



Elms

LEARNING MANAGEMENT SYSTEM

Thomas Mathew
22PMC157

ABSTRACT

The Learning Management System (LMS) is a Django project written in Python that aims to provide a comprehensive platform for managing and delivering online courses and educational resources. The project caters to the needs of both faculty and students, offering features and functionalities that facilitate efficient teaching and learning processes. To enable faculty to create and organize courses, deliver content, and assess student performance. To enhance communication and collaboration between faculty and students within the LMS platform.





REQUIREMENTS

STUDENT

- Registration
- Course Selection
- Read Study Materials
- Read Announcement
- Profile Management

FACULTY

- Authenticated
- Manage Course Materials
- Manage Course
- Manage Announcements
- Manage Profile

ADMIN

- Manage Students
- Manage Faculty
- Manage Course
- Manage Announcements
- Manage Course Materials

FEATURES AND HIGHLIGHTS

User Authentication and Roles:

- User registration and authentication for faculty and students.

Course Management By faculty:

- Creation and organization of courses by faculty.
- Delivery of course content in multiple formats, including text, multimedia, and interactive elements.

Course Access and Content Consumption By Student:

- Students can register, authenticate, and create their profiles on the LMS.
- Once enrolled, students can access course materials, including lectures, presentations, and supplementary resources.
- Content is presented in an organized and structured manner, facilitating easy navigation and learning.

TECHNICAL ASPECTS

Django Framework: Django is the primary framework used for building the LMS project. It provides a robust foundation for developing web applications, offering features such as URL routing, database connectivity, authentication, and templating.

Python: The LMS project is written in Python, a versatile and powerful programming language known for its simplicity and readability. Python is used to implement the backend logic, handle data processing, and perform various system-level operations.

HTML/CSS/JavaScript: The project utilizes front-end technologies such as HTML, CSS, and JavaScript to develop the user interface and enhance user interactions. These technologies are essential for creating responsive and visually appealing web pages.

Database Management System (DBMS): The project utilizes a DBMS to store and manage data related to students, courses, faculty, materials and announcement. DBMS in Django project include SQLite.

THIRD-PARTY LIBRARIES

Bootstrap: Bootstrap is a popular CSS framework used for responsive web design and UI components, making it easier to create visually appealing and mobile-friendly interfaces.

Pillow: A powerful library for image processing and manipulation

jQuery: jQuery is a JavaScript library used to simplify DOM manipulation and handle AJAX requests within the project's front-end components.

Django REST Framework: This library is used to build APIs for the LMS project, enabling seamless communication between the front-end and backend.

