

Project Report – Jeremias Wahlsten

Introduction

This project is a full-stack web application designed as a Kanban board. Registered users can create customizable columns and cards, which can be freely edited and moved between columns. The application provides an interactive and intuitive interface for organizing tasks and workflows. The application is designed with responsive design in mind, which means it can be used in any kind of device like a desktop or a phone.

As a full-stack project, it includes both a frontend and a backend, each built using modern web development technologies. This application was developed as the final project for a course.

This report outlines the technologies used in the project, provides installation instructions for both the frontend and backend, and presents a feature list along with a scoring breakdown. A declaration of AI usage is also included at the end of the document.

Technology

This project is built using the **MERN stack**, which includes **MongoDB**, **Express**, **React**, and **Node.js**. This is the most used tech-stack when it comes to Full-stack applications. The source code is divided into two main parts:

- The **client**, which serves as the frontend
- The **server**, which handles the backend logic

Backend

The backend is developed with **Node.js** and **Express**, and it uses **MongoDB** as the database. Additional technologies and middleware used include:

- **Morgan** – for HTTP request logging
- **CORS** – to enable Cross-Origin Resource Sharing
- **JWT (JSON Web Tokens)** – for user authentication and secure route access
- **bcrypt** – Used for hashing passwords and ensure their safety

Frontend

The frontend is built using **React** with **TypeScript** and powered by **Vite** for fast development. Styling is handled with a combination of:

- **TailwindCSS** – for utility-first styling
- **Material UI** – for ready-made, responsive UI components

Other libraries used to enhance functionality include:

- **dnd-kit** – for drag-and-drop functionality
- **react-i18next** – for internationalization support
- **react-router** – for client-side routing
- **axios** – for making API requests

In addition to these libraries, **prettier** was used as the linter for the whole project.

Installations

To install this and try out for yourself, you first need some pre-requisites:

- Node.JS (version 22.5)
- node package manager (npm, version 10.9)
- MongoDB (community edition, version 8.0.9)

After you've gotten the pre-requisites, you can proceed to the installation of this application. First, clone the GitHub repository. Once cloned, open the terminal in the root folder and run the command

```
$ npm run install:all
```

This script installs all libraries and dependencies on both the backend and frontend. Once the download is finished, go ahead and run

```
$ npm run dev:client
```

then open another terminal on the same location and run

```
$ npm run dev:server
```

These commands start the frontend and backend separately (both commands need their own terminal). Now you should be able to go to the application with your preferred browser. Reminder: You need to register an account first to use the kanban board.

Features and scoring

This is a general features list for the project. it lists the biggest features that this project includes, and gives them a point rating. The point rating is from 0 to 50, and taken from the Advanced Web application course.

Feature	Points
Basic features with well written documentation	25
Utilization of a frontside framework (react)	3
Cards and columns have Drag n Drop	2
Columns can be reordered	1
Card's color can be changed	1
Cards have comments	3
Translation (English and Finnish)	2
Cards and comments have timestamps	4
Admin panel	3
time spent with the card's task	1
Total Points	45

Declaration of AI usage:

Following lists includes every Generative AI tool used in the project and how it was used:

- Copilot – student license, used in code:
 - o code auto complete (boilerplate)
 - o chat (error fixing and brainstorming ideas)
- ChatGPT, used for this document:
 - o Chat (for refining text to be more readable)