encouragement to make this venture a success.

ATHIRA BIJU

ABSTRACT

.

COURSE MANAGEMENT SYSTEM

The COURSE MANAGEMENT is a Learning Management System (LMS) based web application intended for online users. In this busy world searching of tutors for any subject is a very difficult. In order to make it simple here with we proposed an idea to find a subject expert tutor through web application. The proposed work has the common platform where the tutor and the student can access on their respective available views. In this application students can register and view the availability of the subject expert based on rating of the tutor. Teachers can also get a student just by logging onto the website and setting up the profile. In the course finder system, there are three entities namely, Admin ,Tutor and student. Admin can login, manage tutor new teachers. Admin can also check for the registered students and add Students. Student can register and login, tutors can be viewed by students. students can filter and select the tutor by ratings, they can book the tutor online, rate the tutor. After registration students can view video tutorial, live sessions and view the E-Books. The tutor can login by using credentials that will be provided by mail. They can also check the booking done. They need to set their profile. This course finder system can help the tutors to get students to find the best tutors for their children.

CONTENT

List of Abbreviation

IDE - Integrated Development Environment
HTML - Hyper Text Markup Language.
CSS - Cascading Style Sheet
SQL - Structured Query Language
UML - Unified Modeling Language
PHP – Hyper Text Preprocessor

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

The COURSE MANAGEMENT is a Learning Management System (LMS) based web application intended for online users. In this busy world searching of tutors for any subject is a very difficult. In order to make it simple here with we proposed an idea to find a subject expert tutor through web application. The proposed work has the common platform where the tutor and the student can access on their respective available views. In this application students can register and view the availability of the subject expert based on rating of the tutor. Teachers can also get a student just by logging onto the website and setting up the profile. In the course finder system, there are three entities namely, Admin, Tutor and student. Admin can login, manage tutor new teachers. Admin can also check for the registered students and add Students. Student can register and login, tutors can be viewed by students. Students can filter and select the tutor by ratings, they can book the tutor online, rate the tutor. After registration students can view video tutorial, live sessions and view the E-Books. The tutor can login by using credentials that will be provided by mail. They can also check the booking done. They need to set their profile. This course finder system can help the tutors to get students to find the best tutors for their children.

1.2 PROJECT SPECIFICATION

The purpose of "Course Management" application is to assist the students to find tutors in an interactive manner. It aims to complement the efforts of a student to find a desirable teacher. There are total two stakeholders they are Admin, Tutor and Student.

- Admin

Student and Instructor management. Able to add, remove, view the students enrolled for the course and approve instructors that newly registered to the course. Can view approved teachers, students and which are the student is assigned to which teacher and their scheduling etc.

- Student

Can enroll for a course that they wish to join and view lectures that are uploaded by the instructor for them. Attend live sessions scheduled with the tutor. Receives notifications, update profile, change password, can view class schedules. Attend the exams and assignments.

- Tutor

Uploads the referential videos, podcasts for the student, upload links to meeting and timetable for that particular student or group classes, communicates with the students, can view how many students enrolled for that particular instructor, Update profile, change password.

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

System analysis is the process of acquiring and analyzing data, diagnosing issues, and using data to suggest system changes. The system users and system developers must extensively communicate during this problem-solving process. A system analysis or research should be the first step in any system development process. The system is meticulously examined and assessed. The system analyst takes on the role of an interrogator and investigates the operation of the current system in great detail. The system's input is identified, and the system as a whole is viewed. The various procedures can be linked to the outputs from the organizations. the Understanding issue, identifying the pertinent and important variables, evaluating and synthesizing the many elements, and selecting the best or, at the very least, most acceptable course of action is all part of system analysis.

The process must be thoroughly studied using a variety of methodologies, including questionnaires and interviews. To reach a conclusion, the information gathered by these sources must be carefully examined. Understanding how the system works is the conclusion. The current system is the name of this system. Now, the current system is carefully examined, and issue areas are found. Now the designer takes on the role of a problem-solver and attempts to fix the problems the company is experiencing. The solutions are replaced with proposals. The best option is then selected after an analytical comparison of the proposal and the existing system. The user is given the opportunity to approve or reject the suggestion. On user request, the proposal is assessed and appropriate revisions are made. As soon as the user is content with the suggestion, this loop breaks.

The process of acquiring and analyzing data in order to use it for future system studies known as preliminary study. Initial research is a problem-solving activity that necessitates close coordination between system users and developers. It conducts a number of feasibility studies. These investigations provide an approximate estimate of the system activity, which can be used to determine the tactics to be used for an efficient system research and analysis.

2.2 EXISTING SYSTEM

Existing system is not a fully automated system. The existing system have the disadvantages such as time consuming and expensive. It is necessary to modify the existing system in order to include additional information and make the system efficient, flexible and secure**.**

2.2.1 NATURAL SYSTEM STUDIED

In this system, the user who wish to join the academy has to everything manually. This site contains only the details of classes conducted and its corresponding course fee structure. The clients can enquire about the classes through WhatsApp or directly contact them through phone calls. For those people who face travelling problems and health issues, it is very difficult for them to come and attend the lessons at the right time.

2.2.2 DESIGNED SYSTEM STUDIED

@Udemy.com

Taking online courses is one of the easiest and cheapest ways to learn new skills. There are a lot of websites and platforms where you can find courses, but they're not all created equally. Some are really good and others are really, really bad. Udemy is one of the biggest online course platforms around. Udemy is a fantastic option for casual learners because you can find courses about anything. Anyone can create a course though, so quality isn't always the greatest. Udemy also doesn't offer recognized certificates like other education websites.

2.3 DRAWBACKS OF EXISTING SYSTEM

- No proper recording of registration, booking etc.
- Payment's calculations are done manually.
- Information redundancy.
- Time consuming.
- High cost
- The tutor may not be certified

2.4 PROPOSED SYSTEM

In the proposed system we are using the Learning Management System to create interactive sessions with the user. He/ She can watch the recordings of the classes according to their convenience. User can enroll for the courses easily and the client can access recorded classes anywhere through any devices. It aims to complement the efforts of an student to find desirable teacher.

2.5 ADVANTAGES OF PROPOSED SYSTEM

- allows 24*7 sessions
- Affordable fees.
- Save time.
- Through discussion boards and chats, you are able to interact with everyone online and also clear your doubts if any.

CHAPTER 3
REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

A feasibility study is conducted to determine whether the project will, upon completion, fulfil the objectives of the organization in relation to the work, effort, and time invested in it. A feasibility study enables the developer to predict the project's usefulness and potential future. A system proposes the workability, which includes the influence on the organization, capacity to satisfy user needs, and efficient use of resources, is the basis for a feasibility study. As a result, before a new application is accepted for development, it often undergoes a feasibility assessment**.**

The document outlines the project's viability and contains a number of factors that were carefully taken into account throughout this project's feasibility study, including its technical, economic, and operational viabilities. It has the following characteristics: -

3.1.1 Economical Feasibility

Cost and benefit analyses are required to support the emerging system. Criteria to make sure that focus is placed on the project that will yield the best results the earliest. The price that would be involved in developing a new system is one of the variables.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

The proposed system was created as part of a project; hence, there are no manual expenses associated with it. Additionally, the fact that all of the resources are already at hand indicates that the system may be developed affordably.

3.1.2 Technical Feasibility

The system needs to be assessed first from a technical standpoint. An overview design of the system's requirements in terms of input, output, programmers, and procedures must serve as the foundation for the assessment of this viability. The inquiry must next advise the kind of equipment, necessary procedure for building the system, and means of once it has been created, using the system after having identified an outline system.

The following technical difficulties came up throughout the investigation:

- Is the proposed technology compatible with the current technology?
- Can the system grow if it is improved?

The project uses cryptographic methods and calls for a high-resolution scanning device. The system may still be used even though the technology may become outdated after a while because a newer version of the same software still works with an earlier version. Therefore, this project only has a few limitations. The system was created using Python Django for the front end and a PostgreSQL server for the back end; it is technically feasible to complete the project. The system was created using Python Django for the front end and a PostgreSQL server for the back end; it is technically feasible to complete the project. The system was also well-performing, with an Intel i5 core processor, 8 GB of RAM, and a 1 TB hard drive.

3.1.3 Behavioral Feasibility

The proposed system includes the following questions:

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

The project would be advantageous because, when created and implemented, it would achieve the goals. The project is deemed to be behaviorally feasible after carefully weighing all behavioral factors.

3.1.4 Questionnaire

1. How students access the course materials.?

   Through the Google classroom platform

2. How to schedule online classes and how the class schedule is communicated to the students.?

   Using Online video conferencing platform and share the link through whatsapp

3. How to take an online exam.?

   Questions are commonly shared through Online Platform

4. How the students are informed of updates to their marks.?

Through theOnline platform

5. How should students submit their exam answers.?

   Answers are submitted through Online Platform

6. How may a student be added to a class.?

Add each student to the class register.

7. How many different subjects can a teacher teach.?

   One/Two subject

8. How to take assignments.?

   Questions and Answers are commonly shared through Online Platform

9. what is your current payment status?

   through net banking

10. Is that online / offline is better?

    Online

3.2 SYSTEM SPECIFICATION

3.2.1 Hardware Specification

Processor - Intel i5 processer

RAM - 8 G B

Hard disk - 1 T B

3.2.2 Software Specification

Front End - HTML, CSS

Backend - MYSQL

Client on PC - Windows 7 and above.

Technologies used - JS, HTML5, J Query, PHP, CSS

3.3 SOFTWARE DESCRIPTION

3.3.1 PHP

In addition to being used as a general-purpose programming language, PHP is a server-side scripting language created for web development. Currently, PHP is present on2.1million web servers and more than 244 million websites. The reference implementation of PHP, which was first developed by Rasmus Ledorf in 1995, is now created by the PHP group. The recursive acronym PHP: Hyper text Preprocessor has replaced the original meaning of PHP, which was personal Home page. A PHP processor module on a web server interprets PHP code to produce the final web page.PHP commands can be embedded directly into a HTML source document rather than calling an external file to process data. It has also grown to include a command-line interface and can be used in stand alone mode, which is incompatible with the GNU General Public License (GPL) due to restrictions on the use of the term PHP.PHP can be deployed on most web server and also as a standalone shell on almost every operating system and platform, free of charge

3.3.2 MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Website provides the latest information about MySQL software.

• **MySQL is a database management system**.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporatenet work. To add, access, and process data stored in a computer database, database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

• **MySQL databases are relational**.

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and "pointers" between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data. The SQL part of "MySQL" stands for "Structured Query Language". SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax. SQL is defined by the ANSI/ISOSQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, "SQL92" refers to the standard released in1992, "SQL: 1999" refers to the standard released in 1999, and "SQL: 2003" refers to the current version of the standard. We use the phrase "the SQL standard" tomean the current version of the SQL Standard at any time.

- **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), toss define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL or need to embed

MySQL code into a commercial application, you can buy a commercially licensed version from us. Seethe MySQL Licensing Overview for more information.

