A Project report
On

“LEARNING MANAGEMENT SYSTEM”

Submitted to the
Department of MCA
In partial fulfillment of the

MASTER OF COMPUTER APPLICATIONS

Under the guidance of
Dr. Shine K. George

Project Done by
ERICS ANTONY
Reg No:213242210218

DEPARTMENT OF MCA
UNION CHRISTIAN COLLEGE
ALUVA, KERALA

August-2023

UNION CHRISTIAN COLLEGE

ALUVA, KERALA

BONAFIDE CERTIFICATE

Certified that the Project Work entitled

“LEARNING MANAGEMENT SYSTEM”

is a bonafide work done by

Erics Antony

In partial fulfillment of the requirement for the Award of

MASTER OF COMPUTER APPLICATIONS

Degree From

Mahatma Gandhi University, Kottayam

(2021-2023)

Head of the Department **Project Guide**

Submitted for the Viva-Voce Examination held on ………………….

External Examiner 1 **External Examiner 2**

(Name & Signature) **(Name & Signature)**

UNION CHRISTIAN COLLEGE

ALUVA, KERALA

CERTIFICATE

This is to certify that the project entitled "LEARNING MANAGEMENT SYSTEM" has been successfully carried out by ERICS ANTONY (Reg No: 213242210218) in partial fulfilment of the Course Master of Computer Applications.

INTERNAL GUIDE

Date: **HEAD OF THE DEPARTMENT**

UNION CHRISTIAN COLLEGE

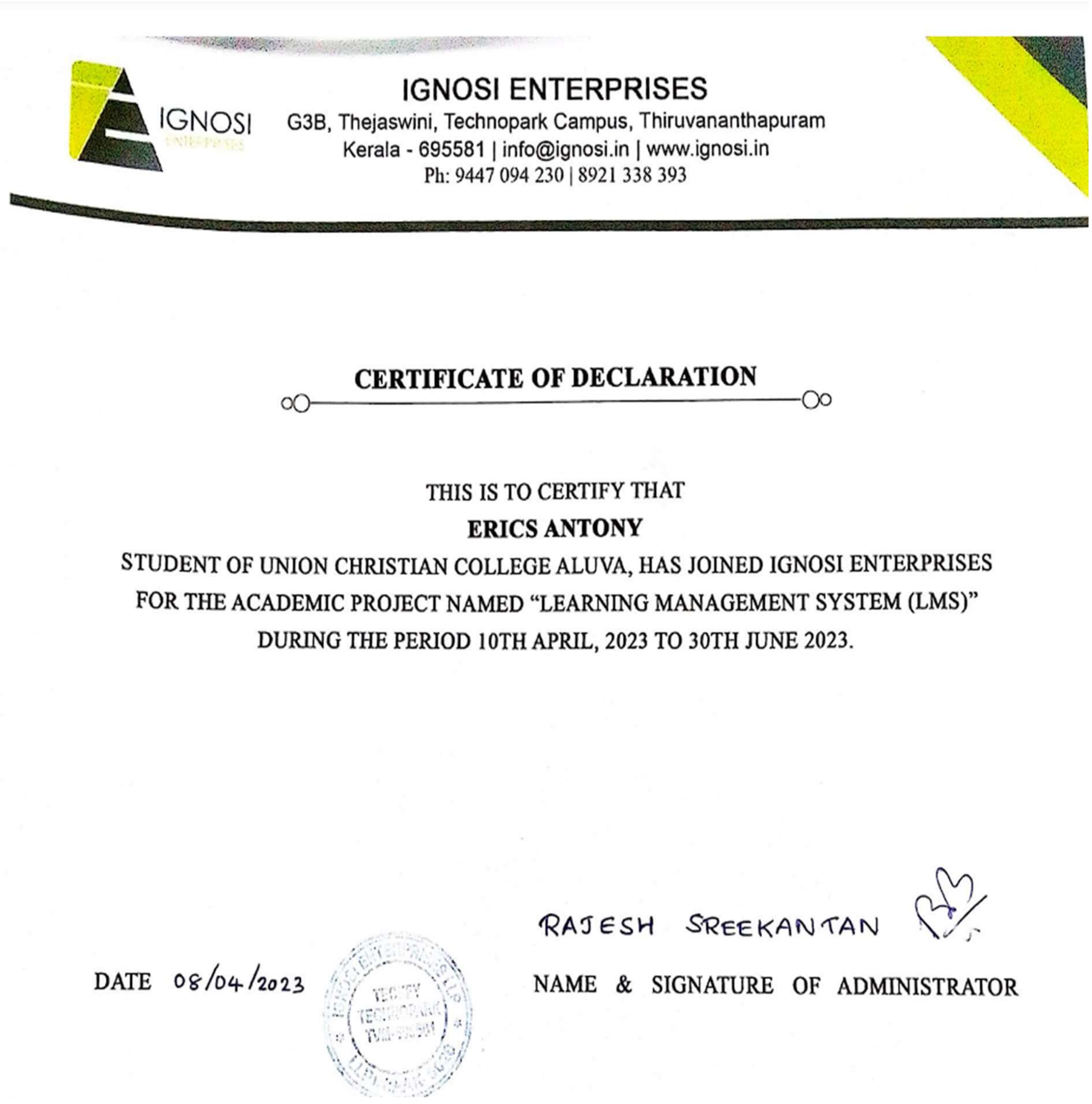
ALUVA, KERALA

CERTIFICATE

This is to certify that the project entitled "LEARNING MANAGEMENT SYSTEM" has been successfully carried out by ERICS ANTONY (Reg No: 213242210218) in partial fulfilment of the Course Master of Computer Applications under my guidance.

Date: **Dr. Shine K. George**

INTERNAL GUIDE



IGNOSI ENTERPRISES

G3B, Thejaswini, Technopark Campus, Thiruvananthapuram
Kerala - 695581 | info@ignosi.in | www.ignosi.in
Ph: 9447 094 230 | 8921 338 393

CERTIFICATE OF DECLARATION

THIS IS TO CERTIFY THAT

ERICS ANTONY

STUDENT OF UNION CHRISTIAN COLLEGE ALUVA, HAS JOINED IGNOSI ENTERPRISES FOR THE ACADEMIC PROJECT NAMED "LEARNING MANAGEMENT SYSTEM (LMS)" DURING THE PERIOD 10TH APRIL, 2023 TO 30TH JUNE 2023.

RAJESH SREEKANTAN

DATE 08/04/2023

IGNOSI ENTERPRISES LLP
TECHNOPARK

NAME & SIGNATURE OF ADMINISTRATOR

UNION CHRISTIAN COLLEGE

ALUVA, KERALA

THE TRUTH SHALL MAKE YOU FREE

DECLARATION

I, ERICS ANTONY, hereby declare that the project work entitled "LEARNING MANAGEMENT SYSTEM" is an authenticated work carried out by me at IGNOSI ENTERPRISES PVT.LTD. under the guidance of Mr. Biju Kumar S.P. for the partial fulfilment of the course MASTER OF COMPUTER APPLICATIONS. This work has not been submitted for similar purpose anywhere else except to UNION CHRISTIAN COLLEGE, ALUVA.

I understand that detection of any such copying is liable to be punished in any way the school deems fit.

ERICS ANTONY

signature

Date:

Place:

UNION CHRISTIAN COLLEGE

ALUVA, KERALA

ACKNOWLEDGEMENT

First and foremost, I express my gratitude to God almighty for giving me an opportunity to excel in my efforts to complete this main project on time. I wish to express my deep sense of gratitude to **Dr. M.I. PUNNOOSE, Principle of Union Christian College, Aluva** for the support provided to me. I would like to express my gratitude and thanks to **Dr. A V Alex, Director of MCA, and Ms. Sherna Mohan, Head of the Department** for providing me the best facilities and atmosphere for the completion of the project.

My sincere thanks to **Project Coordinator, Ms. Sikha B. Kadayath, and my Internal Guide, Dr. Shine K. George** for their valuable guidance that helped me to complete the project work in a determined way within the stipulated time.

Finally, I would like to express my sincere thanks to my family and friends, for always being a source of inspiration, and for their undying support and encouragement without which this project wouldn’t have been a success.

TABLE OF CONTENTS

ABSTRACT

A Learning Management System (LMS) is a term used to describe software tools designed to manage user learning interventions. LMS is a web-based technology used to plan, implement and assess a specific learning process. LMS provides workspaces to facilitate information sharing and communication among students and lecturers to participate in course activities. Educators can distribute information to students, produce content material, prepare assignments and tests, engage in discussions, manage distance learning, and enable collaborative learning using forums, chats, and news services. Institutions use the LMS to supplement traditional face-to-face delivery where faculty members develop and share digital learning materials via the Internet. In this case, the LMS is used as an electronic repository of learning materials. Other institutions especially those offering distance education, have been combining LMS with traditional face-to-face delivery to reach more learners across various geographical boundaries. A Learning Management System (LMS) is an online web application used for creating, delivering, tracking, and reporting educational courses and outcomes. The project is having a view of distributed architecture, with centralized storage of the database. The application for the storage of the data is MySql server and all the user interfaces will be designed using HTML and CSS technology. Database connectivity is the SQL connection methodology. The standards of security and data protective mechanisms will be maintained. The specification will be normalized up to 3NF to eliminate all the anomalies that may arise due to the database transactions that are executed by the general users and the organizational administration. The basic constructs of table spaces, clusters, and indexes have been exploited to provide higher consistency and reliability for data storage. The MySql server was a choice as it provides a high level of reliability and security. The total front end will be dominated using HTML and CSS. At all proper levels, high care will be taken to check that the system manages the data consistently with proper business rules or validation.

CHAPTER 1 INTRODUCTION

1.1.Introduction

A Learning Management System, commonly abbreviated as LMS, is a web-based platform designed to facilitate the management, delivery, tracking, and assessment of educational content, etc. LMS platforms have become increasingly popular in various sectors, including corporate training, academic institutions, government organizations, and nonprofit entities. They serve as a centralized hub for organizing and delivering learning materials, as well as tracking the progress and performance of learners.

Key Features of a Learning Management System:

- **Course Management:** LMS platforms allow instructors to create, organize, and manage courses. They can upload contents, which can be helpful in the learning process.
- **User Management:** LMS systems offer user management features, enabling administrators to add, remove, and manage learners and instructors. This ensures that only authorized individuals can access the courses and materials.
- **Attendance:** LMS platforms provide a facility to mark attendance and calculate the percentages of students.
- **Communication and Collaboration:** LMS platforms often include communication tools such as discussion forums, chat features, and messaging systems. These foster interaction and collaboration among learners and instructors, facilitating a dynamic learning environment.
- **Flexibility and Accessibility:** LMS platforms are designed to be accessible from anywhere with an internet connection. Learners can access their courses and materials at their convenience, allowing for self-paced learning.
- **Assessment and Feedback:** LMS platforms include tools for creating quizzes, assignments, and assessments. Instructors can provide feedback on learners' work.

1.2.Problem Statement

The Union Christian College is facing an issue in delivering course content efficiently. The students are not able to get study materials from teachers in an organized manner. They need a centralized platform the access study materials, assess students, mark attendance, and for the management of students and teachers. Due to the unorganized way of delivering study materials, students are facing difficulty in accessing their materials. There also arises confusion related to attendance marking manually. So, the college needs a learning management system to tackle the problem.

1.3.Scope and Relevance of the Project

The scope of the project includes various contents mainly,

- **Learning Management:** LMS platforms provide a centralized system for managing various learning content. This includes organizing course materials, creating assessments, and monitoring learner progress.
- **User Management:** LMS systems facilitate the management of learners and teachers. They allow for user registration, role assignment, and access control to ensure the right individuals have appropriate permissions.
- **Progress Tracking:** LMS systems track learner progress, including assessment scores, and participation. This data can be used to assess learner performance and the effectiveness of the learning programs.
- **Communication:** LMS platforms often include communication tools like discussion forums, chat features, and messaging systems. These foster interaction and collaboration among learners, instructors, and peers, enhancing the learning experience.

The learning management system is relevant due to the following reasons:

- **Centralized platform:** The system can be used as a centralized storage of learning materials and related contents. The students can access all their materials from a single platform which is more convenient.
- **Easy management:** The teacher can easily manage and assess a student by conducting quizzes. Attendance can be marked daily and teachers can easily calculate attendance percentages by a click which saves a lot of time.
- **Time efficient:** As all the required functionalities are under a single platform, the time consumption for both teachers and students is reduced by a significant margin.
- **Error-free management of data:** The data stored in the LMS is validated properly and consistency is maintained throughout. So the possibility of errors is minimum.
- **Accessibility:** The data stored in LMS is accessible anywhere anytime. The only requirement for accessing data like study materials, and attendance data is proper internet connectivity for authorized users.

1.4.Objectives

The objectives of a Learning Management System (LMS) revolve around facilitating effective and efficient management, delivery, and tracking of students. The primary goals of an LMS are to enhance the learning experience, improve learner outcomes, and provide valuable insights for instructors.

- **Centralized Course Management:** To provide a centralized platform for instructors to create, organize, and manage courses and training materials. This includes uploading various content, such as quizzes, and assignments.
- **Enhanced Learner Engagement:** To engage learners through interactive content, discussion forums, and collaboration tools, fostering an active and participatory learning experience.
- **Personalized Learning Paths:** To offer personalized learning paths for individual learners, tailoring the content and assessments based on their strengths, weaknesses, and learning preferences.
- **Progress Tracking and Assessment:** To track learner progress and performance throughout the courses including quiz scores, and overall performance. This data helps instructors assess the effectiveness.
- **Accessibility and Flexibility:** To enable learners to access the LMS platform from anywhere with an internet connection, allowing for flexible and self-paced learning.
- **Collaboration and Communication:** To facilitate communication and collaboration among learners, instructors, and peers through discussion forums, chat features, and messaging systems.
- **Continuous Improvement:** To enable instructors and administrators to continuously improve courses and training materials based on learner feedback and performance data.

Overall, the objectives of an LMS are to provide a robust and comprehensive platform that optimizes the learning process, empowers learners with valuable knowledge and skills, and supports the growth and success of individuals and organizations alike.

CHAPTER 2 SYSTEM ANALYSIS

2.1 Introduction

Currently the study materials are provided in different ways such as google classroom, WhatsApp, Gmail, etc. The materials are provided unorganized. Students find this method inefficient and they are facing difficulty in finding the correct material when needed. Also, human errors may occur while marking students' attendance, leading to differences in attendance percentages. Internal exam scores are kept in Excel sheets or noted in books manually. Too many applications are included on the run.

2.2 Existing System

The current procedure is similar to traditional practices what we are seen. Like marks are stored in Excel sheets or books, attendance is stored in attendance registers in which there is a chance of incorrect or missing data. As already mentioned, the study materials are provided through google classroom, Gmail, and rarely WhatsApp. This may set up confusion in student mind about where the notes are. This practice also can cause a waste of time due to unnecessary searches. Doubts are cleared by making a phone call, through messaging, or meeting the teacher personally. The teacher may not be available when needed. The availability of previous year's question papers is a concern for students.

2.3 Proposed System

The learning management system aims at the centralization of study materials. That is, learning materials are made available under a single platform which eliminates a lot of difficulties for students. The materials can be easily accessed from anywhere. The internal marks of exams can be stored in the system for later purposes. Attendance can be stored in the system on a daily basis. This can reduce irregularities while marking attendance. A teacher can assess a student by giving assignments and quizzes through the system. The teacher can view the submitted assignments by students and also view extended details about a student's performance in quizzes. The system also has a provision to store the previous year's question papers. So, the students can refer to the question papers. Doubts of students can be cleared by asking their classmates through a discussion forum or by using write to teacher feature.

2.4 Feasibility Study

Technical Feasibility

The system is compatible with the organization's existing technical infrastructure, including hardware, operating system, and database. The database's purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users would be granted based on the role specified therefore it provides a technical guarantee of accuracy, reliability, and security. LMS includes features such as user authentication, and data encryption, to address any security vulnerabilities. LMS also provides a user-friendly interface that is easy to navigate and understand for both administrators and learners. The software, and hardware for the development of this project are already available in-house or are available as free and open source. The work for the project is done with current equipment and existing software technologies. Necessary bandwidth exists for providing fast feedback irrespective of the number of users using the system simultaneously.

Operational Feasibility

The operational feasibility depends on the acceptance and willingness of the faculty and students to embrace the system. The organization should provide sufficient training and support to faculty and students to ensure they understand how to use the LMS effectively. Training programs, documentation, etc. can help users navigate the system, create and manage classes, and engage with learning materials. The LMS also supports for easy migration of existing learning materials to the system. LMS provides administrative tools that simplify the management of students, teachers, and subjects. User-friendly interfaces for administrators to create and manage subjects, students, and teachers contribute to the operational feasibility of LMS. LMS will be accessible to users with diverse needs. Compliance with web accessibility standards ensures that all users can effectively navigate and engage with the system. A user-friendly interface, intuitive navigation, and a consistent learning experience across different browsers further enhance operational feasibility. LMS aligns with the existing processes of the organization.

Economic Feasibility

The system is economically feasible. It does not require any additional hardware or software. Since the interface for this system is developed using the existing resources and technologies available, there is nominal expenditure and economically feasibility for certain. Training materials such as ppts, videos, and assessments, can be developed or acquired once and then delivered to a large number of users repeatedly, without the need of continuous investment in new materials. This reduces the overall cost per learner and enhances the economic feasibility of the LMS. An LMS allows users to access training materials at their own convenience. This saves users time and allow them to allocate it more efficiently to their core responsibilities. Increased productivity and reduced time contribute to economic feasibility. With an LMS, users can complete learning modules at their own pace, accelerating the learning process. As a result, the LMS enables faster knowledge acquisition and skill development, leading to improved productivity and cost savings.

2.5 Software Engineering Paradigm Applied

The Waterfall Model is a sequential software development process model that follows a linear, top-down approach. It divides the software development lifecycle into distinct phases, with each phase being completed before moving on to the next. Here are the typical phases of the Waterfall Model:

Requirements Gathering: In this initial phase, project stakeholders identify and document the requirements for the software. This includes gathering information about the desired features, functionality, and constraints of the system.

System Design: Based on the gathered requirements, the system design phase involves creating a detailed design specification for the software. It includes defining the overall system architecture, subsystems, interfaces, and data structures.

Implementation: In this phase, the software developers write code and implement the design. The implementation follows the specifications and guidelines set in the previous phases.

Testing: Once the implementation is complete, the testing phase begins. Testers perform various types of testing, including unit testing, integration testing, system testing, and user acceptance testing. The goal is to identify and fix any defects or issues in the software.

Deployment: After successful testing, the software is deployed or released to the end users or customers. This phase involves activities like installation, configuration, and setting up the system for production use.

Maintenance: The maintenance phase involves ongoing support and maintenance of the software after deployment. It includes bug fixes, updates, enhancements, and addressing user feedback or issues that arise during the software's use.

Key characteristics of the Waterfall Model include:

- Sequential and linear progression from one phase to another.
- Emphasis on thorough documentation and planning.
- Limited scope for changes or modifications once a phase is completed.
- High visibility of progress due to the sequential nature.
- Suitable for projects with well-defined and stable requirements.
- The Waterfall Model is often criticized for its lack of flexibility and limited ability to accommodate changing requirements.

CHAPTER 3 SYSTEM ANALYSIS

3.1 Introduction

The Software Design Document (SDD) for the Learning Management System (LMS) provides a comprehensive overview of the system's architecture, design principles, modules, and interactions. The LMS is a web-based application aimed at facilitating learning and training activities for educational institutions and organizations. The LMS is designed to cater to the needs of both educators and learners. It allows instructors to create classes, manage content, track learner progress, and assess performance. Learners can access courses, submit assignments, participate in discussions, and view their results.

