EliteCA: Responsive Landing Page with API Integration

Introduction:

"EliteCA" is a user-friendly web application designed to help individuals search for Chartered Accountants (CAs) and view detailed profiles. Built with React and Tailwind CSS, the app offers a smooth, dynamic experience with a clean, mobile-first design. Featuring a mock RESTful API for easy data retrieval, users can quickly search for CAs based on specific criteria and navigate effortlessly between a well-organized homepage and detailed profile pages. The application's responsive design ensures it looks great on all devices.

Features:

- **Dynamic Search Functionality**: As the user types, the search bar instantly provides suggestions, retrieving relevant results from a mock RESTful API.
- **Detailed Profile Page**: A comprehensive page showcasing a Chartered Accountant's name, description, service charges, ratings, and image.
- **API Integration**: Seamless communication with a mock RESTful API via json-server to fetch, display, and update data dynamically.
- **Responsiveness**: The design adapts seamlessly to different screen sizes (mobile, tablet, desktop).
- **Error Handling**: Graceful handling of scenarios like no matches found or API request failures.

Profiles Page Design:

The Profile Page includes:

➢ **Search Bar**:

- Positioned prominently at the top for easy access.
- Allows users to type in search queries for Chartered Accountants.
- Displays real-time suggestions based on the user input, dynamically fetched from the API.

EliteCA: Responsive Landing Page with API Integration

➢ **Dynamic Search Suggestions**:

- A list of matching Chartered Accountants retrieved from the mock API.
- Each suggestion is clickable, redirecting the user to a details page.

Details Page Design:

The details page displays:

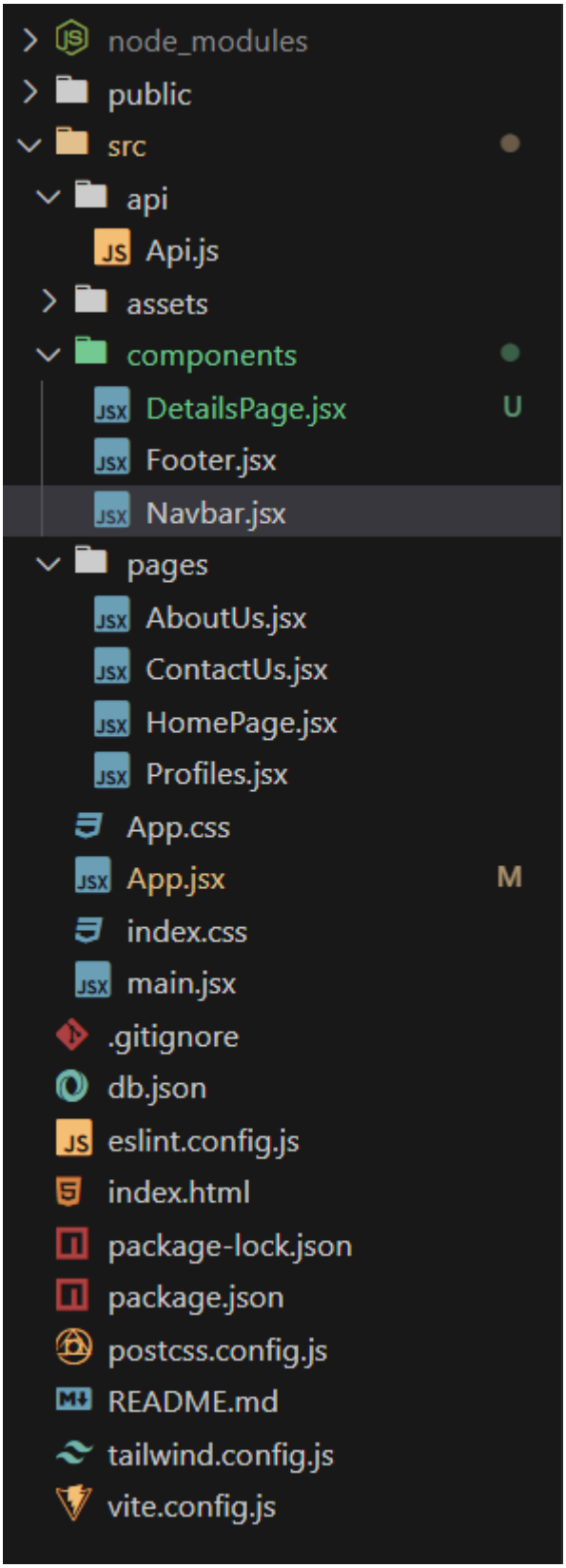
- Chartered Accountant Details:
  - Name. ○ Charges. ○ Description. ○ Ratings.
  - Profile Image.
- Back Button:
  - Allows users to navigate back to the homepage easily.

Technologies Used:

- **React:** Component-based architecture for building the UI.
- **Tailwind CSS:** For styling and ensuring responsive design.
- **json-server:** A mock RESTful API for data management.
- **React Router:** For navigating between the homepage and details page.
- **Axios:** For making API requests to the mock server.
- **Git/GitHub:** Version control and code sharing.

