

A  
PROJECT REPORT  
ON

Learning Management System

Submitted By

**Narode Sagar Santosh (Seat No.9641)**

**Surkunde Sachin Madhukar (Seat No.9653)**

IN PARTIAL FULLFILLMENT OF  
BACHELOR OF BUSINESS ADMINISTRATION

(COMPUTER APPLICATION) For  
**CHANDMAL TARACHAND BORA ARTS,  
COMMERCE & SCIENCE COLLEGE, SHIRUR.**

**NAAC RE-ACCREDITED: A+ GRADE**

Affiliated To  
**SAVITRIBAI PHULE PUNE UNIVERSITY**  
**ACADEMIC YEAR (2024-25)**



**Chandmal Tarachand Bora College, Shirur**

Tal. Shirur, Dist. Pune (Maharashtra) 412 210

॥ न हि ज्ञानेन सदूरं पवित्रमिह विद्यते ॥

**Department of B.B.A.(C.A.)**

**CERTIFICATE**

This is to certify that project entitled “**Learning Management System**” is submitted by –

Narode Sagar Santosh (Seat No: 9641)

Surkunde Sachin Madhukar (Seat No:9653)

Students of Bachelor of Computer Application Department, Chandmal Tarachand Bora College, and Shiur in fulfilment of the requirement for the Bachelors of Business Administration (Computer Application) Degree of Savitribai Phule Pune university is a record of student’s own study carried under my supervision and Guidance.

**Prof. J. D. Gat**

**Project Guide**

**Prof. K.R.Taksale**

**Head of Department**

**Internal Examiner**

**External Examiner**

**Exam Date: -**

## **Acknowledgement**

This is a great opportunity to acknowledge and to thanks all those persons without who's support and help this project would have been impossible. We would like to add a few heartfelt words for the people who were part of this project in numerous ways.

We would like to thanks to my project guide Prof. J. D. Gat for her indefatigable guidance, valuable suggestion & moral support, constant encouragement and contribution of time for the successful completion of project work. We are very grateful to her, for providing all the facilities needed during the project development. At the outset, I sincerely thank all faculty members of institution for extra efforts to make our session on line inspire of all idea.

We thank our counselors for indispensable support and encouragement through the project. we would like to thank all those who helped me direct or indirectly.

Last but not the least, we would like to acknowledge the ongoing support of my parents and my family members, who patience and encouragement during these long days and nights have been paramount in making this project a reality.

Thank you.

## Index

<b>SR.NO</b>	<b>TITLE</b>	<b>PAGE NO.</b>
1.	<b>Introduction to system</b>	1
2.	<b>Scope of system</b>	2
3.	<b>Objectives of system</b>	3
4.	<b>Proposed system</b>	4
	<b>Analysis</b>	
5	<b>Fact finding technique</b>	5
6.	<b>Feasibility study</b>	6
7.	<b>Hardware and Software requirements</b>	7
	<b>System Design</b>	
8.	<b>Entity relationship Diagram</b>	8
9.	<b>Data Flow Diagram (context level)</b>	9
10.	<b>Data Flow Diagram (First Level)</b>	9
11.	<b>Data Dictionary</b>	10
*	<b>Form design</b>	
12.	<b>Input and output screen</b>	16
13.	<b>Reports</b>	21
14.	<b>Advantages</b>	22
15.	<b>Limitations</b>	23
16.	<b>Future enhancement</b>	23
17.	<b>Bibliography</b>	24

## **Introduction to system**

The purpose of the project "**Learning Management System (LMS)**" is to create a structured and efficient platform for managing educational content, student progress, and communication between instructors and learners. The proposed web-based system aims to streamline the learning process by providing a centralized platform where students and instructors can interact effectively. The system enables students to access study materials, submit assignments, track their progress, and participate in discussions, while instructors can upload content, evaluate student performance, and provide feedback.

The key feature of this project is that it supports **secure user authentication** and **role-based access**, ensuring that students and instructors have appropriate access to resources. The system allows for **one-time registration** and maintains comprehensive records of student progress and performance. Instructors can manage course content, create quizzes, and monitor student activity, while students can access learning materials anytime, enhancing the flexibility of the learning process.

The administrator plays a crucial role in managing the system by overseeing user accounts, handling course enrollment, and generating reports. The LMS provides a user-friendly interface to facilitate easy navigation and quick access to relevant information. The system also includes a notification feature to keep students and instructors updated about course activities and deadlines.

This project aims to improve the learning experience by providing an organized platform that supports efficient content delivery, student engagement, and progress tracking, thereby enhancing overall academic performance.

## **Scope of System**

The Learning Management System (LMS) will store and manage all academic and personal details of the students enrolled in the platform. It will also maintain records of the courses offered, assignments, and assessments. The system will allow for easy updates, modifications, and deletion of student and course details to ensure that the information remains accurate and up to date.

The LMS will enable instructors to upload course materials, create assignments, conduct quizzes, and track student performance. Students will have the ability to access learning resources, submit assignments, participate in discussions, and monitor their progress.

Additionally, the system will have a **notification feature** to inform students about important updates such as new course materials, assignment deadlines, and instructor announcements. Instructors and administrators can also use the system to send alerts regarding exams, changes in course schedules, and other academic events.

The scope of this system extends to providing a structured, secure, and user-friendly platform that facilitates effective learning, communication, and performance monitoring, ensuring a smooth and engaging educational experience.

## **Objective of System**

The objectives of the proposed **Learning Management System (LMS)** are:

- The main objective of this project is to efficiently manage the details of **students**, **courses**, and **instructors** within a single platform.
- It aims to provide a secure and organized system for handling student registration, course enrollment, and academic progress tracking.
- The system will reduce manual work involved in managing student records, course materials, and instructor feedback, thereby improving operational efficiency.
- The LMS will allow instructors to create and upload learning materials, assign tasks, and evaluate student performance.
- It will enable students to access course content, submit assignments, participate in discussions, and track their progress in real-time.
- The system will provide a structured mechanism for communication between students and instructors, including notifications about course updates, deadlines, and announcements.
- The LMS will reduce the need for unnecessary paperwork and manual record-keeping by maintaining all data in a centralized database, ensuring quick access and easy updates.

