Assignment Title: Full-Stack Developer Challenge - Task Manager Application

Introduction:

You are tasked with building a task management application similar to Trello. The application will allow users to create, update, and manage tasks within different columns. Users should be able to move tasks between columns using drag-and-drop functionality. Additionally, users should be able to sign up and log in, including the option to log in via Google.

Front-End Requirements:

User Interface:

- We have provided you with some mock designs. Try to build according to those designs.
- Drag-and-drop functionality is a must; use any library of your choice.
- Ensure routing is implemented throughout the application.
- Authentication should be required on every page.

Back-End Requirements:

Framework:

- Use Node.js with Express for building the backend of the application.

API Development:

- Create a RESTful API to handle CRUD (Create, Read, Update, Delete) operations for tasks.
- Implement routes for creating, updating, and deleting tasks, as well as for retrieving all tasks.
- Implement routes for user registration, login, and Google login authentication.

Data Storage:

- Use either an SQL (e.g., PostgreSQL, MySQL) or NoSQL (e.g., MongoDB) database to store task data and user information.
- Set up the necessary data models to represent tasks and users.

Validation:

- Implement server-side validation to ensure that task data is valid before saving it to the database. Tasks must have a title and belong to a valid column.
- Ensure validation for user registration and login data.

Error Handling:

- Properly handle errors, including sending appropriate error messages and status codes in response.

General Requirements:

Code Quality:

- Write clean, well-documented, and maintainable code. Use coding best practices and conventions for the chosen programming language and framework.

Version Control:

- Use a version control system (e.g., Git) to track changes in your code and provide a Git repository for the assessment.

Testing:

- Write unit tests for critical parts of your application, such as API endpoints and data validation.

Security:

- Implement basic security measures to protect the application from common vulnerabilities.

Bonus Features (Optional):

You can implement additional features to make your project stand out:

- User profiles with avatars.
- Task due dates and reminders.
- Task sorting and searching capabilities.

Submission:

Provide a link to your version-controlled repository (e.g., GitHub, GitLab). Include clear instructions on how to set up and run your application. Share any additional documentation or notes that might help reviewers understand your project.

Assessment Criteria:

Your assignment will be evaluated based on:

- **Functionality:** Does the application meet the specified requirements and work as expected?
- **Code Quality:** Is the code clean, organized, and well-documented?
- **User Experience:** Is the user interface intuitive and responsive?
- **Security:** Are there basic security measures in place?
- **Testing:** Are there unit tests for critical components?
- **Bonus Features:** If implemented, do they enhance the application's usability?

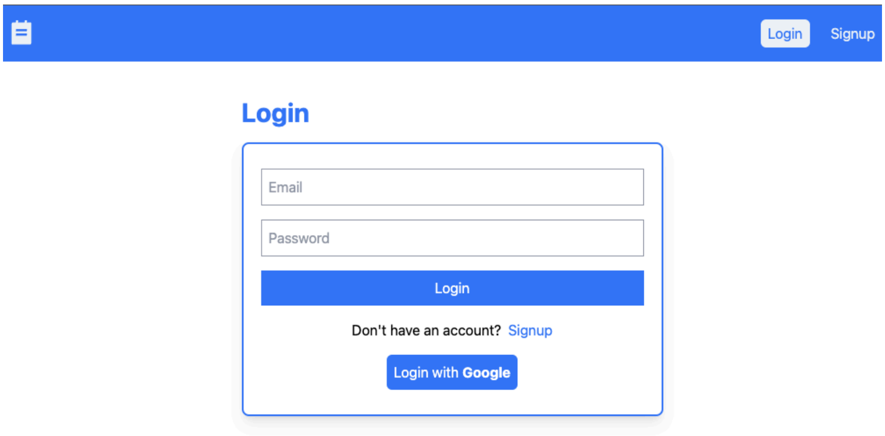
This assignment is designed to assess your full-stack development skills, so feel free to showcase your capabilities and creativity. Good luck!