3.2 Database Design

student table

Sl.no.	Field name	Data type, size	Constraints	Description
1	s_id	int(5)	Primary key	Student id
2	s_name	varchar(25)	Not null	Student name
3	s_email	varchar(30)	Not null	Student email id
4	s_batch	varchar(1)	Not null	Batch of student
5	s_yoa	int(4)	Not null	Year of admission
6	s_pass	varchar(30)	Not null	Password

Table 3.2. 1 student table

teacher table

Sl.no.	Field name	Data type, size	Constraints	Description
1	t_id	int(5)	Primary key	teacher id
2	t_name	varchar(25)	Not null	teacher name
3	t_email	varchar(30)	Not null	teacher email id
4	t_pass	varchar(30)	Not null	Password of teacher

Table 3.2. 2 Teacher table

class table

Sl.no.	Field name	Data type, size	Constraints	Description
1	c_id	int(5)	Primary key	class id
2	c_name	varchar(25)	Not null	class name
3	c_batch	varchar(1)	Not null	Batch (A/B/C)
4	c_yoa	varchar(30)	Not null	Year of admission
5	c_tid	int(5)	Foreign key	Teacher id

Table 3.2. 3 class table

subject table

Sl.no.	Field name	Data type, size	Constraints	Description
1	sub_id	int(5)	Primary key	subject id
2	sub_name	varchar(25)	Not null	subject name
3	sub_cid	int(5)	Foreign key	Class id

Table 3.2. 4 subject table

internal table

Sl.no.	Field name	Data type, size	Constraints	Description
1	i_id	int(5)	Primary key	internal id
2	i_first	int(2)	Not null	First internal marks
3	i_second	int(2)	Not null	Second internal marks
4	i_subid	int(4)	Foreign key	Subject id
5	i_sid	int(5)	Foreign key	Student id

Table 3.2. 5 internal table

attendance table

Sl.no.	Field name	Data type, size	Constraints	Description
1	at_id	int(5)	Primary key	attendance id
2	at_sid	int(5)	Foreign key	Student id
3	at_p1	int(1)	Not null	Period 1
4	at_p2	int(1)	Not null	Period 2
5	at_p3	int(1)	Not null	Period 3
6	at_p4	int(1)	Not null	Period 4
7	at_p5	int(1)	Not null	Period 5
8	at_p6	int(1)	Not null	Period 6
9	at_date	date	Not null	date

Table 3.2. 6 attendance table

assignment table

Sl.no.	Field name	Data type, size	Constraints	Description
1	a_id	int(5)	Primary key	assignment id
2	a_title	varchar(25)	Not null	Assignment name
3	a_description	varchar(100)	Not null	Assignment description
4	a_duedate	date	Not null	Due date
5	a_subid	int(5)	Foreign key	Subject id

Table 3.2. 7 assignment table

document table

Sl.no.	Field name	Data type, size	Constraints	Description
1	d_id	int(5)	Primary key	document id
2	d_sid	int(5)	Foreign key	Student id
3	d_asid	int(5)	Foreign key	Assignment id
4	d_name	varchar(25)	Not null	Document name
5	d_date	date	Not null	Submission date

Table 3.2. 8 document table

quiz table

Sl.no.	Field name	Data type, size	Constraints	Description
1	q_id	int(5)	Primary key	quiz id
2	q_title	varchar(25)	Not null	Quiz name
3	q_description	varchar(100)	Not null	Description or instructions
4	q_date	date	Not null	Date of quiz
5	q_cid	int(5)	Foreign key	Class id
6	q_time	varchar(5)	Not null	Time allotted for quiz

Table 3.2. 9 quiz table

notes table

Sl.no.	**Field name**	**Data type, size**	**Constraints**	**Description**
1	n_id	int(5)	Primary key	Note id
2	n_name	varchar(25)	Not null	note name
3	n_fname	varchar(25)	Not null	Name to be displayed
4	n_date	date	Not null	Date of upload
5	n_subid	int(5)	Foreign key	subject id