The ultimate goal of the LMS is to create a seamless and efficient learning environment that enhances the overall educational experience for both students and instructors.

## **Proposed System**

Managing and retrieving information in a manual learning environment can be time-consuming and inefficient. Finding specific details about student progress, course materials, and instructor feedback often requires going through multiple records and files, making the process slow and error-prone. The manual system increases the workload for both students and instructors, leading to delays in communication and academic updates.

The proposed **Learning Management System (LMS)** is a web-based application designed to simplify and streamline the management of educational resources and student progress. The system will provide a centralized platform where instructors, students, and administrators can manage and access relevant information efficiently.

The LMS will enable students to register, access study materials, submit assignments, and track their progress. Instructors will be able to upload content, create and evaluate assessments, and provide feedback in real-time. The system will also support secure login and role-based access to ensure that data is protected and accessible only to authorized users.

The system aims to reduce the dependency on manual record-keeping and paperwork, improving the overall efficiency of the learning process. It will also provide a structured mechanism for sending notifications about course updates, deadlines, and announcements, ensuring that all stakeholders stay informed.

The proposed system will create a more organized and responsive learning environment, enhancing the overall academic experience for both students and instructors.

## **Fact Finding Techniques**

A software product always begins with identifying the needs of the users. These needs may either be recognized directly by the user or emerge from existing practices that require automation or improvement. The initial requirements are often informal and lack standardization, making it necessary to refine them into a formal set of requirements. This phase focuses on converting informal user needs into a well-defined **System Requirements Specification (SRS)**. The requirement phase consists of the following key stages:

**1. Problem Analysis**

The goal of problem analysis is to gain a clear understanding of the requirements from both the client and the end users. This involves conducting interviews with stakeholders, analyzing existing processes, and identifying gaps or inefficiencies in the current system.

**2. Requirement Specification (SRS)**

The SRS serves as a formal agreement between the client (user) and the developer, outlining the functionality, features, and performance expectations of the system. It defines what the system will do and sets the foundation for system design and development.

**3. Requirement Validation**

This stage involves verifying that the requirement specification document is complete, accurate, and free from errors. Common issues like incorrect facts, inconsistencies, and ambiguous requirements are identified and resolved to ensure the system meets user expectations.

By following these fact-finding techniques, the proposed **Learning Management System (LMS)** will be designed to meet user needs, improve efficiency, and provide a structured learning environment.

## **Feasibility Study**

A **feasibility study** is an analysis that evaluates various aspects of a project, including **economic, technical, legal, and scheduling** factors, to determine the likelihood of its successful completion. It helps to identify potential challenges and opportunities, allowing project managers to make informed decisions before committing significant time and resources.

### **Benefits of a Feasibility Study**

#### **1. Risk Assessment:**

A feasibility study helps in identifying potential risks and challenges associated with the project, enabling project managers to devise strategies to mitigate them.

#### **2. Informed Decision-Making:** It provides detailed insights into the project's requirements, potential obstacles, and expected outcomes, helping stakeholders make data-driven decisions.

#### **3. Cost and Resource Management:**

By estimating the costs, resource requirements, and timeframes, the feasibility study ensures that the project stays within budget and is completed on time.

#### **4. Market Analysis and Competition:**

The study includes an evaluation of market conditions and competition, allowing the project team to adjust their approach for better market positioning.

### **Types of Feasibility**

#### **1. Technical Feasibility:**

Evaluates whether the proposed system can be developed using available technology, tools, and expertise.

#### **2. Economic Feasibility:**

Assesses the cost-effectiveness of the project by comparing expected benefits with estimated costs.

#### **3. Operational Feasibility:**

Determines whether the proposed system can function smoothly within the existing operational environment.

#### **4. Legal Feasibility:**

Examines if the project complies with legal and regulatory requirements.

By conducting a comprehensive feasibility study, the proposed **Learning Management System (LMS)** will be evaluated for its practicality and potential success, ensuring that it meets the educational goals and technical requirements of the institution.