ARCHITECTURE

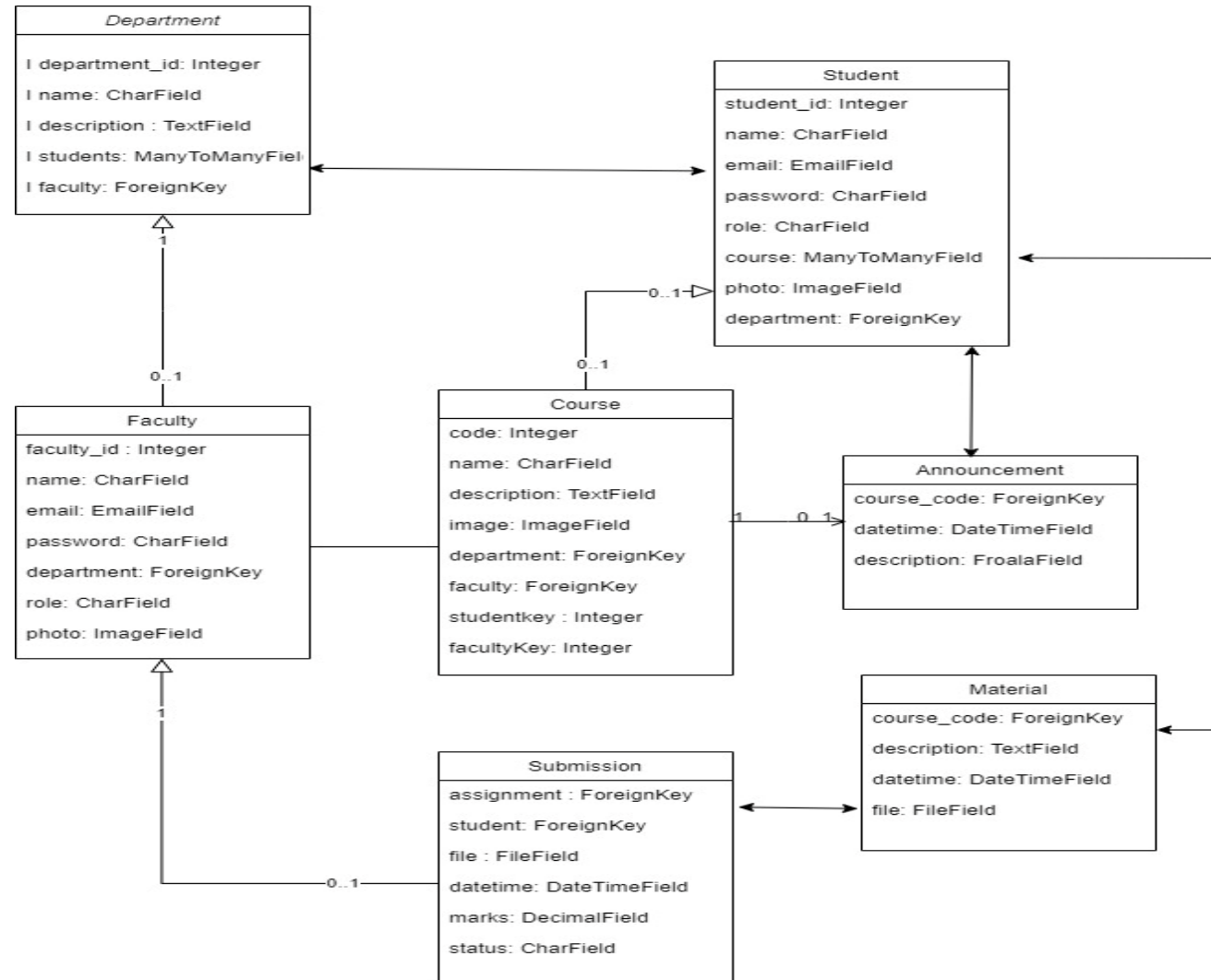
Models: The models define the data structures and relationships within the LMS, including entities such as students, courses, faculty, materials and Announcements.

View Models: view models handle the business logic data flow between the models and templates. They handle user requests, retrieve data from the database, and prepare the data for rendering in the templates.

Templates: Templates contain the HTML/CSS structure and presentation logic for rendering the user interface. They are responsible for displaying the data fetched from the views and rendering it in a user-friendly format.

Routers: Routers handle the URL routing and map the incoming requests to the appropriate views or view models. They define the endpoints and HTTP methods for handling user requests.

CLASS DIAGRAM






CHALLENGES FACED

User Interface Design and User Experience: Designing an intuitive and user-friendly interface that meets the needs of both faculty and students can be challenging.