Table 3.2. 10 notes table

3.3 Object-Oriented Diagrams

UML Diagrams

Use Case Diagram

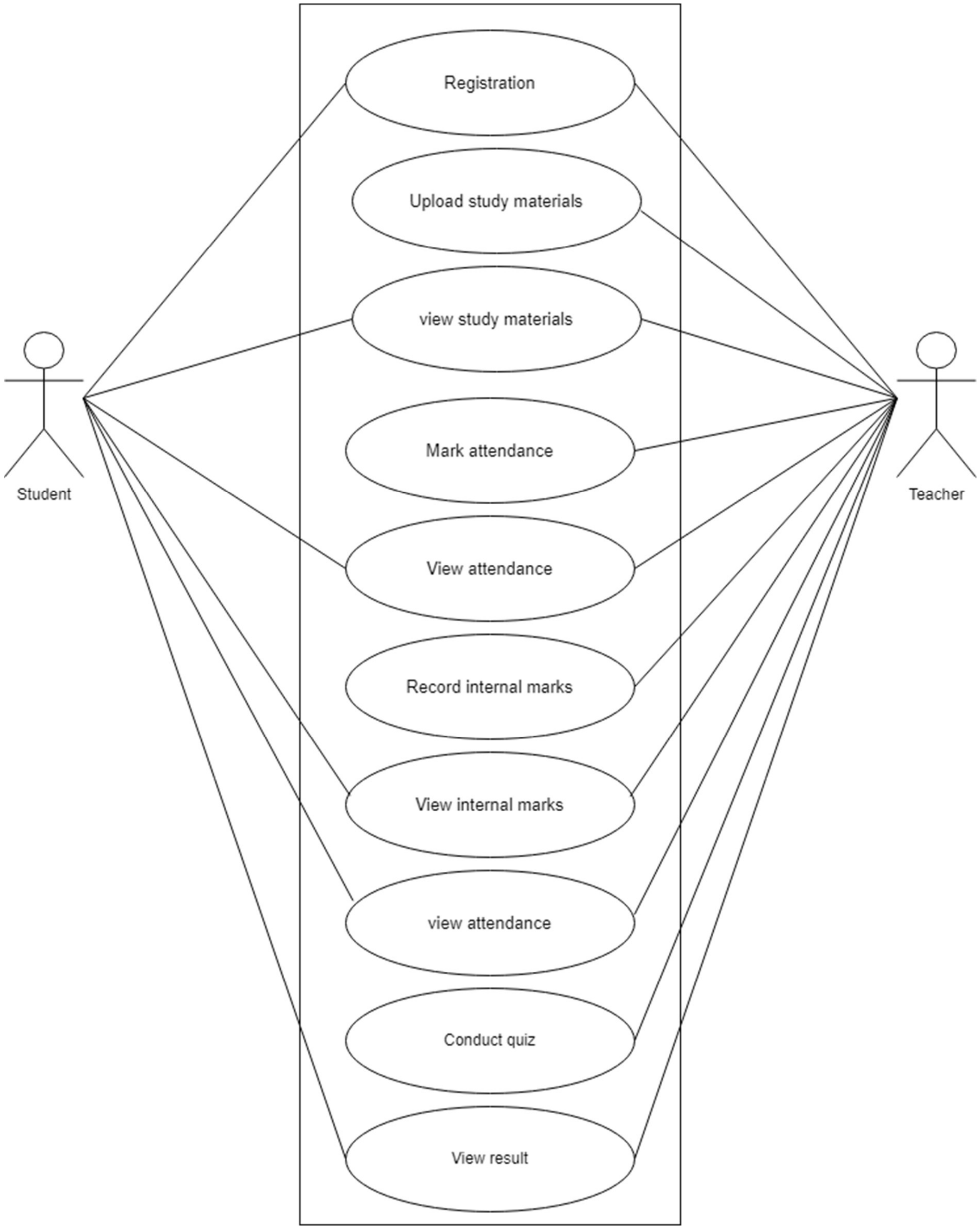


Figure 3.3. 1 Use case diagram

Activity Diagrams

Student Registration

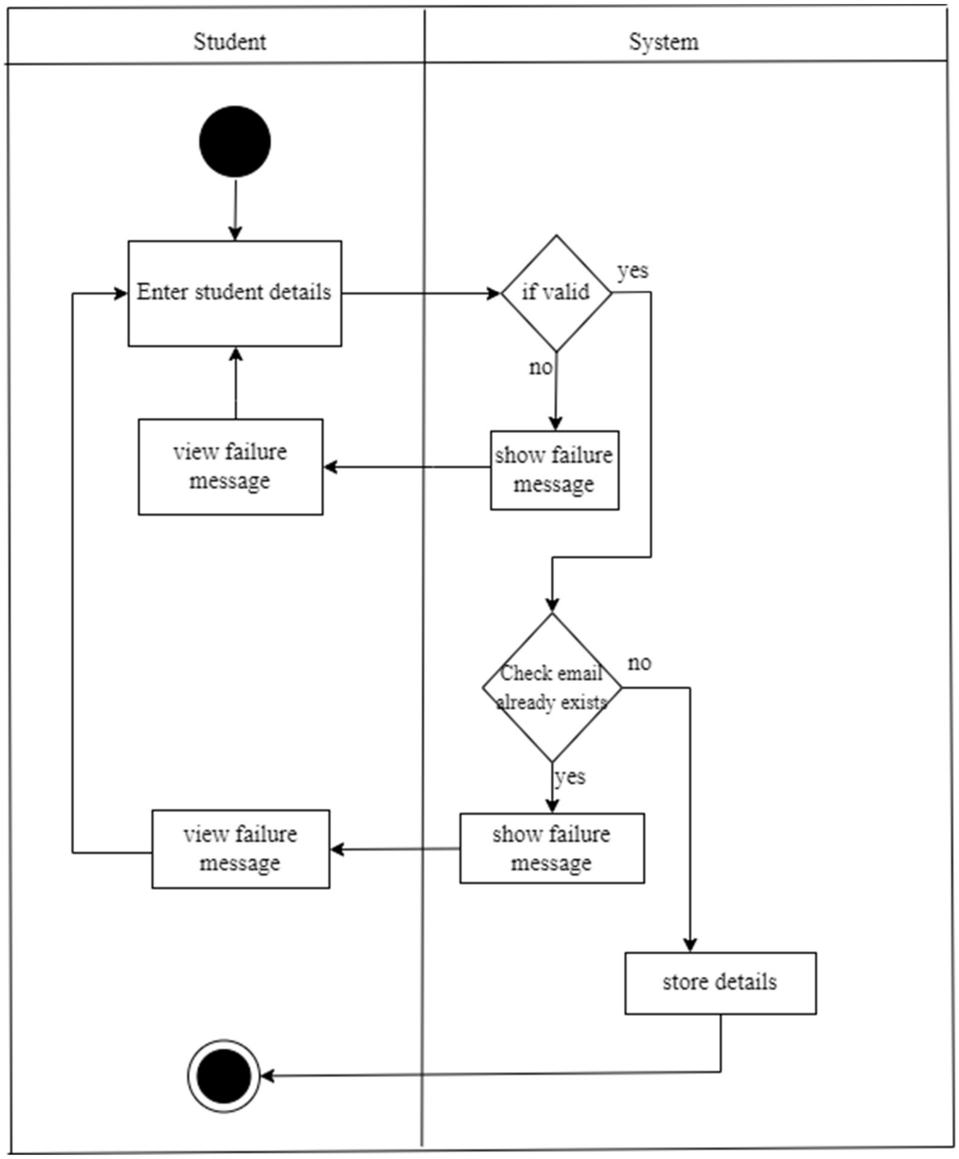


Figure 3.3. 2 student registration

Recording internal marks

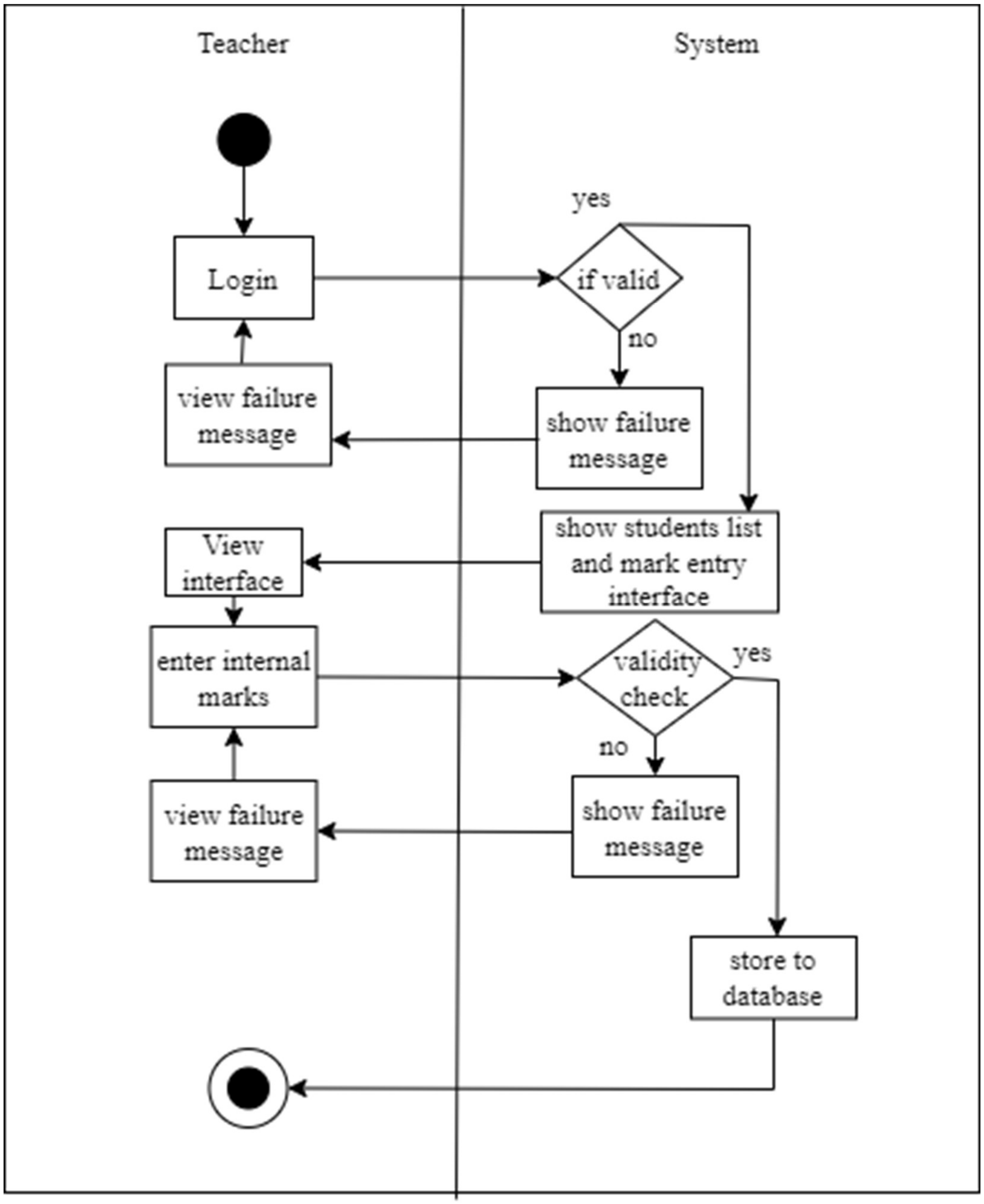


Figure 3.3. 3 Recording internal marks

Marking attendance

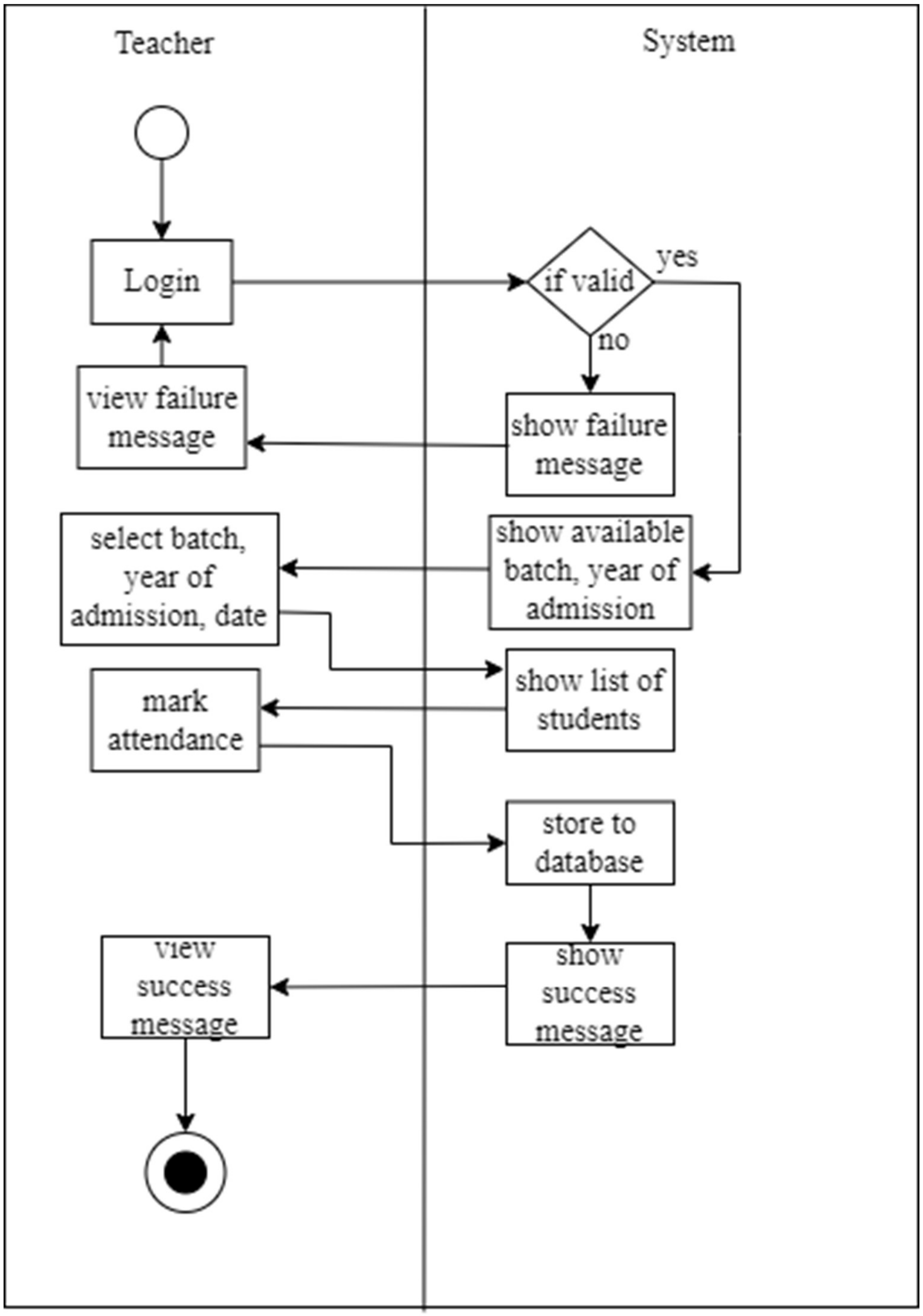


Figure 3.3. 4 marking attendance

Creating quiz

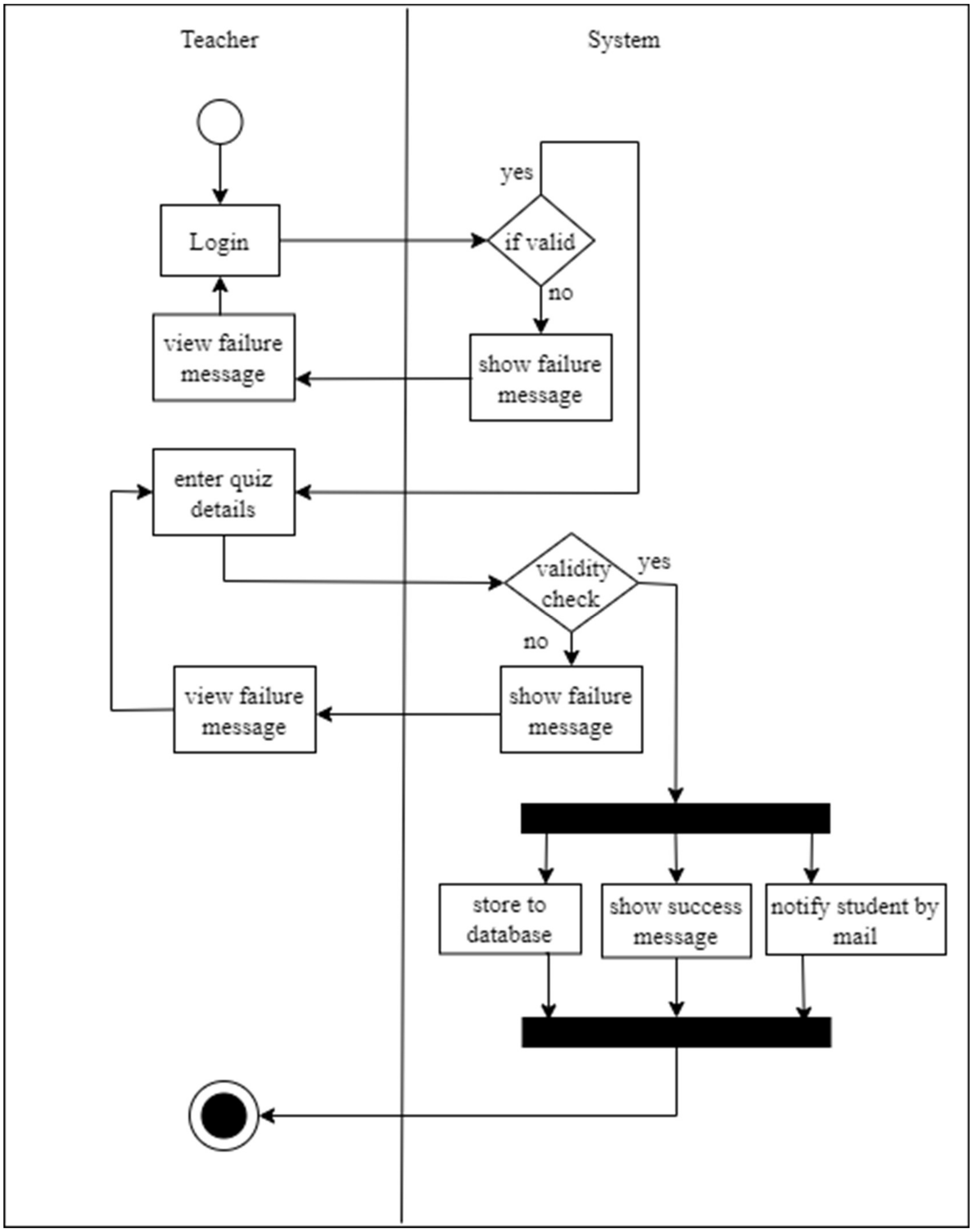


Figure 3.3. 5 create quiz

Upload notes

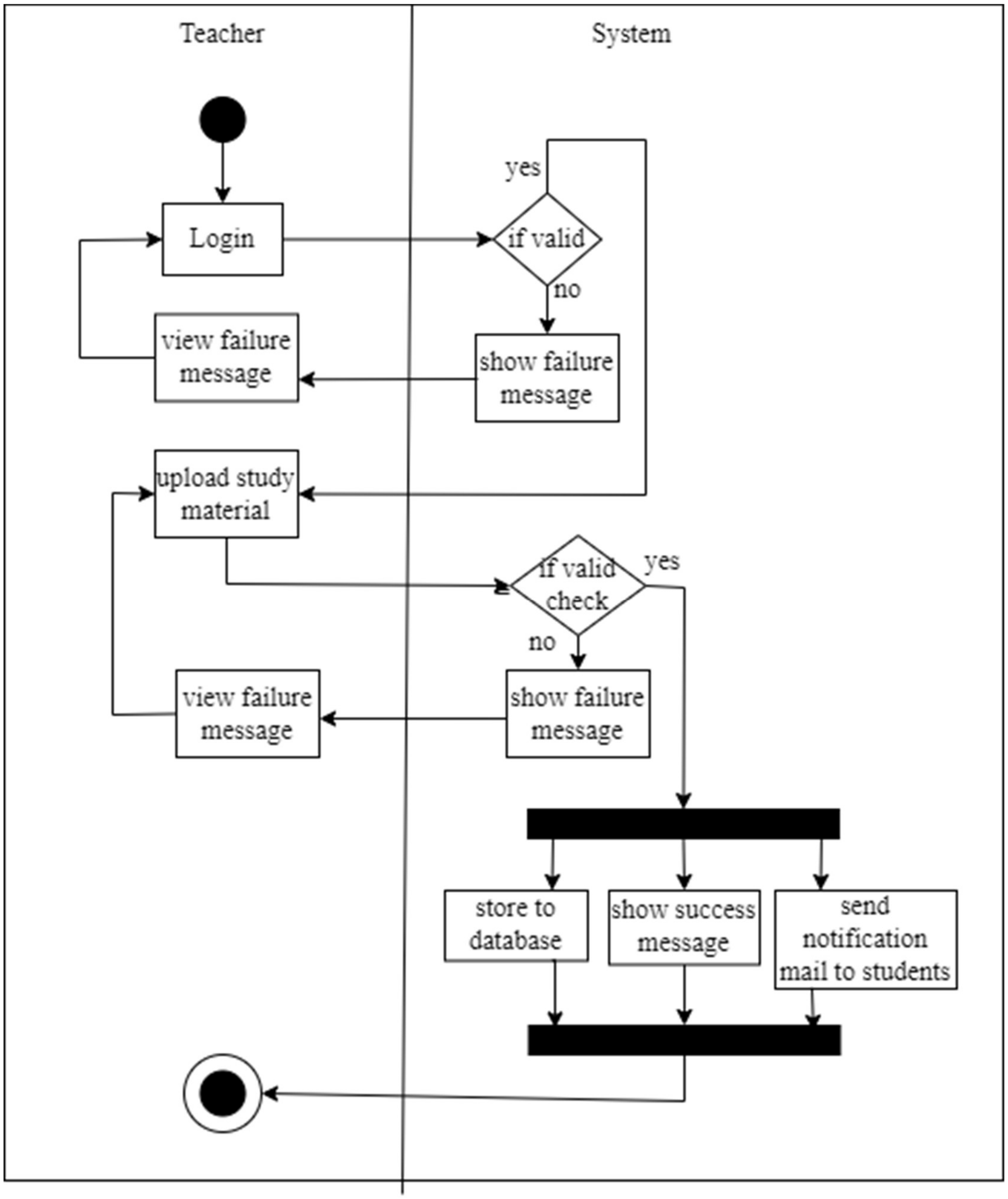


Figure 3.3. 6 upload notes

Class Diagrams

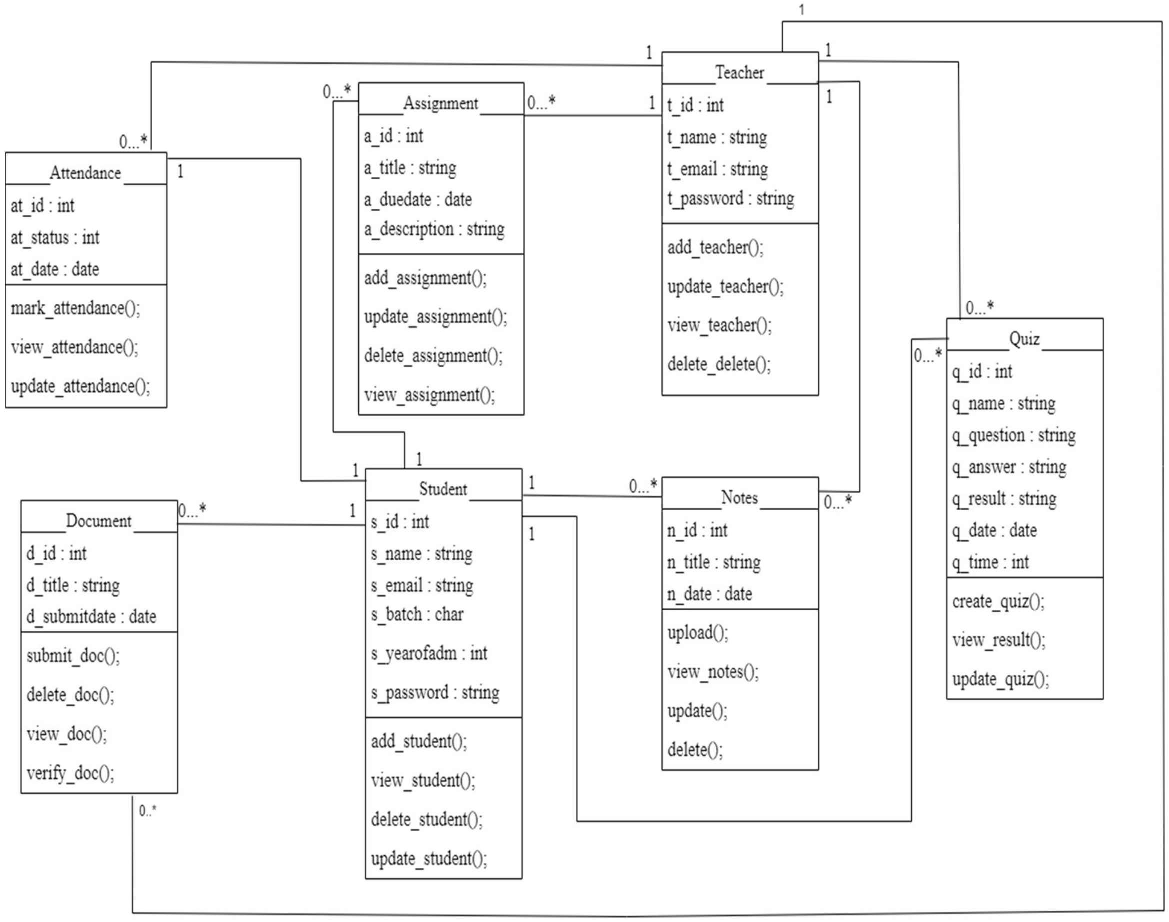


Figure 3.3. 7 class diagram

Sequence Diagram

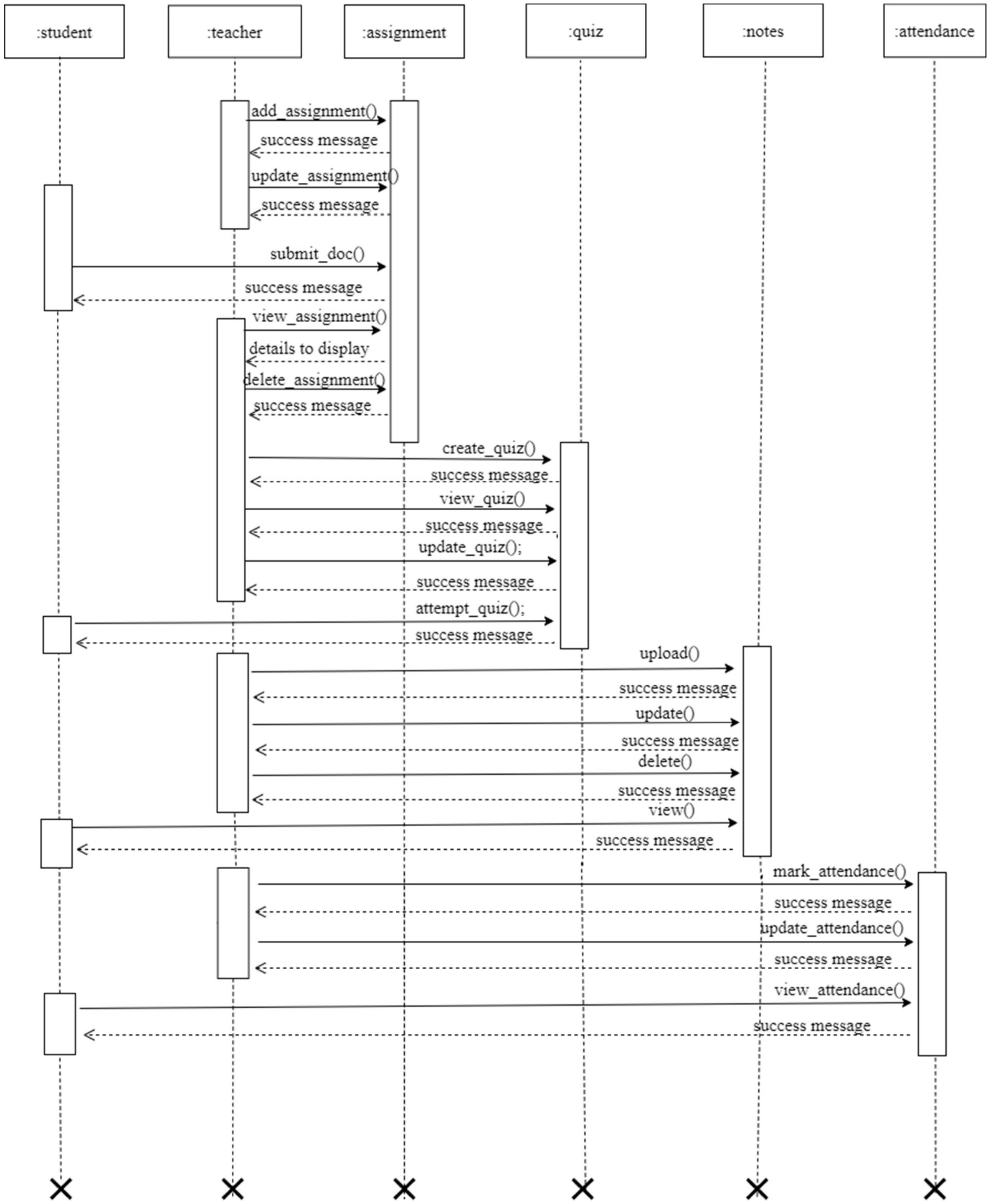


Figure 3.3. 8 sequence diagram

3.4 Modular Design

- **Upload study materials**

  The Study Material Module is a critical component of the Learning Management System (LMS) that enables instructors to upload and manage course materials for learners to access. This module provides a user-friendly interface for instructors to organize and deliver various study materials, such as lectures, presentations, and documents in pdf format. Learners can access these materials at their convenience to enhance their understanding of the course topics. Instructors have the ability to upload study materials to the LMS. Supported file formats may include PDFs. Instructors can edit, update, or remove study materials as needed, ensuring that the content remains up-to-date and relevant throughout the course duration. Learners can preview the study materials before downloading or accessing them. This feature enables learners to quickly assess whether the content is relevant to their needs. Learners can download study materials to their devices, enabling them to access the content even without an internet connection. This feature is particularly beneficial for learners who prefer offline studying. The module implements robust security measures to protect the integrity of the study materials and prevent unauthorized access. Role-based permissions ensure that only authorized users, such as instructors and administrators, can upload, edit, or delete study materials. Learners can access the materials but are restricted from modifying them.

- **Recording internal marks**

  The "Recording Internal Exam Marks" module is an essential component of the Learning Management System (LMS) that enables instructors and administrators to manage and record the marks obtained by learners in internal examinations. This module provides a user-friendly interface for inputting and tracking exam scores, allowing educators to evaluate learners' progress and performance accurately. After conducting the internal exams, instructors can enter the marks obtained by each learner. The module may include a moderation feature that allows designated faculty members or administrators to review and validate the recorded exam marks. This process ensures fairness and consistency. The "Recording Internal Marks" module adheres to strict security protocols to ensure that exam data remains confidential and protected from unauthorized access.

- **Marking the attendance**

  The "Marking the Attendance" module is a fundamental component of the Learning Management System (LMS) that enables instructors and administrators to efficiently track and record learners' attendance during classes or training sessions. This module provides an easy-to-use interface for instructors to take attendance and allows learners to view their attendance records. During the class or session, instructors can efficiently mark attendance for each learner. Instructors can view the list of enrolled learners and mark their attendance status. The module offers real-time attendance tracking, enabling instructors to monitor attendance records during the session. This feature allows them to take immediate action if any issues arise, such as identifying absent learners. Learners can access their individual attendance records through the LMS interface. They can view their attendance percentages and history for each session attended. Access to attendance records helps learners stay aware of their attendance status and take necessary actions to improve their attendance if needed. The "Marking the Attendance" module follows strict security measures to ensure the confidentiality of attendance data.

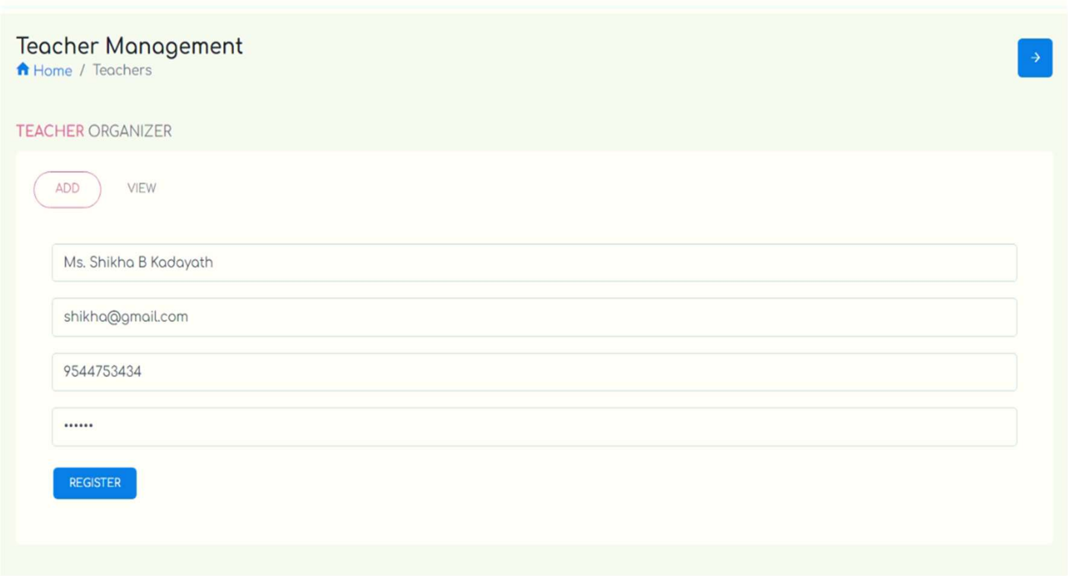
- **Giving assignments**

  The "Giving Assignments" module is a crucial component of the Learning Management System (LMS) that empowers instructors to create, distribute, and manage assignments for learners. This module facilitates efficient assignment delivery and submission, enabling instructors to assess learners' understanding of study material and track their progress. Instructors have the authority to create assignments within the LMS. They can define assignment details, such as the assignment title, description, due date, and any additional instructions or resources. Learners receive notifications about newly created assignments through the LMS notification system. The notification includes assignment details, submission instructions, and due dates. These notifications ensure that learners are informed promptly about upcoming assignments, enhancing their time management and preparedness. Learners can access the assignment details and submit their work electronically through the LMS. The module may allow learners to upload in pdf file formats. The module may automatically track late submissions.

- **Conducting quizzes**

  The "Conducting Quizzes" module is an integral part of the Learning Management System (LMS) that enables instructors to create, administer, and evaluate quizzes and assessments for learners. This module provides a comprehensive platform for conducting various types of quizzes, allowing instructors to gauge learners' knowledge, comprehension, and progress. Instructors have the authority to create quizzes within the LMS. They can define quiz details, such as the quiz title, description, duration, maximum score, and any specific instructions. The module supports only one question type, which is multiple-choice. The module may include a question bank feature, allowing instructors to store and organize a collection of questions for easy quiz creation. The module may support the randomization of questions and answer choices for each quiz attempt. This feature ensures that learners receive unique quiz versions, reducing the likelihood of cheating. The module may also support time limits for quiz completion, further enhancing quiz security and fairness. Learners receive notifications about upcoming quizzes through the LMS notification system. The notification includes quiz details, available time, and any additional instructions. Learners can access the quiz and begin their attempts. The "Conducting Quizzes" module automatically grades objective-type questions, such as multiple-choice and true/false.