• **The MySQL Database Server is very fast, reliable, scalable, and easy to use**

. You ought to give it a shot if that is what you're after. In addition to your other apps, web servers, and other software, MySQL Server can function smoothly on a desktop or laptop while requiring little to no maintenance. You can modify the settings to utilize all the RAM, CPU power, and I/O capacity if you dedicate an entire machine to MySQL. MySQL Server works in client/server or embedded systems. The MySQL Database Software is a client/server system made up of a multithreaded SQL server that supports several client program and libraries, administration tools, and a broad range of application programming interfaces(APIs). Additionally, we provide MySQL Server as an integrated multi-threaded library that you can link into your program to create a standalone offering that is smaller, faster, and simpler to operate.

CHAPTER 4
SYSTEM DESIGN

4.1 INTRODUCTION

Any engineered system or product's development process begins with design. A creative process is design. The secret to an efficient system is a decent design. The process of using different methodologies and concepts to specify a process or a systemin enough detail to allow for its physical implementation is referred to as "design." One way to describe it is as the process of using different methodologies and concepts to specify a device, a process, or a system in enough detail to allow for its physical reality. Regardless of the development paradigm that is employed, software design forms the technical core of the software engineering process. The architectural detail needed to construct a system or product is developed through the system design. This programmed has also through the best possible design phase, fine tuning all efficiency, performance, and accuracy levels, as in the case of any systematic technique. A user-oriented document is converted into a document for programmers or database staff throughout the design phase. The two stages of system design development are logical design and physical design.

4.2 UML DIAGRAM

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by the Object Management Group(OMG)and UML 1.0 specification draft was proposed to the OMG in January 1997. UML stands for Unified Modeling Language. UML is different from the other common programming languages such as C++, Java, COBOL, etc. UML is a pictorial language used to make software blueprints. UML can be described as a general purpose visual modeling language to visualize, specify, construct, and document software system. Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flow in a manufacturing unit, etc. UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object oriented analysis and design. After some standardization, UML has become an OMG standard. All the elements, relationships are used to make a complete UML diagram and the diagram represents a system. The visual effect of the UML diagram is the most important part of the entire process. All the other elements are used to make it complete. UML includes the following nine diagrams

- Class diagram
- Object diagram
- Use case diagram
- Sequence diagram
- Collaboration diagram
- Activity diagram
- State chart diagram
- Deployment diagram
- Component diagram

4.2.1 USE CASE DIAGRAM

A use case diagram is a visual representation of the interactions between system components. A approach for identifying, outlining, and organizing system requirements is called a use case. The word "system" here refers to a thing that is being created or run, like a website for mail-order product sales and services. UML (Unified Modeling Language), a standard language for the modelling of real-world objects and systems, uses use case diagrams.

The planning of general requirements, the validation of a hardware design, the testing and debugging of a software product in development, the creation of an online help reference, or the completion of a job focused on customer support are all examples of system objectives. For instance, use cases in a product sales context can involve customer service, item ordering, catalogue updating, and payment processing. There are four elements in a use case diagram.

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which are the specific roles are played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

Use case diagrams are drawn to capture the functional requirements of a system. After identifying the above items, we have to use the following guidelines to draw an efficient use case diagram.

- The name of a use case is very important. 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, as the main purpose of the diagram is to identify the requirements.
- Use notes whenever required to clarify some important points**.**

Use-Case Diagram

4.2.1 SEQUENCE DIAGRAM

A sequence diagram essentially shows how things interact with one another sequentially, or the order in which these interactions occur. A sequence diagram can also be referred to as event diagrams or event scenarios. Sequence diagrams show the actions taken by the components of a system in chronological order. Businesspeople and software engineers frequently use these diagrams to record and comprehend the requirements for new and current systems.

Sequence Diagram Notations –

- **Actors** – In a UML diagram, an actor represents a particular kind of role that interacts with the system and its objects. An actor is always beyond the purview of the system that we want to use the UML diagram to represent. We employ actors to portray a variety of roles, including those of human users and other outside subjects. In a UML diagram, an actor is represented using a stick person notation. In a sequence diagram, there might be several actors.
- **Lifelines** – A lifeline is a named element in a sequence diagram that represent an individual participant. So, in a sequence diagram, each incident is represented by a lifeline. A sequence diagram's lifeline elements are at the top.
- **Messages** – Messages are used to show how objects communicate with one another. The messages are displayed on the lifeline in chronological sequence. Arrows are how messages are represented. A sequence diagram's main components are lifelines and messages.

Messages can be broadly classified into the following categories:

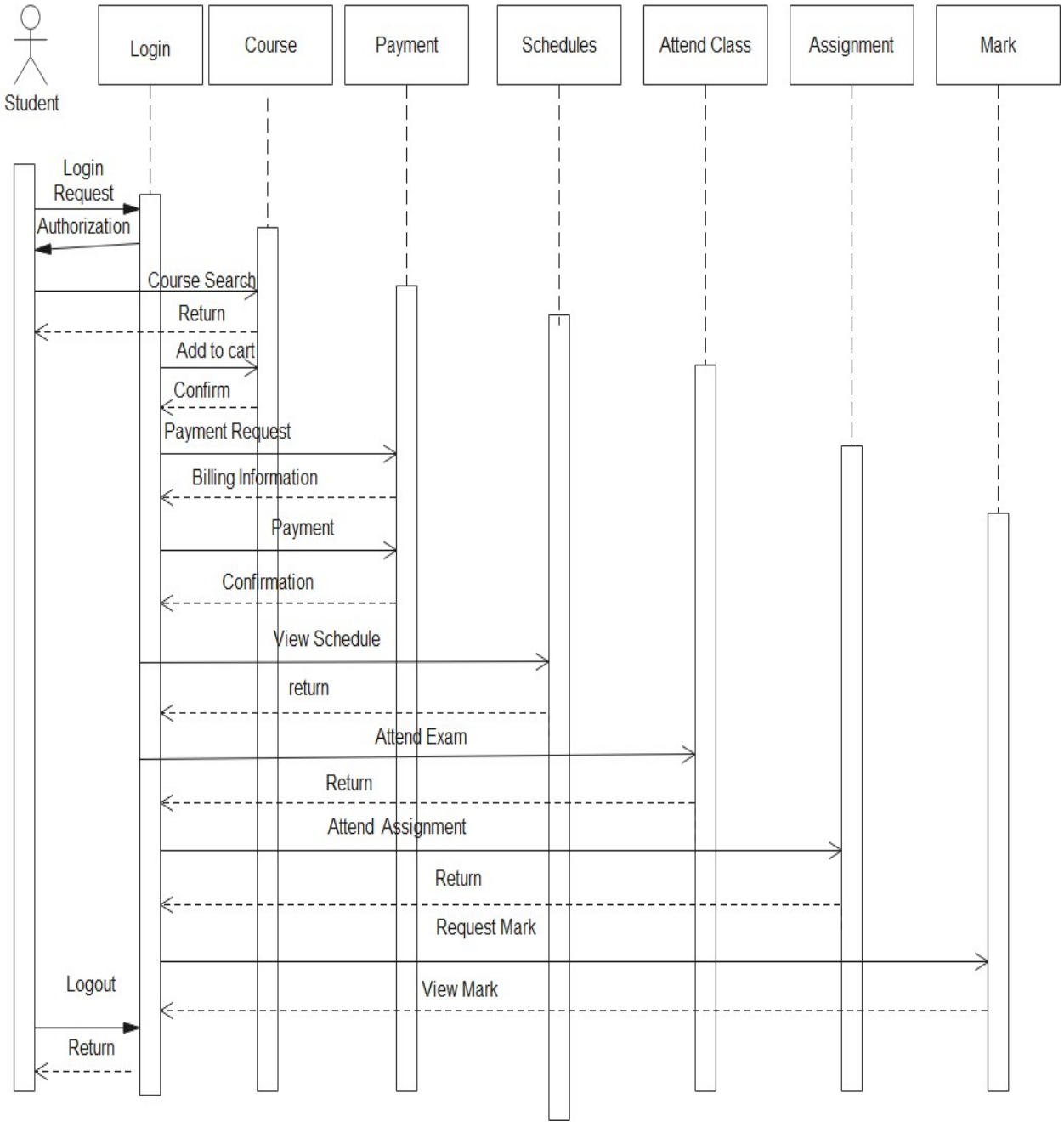
- Synchronous messages
- Asynchronous Messages
- Create message
- Delete Message
- Self-Message
- Reply Message
- Found Message
- Lost Message

Guards – To model conditions we use guards in UML. They are used when we need to restrict the flow of messages on the pretext of a condition being met. Guards play an important role in letting software developers know the constraints attached to a system or a particular process.

Uses of sequence diagrams –

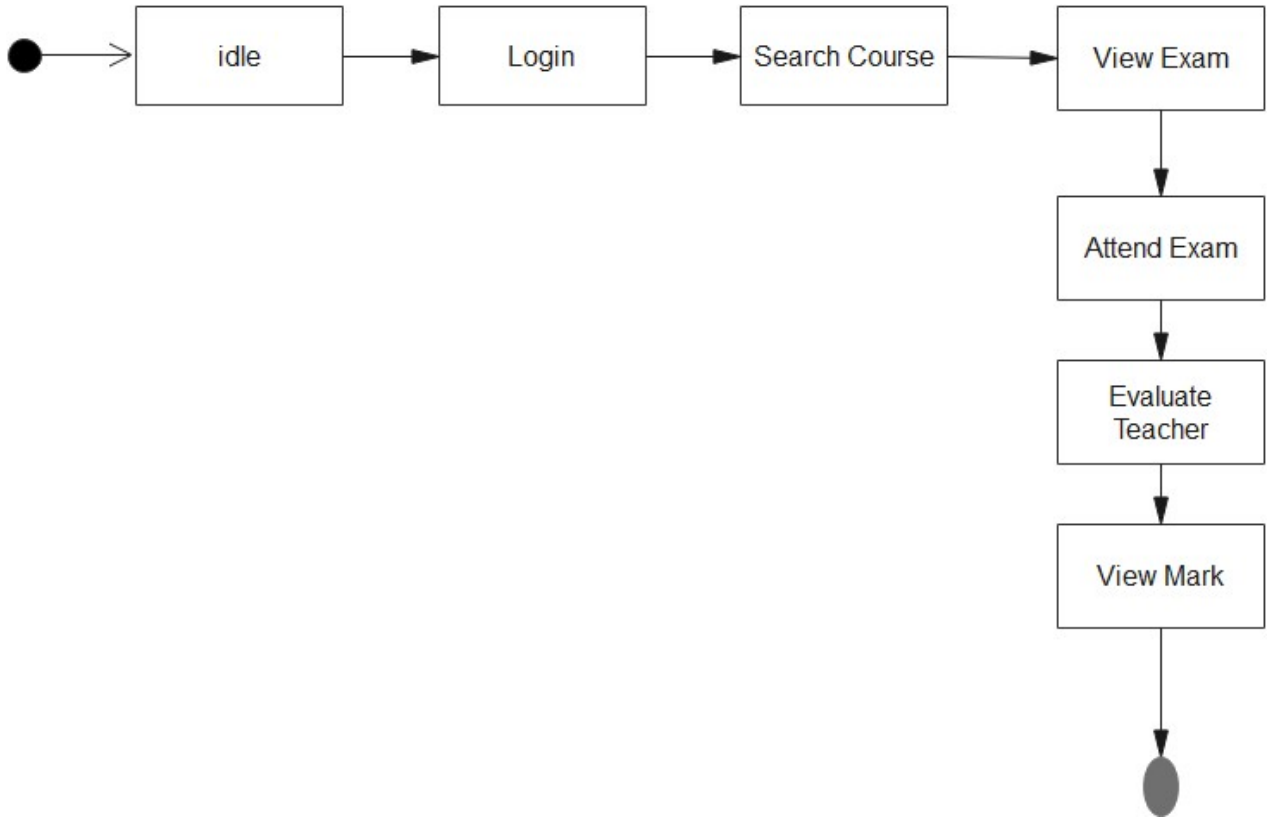
- Used to model and visualize the logic behind a sophisticated function, operation or procedure.
- They are also used to show details of UML use case diagrams.
- Used to understand the detailed functionality of current or future systems.
- Visualize how messages and tasks move between objects or components in a system.

