Anjuman-I-Islam's

M. H. SABOO SIDDIK COLLEGE OF ENGINEERING

8, Saboo Saddik Polytechnic Road, Byculla, Mumbai, Maharashtra 400008

DEPARTMENT OF INFORMATION TECHNOLOGY

REPORT

Title of the Project

Learning Management System

BY

PATWEKAR Sofiya-242465
Qureshi Abdurrehman-24266

University of Mumbai

Academic Year (2024 -25)

DECLARATION

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

We will be solely responsible if any kind of plagiarism is found.

ABSTRACT

This project presents a lightweight and user-friendly Learning Management System (LMS) designed to streamline digital education. The system facilitates seamless interaction between administrators, instructors, and students by offering a centralized platform for managing educational content, conducting assessments, and tracking learner progress.

Developed using Python and Flask, the LMS supports essential functionalities such as user authentication, role-based access, course creation, content uploading, and quiz management. Instructors can create and organize courses into structured modules, upload learning materials, and assess students through customizable quizzes. Students can enroll in courses, access study materials, complete assignments, and monitor their performance via an intuitive dashboard.

The project emphasizes simplicity, modular design, and effective data management using SQLite, making it scalable and adaptable for small to medium-sized educational institutions or training platforms. This LMS aims to enhance the e-learning experience through efficient course delivery, real-time evaluation, and easy access to educational resources.

INTROUCTION

In recent years, the rise of digital education has transformed the traditional learning environment, prompting the need for flexible, accessible, and interactive platforms. A Learning Management System (LMS) plays a pivotal role in this shift by offering a centralized solution for delivering educational content, managing coursework, and fostering communication between educators and learners.

This project introduces a simplified and efficient LMS developed using Python and Flask, aimed at bridging the gap between instructors and students through a user-friendly web application. The system is designed to support multiple roles—admin, instructor, and student—each with distinct capabilities tailored to their educational needs.

With features such as course management, lecture material uploads, quiz creation, and progress tracking, the LMS ensures a streamlined educational workflow. Instructors can effortlessly design courses and assessments, while students benefit from structured learning paths and real-time feedback. The system also incorporates user authentication and role-based access control, ensuring a secure and personalized learning environment.

The objective of this LMS is to provide a lightweight yet functional educational platform that can be deployed in schools, colleges, or corporate training environments, especially where resources are limited and simplicity is preferred. By combining core e-learning features with scalable architecture, the project demonstrates the practical implementation of web development and database integration concepts in real-world educational scenarios.

IMPLEMENTATION AND DESIGN

The Learning Management System (LMS) is developed using a lightweight yet efficient tech stack that ensures modularity, scalability, and ease of use. The system architecture is based on a Model-View-Controller (MVC) pattern, separating concerns and maintaining a clean codebase. The LMS provides distinct interfaces and functionalities for Admins, Instructors, and Students, ensuring a smooth educational workflow

Technology Stack

- **Frontend:**
    - **HTML5**: Structure of web pages
    - **Tailwind CSS**: Utility-first CSS framework for responsive and modern UI design
    - **JavaScript**: Client-side interactivity (e.g., form validations, dynamic elements)
- **Backend:**
    - **Python Flask**: Lightweight web framework for routing, authentication, and business logic
    - **SQLite3**: Built-in relational database for storing users, courses, quizzes, and progress data

System Modules

1. User Authentication & Role-Based Access

- Users can **register** and **log in** securely.
- Roles are assigned as **Admin**, **Instructor**, or **Student**.
- Role-based access control restricts access to pages and actions based on user roles.
- Passwords are hashed for secure storage.

2. Course Management (Instructor/Admin)

- Admins and Instructors can:
    - Create, edit, and delete courses
    - Add course details (title, description, category)
    - Upload related resources (PDFs, external links, etc.)
- Courses are displayed dynamically on the home page and dashboard using HTML templates and Flask routing.

3. Lesson and Content Upload

- Instructors can organize content into **modules/lessons**.
- Lessons support:
    - **Text** (rich content input)
    - **PDF file uploads**
    - **Embedded videos** (e.g., YouTube links)
- Stored metadata and files are managed via SQLite.

4. Quiz & Assessment Module

- Instructors can create **quizzes** with:

    - Multiple-choice questions
    - True/False and short answers
- Quizzes are rendered using HTML/JS and evaluated automatically (for objective questions).
- Student results are stored and displayed in the dashboard.

5. Student Dashboard

- Students can:
    - View and enroll in available courses
    - Track lesson completion status
    - Attempt quizzes and view results
- Progress tracking is handled through backend routes and stored in the database.

6. Admin Dashboard

- Admins have control over:
    - User management (view/delete users)
    - Course oversight
    - Monitoring system activity

Design Highlights

- **Tailwind CSS** ensures a clean, responsive, and mobile-friendly UI.
- Templates are rendered using **Jinja2**, Flask's built-in templating engine.

- Interactive elements like quiz validation and content toggling are handled using **JavaScript**.
- Flask blueprints (optional) help organize routes for users, courses, and admin panels.
- SQLite is used to keep the database portable and efficient for small-to-medium applications.

Conclusion

The Learning Management System (LMS) developed using Flask, SQLite3, and modern frontend tools like Tailwind CSS and JavaScript successfully delivers a streamlined platform for digital education. The system provides essential functionalities such as user authentication, course and content management, quiz handling, and role-based dashboards for Admins, Instructors, and Students.

With a clean, responsive design and intuitive interface, the LMS enables instructors to upload course material and create quizzes with ease, while allowing students to access content, track progress, and engage in assessments efficiently. The use of SQLite3 makes the application lightweight and easy to deploy, especially for small institutions or internal training environments.

This project demonstrates the effective integration of modern web technologies to solve real-world educational challenges, showcasing modularity, ease of use, and functional depth in a minimal setup.