KanbanBoard

This project involved utilizing a lot of different techniques to create responsive and dynamic kanban board. The project is deployed on GitHub. This documentation has Declaration of AI usage, Tools description, justification of points, installation guidelines, user manual, list of deprecated features and other sources that helped in creation of this project, in this particular order.

Declaration of AI Usage

Several models were used in the creation of the project. ChatGPT o1, ChatGPT 4o, ChatGPT o3-mini-high, Copilot

- AI systems were used in the development of this project. Specifically in debugging, error handling, commenting and verification of code. Providing good ideas for the code and helping in generating code. Especially in the areas like (drag and drop logic, debugging logs, sortable library, dnd library, modals, double tap, css styling, timestamps, and in some backend logic and backend debugging, and a little help in other technologies/tools). AI has been used in such files as: App.tsx, App.css, AuthPage.css, AuthPage.tsx, CardComponent.tsx, ColumnComponent.tsx, KanbanBoard.css, KanbanBoard.tsx, ModalPrompt.css, ModalPrompt.tsx, useDoubleTap.ts, index.ts, KanbanRoute.ts, validateToken.ts, server.ts. And overall, through whole project.
- AI systems were used in the development of this document to proofread and find grammar errors.

Tools and Libraries Frontend

- **HTML5:** Markup language, one file of it to initialize React.
- **REACT + VUE + TS:** Whole frontend is built on the frontend (frontend, hooks, etc.)
- **CSS:** Styling of the application
- **Sortable:** Used for reordering elements, REACT library
- **DND-kit:** Drag-and-drop functionality, REACT library
- **Jwt:** Used to delete the token if expired.
- **I18n:** Provides localization support.
- **Toastify:** Displaying toast notifications.
- **React-responsive:** Managing responsive breakpoints

Tools and Libraries Backend:

- **Node.js & Express:** Building the foundation of the server and API routes.
- **MongoDB:** Database to store boards, columns, cards, and comments.
- **TypeScript:** Backend is built on the typescript. All APIs routes and server logic is implemented with it.
- **JWT:** Authentication tokens.
- **Bcrypt:** Hashing the passwords
- **Dotenv:** Environment variables.

Development tools:

- **Github:** Version control and submission.
- **VSCode:** Code editing and development.
- **AI ChatGPT systems (o3-mini-high, o1, 4o):** All help from AI described previously in the document.

Justification of points and implemented features:

Feature	Points	Reason.
Basic features (as stated in the previous chapter) with well written documentation.	25	All basic features implemented. Documentation is clear
Utilization frontend framework. React and Vue	+3	Utilized React and Vue to develop frontend
Cards can be reordered with drag and drop	+2	Users can drag and drop cards in columns and between them. Using drag handle.
Columns can be reordered	+1	Columns can be reordered with drag and drop. Using drag handle for drag and dropping.
User can set the color of the card	+1	Users can set the color of the card. Can choose between all RGB patterns.

Provide a search that can filter out only those cards that have the searched keyword	+3	Users can filter a card. When user filters a card, only specific card is shown and column in which this card is.
User has the option just to double click any edible content (like header or card description) and edit it	+4	Users can double click and edit column title, card title, card description, card comments, and board title as an extra.
Cards can have comments in them one or many	+3	Users can set as many comments as they want.
Cards and comments have visible timestamps when they have been created and updated	+4	Each card and comment have timestamps when cards or comments were created and updated.
Cards have estimated time when work is done	+1	User can set the estimated time in specific time format, edit it and delete it.
Translation of the whole UI in two or more languages	+2	UI is translated in 2 languages. Finnish and English
EXTRA: User can delete and edit comments with double click.	+2?	Not mentioned in the possible features (EXTRA). Users can delete or edit comments.
EXTRA: Board Name can be changed with double click.	+1?	Board Name can be changed with double click.
EXTRA: Usage of modals instead of popup prompts.	+ 1?	Usage of modals to delete (card or column) to add card or column and etc. Used instead of prompts.
Project is returned before February 23rd	+1	Returned on 19.02.2025

Total:	54 with extra, without extra 50.	

