

Project Documentation

Advanced Web Applications 2024

Description

This document is the documentation for Advanced Web Applications 20205 course project work. The task was to implement a web application that lets users register, login and post cards to a kanban board (similar to, for example, Trello).

Mandatory requirements

Implementation of backend with Node.js

- You can of course use Express, Meteor or any other additional framework
- Java, PHP, Perl, Python, Ruby or other language solutions are not allowed in this course

Utilization of database

- MongoDB, MariaDB or any other you see fit
- All the data needs to be stored in a database

Authentication

- Users have to have an option to register and login
- You can use JWT or session based authorization
- Only authenticated users can see, add or remove columns or cards

Features

- Authenticated users can:
 - Add/remove/rename columns to/of their own board
 - Add/move/remove cards on/of their own board
 - Cards should be movable to both up and down and between columns (you can simply build buttons or have drag and drop feature)
 - Logout
- Non-authenticated users can register and login

Responsive design

- The app needs to be usable with mobile devices and desktop browsers
 - Use of Materialize, Tailwind or Bootstrap is recommended

Documentation

- There needs to be documentation describing the technology choices, installation guidelines and user manual
- Also list the features you have implemented and the number of points you are aiming at
- The documentation MUST also include declaration of AI usage:
 - Name all AI systems that were used in the development of the contents of this document, and for each
 - How and where they were used (illustrations, proofreading, getting ideas for text, to generate diagrams etc.) or
 - Clearly state that no AI assistance or tools were used in this assignment.
- This Declaration is a mandatory part of the submission and leaving it out means that the work is incomplete and should not be graded. Declaring that no AI was used in the development of the document, but getting a high number in Turnitin AI check is also a valid grounds for failing the work just by itself.

Declaration of AI usage

AI systems were used in the making of this project.

AI systems that were used: ChatGPT

AI was used for:

- Generating ideas
- Research and suggesting best practices
- Troubleshooting and bug fixing

Main ideas behind the work

I wanted to create a kanban board with a modern look, so I decided to use Material UI and React. I used MUI templates to style the app and build some of the components. React Router was used to achieve a dynamic web app. Themes can be changed between light, dark and system default.

User can register with username and password, and login / logout. When not logged in, user can only see a Welcome page and the login and registration pages. Authentication uses JWT.

The user can have create multiple Boards (kanban boards). Boards have a name and consist of columns; user can rename and delete them. Boards can be shared with multiple users for collaboration. Collaborators can be added from the Collab menu.

Columns have a name and multiple Cards; columns can be added, deleted, rearranged (actions are in a dropdown menu behind a three dots button).

Cards have a title and a body text. User can add Cards to columns, delete them and rearrange them inside the Column using drag and drop, and also move them to other Columns using drag and drop.

When the user logs in, the first page shown is the Boards page. Here the user can see his/her boards, create new boards, delete boards and add collaborators to boards. Board names can be edited by double clicking the name; board can be opened by clicking the body of a board card.

The backend uses express for routing and MongoDB as the database. There are routes to register a user and login (authRoutes). Boards, Cards and Columns have their own routes with POST, GET, PUT, and DELETE methods.

The front end and back end are proxied together.

Points

Feature	Max points	Implementation	Points proposition
Basic features	25	Basic features have been implemented	25
Frontside Framework	3	Front end uses React	3
Cards can be reordered with drag and drop	2	Implemented	2
Columns can be reordered	1	Implemented, each column has a move left /move right button (behind 3 dots button)	1
User can set the color of a card	1	Not implemented	0
Login with Facebook, Google, X etc.	3	Not implemented	0
Content can be edited by double clicking	4	Implemented for card title and text, column name and board name (in the breadcrumbs)	4
Multiple users can work with the same board and the (card/column/comment) is blocked from another user	5	Multiple users can work on the same board	4
		User can have multiple boards	1
		Modern look and feel	2
		Themes: Light / Dark / System	2
		User can forget his/her password	0
Sum			44

Sources

MUI Templates

Material UI (MUI) template code was used heavily in the front end to easily achieve the look and feel of the app. These templates were used:

- Dashboard: <https://github.com/mui/material-ui/tree/v6.4.6/docs/data/material/getting-started/templates/dashboard>
 - The theme code is also from this template
- Sign-in: <https://github.com/mui/material-ui/tree/v6.4.6/docs/data/material/getting-started/templates/sign-in>
- Code for dropdown menus: <https://mui.com/joy-ui/react-menu/?srsltid=AfmBOooljrb1XjBflc4a0iDGFaKvwplh9-aJk3OA2NbTFOD6gtqJFKCX>

MUI templates are available at: <https://mui.com/material-ui/getting-started/templates/>
MUI documentation: <https://mui.com/material-ui/getting-started/>

Other Sources

- JWT Decode, available at <https://www.npmjs.com/package/jwt-decode>
- DND kit, available at <https://dndkit.com/>
- DND kit, Sortable Context, available at <https://docs.dndkit.com/presets/sortable/sortable-context>
- React Router, Navigating, available at <https://reactrouter.com/start/library/navigating>
- HTTP, Response Status Codes, available at <https://developer.mozilla.org/en-US/docs/Web/HTTP>
- StackOverFlow, Adding sx prop to custom component, available at <https://stackoverflow.com/questions/69624033/adding-sx-prop-to-custom-component>
- React, Rendering lists, available at <https://react.dev/learn/rendering-lists#keeping-list-items-in-order-with-key>
- MUI, Material Icons, available at <https://mui.com/material-ui/material-icons/>