

Title: Development of an Educational Platform for Online Learning

Note: This assignment carries a weightage of 25% towards the final grade. It is a group project with 3 to 4 members.

You have been asked to develop an educational platform similar to Coursera or Udemy, tailored to offer a wide range of courses to learners. Below are the requirements provided by the client and/or the Business Analyst.

Requirements:

- **Web/Mobile Interface:** Develop a web/mobile interface where learners can browse, enroll in, and access courses. Ensure the interface is user-friendly and supports various devices.
- **Course Management Service:** Implement a service where course instructors can add, update, and delete course information. Instructors should be able to manage course content and enrollment details.
 - **Instructor role** - Add, update, and delete course content (lecture notes, videos, quizzes), monitor learner progress.
 - **Admin role** – Approve course content, integrate payment gateways, handle financial transactions related to course enrollments.
- **Learner Service:** Develop a service where learners can enroll in courses and track their progress. Allow learners to cancel course enrollment if needed.
- **Multiple Course Enrollment:** Learners should be able to enroll in multiple courses simultaneously without scheduling conflicts.
- **Payment Integration:** Integrate payment gateways to facilitate course enrollment payments. (Use external third-party services such as payhere, sandbox environment).

- **Confirmation and Notification:** Upon successful enrollment, learners should receive confirmation via SMS and email. Utilize third-party SMS and email services for sending notifications. (For email and SMS notifications, you may try to use an available service on the Internet.)

You may add new functionalities other than those mentioned in the description.

Implementation:

1. Based on the provided requirements, develop a set of RESTful web services to implement the educational platform. You may choose any technology stack to implement the services. Ensure the services are designed following REST principles, maintain scalability, security, and performance.
2. You must use the Microservices architecture to develop/integrate the API. Ensure that you use Docker and Kubernetes. If you are using any other tool for Microservices orchestration/integration, you may justify that in the report and during the viva.
3. Develop an asynchronous web client using any JavaScript framework that supports asynchronous programming (such as Angular, React, etc.) or use regular jQuery + AJAX. However, for the scope of this assignment, implementing just an asynchronous web client is sufficient.
4. Use appropriate security/authentication mechanisms to uniquely identify each user and authenticate them. There should be three roles: learner, course instructor, and system admin.

Deliverables:

1. A text file called 'submission.txt', containing a Github repository link containing all the source code. The source code should contain the source of the back-end project containing the API, client, and any other relevant source/resource files (e.g. database scripts), arranged in a proper directory structure.
2. The 'submission.txt' should contain a youtube video link of a presentation/demo of the project. Each member may use maximum 3 mins to explain their contribution, so that the total video length should be no more than 12 mins.
3. A 'readme.txt' document, listing down the steps to deploy the above deliverables.
4. A 'members.txt' file, containing the names, registration numbers and the IDs of the group members.

5. A report in pdf format. The report should include a high level architectural diagram showing the services and their interconnectivity. Also, it should list the interfaces (NOT the user interfaces, but the service interfaces) exposed by each service and should briefly explain each of the workflows used in the system (you may use design diagrams of your choice to do this). You can also include the details about the authentication/security mechanisms adopted. **Mention the individual contribution of each member in the report.**

You may use code snippets in the report to explain the above.

The report must have an appendix with all the code that you have written (**excluding the auto-generated code**). **Do not paste screenshots of the code in the appendix and copy the code as text. If screenshots are added in the appendix, only the minimum mark may be offered.**

Note: All reports will be uploaded to Turnitin for plagiarism checking. If the turnitin similarity is above **20%**, **marks will be penalized.**

Submission: All files should be uploaded in a single zip archive. The zip file name should be 'Group ID_DS-Assignment'. Only one member needs to upload the submission.

Submission Deadline: 13th week of the semester

Viva Session will be held within 14th week of the semester.

Marking Rubric

Criteria	Allocated Mark	Student Mark
Application of Microservices principles in the architecture and the design	20	
Having clearly defined interfaces, that facilitate reusability	10	
Use of proper payment integration mechanism and SMS, Email Notification	10	
Quality and the readability of the code, with meaningful and detailed comments.	10	
Integration/Orchestration of services using Docker + Kubernetes	20	
Adoption of appropriate authentication/security mechanisms	10	
Comprehensiveness and the quality of the report	20	