**LEARNING MANAGEMENT SYSTEM**

**A Project Report**

**Submitted to Department of Information and Communication Technology**

**In Partial Fulfillment of the requirements for the subject ICT. Ed. 534 Software Project Management of Master Degree in ICT Education.**

**Submitted by:**

**Dadip Bhattarai**

**Symbol No.: 7821092**

**T.U Reg. No.:**

**Pramila Karki**

**Symbol No.: 7821097**

**T.U Reg. No.:**

**Richa Bhandari**

**Symbol No.:** **7821098**

**T.U Reg. No.:**

**Sharmila Rai**

**Symbol No.:** **78210100**

**T.U Reg. No.:**



**TRIBHUVAN UNIVERSITY**

**FACULTY OF EDUCATION**

**SANOTHIMI CAMPUS**

**SANOTHIMI, BHAKTAPUR**



TRIBHUVAN UNIVERSITY

**Faculty of Education**

**Department of Information and Communication Technology**

Sanothimi Campus

Sanothimi, Bhaktapur

# DECLARATION

We hereby declare that this project entitled **Learning Management System** was carried out by me for partial fulfillment of the requirement for the master’s 3rd semester of Information and Communication Technology Education under the guidance and supervision, of Mr. **Govinda Neupane**, lecture Department of Information and Communication Technology (ICT), Sanothimi Campus, Nepal. We declare that the result reported in this thesis is the result of our own work, except where due to reference is made and have not been submitted to any other University or Institute for the award of any degree.

**February, 2025**

…………………….

Dadip Bhattarai

Symbol No: 7821092

……………………

Pramila Karki  
Symbol No: 7821097

…………………….

Sharmila Rai

Symbol No: 78210100

…………………….

Richa Bhandari

Symbol No: 7821098



# RECOMMENDATION

I hereby recommend that this project be prepared under my supervision and guidance by Mr. Dadip Bhattarai, Ms. Pramila Karki, Ms. Richa Bhandari, Ms. Sharmila Rai project entitled “Learning Management System” for the partial fulfillment of the requirements for the degree of master’s in Information and Communication Technology Education be processed for the evaluation. I supervised and analyzed this project, which is too good in its quality, scope, and carrier useful project in the future

Date: .........................

……………………….

**Mr. Govinda Neupane**  
Project Supervisor  
Sanothimi Campus,

Sanothimi, Bhaktapur



# 

# LETTER OF APPROVAL

This project entitled “Learning Management System” submitted by the power pack group, we certify that we have read this project work report, and, in our opinion, it is satisfactory in the scope and quality as a final semester project report in partial fulfillment of the requirement for a master’s degree of Education in Information and Communication Technology has been approved.

**Evaluation Committee Signature**

**Mr. Nawaraj Ghimire**Assistant Professor  
Sanothimi Campus  
Sanothimi, Bhaktapur

…………………………  
**Head of ICTE Department**

**Mr. Rom Kant Pandey**

Assistant Professor  
Tribhuvan University  
Kirtipur, Kathmandu

…………………………  
**External Supervisor**

**Mr. Govinda Neupane**Project Supervisor  
Sanothimi Campus  
Sanothimi, Bhaktapur

…………………………  
**Internal Supervisor**

# ACKNOWLEDGEMENT

We express our sincere gratitude to everyone who played a role in the successful completion of our project titled **"Learning Management System"** as part of our academic curriculum.

Our deepest appreciation goes to our internal supervisor, **Mr. Govinda Neupane**, whose expert guidance, encouragement, and insightful feedback were instrumental in shaping this project. His continuous support ensured that we stayed on the right path and delivered our best work.

We extend our heartfelt thanks to the faculty members and staff of our department for their unwavering support, motivation, and valuable knowledge-sharing. Their commitment to fostering academic excellence has been truly inspiring.

A special acknowledgment goes to **Dr. Madhav Kumar Shrestha, Campus Chief of Sanothimi Campus, Bhaktapur, Nepal**, for providing essential resources and institutional backing that facilitated the successful execution of this project.

We also wish to extend our appreciation to our friends and peers for their invaluable feedback, support, and encouragement throughout this journey. A special note of thanks to **Mr. Tilak K.C.** for his guidance and contributions, which played a crucial role in our progress.

Once again, we appreciate everyone who contributed in any way to the successful completion of this project. Your support and guidance have been invaluable to us.

**Thank you.**

|  |  |  |  |
| --- | --- | --- | --- |
| …………………  Dadip Bhattarai  Roll No.: 7821092  Reg. No.:  Sanothimi Campus | …………………  Pramila Karki  Roll No.: 7821097  Reg. No.:  Sanothimi Campus | …………………...  Richa Bhandari  Roll No.: 7821098  Reg. No.:  Sanothimi Campus | …………………  Sharmila Rai  Roll No.: 78210100  Reg. No.:  Sanothimi Campus |

# Abstract

The **Learning Management System (LMS)** is a web-based platform designed to enhance the teaching and learning experience by providing a centralized system for course management, content delivery, and student engagement. This system aims to improve accessibility, efficiency, and interactivity in educational institutions and corporate training environments.

The LMS enables instructors to create, manage, and distribute course materials while offering students a user-friendly interface to access lessons, assignments, and assessments. Key features include course enrollment, progress tracking, discussion forums, automated grading, and real-time feedback. Additionally, the system integrates with modern web technologies and cloud computing to ensure scalability, security, and seamless accessibility from any location.

By automating administrative tasks and enhancing communication between educators and learners, the LMS minimizes manual effort, reduces paperwork, and promotes data-driven decision-making through insightful analytics and reporting tools. This report outlines the development process, system requirements, and phased implementation strategies. The LMS is expected to improve learning outcomes, foster collaboration, and contribute to the overall efficiency and effectiveness of educational delivery.

**Table of Content**

[DECLARATION ii](#_Toc190070073)

[RECOMMENDATION iii](#_Toc190070074)

[LETTER OF APPROVAL iv](#_Toc190070075)

[ACKNOWLEDGEMENT v](#_Toc190070076)

[Abstract vi](#_Toc190070077)

[List of figures x](#_Toc190070078)

[List of tables xi](#_Toc190070079)

[Chapter-1 Introduction 1](#_Toc190070080)

[1.1 Background 1](#_Toc190070081)

[1.2 Purpose 1](#_Toc190070082)

[1.3 Scope 2](#_Toc190070083)

[Chapter-2 Literature Review 4](#_Toc190070084)

[2.1 Introduction 4](#_Toc190070085)

[2.2 Problem Statement 4](#_Toc190070086)

[2.3 Technological Evolution 5](#_Toc190070087)

[2.4 Existing Solutions: 5](#_Toc190070088)

[2.5 Features and Functionalities 5](#_Toc190070089)

[2.6 Impact on Learning Outcomes 6](#_Toc190070090)

[2.7 Challenges and Limitations 6](#_Toc190070091)

[2.8 Gaps in Current Research 6](#_Toc190070092)

[2.9 Future Directions 7](#_Toc190070093)

[2.10 Conclusion 7](#_Toc190070094)

[Chapter-3 Methodology 8](#_Toc190070095)

[3.1 System Architecture and Design 8](#_Toc190070096)

[3.2 Development Process 8](#_Toc190070097)

[3.3 Proposed System 11](#_Toc190070098)

[3.4 Feasibility Analysis 12](#_Toc190070099)

[3.4.1 Technical Feasibility 12](#_Toc190070100)

[3.4.2 Operational Feasibility 13](#_Toc190070101)

[3.4.3 Economic Feasibility 13](#_Toc190070102)

[3.5 Requirement Analysis 14](#_Toc190070103)

[3.5.1 Functional Requirement 14](#_Toc190070104)

[3.5.2 Non-Functional Requirements 15](#_Toc190070105)

[Chapter-4 System Design 16](#_Toc190070106)

[4.1 ER Diagram 16](#_Toc190070107)

[4.2 Flow Chart 17](#_Toc190070108)

[4.3 Use Cases 18](#_Toc190070109)

[Chapter-5 Development (Implementation) 21](#_Toc190070110)

[5.1 Overview 21](#_Toc190070111)

[5.2 Development Environment 21](#_Toc190070112)

[5.3 System Implementation Process 21](#_Toc190070113)

[5.4 Deployment on Localhost 22](#_Toc190070114)

[Chapter-6 Testing 23](#_Toc190070115)

[6.1 Overview 23](#_Toc190070116)

[6.2 Testing Phase 23](#_Toc190070117)

[6.3 Testing of System 24](#_Toc190070118)

[Chapter-7 Deployment 28](#_Toc190070119)

[7.1 Overview 28](#_Toc190070120)

[7.2 Deployment Environment 28](#_Toc190070121)

[7.3 Deployment Steps 28](#_Toc190070122)

[7.4 Future Deployment Considerations 29](#_Toc190070123)

[Chapter-8 Maintenance 30](#_Toc190070124)

[8.1 Overview 30](#_Toc190070125)

[8.2 Maintenance Phase 30](#_Toc190070126)

[8.3 Ongoing Monitoring and Feedback 30](#_Toc190070127)

[Chapter-9 Gantt Chart and Project Planning 31](#_Toc190070128)

[9.1 Gantt Chart 31](#_Toc190070129)

[9.2 System Development Activities 31](#_Toc190070130)

[References 32](#_Toc190070131)

[Appendices 33](#_Toc190070132)

# List of figures

[Figure 1: Iterative Waterfall Model 9](#_Toc190070133)

[Figure 2: ER Diagram 16](#_Toc190070134)

[Figure 3: Flow Chart of LMS 17](#_Toc190070135)

[Figure 4: Admin Use Case 18](#_Toc190070136)

[Figure 5: Teacher Use Case 19](#_Toc190070137)

[Figure 6: Student Use Case 20](#_Toc190070138)

[Figure 7: Using wrong credential to test 27](#_Toc190070139)

[Figure 8: Using correct credential to test 27](#_Toc190070140)

[Figure 9: Gantt Chart 31](#_Toc190070141)

# List of tables

[Table 1: Functional Testing 24](#_Toc190070142)

[Table 2: Non-Functional Requirements testing 25](#_Toc190070143)

[Table 3: Contribution Activities 31](#_Toc190070144)

# Introduction

## Background

A Xemon Learning Management System (LMS) is a digital platform designed to streamline the administration, delivery, and tracking of educational and training programs. Commonly implemented in schools, colleges, and universities, LMS platforms have revolutionized the way educational institutions manage learning. However, their use extends beyond academia, playing a crucial role in corporate training, professional development, and even industry-specific sectors like stock market trading.

