

Assignment

- Course Title: **Distributed Systems and Applications**
- Course Code: **DSA621S**
- Assessment: Third Assignment
- Released on: 26/10/2021
- Due Date: 11/11/2021

Problem

The [Faculty of Computing and Informatics \(FCI\)](#) at [NUST](#) embarked upon a software development project to manage the **course outline** automation. A course outline summarises the essential information about a course being taught during a semester (or a year). It includes course and lecturer information, learning outcomes and course content (the key topics being covered), course and assessment schedule. For further conceptual guidance, you may refer to the course outline handed to you at the beginning of the semester for this course. The final step in generating of a course outline involves a (digital) signature by the lecturer in charge, approval by the Head of Department (HoD) and an acknowledgement by each student taking the course.

All the information used during the generation of a course outline should be input into the system and stored on a persistent storage (e.g., [distributed] file system or data store). In addition, any operation in the system requires a successful authentication of the user with the appropriate authorisation. The authorisation model is defined as follows:

1. There are three user profiles: **HoD**, **lecturer** and **student**
2. A student can:
 - view a course outline generated for a course he/she is taking;
 - acknowledge reception of a generated course outline for a course he/she is taking.
3. A lecturer can:
 - input information about a course he/she has been assigned;
 - view course outlines he/she has generated;
 - generate and sign a course outline for a course he/she has been assigned.
4. An HoD can:
 - input information about a course, a lecturer and a student;
 - view all generated course outlines;
 - (digitally) approve a course outline.

You are required to adopt a microservice architectural style to design the distributed system. Furthermore, you will use a Kafka instance to support the communication between a client and your service(s).

Your task is to:

1. design this application following a micro-service architectural style;
2. Set up a Kafka cluster or instance for the communication;
3. Implement the service(s);
4. deploy the services using Kubernetes orchestrator.

Evaluation criteria

The following criteria will be followed to assess each submission

- Problem decomposition into services (15%).
- Setup of the Kafka cluster, including topic management (15%).
- Implementation of the services and their deployment with Kubernetes (55%).
- The business logic inside each service, as well as the authentication and authorisation. (15%)

Submission Instructions

- This assignment is to be completed by groups of *at most* two students each.
- For each group, a repository should be created on [Gitlab](#). The repository URL should be communicated by Wednesday, November 3 2021, with all group members set up as contributors. Please indicate the study mode and strand for each group member.
- The submission date is Thursday, November 11 2021, at midnight. Please note that *commits* after that deadline will not be accepted. Therefore, a submission will be assessed based on the clone of the repository at the deadline. For submissions with commits after the deadline, there will be a deduction of 10% per week.
- Any group that fails to submit on time will be awarded the mark 0.
- Although this is a group project, each member will receive a mark that reflects his/her contribution to the project. Each group member will be requested to highlight his/her contribution during the presentation of the project. More particularly, if a student's username does not appear in the commit log of the group repository, that student will be assumed not to have contributed to the project and thus be awarded the mark 0.
- Each group is expected to present its project after the submission deadline.
- There should be no assumption about the execution environment of your code. It could be run using a specific framework or on the command line.
- In the case of plagiarism (groups copying from each other or submissions copied from the Internet), all submissions involved will be awarded the mark 0, and each student will receive a warning.