3.5 Form Design



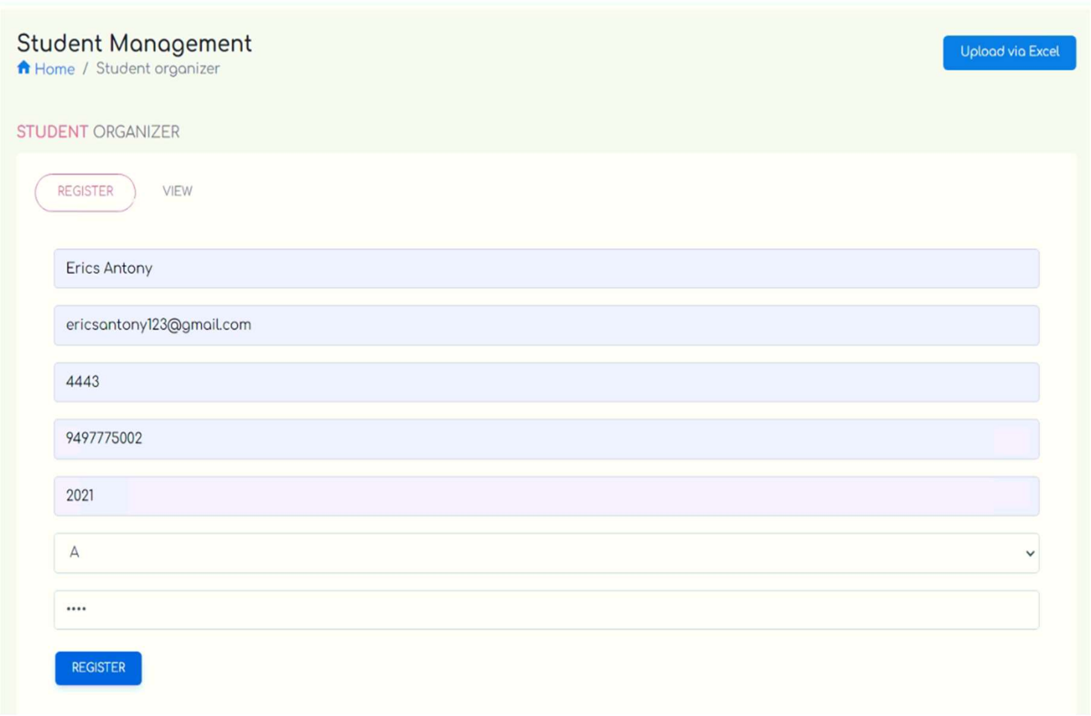
Ms. Shikha B Kadayath

shikha@gmail.com

9544753434

••••••

Figure 3.5. 1 Teacher registration form



Erics Antony

ericsantony123@gmail.com

4443

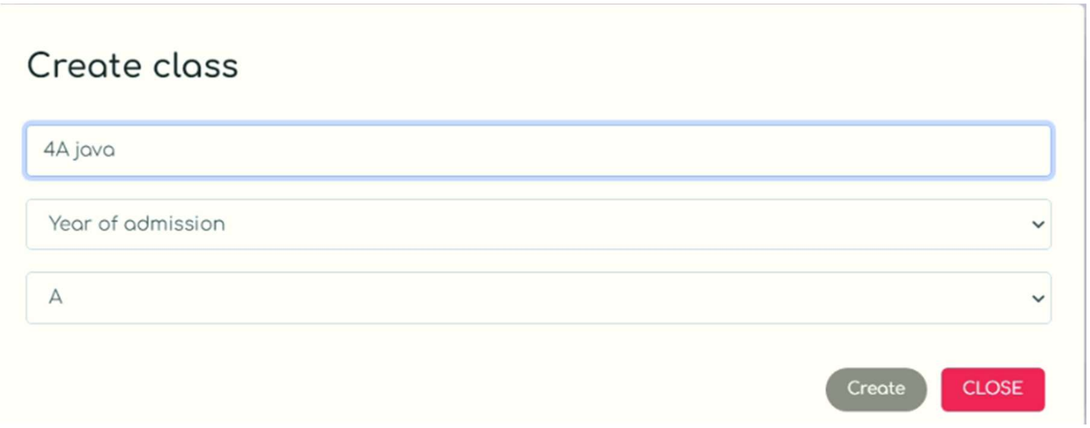
9497775002

2021

A

••••

Figure 3.5. 2 student registration form

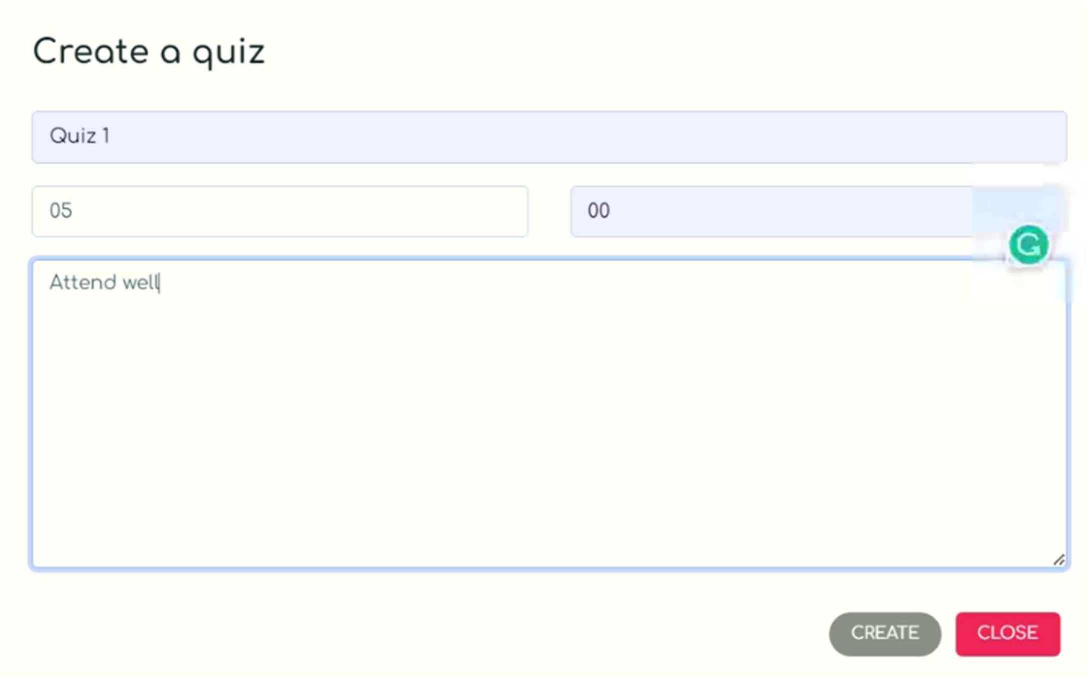


4A java

Year of admission

A

Figure 3.5. 3 create class form



Create a quiz

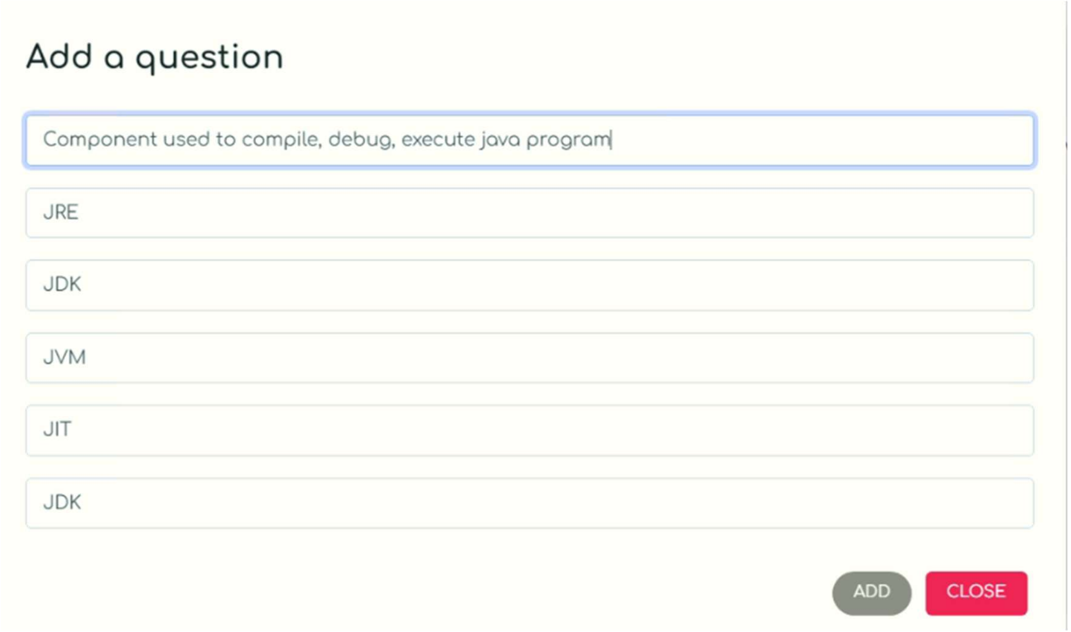
Quiz 1

05

00

Attend well

Figure 3.5. 4 create quiz form



Add a question

Component used to compile, debug, execute java program

JRE

JDK

JVM

JIT

JDK

Figure 3.5. 5 add question form

CHAPTER 4 SYSTEM ENVIRONMENT

4.1 Software Requirement Specification

The Software Requirements Specification (SRS) for the Learning Management System (LMS) outlines the functional and non-functional requirements of the system. The LMS is a web-based application designed to facilitate delivery, management, and evaluation. The primary stakeholders of the LMS include learners and instructors.

Functional requirements

- **User management**

  The LMS should support user registration and authentication for learners, instructors, and administrators. Learners should be able to update their profiles, including personal information and preferences. Instructors and administrators should have access to user management functionalities, such as adding, editing, and deactivating user accounts.

- **Content management**

  Instructors should be able to create, update, and delete content with relevant details like title, description, etc. The LMS should allow instructors to organize course materials and resources. Class enrollment and unenrollment should be automated. Content should be accessible to learners through a user-friendly interface with clear navigation.

- **Assignments and quizzes**

  Instructors should be able to create and manage assignments and quizzes. The LMS should support automated scoring for objective questions. Teachers should be able to view submitted assignments by students. email notification should be made for each student regarding the assignment or quiz creation.

- **Attendance management**

  The LMS should allow instructors to take attendance during classes or training sessions and record attendance status. Learners should be able to view their attendance records within the system. Instructors should also be able to change the attendance once marked.

- **Communication and collaboration**

  The LMS should facilitate communication between learners and instructors through messaging, discussion forums, and announcements. Learners should be able to interact with their peers through group discussions.

Non-functional requirements

- **Usability**

  The LMS should have an intuitive and user-friendly interface that is easy to navigate and understand for all user categories.
- **Performance**

  The LMS should be able to handle concurrent user traffic efficiently without significant performance degradation. Response times for essential actions, such as content delivery, should be optimized to provide a seamless user experience.
- **Security**

  Role-based access control should be enforced to restrict access to specific functionalities based on user roles.

4.2 Tools, Platforms

Front end tools

The front end is an interface between the user and the back end. The front and back ends may be distributed amongst one or more systems.

In network computing, *front end* can refer to any hardware that optimizes or protects network traffic. It is called application Front-end hardware because it is placed on the network's outward-facing front end or boundary. Network traffic passes through the front-end hardware before entering the network.

In compilers, the front end translates a computer programming source code into an intermediate representation, and the back end works with the intermediate representation to produce code in a computer output language. The back end usually optimizes to produce code that runs faster. The front-end/back-end distinction can separate the parser section that deals with source code and the back end that generates code and optimizes.

These days, front-end development refers to the part of the web users interact with. In the past, web development consisted of people who worked with Photoshop and those who could code HTML and CSS. Now, developers need a handle of programs like Photoshop and be able to code not only in HTML and CSS, but also JavaScript or jQuery, which is a compiled library of JavaScript.

Most of everything you see on any website is a mixture of HTML, CSS, and JavaScript, which are all controlled by the browser. For example, if you’re using Google Chrome or

Firefox, the browser is what translates all of the code in a manner for you to see and with which to interact, such as fonts, colors, drop-down menus, sliders, forms, etc. In order for all of this to work, though, there has to be something to support the front-end; this is where the backend comes into play.

PHP: -

Introduction

PHP is now officially known as "**PHP: Hypertext Preprocessor**". It is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page, a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module, which is server-side installed. HTML elements in the script are left alone, but PHP code is interpreted and executed. PHP code in a script can query databases, create images, read and write files, talk to remote servers – the possibilities is endless. The output from PHP code is combined with the HTML in the script and the result sent to the user's web-browser, therefore it can never tell the user whether the web-server uses PHP or not, because the entire browser sees is HTML.

PHP's support for Apache and MySQL further increases its popularity. Apache is now the most-used web-server in the world, and PHP can be compiled as an Apache module. MySQL is a powerful free SQL database, and PHP provides a comprehensive set of functions for working with it. The combination of Apache, MySQL and PHP is all but unbeatable.

That doesn't mean that PHP cannot work in other environments or with other tools. In fact, PHP supports an extensive list of databases and web-servers. While in the mid-1990s it was ok to build sites, even relatively large sites, with hundreds of individual hard-coded HTML pages, today's webmasters are making the most of the power of databases to manage their content more effectively and to personalize their sites according to individual user preferences.

Reasons for using PHP

There are some indisputable great reasons to work with PHP. As an open source product, PHP is well supported by a talented production team and a committed user community. Furthermore, PHP can be run on all the major operating systems with most servers.

a) Learning PHP is easy

Basic is easy any interpreted language should be easy to learn. Since you are isolated from the system (no pointers to use, no memory to allocate). The other advantage that all modern interpreted languages share is good associative array constructs.

b) Its Performance

While we can build an application that serves millions of pages a day on a server, when we really look at the performance of the language it sucks. We are still orders of magnitude from real performance. Not only that, but since PHP is designed around a single process model our ability to share data structures or connection pool resources is left to native code libraries.

- **The low cost**

There are many languages which are available at very less cost. There are some languages which are available at very less cost like below:

- PHP
- C
- C++ etc.

d) It's Open Source, we can modify it

We can modify it if you need a hole in your head! Technically the point is that it's an open source project and they release patches often. You are point is that the community is actively working out the bugs. So, what any active language is doing this...

Unfortunately, C, C++ and Perl have all "died" at this point and will pretty much remain static at their current functionality.

Its Portability

C is portable; it's just the OS bits that aren't. A lot PHP isn't portable to Windows since people don't use the OS abstractions to avoid some problems.

It has interfaces to a large variety of database systems

PHP supports a large variety of the database.

Support available

Online Support is available for using PHP.

HTML: -

Introduction

HTML or **Hyper Text Markup Language** is the standard markup language used to create web pages.

HTML was created in 1991 by Tim Berners-Lee at CERN in Switzerland. It was designed to allow scientists to display and share their research.

HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets(like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent *empty elements* and so are unpaired, for example <img>. The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*).

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages

CSS

Introduction

CSS tutorial or CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

- CSS stands for Cascading Style Sheet.
- CSS is used to design HTML tags.
- CSS is a widely used language on the web.

- HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

Cascading Style Sheets (**CSS**) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all webpages use CSS style sheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).

CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

With plain HTML you define the colors and sizes of text and tables throughout your pages. If you want to change a certain element you will therefore have to work your way through the document and change it. With CSS you define the colors and sizes in "styles". Then as you write your documents you refer to the styles. Therefore: if you change a certain style it will change the look of your entire site. Another big advantage is that CSS offers much more detailed attributes than plain HTML for defining the look and feel of your site.

JAVASCRIPT

Introduction

JavaScript (**JS**) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side network programming (with Node.js), game development and the creation of desktop and mobile applications.

JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

The application of JavaScript in use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

JavaScript was formalized in the ECMA Script language standard and is primarily used as part of a web browser (client-side JavaScript). This enables programmatic access to objects within a host environment.

JavaScript is the most popular programming language in the world.

It is the language for HTML, for the Web, for computers, servers, laptops, tablets, smart phones, and more.

You can use JavaScript to:

- Change HTML elements
- Delete HTML elements
- Create new HTML elements
- Copy and clone HTML elements

Back end tools

MySQL

MySQL is an open source relational database management system that was originally released in 1995.

MySQL is popular among all databases, and is ranked as the 2nd most popular database, only slightly trailing Oracle Database. Among open source databases, MySQL is the most popular database in use today. Known as one of the most reliable and performative databases out there, it was named after it's founders daughter My, and is known for organizing data into one or more data tables in which data types are related to each other. These relations help structure data, as SQL is a language programmers use for creation, modification and extraction of data from a relational database.

MySQL uses standalone clients that allow users to interact with MySQL, and also to use it with other programs for applications that need relational database capabilities. MySQL's reputation for reliability has led to its inclusion in the popular LAMP stack (Linux, Apache, MySQL, Python/Perl/PHP) and is also used as the default DBMS in popular CMS options like Drupal, Joomla, phpBB, and WordPress.

MySQL is licensed under the GNU General Public License and is also available under several proprietary licenses. When Oracle bought MySQL AB in 2010, Michael "Monty" Widenius, MySQL founder, forked MySQL into a free, open source database called MariaDB -- with the intention of keeping the MariaDB project free and open source forever. MySQL has several versions available, but there are essentially two options: a community version, which is free to use; and paid versions, which include additional functionality, extensions, and support through Oracle. Despite the branding for the paid version, the community version is still considered to be production-ready and is often used in the enterprise. MySQL is a relational database that uses structured query language. Relational databases are a type of database that uses a structure that allows us to identify and access the data in relation to another piece of data inside of the database. This format is often organized as tables. MySQL is a highly scalable product and that scalability can come from several different performance tuning techniques. For starters, you can tune MySQL from the application level. Using a product like Redis which is also supported by OpenLogic, you can cache database queries in an in-memory database. This technique works well with databases containing a high read level and a low write level. An example would be queries for static content on your site.

Another technique is pre-fetching records. This is done to prevent n+1 queries which is a type of query that fetches n amount of records and then runs additional queries for each of the records. Caching columns and query results as variables are two other techniques available.

There are some other optimizations you can make at the MySQL level, such as only returning selected columns in search results, using ENUM for categorical data types, removing unused indexes, breaking up complex queries into smaller, simpler queries and using the slow query log are among other optimizations you can make when scaling. Setting up replica databases used for read operations is a way to horizontally scale your environment and MySQL comes with built-in support for replication.

Five Key Features and Benefits of MySQL

We've outlined many of the benefits for MySQL already. It's versatile, mature, open source, and extensible. But if we had to list five key MySQL features and benefits, they would be the following.

- Easy to Use: MySQL is considered easy to use among RDBMS. It works with basic SQL and, given its maturity and adoption, there is abundant documentation available.
- Secure: MySQL's maturity also lends itself to security. It's regularly updated, has a vibrant developer community, and, because of it's wide adoption within the enterprise, many CVE patches are released before the CVE is announced. These factors combine to make MySQL a stable and secure choice among RDBMS.
- Open Source: The community edition of MySQL is enterprise ready, and supported by a GNU General Public License. For users who want access to equitable proprietary functionality of MySQL without the added price tag, there are other options within the ecosystem — like MariaDB — that can add similar levels of functionality and beyond.
- Scalable: MySQL is highly scalable for an RDBMS, with a wide range of options not covered in this blog that allow for tuning, customizing and enhancing your MySQL experience.
- Reliable: MySQL is reliable — not just from a data perspective, but from a development perspective. It's mature, it has regular releases, patches, and an entrenched developer community that works with it. This makes it a safe choice compared to newer, less mature RDBMS options.

Operating System

As the project is a web application, there are no major operating system requirements. The only requirement is a web browser and proper internet connectivity.