## **1.HARDWARE REQUIREMENTS**

Processor: Pentium III onwards

RAM :512 MB & Above

Hard disk :1GB & Above

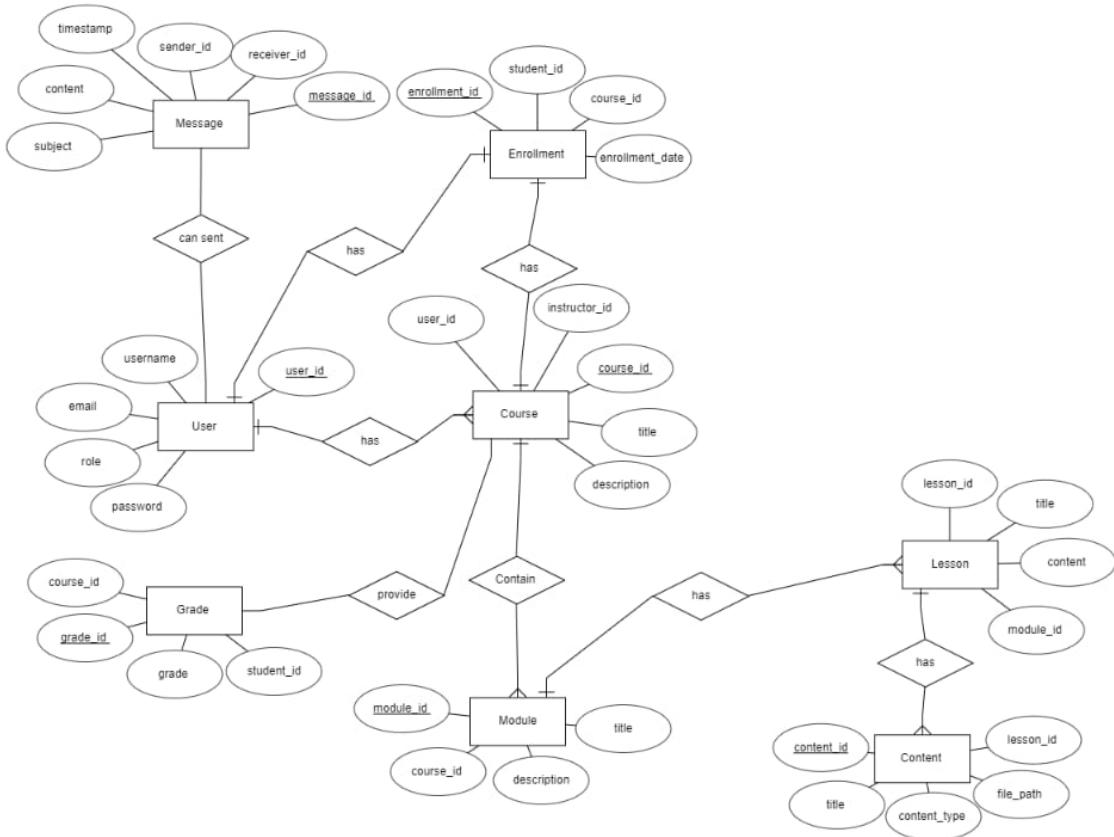
## **2.SOFTWARE REQUIREMENT**

Operating System: Windows 10 & Onwards

Front End: HTML, CSS, JavaScript, React JS, Tailwind CSS

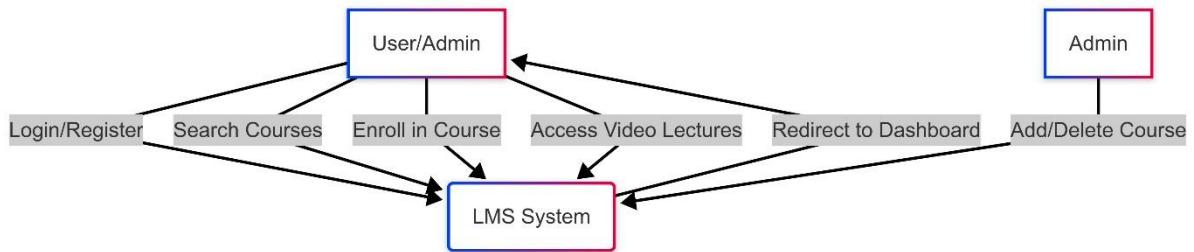
Backend: Node JS, Express JS, MongoDB

## Entity Relationship Diagram

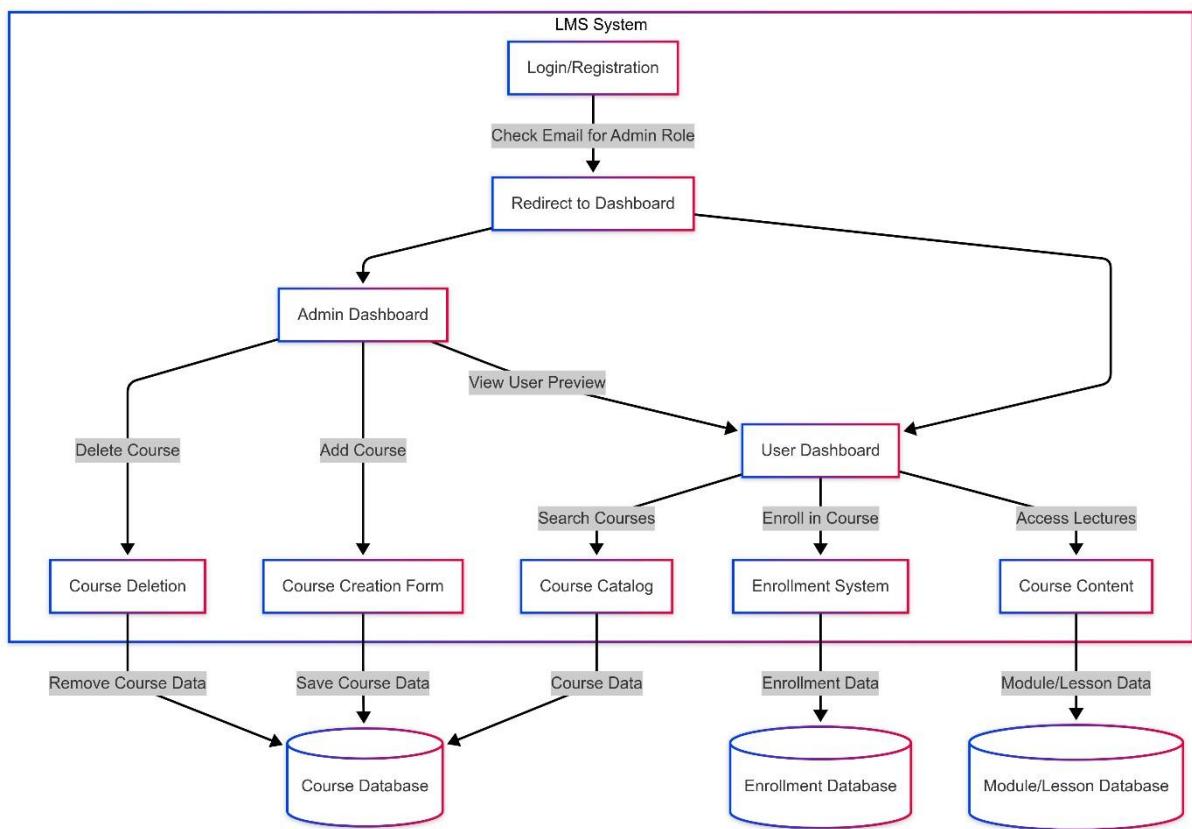


## Data Flow Context Diagram

### Level 0 DFD (Context Diagram):



### Level 1 DFD (Process Decomposition):



## MongoDB Schema for LMS

Here's the equivalent MongoDB schema design for your LMS project:

---

### 1. Admin Schema

Equivalent to the Admin Table in MySQL.

```
import mongoose from 'mongoose'
```

```
const adminSchema = new mongoose.Schema(
```

```
{
```

```
  name: {
```

```
    type: String,
```

```
    required: true,
```

```
    trim: true,
```

```
    maxlength: 50
```

```
},
```

```
  role: { type:
```

```
    String, default:
```

```
    "admin"
```

```
}
```

```
},
```

```
{
```

```
  timestamps: true
```

```
}
```

```
)
```

```
const Admin = mongoose.model('Admin', adminSchema)
```

```
export default Admin
```

---

## 2. Interview Schedule Schema

Equivalent to the Interview schedule table in MySQL.

```
const interviewSchema = new mongoose.Schema(
```

```
{
```

```
    date: {
```

```
        type: Date,
```

```
        required: true
```

```
    },
```

```
    time: {
```

```
        type: String,
```

```
        required: true
```

```
    },
```

```
    place: {
```

```
        type: String,
```

```
        required: true,
```

```
        trim: true
```

```
    }
```

```
},
```

```
{
```

```
    timestamps: true
```

```
}
```

```
)
```

```
const Interview = mongoose.model('Interview', interviewSchema) export
```

```
default Interview
```

---

### **3. Placed Student Schema**

**Equivalent to the Place student table in MySQL.**

```
const placedStudentSchema = new mongoose.Schema(  
  {  
    jobDate: {  
      type: Date,  
      required: true  
    },  
    location: {  
      type: String,  
      required: true,  
      trim: true  
    }  
  },  
  {  
    timestamps: true  
  }  
)
```

```
const PlacedStudent = mongoose.model('PlacedStudent', placedStudentSchema)
```

```
export default PlacedStudent
```

---

### **4. Student Schema**

**Equivalent to the Student table in MySQL.** const

```
studentSchema = new mongoose.Schema(  
  {
```

```
name: {  
    type: String,  
    required: true,  
    trim: true  
},  
  
studentId: {  
    type: Number,  
    required: true,  
    unique: true  
},  
  
gender: {  
    type: String,  
    enum: ["Male", "Female", "Other"]  
},  
  
percentage: {  
    type: Number  
},  
  
academicYear: {  
    type: Date,  
    required: true  
},  
  
contactNo: {  
    type: Number,  
    required: true  
}  
},  
{
```

```
    timestamps: true  
  }  
)  
  
const Student = mongoose.model('Student', studentSchema)  
export default Student
```

---

## 5. Company Schema

Equivalent to the Company table in MySQL.

```
const  
companySchema = new mongoose.Schema(  
{  
  name: {  
    type: String,  
    required: true,  
    trim: true  
  },  
  address: {  
    type: String,  
    required: true,  
    trim: true  
  },  
  criteria: {  
    type: String  
  }  
},  
{  
  timestamps: true
```

```
}
```

```
)
```

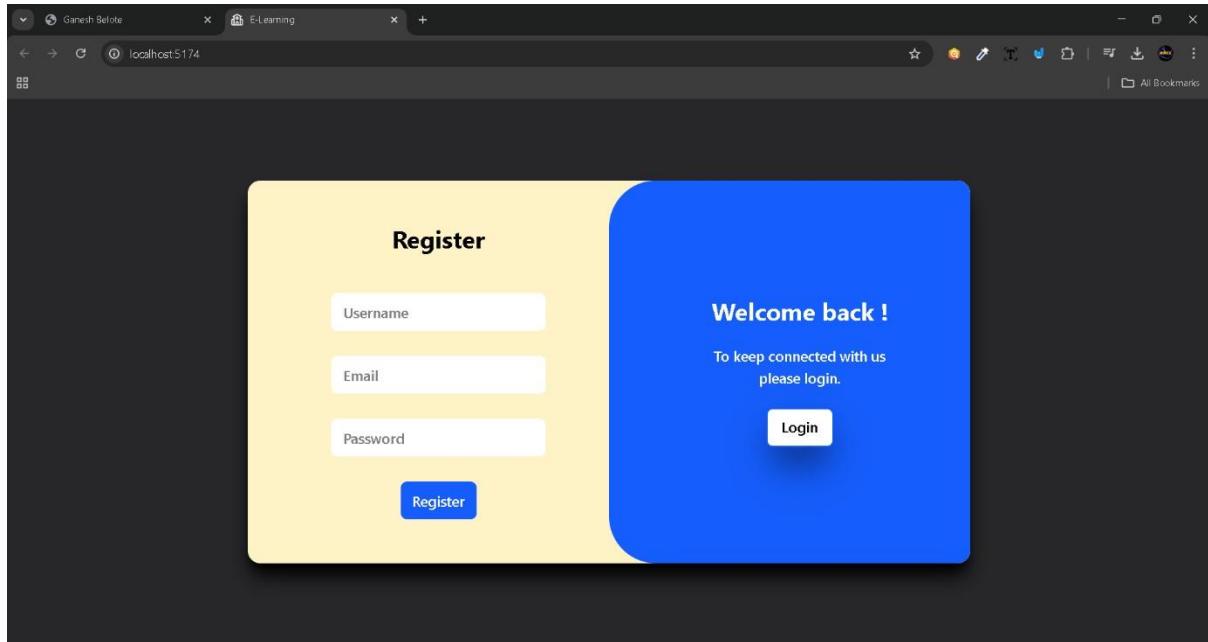
```
const Company = mongoose.model('Company', companySchema)
```

```
export default Company
```

