Motivation

Your team should create a REST API for the "Clean Backend Competition". There are some requirements described in the following sectio technologies will be Java Spring Boot, Docker, MySQL and MinIO. Create an MVP (Minimum Viable Product) to demonstrate your skills.

Technical Specifications/Requirements

Fork this repository https://github.com/marvkos/bweng-spring-rest-backend to start your project.

Your GitHub repository has to be **public** for the submission.

The frontend is provided by the Frontend Web Engineering course and must be adhered to.

The API must be implemented according to the REST architecture. For the API operations, use the **correct** HTTP methods (GET, POST, PU and codes (200, 201, 404, ...) provided for this purpose . The data should be delivered and processed in JSON format (except files). All dat be **validated**.

All data should be stored in a **database** (or in an object storage).

Authorization must be accomplished using JWT tokens. Be sure to control user roles when accessing data.

The Spring Boot framework is to be used as a future-proof technology.

?

At least 80% code coverage must be achieved during testing.

Functional Requirements

- Data should be displayed for anonym (not logged-in) users
  - A homepage with latest news or blog entries or products
- You can create an account
  - Anonym users cannot add data to the system
  - You must have an account to create project logs or comments or blog entries
- Required user data
  - Email (valid email)
  - Username (min 5 chars)
  - Password (min 8 chars, numbers, lower- and uppercase chars)
  - Country (valid country code)
  - Profile picture (placeholder, if not provided)
  - **also Frontend Development requirements**
- There are users and admins (different features, according to application)
  - Admins have an overview of all users
  - Admins have an overview of all data (products, tweets, items, ...)
  - Admins can delete or edit everything (data, users, ...)
- Your data should be extended with files (images, documents, ...)
  - Blog entries have images or project meeting notes have pdfs or products have images
  - Make sure the user can only upload the correct file type
- User are related to the data
  - When a user creates data (blog entry or comment or ticket) only this user (and admins) can edit the data
- Data should be sortable or filterable
  - Every data entry has "created at" and "updated at" info (https://www.baeldung.com/hibernate-creationtimestamp-updatetim

Suggestions

- A micro blogging tool (Twitter, Instagram, ...)
- An online shop (clothing, tickets, hardware, ...)
- A company page with project management
- A lost and found service
- Education management service (CIS, ...)

Points

Topic	**Points**
Use REST principles **(routes, JSON, ids, ...)**	20
Use a SQL Database and an ORM	10
Clean CRUD controllers **(only call services, handle Exceptions with HTTP codes)**	15
Validate data **(use Spring Boot data validation, e.g. emails should be emails, image upload should only allow images, ...)**	5
Access security **(anonym, user, admin, JWT, ...)**	10

?

Permission security **(user specific access)**	5
Correct file handling	5
Functional requirements **(see above)**	20
Testing	10

Last modified: Wednesday, 2 April 2025, 11:52 AM

Class 7: 1st Milestone

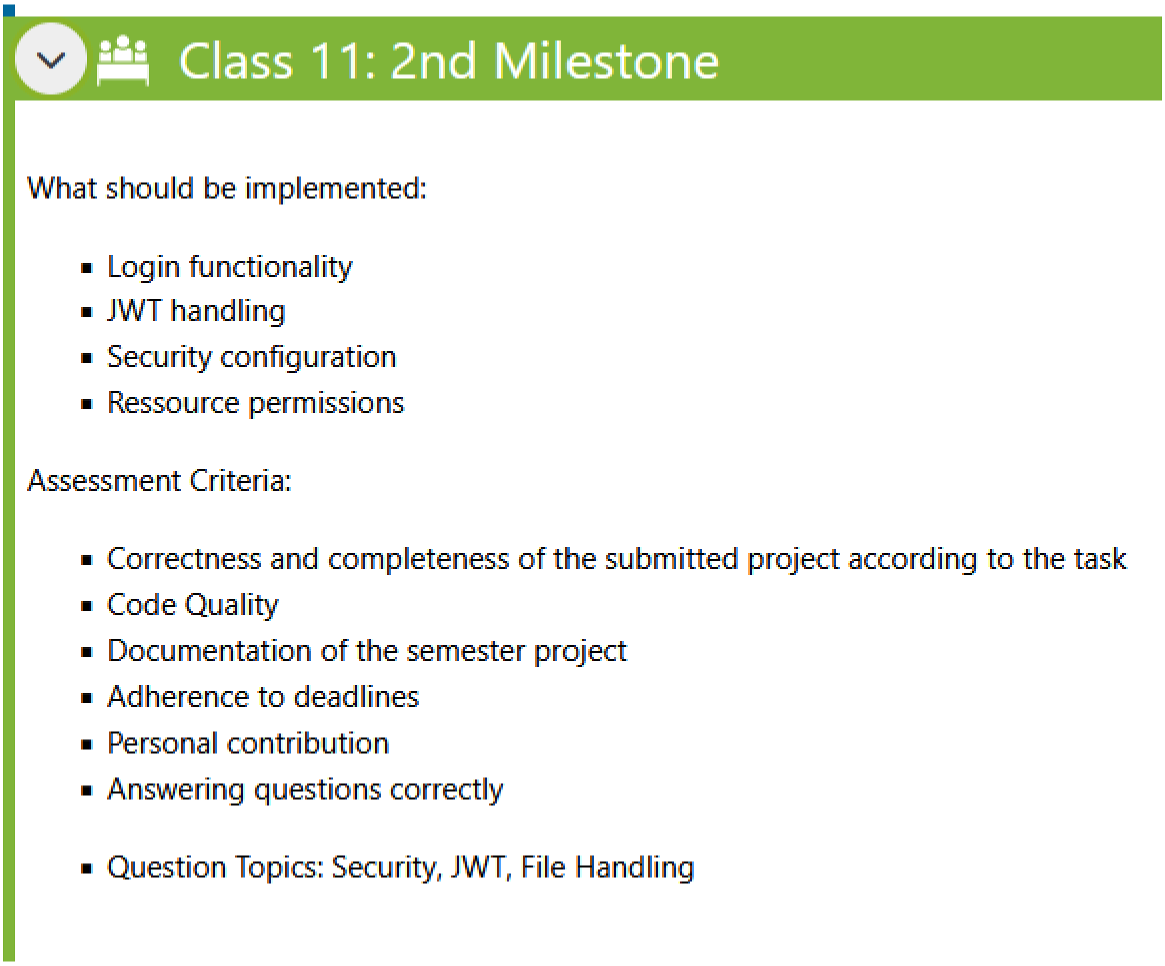
What should be implemented:

- REST Controller with correct paths and methods for your application
- Models with validation
- Database configuration
- Repositories

Assessment Criteria:

- Application must fit to the project description and scope (a simple to-do list is not enough)
- Correctness and completeness of the submitted project according to the task
- Code Quality
- Documentation of the semester project
- Adherence to deadlines
- Personal contribution
- Answering questions correctly

- Question Topics: Spring Boot, REST, ORM, HTTP, MVC, CRUD



Class 11: 2nd Milestone

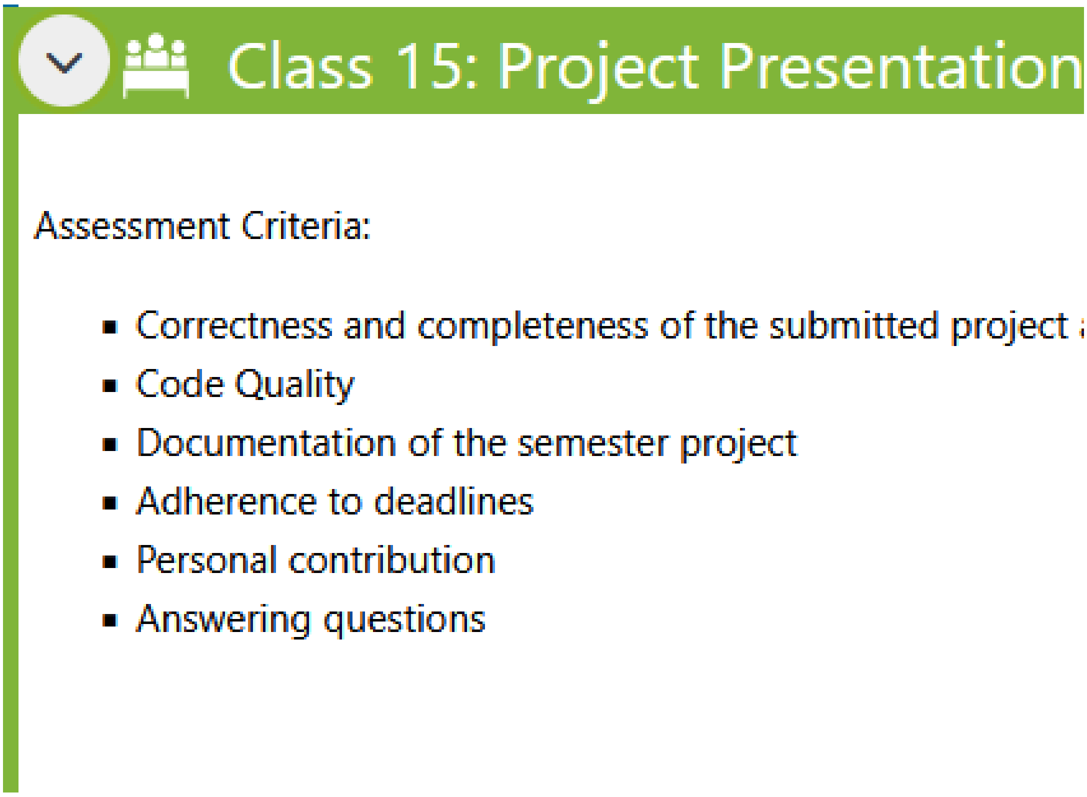
What should be implemented:

- Login functionality
- JWT handling
- Security configuration
- Ressource permissions

Assessment Criteria:

- Correctness and completeness of the submitted project according to the task
- Code Quality
- Documentation of the semester project
- Adherence to deadlines
- Personal contribution
- Answering questions correctly

- Question Topics: Security, JWT, File Handling



Class 15: Project Presentation

Assessment Criteria:

- Correctness and completeness of the submitted project
- Code Quality
- Documentation of the semester project
- Adherence to deadlines
- Personal contribution
- Answering questions