Sequence Diagram



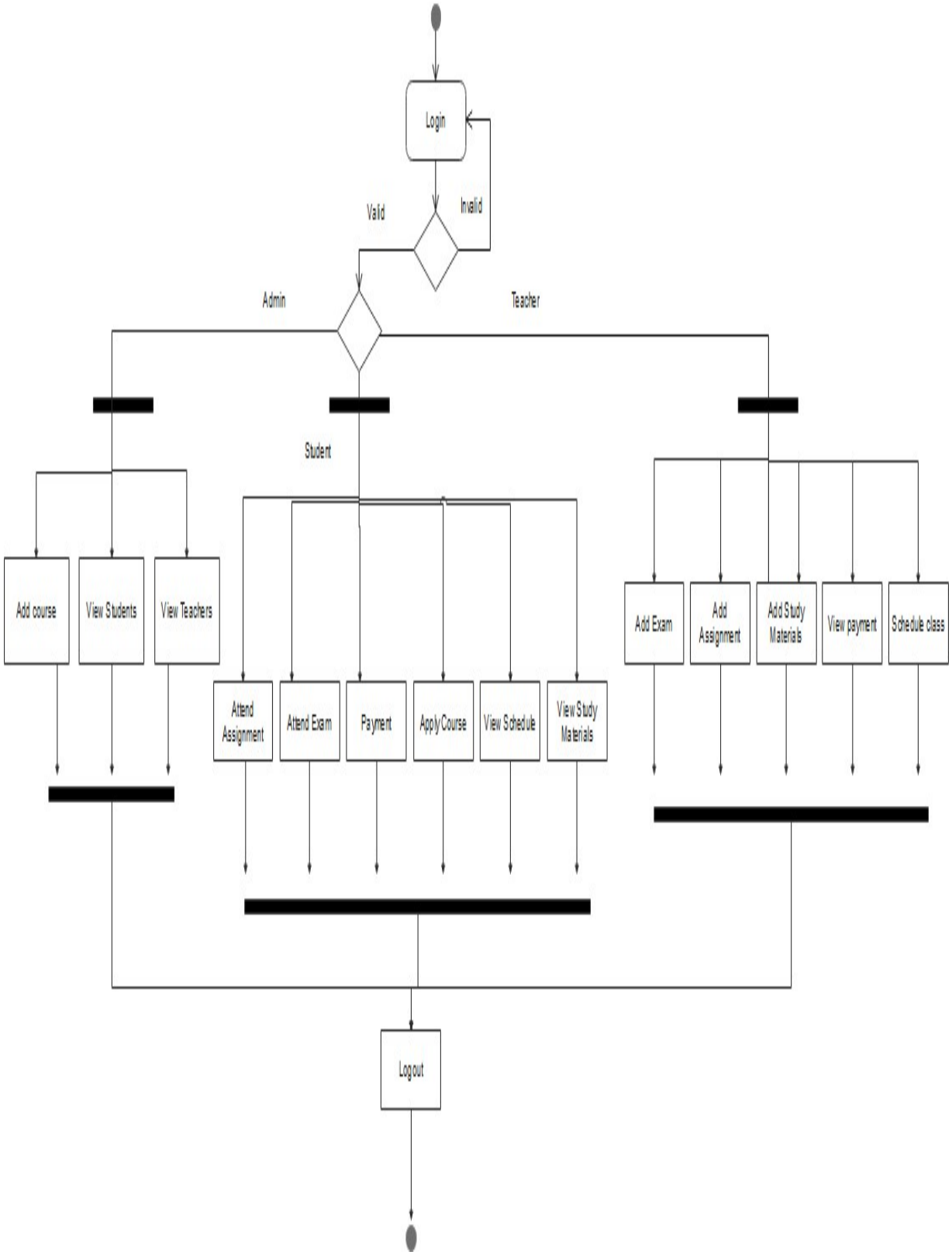
4.2.2 State Chart Diagram

A state diagram, also known as a state machine diagram or state chart diagram, is an illustration of the states an object can attain as well as the transitions between those states in the Unified Modeling Language (UML).



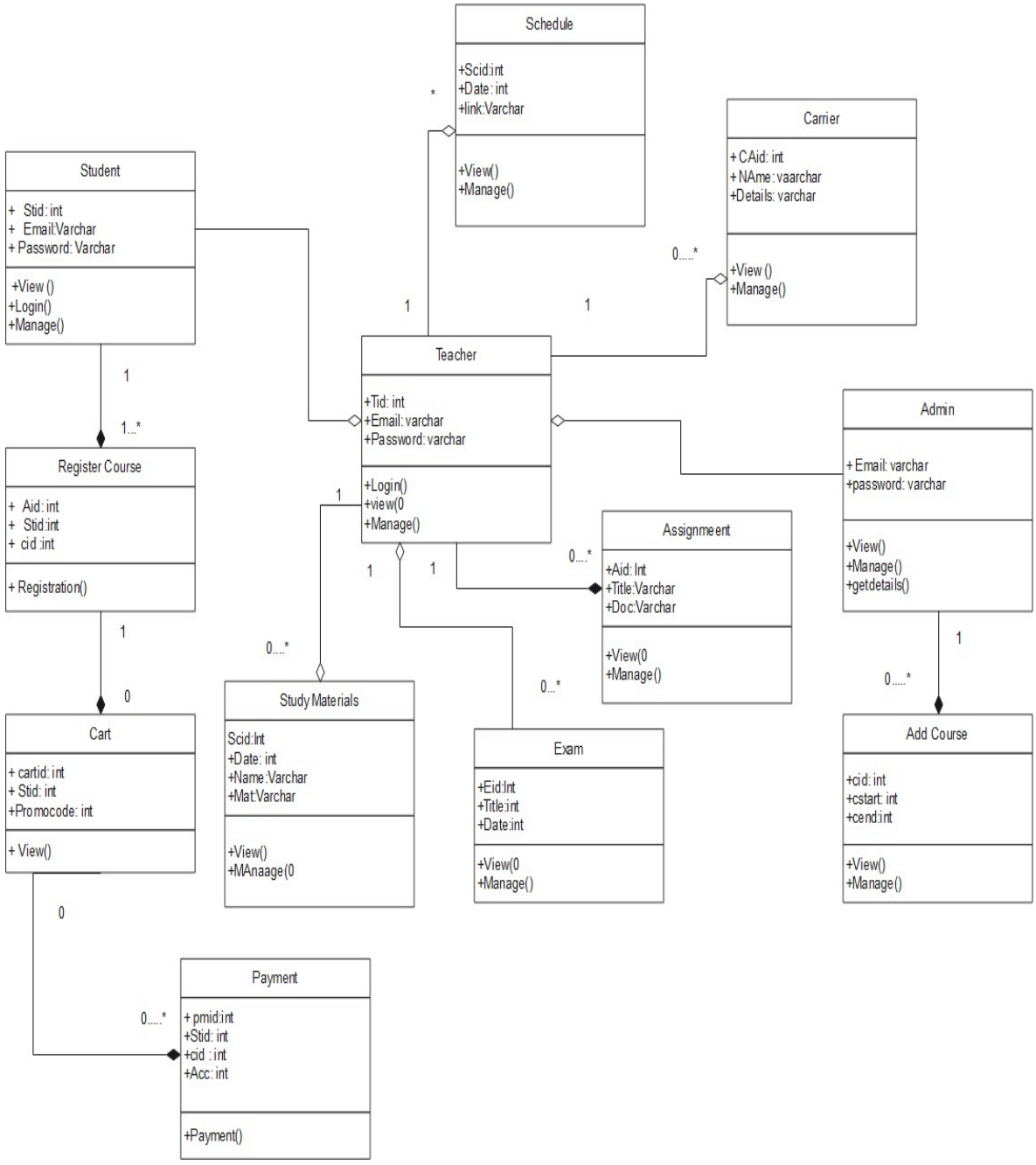
4.2.2 Activity Diagram

Another crucial UML diagram for describing the system's dynamic elements is the activity diagram. An activity diagram is essentially a flowchart that shows how one activity leads to another. The action might be referred to as a system operation. One operation leads to the next in the control flow. This flow may be parallel, contemporaneous, or branched. Activity diagrams use many features, such as fork, join, etc., to cope with all types of flow control. An activity diagram is a behavioral diagram i.e., it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.


Login
Valid
Invalid
Admin
Teacher
Student
Add course
View Students
View Teachers
Attend Assignment
Attend Exam
Payment
Apply Course
View Schedule
View Study Materials
Add Exam
Add Assignment
Add Study Materials
View payment
Schedule class
Logout

