

## Task Description

Your task is to create a simple TODO list application using React.js. The application should allow users to create, read, update, and delete TODO items. Each TODO item should have a title, description, and due date.

## Requirements

- The application should have the following components:
  - A TODO list component that displays a list of all TODO items.
  - A TODO item component that displays the details of a single TODO item.
  - A form component that allows users to create or edit a TODO item.
- The application should allow users to create, read, update, and delete TODO items.
- The TODO list component should display the title, description, and due date of each TODO item.
- The form component should have fields for the title, description, and due date of a TODO item.
- The application should validate the user input for the form fields and display an error message if the input is invalid.
- The application should persist TODO items in the browser's local storage.
- The application should have a simple and intuitive user interface.
- The application should use React.js best practices, such as functional components and hooks.
- The application should be responsive and work well on different screen sizes.

## Bonus Requirements

- Implement sorting and filtering for the TODO list component, allowing users to sort by due date or filter by title or description.
- Implement pagination for the TODO list component, allowing users to specify a page size and navigate between pages.
- Implement authentication and authorization for the application, requiring users to log in and allowing only authorized users to create, read, update, and delete TODO items.
- Implement tests for the components and logic of the application using a testing framework of your choice.

## Evaluation Criteria

We will evaluate your submission based on the following criteria:

- **Correctness:** Does the application meet the requirements and behave as expected?
- **Code quality:** Is the code well-organized, easy to read, and maintainable?
- **User experience:** Is the user interface intuitive and easy to use?
- **Scalability:** Could the application be easily scaled to handle larger volumes of data or traffic?
- **Security:** Does the application implement proper authentication and access controls to protect against common web application vulnerabilities?
- **Testing:** Are there sufficient tests to ensure that the application behaves correctly and is maintainable?