



Development of a Food Ordering Web Application with Spring and React Technologies

Project Laboratory - Schedule
Spring Semester 2024/2025

Mia Princz

3rd year, Computer Engineering student
BSc Software Engineering specialization

Consultant:

Gábor Imre, Assistant lecturer
(Department of Automation and Applied Informatics)

Contents

1	Introduction	1
2	Purpose of the Schedule	1
3	Development Schedule	1
	Week 1 - Project Initialization and Repository Creation	1
	Week 2 - User Management	2
	Week 3 - Restaurant Management	2
	Week 4 - Product and Cart Management	3
	Week 5 - Order and Payment Management	3
	Week 6 - Reviews and Coupons Management	4
	Week 7 - Admin Panel and Restaurant Management	4
	Week 8 - UI/UX Development and Refinement	5
	Week 9 - Responsive Design	5
	Week 10 - Containerization and DevOps Integration	5
	Week 11 - Testing and Bug Fixing I.	6
	Week 12 - Testing and Bug Fixing II.	6
	Week 13 - Developer Documentation and Project Presentation I.	6
	Week 14 - Developer Documentation and Project Presentation II.	7
	Week of Repeats - Project Demo	7

1 Introduction

This document outlines the development schedule for the project titled Development of a Food Ordering Web Application with Spring and React Technologies, chosen as the topic for the course BMEVIAUAL04 – *Project Laboratory*. The course is offered by the Department of Automation and Applied Informatics at the Budapest University of Technology and Economics during the Spring semester of 2025.

2 Purpose of the Schedule

The purpose of the schedule is to support the structured, balanced, and transparent execution of the project development. It helps to:

- prioritize tasks and understand the interconnections associated with individual features;
- ensure that every step of the implementation process provides adequate time for testing and validation;
- adhere to project deadlines and track progress effectively.

The schedule follows an iterative development model, where functionalities and the user interface (UI/UX) are implemented in parallel, with maximum consideration given to quality assurance aspects.

3 Development Schedule

The following schedule details the development goals and expected outcomes, as well as potential technological solutions, on a weekly basis. The planned milestones ensure the phased verifiability of the project and the efficiency of the implementation.

Week 1 - Project Initialization and Repository Creation

In the first week, the project will be initiated, including the preparation of this document and the development environment setup. This involves configuring the version control system, creating the project skeleton, and defining the core classes and components.

Objectives

- Set up the version control system (Git and GitHub).
- Create the server-side foundation using the Spring Boot framework.
- Initialize the client-side React project.
- Create the skeleton for the entities specified in the project specification.
- Develop the source code for the basic components (e.g., homepage, registration page).

Technological Solutions

- **Git Version Control:** To ensure traceability of the project development and facilitate collaboration.
- **Spring Boot:** A modern, robust server-side framework based on Java.
- **React and Vite:** Fast and flexible client-side development.

Tasks

- Create a remote repository on GitHub and assign the project consultant.
- Initialize the server-side using Spring Initializr (Java 21, Spring Boot 3.4.2).
- Create the React project using Vite (JavaScript-based).
- Define the class skeletons for the entities specified in the project specification (e.g., **User**).
- Define the skeletons for the basic components (e.g., homepage, registration page).

Week 2 - User Management

In the second week, user management, authentication, and authorization will be implemented, including the registration, login, and verification processes.

Objectives

- Implement server-side user registration and verification.
- Authentication and authorization secured with JWT (JSON Web Token).
- Introduce caching to track login attempt counts.
- Implement client-side registration, login, and verification pages.

Technological Solutions

- **Spring Security:** For handling authentication and authorization processes. Caching solution to track login attempts and prevent brute force authentication attacks.
- **JWT (JSON Web Token):** For securely managing user sessions.
- **Spring Email API:** For implementing email-based verification.

Tasks

- Implement the server-side registration process (data validation, saving hashed passwords).
- Implement email-based verification (using Spring Email API).
- Handle login with JWT token generation and management.
- Configure Spring Security for caching login attempts.
- Develop client-side pages using React.

Week 3 - Restaurant Management

In the third week, CRUD (Create, Read, Update, Delete) operations for restaurants will be implemented, along with client-side display.

Objectives

- Manage restaurant data on the server-side.
- Search and filter restaurants by category.
- Display the menu of a selected restaurant.
- Implement the shopping cart (add and remove products).

Technological Solutions

- **Spring Data JPA:** For simplified database operations management.

Tasks

- Implement CRUD operations for restaurant management.
- Implement filtering by category on both server-side and client-side.
- Create the menu page and shopping cart display in React.
- Manage the cart's state (add, remove, edit items).

Week 4 - Product and Cart Management

In the fourth week, product management for restaurants and the implementation of the shopping cart functionality will take place. The main objectives are to manage products added to the user's cart, categorize products, and make them available for ordering.

Objectives

- Manage and manipulate product data (CRUD operations).
- Add products to the shopping cart.
- Display, edit, and finalize the contents of the cart.
- Implement category-based filtering for products.
- Implement client-side cart functionalities and an interactive product catalog.

Tasks

- Develop server-side APIs for product management (CRUD).
- Implement the cart data model on the server-side.
- Develop cart interactions in the React application (adding and removing products).
- Implement product filtering by category through API and frontend.
- Use React components and Redux for managing the cart's content.