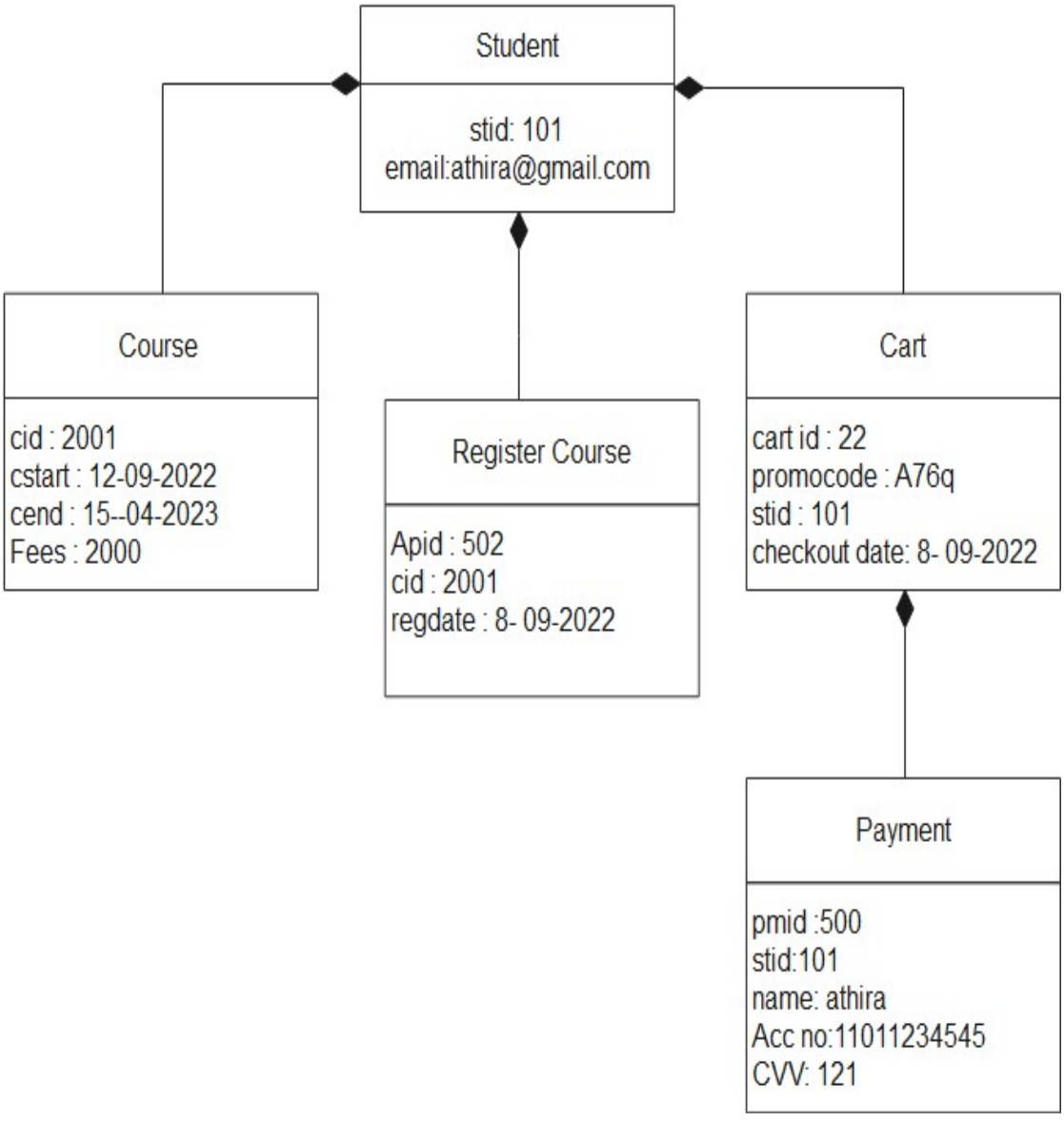
4.2.3 Class Diagram

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling, translating the models into programming code. Class diagrams can also be used for data modeling.



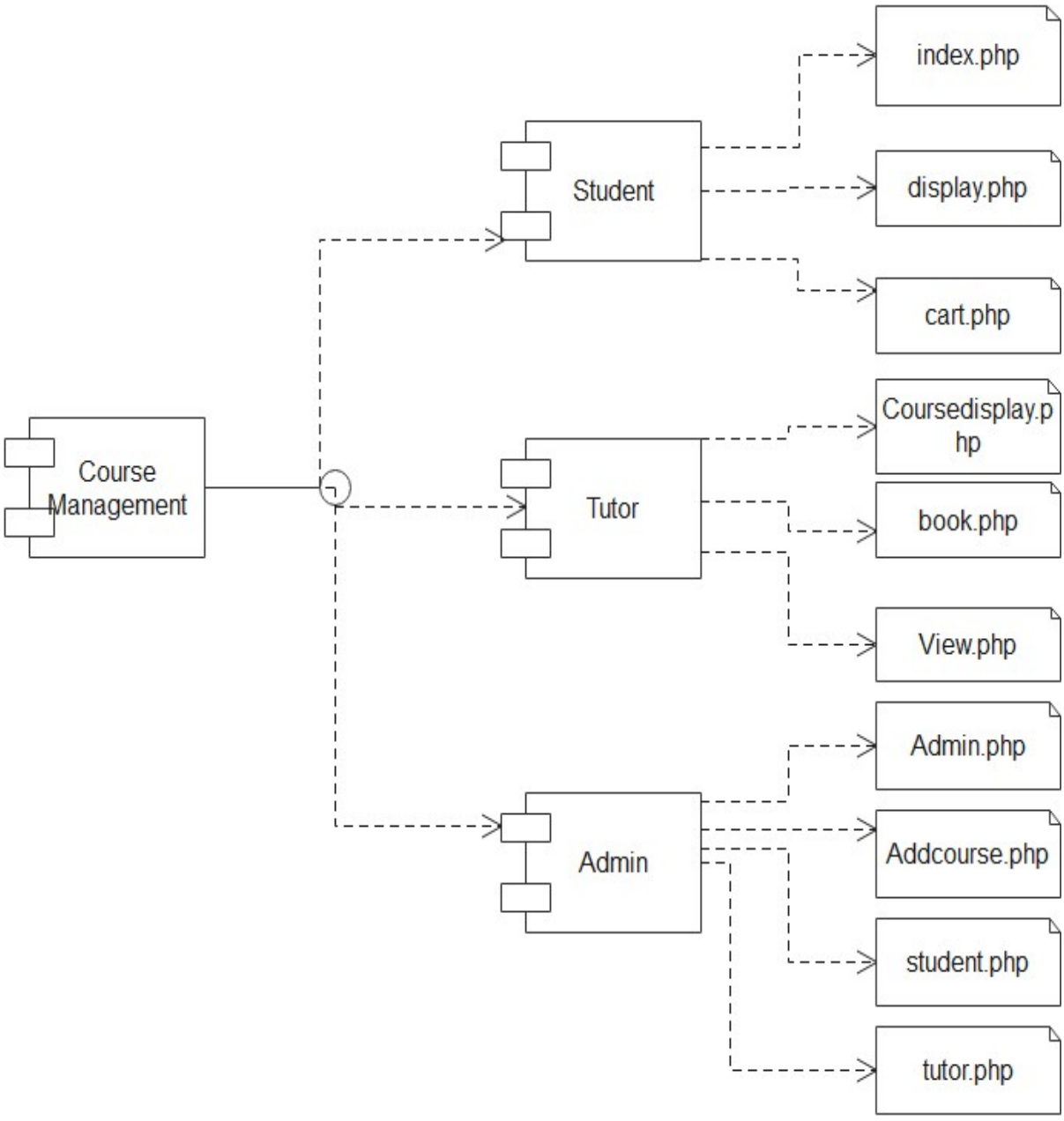
4.2.4 Object Diagram

Since class diagrams are the source of object diagrams, class diagrams are a prerequisite for object diagrams. An instance of a class diagram is represented by an object diagram. Class and object diagrams both use the same fundamental ideas. The static view of a system is also represented by object diagrams, but this static view represents a momentary snapshot of the system. To represent a group of items and their connections as an instance, object diagrams are employed.



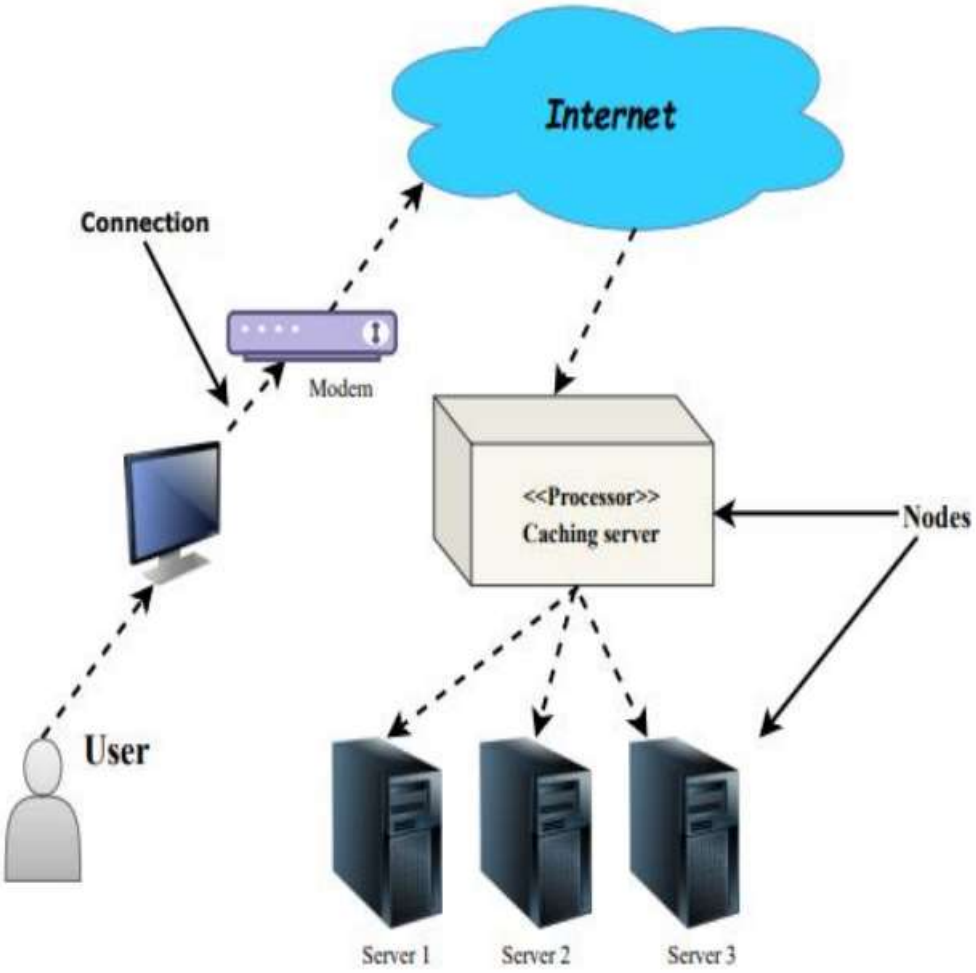
4.2.5 Component Diagram

A component diagram depicts how components are wired together to form larger components or software systems. They are used to illustrate the structure of arbitrarily complex systems.



4.2.6 Deployment Diagram

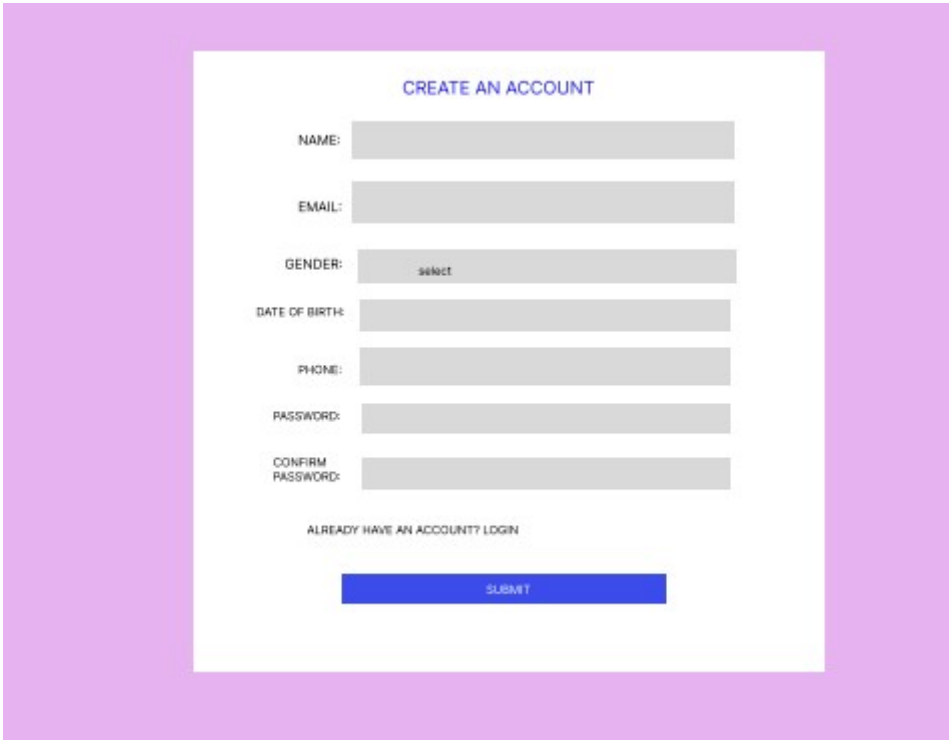
An execution architecture of a system, containing nodes like hardware or software execution environments, and the middleware linking them, is shown in a deployment diagram, a form of UML diagram. Typically, deployment diagrams are used to represent the actual hardware and software of a system. By using it, you can comprehend how the hardware will physically deliver the system. In contrast to other UML diagram types, which primarily depict the logical components of a system, deployment diagrams assist describe the hardware structure of a system.