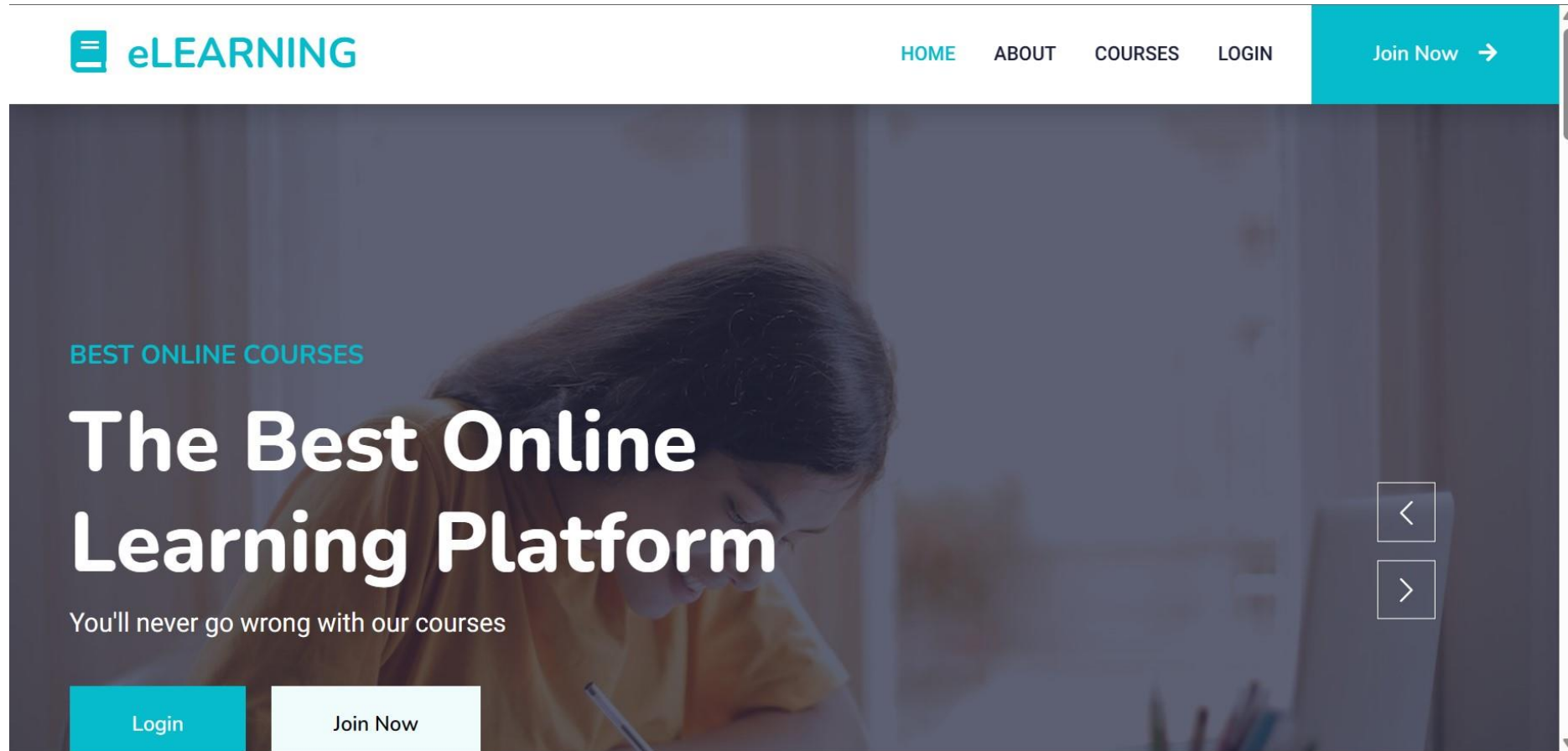
Testing and Quality Assurance: Ensuring the LMS is thoroughly tested across different devices, browsers, and user scenarios can be time-consuming.

Scalability and Performance: As the number of users and courses increases, ensuring the LMS can handle high user loads and maintain optimal performance becomes crucial.

Customization and Flexibility: Balancing the need for customization with maintaining a standardized user experience and codebase requires careful planning and design.