Week 5 - Order and Payment Management

The goal for the fifth week is to implement order placement, order status management, and payment processes. By handling orders in detail and integrating payment systems, the full order cycle for the application can be implemented.

Objectives

- Server-side management of order data (CRUD operations).
- Management of order statuses (e.g., In Progress, In Delivery).
- Payment integrations, such as PayPal and cash payments.
- Client-side functionality for placing an order.

Technological Solutions

- **Stripe/PayPal API:** To implement payment integrations.

Tasks

- Develop server-side order API and implement status management.
- Design and implement the order process UI in React.
- Integrate handling of payment methods (PayPal, cash).
- Finalize the order and request payment details on the client side.

Week 6 - Reviews and Coupons Management

The focus of the sixth week is on the reviews and coupons system. Integrating user feedback and promotional codes allows for improving the user experience and the functionality of the application.

Objectives

- Implementing user reviews and ratings for restaurants.
- Managing coupons and promotional codes.
- Handling reviews and coupons on the client side.

Tasks

- Develop server-side API for reviews and coupons.
- Store and display user reviews and ratings.
- Generate and apply coupons during the order process.
- Implement review and coupon management on the client side.

Week 7 - Admin Panel and Restaurant Management

In the seventh week, the focus is on completing the administrator panel and restaurant management. Implementing roles and administrative tools will conclude the development of the application's primary features.

Objectives

- Development of the admin panel and its functionality.
- Implementation of restaurant activation and authentication processes.
- Management of account suspensions and handling.
- Handling promotions and coupons by administrators.

Tasks

- Managing administrators and restaurants on the server side.
- Developing the admin interface and UI components.
- Managing coupons and promotions on the admin panel.

Week 8 - UI/UX Development and Refinement

The goal of the eighth week is to fine-tune the user interface, ensuring smooth functionality and a pleasant user experience, as well as presenting the current state of the project to the project consultant.

Objectives

- UI/UX fine-tuning, optimization of interactions.
- Client-side bug fixes and optimization.
- Presentation of the current state of the project to the project consultant.

Tasks

- Refining the user interface, improving the color scheme, typography, and layout.
- Holding a demonstration for the project consultant, showcasing the current state of the project, including the implemented features, remaining tasks, and any issues encountered and their resolution.

Week 9 - Responsive Design

The goal of the ninth week is to further refine the user interface by implementing responsive design on the client side.

Objectives

- Ensuring responsive design and cross-browser compatibility.
- Further fine-tuning of UI/UX and optimization of interactions.
- Client-side bug fixes and optimization.

Tasks

- Implementing responsive design optimized for various display devices.
- Fixing client-side bugs and testing on different devices.

Week 10 - Containerization and DevOps Integration

The goal of the tenth week is to implement the containerization of the project and integrate DevOps tools, preparing the application for continuous deployment and maintenance.

Objectives

- Containerize the application using Docker.
- Set up a CI/CD pipeline that automates the testing and deployment processes.
- Integrate DevOps tools - GitHub Actions.

Technological Solutions

- **Docker:** For containerizing the application and running it across multiple environments.
- **CI/CD:** Ensuring automated processes for quick and reliable deployments.

Tasks

- Create and optimize Docker images.
- Set up a CI/CD pipeline that includes build, test, and deployment steps.

Week 11 - Testing and Bug Fixing I.

The eleventh week focuses on the comprehensive testing of the project, which includes detailed checks of individual modules and features.

Objectives

- Functional testing of the application and identification of bugs.
- Fixing identified bugs to improve functionality.

Technological Solutions

- **JUnit framework:** For testing code snippets.

Tasks

- Writing and running unit tests.
- Detecting and fixing any bugs that are discovered.

Week 12 - Testing and Bug Fixing II.

The twelfth week continues the testing process, focusing on integration and end-user testing.

Objectives

- Integration testing of the application to ensure proper collaboration of different modules.
- Fixing bugs and further fine-tuning of the application based on user feedback.

Technological Solutions

- **Spring Boot built-in testing tools:** For performing integration testing.

Tasks

- Performing integration tests.
- Organizing and analyzing user tests.

Week 13 - Developer Documentation and Project Presentation I.

During the thirteenth week, the full development documentation of the project will begin, along with the preparation of the results for presentation.

Objectives

- Detailed preparation of the developer documentation.
- Designing and preparing the presentation necessary for the project demonstration.

Technological Solutions

- **LaTeX:** For creating professional documentation.
- **PowerPoint:** For creating professional presentations.

Tasks

- Writing the documentation describing the functionality of the project.
- Designing and drafting the outline of the presentation.

Week 14 - Developer Documentation and Project Presentation II.

During the fourteenth week, the finalization of the documentation and the project presentation will take place.

Objectives

- Finalizing and reviewing the developer documentation.
- Finalizing and reviewing the presentation.
- Planning and preparing for the project demonstration.

Technological Solutions

- **LaTeX:** For fine-tuning the documentation.
- **PowerPoint:** For fine-tuning the presentation.

Tasks

- Finalizing and submitting the documentation.
- Finalizing and submitting the presentation.
- Practicing the project demonstration.

Week of Repeats - Project Demo

During the week of repeats, the final project presentation will take place, where I will demonstrate the development process and the results.

Objectives

- Successful presentation of the project.

Tasks

- Present the project to the supervising board during the report.