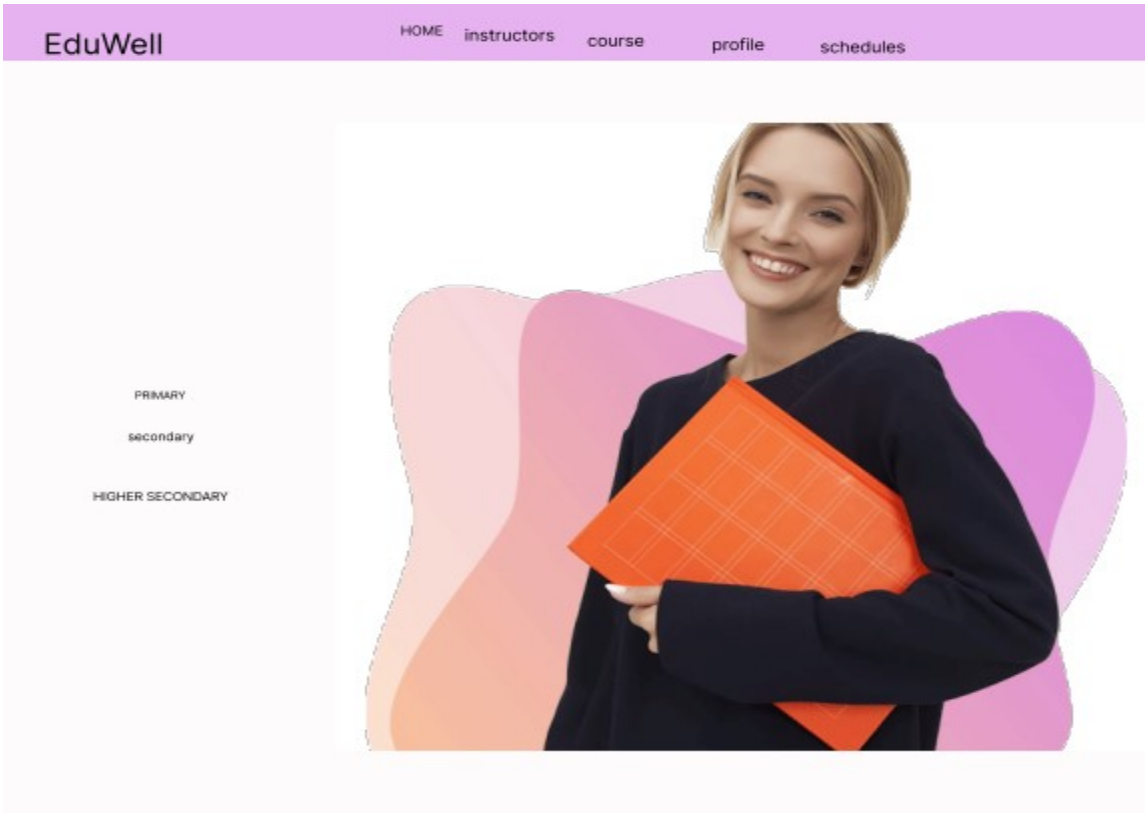
4.3 USER INTERFACE DESIGN USING FIGMA

Form Name: Login form

Form Name: Sign Up



Form Name: Index page



Form Name: Course Add

ADD COURSE

Category Name:

Subcategory Name:

Course Name:

Subject code:

Start Date:

End Date:

Fees:

Description:

Image:

Apply

4.4 DATABASE DESIGN

4.4.1 Relational Database Management System (RDBMS)

A database is an organized mechanism that has the 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 a two-level process. In the first step, user requirements are gathered together 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 second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity
- Data independence

4.4.2 Normalization

A relational model represents the database as a collection of relations. Each relationresembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, eachof which is assigned a unique name. A row in a tale represents a set of related values.

Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method ofspecifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values. Every value in a relation is atomic, that is not decomposable.

- Table relationships are established using Key. The two main keys of prime importance are

Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.

- Entity Integrity enforces that no Primary Key can have null values.
- Referential Integrity enforces that no Primary Key can have null values.
- Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

4.6.1 Normalization

Data are grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity. Normalization is a technique of separating redundant fields and breaking up a largetable into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies. Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table. There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form.

As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

- ✓ Normalize the data.
- ✓ Choose proper names for the tables and columns.
- ✓ Choose the proper name for the data.

First Normal Form

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute. In other words, 1NF disallows “relations within relations” or “relations as attribute values within tuples”. The only attribute values permitted by 1NFare single atomic or indivisible values. The first step is to put the data into First Normal Form. This can be donor by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the project. In this we form new relations for each

non-atomic attribute or nested relation. This eliminated repeating groups of data. A relation is said to be in first normal form if only if it satisfies the constraints that contain the primary key only.

Second Normal Form

According to Second Normal Form, for relations where primary key contains multiple attributes, no non-key attribute should be functionally dependent on a part of the primary key. In this we decompose and setup a new relation for each partial key with its dependent attributes. Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. This step helps in taking out data that is only dependent on a part of the key. A relation is said to be in second normal form if and only if it satisfies all the first normal form conditions for the primary key and every non-primary key attributes of the relation is fully dependent on its primary key alone.

Third Normal Form

According to Third Normal Form, Relation should not have a non-key attribute functionally determined by another non-key attribute or by a set of non-key attributes. That is, there should be no transitive dependency on the primary key. In this we decompose and set up relation that includes the non-key attributes that functionally determines other non-key attributes. This step is taken to get rid of anything that does not depend entirely on the Primary Key. A relation is said to be in third normal form if only if it is in second normal form and more over the non key attributes of the relation should not be depend on another non-key attribute.

4.4.3 Sanitization

Sanitizing data means removing any illegal character from the data. Sanitizing user input is one of the most common tasks in a web application. To make this task easier PHP provides native filter extension that you can use to sanitize the data such as e-mail addresses, URLs, IP addresses, etc.

PHP filters are used to sanitize and validate external input. The PHP filter extension has many of the functions needed for checking user input, and is designed to do data sanitization easier and quicker. This function, when using the flag in the example, is making sure that the code removes all characters except letters, digits and the following characters !#$%&'*+-=?_`{|}~@.[] . Many web applications receive external input. External input/data can be:

- User input from a form
- Cookies

- Web services data
- Server Variables
- Database query results

4.4.4 Indexing

The index stores the value of a specific field or set of fields, ordered by the value of the field. The ordering of the index entries supports efficient equality matches and range-based query operations. Indexes are used to quickly locate data without having to search every row in a database table every time a database table is accessed. Indexes can be created using one or more columns of a database table, providing the basis for both rapid random lookups and efficient access of ordered records. Indexes support the efficient execution of queries in PHP. An "index" can improve the speed of operation in a table. MySQL automatically creates an index for primary key, foreign key, and unique constraints. In addition, you may want to create "indexes" for other columns that are frequently used in joins or search conditions. The user cannot see indexes. You must have used a "CREATE INDEX" statement to create an index for one or more columns of a table. To create an index, write the table name and column names after the "on" clause. You can also use "UNIQUE" keywords to specify that an "index" has only unique values. You can also specify "ASC" and "DESC" keywords with a column name to indicate whether you want the "index" stored in ascending or descending order. If you do not specify "asc" or "desc", then "asc" is the default same as the "order by" keyword. .

4.5 TABLE DESIGN

1. Tbl_Login

Primary key: Log**id**

No:	Field Name	Data Type	Key Constarints	Description
1	Logid	Int	Primary key	Login id
2	Email	Varchar	Null	Email id
3	Password	Varchar	Null	Password
4	Role	Varchar	Null	Role
5	Status	Int	Null	Status given to user

2. Tbl _studentregister

Primary Key: stid

Foreign Key: Logid

No:	Field Name	Data Type	Key Constraints	Description
1	Stid	Int	Primary Key	Student id
2	Logid	Int	Foreign Key	Login id
3	Name	Varchar	Null	Name
4	Phone	Int	Null	Phone
5	Highest Qual	Varchar	Null	Highest quality
6	Gender	Varchar	Null	Gender
7	DOB	Int	Null	Date of birth
8	Img	Varchar	Null	Image
9	Status	Int	Null	Status given to user

3. Tbl _Instructorreg

Primary Key: Tid

Foreign Key: Logid,cid

No:	Field Name	Data Type	Key Constraints	Descrition
1	Tid	Int	Primary Key	Teacher id
2	Cid	int	Foreign Key	Course id
3	Logid	Int	Foreign key	Login id
4	Name	Varchar	Null	Name
5	Phone	Int	Null	Phone
6	Highest Qual	Varchar	Null	Highest quality
7	Gender	Varchar	Null	Gender
8	DOB	Int	Null	Date of birth
9	Img	Varchar	Null	Image
10	Status	Int	Null	Status given to user

4. tbl_Assignment

Primary Key: Aid

Foreign Key: Tid,cid

No:	Field Name	Data Type	Key Constraints	Description
1	Aid	Int	Primary Key	Assignment id
2	Tid	Int	Foreign key	Teacher id
3	Cid	Int	Foreign Key	Course id
4	Title	Varchar	Null	Title
5	Doc	Varchar	Null	Documents
6	Start time	Int	Null	Starting time
7	End time	Int	Null	Ending time
8	Start date	Varchar	Null	Staring date
9	End date	Varchar	Null	Ending date
10	Status	Int	Null	Status

5. Tbl_Assignsubmit

Primary Key: Asid

Foreign Key: Aid,Stid

No	Field Name	Data Type	Key Constraints	Description
1	Asid	Int	Primary Key	Assignment submit id
2	Aid	Int	Foreign Key	Assignment id
3	Stid	Int	Foreign Key	Student id
4	Subdate	Int	Null	Subject date
5	Submission	Varchar	Null	Submission
6	Mark	Int	Null	Mark
7	Status	Int	Null	Status

6. Tbl _Exam

Primary Key: Eid

Foreign Key: Tid,Cid

No	Field Name	Data Type	Key Constraint	Description
1	Eid	Int	Primary key	Exam id
2	Tid	Int	Foreign Key	Teacher id
3	Cid	Int	Foreign Key	Course id
4	Title	Varchar	Null	Title
5	Docx	Varchar	Null	Documents
6	Start time	Int	Null	Start time
7	End time	Int	Null	End time
8	Date	Varchar	Null	Date
9	Status	Int	Null	Status given to user