It must be cost-effective, user-friendly, scalable, and sustainable to ensure long-term efficiency, features include course creation, material upload, content management, assessments, progress tracking, communication tools etc.

The implementation of an LMS not only simplifies course management but also ensures a more organized, accessible, and engaging educational experience, benefiting both educators and learners by fostering a more effective learning environment.

Xemon LMS was designed as a centralized platform where teachers and students can seamlessly interact, share resources, and collaborate effectively. By providing a structured and user-friendly environment, the system enhances the overall learning experience while giving administrators complete control over course management. It can be defined as a software application or web-based application for organizing online learning services for students, teachers, and administrators.

## Purpose

The primary purpose of Xemon LMS is to provide an efficient, interactive, and structured digital learning environment that enhances collaboration between students, teachers, and administrators. It aims to streamline the teaching and learning process by offering a centralized platform for content sharing, assessment management, and progress tracking.

Xemon LMS offers a structured platform where teachers can upload study materials (PDFs, Word documents, videos, presentations), create assignments, and assess student performance. Students can access course content, submit assignments, participate in discussions, and track their progress. Administrators oversee all activities, managing user roles, course enrollments, and activity logging system. The platform also includes a collaborative messaging system, allowing teachers and students to communicate efficiently, enhancing the overall learning experience. A dashboard provides insights into students, teachers, material and classes.

Objectives of Xemon LMS:

* Enable students and teachers to interact in a virtual space where study materials, assignments, and discussions are easily accessible.
* Provide tools for messaging, feedback, and real-time engagement to create an interactive learning experience.
* Allow teachers to upload, organize, and manage learning materials efficiently.
* Automate Assessments & Progress Tracking.
* Ensure learning resources are available anytime, anywhere, making education more accessible to students.
* Administrators full control over user management, course monitoring, and system functionalities.

## Scope

The scope of Xemon LMS covers the entire lifecycle of the Learning Management System (LMS) designed to support large-scale learning environments, Xemon LMS ensures performance, security, and scalability while enhancing collaboration between students, teachers, and administrators.

Xemon LMS provides a centralized and structured platform for managing courses, assessments, student progress, and communication. The system supports various content formats, including PDFs, Word documents, videos, and presentations, making it a flexible and comprehensive tool for modern education. It is designed to work seamlessly across multiple devices, allowing students and educators to access learning materials anytime, anywhere.

**Function/feature of LMS:**

* Upload study materials, create assessments, track student performance, and engage with students through interactive tools.
* Access course content, submit assignments, communicate with instructors and students and track progress.
* Oversee user management, monitor system activity, and ensure the smooth operation of course structures.
* Various content formats, including PDFs, videos, presentations, and documents, can be uploaded to enhance learning materials.
* Courses, lessons, and modules can be efficiently organized, managed, and updated based on the learning requirements.
* An integrated messaging system allows teachers and students to communicate effectively within the platform.
* Notifications and announcements keep students informed about important updates and deadlines.
* Secure authentication and access control ensure data privacy and system integrity.
* A cloud-based infrastructure enables flexibility, seamless access, and future expansion possibilities.

**Limitations**

* Requires internet access for full functionality.
* Maintenance or unexpected downtime can disrupt access to course materials or updates.
* Performance may be impacted during peak usage with high numbers of concurrent users.

# Literature Review

## Introduction

Learning Management Systems (LMS) have become a critical part of modern education and corporate training environments, providing a structured platform for delivering, tracking, and managing learning experiences. The concept of LMS emerged with the rise of digital education tools, offering an integrated system that simplifies content delivery, student assessments, and learner engagement. LMS platforms have evolved over the years, playing a pivotal role in transforming traditional learning environments by enabling e-learning, distance education, and continuous professional development in various sectors.

## Problem Statement

The rapid integration of digital technologies into education has led to the widespread adoption of Learning Management Systems (LMS) across academic institutions and corporate training environments. Despite the growing use of LMS platforms, many challenges persist in ensuring that these systems effectively meet the diverse needs of both educators and learners. Issues such as technological limitations (e.g., internet access, device compatibility), user resistance, and the complexity of the platforms often hinder the optimal use of LMS. Additionally, while LMS platforms offer valuable features like content management, assessment tools, and progress tracking, the lack of personalization, integration with emerging technologies, and scalability in large institutions still pose significant barriers.

Furthermore, the ability of LMS to engage students and improve learning outcomes is not always consistent, and there is a lack of empirical data to measure their long-term effectiveness in various educational settings. As LMS platforms evolve to incorporate advanced technologies such as AI, gamification, and virtual reality, it is essential to address these challenges and gaps in the current systems to ensure that they can provide an inclusive, efficient, and effective learning environment for all stakeholders.

Thus, the problem lies in optimizing the functionality, scalability, and user experience of LMS platforms to meet the growing demand for digital learning solutions while overcoming existing barriers to their successful implementation and usage.

## Technological Evolution

The development of LMS has been shaped by the continuous advancements in technology. Initially, LMS platforms were designed to automate administrative tasks and provide a basic framework for course management. As technology advanced, LMS platforms began incorporating cloud computing, mobile learning, and social learning capabilities. This transformation allowed for greater flexibility, scalability, and accessibility, as LMS systems could now be accessed anytime and anywhere via the internet. Recent innovations have also integrated AI, machine learning, and virtual reality (VR), further enhancing the personalization and interactivity of learning experiences.

## Existing Solutions:

Examples of Established LMS Platforms:

* **Blackboard**: Known for its robust course management and grading features, it offers extensive online teaching and learning tools, including discussion boards and content management.
* **Moodle**: An open-source LMS praised for its flexibility and customization options, Moodle provides a range of plugins and modules to tailor the system to specific needs.
* **Canvas**: Noted for its user-friendly interface and strong integration capabilities, Canvas supports a variety of multimedia content and offers tools for real-time feedback and collaboration.
* **360Learning**: 360Learning is an all-in-one learning solution that combines the features of an LXP LMS, and Academies with collaboration learning.
* **Meroschool**: Mero School is an online learning platform that provides school, and bachelor’s level students with online video classes in areas such as engineering, business, and science.

## Features and Functionalities

LMS platforms typically include a range of features designed to support various learning needs. Functionalities include:

* **Content Management**: The ability to upload and organize learning materials in multiple formats (videos, PDFs, presentations, quizzes).
* **Assessment Tools**: LMS allows the creation of automated quizzes, exams, and assignments, providing instant feedback to learners.
* **User Management**: Admins can manage students, instructors, and other stakeholders by assigning roles, setting permissions, and monitoring participation.
* **Collaboration and Communication**: Integrated tools such as messaging systems, discussion forums, and announcements enable interaction between students and teachers, facilitating collaborative learning.
* **Progress Tracking**: Real-time analytics and reporting allow for tracking student engagement, course completion, and overall performance. These features help instructors identify learning gaps and address them promptly.

## Impact on Learning Outcomes

Research has shown that LMS can significantly impact student engagement, performance, and retention. By offering a centralized hub for learning materials, LMS platforms provide students with easy access to resources, encouraging self-paced learning. Features like progress tracking and personalized learning pathways have been linked to improved student outcomes, as they offer a tailored educational experience. Furthermore, LMS platforms enable immediate feedback through automated assessments, promoting a more interactive and responsive learning environment.

## Challenges and Limitations

While LMS platforms offer numerous benefits, they also face several challenges. One of the issues is technology-related constraints, such as poor internet connectivity or device compatibility, which may limit access for some students. Another challenge is user resistance, as both students and instructors may struggle to adapt to new systems or feel overwhelmed by the platform’s complexity. Additionally, content overload can become a concern when too much material is provided, leading to information fatigue among learners. Some LMS platforms also struggle with scalability when faced with large volumes of users or content, causing performance issues during peak times. Finally, there are concerns about data privacy and security, as sensitive user data must be protected within these platforms to avoid breaches.

## Gaps in Current Research

Despite the extensive literature on LMS, there are still gaps that need further exploration. For instance, most existing research focuses on LMS's impact in formal education settings; however, the application of LMS in non-traditional learning environments such as corporate training or informal education remains underexplored. Additionally, as newer technologies such as AI and VR become integrated into LMS, more research is needed to understand how these technologies impact learning outcomes. Further studies are also required to explore how LMS platforms can be tailored for specific learner demographics, including those with special educational needs or non-native language learners.

## Future Directions

The future of LMS will likely involve greater integration with emerging technologies, such as AI-driven personalized learning experiences, gamification, and advanced analytics to predict student success and provide targeted interventions. Additionally, the use of mobile platforms and apps will continue to expand, offering learners the flexibility to learn on-the-go. Further research is also needed on the effectiveness of LMS in enhancing teacher-student interaction, fostering collaboration, and improving student retention in both online and hybrid learning environments.

## Conclusion

The literature on Learning Management Systems highlights their transformative role in education and corporate training by enabling more personalized, efficient, and accessible learning experiences. While LMS platforms offer several advantages, there are still challenges related to technology, user adaptation, and data security. As technological innovations continue to shape the learning landscape, LMS platforms will evolve to meet the demands of learners and educators alike. Ongoing research will be essential to address these challenges and fully leverage the potential of LMS in various educational contexts.

# Methodology

## System Architecture and Design

The Xemon Learning Management System (LMS) is designed with a focus on scalability, flexibility, and high availability to meet the demands of students, teachers, and administrators. The system will leverage a cloud-based infrastructure, ensuring accessibility, reliability, and the ability to handle varying traffic loads and large-scale user engagement. The design incorporates essential components that ensure smooth functionality, ease of use, and data security.