CHAPTER 5 SYSTEM IMPLEMENTATION

5.1 Coding

a_student.php

```
<?php
session_start();
if (isset($_SESSION['teacher'])) {
    ?>
    <!doctype html>
    <html class="no-js " lang="en">

    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=Edge">
        <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport">
        <meta name="description" content="Responsive Bootstrap 4 and web Application ui
kit.">
        <title>LEAP | student</title>
        <link rel="icon" href="favicon.ico" type="image/x-icon"> <!-- Favicon-->
        <link rel="stylesheet" href="assets/plugins/bootstrap/css/bootstrap.min.css">
        <link rel="stylesheet" href="assets/plugins/jvectormap/jquery-jvectormap-
2.0.3.min.css" />
        <link rel="stylesheet" href="assets/plugins/charts-c3/plugin.css" />

        <link rel="stylesheet" href="assets/plugins/morrisjs/morris.min.css" />
        <!-- Custom Css -->
        <link rel="stylesheet" href="assets/css/style.min.css">
    </head>

    <body class="theme-blush">

        <!-- Page Loader -->
        <div class="page-loader-wrapper" id="loader">
            <div class="loader">
                <div class="m-t-30"><img class="zmdi-hc-spin"
src="assets/images/loader.svg" width="48" height="48"
                        alt="Aero"></div>
                <p>Please wait...</p>
            </div>
        </div>

        <!-- Overlay For Sidebars -->
        <div class="overlay"></div>
        <!-- Right Icon menu Sidebar -->
        <div class="navbar-right">
            <ul class="navbar-nav">
                <li><a href="javascript:void(0);" class="js-right-sidebar" title="Setting"><i
                            class="zmdi zmdi-settings zmdi-hc-spin"></i></a></li>
                <li><a href="#" class="mega-menu" title="Log Out" data-toggle="modal"
data-target="#colorModal"><i
                            class="zmdi zmdi-power"></i></a></li>
            </ul>
            </ul>
        </div>

        <!-- Left Sidebar -->
        <aside id="leftsidebar" class="sidebar">
            <div class="navbar-brand">
                <button class="btn-menu ls-toggle-btn" type="button"><i class="zmdi zmdi-
menu"></i></button>
                <a href="a_index.php"><img src="assets/images/logo.svg" width="25"
alt="Aero"><span
                        class="m-l-10">LEAP</span></a>
            </div>
            <div class="menu">
                <ul class="list">
                    <li>
                        <div class="user-info">
                            <a class="image" href="a_profile.php"><img
src="assets/images/profile_av.jpg" alt="User"></a>
                            <div class="detail">
                                <h4>
                                    <?php echo $_SESSION['tname']; ?>
                                </h4>
                                <small>Super User</small>
                            </div>
                        </div>
                    </li>
                    <li class="open"><a href="a_teacher.php"><i class="zmdi zmdi-account-
add"></i><span>Teacher</span></a>
                    </li>
                    <li class="open"><a href="a_student.php"><i class="zmdi zmdi-accounts-
alt"></i><span>Student</span></a>
                    </li>
                    <li class="open"><a href="a_classes.php"><i class="zmdi zmdi-
face"></i><span>My classes</span></a></li>
                </ul>
            </div>
        </aside>

        <!-- Right Sidebar -->
        <aside id="rightsidebar" class="right-sidebar">
            <div class="tab-content">
                <div class="tab-pane active" id="setting">
                    <div class="slim_scroll">
                        <div class="card">
                            <h6>Theme Option</h6>
                            <div class="light_dark">
                                <div class="radio">
                                    <input type="radio" name="radio1" id="lighttheme" value="light"
checked="">
                                    <label for="lighttheme">Light Mode</label>
                                </div>
                                <div class="radio mb-0">
                                    <input type="radio" name="radio1" id="darktheme"
value="dark">
                                    <label for="darktheme">Dark Mode</label>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </aside>


        <section class="content">
            <div class="body_scroll">
                <div class="block-header">
                    <div class="row">
                        <div class="col-lg-7 col-md-6 col-sm-12">
                            <h2>Student Management</h2>
                            <ul class="breadcrumb">
                                <li class="breadcrumb-item"><a href="admin_dash.php"><i
class="zmdi zmdi-home"></i> Home</a>
                                </li>
                                <li class="breadcrumb-item active">Student organizer</li>
                            </ul>
```

```
              <button class="btn btn-primary btn-icon mobile_menu"
type="button"><i
                  class="zmdi zmdi-sort-amount-desc"></i></button>
            </div>
            <div class="col-lg-5 col-md-6 col-sm-12">
              <a class="btn btn-primary float-right"
href="a_addviaexcel.php">Upload via Excel</a>
            </div>
          </div>
        </div>
        <div class="container-fluid">
          <!-- Example Tab -->
          <div class="row clearfix">
            <div class="col-sm-12">
              <div class="card">
                <div class="header">
                  <h2><strong>STUDENT</strong> ORGANIZER</h2>

                </div>
                <div class="body">

                  <!-- Nav tabs -->
                  <ul class="nav nav-tabs p-0 mb-3">
                    <li class="nav-item"><a class="nav-link active" data-
toggle="tab"
                        href="#home">REGISTER</a></li>
                    <li class="nav-item"><a class="nav-link" data-toggle="tab"
href="#profile"
                        id="view">VIEW</a></li>
                  </ul>
                  <!-- Tab panes -->
                  <div class="tab-content">
                    <div role="tabpanel" class="tab-pane in active" id="home">
                      <div class="body">
                        <form id="myform" method="POST">
                          <div class="form-group form-float">
                            <input type="text" class="form-control"
placeholder="Full Name"
                              name="name" id="as_name">
                          </div>
                          <div class="form-group form-float">
                            <input type="email" class="form-control"
placeholder="Email"
```

```
                                                            name="email" id="as_email">
                                                    </div>
                                                    <div class="form-group form-float">
                                                        <input type="text" class="form-control"
                                                            placeholder="Admission Number" name="adm"
id="as_adm">
                                                    </div>
                                                    <div class="form-group form-float">
                                                        <input type="text" class="form-control"
placeholder="Mobile"
                                                            name="mob" id="as_phn" pattern="[0-9]{10}"
minlength="10">
                                                    </div>
                                                    <div class="form-group form-float">
                                                        <input type="text" class="form-control"
                                                            placeholder="year of admission" name="yoa"
id="as_yoa">
                                                    </div>
                                                    <div class="form-group form-float">
                                                        <select class="form-control show-tick ms select2"
                                                            data-placeholder="Select" name="batch"
id="as_batch">
                                                            <option disabled selected hidden>Batch</option>
                                                            <option>A</option>
                                                            <option>B</option>
                                                            <option>C</option>
                                                        </select>
                                                    </div>
                                                    <div class="form-group form-float">
                                                        <input type="password" class="form-control"
placeholder="Password"
                                                            name="pwd" id="as_pwd">
                                                    </div>
                                                    <button class="btn btn-raised btn-primary waves-effect"
type="submit"
                                                        id="register">REGISTER</button>
                                                </form>
                                            </div>
                                        </div>
                                        <div role="tabpanel" class="tab-pane" id="profile">
                                            <div class="row clearfix">

                                                <div class="col-lg-12">
                                                    <div class="card">
```

```
                                                        <div class="body">
                                                            <div class="table-responsive table-hover table-
striped"
                                                                id="live_data">

                                                            </div>
                                                        </div>
                                                    </div>
                                                </div>
                                            </div>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </section>
        <!--Delete student modal-->
        <div class="modal fade" id="delstudentmodel" tabindex="-1" role="dialog">
            <div class="modal-dialog" role="document">
                <div class="modal-content bg-teal">
                    <div class="modal-header">
                        <h4 class="title" id="defaultModalLabel">DELETE</h4>
                    </div>
                    <div class="modal-body text-light">Are you sure that you want to delete the
student details</div>
                    <div class="modal-footer">
                        <input type="hidden" id="stid">
                        <button type="button" class="btn btn-link waves-effect text-light" data-
dismiss="modal"
                            id="delstudentconfirm">CONFIRM</button>
                        <button type="button" class="btn btn-link waves-effect text-light"
                            data-dismiss="modal">CLOSE</button>
                    </div>
                </div>
            </div>
        </div>

        <!--update student modal-->
        <div class="modal fade" id="defaultModal" tabindex="-1" role="dialog">
            <div class="modal-dialog" role="document">
                <div class="modal-content">
```

```
          <div class="modal-header">
            <h4 class="title" id="defaultModalLabel">Edit Student Details</h4>
          </div>
          <div class="modal-body">
            <form id="form_validation" method="POST" role="form">
              <div class="form-group form-float">
                <input type="hidden" class="form-control" id="sid" readonly>
                <input type="text" class="form-control" id="admno"
placeholder="Admisiion Number" required>
              </div>
              <div class="form-group form-float">
                <input type="text" class="form-control" id="sname"
placeholder="Name" required>
              </div>
              <div class="form-group form-float">
                <input type="text" class="form-control" placeholder=" Email"
id="semail" required>
              </div>
              <div class="form-group form-float">
                <input type="text" class="form-control" placeholder="Batch"
id="sbatch" required>
              </div>
              <div class="form-group form-float">
                <input type="text" class="form-control" placeholder="Mobile"
id="smob" required>
              </div>
              <div class="form-group form-float">
                <input type="text" class="form-control" placeholder="Year of
admission" id="syoa" required>
              </div>
              <div class="modal-footer">
                <button type="button" data-dismiss="modal" id="Updatestudent"
                  class="btn btn-success waves-effect">Update</button>
                <button type="button" class="btn btn-danger waves-effect"
id="closem"
                  data-dismiss="modal">CLOSE</button>
              </div>
            </form>
          </div>
        </div>
      </div>
    </div>

    <!--Logout modal-->
```

```
        <div class="modal fade" id="colorModal" tabindex="-1" role="dialog">
            <div class="modal-dialog" role="document">
                <div class="modal-content bg-red">
                    <div class="modal-header">
                        <h4 class="title" id="defaultModalLabel">LOG OUT</h4>
                    </div>
                    <div class="modal-body text-light">Are you sure that you want to exit the
current session</div>
                    <div class="modal-footer">
                        <button type="button" class="btn btn-link waves-effect text-light"
id="logout">LOG OUT</button>
                        <button type="button" class="btn btn-link waves-effect text-light"
                            data-dismiss="modal">CLOSE</button>
                    </div>
                </div>
            </div>
        </div>

        <script src="assets/bundles/libscripts.bundle.js"></script> <!-- Lib Scripts Plugin Js
-->
        <script src="assets/bundles/vendorscripts.bundle.js"></script> <!-- Lib Scripts
Plugin Js -->
        <!-- Jquery DataTable Plugin Js -->
        <script src="assets/bundles/datatablescripts.bundle.js"></script>
        <script src="assets/plugins/jquery-
datatable/buttons/dataTables.buttons.min.js"></script>
        <script src="assets/plugins/jquery-
datatable/buttons/buttons.bootstrap4.min.js"></script>
        <script src="assets/plugins/jquery-
datatable/buttons/buttons.colVis.min.js"></script>
        <script src="assets/plugins/jquery-datatable/buttons/buttons.flash.min.js"></script>
        <script src="assets/plugins/jquery-datatable/buttons/buttons.html5.min.js"></script>
        <script src="assets/plugins/jquery-datatable/buttons/buttons.print.min.js"></script>

        <script src="assets/js/pages/tables/jquery-datatable.js"></script>

        <script src="assets/js/pages/ui/sweetalert.js"></script>
        <script src="assets/js/pages/ui/notifications.js"></script> <!-- Custom Js -->
        <script src="assets/plugins/bootstrap-notify/bootstrap-notify.js"></script> <!--
Bootstrap Notify Plugin Js -->
        <script src="assets/plugins/sweetalert/sweetalert.min.js"></script> <!-- SweetAlert
Plugin Js -->

        <script src="../../js/regStudent.js"></script>
        <script src="../../js/student.js"></script>
        <script src="../../js/logout.js"></script>
        <script>
            //Check Email Exist
            $(document).ready(function () {
                console.log("hai");
                $("#as_email").focusout(function () {
                    var username = $("#as_email").val().trim();
                    const button1 = document.getElementById("register"); // assuming the button
is assigned an id named "button"
                    if (username != "") {
                        $.ajax({
                            url: '../../php/checkemailexist.php',
                            type: 'post',
                            data: { username: username },
                            success: function (response) {
                                var data = JSON.parse(response);
                                if (data.count >= 1) {
                                    showNotification("red", "Email already exists", "bottom", "right",
"bg-red", "");
                                    button1.disabled = true;
                                    return false
                                }
                                else {
                                    button1.disabled = false;
                                }
                            }
                        });
                    }
                });
            });
        </script>

    </html>
    <?php
} else {
    header("Location: ../../index.php");
}
?>
```

addexcel.js

```
$(document).ready(function (e) {
    $("#exelform").on('submit', function (e) {

        e.preventDefault();
        $.ajax({
            type: 'POST',
            url: '../../php/addviaexcelpro.php',
            data: new FormData(this),

            contentType: false,
            cache: false,
            processData: false,
            success: function (response) {

                if (response == 1) {

                    showNotification("alert-success", "Details uploaded", "bottom", "right", "",
"")
                    document.getElementById("sub").disabled = true;
                    $("#file").val(");


                } else if (response == 2) {


                    showNotification("alert-danger", "Please choose a file", "bottom", "right", "",
"")
                } else {
                    console.log(response)
                    showNotification("alert-danger", "Something wrong", "bottom", "right", "",
"")
                }
            },
            error: function (xhr, ajaxOptions, thrownError) {
                console.log(xhr.status);
                console.log(thrownError);
            }

        });
    });
});
```

addviaexcelpro.php

```
<?php
  $url='localhost';
  $username='root';
  $password=";
  $conn=mysqli_connect($url,$username,$password,"lms");


  $flag=0;

  $file = $_FILES['file']['tmp_name'];
  if ($file != ") {
   $handle = fopen($file, "r");
   $c = 1;
   while(($filesop = fgetcsv($handle, 1000, ",")) !== false)
   {
    $sql = mysqli_query($conn,"SELECT * FROM `student`");
    $name = $filesop[1];
    $email = $filesop[2];
    $adm_no= $filesop[0];
    $yoa=$filesop[4];
    $batch=$filesop[3];
    while($row=mysqli_fetch_assoc($sql))
    {

      if($email == $row['s_email'])
      $flag=1;

    }
    if($flag == 0)
    {
                                   $sql = "INSERT INTO student(s_name,s_email,s_admno,s_batch,s_yoa)values('$name','$email','$adm_no','$batch','$yoa')";
    $stmt = mysqli_prepare($conn,$sql);
    if(mysqli_stmt_execute($stmt))
    $m = 1;
    $c = $c + 1;
    }
    $flag=0;
  }
  echo 1;
}
```

```
else
{
  echo 2;
}
?>
```

a_notes.php

```
<?php
$con = mysqli_connect("localhost", "root", "", "lms");
session_start();
$subid = base64_decode($_REQUEST['sub_id']);
$sql = "SELECT * from `subject` where sub_id = $subid";
$res = mysqli_query($con, $sql);
$row = mysqli_fetch_assoc($res);
if (isset($_SESSION['teacher'])) {
    ?>
    <!doctype html>
    <html class="no-js " lang="en">

    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=Edge">
        <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport">
        <meta name="description" content="Responsive Bootstrap 4 and web Application ui
kit.">

        <title>LEAP | Notes</title>
        <link rel="icon" href="favicon.ico" type="image/x-icon">
        <!-- Favicon-->
        <link rel="stylesheet" href="assets/plugins/bootstrap/css/bootstrap.min.css">
        <!-- JQuery DataTable Css -->
        <link rel="stylesheet" href="assets/plugins/jquery-
datatable/dataTables.bootstrap4.min.css">
        <!-- Custom Css -->
        <link rel="stylesheet" href="assets/css/style.min.css">
    </head>

    <body class="theme-blush">
        <!-- Page Loader -->
        <div class="page-loader-wrapper" id="loader">
            <div class="loader">
        <div class="m-t-30"><img class="zmdi-hc-spin"
src="assets/images/loader.svg" width="48" height="48"
            alt="Aero"></div>
        <p>Please wait...</p>
      </div>
    </div>

    <div class="overlay"></div>

    <!-- Right Icon menu Sidebar -->
    <div class="navbar-right">
      <ul class="navbar-nav">

        <li><a href="javascript:void(0);" class="js-right-sidebar" title="Setting"><i
              class="zmdi zmdi-settings zmdi-hc-spin"></i></a></li>
        <li><a href="#" class="mega-menu" title="Log Out" data-toggle="modal"
data-target="#colorModal"><i
              class="zmdi zmdi-power"></i></a></li>
      </ul>
    </div>

    <!-- Left Sidebar -->
    <aside id="leftsidebar" class="sidebar">
      <div class="navbar-brand">
        <button class="btn-menu ls-toggle-btn" type="button"><i class="zmdi zmdi-
menu"></i></button>
        <a href="a_index.php"><img src="assets/images/logo.svg" width="25"
alt="Aero"><span
            class="m-l-10">LEAP</span></a>
      </div>
      <div class="menu">
        <ul class="list">
          <li>
            <div class="user-info">
              <a class="image" href="a_profile.php"><img
src="assets/images/profile_av.jpg" alt="User"></a>
              <div class="detail">
                <h4>
                  <?php echo $_SESSION['tname']; ?>
                </h4>
                <small>Super User</small>
              </div>
            </div>
          </li>
          <li class="open"><a href="a_teacher.php"><i class="zmdi zmdi-account-
add"></i><span>Teacher</span></a>
          </li>
          <li class="open"><a href="a_student.php"><i class="zmdi zmdi-accounts-
alt"></i><span>Student</span></a>
          </li>
          <li class="open"><a href="a_classes.php"><i class="zmdi zmdi-
face"></i><span>My classes</span></a></li>

              <li><a href="a_viewAssignment.php?sub_id=<?php echo
base64_encode($subid) ?>"><i class="zmdi zmdi-
assignment"></i><span>Assignments</span></a>
              </li>
              <li><a href="a_resources.php?sub_id=<?php echo
base64_encode($subid) ?>"><i class="zmdi zmdi-assignment-o"></i><span>Question
papers</span></a>
              </li>
              <li><a href="a_addmarks.php?sub_id=<?php echo
base64_encode($subid)?>"><i class="zmdi zmdi-border-color"></i><span>Internal
Marks</a></span></li>

          </li>

        </ul>

      </div>
    </aside>

    <!-- Right Sidebar -->
    <aside id="rightsidebar" class="right-sidebar">

      <div class="tab-content">
        <div class="tab-pane active" id="setting">
          <div class="slim_scroll">
            <div class="card">
              <h6>Theme Option</h6>
              <div class="light_dark">
                <div class="radio">
                  <input type="radio" name="radio1" id="lighttheme" value="light"
checked="">
                  <label for="lighttheme">Light Mode</label>
                </div>
                <div class="radio mb-0">
```

```
                                    <input type="radio" name="radio1" id="darktheme"
value="dark">
                                    <label for="darktheme">Dark Mode</label>
                                </div>
                            </div>
                        </div>

                    </div>
                </div>
            </div>
        </aside>

        <input type="hidden" id="subid" name="subid" value="<?php echo $subid ?>" />
        <section class="content file_manager">
            <div class="body_scroll">
                <div class="block-header">
                    <div class="row">
                        <div class="col-lg-7 col-md-6 col-sm-12">
                            <h2>Notes</h2>
                            <ul class="breadcrumb">
                                <li class="breadcrumb-item"><a href="a_index.php"><i class="zmdi
zmdi-home"></i> Home</a>
                                </li>
                                <li class="breadcrumb-item"><a href="viewSubject.php?c_id=<?php
echo base64_encode($row['sub_cid']);?>">Subjects</a></li>
                                <li class="breadcrumb-item active">Notes</li>
                            </ul>
                            <button class="btn btn-primary btn-icon mobile_menu"
type="button"><i
                                    class="zmdi zmdi-sort-amount-desc"></i></button>
                        </div>
                        <div class="col-lg-5 col-md-6 col-sm-12">
                            <button class="btn btn-primary btn-icon float-right
right_icon_toggle_btn" type="button"><i
                                    class="zmdi zmdi-arrow-right" ></i></button>
                            <button class="btn btn-success btn-icon float-right" type="button"
id="nupload" data-sub_id="<?php echo $subid?>" data-toggle="modal"
                                data-target="#noteUploadModal"><i
                                    class="zmdi zmdi-upload"></i></button>
                        </div>
                    </div>
                </div>
                <div class="container-fluid">
                    <div class="row clearfix">
```

```
                        <div class="col-lg-12">
                            <div class="card">
                                <div class="tab-content">
                                    <div class="tab-pane active" id="doc">
                                        <div class="row clearfix">
                                            <?php
                                            include "../../php/config.php";
                                            $tid = $_SESSION['tid'];
                                            $query = "select * from notes where n_subid='$subid' order
by n_id desc ";
                                            $result = mysqli_query($con, $query);
                                            $count = 1;
                                            while ($row = mysqli_fetch_array($result)) {
                                                ?>
                                                <div class="col-lg-3 col-md-4 col-sm-12">
                                                    <div class="card">
                                                        <div class="file">
                                                            <a href="#" data-toggle="modal"
                                data-target="#colorModalDeleteNote" id="delNote" data-
n_id="<?php echo $row['n_id'];?>">
                                                                <div class="hover">
                                                                    <button type="button"
                                                                        class="btn btn-icon btn-icon-mini btn-round
btn-danger" >
                                                                        <i class="zmdi zmdi-delete"></i>
                                                                    </button>
                                                                </div>
                                                                <div class="icon">
                                                                    <i class="zmdi zmdi-file-text"></i>
                                                                </div>
                                                                <div class="file-name">
                                                                    <p class="m-b-5 text-muted"><a href="#"
onclick="window.open('../../php/notes_folder/<?php echo $row['n_fname']; ?>', '_blank',
'fullscreen=yes','targetWindow',width=1100,height=2000); return false;"><?php echo
$row['n_name']?></a></p>
                                                                    <small>Date of upload <span class="date text-
muted"><?php echo date("d/m/Y", strtotime($row['n_date']));?></span></small>
                                                                </div>
                                                            </a>
                                                        </div>
                                                    </div>
                                                </div>
                                                <?php
                                            }
                                            ?>
                                            </div>

                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </section>

        <!-- Jquery DataTable Plugin Js -->
        <script src="assets/bundles/datatablescripts.bundle.js"></script>

        <script src="assets/js/pages/ui/notifications.js"></script> <!-- Custom Js -->
        <script src="assets/plugins/bootstrap-notify/bootstrap-notify.js"></script>
        <!-- Bootstrap Notify Plugin Js -->
        <script src="assets/js/pages/tables/footable.js"></script><!-- Custom Js -->

    </html>

    <!--Note upload modal-->

    <div class="modal fade" id="noteUploadModal" tabindex="-1" role="dialog">
        <div class="modal-dialog modal-lg" role="document">
            <div class="modal-content">
                <div class="modal-header">
                    <h4 class="title" id="largeModalLabel">Note upload</h4>
                </div>
                <div class="modal-body">
                    <form id="form_validation" method="POST" enctype="multipart/form-
data">
                    <div class="form-group form-float">
                        <input type="hidden" id="n_subid" class="form-control" value=<?php
echo $subid; ?>>
                            <input type="file" id="n_file" name="file" accept=".pdf" class="form-
control file" onchange="fileValidation()" placeholder="Choose file"
                            required>
                        </div>

                </div>
                <div class="modal-footer">
                    <button type="button" id="upload_note" data-dismiss="modal"
```

```
                        class="btn btn-default btn-round waves-effect"
disabled>UPLOAD</button>
                    <button type="button" class="btn btn-danger waves-effect" id="closem"
                        data-dismiss="modal">CLOSE</button>
                </div>
                </form>
            </div>
        </div>
    </div>

    <!--Delete assignment modal-->
    <div class="modal fade" id="colorModalDeleteNote" tabindex="-1" role="dialog">
        <div class="modal-dialog" role="document">
            <div class="modal-content bg-teal">
                <div class="modal-header">
                    <h4 class="title" id="defaultModalLabel">DELETE</h4>
                </div>
                <div class="modal-body text-light">Are you sure that you want to delete the
note</div>
                <div class="modal-footer">
                    <input type="hidden" id="ass_id">
                    <button type="button" class="btn btn-link waves-effect text-light" data-
dismiss="modal"
                        id="delnoteconfirm">CONFIRM</button>
                    <button type="button" class="btn btn-link waves-effect text-light" data-
dismiss="modal">CLOSE</button>
                </div>
            </div>
        </div>
    </div>

    <!--Logout modal-->
    <div class="modal fade" id="colorModal" tabindex="-1" role="dialog">
        <div class="modal-dialog" role="document">
            <div class="modal-content bg-red">
                <div class="modal-header">
                    <h4 class="title" id="defaultModalLabel">LOG OUT</h4>
                </div>
                <div class="modal-body text-light">Are you sure that you want to exit the
current session</div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-link waves-effect text-light"
id="logout">LOG OUT</button>
```

```
                    <button type="button" class="btn btn-link waves-effect text-light" data-dismiss="modal">CLOSE</button>
                </div>
            </div>
        </div>
    </div>

    <script src="assets/bundles/libscripts.bundle.js"></script>
    <!-- Lib Scripts Plugin Js ( jquery.v3.2.1, Bootstrap4 js) -->
    <script src="assets/bundles/vendorscripts.bundle.js"></script> <!-- slimscroll, waves Scripts Plugin Js -->
    <script src="assets/bundles/mainscripts.bundle.js"></script>
    <script src=../../js/notes.js></script>
    <script src="assets/js/pages/ui/notifications.js"></script> <!-- Custom Js -->
        <script src="assets/plugins/bootstrap-notify/bootstrap-notify.js"></script> <!-- Bootstrap Notify Plugin Js -->
        <script src="assets/plugins/sweetalert/sweetalert.min.js"></script> <!-- SweetAlert Plugin Js -->
    <script src="../../js/logout.js"></script>
    <script>
        function fileValidation(){
    var fileInput = document.getElementById('n_file');
    var filePath = fileInput.value;
    var allowedExtensions = /(\.pdf)$/i;
    if(!allowedExtensions.exec(filePath)){
        showNotification("alert-danger", "Please Select pdf file ", "bottom", "right", "", "")
        fileInput.value = '';
        return false;
    }
    else
    document.getElementById('upload_note').disabled=false;
}
        </script>


    <?php
} else {
    header("Location: ../index.php");
}
?>
```