7. Tbl _Examsubmit

Primary Key: Exid

Foreign Key: Eid,Stid

No	Field Type	Data Type	Key Constraints	Description
1	Exid	Int	Primary key	Exam submit id
2	Eid	Int	Foreign Key	Exam id
3	Stid	Int	Foreign Key	Student id
4	Submission	Varchar	Null	Submission
5	Mark	Int	Null	Mark
6	Status	Int	Null	Status given to user

8. Tbl_ApplyCourse

Primary Key: Apid

Foreign Key: Stid,Cid

No	Field Name	Data Type	Key Constraints	Description
1	Apid	Int	Primary Key	Course register id
2	Stid	Int	Foreign Key	Student id
3	cid	Int	Foreign Key	Course id
4	Regdate	Date	Null	Registration id
5	Status	Int	Null	Status

9. Tbl_Course

Primary Key: Cid

Foreign Key: Catid,Subcatid,Tid

No	Field Name	Data Type	Key constraints	Description
1	Cid	Int	Primary key	Course id
2	Catid	Int	Foreign Key	Category Id
3	subcatid	Int	Foreign Key	Subcategory Id
4	Tid	Int	Foreign Key	Teacher id
5	Cstart	Date	Null	Course start
6	Cend	Date	Null	Course end
7	Fees	Int	Null	Fees
8	Status	Int	Null	Status

10. Tbl_payment

Primary Key: Pmid

Foreign Key: Stid,Cid

No	Field name	Data Type	Key Constraints	Description
1	Pmid	Int	Primary Key	Payment id
2	Stid	Int	Foreign Key	Student id
3	cid	Int	Foreign Key	Course id
4	Holder name	Varchar	Null	Holder name
5	Acc No	Int	Null	Account number
6	CVV No	Int	Null	Cvv number
7	EXP date	Date	Null	Expire date
8	Amount	Int	Null	Amount
9	Paymentdate	Date	Null	Payment date
10	Status	Int	Null	Status

11. Tbl_Cart

Primary Key: Cartid

Foreign Key: Stid

No	Field Name	Data Type	Key Constraints	Description
1	Cartid	Int	Primary key	Cart id
2	Stid	Int	Foreign Key	Student id
3	Promocode	Varchar	Null	Promo code
4	Checkout date	Date	Null	Check out date
5	Status	Int	Null	Status

12. Tbl_Cartproduct

Primary Key: Cpid

Foreign Key: Cartid

No	Field name	Data Type	Key constraints	Description
1	Cpid	int	Primary key	Cart Product id
2	Cartid	Int	Primary key	Cart id
3	Status	Int	Null	Status of given cart

13. Tbl_Rating

Primary Key: Rid

Foreign Key: Tid,Cid,Stid

No	Field Name	Data Type	Key Constraints	Description
1	Raid	Int	Primary key	Rating id
2	Tid	Int	Foreign key	Teacher id
3	cid	Int	Foreign key	Course id
4	Stid	Int	Foreign Key	Student id
5	Rate	Int	Null	Rate
6	Comment	Varchar	Null	Comment
7	Date	Date	Null	Date

14. Tbl_Schedule

Primary Key: Scid

Foreign Key: Cid,Tid

No	Field Name	Data Type	Key Constraints	Description
1	Scid	Int	Primary key	Schedule id
2	cid	Int	Foreign Key	Course id
3	Tid	Int	Foreign Key	Teacher id
4	Date	Date	Null	Date
5	Time	Int	Null	Time
6	Link	Int	Null	Link
7	Status	Int	Null	Status

15.Tbl_Carrier

Primary Key: Catid

Foreign Key: Tid

No	Field Name	Data Type	Key Constraints	Description
1	Caid	Int	Primary Key	Carrier id
2	Tid	Int	Foreign key	Teacher id
3	Name	Varchar	Null	Name
4	Details	Varchar	Null	Details
5	Status	Int	Null	Status

16. tbl_study materials

Primary Key: Smid

Foreign Key: cid,tid

No	Field name	Data Type	Key constraints	Description
1	Smid	Int	Primary key	Materials id
2	cid	Int	Foreign key	Course id
3	Tid	Int	Foreign Key	Teacher id
4	Name	Varchar	Null	Name
5	Studmat	Varchar	Null	Materials
6	Status	Int	Null	Status

17. tbl_Category

Primary Key: Catid

No	Field name	Data Type	Key Constraints	Description
1	Catid	Int	Primary key	Category Id
2	Catname	Varchar	Null	Category Name
3	Status	Int	Null	Status

18. tbl_Sybcategory

Primary Key: subtid

Foreign Key: Catid

No	Field name	Data Type	Key Constraints	Description
1	Subid	Int	Primary Key	Subcategory id
2	Catid	Int	Foreign key	Category id
3	Class	Varchar	Null	Class name
4	Status	Int	Null	Status

CHAPTER 5
SYSTEM TESTING

5.1 INTRODUCTION

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified? Software testing is often used in association with the term's verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

There are several rules that can serve as testing objectives. They are:

Testing is a process of executing a program with the intent of finding an error.

- A good test case is one that has high possibility of finding an undiscovered error.
- A successful test is one that uncovers an undiscovered error.

If a testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrate that the software function appears to be working according to the specification, that performance requirement appears to have been met. There are three ways to test program.

- For correctness
- For implementation efficiency
- For computational complexity

Test for correctness is supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

5.2 TEST PLAN

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers are always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder totest the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours perregression test all should be stated within the test plan. The levels of testing include:

- Unit testing
- Integration Testing
- Data validation Testing
- Output Testing

5.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Good design dictates that error conditions be anticipated and error handling paths set up to rerouteor cleanly terminate processing when an error does occur.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of th modules were found and were rectified. After coding each module is tested and run individually. All unnecessary code were removed and ensured that all modules are working, and gives the expected result.

5.2.2 Integration Testing

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing.The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Once these errors are corrected, new ones appear and the process continuesin a seemingly endless loop. After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover, differences in program structures were removed and a unique program structure was evolved.

5.2.3 Validation Testing or System Testing

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly knownas Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions thatwill fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

5.2.4 Output Testing or User Acceptance Testing

The system considered is tested for user acceptance; here it should satisfy the firm's need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

- Input Screen Designs,
- Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system understudy is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

5.2.5 Automation Testing

Automated testing is a process that validates if software is functioning appropriately and meeting requirements before it is released into production. This software testing method uses scripted sequences that are executed by testing tools. UI automation testing is a technique where these testing processes are performed using an automation tool. Instead of having testers click through the application to verify data and action flows visually, test scripts are written for each test case. A series of steps to follow when the verifying data is then added. Automatic testing is required when you want to run the same test cases across multiple machines at the same time.

5.2.6 Selenium Testing

Selenium is an open-source, automated, and valuable testing tool that all web application developers should be well aware of. A test performed using Selenium is usually referred to as Selenium automation testing. However, Selenium is not just a single tool but a collection of tools, each catering to different Selenium automation testing needs. In this tutorial you will learn all about Selenium and the various types of Selenium automation testing tools. Manual testing, a vital part of the application development process, unfortunately, has many shortcomings, chief of them being that the process is monotonous and repetitive. To overcome these obstacles, Jason Huggins, an engineer at Thought works, decided to automate the testing process. He developed a JavaScript program called the JavaScriptTestRunner that automated web application testing. This program was renamed Selenium in 2004.

CHAPTER 6
IMPLEMENTATION

6.1INTRODUCTION

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover. The implementation state involves the following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.

6.2 IMPLEMENTATION PROCEDURES

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will_commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that:

- The active user must be aware of the benefits of using the new system. Their confidence in the software is built up.
- Proper guidance is imparted to the user so that he is comfortable in using theapplication.

Before going ahead and viewing the system, the user must know that for viewing theresult, the server program should be running in the server. If the server object is not up running on the server, the actual process won't take place.

6.2.1 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

6.2.2 Training on the Application Software

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the date entered. It should thencover information needed by the specific user/ group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy

6.2.3 System Maintenance

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment. Software maintenance is of course, far more than "FindingMistakes".

CHAPTER 7
CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

This Course Management is to access learning materials anywhere from any device and opportunity to collaborate during the learning process. You can join a class in real-time or take a pre-recorded session whenever it works for you. In this busy world searching of tutor for any subject is very difficult. In order to make it simple here with simple here with we proposed an idea to find a subject expert tutor through this application. Online education enables the teacher and the student to set their own learning pace, and there's the added flexibility of setting a schedule that fits everyone's agenda. As a result, using an online educational platform allows for a better balance of work and studies, so there's no need to give anything up. Studying online teaches you vital time management skills, which makes finding a good work-study balance easier. Having a common agenda between the student and teacher can also prompt both parties to accept new responsibilities and have more autonomy.

7.2 FUTURE SCOPE

Online learning flexibility, accessibility adds to the scope of learning. From the comfort of home, you can access the online learning programs and continue your study without stepping out of your living room. This saves time and is also a cost-effective medium to enhance your knowledge and build a better portfolio. Scope of this project is very broad in terms of other manually finding teachers. This can be used by parents to find teachers whenever needed. Can be used anywhere any time as it is a web application. This app can be used by many students who love Teaching as a part time job. It provide more security. it eliminate the travel time for both parents and tutors.

Here we can add parent login for reporting parents for their children progress and they view students marks and grades. Adding groups for a class, where they can share more informations. It including a chat box for a public benefit where students and parents can know more about the tutors.

CHAPTER 8
BIBLIOGRAPHY

REFERENCES:

• Gary B. Shelly, Harry J. Rosenblatt, “System Analysis and Design”, 2009.
• Roger S Pressman, “Software Engineering”, 1994.
• PankajJalote, “Software engineering: a precise approach”, 2006.
• IEEE Std 1016 Recommended Practice for Software Design Descriptions.

