STACKWISR

Solent Business Park, Steel House
Plot 4300, Fareham Hampshire
PO15 7FP
Company No. 12092475

Frontend Developer Task: Building a Simple Dashboard

Objective:
Develop a simple dashboard application using React, JavaScript, and GraphQL. This task will assess your proficiency in frontend development, including component design, state management, and GraphQL integration.

Task Overview:
Create a dashboard that fetches and displays a list of users along with their details from a GraphQL API. The user should be able to search, filter, and sort the list based on specific fields.

Requirements:

Setup:
- Use React (preferably with hooks).
- Use JavaScript (TypeScript is a plus, but optional).
- Fetch data using GraphQL.
- Optionally, use a public GraphQL API or mock one using a tool like GraphQL Zero.

Features:
1. User List:
Display a list of users with the following fields:
- Name
- Email
- Phone

Company Name

2. Search:
- Implement a search bar to filter users by name or email.

3. Sort:
- Provide sorting functionality by name or company.

4. Pagination:
- Implement basic pagination to navigate through the list of users.

5. User Detail View:
- When a user is clicked, display more detailed information in a side panel or modal, including:
  - Name
  - Email
  - Phone
  - Address

  Company Information

Design:
- Maintain a clean and simple UI.
- Use any UI framework or CSS-in-JS solution (e.g., Material-UI, Tailwind CSS, Styled-Components).

GraphQL:
- Fetch the list of users using a GraphQL query.
- Implement pagination through GraphQL queries.
- Handle loading and error states gracefully.

Bonus Points:
- Implement sorting and filtering logic on the server-side using GraphQL queries.
- Use a state management solution (like Redux or React Context) to manage the application state.
- Write basic unit tests for your components.
- Deploy the application using a service like Vercel, Netlify, or GitHub Pages.

Submission:

Provide a GitHub repository link with your code.
- Include a README.md file explaining your thought process, setup instructions, and any assumptions or decisions made during development.

(Optional) Deploy the app and provide a live demo link.