The Xemon Learning Management System (LMS) is designed with a robust architecture that ensures scalability, security, and seamless user experience. The User Interface (UI) Layer provides a user-friendly platform for students, teachers, and administrators to interact with the system, supporting both web and mobile applications for easy access to course content, assignments, progress tracking, and communication. The Application Layer contains core functionalities like course management, assessments, content delivery, and communication tools, allowing teachers to create and manage courses, track student performance, and facilitate interactive learning environments. The Database Management System (DBMS) ensures secure, efficient storage of all critical data, including user profiles, course content, performance metrics, and communications, while maintaining fast data retrieval and compliance with privacy regulations.

For Content Delivery and Storage, ensure quick access to media-rich content, with secure cloud-based storage and backup systems for high availability and data integrity. Security and Access Control are integral, with encryption and role-based access control (RBAC) to protect sensitive data. Together, these components ensure a flexible, secure, and efficient learning platform, providing a seamless experience for all users while adhering to compliance standards

## Development Process

The development process for the Xemon Learning Management System (LMS) follows a well-structured, phased approach to ensure the platform meets educational objectives while being user-friendly, scalable, secure, and reliable.

For Xemon LMS, following the Iterative Waterfall Model means that the project will be developed in the Iterative approach allows for revisiting previous phases after each iteration, making it more adaptable to changes and refinements during development. Below is the development process tailored for Xemon LMS based on the Iterative Waterfall Model:

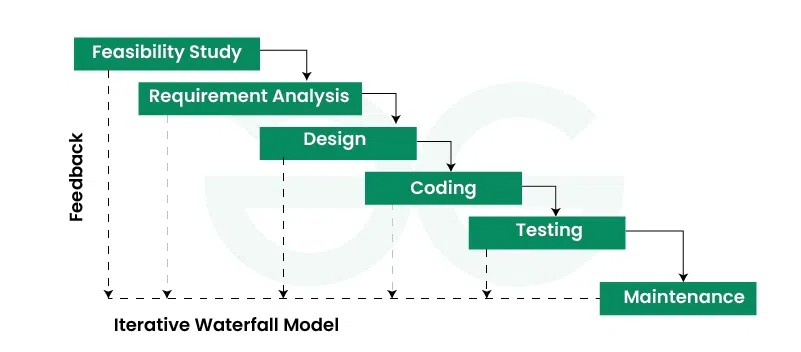


Figure 1: Iterative Waterfall Model

1. **Requirement Gathering & Analysis:**

The development process for Xemon LMS will start with the comprehensive gathering and analysis of all project requirements. At the end of this phase, a detailed Requirements Document will be produced that outlines all necessary features and functional specifications for the system. In this phase, the following will occur:

* Stakeholder Interviews: Understanding the needs of students, teachers, and administrators to ensure the system addresses the right features, such as course management, progress tracking, and assessments.
* System Specifications: Defining the business, technical, and functional requirements, ensuring alignment with user needs and the platform’s goals.
* Market Research: Analyzing other LMS platforms to capture best practices and features that can be implemented in Xemon LMS.

1. **System Design:**

Once requirements are fully defined and documented, the design phase will begin. This phase will result in detailed Design Documentation which will serve as the blueprint for development. The design will be structured as follows:

* System Architecture: The overall architecture of Xemon LMS will be designed, including the User Interface (UI) layer, application layer, database structure, and cloud-based infrastructure.
* Database Design: The schema for the relational database will be planned, ensuring secure and fast access to structured data, such as user profiles, courses, assignments, and performance metrics.
* UI/UX Design: A user-friendly and responsive interface for students, teachers, and administrators will be designed, ensuring accessibility and ease of use.

1. **Implementation (Development):**

With the design in place, the development team will begin coding and implementing the core functionalities of the LMS. Each feature will be implemented in stages, with a focus on completing a functional component in each iteration. At the end of each iteration, the feature will be tested and reviewed for improvements or refinements. Develop the frontend with HTML, CSS, and JavaScript and implement backend functionalities using PHP and MySQL. Key elements that will be developed include:

* Course Management: Teachers can create, manage, and organize courses and assignments.
* Assessment: Tools to create quizzes, assignments, as well as testing systems.
* Communication Features: Messaging, and announcements to facilitate interaction between teachers and students.

1. **Testing (Validation & Verification):**

Testing will be conducted iteratively as each module is completed to ensure the quality and functionality of the system. Each testing phase will be followed by bug fixes and improvements based on the results of the tests. Testing will occur in several forms:

* Unit Testing: Testing individual components of the system to ensure they work as intended.
* Integration Testing: Ensuring different system components, such as user management, assessments, and progress tracking, work together seamlessly.
* System Testing: Full-scale testing to validate the overall system performance, security, and user experience.
* User Acceptance Testing (UAT): Involving stakeholders such as teachers and students to test the system in real-world conditions and provide feedback for refinement.

1. **Deployment:**

Once testing has been completed and all issues have been addressed, Xemon LMS will be deployed to the cloud-based production environment. Deployment will be staged and monitored closely to ensure that any issues or performance concerns are addressed promptly. The deployment will include:

* Cloud Hosting: The LMS will be deployed on a cloud platform (e.g., AWS, Azure) to ensure scalability, flexibility, and global accessibility.
* Data Migration: Migrating any existing course content, user data, and performance data into the new system.
* Final Deployment: The fully functional LMS will be released for use by students, teachers, and administrators.

1. **Post-Deployment Maintenance:**

After the system is live, continuous monitoring will be implemented to ensure smooth operation. Periodic updates and improvements will be implemented in iterations based on ongoing user feedback and system performance analysis. This phase will include:

* Bug Fixing: Immediate attention to any bugs or issues reported by users.
* Feature Enhancements: Based on user feedback, additional features or modifications will be planned for future iterations.
* Performance Monitoring: Ensuring the system remains scalable and handles traffic effectively as more users engage with the platform.
* Security Updates: Keeping the system up to date with security patches to protect user data.

## Proposed System

The proposed Learning Management System (Xemon LMS) is designed to provide a scalable, secure, and user-friendly platform that enhances online learning for students, teachers, and administrators. It will streamline course management, assessment tracking, and communication interactive learning while maintaining security, scalability, and ease of access.

**Admin Dashboard:**

* Full control over user management, manage components, and monitoring system activity.

**Teacher Dashboard:**

* Teachers can upload study materials, create courses, and manage assignments.
* Assessment tools allow the creation of quizzes and assignments.
* Performance tracking and real-time feedback system for monitoring student progress.

**Student Dashboard:**

* Access to course content, assignments, quizzes, and announcements.
* Ability to communicate with instructors and participate.
* Progress tracking to help students monitor their progress.

**Efficient Course & Content Management:**

* Easy course creation and organization for structured learning.
* Supports multiple content formats (PDFs, videos, presentations, documents).
* Content delivery for a progressive learning experience.

**Secure and Scalable System:**

* Role-based access control to protect sensitive user data.

**Enhanced Communication & Collaboration:**

* Built-in messaging system for teacher-student communication.
* Announcement boards for collaborative learning.

## Feasibility Analysis

The feasibility analysis of Xemon LMS assesses the technical, operational, economic, and legal aspects to determine its viability and success in real-world implementation.

### Technical Feasibility

The Xemon Learning Management System (LMS) is designed to provide a user-friendly, secure, and efficient platform for managing online education. The system is built to be easily maintainable, requiring minimal developer intervention while ensuring accessibility for users with basic technical knowledge.

* **System Architecture:** The LMS will be built using a localhost-based architecture using XAMPP, which provides an Apache server, MySQL database, PHP, and phpMyAdmin for database management.
* **Hardware** **Requirements**: Minimum 4GB RAM, 10GB free disk space, and a multi-core processor for smooth execution.
* **Software** **Requirements**: XAMPP for local server management, PHP for backend logic, MySQL for database handling, and HTML, CSS, Bootstrap for front-end design.
* Role-based access control (RBAC) for teachers, students, and admins.
* System updates can be performed by modifying PHP scripts and database schemas.

### Operational Feasibility

The Xemon LMS is designed for efficient, user-friendly operation, ensuring smooth functionality for students, teachers, and administrators.

* The LMS is fully responsive using Bootstrap, ensuring compatibility across desktops, tablets, and mobile devices.
* The dashboard interface will be designed for easy navigation by students, teachers, classes, subjects and downloaded materials.
* Privacy policies for student data protection and user consent agreements implemented.
* A simple and intuitive dashboard tailored for each user role.
* User Roles and Functions
  + Admin: Manages users, and oversees course and activities.
  + Teacher: Uploads courses materials, creates assignments & announcements and tracks student progress.
  + Student: Join in courses, accesses study materials, submits assignments, and participates in discussions.

### Economic Feasibility

The Xemon LMS is designed to be a cost-effective solution that minimizes development expenses while maximizing benefits, such as improving education services and reducing administrative workload.

* **Software Costs**: No major costs since XAMPP, PHP, MySQL, HTML, CSS, and Bootstrap are open-source and free.
* **Development Time**: Estimated 2-3 months for a functional prototype.
* **Hardware Costs**: Since the system runs on localhost, no additional server costs are needed.
* No server hosting fees (as it runs on localhost).
* Minimal maintenance costs since PHP and MySQL-based systems are easy to update.
* **Return on Investment (ROI):**
  + Institutions can save costs on third-party LMS platforms.
  + The system reduces manual workload by automating student progress tracking, assignment submissions, and grading.
* **Cost-Benefit Analysis:**
  + High benefits with low initial investment due to the use of open-source technologies.
  + Time-saving for teachers and students due to automation in assessment and reporting.

## Requirement Analysis

### Functional Requirement

Functional requirements define the core operations and features of the Xemon LMS that ensure smooth learning and management.

* **User Management**
  + Users (Admins, Teachers, and Students) can register, log in, and manage their profiles.
  + Admins can add, update, and delete user (teachers and students) accounts.
* **Course Management**
  + Teachers can create, edit, and delete courses, classes and materials.
  + Teacher creates announcement and event.
  + Students can enroll in courses and access study materials.
  + Courses should support multiple content formats (PDFs, videos, quizzes, and documents).
* **Assessment and Evaluation**
  + Teachers can create quizzes and assignments.
  + The system provides automated grading for multiple-choice questions.
  + Students can view feedback after assignment submission.
* **Progress Tracking and Analytics**
  + Teachers can track student progress.
  + Students can monitor their own progress and performance.
* **Communication and Collaboration**
  + Integrated messaging system for teacher-student communication.
  + Discussion forums for peer collaboration and knowledge sharing.
* **Security and Access Control**
  + Role-based access control (RBAC) ensures admins, teachers, and students have appropriate permissions.