WEBSITES:

• www.w3schools.com
• www.jquery.com
• www.bootstrap.com

CHAPTER 9
APPENDIX

9.1 Sample Code

Login

```
<?php
session_start();
include('config.php');

if(isset($_POST['sub']))
{
 echo 0;
 $email=$_POST["email"];
 $password=$_POST["password"];

 $sql="SELECT * from tbl_login where (email='$email' and password='$password');";
 $result = mysqli_query($conn, $sql);


 if (mysqli_num_rows($result) > 0){
  echo 1;
   foreach($result as $data)
   {
     $email=$data['email'];
     $password=$data['password'];
     $role=$data['role'];
     $login_id= $data['logid'];

   }
   $_SESSION['role']="$role";
   $_SESSION['email']="$email";
   $_SESSION['login_id']= $login_id;
   $_SESSION['auth_user']=[
     'email'=>$email,
     'password'=>$password
          ];
```

```
    if($_SESSION['role']=='admin')
    {
      echo 2;
      $_SESSION['message']="Welcome";
      header("location:Adminpanel.php");
      exit(0);
    }
    else if($_SESSION['role']=='teacher')
    {
      echo 3;
      $_SESSION['message']="Welcome";
       header("location:index.php");
      exit(0);
    }
    else if($_SESSION['role']=='student')
    {
      echo 4;
      $_SESSION['message']="Welcome";
       header("location:Student.html");
      exit(0);
    }
}
else
{
  echo 5;
    echo "Invalid Email ID/Password";
}
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>LOGIN</title>
   <link rel="stylesheet" href="login.css">
   <link rel="stylesheet"
href="//cdn.jsdelivr.net/npm/alertifyjs@1.13.1/build/css/alertify.min.css"/>
                <link rel="stylesheet"
href="//cdn.jsdelivr.net/npm/alertifyjs@1.13.1/build/css/themes/bootstrap.min.css"/>
                <script
src="//cdn.jsdelivr.net/npm/alertifyjs@1.13.1/build/alertify.min.js"></script>
   <!-- <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css" integrity="sha512-
Fo3rlrZj/k7ujTnHg4CGR2D7kSs0v4LLanw2qksYuRlEzO+tcaEPQogQ0KaoGN26/zrn20ImR1D
fuLWnOo7aBA==" crossorigin="anonymous" referrerpolicy="no-referrer" /> -->

   <script src="https://kit.fontawesome.com/2bbac3a66c.js" crossorigin="anonymous" ></script>
</head>
<body>
   <form id="create-account-form" action="Multilog.php" method="POST">

      <div class="title">
         <h2>Create Account</h2>
      </div>


      <!-- EMAIL -->
      <div class="input-group">
         <label for="mail">Email</label>
         <input type="email" id="email" name="email" placeholder="Email">
         <i class="fas fa-check-circle"></i>
         <i class="fas fa-exclamation-circle"></i>
         <p>Error Message</p>
      </div>
      <!-- PASSWORD -->
      <div class="input-group">
         <label for="pass">Password</label>
         <input type="password" id="password" name="password" placeholder="Password">
         <i class="fas fa-check-circle"></i>
         <i class="fas fa-exclamation-circle"></i>
         <p>Error Message</p>
      </div>

      <center>
                <p>Forgot password?<a href="forgot.php">Click here</a></p>
                </center><br>
      <button type="submit" name="sub"class="btn">Submit</button>
      <p>Create An Account<a href="Studentregister.php">Click here</a></p>
   </form>


   <!-- JAVASCRIPT -->
   <script src="login.js"></script>
```

```
</body>
</html>
```

Course Add

```
<?php
        include 'config.php';
        $targetDir="shopping/cimages/";
if(isset($_POST['submit']))
{

        $catid = $_POST['catid'];
  $cname = $_POST['cname'];
  $subcatid = $_POST['subcatid'];
        $subcode = $_POST['subcode'];
        $cstart = $_POST['cstart'];
  $pimage=$_FILES["pimage"]["name"];
        $cend = $_POST['cend'];
        $fees = $_POST['fees'];
        $description = $_POST['description'];
  $targetFilePath = $targetDir. $pimage;

  move_uploaded_file($_FILES["pimage"]['tmp_name'],$targetFilePath);

        $sql = "INSERT INTO `tbl_courseadd` (`catid`, `subcatid`, `cname`, `subcode`, `cstart`,
`cend`, `fees`, `description`, `pimage`, `status`) VALUES
('$catid','$subcatid','$cname','$subcode','$cstart','$cend','$fees', '$description', '$pimage', '1')";
        $result = mysqli_query($conn,$sql);
        if($result){
                echo "New record added";
                header('LOCATION:courseadddisplay.php');
        }
        else{
         echo mysqli_error($conn);
        }
}
?>
```

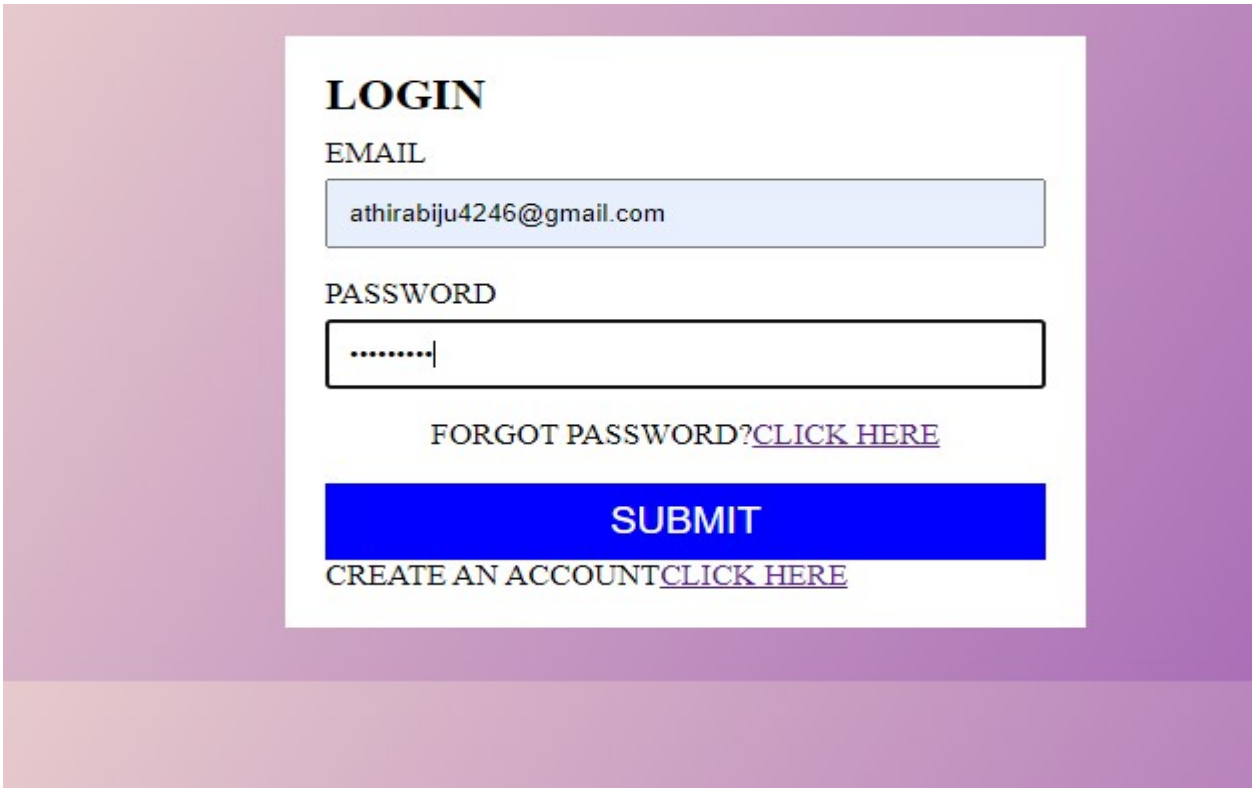
```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <link rel="stylesheet" href="stylescadd.css">
  </head>

  <body>
    <form method="POST" action="#" enctype="multipart/form-data">
<div class="wrapper">
    <div class="title">
      ADD COURSE
    </div>
    <div class="form">
    <div class="inputfield">
                          <?php
                          $con=mysqli_connect("localhost","root","","tutor");


                          $sql=mysqli_query($conn,"select * from tbl_category WHERE status=1");
                          ?>
                          <label>Category Name</label><br>


                          <select   name="catid" id="category" onchange="showResult(this.value)"
class="form-control m-bot15" required >
                          <option value=" ">--select--</option>
                          <?php
                          while($row=mysqli_fetch_array($sql))
                          {

                          ?>
                          <option value="<?php echo $row[0] ?>"> <?php echo $row[1] ?></option>

                          <?php

                          }
                          ?>

                          </select></div>
                          <div class="inputfield">
                          <?php
                          $con=mysqli_connect("localhost","root","","tutor");


                        $sql=mysqli_query($con,"select * from tbl_subcategory WHERE status=1");
                          ?>
                          <label>Subcategory Name</label><br>
```

```
                    <select   name="subcatid" id="sub_category"
                onchange="showResult(this.value)"   class="form-control m-bot15" required >
                    <option value="">--select--</option>
                    <?php
                    while($row=mysqli_fetch_array($sql))
                    {

                    ?>
                    <option value="<?php echo $row[0] ?>" ><?php echo $row[2] ?></option>
                    <?php

                    }
                    ?>

                    </select></div>
    <div class="inputfield">
      <label>Course Name</label>
      <input type="text" class="input" name="cname" placeholder="Course name"
onkeyup="this.value = this.value.toUpperCase();">

    </div>

    <div class="inputfield">
     <label>Subject Code</label>
     <input type="text" class="input" name="subcode" placeholder="Subject Code"
onkeyup="this.value = this.value.toUpperCase(subcode);" >
   </div>
    <div class="inputfield">
      <label>Start Date</label>
      <input type="date" class="input" name="cstart" placeholder="Start Date"  id="myDate"
min="2023-11-25" max="" >
      <script>
        function myFunction() {
          var x = document.getElementById("myDate").max = "2014-01-01";
                document.getElementById("inputfield").innerHTML = "The value of the max
        attribute was changed from '2000-01-01' to '2014-01-01'.";
        }
        </script>

     </div>
   <div class="inputfield">
      <label>End Date</label>
      <input type="date"  name="cend" class="input" min="2024-12-31">
       <script>
        function myFunction() {
          var x = document.getElementById("myDate").max = "2014-01-01";
                document.getElementById("inputfield").innerHTML = "The value of the max
        attribute was changed from '2000-01-01' to '2014-01-01'.";
        }
        </script>
```