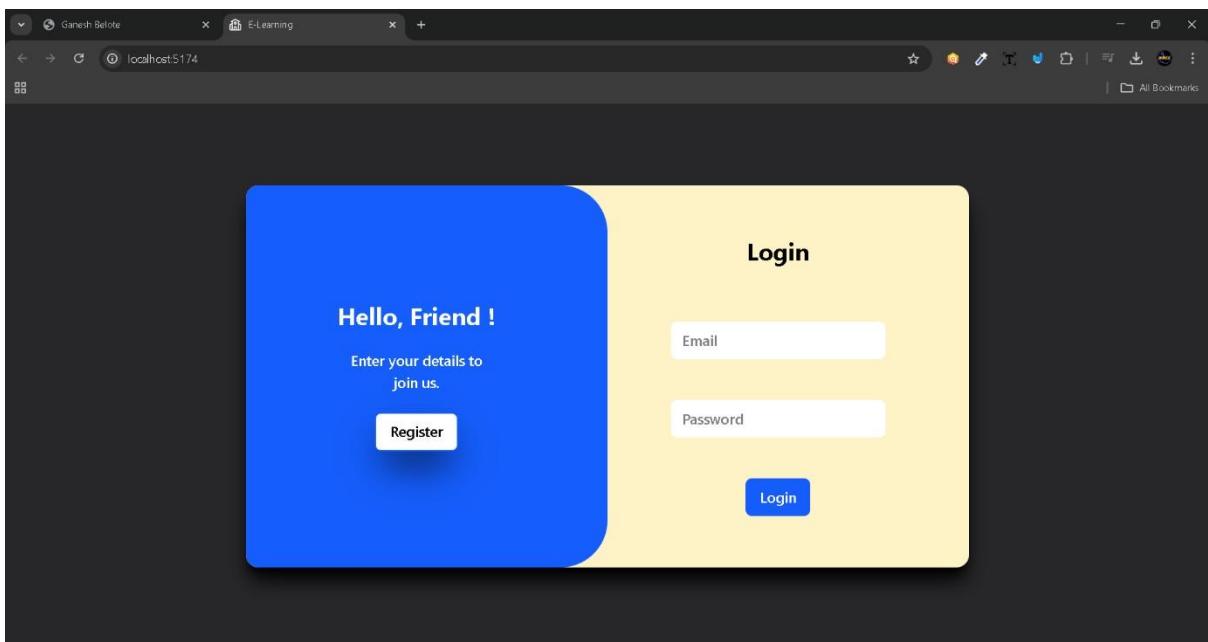
---

## Input Screens

### Registration Page:

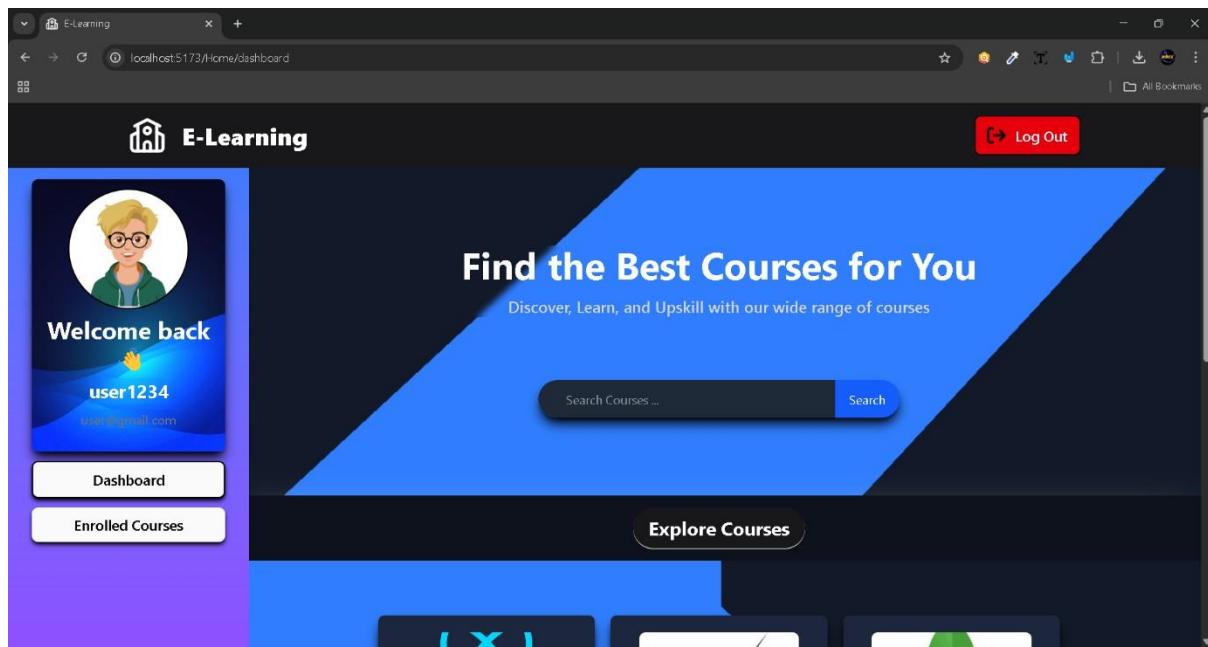


