thws
Technische Hochschule
Würzburg-Schweinfurt

Backend Systems
Portfolio 01
Prof. Dr. Peter Braun

1 Introduction

As part of the portfolio exam this semester, you will work on a software development project as a group. This document describes the requirements of the **first** portfolio assignment, which is about finding a topic and collecting approval from the lecturer. The deadline for submitting your document is October 31, 2025 (see Moodle).

2 Outline of the Project

The semester project for this course will involve developing a backend application that provides an API using the REST paradigm. The project's primary goal is to implement a functional backend system with a well-defined API, adhering to the principles of a hexagonal architecture and utilizing the Quarkus framework. The project will also require the use of Java Persistence API (JPA) to handle persistence, meaning that you must design and implement entities that map to database tables. The database interaction should adhere to standard JPA practices.

The domain of the application is flexible and can be chosen by the student group. Still, it must include at least one one-to-many relationship between entities to demonstrate an understanding of this key database concept. For instance, a typical example could involve customers and orders or authors and books, where one entity is related to many instances of another entity. Furthermore, the API should expose typical CRUD operations (Create, Read, Update, Delete) for managing these entities.

In addition to the core functionality, students must ensure that their codebase includes comprehensive test cases. Unit tests should verify individual components of the system, particularly the business logic, to ensure its integrity. In contrast, integration tests should ensure the correct functioning of the entire application, including database operations and API responses. This focus on testing will ensure that the system is reliable, maintainable, and can evolve. The project will be developed in groups, with regular checkpoints to review progress and adherence to good development practices.

Additional technical requirements will be outlined in a separate document.

3 Pick your Topic

1. In the first step, please form a group of at least two and at most four students.

2. Discuss the topic (or domain) of your project. Use all kinds of (AI) assistants to create project ideas. Your project must neither be trivial (e.g., involving only two entities and standard CRUD operations) nor overly complex. I always prefer good quality over many features.

3. **Please take this point seriously: avoid creating a too complex app.** Time is limited, and the implementation is only one part of the overall portfolio. Last semester, some projects were overly complex in terms of features and lines of code, and students complained about the time and effort required to complete them. I don't feel responsible if you invest more than you need to. In order to assist you in finding the balance between complexity and effort, I request that you submit a short description as part of this first assignment.

4. If you are studying in the *Informatik dual* programme, please try to find a topic with a relation to your company.

5. Please prepare a short paper (two to three pages) about your project idea and submit this on October 31st, 2025, in Moodle. Only one submission is required from one of the team members.

6. Create a PDF version of the paper and name the file following this pattern: `BS_Portfolio_01_{TOPIC}.pdf`. Replace {`TOPIC`} with **one word** that describes your project topic best. Here is an example: `BS_Portfolio_01_LibraryService.pdf`.

4 Structure of your Document

The following box proposes a structure for your document:

Your document must have a title, and all group members must be listed as authors. Please include your immatriculation number after your name.

1. Project Overview

Provide a one-sentence elevator pitch that clearly states the problem, identifies the target users, and conveys the core value. Summarize in more sentences why the problem matters in this context and outline the essential capabilities you plan to deliver now, while explicitly naming one or two non-goals to keep the scope focused.

2. Domain Model

Include a UML class diagram that shows key entities with identifiers and attributes, associations with multiplicities, and any important invariants or validation rules. Briefly explain the most important entities and relationships in prose, and, if helpful, add one or two tiny example records to illustrate how real data would look.

3. Use Cases

Describe four to six core use cases; for each, provide an ID and title, the primary actor, the goal in user-story form, preconditions, the main success scenario as a short sequence, and the key alternate or failure flows. Conclude each use case with postconditions and concise, testable acceptance criteria, and assign a priority (Must/Should/Could) to make the scope explicit.

5 Deliverables

Please upload the document by the given deadline published in Moodle. You will get feedback from the lecturer about the level of complexity, either orally or in writing.

This document will then serve as the basis for evaluating your implementation against the requirements.