### Non-Functional Requirements

Non-functional requirements ensure system **performance, security, scalability, and usability**.

* **Performance & Scalability**
  + The system must handle multiple users concurrently without performance issues.
  + Efficient database queries ensure fast data retrieval.
  + Cloud infrastructure support for future scalability.
* **Security & Data Protection**
  + Compliance with data privacy regulations to protect student and course data.
  + Role Based Access Control for users (admins, teachers and students).
* **Usability & Accessibility**
  + A user-friendly interface with intuitive navigation.
  + The LMS should be responsive and work across desktop, tablet, and mobile devices.
* **Reliability & Availability**
  + The system should have high uptime on the local XAMPP server.
  + Regular backups to prevent data loss.
* **Maintainability & Extensibility**
  + Modular design to allow future updates and feature enhancements.
  + Detailed documentation for developers and admins.
  + Logs should be maintained for user activities and system performance.

Upon completion of each requirement, continuous feedback is gathered from intended users to ensure improvements and refinements, aligning with the principles of the iterative waterfall model.

# System Design

## ER Diagram

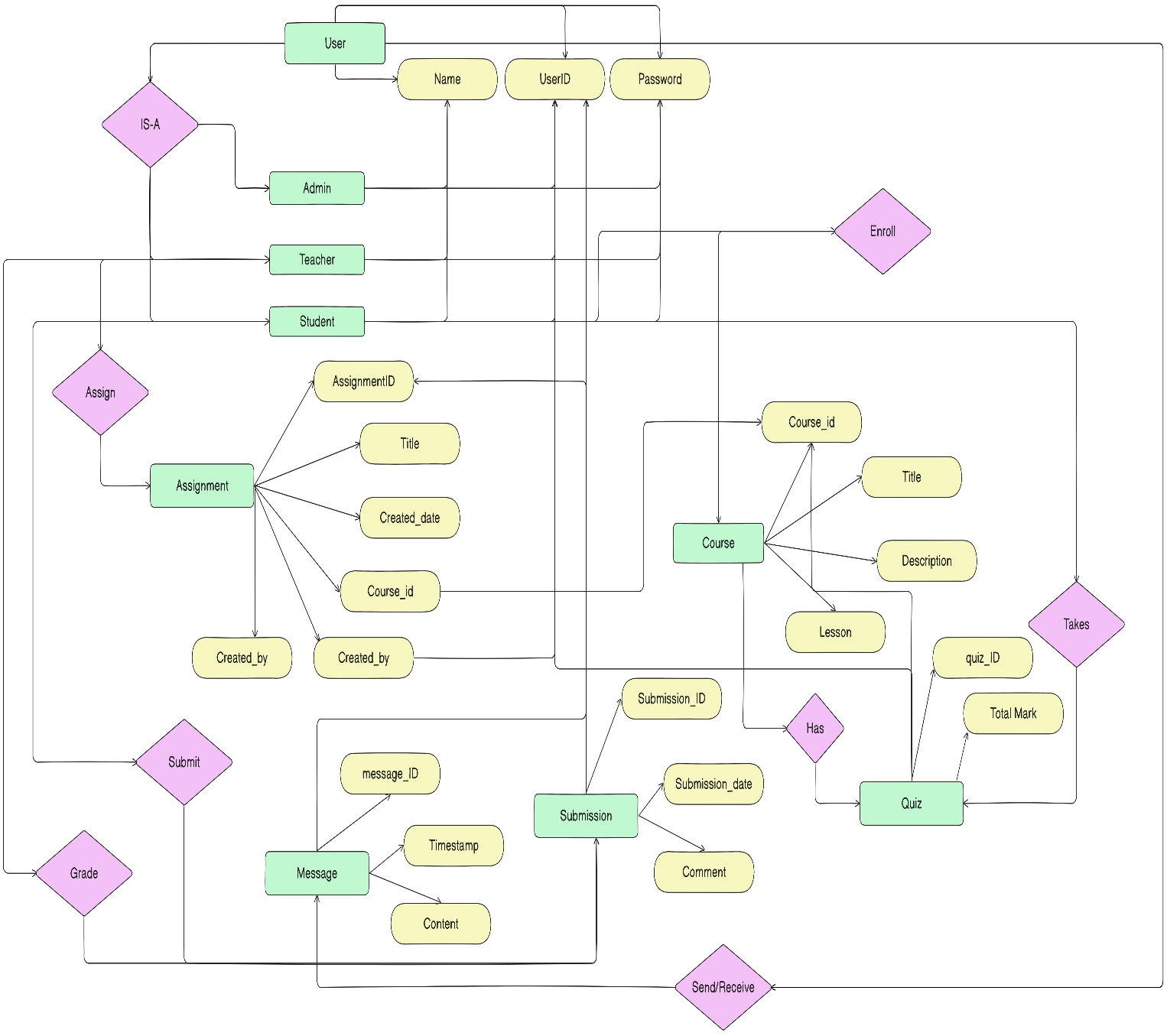


Figure 2: ER Diagram

## Flow Chart

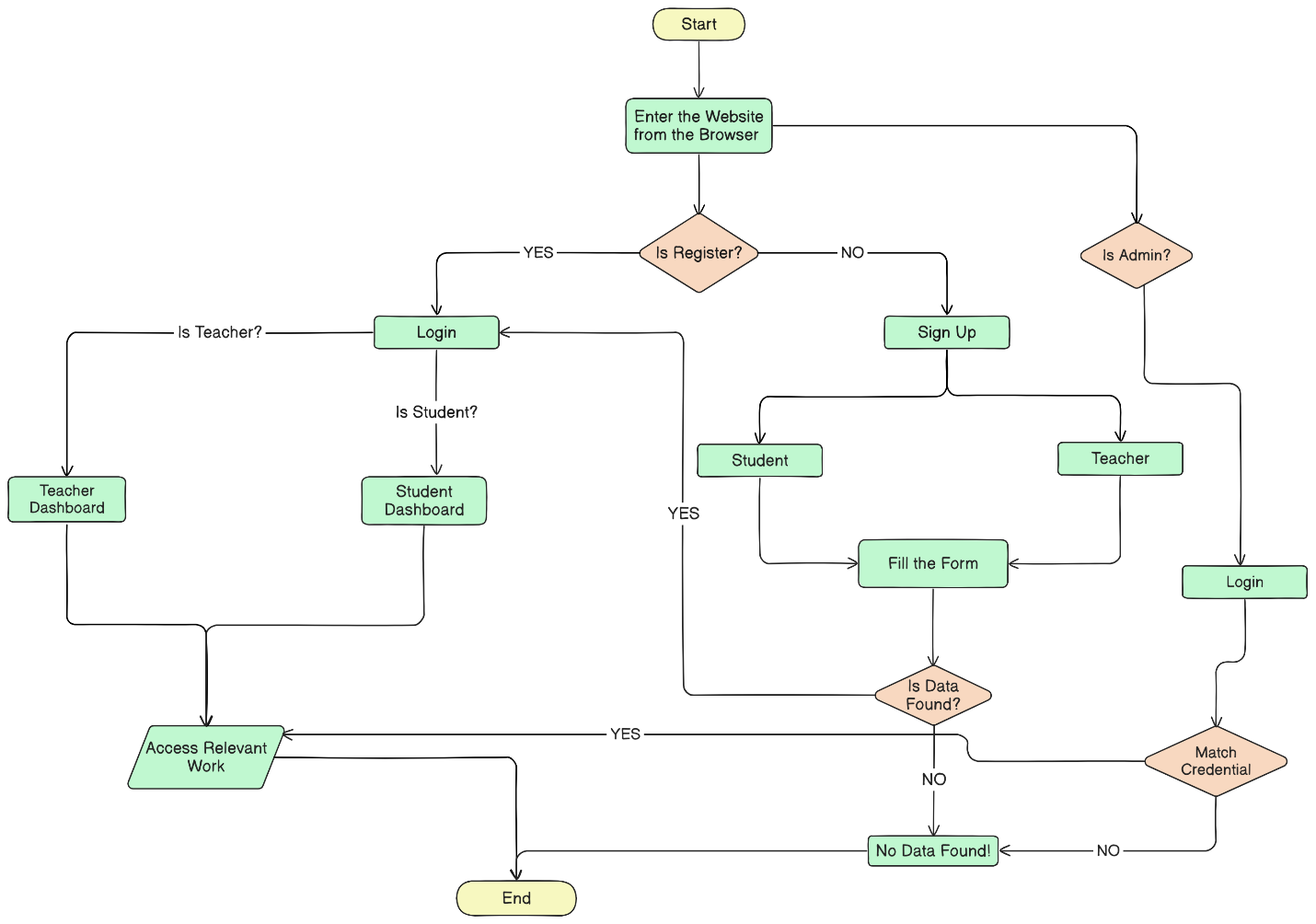


Figure 3: Flow Chart of LMS

## Use Cases

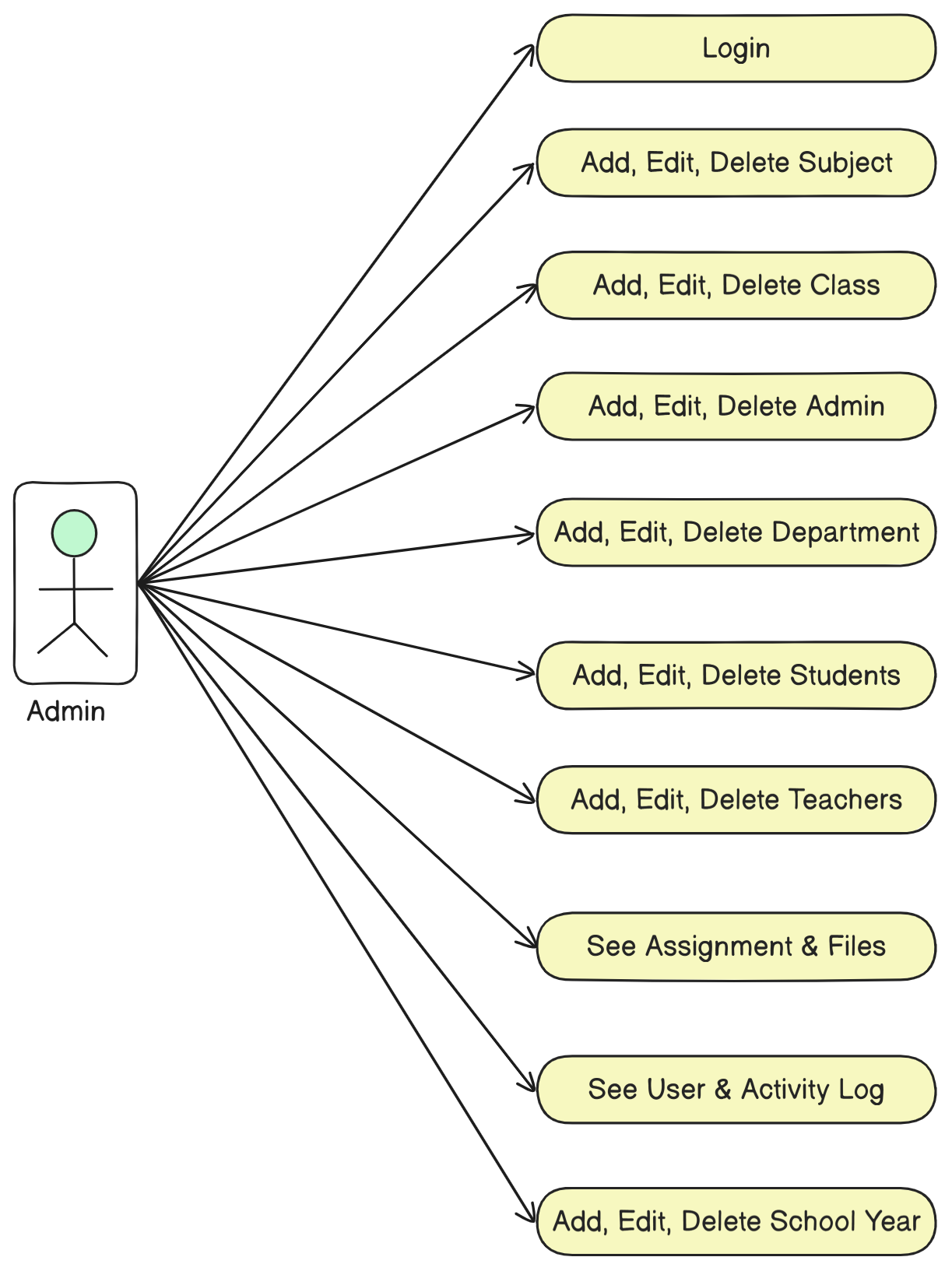


Figure 4: Admin Use Case

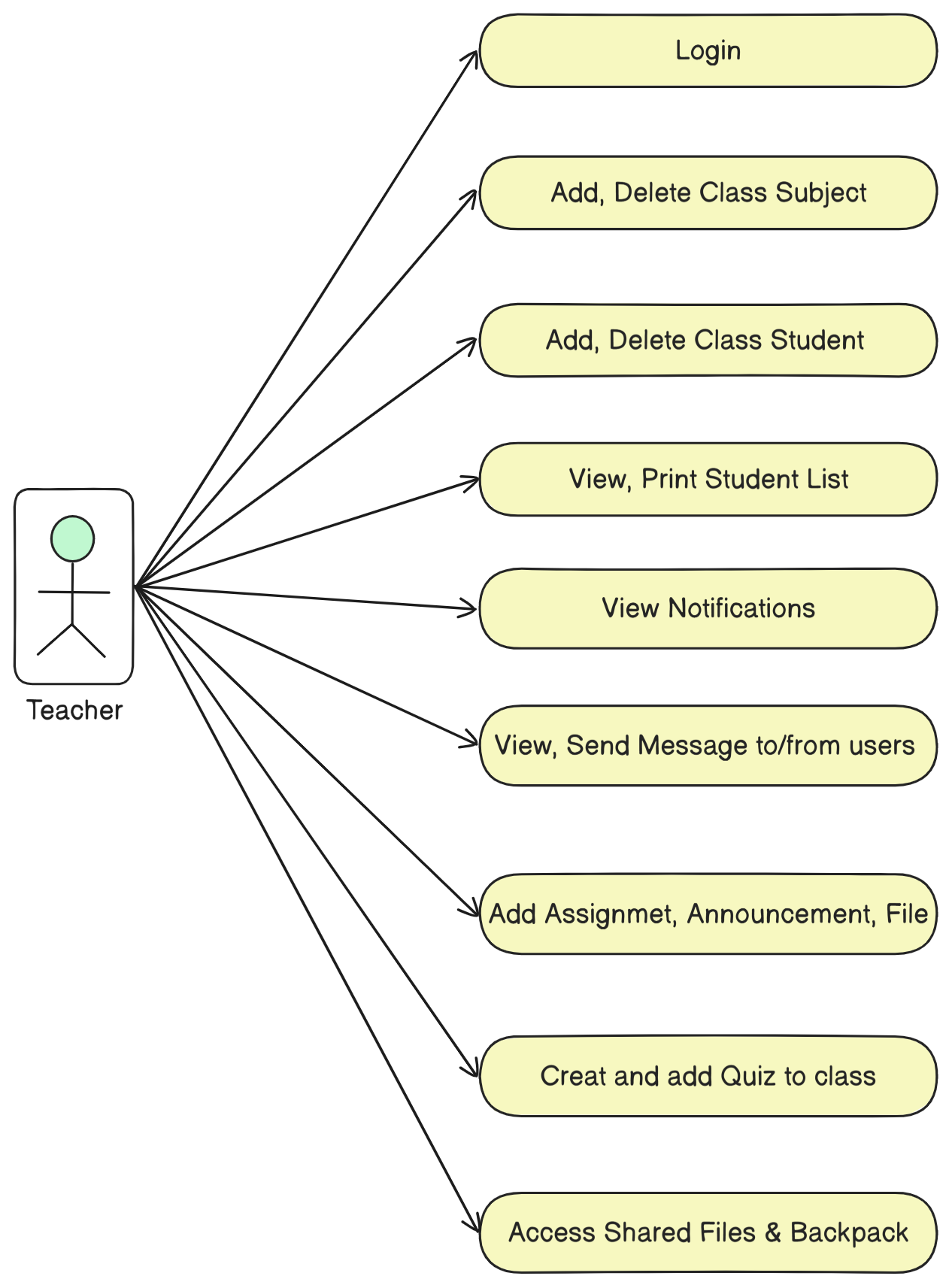


Figure 5: Teacher Use Case

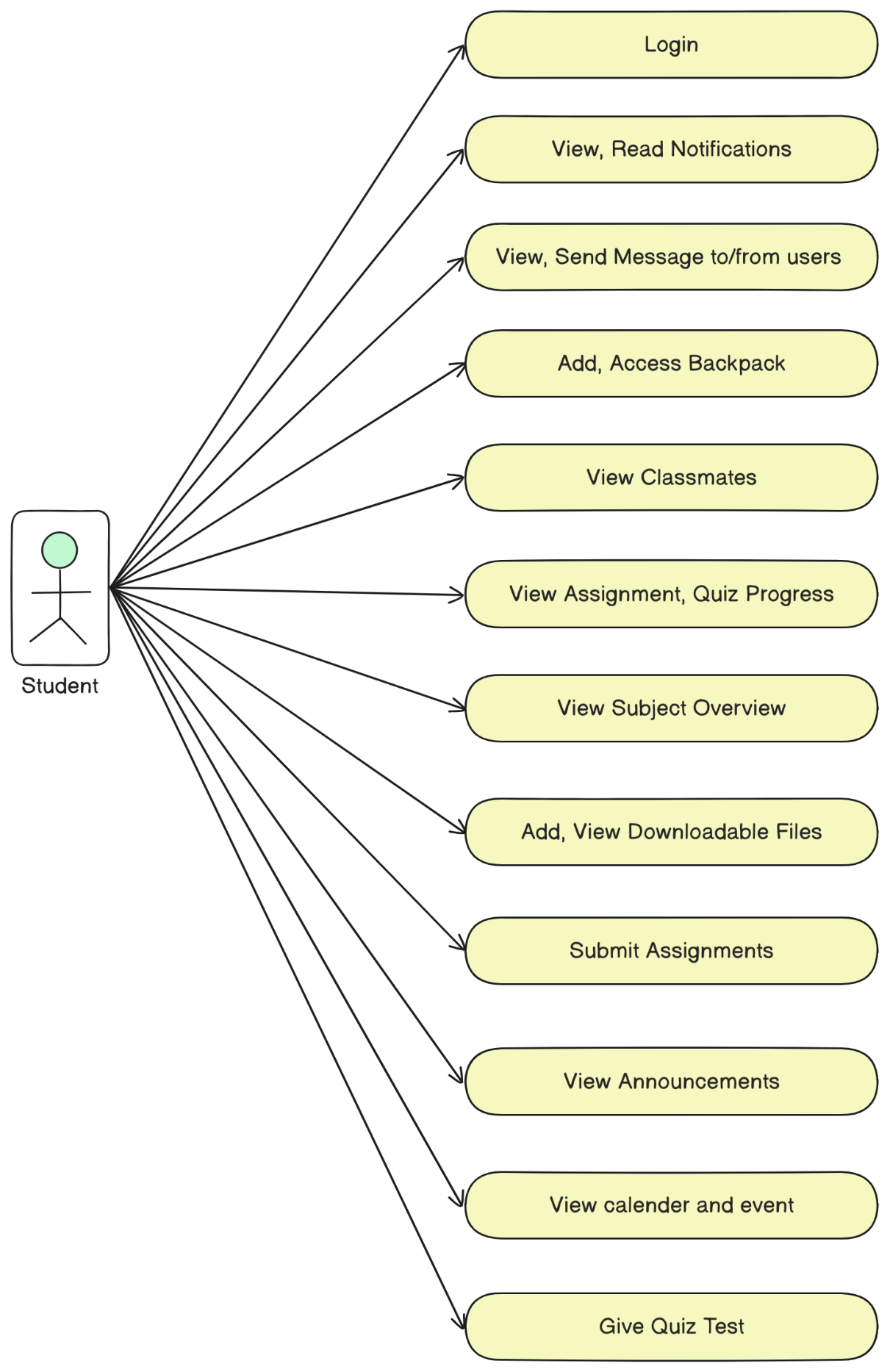


Figure 6: Student Use Case

# Development (Implementation)

## Overview

The implementation phase focuses on transforming the system design into a fully functional Learning Management System (LMS). It involves coding, integrating various system components, testing, and refining the system for optimal performance. The development follows an iterative waterfall model, ensuring continuous feedback and improvements from intended users.

