

React.js Todo List Application Task for Kadritech.AB

Project Description

You are required to build a **Todo List Application** using React.js. This application will feature a Kanban-style interface with three distinct columns: **New**, **Ongoing**, and **Done**. Each column will serve as a container for the todo items, allowing users to manage tasks through different stages of completion. The todo items should have a title and description, and each item should be labeled according to its current status with a distinct color for easy identification.

Detailed Requirements

1. Three Columns for Todo Management:

- **New:** The first column where new todo items are created.
- **Ongoing:** The second column where tasks that are currently being worked on are moved.
- **Done:** The third column where completed tasks are placed.

2. Todo Item Creation:

- Users should only be able to create new todo items in the **New** column.
- Each todo item must have:
 - A **title**: Short text describing the task.
 - A **description**: Detailed information about the task.
 - A **status label**: Indicates whether the task is **New**, **Ongoing**, or **Done** with distinct colors for each status (e.g., blue for New, orange for Ongoing, green for Done).

3. Moving Items:

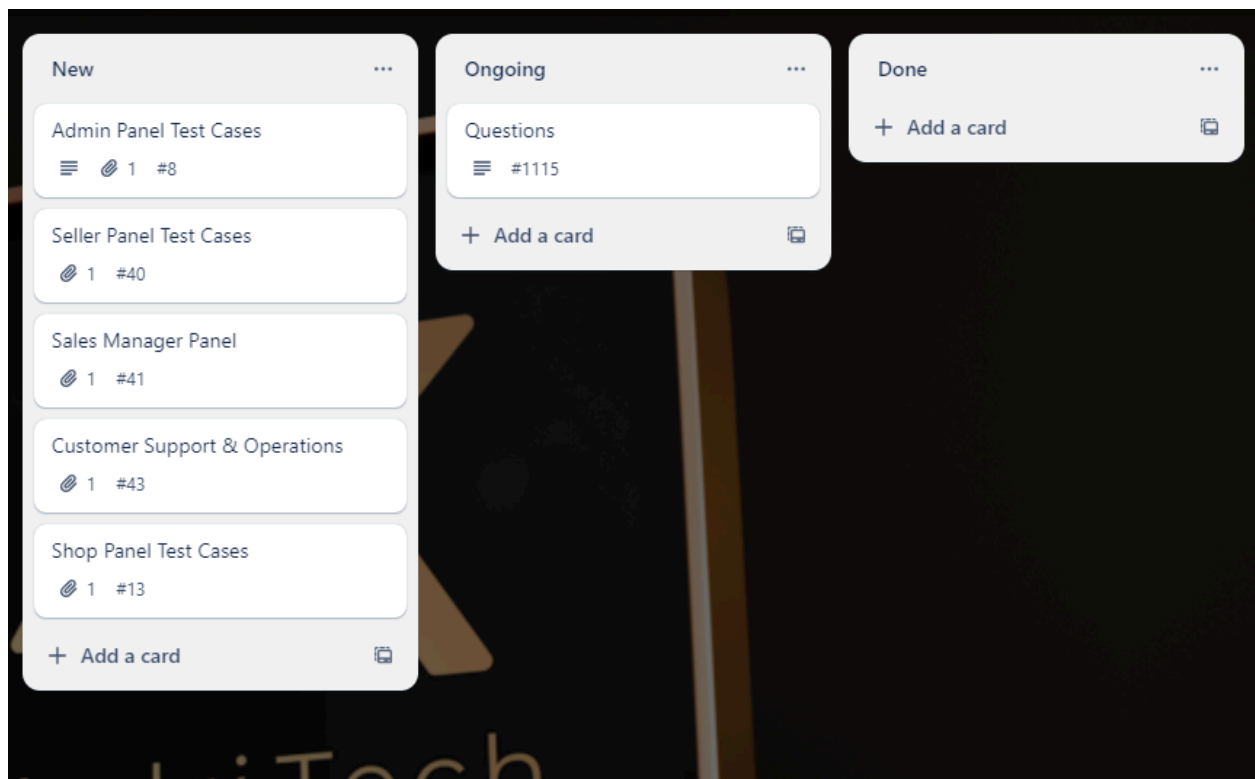
- Implement a right-click context menu for each todo item.
- The context menu should display options to move the item to other columns, excluding the current column. For example:
 - If the todo item is in the **New** column, the context menu should show options for **Ongoing** and **Done**.
 - If the todo item is in the **Ongoing** column, the context menu should show options for **New** and **Done**.
 - If the todo item is in the **Done** column, the context menu should show options for **New** and **Ongoing**.
- Selecting an option should move the todo item to the corresponding column.

React.js Todo List Application Task for Kadritech.AB

4. Task Management:

- **New Column:** Users can add new tasks here. Each new task should initially be placed at the top of the column.
- **Ongoing Column:** Users can move tasks to this column from the **New** column. The tasks should be displayed in the order they were moved. In addition, there should be a dateTime selector so that users can set time duration in order to check if the ongoing task is overdue or not. If it is due time then we must show an alert message.
- **Done Column:** Users can move tasks to this column from the **Ongoing** column. The tasks should be displayed in the order they were completed.
- Ensure the application is responsive and looks good on different screen sizes.
- Additionally, showcasing your own design ideas and creativity will be considered an advantage in your application.

For Example (Only demo purpose):



React.js Todo List Application Task for Kadritech.AB

Evaluation Criteria

This project will assess your proficiency in React.js, your ability to design and implement interactive UI components, and your problem-solving skills in handling task management logic effectively.

This detailed description provides clear guidance on what is expected from the candidate, emphasizing both technical implementation and design considerations. Adjustments can be made based on specific preferences or additional requirements for the role.