notes.js

```
 $('#upload_note').on('click', function() {

   var file_data = $('#n_file').prop('files')[0];
   var form_data = new FormData();
   var sid =  $("#n_subid").val().trim();
   form_data.append('as_file', file_data);
   form_data.append('as_subid', sid);
   document.getElementById("loader").style.display='block';

   $.ajax({
       url: '../../php/addnotes.php', // <-- point to server-side PHP script
         // <-- what to expect back from the PHP script, if anything
       cache: false,
       contentType: false,
       processData: false,
       data: form_data,
       type: 'post',
       success: function(response){
         console.log(response);
               if (response == 1) {
                 location.reload();

               }
               else {
                 showNotification("alert-error", "Note upload failed", "bottom", "right", "", "")
               }

       }
     });
});


 $("#delnoteconfirm").click(function () {
   var delid = $('#ass_id').val().trim();

   //debugger
   $.ajax({
     url: '../../php/delNote.php',
     type: 'post',
     data: { id: delid },
     success: function (response) {
       if (response == 1) {
         location.reload();

       }
       else {
         showNotification("alert-error", "Assignment removal failed", "bottom", "right", "",
"")
       }
     },
     error: function (xhr, ajaxOptions, thrownError) {
       console.log(xhr.status);
       console.log(thrownError);
     }
   });

 });


 //delete
 $(document).on('click', '#delNote', function () {

   var a_id = $(this).data("n_id");

   document.getElementById("ass_id").value = a_id;


 });
 //updatebutton click
 $(document).on('click', '#updateassbtn', function () {

   var a_id = $(this).data("id1");
   var assname = $(this).data("id2");
   var assduedate = $(this).data("id5");
   var t_doc = $(this).data("id3");
   var t_des = $(this).data("id4");

   document.getElementById("updt_ass_id").value = a_id;
   document.getElementById("assname").value = assname;
   document.getElementById("date_picker").value = assduedate;
   document.getElementById("des").value = t_des;

 });
 //confirm update btn
 $("#confirmupdateassignment").click(function () {
   var assid = document.getElementById("updt_ass_id").value;
   var assname = document.getElementById("assname").value;
   var assduedate = document.getElementById("date_picker").value;
   var des = document.getElementById("des").value;

   var file_data = $('#file_up').prop('files')[0];
   var form_data = new FormData();

   form_data.append('file_up', file_data);
   form_data.append('topic', assname);
   form_data.append('assid', assid);
   form_data.append('ddate', assduedate);
   form_data.append('description', des);
   //debugger
   $.ajax({
     url: '../../php/updateassignment.php',
     cache: false,
     contentType: false,
     processData: false,
     data: form_data,
     type: 'post',
     success: function(response){
             if (response == 1) {
               showNotification("alert-success", "Assignment updated", "bottom", "right", "",
"")

             }
             else {
               showNotification("alert-success", "Assignment updated", "bottom", "right", "",
"")
             }

     }
   });
 });
```

addnotes.php

```
<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require './PHPMailer/src/Exception.php';
require './PHPMailer/src/PHPMailer.php';
require './PHPMailer/src/SMTP.php';

include "config.php";
if(isset($_FILES['as_file']['name']))
{
$sid = $_POST['as_subid'];
$file_name = $_FILES['as_file']['name'];

$file_tmp = explode(".", $_FILES["as_file"]["name"]);
$newfilename = round(microtime(true)) . '.' . end($file_tmp);
$_FILES["as_file"]["name"]=$newfilename;
$targetfolder = "notes_folder/";
$targetfolder = $targetfolder . basename($_FILES['as_file']['name']);
move_uploaded_file($_FILES['as_file']['tmp_name'], $targetfolder);


$sql = "INSERT INTO notes (n_subid,n_date,n_name,n_fname) VALUES
('$sid',NOW(),'$file_name','$newfilename')";
$result = mysqli_query($con, $sql);

$s = mysqli_query($con, "SELECT sub_name FROM `subject` WHERE sub_id =
'$sid'");
$r = mysqli_fetch_assoc($s);
$content = '<!DOCTYPE html>
<html lang="en" xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-
microsoft-com:vml" xmlns:o="urn:schemas-microsoft-com:office:office">
<head>
   <meta charset="utf-8"> <!-- utf-8 works for most cases -->
   <meta name="viewport" content="width=device-width"> <!-- Forcing initial-scale
shouldnt be necessary -->
   <meta http-equiv="X-UA-Compatible" content="IE=edge"> <!-- Use the latest (edge)
version of IE rendering engine -->
   <meta name="x-apple-disable-message-reformatting">  <!-- Disable auto-scale in iOS
10 Mail entirely -->
   <title></title> <!-- The title tag shows in email notifications, like Android 4.4. -->

   <link
href="https://fonts.googleapis.com/css?family=Poppins:200,300,400,500,600,700"
rel="stylesheet">

   <!-- CSS Reset : BEGIN -->

<style>
html,
body {
margin: 0 auto !important;
padding: 0 !important;
height: 100% !important;
width: 100% !important;
background: #f1f1f1;
}

* {
-ms-text-size-adjust: 100%;
-webkit-text-size-adjust: 100%;
}

div[style*="margin: 16px 0"] {
margin: 0 !important;
}

table,
td {
mso-table-lspace: 0pt !important;
mso-table-rspace: 0pt !important;
}

table {
border-spacing: 0 !important;
border-collapse: collapse !important;
table-layout: fixed !important;
margin: 0 auto !important;
}

img {
-ms-interpolation-mode:bicubic;
}

a {
text-decoration: none;
}


*[x-apple-data-detectors],  /* iOS */
.unstyle-auto-detected-links *,
.aBn {
border-bottom: 0 !important;
cursor: default !important;
color: inherit !important;
text-decoration: none !important;
font-size: inherit !important;
font-family: inherit !important;
font-weight: inherit !important;
line-height: inherit !important;
}

.a6S {
display: none !important;
opacity: 0.01 !important;
}

.im {
color: inherit !important;
}

img.g-img + div {
display: none !important;
}


@media only screen and (min-device-width: 320px) and (max-device-width: 374px) {
u ~ div .email-container {
min-width: 320px !important;
}
}
@media only screen and (min-device-width: 375px) and (max-device-width: 413px) {
u ~ div .email-container {
min-width: 375px !important;
}
}
@media only screen and (min-device-width: 414px) {
u ~ div .email-container {
min-width: 414px !important;
}
}


</style>

<!-- CSS Reset : END -->

<!-- Progressive Enhancements : BEGIN -->
```

```
<style>

.primary{
background: #17bebb;
}
.bg_white{
background: #ffffff;
}
.bg_light{
background: #f7fafa;
}
.bg_black{
background: #000000;
}
.bg_dark{
background: rgba(0,0,0,.8);
}
.email-section{
padding:2.5em;
}

/*BUTTON*/
.btn{
padding: 10px 15px;
display: inline-block;
}
.btn.btn-primary{
border-radius: 5px;
background: #17bebb;
color: #ffffff;
}
.btn.btn-white{
border-radius: 5px;
background: #ffffff;
color: #000000;
}
.btn.btn-white-outline{
border-radius: 5px;
background: transparent;
border: 1px solid #fff;
color: #fff;
}
.btn.btn-black-outline{
border-radius: 0px;
background: transparent;
border: 2px solid #000;
color: #000;
font-weight: 700;
}
.btn-custom{
color: rgba(0,0,0,.3);
text-decoration: underline;
}

h1,h2,h3,h4,h5,h6{
font-family: "Poppins", sans-serif;
color: #000000;
margin-top: 0;
font-weight: 400;
}

body{
font-family: "Poppins", sans-serif;
font-weight: 400;
font-size: 15px;
line-height: 1.8;
color: rgba(0,0,0,.4);
}

a{
color: #17bebb;
}

table{
}
/*LOGO*/

.logo h1{
margin: 0;
}
.logo h1 a{
color: #17bebb;
font-size: 24px;
font-weight: 700;
font-family: "Poppins", sans-serif;
}

/*HERO*/
```

```
.hero{
position: relative;
z-index: 0;
}

.hero .text{
color: rgba(0,0,0,.3);
}
.hero .text h2{
color: #000;
font-size: 34px;
margin-bottom: 0;
font-weight: 200;
line-height: 1.4;
}
.hero .text h3{
font-size: 24px;
font-weight: 300;
}
.hero .text h2 span{
font-weight: 600;
color: #000;
}

.text-author{
bordeR: 1px solid rgba(0,0,0,.05);
max-width: 50%;
margin: 0 auto;
padding: 2em;
}
.text-author img{
border-radius: 50%;
padding-bottom: 20px;
}
.text-author h3{
margin-bottom: 0;
}
ul.social{
padding: 0;
}
ul.social li{
display: inline-block;
margin-right: 10px;
}
```

```
/*FOOTER*/

.footer{
border-top: 1px solid rgba(0,0,0,.05);
color: rgba(0,0,0,.5);
}
.footer .heading{
color: #000;
font-size: 20px;
}
.footer ul{
margin: 0;
padding: 0;
}
.footer ul li{
list-style: none;
margin-bottom: 10px;
}
.footer ul li a{
color: rgba(0,0,0,1);
}


@media screen and (max-width: 500px) {


}
</style>
</head>

<body width="100%" style="margin: 0; padding: 0 !important; mso-line-height-rule:
exactly; background-color: #f1f1f1;">
  <center style="width: 100%; background-color: #f1f1f1;">
    <div style="display: none; font-size: 1px;max-height: 0px; max-width: 0px; opacity: 0;
overflow: hidden; mso-hide: all; font-family: sans-serif;">
      &zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nb
sp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbs
p;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbsp;&zwnj;&nbs
p;
    </div>
    <div style="max-width: 600px; margin: 0 auto;" class="email-container">
      <!-- BEGIN BODY -->
      <table align="center" role="presentation" cellspacing="0" cellpadding="0" border="0" width="100%" style="margin: auto;">
        <tr>
          <td valign="top" class="bg_white" style="padding: 1em 2.5em 0 2.5em;">
            <table role="presentation" border="0" cellpadding="0" cellspacing="0" width="100%">
              <tr>
                <td class="logo" style="text-align: center;">
                  <h1><a href="#">NOTES</a></h1>
                </td>
              </tr>
            </table>
          </td>
        </tr><!-- end tr -->
        <tr>
          <td valign="middle" class="hero bg_white" style="padding: 2em 0 4em 0;">
            <table role="presentation" border="0" cellpadding="0" cellspacing="0" width="100%">
              <tr>
                <td style="padding: 0 2.5em; text-align: center; padding-bottom: 3em;">
                  <div class="text">
                    <h2>'.$r['sub_name'].' notes has been uploaded . Please do check.</h2>
                  </div>
                </td>
              </tr>
              <tr>
                <td style="text-align: center;">
                  <div class="text-author">
                    <img src="https://png.pngtree.com/png-clipart/20190920/original/pngtree-sticky-paper-sticky-sticky-note-png-image_4664401.jpg" alt="" style="width: 100px; max-width: 600px; height: auto; margin: auto; display: block;">
                   </div>
                </td>
              </tr>
            </table>
          </td>
        </tr><!-- end tr -->
      <!-- 1 Column Text + Button : END -->
      </table>

    </div>
  </center>
</body>
</html>';
```

```
$sql_query = "SELECT * FROM `student` where `s_batch` = (SELECT c_batch from
`class` where c_id = (SELECT sub_cid from `subject` where sub_id = '$sid')) and `s_yoa`
= (SELECT c_yoa from `class` where c_id = (SELECT sub_cid from `subject` where
sub_id = '$sid'))";
$res = mysqli_query($con,$sql_query);
while($row = mysqli_fetch_assoc($res))
{
// echo $row['email'];
$mail = new PHPMailer(true);
  $mail->isSMTP();
  $mail->Host = 'smtp.gmail.com';
  $mail->SMTPAuth = true;
  $mail->Username = 'ericsantony123@gmail.com';
  $mail->Password = 'smnjgoudahkhjgvf';
  $mail->Port = 465;
  $mail->isHTML(true);
  $mail->setFrom('ericsantony123@gmail.com', 'LEAP');
  $mail->addAddress($row['s_email']);
  $mail->Subject = ('NOTES');
  $mail->Body = ($content);
  $mail->send();
}

if ($result)
  echo "1";
else
  echo mysqli_error($con);
}
else
{
    echo "3";
}

?>
```

delnotes.php

```
<?php
include "config.php";
$aid = $_POST['id'];
$s="SELECT * from notes where n_id='$aid'";
$result=mysqli_query($con,$s);
$row = mysqli_fetch_array($result);

$dir = $_SERVER['DOCUMENT_ROOT'] . "/mainProject/php/notes_folder";
```

```
$data = $row['n_fname'];

$sql = "DELETE FROM `notes` WHERE n_id='$aid'";
$result = mysqli_query($con, $sql);

unlink($dir . '/' . $data);

if ($result)
  echo "1";
else
  echo mysqli_error($con);
?>
```

CHAPTER 6 SYSTEM TESTING

6.1 Introduction

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During testing, the program is executed with a set of test cases, and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected to perform.

Unit Testing:

Unit Testing is done on individual modules as they are completed and become executable. It is confined only to the designer's requirements. Each module can be tested using the following two Strategies:

- Black Box Testing:

In this strategy, some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been used to find errors in the following categories:

  - Incorrect or missing functions
  - Interface errors
  - Errors in data structure or external database access
  - Performance errors
  - Initialization and termination errors.

In this testing, only the output is checked for correctness. The logical flow of the data is not checked.

- White Box testing:

In this the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all the cases. It has been used to generate the test cases

in the following cases:

  - Guarantee that all independent paths have been Executed.
  - Execute all logical decisions on their true and false Sides.
  - Execute all loops at their boundaries and within their operational bounds.
  - Execute internal data structures to ensure their validity.

Integrating Testing:

Integration testing ensures that software and subsystems work together as a whole. It tests the interface of all the modules to make sure that the modules behave properly when integrated. The purpose of unit testing is to determine that each independent module is correctly implemented. This gives little chance to determine that the interface between modules is also correct, and for this reason integration testing must be performed. One specific target of integration testing is the interface: whether parameters match on both sides as to type, permissible ranges, meaning and utilization.

System Testing:

Involves in-house testing of the entire system before delivery to the user. System testing examines every component of an application to make sure that they work as a complete and unified whole. Of the three levels of testing, the system level is closet to everyday experiences. We test many things; a used car before we buy it, an on-line cable network service before we subscribe, and so on. A common pattern in these familiar forms is that we evaluate a product in terms of our expectations; not with respect to a specification or a standard. Consequently, goal is not to find faults, but to demonstrate performance. Because of this we tend to approach system testing from a functional standpoint rather than from a structural one. Since it is so intuitively familiar, system testing in practice tends to be less formal than it might be, and is compounded by the reduced testing interval that usually remains before a delivery deadline.

During system testing, we should evaluate a number of attributes of the software that are vital to the user and are listed below. These represent the operational correctness of the product and may be part of the software specifications.

- Usable - Is the product convenient, clear, and predictable?
- Secure - Is access to sensitive data restricted to those with authorization?
- Compatible - Will the product work correctly in conjunction with existing data, software, and procedures?
- Dependable - Do adequate safeguards against failure and methods for recovery exist in the product?
- Documented - Are manuals complete, correct, and understandable?

6.2 Unit Testing

Unit testing is a level of testing that focuses on testing individual units or components of the software in isolation. In unit testing, every module is tested individually. Registration, attendance, study material upload, internal mark, resources, assignments, quiz undergo unit testing.

The purpose of this unit testing is to validate the correctness and functionality of key individual units in the Learning Management System (LMS) application. This test plan covers the unit testing of critical components of the LMS, including user authentication, class & subject creation, content delivery, and quiz & assignment functionalities. Unit testing will follow a bottom-up approach, where individual units will be tested first, and integration testing will follow in subsequent stages. The following key components will be tested as part of the unit testing:

- User authentication and login process.
- class & subject creation and management.
- Content delivery and accessibility.
- quiz and assignment functionalities.

Develop focused test cases for each unit/component to ensure maximum test coverage. nit testing will be considered complete when all key components of the LMS have been tested.

Test case ID	Test Objective	Precondition	Steps/Cases	Test Data	Expected result	Actual result	Postcondition
TC_M1_01	Successful student registration	The site is launched on a compatible browser. A valid email id should be available	The student enters valid details. Clicks register button.	Name Email Batch Year of admission Email password	The student registered successfully	As expected	The details are stored in the LMS correctly.
TC_M1_02	Successful teacher registration	The site is launched on a compatible browser. A valid email id should be available	The HOD enters valid details. Clicks register button.	Name Email Password	The teacher registered successfully	As expected	The details are stored in the LMS correctly.
TC_M1_03	Successful user login	The site is launched on a compatible browser. A valid username and password should be in place to log in to the system.	The user enters a username and password. Clicks login button	Username Password	The user login successful	As expected	The user is redirected to the respective dashboard
TC_M2_01	Successful class creation	The site is launched on a compatible browser. The teacher should be logged into the system	The teacher clicks on Create Class button. The teacher enters details such as class name, batch, and year of admission. Clicks create button	Class name Batch Year of admission	Class created successfully	As expected	The class details are stored in the LMS. Email notification is sent to each student in the given batch and year of admission.
TC_M2_02	Successful subject creation	The site is launched on a compatible browser.	The teacher clicks on add subject button.	Subject name	Subject created successfully	As expected	Subject details like subject name,

		A class is to be created to create a subject in the class	The subject name is entered by the teacher The teacher clicks the add button.				and class in which subject created are stored in the LMS. Each student gets an email notification regarding the subject.
TC_M2_03	Upload study materials	The site is launched on a compatible browser. A teacher is to be logged into the system. A subject is to be created to upload study materials.	The teacher clicks the upload notes button. The teacher uploads the file in pdf format. Clicks upload button	Study material	Notes upload successfully	As expected	The notes are stored in the LMS. Notes are made accessible to students. An email notification is sent to each student regarding the note upload.
TC_M2_04	Schedule assignments	The site is launched on a compatible browser. A teacher is to be logged into the system. A subject is to be created to schedule an assignment.	The teacher clicks on the schedule assignment button. Enter details such as title, due date, upload a file if any, and description. Clicks assign button	Title Due date File (if any) Description	Assignment scheduled successfully	As expected	The assignment details are stored correctly. Assignments are made available to students. Email notification is sent

							to each student.
TC_M2_05	Successful entry of internal marks	The site is launched on a compatible browser. A teacher is to be logged into the system. A subject is to be created to enter internal marks	The teacher clicks enter internal marks button. The marks for each student are entered in the interface. The teacher clicks the save button.	Marks	Internal mark entry successful	As expected	Internal marks are stored in the system. Made available to students for viewing purposes.
TC_M2_06	Successful quiz creation	The site is launched on a compatible browser. A teacher is to be logged into the system. A class is to be created to create a quiz	The teacher clicks create quiz button. The teacher enters quiz details. The teacher clicks Create button.	Title Time allotted Description Question Choice 1 Choice 2 Choice 3 Choice 4 Correct answer	Quiz created successfully	As expected	Quiz details are stored in the system. Quiz is made available for students to participate in. Email notification is sent to each student.
TC_M2_07	Successful attendance marking	The site is launched on a compatible browser. A teacher is to be logged into the system. A class is to be created to mark attendance.	The teacher clicks the mark attendance button. The attendance marking interface with all the students listed in the class is shown. The teacher mark only the absentees and	Absent students	Attendance marking successful	As expected	Attendance details of students are stored in LMS. It can be viewed by students.

