Project specification

Functional specification

These are guidelines for the course project:

Backend (applies to everybody)

1. **DB/persistence layer**:
   - usage of Postgres database (if you want to use a different DB please discuss it with us first)
   - use at least 4 tables, 2 of these should use foreign keys
   - the project should use some kind of DB versioning library (Flyway will be discussed during the course but you are free to choose something else)
2. **API**:
   - the app should expose at least 7-8 endpoints
   - one of the endpoints should expose a non JSON resource (pdf, xml, jpg etc)
   - there should be at least one domain object for which all CRUD operations are supported via endpoints (GET, DELETE, POST, (PUT))
   - there should be at least one endpoint which supports pagination logic (takes limit and sort parameters)
3. **Security and user management:**
   - endpoints should be protected - a minimal requirement is basic authentication but you are free to use OAuth
   - define at least 2 roles (eg admin/normal user)
   - add role permission to your endpoints
4. **Documentation**
   - add documentation to your project, all documentation should be available within your github project
   - README file
   - OpenApi/Swagger documentation of your API (can be autogenerated by Spring)
   - UML diagram about the domain entity structure
   - optional extra documentation:
     - Postman collection to test the endpoints
     - sequence diagrams
     - or anything else you find useful

Frontend requirements (if you are taking the Angular course in parallel)

- the app should have a menu which allows users to navigate
- the app should have at least 4-5 view/”page”, with 5-6 components
- the app should have 2-3 modules
- the app should use 2-3 services
- http communication
  - at least 2-3 GET
  - at least 1-2 POST
- the app should have at least 1 form
- usage of design elements is optional (css, scss)

Backend requirements for extra points (max 10 points)

5. **Microservice architecture**
    - if you are only taking the backend class:
        - separate your application logic into 2 microservices
        - there should be at least 2 endpoints exposed by the second service
        - there should be at least 2 method calls from one of the services to the other
        - only one of the microservices needs to be protected by security
        - they can use the same database

Non functional specification for backend

- the project has to be handed in through github
- the project should be runnable in the environment used during the course (please discuss any changes to the environment with us first)
- the project should store the data in a persistent database
- project should use maven or gradle
- there should be a README describing how to run the project - we should be able to run the project locally based on the README with little effort
- any initial data needed to run the application should be inserted into the DB by the DB versioning library

Project presentation

You can either present your project on your laptop or on one of the lab PCs. You should prepare a short demo (5 min) where you go through your project and explain how it works. We will ask you a few follow up questions about your implementation and your code.

It is important that your project should be runnable: we advise you to reduce your scope if you feel that you are out of time.

Project demo: **December 9th, 2022**

GRADING

5 points - specification

25 points - written test

70 points - course project

Grades

90 - 100 points -> 5

75-90 points -> 4

65-75 points -> 3

55-65 -> 2

MINIMAL PASS REQUIREMENT: at least 40% on both written test (10 points) and project (28points)

DEADLINES

Specification

A 1-2 page specification for your project is also a requirement for the course. The specification should include a short description of your project, what functionalities you intend to implement and the design of your data domains (database tables and corresponding entities) you will be using.

Specification should be in pdf format.

You cannot pass the course if you do not hand in the specification. Please send your specification before the deadline via Teams task.

You receive max **5 points** for the specification.

DEADLINE: November 14th, 2022, 12 pm

Project

Projects should be submitted by sending the instructors the github link via Teams.

You receive max **70 points** for the project implementation.

DEADLINE: December 7th, 2022 12pm