Installation guide:

1. Go to your system for e.g VsCode. Create new folder, in this new folder insert command next command to the terminal to get project:
   a. git clone https://github.com/EliasKukkonen/AdvancedWebProject.git

   ```
   PS C:\Users\elias\New Test Folder> git clone https://github.com/EliasKukkonen/AdvancedWebProject.git
   Cloning into 'AdvancedWebProject'...
   remote: Enumerating objects: 113, done.
   remote: Counting objects: 100% (113/113), done.
   remote: Compressing objects: 100% (84/84), done.
   Receiving objects:  89% (101/113)sed 110 (delta 16), pack-reused 0 (from 0)Receiving objects:  86% (98/113)
   Receiving objects: 100% (113/113), 130.26 KiB | 1.53 MiB/s, done.
   Resolving deltas: 100% (16/16), done.
   ```

2. Next go to the project itself.
   a. cd AdvancedWebProject

   ```
   PS C:\Users\elias\New Test Folder> cd AdvancedWebProject
   PS C:\Users\elias\New Test Folder\AdvancedWebProject>
   ```
   b.

3. Next run npm install to install dependencies into the root of the project.
   a. npm install

   ```
   PS C:\Users\elias\New Test Folder\AdvancedWebProject> npm install

   added 50 packages, and audited 51 packages in 7s

   7 packages are looking for funding
     run `npm fund` for details
   ```
   b.
   ```
   found 0 vulnerabilities
   ```

4. Next go to server folder and install dependencies there.
   a. cd server
   b. npm install

   ```
   PS C:\Users\elias\New Test Folder\AdvancedWebProject> cd server
   PS C:\Users\elias\New Test Folder\AdvancedWebProject\server> npm install
   npm WARN deprecated react-beautiful-dnd@13.1.1: react-beautiful-dnd is now deprecated. Context and options: https:/
   github.com/atlassian/react-beautiful-dnd/issues/2672

   added 228 packages, and audited 229 packages in 2s

   22 packages are looking for funding
     run `npm fund` for details
   ```
   c.
   ```
   found 0 vulnerabilities
   ```

5. Next go back to root folder, switch to the client/FrontEndReactKanban and install dependencies
   a. cd ..
   b. cd client\FrontEndReactKanban
   c. npm install
   d.

```
● PS C:\Users\elias\New Test Folder\AdvancedWebProject\server> cd ..
● PS C:\Users\elias\New Test Folder\AdvancedWebProject> cd client/FrontEndReactKanban
● PS C:\Users\elias\New Test Folder\AdvancedWebProject\client\FrontEndReactKanban> npm install
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'vite@6.0.11',
npm WARN EBADENGINE   required: { node: '^18.0.0 || ^20.0.0 || >=22.0.0' },
npm WARN EBADENGINE   current: { node: 'v21.2.0', npm: '10.5.2' }
npm WARN EBADENGINE }

added 210 packages, and audited 211 packages in 5s

44 packages are looking for funding
  run `npm fund` for details

3 moderate severity vulnerabilities

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

6. Next go back to the root folder, build the project, and start it.
   a. Cd ..   (2 times)
   b. npm run build
   c. npm run start
   d. Example:
   e.

```
PS C:\Users\elias\New Test Folder\AdvancedWebProject\client\FrontEndReactKanban> cd ..
● PS C:\Users\elias\New Test Folder\AdvancedWebProject\client> cd ..
● PS C:\Users\elias\New Test Folder\AdvancedWebProject> npm run build
```

   f. Example of success build:
   g.