## Development Environment

The LMS is developed using XAMPP (Localhost) with the following technologies:

* Frontend: HTML, CSS (Bootstrap), JavaScript, jQuery
* Backend: PHP (Server-side scripting)
* Database: MySQL (Relational Database Management System)
* Server: Apache (via XAMPP)
* Security: Role-based authentication

## System Implementation Process

**Step 1: Backend Development**

* **User Authentication**: Implemented using PHP sessions and MySQL to manage user roles (Admin, Teacher, Student).
* **Database Configuration**: Tables for users, courses, assignments, assessments, progress tracking, and files etc.
* **Course Management System**: Teachers can upload study materials, create assessments, and track student progress.
* **Assessment & Grading System**: Automated quiz grading and manual assignment evaluation.

**Step 2: Frontend Development**

* User Dashboard: Intuitive interface for students, teachers, and admins to interact with the LMS.
* Responsive UI: Optimized for desktops, tablets, and mobile devices using Bootstrap.
* Course Access & Navigation: Students can browse and access in courses, view study materials, and submit assignments.

**Step 3: Database Integration**

* Tables and relationships were designed in MySQL.
* Secure and optimized queries ensure efficient data retrieval and storage.
* Implemented CRUD (Create, Read, Update, Delete) operations for course and user management.

**Step 4: Security Implementation**

* Role-based authentication to restrict unauthorized access.
* Password hashing for secure user credentials.

**Step 6: Testing and Debugging**

* Unit Testing: Ensuring individual modules (authentication, course management, assessments) work correctly.
* Integration Testing: Verifying smooth interaction between frontend, backend, and database.
* User Acceptance Testing (UAT): Gathering feedback from intended users to refine functionality.

## Deployment on Localhost

After successful testing, the system was deployed on XAMPP (Localhost) for final validation before moving to a live server.

The implementation phase successfully converts the design into a working Xemon LMS with essential features like user authentication, course management, assessments, and reporting. Through iterative feedback and testing, the system is continuously refined for better usability and performance.

# Testing

## Overview

After the development of Xemon Learning Management System (LMS), thorough testing and maintenance are essential to ensure system stability, security, and usability. This phase ensures the LMS operates smoothly, meets functional requirements, and provides a seamless experience for users.

## Testing Phase

The testing process is conducted in multiple stages to identify and resolve potential issues before full deployment.

* **Unit Testing**
* Each module (user authentication, course management, assessment system) is tested independently.
* Ensures that individual components function as expected without dependencies.
* **Integration Testing**
* Verifies smooth interaction between frontend, backend, and database.
* Tests data flow between different modules (e.g., student access material, course enrollment, submission of assignments, Quiz test).
* **System Testing**
* Evaluates the entire system under real-world conditions to check performance and reliability.
* Ensures compatibility with different devices (desktop, mobile, tablets).
* **User Acceptance Testing (UAT)**
* Conducted with teachers, students, and administrators to gather feedback.
* Ensures the system meets user needs and is easy to navigate.
* **Security Testing**
* Tests role-based authentication to prevent unauthorized access.
* Validates data encryption and secure storage of user credentials.
* Assesses the protection of sensitive information like students.
* **Performance Testing**
* Evaluates system response time, load handling, and scalability.
* Ensures the system performs efficiently with multiple users accessing courses and assessments simultaneously.

## Testing of System

Table 1: Functional Testing

| **Feature** | **Test Case** | **Expected Outcome** | **Real-World Practice** |
| --- | --- | --- | --- |
| **Admin Panel** | | | |
| Admin Login | Verify login with valid/invalid credentials | Successful login for valid users, error message for invalid users | Secure authentication using encrypted credentials |
| Manage Users (Add/Edit/Delete) | Test user addition, modification, and deletion | User is successfully added, edited, or removed | Role-based access ensures only authorized admins can make changes |
| Manage Subjects, Classes, Departments | Create, update, delete subjects and classes | Data should be correctly saved and updated | University’s structure courses dynamically based on semester needs |
| View Logs (User Log, Activity Log) | Check if all activities are being recorded correctly | Logs should reflect real-time actions | Ensures security and accountability |
| Calendar of Events | Add events and verify they display correctly | Events should appear for all relevant users | Used in institutions for scheduling classes, deadlines, and exams |
| **Teacher Panel** | | | |
| Manage Subjects for Class | Assign subjects to different classes | Subjects should appear in relevant class sections | Teachers dynamically manage their courses in LMS platforms |
| Manage Students (Add, Remove, Print) | Add, remove students, and print student list | The class roster updates correctly | Ensures classroom organization and documentation |
| Send & Receive Messages | Test message sending and receiving between teachers and students | Messages should be delivered in real-time | Enables smooth communication in online learning |
| Add Course Materials (Files, Assignments) | Upload and test access for students | Students should be able to download/view content | Ensures course material availability like Moodle, Google Classroom |
| Create Quizzes & Assignments | Create quizzes and submit responses as a student | The system should grade quizzes automatically | Automated assessment like Google Forms or Coursera |
| **Student Panel** | | | |
| View Notifications | Verify new notifications are displayed correctly | Notifications should appear in the dashboard | Alerts students of deadlines, updates, and announcements |
| Send & Receive Messages | Test student-to-teacher messaging | Messages should be delivered instantly | Enhances student-teacher engagement |
| Store Personal Files (Backpack) | Upload and download files | Files should be saved and retrievable | Similar to OneDrive or Google Drive for students |
| View Classmates, Progress, and Subject Overview | Verify if students can see their class members and progress | Data should be accurate and updated | Ensures transparency of academic performance |
| Submit Assignments & Take Quizzes | Upload assignments and attempt quizzes | Submission should be successful, and quizzes should be graded automatically | Automation improves efficiency in grading |
| View Announcements & Events | Test if students can see all events and announcements | Information should be clearly visible | Keeps students updated like Google Classroom notices |

Table 2: Non-Functional Requirements testing

| **Category** | **Requirement** | **Testing Approach** | **Expected Outcome** | **Real-World Practice** |
| --- | --- | --- | --- | --- |
| **Performance** | Handle multiple users simultaneously | Load testing with 10 users | System should not slow down or crash | Universities & schools handle thousands of students daily |
| **Security** | Role-based access control (RBAC) | Test unauthorized access attempts | Only permitted users should access restricted sections | Ensures data privacy (GDPR, FERPA compliance) |
| **Usability** | User-friendly interface | Conduct usability testing with students and teachers | Easy navigation and accessibility | Similar to modern LMS platforms like Canvas or Blackboard |
| **Scalability** | Support multiple institutions & courses | Stress test with large datasets | The system should still perform efficiently | Universities need to scale as enrollment grows |
| **Maintainability** | Easy system updates and debugging | Test updating a feature/module without breaking others | The system should remain stable | Agile development ensures flexibility |
| **Data Integrity** | Ensure proper storage and retrieval | Test data consistency in database operations | No data loss or corruption should occur | Critical for academic records and grading |
| **Automation** | Auto-grading of quizzes | Verify if quiz answers are correctly evaluated | Results should be instant and accurate | Used in platforms like Udemy, Coursera for quick assessments |

This structured approach ensures that the Xemon LMS meets all user needs while maintaining high standards of security, usability, and efficiency.

**Testing with wrong and correct credential to admin login**

|  |
| --- |
| Figure 7: Using wrong credential to test |
| Figure 8: Using correct credential to test |

# Deployment

## Overview

The deployment phase of the Xemon Learning Management System (LMS) involves installing, configuring, and setting up the system on a local server environment using XAMPP. The deployment process will involve setting up a local environment, configuring the database, and ensuring all dependencies are properly installed.

This ensures smooth functionality and accessibility for students, teachers, and administrators. The deployment process ensures that Xemon LMS is fully functional in a local XAMPP environment.

## Deployment Environment

The system is designed to run on a localhost environment before moving to an online server. The technologies used include:

* Web Server: Apache (via XAMPP)
* Database Server: MySQL
* Backend: PHP
* Frontend: HTML, CSS (Bootstrap), JavaScript, jQuery
* Security Measures: Role-based authentication

## Deployment Steps

**Step 1: Install XAMPP**

* Download and install XAMPP on the local machine.
* Start Apache (for PHP) and MySQL (for database) from the XAMPP Control Panel.

**Step 2: Configure the Database**

* Open phpMyAdmin (http://localhost/phpmyadmin).
* Create a new database named lmsdb.
* Import the SQL database file (lmsdb.sql) into phpMyAdmin.

**Step 3: Setup the LMS Project**

* Copy the LMS project folder to the htdocs directory in XAMPP.
* Modify config.php to update database connection details:

*$servername = "localhost";*

*$username = "root";*

*$password = "";*

*$dbname = "lmsdb";*

**Step 5: Launch the LMS on Localhost**

* Open a web browser and enter:

*http://localhost/lms/*

* The login page should appear for user access.

**Step 6: Testing and Debugging**

* Verify user authentication, course management, file uploads, assessments etc.
* Fix any issues before the system is fully operational.