### Login Page:

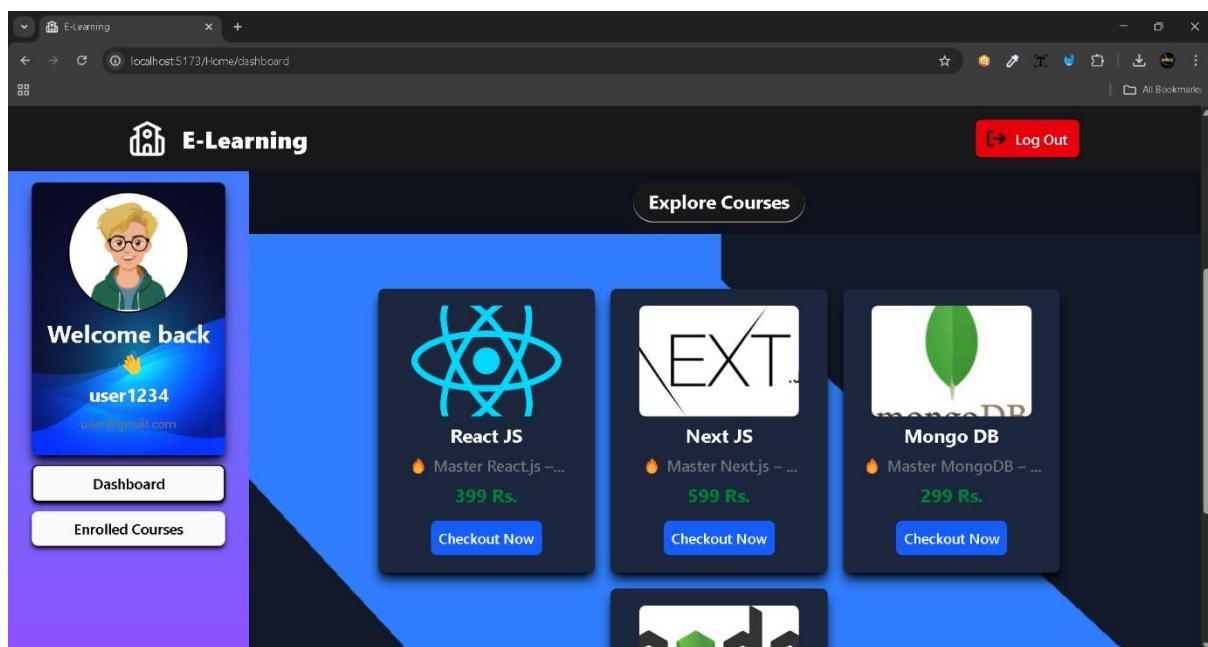


## User Dashboard:

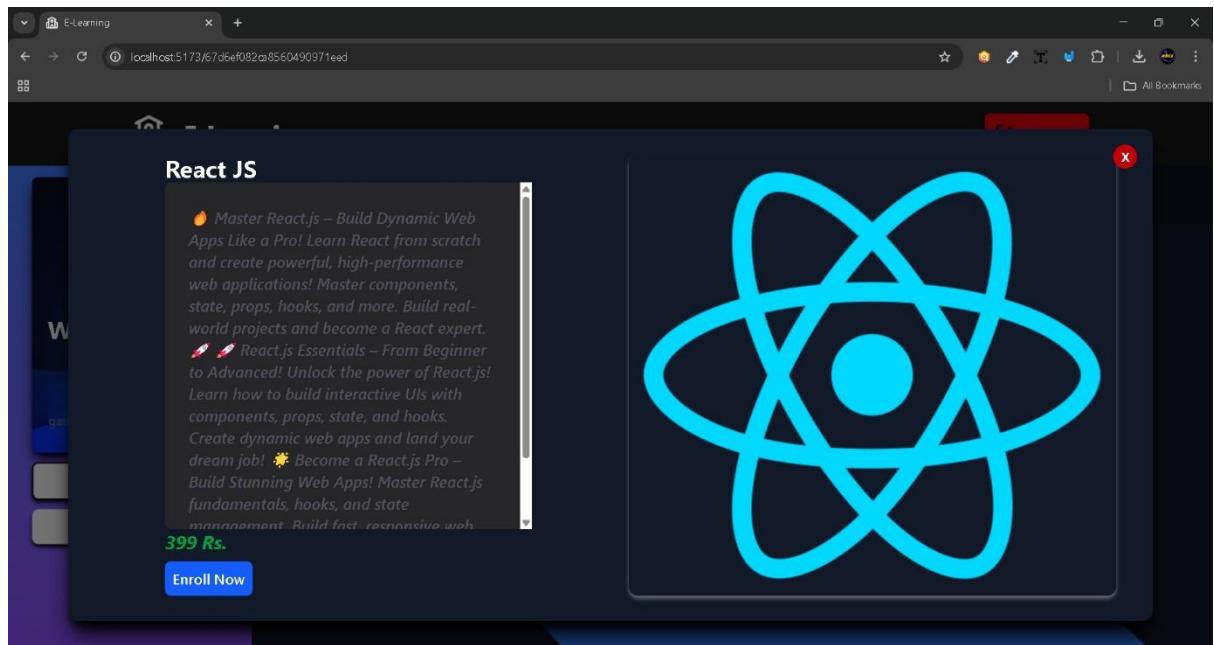
User Search Feature for course :