SCREENSHOTS



Home Page

SCREENSHOTS

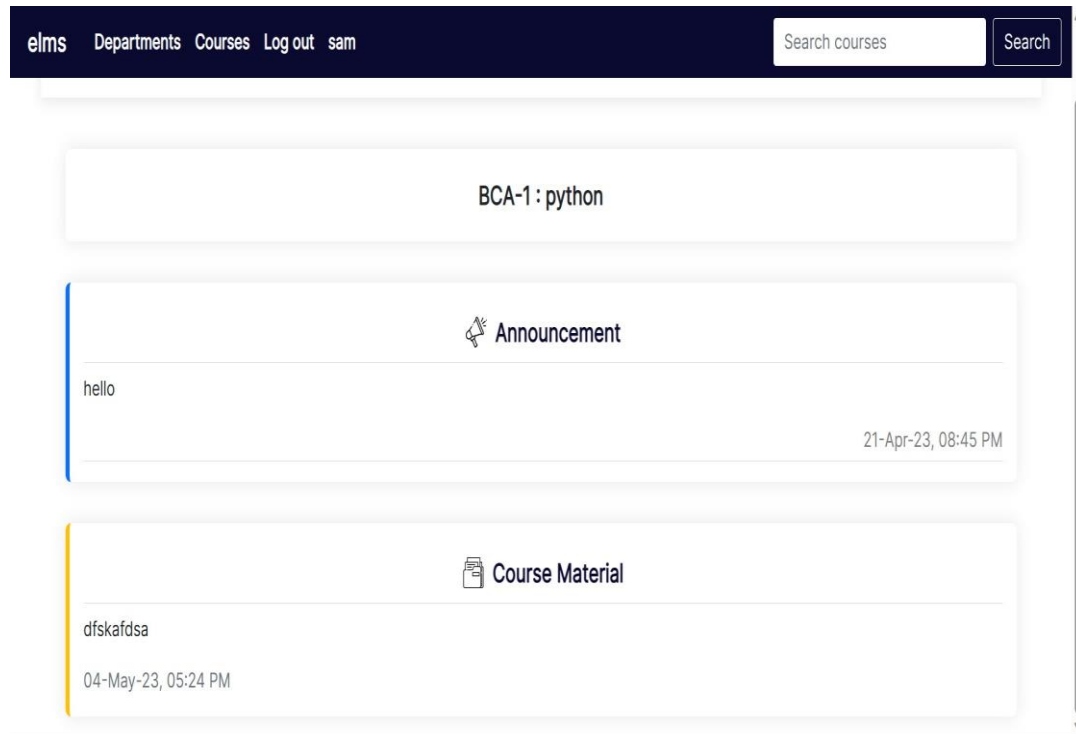
The screenshot shows the 'Student page' of a web application. At the top is a dark blue navigation bar with the text 'elms Departments Courses Log out sam' on the left and a search bar with 'Search courses' and a 'Search' button on the right. Below the navigation bar is a white box labeled 'My courses'. In the center, there is a card for 'Dept. of BCA' with the course 'BCA-1 : python', an 'Enter' button, and 'Course Teacher : Mc Thomas' at the bottom.

Student page

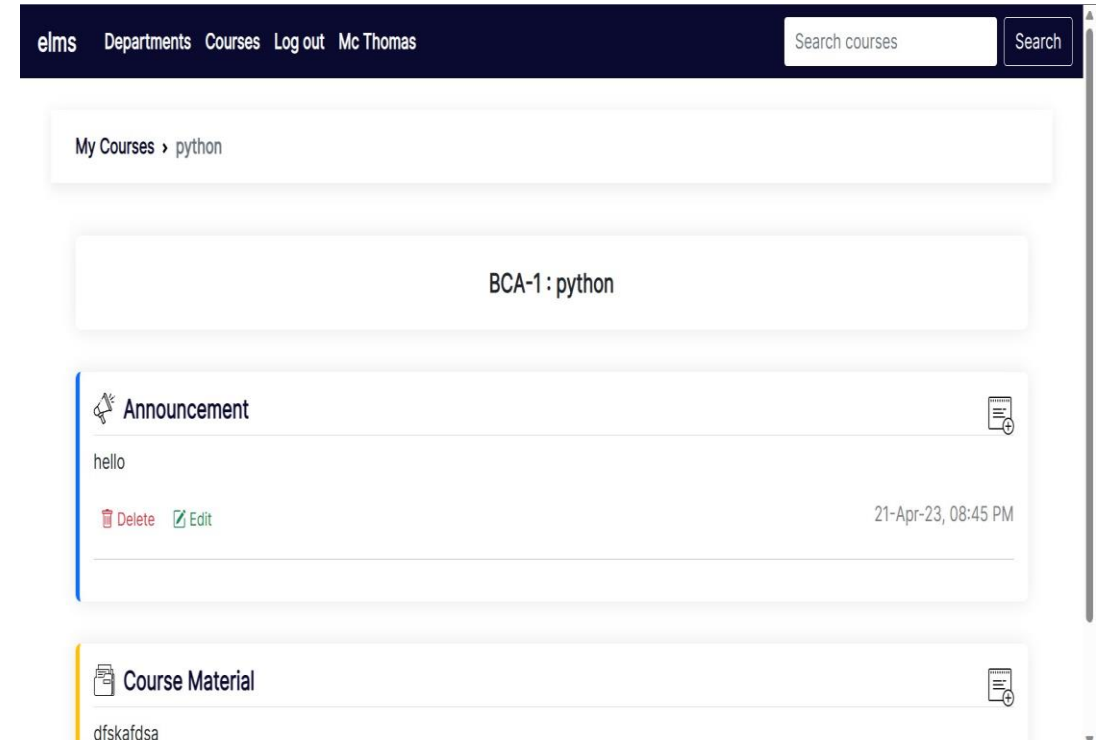
The screenshot shows the 'Faculty page' of the same web application. It has the same dark blue navigation bar at the top, but the user name is 'Mc Thomas'. Below the navigation bar is a white box labeled 'My Courses'. There are two course cards displayed side-by-side. The first card is for 'Dept. of BCA' with the course 'BCA-1 : python', an 'Enter' button, and a blank bottom section. The second card is for 'Dept. of BCA' with the course 'BCA-2 : Operations research', an 'Enter' button, and a blank bottom section.