## Future Deployment Considerations

Once tested on localhost, the LMS can be moved to an **online server** by:

* Hosting it on a cloud service like AWS, DigitalOcean, or Hostinger.
* Using a domain name and SSL certificate for security.
* Implementing load balancing to manage multiple users efficiently.

The deployment process ensures that Xemon LMS is fully functional in a local XAMPP environment. Through iterative feedback and testing, the system is continuously refined for better usability and performance.

# Maintenance

## Overview

Once deployed, Xemon LMS requires ongoing maintenance to address bugs, introduce enhancements, and ensure system uptime. Maintenance ensure Xemon LMS runs efficiently, securely, and meets user needs. With ongoing support and updates, the system remains reliable, scalable, and future-ready, providing an effective learning experience for students and instructors.

## Maintenance Phase

* **Corrective Maintenance**
* Fixes any bugs or errors identified after deployment.
* Resolves compatibility issues with different browsers and devices.
* **Adaptive Maintenance**
* Modifies the system to support new technologies, security updates, or regulatory changes.
* Ensures LMS remains compatible with evolving educational and security standards.
* **Perfective Maintenance**
* Enhances system features based on user feedback and evolving needs.
* Implements UI/UX improvements for better usability.
* **Preventive Maintenance**
* Regular database backups to prevent data loss.
* Monitoring server performance and applying security patches to prevent vulnerabilities.

## Ongoing Monitoring and Feedback

* **User Feedback System:** Allows students, teachers, and administrators to report issues and suggest improvements.
* **Log Analysis:** Regular monitoring of system logs to detect unusual activities or errors.
* **Periodic Updates:** Rolling out new features, performance enhancements, and security patches based on analysis.

# Gantt Chart and Project Planning

## Gantt Chart

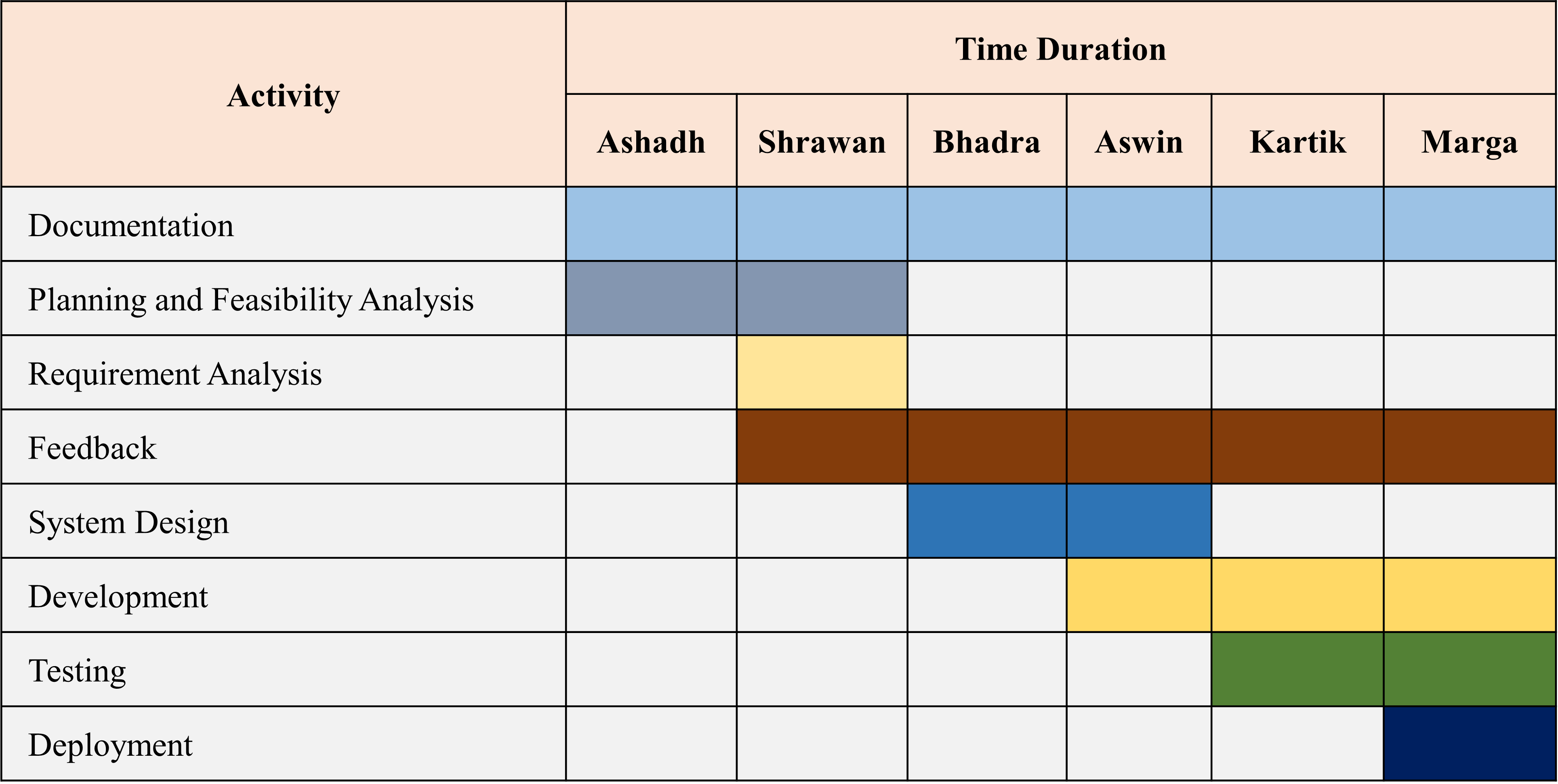


Figure 9: Gantt Chart

## System Development Activities

Table 3: Contribution Activities

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase** | **Activity** | **Responsible Members** | | | |
| **Dadip Bhattarai** | **Pramila Karki** | **Richa Bhandari** | **Sharmila Rai** |
| **Planning** | Requirement gathering, feasibility study, scope analysis. | ✓ | ✓ | ✓ | ✓ |
| **Design** | Creating system architecture, ER diagrams. | ✓ | ✓ | ✓ | ✓ |
| **Development** | Backend and frontend coding, database creation. | ✓ | ✓ | ✓ | ✓ |
| **Testing & Debugging** | Functional testing, security testing, bug fixing. | ✓ | ✓ | ✓ | ✓ |
| **Deployment** | Setting up on XAMPP, configuring localhost testing. | ✓ | ✓ | ✓ | ✓ |
| **Documentation** | Writing Reports. | ✓ | ✓ | ✓ | ✓ |

# References

Al-Busaidi, K. A., & Al-Shihi, H. (2012). Key factors for implementing learning management systems in higher education institutions. *Educational Technology & Society, 15*(4), 362-378.

Ellis, R. (2009). *Field guide to learning management systems.* ASTD Learning Circuits.

Sclater, N. (2016). *Learning analytics: The current state of play in higher education.* EDUCAUSE Review.

Wang, Y. S. (2003). Assessment of learner satisfaction with asynchronous electronic learning systems. *Information & Management, 41*(1), 75-86.

Moodle. (2023). *Understanding LMS functionality and deployment.* Retrieved from <https://docs.moodle.org>

Blackboard Inc. (2022). *Security and scalability in learning management systems.* Retrieved from <https://www.blackboard.com>

eLearning Industry. (2021). *Best practices for LMS implementation.* Retrieved from <https://elearningindustry.com>

W3Schools. (2024). *PHP and MySQL for dynamic web applications.* Retrieved from <https://www.w3schools.com>

Bootstrap. (2024). *Responsive UI design using Bootstrap.* Retrieved from <https://getbootstrap.com>

MySQL. (2024). *Database design and optimization.* Retrieved from <https://dev.mysql.com/doc>

Pressman, R. S. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill.

Sommerville, I. (2019). *Software engineering* (10th ed.). Pearson Education.

Schwaber, K., & Sutherland, J. (2020). *Scrum guide: The definitive guide to Scrum.* Retrieved from <https://www.scrum.org>

OWASP Foundation. (2023). *Best practices for web security and role-based authentication.* Retrieved from <https://owasp.org>

GDPR.eu. (2023). *Ensuring LMS platforms meet data protection regulations.* Retrieved from <https://gdpr.eu>

# Appendices

**Admin Login**

*<?php*

*include('dbcon.php');*

*session\_start();*

*$username = $\_POST['username'];*

*$password = md5($\_POST['password']);*

*$query = mysqli\_query($conn, "SELECT \* FROM users WHERE username='$username' AND password='$password'") or die(mysqli\_error());*

*$count = mysqli\_num\_rows($query);*

*$row = mysqli\_fetch\_array($query);*

*if ($count > 0) {*

*$\_SESSION['id'] = $row['user\_id'];*

*echo 'true';*

*mysqli\_query($conn, "insert into user\_log (username,login\_date,user\_id)values('$username',NOW()," . $row['user\_id'] . ")") or die(mysqli\_error());*

*} else {*

*echo 'false';*

*}*

*?>*

**Admin Login Form**

*<body id="login">*

*<div class="container">*

*<form id="login\_form" class="form-signin" method="post">*

*<h3 class="form-signin-heading"><i class="icon-lock"></i> Please Login</h3>*

*<input type="text" class="input-block-level" id="username" name="username" placeholder="Username" required>*

*<input type="password" class="input-block-level" id="password" name="password" placeholder="Password"*

*required>*

*<button name="login" class="btn btn-info" type="submit"><i class="icon-signin"></i> Sign in</button>*

*</form>*

*</div>*

*<script>*

*jQuery(document).ready(function () {*

*jQuery("#login\_form").submit(function (e) {*

*e.preventDefault();*

*var formData = jQuery(this).serialize();*

*$.ajax({*

*type: "POST",*

*url: "login.php",*

*data: formData,*

*success: function (html) {*

*if (html == 'true') {*

*$.jGrowl("Welcome to Xemon Learning Management System", { header: 'Access Granted' });*

*setTimeout(function () { window.location = 'dashboard.php'; }, 1000);*

*} else {*

*$.jGrowl("Please check your username and password", { header: 'Login Failed' });*