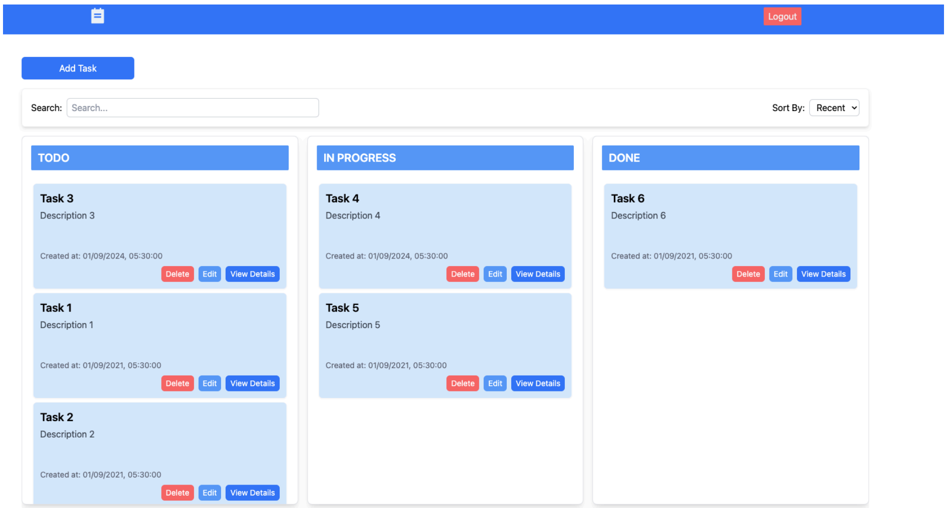
Please Note:

- Share the REPO link (Github)
- Deploy your app and share the app link. This is mandatory. You can use free services like Firebase, MongoDB, Netlify, etc., whichever you are comfortable with. If not shared, the deployment link will be rejected.
- Sharing the code over a Zip file or PDF will be **Rejected.**
- Errors should be handled beautifully on the client side so users have an idea of what is wrong.
- You must consider all possible errors and create solutions for them.
- Your app should work properly on the deployed link. If it is not working properly, it will be rejected, as we will check every functionality.
- Google login must be implemented.
- Send the completed assignment to richa@voosh.in.
- Some free resources.

A mock design is provided below; try to build it like that.

Design:





TODO

Task 3
Description 3
Created at: 01/09/2024, 05:30:00

Task 1
Description 1
Created at: 01/09/2021, 05:30:00

Task 2
Description 2
Created at: 01/09/2021, 05:30:00

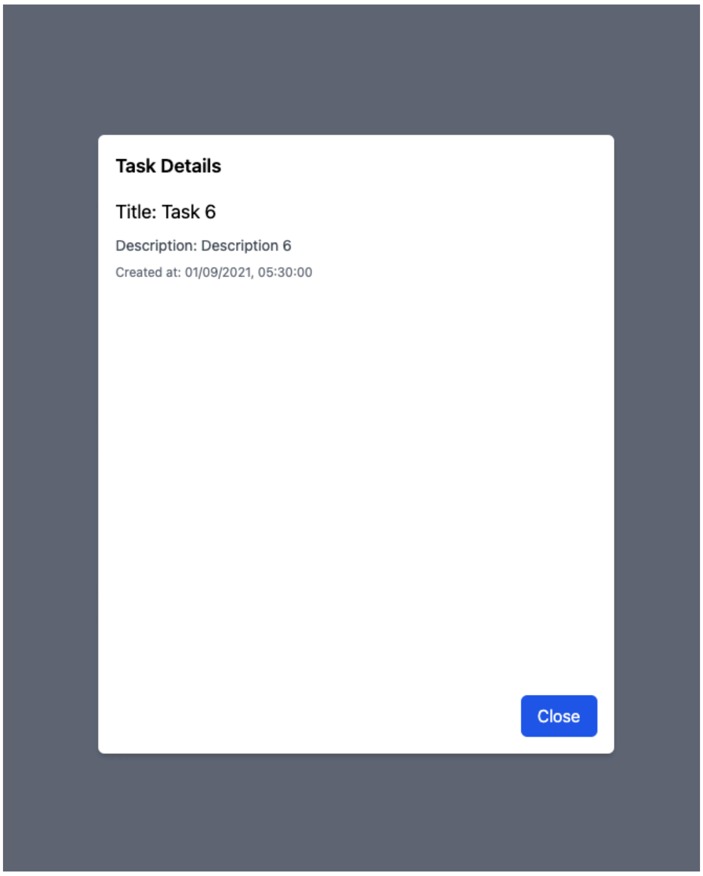
IN PROGRESS

Task 4
Description 4
Created at: 01/09/2024, 05:30:00

Task 5
Description 5
Created at: 01/09/2021, 05:30:00

DONE

Task 6
Description 6
Created at: 01/09/2021, 05:30:00

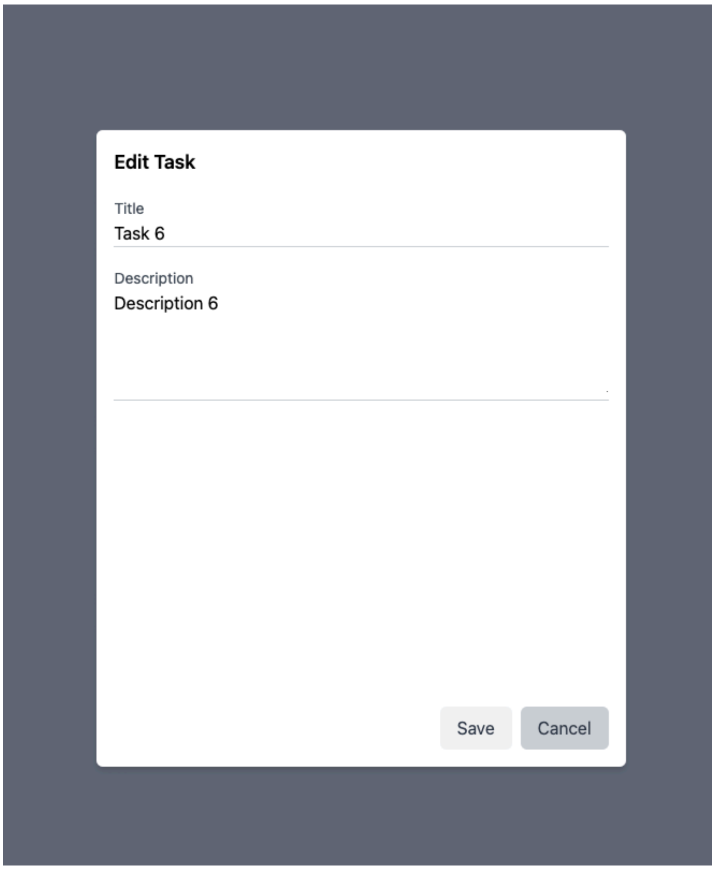


Task Details

Title: Task 6

Description: Description 6