		Students must be registered to the system	others are considered present. The teacher clicks mark button.				

Table 6.2. 1 Unit testing

6.3 Integration Testing

Integration testing for an LMS involves testing the interactions and integration between different components, modules, and external systems.

- Identify Integration Points: Identify the different integration points within the LMS, such as user management, course management, content delivery, etc.
- Define Integration Test Scenarios: Define integration test scenarios that simulate the interaction between different components. Each scenario should cover a specific integration point and test the flow of data and functionality between the integrated components.
- Prepare Test Data: Prepare test data that represent realistic scenarios and cover different edge cases for the integration points being tested. This may include user profiles, course information, and assessment results.
- Execute Integration Tests: Execute the integration tests according to the defined test scenarios. This involves interacting with the LMS components and verifying the expected behavior and data flow between them. Record the test results and any observed issues or discrepancies.
- Validate Data Consistency: Validate the consistency and accuracy of data across integrated components. Verify that data is properly synchronized and updated between relevant modules.
- Debug and Resolve Integration Issues: If any integration issues or failures are identified, resolve them. This may involve identifying and fixing communication errors, data mismatches, or compatibility issues.
- Retest and Regression Testing: After fixing integration issues, rerun the integration tests to validate the successful resolution of the issues. Additionally, perform regression testing to ensure that the fixes do not introduce new issues or break existing functionality.

Test case ID	Test Objective	Precondition	Steps/Cases	Test Data	Expected result	Actual data	Postcondition
TC_M1_01	To verify that the functionality to record internal marks in the LMS works correctly, ensuring that marks are accurately recorded for students	The site is launched on a compatible browser. The teacher must be logged into the system. Students must be registered to the system.	Log in using valid credentials with the required permissions for recording internal marks. Navigate to the section of the LMS where internal marks recording functionality is available (select class ->select subjects -> click add internal marks button). Enter the internal marks for each student. Save or submit the recorded internal marks. Verify that the recorded internal marks are correctly associated with each student.	marks	The internal marks recording functionality should be accessible for mark recording without any issues. After recording the internal marks, the marks should be accurately associated with each student. The recorded internal marks should be saved and visible for future reference.	As expected	The internal marks are successfully recorded and stored in the LMS database. Students and instructors can view the recorded internal marks
TC_M2_01	To verify that the functionality to mark attendance in the LMS works correctly, ensuring that students' attendance is accurately recorded	The site is launched on a compatible browser. The user has valid login credentials with the necessary permissions.	Log in using valid credentials with the required permissions to mark attendance. Navigate to the section of the LMS where attendance marking functionality is available(attendance section)	Batch Year of admission Date Attendance status.	The attendance marking functionality should be accessible, and the user should be able to select the class and session for attendance marking without any issues. After marking the attendance, the	As expected	The attendance data is successfully recorded and stored in the LMS database. Students and instructor

			Select the batch, year of admission, and date for marking attendance. Find the list of enrolled students for the selected course. Mark the attendance for each student according to the session's attendance status. Save or submit the marked attendance. Verify that the attendance is accurately recorded for each student and associated with the correct course and session.		attendance status for each student in the selected class and session should be accurately recorded. The marked attendance should be saved and visible for future reference.		s can view the marked attendance for the respective class and session.
TC_M3_01	To verify that the content upload functionality in the LMS works correctly	The site is launched on a compatible browser. A valid username and password should be in place to log in to the system.	Log in using valid credentials with the required permissions for content upload. Navigate to the section of the LMS where content upload functionality is available(my classes -> subject -> note upload). Click on the "Upload" button to initiate the content upload process.	File to upload	The content upload functionality should be accessible, and the user should be able to choose a file for upload without any issues. After the upload process is completed, the uploaded content should be available in the designated	As expected	The content is successfully uploaded and available within the LMS for authorized users. The LMS database is updated to reflect

			Choose a file from the local system to upload (document file). Wait for the upload process to complete. Verify that the uploaded content is visible and can be accessed by users with the appropriate permissions. Attempt to access the uploaded content with a user who does not have the necessary permissions.		location within the LMS. Users with the appropriate permissions should be able to access and view the uploaded content without any difficulties. Users without content upload privileges should be denied access to the content and redirected to an error page or shown an appropriate error message.		the uploaded content
TC_M4_01	To verify that the functionality to schedule assignments in the LMS works correctly, ensuring that assignments are scheduled with the correct details	The site is launched on a compatible browser. A valid username and password should be in place to log in to the system.	Log in using valid credentials with the required permissions to schedule assignments. Navigate to the section of the LMS where assignment scheduling functionality is available (select class -> select subject -> assignments) Click on the "Schedule Assignment" button. Fill in the details for the new assignment, such as assignment name, description, due date, and any other relevant information.	Title Due date File(if any) description	The assignment scheduling functionality should be accessible, and the user should be able to select the class and fill in the assignment details without any issues. After scheduling the assignment, the assignment details (name, description,file,due date) should be accurately associated with the selected course. The scheduled assignment should be saved and visible in the	As expected	The assignment is successfully scheduled and stored in the LMS database. Students and instructors can view the scheduled assignment

			Save or submit the assignment scheduling form. Verify that the assignment is correctly scheduled for the selected course and associated with the specified due date.		assignments section of the selected course.		
TC_M5_01	To verify that the functionality to create quizzes in the LMS works correctly, ensuring that quizzes are created with the correct details	The site is launched on a compatible browser. A valid username and password should be in place to log in to the system. A class should be created to add quiz.	Log in using valid credentials with the required permissions to create quizzes. Navigate to the section of the LMS where quiz creation functionality is available (select class -> click quiz section -> click create quiz button). Click on the "Create Quiz" button. Fill in the details for the new quiz, such as quiz title, description, time limit, questions, options, and any other relevant information. Save or submit the quiz creation form. Verify that the quiz is correctly created with the specified settings and questions.	Title Time allotted Description Questions Choice 1 Choice 2 Choice 3 Choice 4 Correct answer	The quiz creation functionality should be accessible, and the user should be able to select the class and fill in the quiz details without any issues. After creating the quiz, the quiz title, description, time limit, questions, and options should be accurately associated with the selected course. The created quiz should be saved and visible in the quizzes section of the selected course.	As expected	The quiz is successfully created and stored in the LMS database. Students and instructors can view the created quiz.

Table 6.2. 2 Integration testing

6.4 System Testing

Test case ID	Test Objective	Precondition	Steps/Cases	Test Data	Expected result	Actual result	Postcondition
TC_U1_01	To evaluate the user interface and usability aspects of the Learning Management System (LMS) to ensure it provides a user-friendly and intuitive experience for all users.	The site is launched on a compatible browser. The teacher must be logged into the system.	Log in using valid credentials with the required permissions. Explore the application.	NA	The login process is straightforward, and users can easily find the login or sign-in button. The main dashboard or landing page provides clear navigation and a visually appealing layout. The user interface is responsive. The primary navigation elements (sidebar) are easily accessible and self-explanatory. class listings are presented in an organized and easily scannable manner. Users can quickly access their class and view subjects. subjects are organized logically, with clear headings.	As expected	Users have successfully explored the LMS, and it has met the usability expectations, ensuring an intuitive and user-friendly experience for all users.

					Course materials, such as documents, assignments, and quizzes, are easy to access and download. Links, buttons, and interactive elements provide clear visual cues and are easily distinguishable from regular text. The LMS's typography and color scheme promote readability and a pleasant user experience. Interactive elements (e.g., buttons, dropdowns) respond promptly to user interactions without significant delays. Form inputs and validation messages are clear and helpful in guiding users to enter correct information. Error messages are informative and help users understand how		

					to correct issues effectively. The user interface is consistent throughout different pages and sections of the LMS. The system provides appropriate feedback or loading indicators during background processes. The LMS functions correctly on different web browsers and devices without significant compatibility issues.		
TC_S1_01	To verify that the Learning Management System (LMS) is protected against SQL injection attacks on forms and inputs, ensuring the security and integrity of the system's database.	The site is launched on a compatible browser. The system has a form or input field that interacts with the database, such as login credentials or user registration.	Identify a vulnerable input field in the LMS, such as the login form or any other form. Craft an SQL injection attack string that can potentially exploit the vulnerability.	the test data will include the SQL injection attack string mentioned.	When the SQL injection attack string is input into the identified vulnerable form or input field, the system should detect the attack attempt and prevent any unauthorized access to the database. The system should handle the attack gracefully, without displaying any	As expected	The LMS successfully mitigates the SQL injection attack and maintains the confidentiality and integrity of its database.

			Example SQL Injection Attack String: ' OR 1=1; -- This string is designed to return all records from the database, bypassing any username/password checks.		sensitive error messages that might provide clues to potential attackers.		
TC_C1_01	To verify that the Learning Management System (LMS) functions correctly and displays consistently across various web browsers.	The LMS is deployed and accessible in a testing environment, and the following web browsers are available for testing: Google Chrome Mozilla Firefox Microsoft Edge	Launch Google Chrome and navigate to the LMS login page. Enter valid login credentials and log in to the LMS. Perform basic navigation through different sections of the LMS, such as classes, user dashboard, and attendance marking. Verify the appearance, layout, and functionality	Username password	The LMS login page loads without any errors, and the login process is successful in all browsers. All essential features of the LMS are accessible and perform smoothly. The LMS user interface is consistent across pages and aligns with the expected design. Buttons, links, and other interactive elements respond correctly to user interactions. Course content, assignments,	As expected	The LMS has been tested for compatibility with Google Chrome. Repeat the same steps for Mozilla Firefox, Microsoft Edge, to verify compatibility across multiple browsers. The LMS should perform equally well and consistently on all tested browsers.

			of critical elements, such as menus, buttons, forms, and content rendering. Attempt to access various courses, quizzes, and learning materials to ensure they load correctly.		attendance, internal marks, quizzes, is displayed accurately and without distortion. Navigation between different sections of the LMS is smooth and intuitive. Forms for course creation, quiz setup, and user management function correctly and accept user inputs as expected. LMS features that rely on browser-specific functionalities (e.g., file uploads, notifications) work as intended in all browsers.		
TC_D1_01	To verify that data is accurately stored and retrieved from the database of the Learning Management System (LMS) during various interactions and operations	The LMS is running. The database is properly configured and connected to the LMS. The LMS has relevant data (e.g., users, courses, quizzes) stored in the database.	Perform user registration and verify that the user's details (e.g., username, email) are accurately stored in the database. Create a new class and verify that all class-	Use test data that includes a range of valid inputs and edge cases to cover different scenarios in the LMS.	All data related to user registration, class creation, content management, and assessment creation is accurately stored in the database. The retrieved data matches the original data input during various operations.	As expected	The database of the LMS accurately stores and retrieves data during various interactions and operations, ensuring a reliable and consistent learning experience for users.

	within the system. To verify that data is accurately stored and retrieved from the database of the Learning Management System (LMS) during various interactions and operations within the system.		related data (e.g., class name, description, batch, year of admission) are accurately stored in the database. Add subjects to the class and confirm that the content data is correctly saved in the database. Create a quiz and ensure that the quiz details, questions, options, and correct answers are accurately stored in the database. Enroll multiple users and check that enrollment data for each user is correctly recorded in the database. Have users complete assessments		The data in the database is consistent, without any discrepancies or data corruption. The LMS functions as expected, with all relevant data being retrieved and displayed correctly to users.		

			(e.g., quizzes, exams) and verify that the assessment scores are accurately saved in the database. Create and send messages between users and confirm that the messaging data is correctly stored in the database. mark attendance and check database for errors. Perform various updates and edits (e.g., changing user information, modifying course content) and verify that the changes are reflected accurately in the database.				

			Delete user accounts, courses, or other data and ensure that the corresponding data is successfully removed from the database.				

Table 6.2. 3 System testing

CHAPTER 7 SYSTEM MAINTENANCE

7.1 Introduction

This software can be modified as need occurs. Maintenance includes all the activities after installation of the software that is performed to keep the system operational. The process of maintaining involves.

- Understanding the existing software
- Understand the effect of change
- Test for satisfaction

Maintenance can be done to this project by simply adding the new requirements that are the form of database the system can be modified. The maintenance process also helps to remove an error that resides in the system even after testing process.

7.2 Maintenance

The purpose of this maintenance plan is to outline the procedures and responsibilities for the ongoing maintenance and support of the Learning Management System (LMS).

The following maintenance activities will be performed on a regular basis:

- Software Updates: Regular software updates, patches, and bug fixes provided. These updates ensure that the system remains secure, stable, and compatible with the latest technologies.
- Server Monitoring: Continuously monitor server resources, including CPU usage, memory utilization, and disk space. Proactively identify and address any performance issues to maintain optimal system performance.
- Database Maintenance: Perform routine database maintenance tasks, including backups, indexing, and optimizing database performance. Regularly check for data integrity and clean up unnecessary data to ensure efficient operations.
- Security Audits: Conduct regular security audits to identify potential vulnerabilities or threats to the LMS. Implement necessary security measures, such as regular password updates, secure authentication methods, and encryption protocols.
- Detailed System Documentation: A comprehensive documentation of the system architecture, components, algorithms, and workflows is created. This document serves as a reference for understanding the system's structure and operation.

- User Manual: A user manual is prepared to guide users on how to interact with the system, including instructions on uploading images, viewing details, and understanding the system's features and functionalities.
- Training and skill development: Provides ongoing training and skill development opportunities for LMS administrators and support staff to ensure they can effectively manage and troubleshoot the system.

CHAPTER 8 SYSTEM SECURITY MEASURES

8.1 Introduction

The level of security incorporated into project management software determines the safety of the project. This covers the protection of data integrity, confidentiality, infrastructure security, and network stability. Insufficient security can make the project vulnerable to hackers and scammers. Conversely, excessive security may limit the team's access to essential information. According to the Professional Services Survey Report, 60% of leaders acknowledged that data security apprehensions hinder their teams' flexibility and hinder the ability to foster better user experience. Ensuring the security of a Learning Management System (LMS) is of paramount importance, as it deals with sensitive user data, course content, and the overall learning experience. By implementing robust security measures, an LMS can protect against unauthorized access, data breaches, and other cyber threats, thus building trust among users and safeguarding their information. A strong user authentication mechanism is crucial to verify the identity of users before granting access to the LMS. This can be achieved through secure login credentials. Additionally, role-based access control ensures that each user is granted appropriate permissions based on their role within the LMS. Implementing strong access controls and session management mechanisms prevents unauthorized access to user accounts and helps protect against session hijacking and replay attacks. Educating LMS users, administrators, and support staff about security best practices is essential in maintaining a secure environment. Regular security awareness training helps users recognize and respond to potential security threats effectively. By adopting these security measures and regularly updating them to stay ahead of evolving threats, an LMS can establish a robust security posture and create a safe learning environment for users, instilling confidence and trust in the platform.

8.2 Operating System Level Security

Operating system level security for a web application LMS (Learning Management System) involves implementing security measures at the operating system level to protect the server and the underlying infrastructure hosting the web application. These security measures aim to safeguard against unauthorized access, data breaches, and other potential threats. Here are some important operating system level security practices for a web application LMS:

- Regular OS Updates: Keep the operating system up-to-date with the latest security patches and updates. This ensures that known vulnerabilities are addressed promptly.

- Firewalls and Network Segmentation: Configure firewalls to control incoming and outgoing network traffic. Employ network segmentation to isolate the LMS server from other critical systems, minimizing the attack surface.
- Limited Access and User Privileges: Restrict access to the LMS server by limiting the number of users with administrative privileges. Utilize least privilege principles to grant users the minimum permissions required to perform their tasks.
- Secure Authentication: Implement strong user authentication mechanisms, such secure password policies, to prevent unauthorized access to the server.
- Intrusion Detection and Prevention: Deploy intrusion detection and prevention systems (IDS/IPS) to monitor and block suspicious activities or potential attacks on the server.
- Disable Unnecessary Services: Turn off or disable any unnecessary services, applications, or ports that are not required for the LMS operation to reduce the attack surface.
- Secure File Permissions: Set appropriate file and directory permissions to prevent unauthorized access or modification of critical files.
- Secure Remote Access: If remote access to the server is required, use secure methods such as VPN (Virtual Private Network) or secure remote desktop protocols to protect against unauthorized access.
- Backup and Disaster Recovery: Regularly back up the LMS data and configurations, and test the backup restoration process to ensure data recovery in case of system failures or data loss.
- Vulnerability Scanning: Conduct regular vulnerability scans on the server to identify potential weaknesses and address them proactively.
- Antivirus and Malware Protection: Install and keep up-to-date antivirus and anti-malware software to detect and prevent malware infections.
- Security Monitoring and Incident Response: Implement security monitoring tools to detect unusual activities and respond to security incidents promptly.
- Access Control Lists (ACLs): Utilize Access Control Lists to control access to specific files, directories, or resources based on user roles and permissions.

By adhering to these operating system level security practices, an LMS web application can enhance its security posture, protect user data, and maintain a robust and reliable learning platform for users.

8.3 Database Level Security

Database security refers to the range of tools, controls, and measures designed to establish and preserve database confidentiality, integrity, and availability. Database security must address and protect the following:

- The data in the database
- The database management system (DBMS)
- Any associated applications
- The physical database server and/or the virtual database server and the underlying hardware
- The computing and/or network infrastructure used to access the database

Database security is a complex and challenging endeavor that involves all aspects of information security technologies and practices. It's also naturally at odds with database usability. The more accessible and usable the database, the more vulnerable it is to security threats; the more invulnerable the database is to threats, the more difficult it is to access and use.

By definition, a data breach is a failure to maintain the confidentiality of data in a database. How much harm a data breach inflicts on your enterprise depends on a number of consequences or factors:

- Compromised intellectual property: Your intellectual property—trade secrets, inventions, proprietary practices—may be critical to your ability to maintain a competitive advantage in your market. If that intellectual property is stolen or exposed, your competitive advantage may be difficult or impossible to maintain or recover.
- Damage to brand reputation: Customers or partners may be unwilling to buy your products or services (or do business with your company) if they don't feel they can trust you to protect your data or theirs.
- Business continuity (or lack thereof): Some business cannot continue to operate until a breach is resolved.
- Fines or penalties for non-compliance: The financial impact for failing to comply with global regulations such as the Sarbannes-Oxley Act (SAO) or Payment Card Industry Data Security Standard (PCI DSS), industry-specific data privacy regulations such as HIPAA, or regional data privacy regulations, such as Europe's General Data Protection Regulation (GDPR) can be devastating, with fines in the worst cases exceeding several million dollars per violation.

- Costs of repairing breaches and notifying users: In addition to the cost of communicating a breach to user, a breached organization must pay for forensic and investigative activities, crisis management, triage, repair of the affected systems, and more.

Best practices to ensure database security:

Databases are nearly always network-accessible, any security threat to any component within or portion of the network infrastructure is also a threat to the database, and any attack impacting a user's device or workstation can threaten the database. Thus, database security must extend far beyond the confines of the database alone. When evaluating database security, consider each of the following areas:

- Physical security: The database must be located within a secure, climate-controlled environment.
- Administrative and network access controls: The practical minimum number of users should have access to the database, and their permissions should be restricted to the minimum levels necessary for them to do their jobs. Likewise, network access should be limited to the minimum level of permissions necessary.
- End user account/device security: Always be aware of who is accessing the database and when and how the data is being used. Data monitoring solutions can alert you if data activities are unusual or appear risky. All user devices connecting to the network housing the database should be physically secure (in the hands of the right user only) and subject to security controls at all times.
- Database software security: Always use the latest version of your database management software, and apply all patches as soon as they are issued.
- Application/web server security: Any application or web server that interacts with the database can be a channel for attack and should be subject to ongoing security testing and best practice management.
- Backup security: All backups, copies, or images of the database must be subject to the same (or equally stringent) security controls as the database itself.
- Auditing: Record all logins to the database server and operating system, and log all operations performed on sensitive data as well. Database security standard audits should be performed regularly.
- Authentication is the process of proving the user is who he or she claims to be by entering the correct user ID and password. Some security solutions allow administrators

to centrally manage the identities and permissions of database users in one central location. This includes the minimization of password storage and enables centralized password rotation policies.