*}*

*}*

*});*

*return false;*

*});*

*});*

*</script>*

*<?php include('script.php'); ?>*

*</body>*

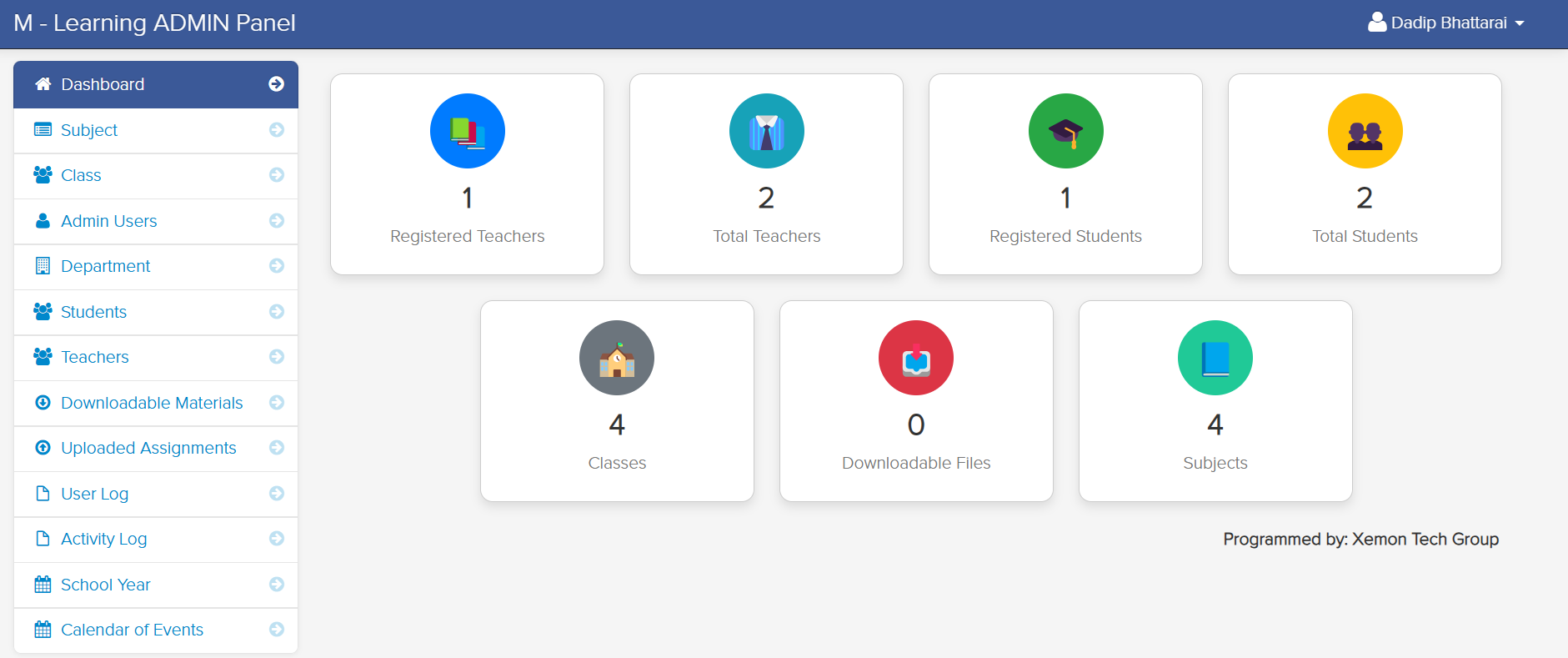
**Database Connection**

*<?php*

*$conn = mysqli\_connect('localhost','root','','lmsdb') or die(mysqli\_error());*

*?>*

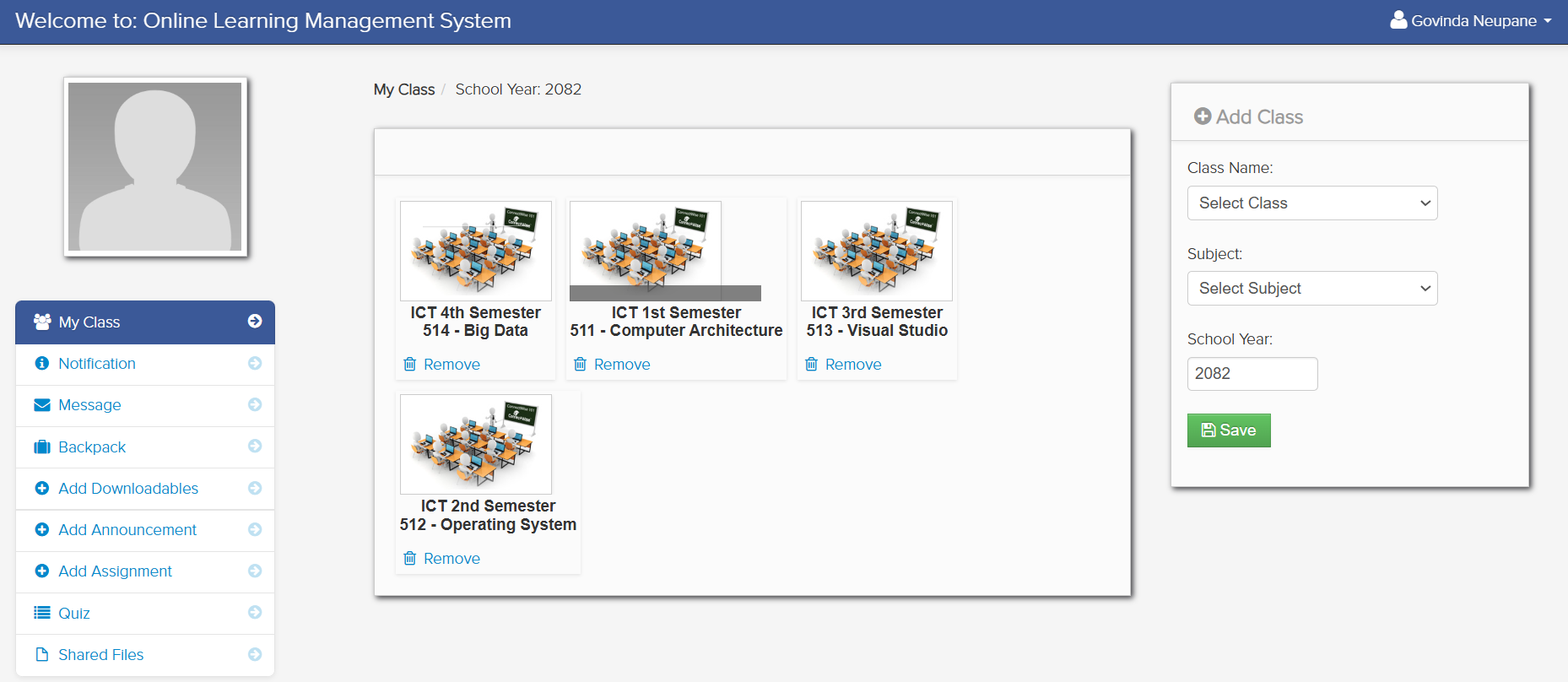
**System Screenshot**

****

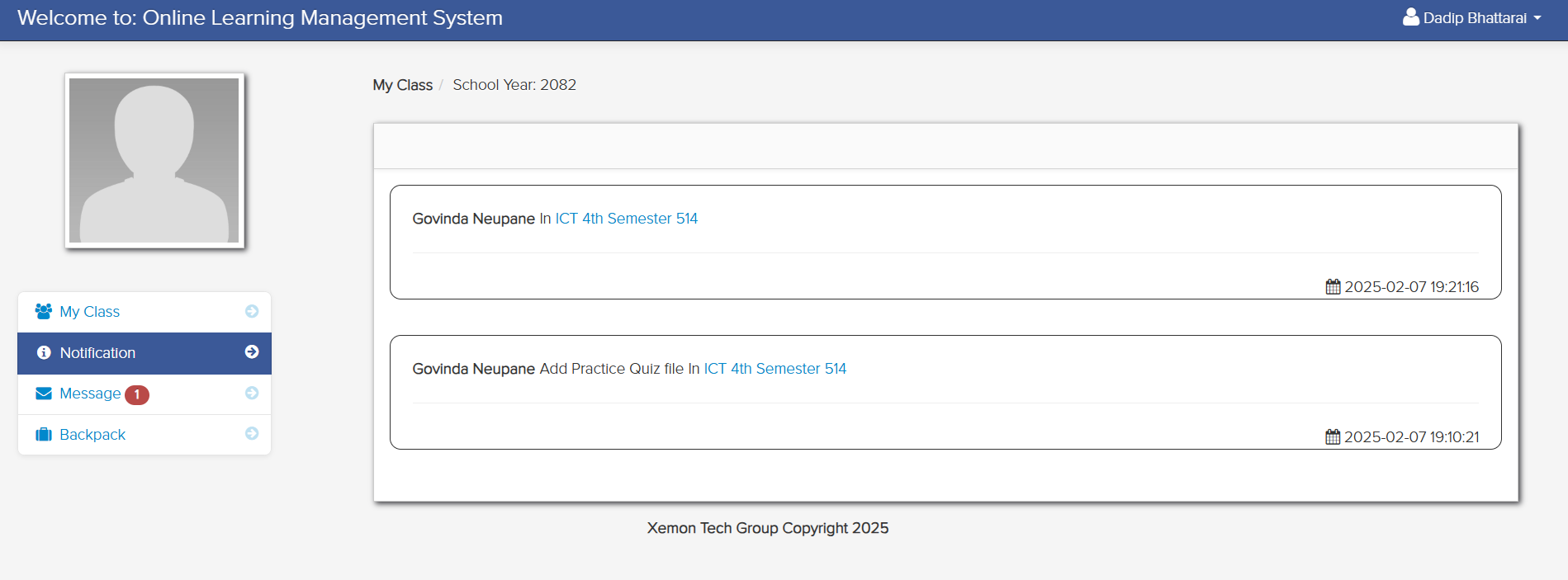
**Admin Dashboard**

****

**Student and Teacher Login Page**

****

**Teacher Dashboard**

****

**Student Dashboard**