Advanced Web Application Project

This is documentation for Advanced Web Applications Project "Board App".

1. Technology choices

Frontend: React.js and Material UI
Backend: Node.js and Express.js, MongoDB and Mongoose, JWT for authentication

Code is written using TypeScript

2. Installation guidelines

Prerequisites: Node.js, npm, MongoDB need to be installed
Step 1: Clone repository
- Run commands "git clone https://github.com/onniriikonen/project.git" and cd project

Step 2: Backend setup
- Go to backend folder "cd server"
- Install dependencies "npm install"

Step 3: Frontend setup
- Go to frontend folder "cd client"
- Install dependencies "npm install"

Step 4: Start
- Go to root folder and run command "npm run dev" to start
- Frontend will run on port 3000, and backend on 8000

3. User manual

The page opens a login form page. To register new user, click "Register" on the navbar. Once registered, user will be redirected to login page. Once logged in, user can add new boards, delete boards and open them.

Once user opens a board, he can add new columns and delete them. By clicking the column header, user can also rename columns. By clicking "Add Card", site opens a modal, where user can put card title and description and add that to a column. User can delete or move cards inside the column by pressing up or down buttons.

Everything is saved to MongoDB in "mongodb://127.0.0.1:27017/kanbandb", so existing users save their data and board data there. Logged users can also logout, by clicking "Logout" on the navbar, which redirects back to the login page. Also, if non authenticated user tries to go to "/board" or "/boards" he gets redirected to login page always.

4. Features implemented and points

This project has all the basic features (25p), and it utilizes frontside framework (React) (3p).

Points proposal: 28 points

5. AI Statement

No AI tools were used