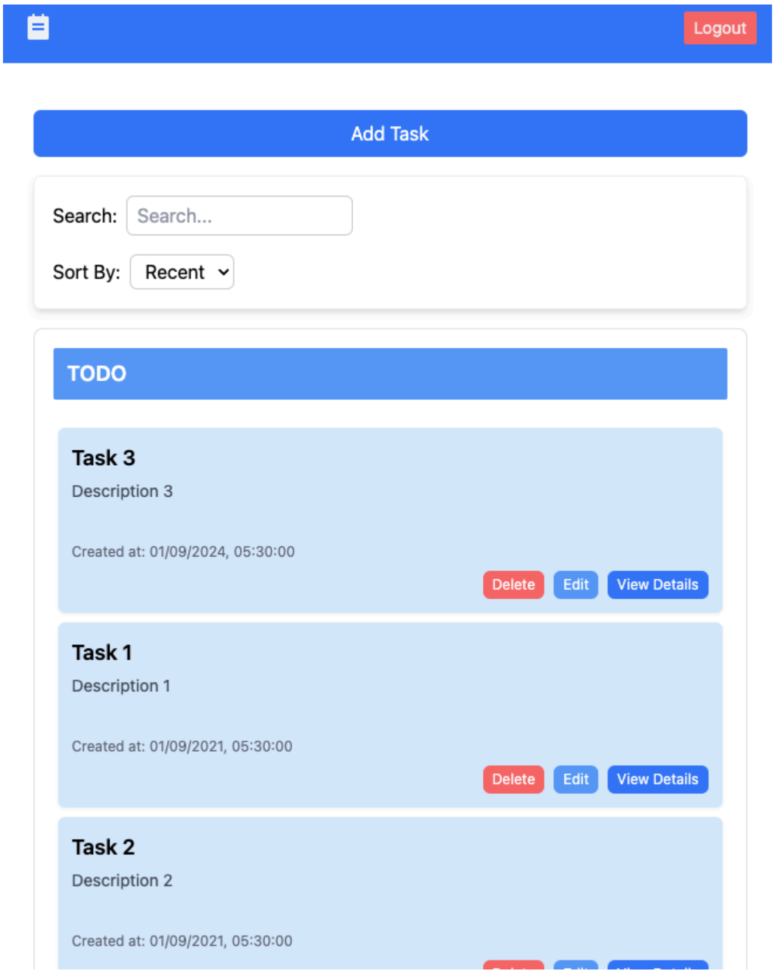
Created at: 01/09/2021, 05:30:00



Edit Task

Title: Task 6

Description: Description 6



Search: Search...

Sort By: Recent

TODO

Task 3

Description 3

Created at: 01/09/2024, 05:30:00

Task 1

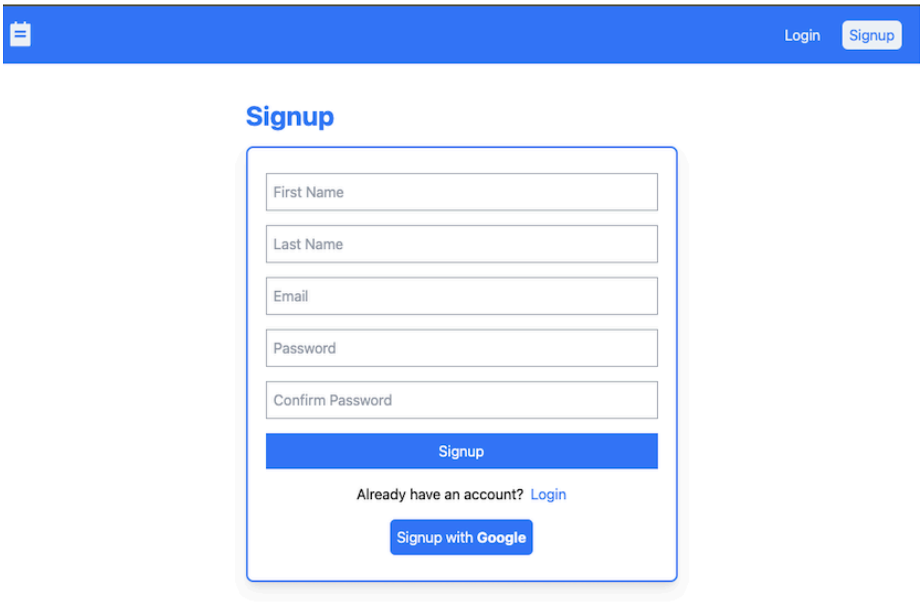
Description 1

Created at: 01/09/2021, 05:30:00

Task 2

Description 2

Created at: 01/09/2021, 05:30:00



Signup

First Name

Last Name

Email

Password

Confirm Password

Already have an account? Login