Faculty page

SCREENSHOTS

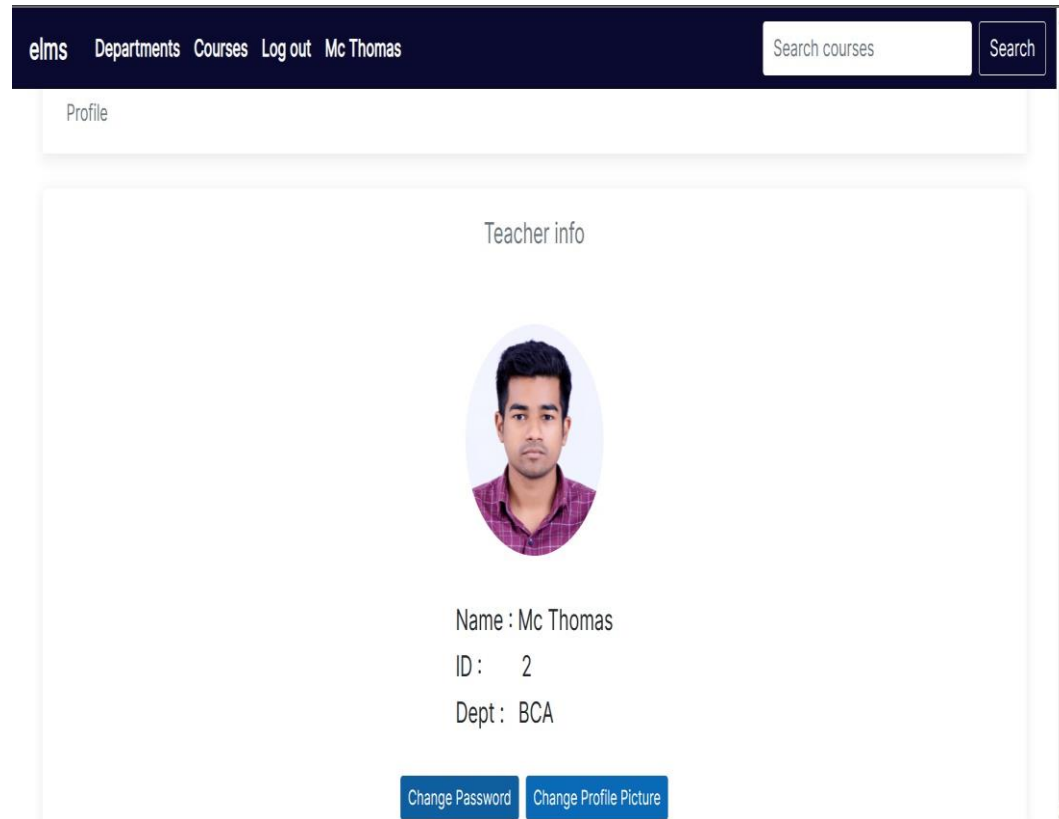


Student Enter page

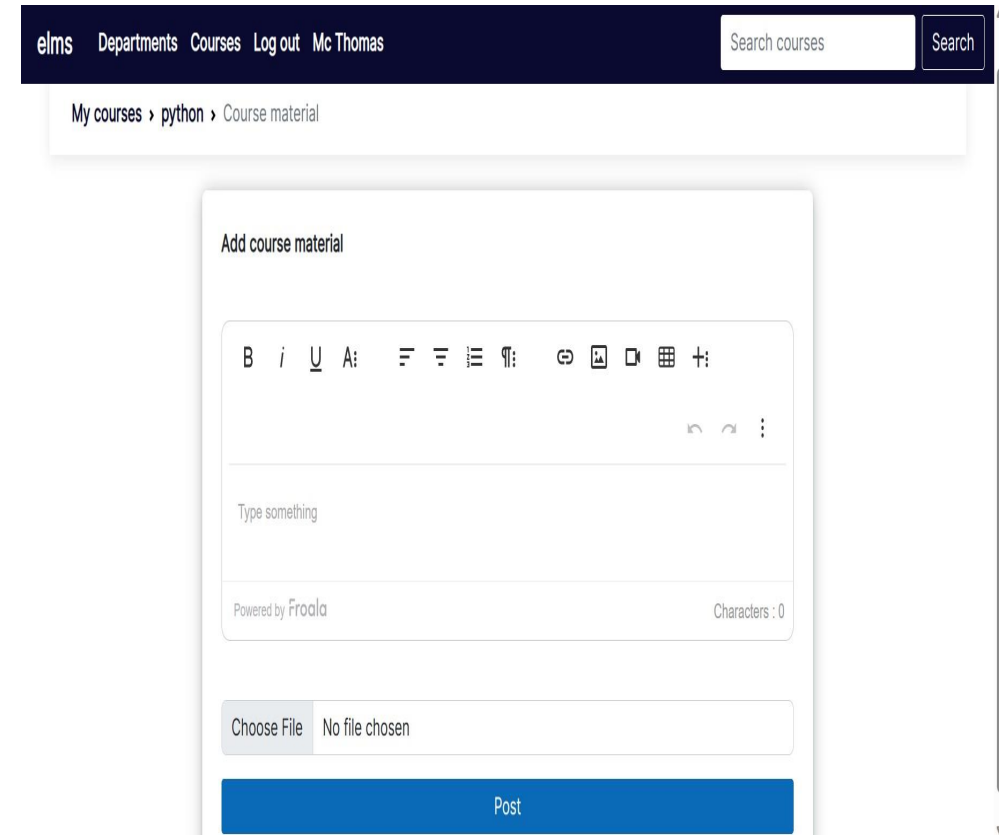


Faculty Enter page

SCREENSHOTS



Student Edit profile page



Faculty Materials uploading page



FUTURE ENHANCEMENTS

Advanced Quiz Features: Integrate a timer feature for quizzes to set time limits for each question or the entire quiz.

Attendance Monitoring Dashboard: Develop a dedicated dashboard for faculty and administrators to easily view and manage attendance records across multiple classes or sessions.



CONCLUSION

Learning Management System (LMS) project is a comprehensive solution designed to facilitate effective teaching and learning experiences for both faculty and students. The project leverages the Django framework in Python to create a robust and scalable web application, offering a comprehensive platform for teaching, learning, and administrative tasks. By providing a user-friendly interface, robust features, and continuous improvements, the LMS aims to enhance the educational experience, promote engagement, and support the growth and development of both faculty and students.



References

<https://youtu.be/OTmQOjsl0eg>

<https://youtu.be/XvU0QXqDQ1Y>

<https://itsourcecode.com/free-projects/python-projects/learning-management-system-project-in-django-with-source-code/>



THANK YOU