```
vite v6.0.11 building for production...
✓ 86 modules transformed.
dist/index.html                         0.48 kB │ gzip:   0.31 kB
dist/assets/index-FBpQXH2u.css         23.14 kB │ gzip:   4.38 kB
dist/assets/browser-ponyfill-CELojYZB.js  10.29 kB │ gzip:   3.52 kB
dist/assets/index-DBx73H_r.js         321.66 kB │ gzip: 100.88 kB
✓ built in 833ms

> projectadvancedfinal@1.0.0 build:server
> cd server && npm install && tsc

up to date, audited 229 packages in 789ms

22 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

   h. Example of started project
   i.

```
PS C:\Users\elias\New Test Folder\AdvancedWebProject> npm run start

> projectadvancedfinal@1.0.0 start
> cross-env NODE_ENV=production node server/dist/server.js

Serving frontend from: C:\Users\elias\New Test Folder\AdvancedWebProject\client\FrontEndReactKanban\dist
Server running on port 3000
```
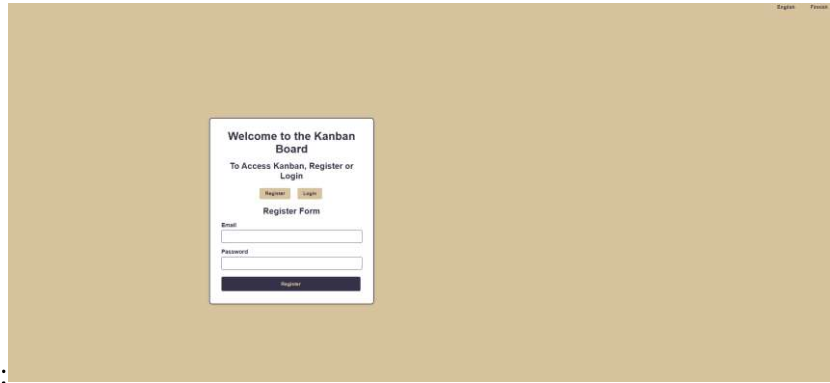
7. See project in the http://localhost:3000/

a. First page:

User manual

Getting started to use the kanban project.

- Register use. When accessing the project, you can see the register first. Insert the email and password to create the account. Click register. Now you are registered.
- Login use. After registration you are automatically redirected to the login page. Insert your new credentials that you have inserted in the register page. And press login.
- If you want to switch between login and register form, just press "register" or "login" button above input texts.
- To switch language, press "English" or "Finnish" buttons in the top right corner.

Usage of Kanban board:

Getting started (add column and rename column).

- Click Add column in top right corner. Modal form popups, insert the name of column. Click "OK" to add.
- To rename column just double-click (or double tap if on mobile device) to the column title to rename it. Press "enter" to save changes or just tap the screen.

Adding cards (adding comments, changing colors, setting estimated time).

- In the column click "Add card". Modal popups insert card title and description (optional), click "OK" to add card.
- To rename card title or card description, just double-click or double tap them. Press enter to save or tap screen.

- To show comment or add comments. "Click Show Comments or Add new Comment". Write a new comment in the area, click "Add Comment". To edit comment, double-click. Delete comment by pressing "X". To hide comments click "Hide Comments"
- To show estimated time, click: "Show Estimated Completion time or Add New". To add new estimated time, click "Set Estimated Time", modal popups, input time in format "YYYY-MM-DD", other formats system declines. Hide Estimated time by clicking "Hide Estimated". Remove with "Remove" button.
- Add color, by clicking "Add Color" button. You can choose from all RGB colors. After you have chosen the color, press somewhere on the screen to close the overlay.
- Delete card by pressing "Delete Card".

Reordering the cards and columns, language change, and search.

- Drag and Drop is implemented via drag handles. Cards and Columns have drag handles. Use a drag handle to move the card or column around. Columns are rearranged only horizontally. Cards can be rearranged inside the column and between columns.
- To change the language simply click, "Eng" or "Fin" button.
- To search for the specific card, input the card title (or description) into the search input and press "search". To show again all cards, delete the text inside search input and press "search" again.

Features that were deprecated and not used.

- **Admin account.** At the start of the project there was an idea to create an admin account. But this feature were deprecated. Still functionality persists in the code, but it is not used, and the user can't create an admin account.

Other sources that helped in the creation of this project.

- Color theme of this kanban board:
  https://2colors.colorion.co/