- Authorization allows each user to access certain data objects and perform certain database operations like read but not modify data, modify but not delete data, or delete data.
- Access control is managed by the system administrator who assigns permissions to a user within a database. Permissions are ideally managed by adding user accounts to database roles and assigning database-level permissions to those roles. For example, row-level security (RLS) allows database administrators to restrict read and write access to rows of data based on a user's identity, role memberships, or query execution context. RLS centralizes the access logic within the database itself, which simplifies the application code and reduces the risk of accidental data disclosure.

8.4 System Level Security

System-level security is part of a multi-layered security approach in which information security (IS) is implemented on an IT infrastructure’s different components, layers or levels. System-level security is typically implemented on end-user computer and server nodes. It ensures that system access is granted only to legitimate and trusted individuals and applications. The key objective behind system-level security is to keep system secure, regardless of security policies and processes at other levels. If other layers or levels are breached, the system must have the ability to protect itself. Secure authentication and user access, role-based access control, encryption, input validation and output sanitization, session management, security awareness training, etc. together constitute a high level of system security. By implementing these system level security practices, an LMS can establish a robust security posture, safeguard user data, and provide a secure learning environment for users. Continuous monitoring, updates, and improvements are essential to maintain the effectiveness of the system security measures over time.

CHAPTER 9 SYSTEM PLANNING AND SCHEDULING

9.1 Introduction

Project planning and scheduling are fundamental aspects of effective project management. They involve defining the project's objectives, outlining the necessary tasks, estimating resource requirements, and establishing a timeline to complete the project successfully. A well-structured plan and schedule help ensure that the project is executed efficiently, within budget, and on time. Project planning is the process of creating a detailed roadmap that outlines the project's goals, scope, tasks, deliverables, and resource requirements. The primary purpose of project planning is to establish a clear direction and guide the project team throughout its lifecycle. It involves collaboration among stakeholders, project managers, and team members to define the project's purpose, identify potential risks, and establish a strategy to overcome challenges. Project scheduling involves creating a time-bound plan that outlines when each task or activity will be executed to meet project objectives. A well-structured schedule helps with resource management, team coordination, and meeting deadlines.

Creating a project plan for a Learning Management System (LMS) involves outlining the key activities, resources, and timelines required to develop, implement, and maintain the system.

Project Initiation:

- Define the project objectives and scope: Determine the primary goals and features of the LMS, such as course management, user registration, assessments, reporting, etc.
- Identify stakeholders: Identify key stakeholders, including educators, administrators, IT staff, and learners, and understand their requirements and expectations.

Requirements Gathering and Analysis:

- Conduct a needs analysis: Determine the target audience's specific learning needs and preferences.
- Gather functional and technical requirements: Document all the necessary features, integrations, and technical specifications needed for the LMS.

Design and Development:

- User Interface (UI) and User Experience (UX) design: Create wireframes and prototypes for the LMS interface to ensure a user-friendly and intuitive experience.
- Database Creation: Designing and developing database and table structure. Also includes finding relationships and association, defining constraints, etc.

Implementation:

- Coding: The designs are coded to its form.
- Testing and Quality Assurance: Conduct thorough testing to identify and resolve any bugs or issues.
- Pilot testing: Launch a limited version of the LMS to a small group of users for testing and feedback.
- User training: Provide training sessions to educators, administrators, and learners to familiarize them with the LMS's features and functionalities.

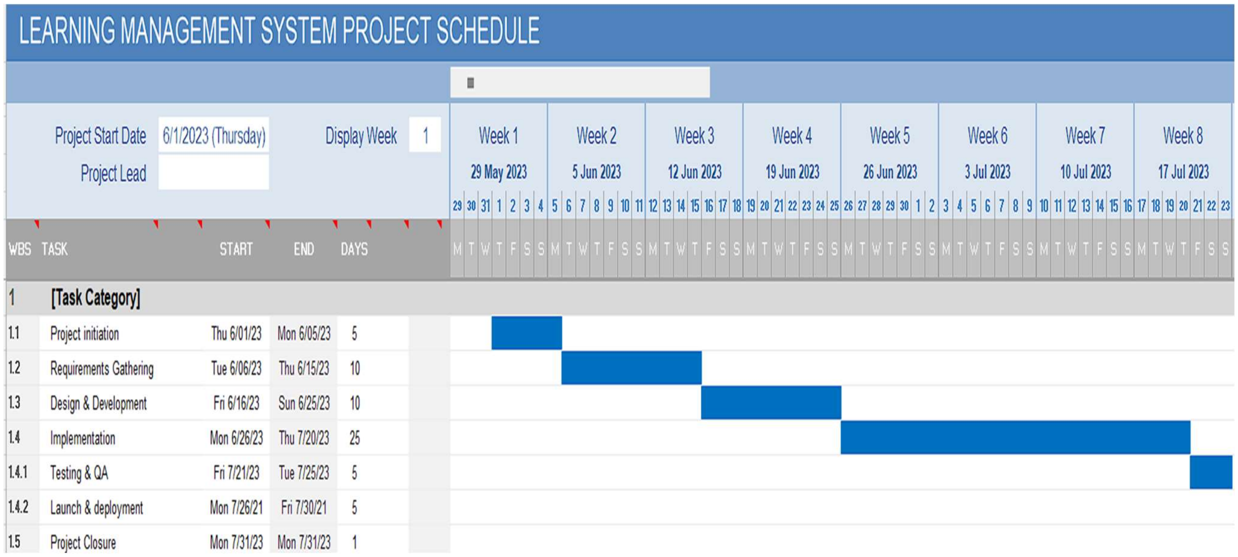
Launch and Deployment:

- Full-scale deployment: Launch the LMS to all intended users, ensuring all technical aspects are in place for a smooth rollout.

Project Closure:

- Document lessons learned: Summarize key takeaways and insights from the project for future reference.
- Handover and maintenance: Ensure a smooth handover to the team responsible for the LMS's ongoing maintenance and management

9.2 GANNT Chart



LEARNING MANAGEMENT SYSTEM PROJECT SCHEDULE

Project Start Date 6/1/2023 (Thursday)

Display Week 1

Project Lead

WBS	TASK	START	END	DAYS
1	[Task Category]			
1.1	Project initiation	Thu 6/01/23	Mon 6/05/23	5
1.2	Requirements Gathering	Tue 6/06/23	Thu 6/15/23	10
1.3	Design & Development	Fri 6/16/23	Sun 6/25/23	10
1.4	Implementation	Mon 6/26/23	Thu 7/20/23	25
1.4.1	Testing & QA	Fri 7/21/23	Tue 7/25/23	5
1.4.2	Launch & deployment	Mon 7/26/21	Fri 7/30/21	5
1.5	Project Closure	Mon 7/31/23	Mon 7/31/23	1

Figure 9.2. 1 GANNT Chart

CHAPTER 10 SYSTEM COST ESTIMATION

10.1 Introduction

System cost estimation is a crucial process in project management that involves predicting the financial resources required to develop, implement, and maintain a system or project. Cost estimation provides valuable insights to stakeholders, project managers, and decision-makers, helping them make informed decisions about resource allocation, budget planning, and project feasibility.

Introduction to System Cost Estimation:

- Purpose: The primary purpose of system cost estimation is to provide a realistic assessment of the financial investment needed to complete a project successfully. It helps in determining whether the project aligns with the organization's budget and financial constraints.
- Project Scope: Cost estimation is closely tied to the project scope. Understanding the project's objectives, features, functionalities, and requirements is essential to accurately estimate the associated costs. A well-defined project scope ensures that all relevant costs are considered.

Factors Influencing Cost: Several factors influence the overall cost of a system development project. These may include but are not limited to:

- ➢ Labor Costs: The salaries and wages of the project team members and other personnel involved in the project.
- ➢ Hardware and Software: The cost of acquiring necessary hardware components and software licenses.
- ➢ Training: Expenses related to training team members and end-users to use the system effectively.
- ➢ Infrastructure: Costs associated with setting up and maintaining the required infrastructure for the system.
- ➢ Risk Management: Funds allocated for mitigating potential risks and uncertainties.
- ➢ Contingency: An additional reserve for unexpected costs or scope changes.
- ➢ Vendor Costs: If outsourcing certain components or services, the fees charged by external vendors.

Estimation Techniques: Various techniques can be used for cost estimation, depending on the project's complexity and available data. Some common methods include:

- ➢ Analogous Estimating: Using data from similar past projects as a reference to estimate costs.
- ➢ Parametric Estimating: Developing mathematical models based on historical data to calculate costs.
- ➢ Bottom-Up Estimating: Estimating costs for individual tasks or components and aggregating them for the overall project cost.
- ➢ Three-Point Estimation: Incorporating optimistic, pessimistic, and most likely scenarios to calculate expected costs.

- Accuracy and Uncertainty: Cost estimation is subject to uncertainties, especially in complex projects with many variables. Project managers must communicate the level of confidence in the estimates and account for potential fluctuations.
- Continuous Refinement: Cost estimation is an iterative process. As the project progresses and more information becomes available, cost estimates may be refined and updated to reflect the current status accurately.
- Cost-Benefit Analysis: Besides estimating the costs, it is essential to conduct a cost-benefit analysis to weigh the projected benefits against the expenses. This analysis helps in evaluating the project's viability and potential returns.

Accurate cost estimation is crucial for the successful management and execution of projects. It helps in making informed decisions, setting realistic budgets, and ensuring that the project's financial resources are utilized effectively to achieve the desired outcomes.

10.2 LOC Based Estimation

Line of code (LOC)-based cost estimation is a software development cost estimation technique that calculates project costs based on the number of lines of code written or expected to be written for the software project. It is a simplistic method that assumes a correlation between the number of lines of code and the overall development effort and cost.

The basic idea behind LOC-based cost estimation is that the more lines of code a project contains, the more effort and time it would take to develop the software, which would ultimately impact the cost. However, it's important to note that this approach has several limitations and should be used with caution:

Code Quality: LOC-based estimation does not consider the quality or complexity of the code. Two projects with the same number of lines of code may require vastly different levels of effort and resources if one has well-structured, efficient code, while the other has poorly written code.

Technology and Skill Level: Different programming languages and technologies can result in varying code sizes for the same functionality. Moreover, the skill and experience of the development team can significantly impact the LOC count.

Code Reusability: Reusing existing code can reduce the number of lines of code needed for a project, but this may not be accounted for in simple LOC-based estimation.

Requirement Changes: If project requirements change, it can lead to a change in the number of lines of code, and thus, the cost estimation may become inaccurate.

Despite its limitations, LOC-based cost estimation can be used as a quick and rough estimate for small projects with well-defined requirements. For larger and more complex projects, more sophisticated cost estimation techniques like Function Point Analysis (FPA) or Use Case Points (UCP) are often preferred, as they consider various factors beyond just the lines of code to provide more accurate estimates.

To calculate the project cost in INR (Indian Rupees), you need to consider the labor cost in INR per person-month. Labor cost per person-month is 25,000 INR.

Step 1: Calculating Effort

Effort = 400000 LOC / 20000 LOC per person-month Effort = 20 person-months

Step 2: Determine Labor Cost (in INR)

Labor Cost per Person-Month = 25,000 INR

Step 3: Calculate Project Cost (in INR)

Project Cost = 20 person-months * 25,000 INR per person-month

Project Cost = 500000 INR

CHAPTER 11 FUTURE ENHANCEMENT AND SCOPE OF FURTHER DEVELOPMENT

11.1 Introduction

A Learning Management System (LMS) is a powerful tool for delivering and managing educational content, and it plays a crucial role in modern educational environments and corporate training programs. As technology and learning methodologies evolve, there is a continuous need for future enhancements and improvements to keep the LMS up-to-date and aligned with the changing needs of users and organizations.

11.2 Merits of the System

A Learning Management System (LMS) offers numerous merits that make it a valuable tool for educational institutions, businesses, and organizations. Some of the key merits of using an LMS include:

- Centralized Learning Platform: An LMS provides a centralized platform for managing and delivering educational content. It allows educators or trainers to upload, organize, and update learning materials in one place, making it easily accessible to learners.
- Flexibility and Convenience: LMS platforms enable learners to access educational content at their own pace and convenience.
- 24/7 Access to Learning Materials: Learners can access the LMS and its resources 24/7, allowing them to study or review content whenever they choose, which is especially advantageous for self-paced and asynchronous learning.
- Tracking and Reporting: LMS systems provide detailed tracking and reporting capabilities, enabling administrators and instructors to monitor learners' progress, participation, and performance. This data can inform decision-making and identify areas for improvement.
- Assessments and assignments: LMS platforms support assessments and assignments, which streamline the evaluation process for instructors.
- Scalability and Reach: LMS systems can scale to accommodate a large number of learners, making it suitable for organizations of all sizes.
- Consistent Content Delivery: With an LMS, learners receive consistent and standardized content delivery, ensuring that all participants receive the same high-quality learning experience.
- Resource Efficiency: LMS platforms reduce the need for physical resources, such as paper-based materials and classrooms, promoting environmental sustainability.

- Easy Content Updates: LMS platforms allow for quick and easy updates to learning materials, ensuring that the content remains relevant and up-to-date.

Overall, an LMS offers a host of benefits that enhance the learning experience for both educators and learners, making it an essential tool for modern educational and training environments.

In conclusion, the future of an LMS lies in its ability to evolve and adapt to changing user needs and technological advancements. An LMS can remain a cutting-edge and valuable tool for facilitating effective learning experiences by continuously exploring areas for improvement and incorporating new features and functionalities.

11.3 Limitations of the System

While Learning Management Systems (LMS) offer numerous benefits, they also have certain limitations and challenges. It's essential to be aware of these limitations to make informed decisions when implementing an LMS.

- Initial Cost and Implementation Complexity: The initial setup and implementation of an LMS can be costly and time-consuming. This includes software licensing fees and training for users.
- Maintenance and Upkeep: LMS platforms require regular maintenance, updates, and technical support to ensure smooth functioning. This ongoing effort may demand additional resources and expertise.
- User Adoption and Training: Getting all users, including instructors, and learners, accustomed to using the LMS might face resistance. Adequate training and support are crucial to ensure optimal adoption and usage.
- Scalability Concerns: As the number of learners or courses grows, the LMS may face scalability challenges. Ensuring the system's performance and responsiveness under increased loads is essential.
- Data Security and Privacy: LMS platforms store sensitive user data, including personal information and learning progress. Ensuring robust data security and compliance with data protection regulations is critical.
- Technical Requirements: LMS systems might have specific technical requirements, such as compatible browsers or devices, which could limit accessibility for certain users.
- Content Format Limitations: LMS platforms have restrictions on the types of content they can support, limiting the use of certain multimedia formats or interactive elements.

- Lack of Human Interaction: In fully online learning environments, the absence of face-to-face interaction may hinder collaborative learning and social engagement among learners.
- Courseware Conversion: Migrating existing course content to the LMS might require significant effort and content conversion, especially if the original materials are in non-digital formats.
- Limited Offline Access: Some LMS platforms might lack offline access options, restricting learning opportunities for users without continuous internet connectivity.
- Accessibility Challenges: Ensuring accessibility for learners with disabilities, such as visually impaired users, can be challenging, and LMS platforms might not meet accessibility standards.

Understanding these limitations can help organizations plan for potential challenges and find strategies to overcome them, maximizing the benefits of using an LMS while addressing its constraints.

11.4 Future Enhancement of the System

- User Experience (UX) Enhancements: Continuously improving the LMS's user interface and user experience can lead to higher user engagement and satisfaction. This includes implementing responsive design for mobile access, intuitive navigation, personalized dashboards, and user-friendly course catalogs.
- Advanced Reporting and Analytics: Enhancing the reporting and analytics capabilities of the LMS can provide valuable insights into learners' progress, course effectiveness, and overall performance. Implementing advanced analytics, data visualization tools, and customized reporting options can help stakeholders make data-driven decisions.
- Artificial Intelligence (AI) Integration: Leveraging AI in the LMS can enable personalized learning paths, automated content recommendations, intelligent assessments, and real-time feedback, making the learning experience more adaptive and effective.
- Gamification Features: Incorporating gamification elements, such as badges, leaderboards, and rewards, can boost learner motivation and engagement, making the learning process more enjoyable and effective.
- Social Learning and Collaboration: Enhancing social learning features, such as discussion forums, chat, and collaborative projects, can foster a sense of community and enable learners to interact, share knowledge, and collaborate with peers.

- Content Authoring Tools: Providing built-in content authoring tools or seamless integration with third-party authoring tools empowers instructors to create interactive and engaging course content without the need for extensive technical knowledge.
- Multilingual Support: Offering multilingual support expands the LMS's reach to a global audience, allowing learners from different regions to access content in their preferred language.
- Accessibility and Inclusivity: Improving accessibility features, such as support for screen readers, closed captioning, and alternative text, ensures that the LMS is inclusive and usable by learners with diverse abilities.
- Security and Data Privacy: Strengthening security measures and ensuring compliance with data protection regulations is critical for safeguarding learner data and maintaining user trust.
- Continuous Support and Updates: Regularly providing software updates, bug fixes, and technical support is essential to address issues, add new features, and keep the LMS running smoothly.

ANNEXURE

12.1 Organization Profile

IGNOSI is a diverse software development and IoT consulting company that is passionate about creating powerful technologies to bring clients' ideas to life. They offer flexible, user-friendly, and responsive solutions tailored to unique requirements. With skilled developers, creative graphic designers, and a systematic approach, they guide clients through the entire process from consulting and planning to building and deploying. IGNOSI also handles aspects like SEO, social media promotions, and Google Ad word listings for their clients. Their expertise lies in integrating with client teams, enabling better communication and idea sharing to deliver successful IoT products through agile methodology. Their focus on collaboration and pushing clients out of their comfort zones ensures success in their endeavors. Additionally, IGNOSI has contributed to expanding access to financial services for socially oriented credit providers through continuous innovation and technology development.

12.2 Document Glossary, Figures, Tables

- **LMS:** Learning management system, a software application or platform designed to facilitate the management, delivery, and tracking of educational courses and materials.
- **User:** An individual who interacts with the LMS, including teachers, and learners.
- **Learner:** An individual who accesses and participates in classes within the LMS, accessing learning materials, submitting assignments, and tracking their progress.
- **UI:** The visual presentation and interaction design of the LMS, including menus, buttons, forms, and other graphical elements.
- **Database:** A structured collection of data that stores user information, class data, assessment results, and other relevant data for the LMS.
- **Scalability:** The ability of the LMS to handle increased usage, data storage, and concurrent user access as the system grows.
- **Accessibility:** The design and implementation of the LMS to ensure it is usable by individuals with disabilities, adhering to accessibility standards and guidelines.
- **Discussion forums:** An online platform within the LMS that allows learners and instructors to engage in discussions, ask questions, and share information.
- **IGNOSI:** (company name) The company where the project was developed.

- **Stakeholders**: Individuals or groups with an interest or involvement in a project or organization. In the context of the LMS project, stakeholders may include educators, administrators, IT staff, and learners.
- **Wireframes:** Basic visual representations or blueprints that outline the layout and structure of the user interface (UI) for the LMS.
- **Prototypes:** Early-stage models or versions of the LMS that are developed to test and validate design ideas before the final product is built.
- **Database Creation**: The process of designing and building the database, which involves defining its structure, tables, relationships, constraints, etc., to store and manage data efficiently.
- **Integration:** In the context of the LMS, integration refers to the process of connecting the LMS with other systems or platforms to enable data exchange and seamless functionality.
- **Project Initiation**: The initial phase of project management, where project objectives, scope, and stakeholders' involvement are identified and defined.
- **Project Planning:** The process of creating a detailed roadmap or plan that outlines the tasks, deliverables, and timelines required to complete the project successfully.
- **Project Scheduling:** Creating a time-bound plan that specifies when each task or activity will be executed to meet project objectives and deadlines.
- **User Interface (UI):** The visual design and layout of the LMS, including how users interact with and navigate through the system.
- **User Experience (UX):** Refers to the overall experience and satisfaction of users when using the LMS, focusing on usability, accessibility, and user satisfaction.
- Resource Management: The process of allocating and managing the necessary resources, such as human resources, equipment, and finances, for the successful execution of the LMS project.
- **Deliverables:** Tangible outcomes or results that are expected to be produced during the LMS project, such as completed modules, documentation, or reports.
- **Assessments:** Activities or tests conducted within the LMS to evaluate learners' knowledge, understanding, or skills.
- **Reporting:** The functionality within the LMS that provides insights and data on learner progress, course completion, and other performance metrics.

List of figures & tables

12.3 References

- Stack overflow https://stackoverflow.com/ 01/06/2023
- CSS-Tricks https://css-tricks.com/ 29/03/2023
- W3Schools https://www.w3schools.com/ 01/06/2023
- Smashing Magazine https://www.smashingmagazine.com/ 10/06/2023
- CodePen https://www.codepen.io 05/06/2023
- Dev.to https://www.dev.to 10/06/2026
- GitHub https://www.github.com 20/06/2023