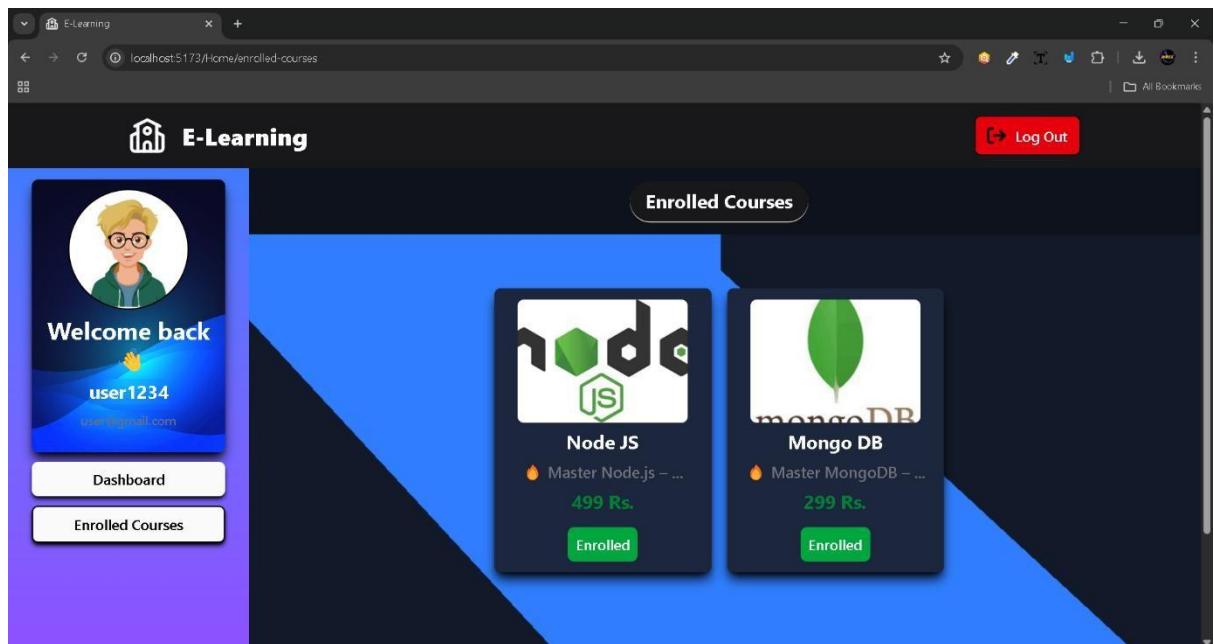
## Course List :



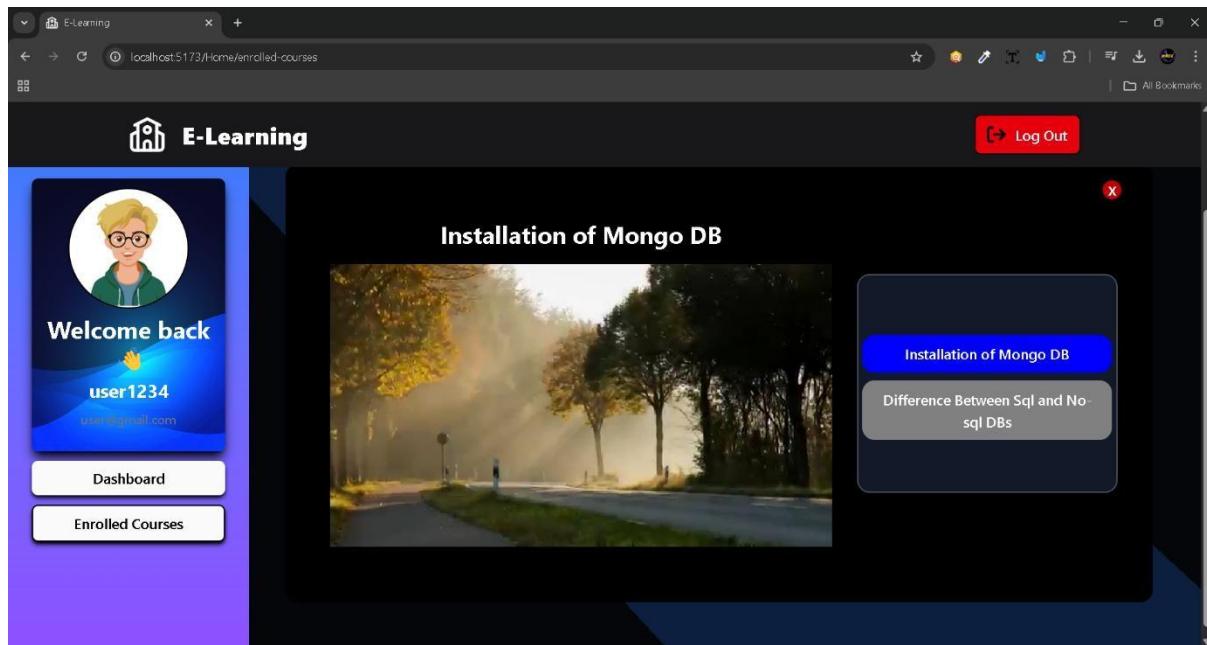
## Course Details Page:



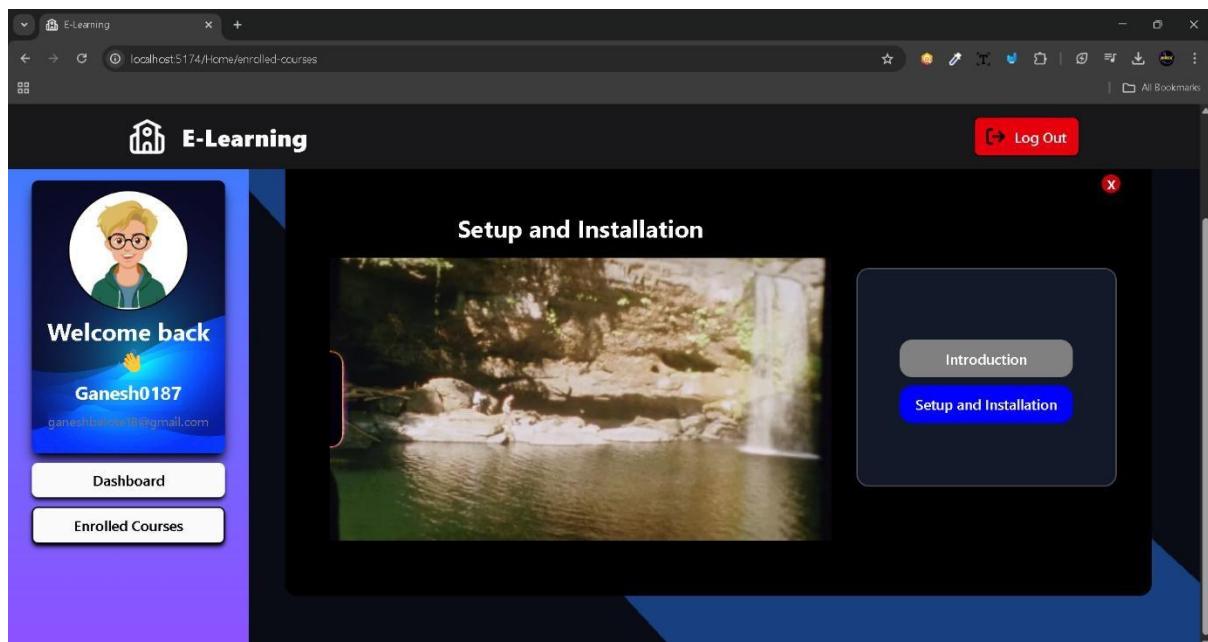
## Enrolled Course Section:



## Video Lecture Page:



## Switching Video Lecture :



## Admin Page :

Admin Course List page with delete option :

The screenshot shows the Admin Course List page. On the left, there's a sidebar with a user profile picture, the text "Welcome back admin", and email "admin@gmail.com". Below this are two buttons: "Dashboard" and "Add Courses". The main area displays four course cards:

- React JS**: Price 399 Rs. with a "Checkout Now" button.
- Next JS**: Price 599 Rs. with a "Checkout Now" button.
- Mongo DB**: Price 299 Rs. with a "Checkout Now" button.
- Node JS**: No price or "Checkout Now" button shown.

Each card has a red "X" icon in the top right corner, indicating a delete option.

## Course Adding Form Page :

The screenshot shows the Course Adding Form page. On the left, there's a sidebar with a user profile picture, the text "Welcome back admin", and email "admin@gmail.com". Below this are two buttons: "Dashboard" and "Add Courses". The main area is titled "Add Course" and contains the following form fields:

- Title ..
- Description..
- Price ..
- Thumbnail\**  
Choose File No file chosen
- Video T\**  
Choose File No file chosen
- Video Title

## **Reports**

The **Learning Management System (LMS)** consists of three core modules: **Admin**, **Student**, and **Instructor**. Each module is designed to handle specific functions and ensure a smooth flow of information within the system. The project incorporates a structured data handling method and an efficient algorithm to ensure the accuracy and reliability of reports.

### **1. Admin Module**

The **Admin Module** is the foundation of the project and was developed first as it plays a key role in managing the overall system. The admin is responsible for:

- Managing user accounts (students and instructors).
- Uploading course content and study materials.
- Handling system-level settings and permissions.
- Generating system-wide reports on user activity and course performance.

### **2. Student Module**

The **Student Module** was developed after the admin module, as it involves direct interaction with the system. The student module includes:

- **Student Registration** – Students can create accounts and log into the system.
- **Course Enrollment** – Students can browse and enroll in available courses.
- **Assignment Submission** – Students can upload assignments and view their results.
- **Performance Tracking** – Students can monitor their progress and receive feedback.

### **3. Instructor Module**

The **Instructor Module** was the final module developed. It focuses on providing tools for instructors to manage course delivery effectively. This module includes:

- Creating and managing courses.
- Uploading study materials and assessments.
- Evaluating student submissions and providing feedback.
- Generating reports on student performance and course engagement.

The structured development of these modules ensures that the LMS functions efficiently and meets the academic and administrative needs of the institution.

## **Advantages**

The **Learning Management System (LMS)** offers several advantages that improve the efficiency and effectiveness of managing the academic process and student performance.

### **Track campus placement process:**

The system allows for seamless tracking of the entire placement process, from student registration to final job offers.

### **Improve student placement and retention:**

By maintaining detailed records of student performance and company feedback, the system helps enhance placement rates and student retention. **Identify companies that match student's expectations:**

The system matches students' skills and qualifications with company requirements, increasing the chances of successful placement.

### **Record list of students placed with respective job designations in a particular academic year:**

The system stores and organizes placement records, making it easy to generate reports and analyze trends.

### **Highly Secure and Reliable:**

The system includes encryption and access control to protect sensitive data and ensure data integrity.

### **Easy-to-use:**

The user-friendly interface requires minimal training, making it easy for students, instructors, and admins to use.

### **Time-Saving:**

Automation reduces manual effort and paperwork, improving the overall efficiency of the system.

### **Online Portal:**

Students can apply for job opportunities and courses through the online portal, ensuring easy access from any location.

### **Cloud-based:**

All data is stored securely on the cloud, providing scalability and long-term data storage.

These advantages make the LMS a valuable tool for improving the academic and placement management process.

## **Limitations**

### **⊖ Input should be given carefully:**

The system relies on accurate data input. Any incorrect or incomplete information may affect the accuracy of reports and placement processes.

### **⊖ Limited for small job portal purposes:**

The system is designed for campus-level or small-scale placement activities and may not handle large-scale recruitment effectively.

### **⊖ Monthly reports of selected students can be stored individually:**

Reports of placed students are stored on a monthly basis, which may limit the ability to generate combined reports over longer periods.

### **⊖ Need for internet connectivity:**

The system requires an active internet connection for data retrieval, communication, and processing, which may limit access in areas with poor connectivity.

These limitations highlight areas for potential improvement in future updates of the system.

## **Future Enhancement**

### **1. Multi-language support:**

The software can be updated to support additional languages like **Marathi** and **Hindi**, improving accessibility and user experience for a wider audience.

### **2. Enhanced Security:**

Advanced security features such as **two-factor authentication (2FA)**, **SSL encryption**, and **role-based access control** can be added to strengthen data protection and prevent unauthorized access.

These enhancements will improve the system's usability, accessibility, and security, making it more robust and user-friendly.

## Bibliography

### Books

- *Learning Management Systems: Concepts and Applications* by O'Reilly Publication
- *Mastering LMS Platforms* by BPB Publication
- *LMS for Beginners* by SPD Publication

### Websites

- [www.youtube.com](http://www.youtube.com)
- [www.lmsworld.com](http://www.lmsworld.com)
- [www.w3schools.com](http://www.w3schools.com)