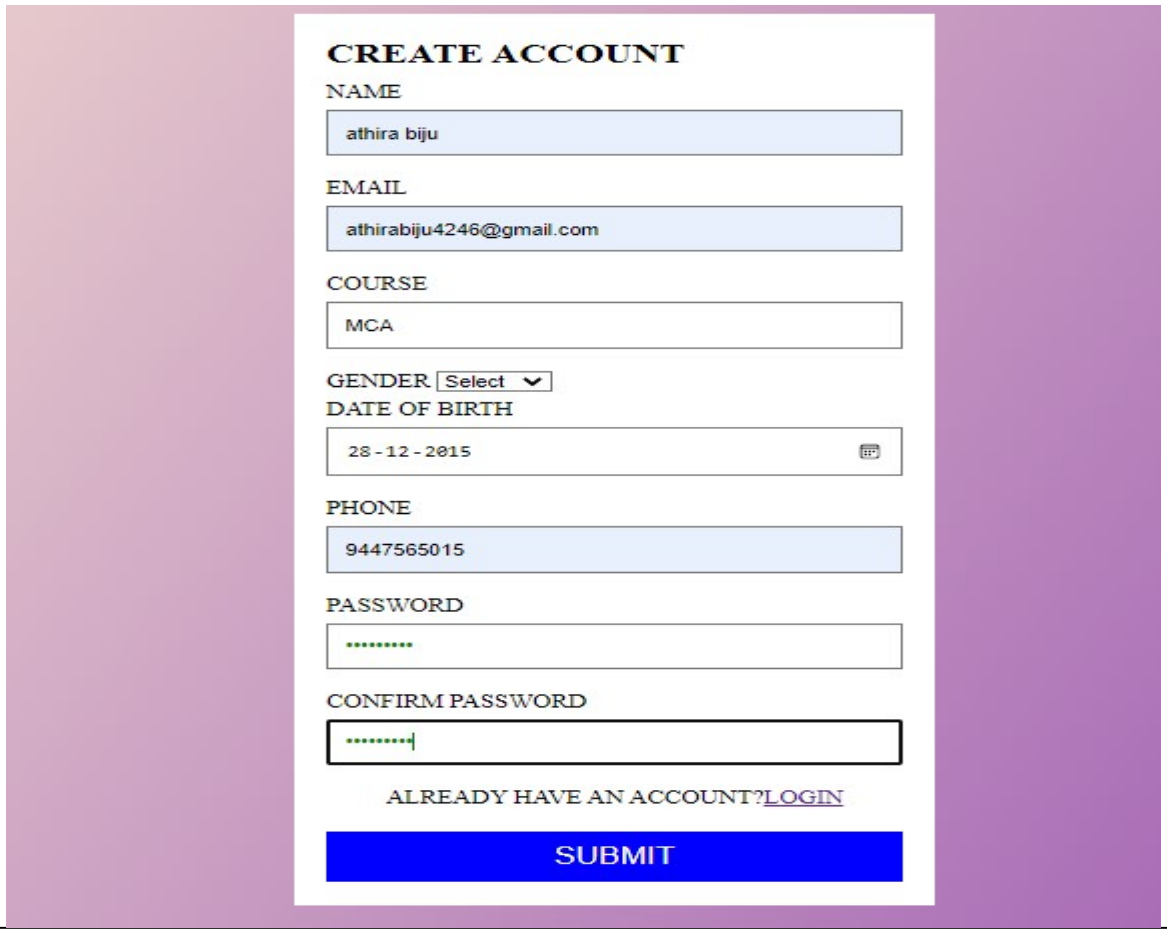
```
      </div>
     <div class="inputfield">
        <label>Fees</label>
                <input type="text" class="input" id="phone" name="fees" placeholder="Fees"
          onkeyup="phn_noValidation(this)" maxlength="4">
      </div>
      <div class="inputfield">
       <label>Description</label>
       <input type="text" class="input" name="description" placeholder="Description">
     </div>
     <div class="inputfield">
                                <label for="image">Product image</label>
                                        <input type="file" class="form-control" accept="image/gif,
          image/jpeg, image/png, image/jpg"  name="pimage" id="image" required>
                              </div>
      <div class="inputfield">
        <input type="submit" value="Apply" name="submit" class="btn">
      </div>
    </div>
</div>
</form>
<script type="text/javascript" src="date.js"></script>

</body>
</html>

<script>
  function phn_noValidation(inputTxt){

   var regx = /^\d{4}$/;
   var textField = document.getElementById("phone");

   if(inputTxt.value != '' ){
      if(inputTxt.value.match(regx)){
         textField.textContent = '';
         textField.style.color = "green";
      }else{
         textField.textContent = 'Your Mobile Number Is Not    Valid.';
         textField.style.color = "red";
      }
   }else{
      textField.textContent = 'your are not allowed  to leave a field  empty';
      textField.style.color = "red";
   }
}
  </script>
```

9.2 Screen Shots

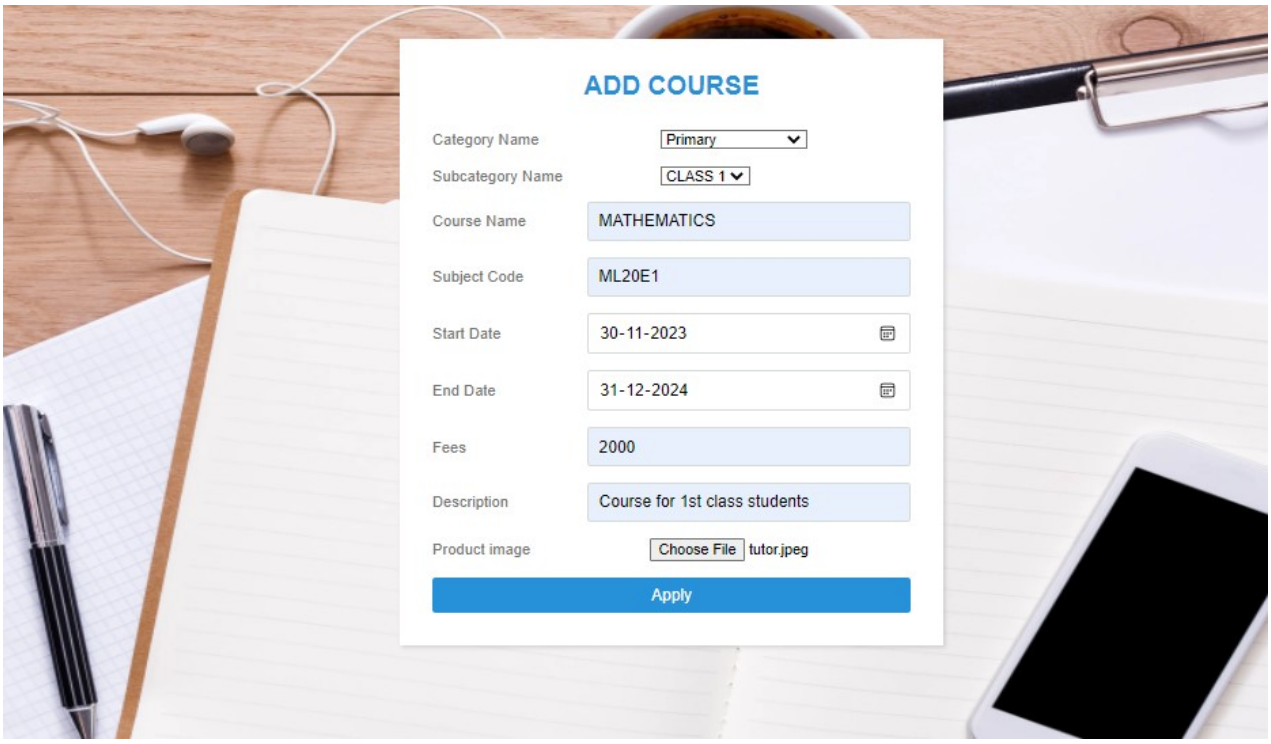
Login



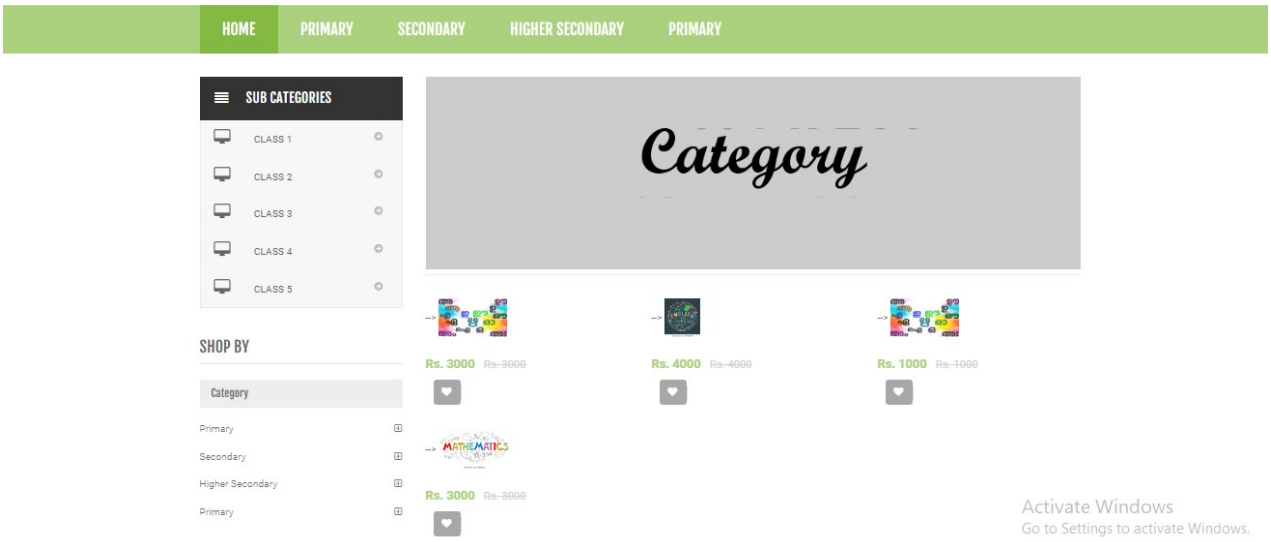
Registration



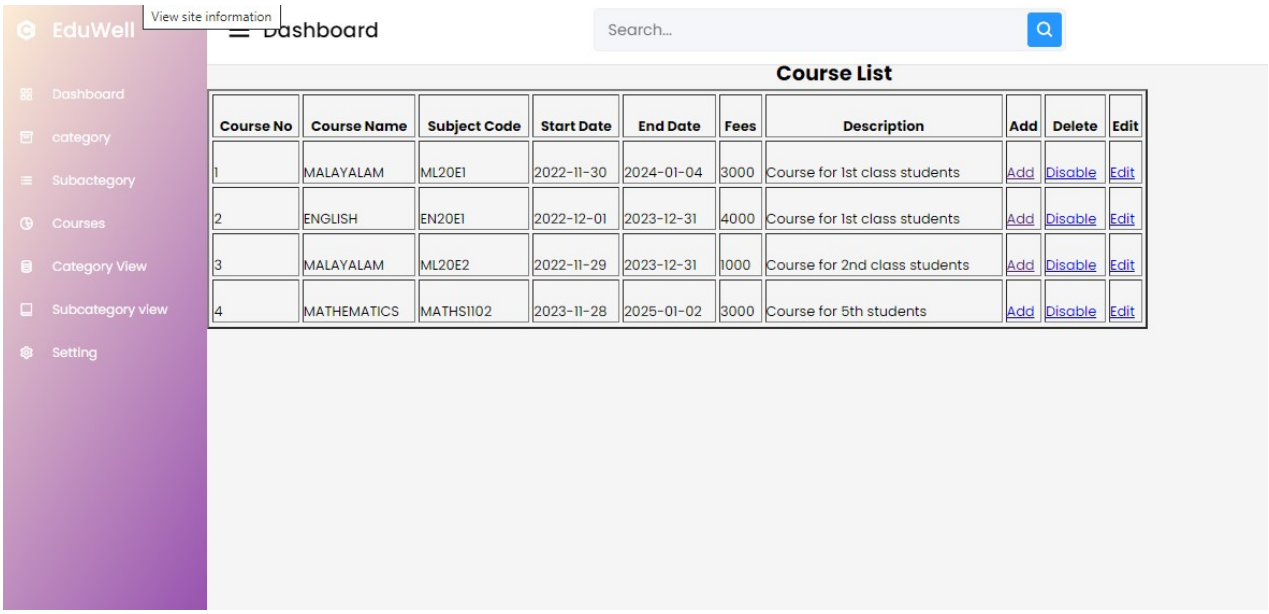
Course Add



Course Display



Course List


View site information
EduWell
Dashboard
Search...
Dashboard
category
Subactegory
Courses
Category View
Subcategory view
Setting
Course List
Course No
Course Name
Subject Code
Start Date
End Date
Fees
Description
Add
Delete
Edit
1 MALAYALAM ML20E1 2022-11-30 2024-01-04 3000 Course for 1st class students Add Disable Edit
2 ENGLISH EN20E1 2022-12-01 2023-12-31 4000 Course for 1st class students Add Disable Edit
3 MALAYALAM ML20E2 2022-11-29 2023-12-31 1000 Course for 2nd class students Add Disable Edit
4 MATHEMATICS MATHS1102 2023-11-28 2025-01-02 3000 Course for 5th students Add Disable Edit