Online Course Management System

Project Overview

The Online Course Management System is a microservices-based web application developed using **Spring Boot** for the backend and **Angular** for the frontend. The main goal of this project is to provide a scalable and modular system where users can view and enroll in courses, while administrators can manage courses efficiently.

This project follows a **microservices architecture**, where each service is independently deployed and registered with **Eureka Service Registry**. All client requests are routed through an **API Gateway**, ensuring centralized access and service discovery.

Technology Stack

Backend

- Spring Boot
- Spring Web
- Spring Data JPA
- REST APIs

Microservices & Cloud Components

- Eureka Server (Service Registry)
- API Gateway

Frontend

- Angular

Database

- MySQL

Build Tool

- Maven

System Architecture

The system consists of the following components:

1. **Eureka Server** – Registers all microservices and enables service discovery.
2. **API Gateway** – Acts as a single entry point for all client requests and routes them to appropriate services.
3. **Course Service** – Manages course-related operations.
4. **User Service** – Handles user registration and user data.
5. **Enrollment Service** – Manages course enrollment and validates users and courses by communicating with other services.
6. **Angular Frontend** – Provides user interface for students and admin.

All backend services are registered with Eureka and communicate through the API Gateway.

Microservices Details

1. Course Service

Responsibilities: - Add new course - Update course - Delete course - View all courses - View course by ID

API Endpoints: - POST `/courses` - GET `/courses` - GET `/courses/{id}` - PUT `/courses/{id}` - DELETE `/courses/{id}`

---

2. User Service

Responsibilities: - Register new user - Get user details - View all users

API Endpoints: - POST `/users` - GET `/users/{id}` - GET `/users`

---

3. Enrollment Service

Responsibilities: - Enroll user into a course - View enrolled courses of a user

Business Rule: Before enrolling a user, the Enrollment Service validates user and course details by calling User Service and Course Service.

API Endpoints: - POST `/enroll` - GET `/enroll/user/{userId}`

API Gateway Routes

- `/api/courses/**` → Course Service
- `/api/users/**` → User Service
- `/api/enroll/**` → Enrollment Service

Frontend (Angular)

Pages Implemented

1. **Home Page** – Displays all available courses
2. **User Registration Page** – Allows new users to register
3. **Course Details Page** – Shows course details with enroll option
4. **My Courses Page** – Displays courses enrolled by the user
5. **Admin Panel** – Allows admin to add, update, and delete courses

---

Project Folder Structure

```
online-course-management-system
│
├── course-service
├── user-service
├── enrollment-service
├── api-gateway
├── eureka-server
├── angular-frontend
├── screenshots
└── README.md
```

Sample API Request & Response

Enroll User to Course

Request:

```
POST /api/enroll
{
  "userId": 1,
```

```
  "courseId": 101
}
```

Response:

```
{
  "enrollmentId": 10,
  "userId": 1,
  "courseId": 101,
  "enrollmentDate": "2026-02-08"
}
```

How to Run the Project

1. Start Eureka Server
2. Start API Gateway
3. Start Course Service, User Service, and Enrollment Service
4. Run Angular frontend using `ng serve`
5. Access the application via browser

Conclusion

This project demonstrates the implementation of microservices architecture using Spring Boot and Angular with proper service registration, routing, and inter-service communication. It provides a scalable and maintainable solution for online course management.