EliteCA: Responsive Landing Page with API Integration

Project Structure:



EliteCA: Responsive Landing Page with API Integration

API Integration:

The application uses json-server to serve data stored in db.json. The mock API includes Chartered Accountant details in the following format:


```
[
  {
    "id": 1,
    "name": "John Doe",
    "charges": "$500/hr",
    "description": "Expert in tax management and financial audits.",
    "rating": 4.8,
    "image": "path/to/image1.jpg"
  },
  {
    "id": 2,
    "name": "Jane Smith",
    "charges": "$300/hr",
    "description": "Specialist in corporate finance and accounting.",
    "rating": 4.5,
    "image": "path/to/image2.jpg"
  }
]
```

EliteCA: Responsive Landing Page with API Integration

Endpoints:

1. GET /cAs: Fetch all Chartered Accountants.
2. GET /cAs?name_like={query}: Fetch Chartered Accountants matching the search query.

Implementation Steps:

1. **Set Up the Environment**:

- Create a new React app using Vite: npm create vite@latest EliteCA --template react cd EliteCA npm install
- Set up Tailwind CSS:

npm install -D tailwindcss postcss autoprefixer

npx tailwindcss init

Configure tailwind.config.js and include Tailwind directives in index.css.

2. **Mock API**:

- Install json-server:

npm install -g json-server •

Create a db.json file with CA data.

- Run the mock server: json-server --watch db.json --port 5000

3. **Develop Components**:

- **SearchBar.jsx**:
  - Handles user input and fetches suggestions from the API dynamically.
- **Profiles.jsx**:

EliteCA: Responsive Landing Page with API Integration

- Displays the list of Chartered Accountants based on search results.

- **DetailsPage.jsx**:
  - Shows detailed information about a selected Chartered Accountant.

4. **Routing**:
   - Use React Router to handle navigation between the homepage and details page.

5. **Responsive Design**:
   - Leverage Tailwind CSS utilities for a mobile-first responsive layout.

6. **Error Handling**:
   - Display a message if no results are found or if the API request fails.

Usage:

1.**Run the Project Locally:** •

Install dependencies:

npm install

- Start the application and API:

npm run dev json-server --

watch db.json --port 5000

- Access the app at http://localhost:3000 and the API at http://localhost:5000.

2.**Search Chartered Accountants:**

- Enter a name in the search bar to see suggestions.
- Click a suggestion to view details.

3.**Test Responsiveness:**

- Use browser dev tools to test the app on mobile, tablet, and desktop.

EliteCA: Responsive Landing Page with API Integration

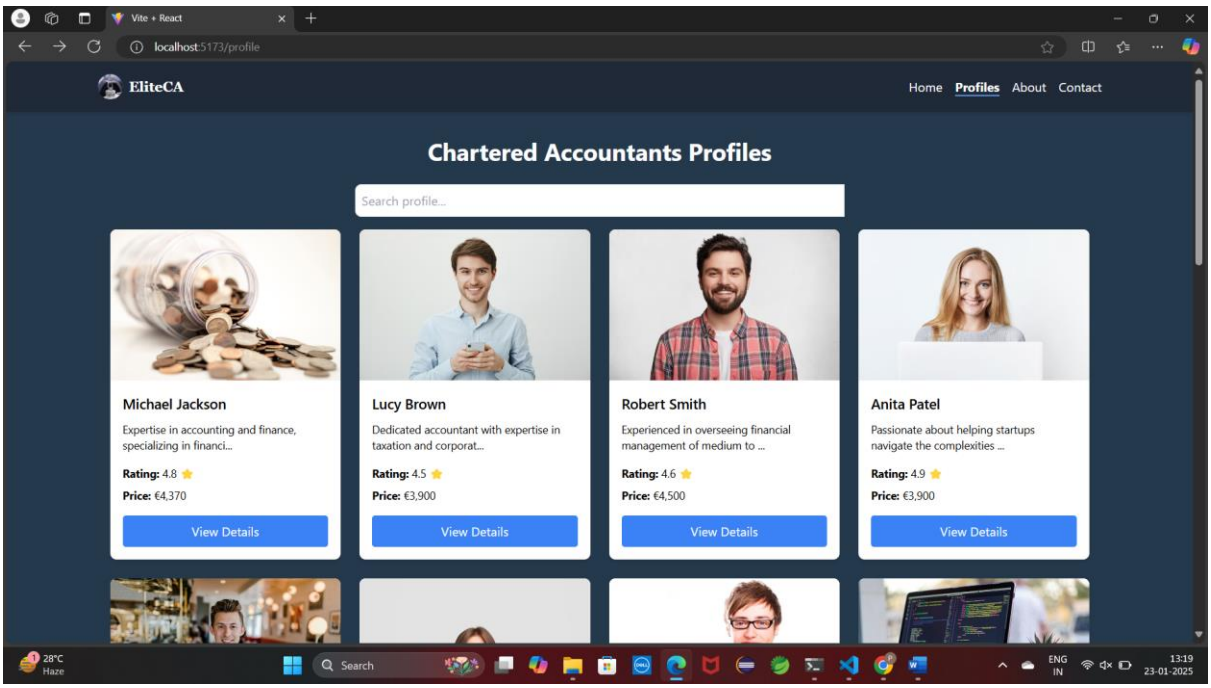
Output:

1.Profile Page:

- Fully responsive design with a search bar and dynamic suggestions.

2.Details Page:

- Displays detailed information about selected Chartered Accountants with a responsive layout.



Conclusion:

The EliteCA application demonstrates a clean, responsive design and efficient API integration using React, Tailwind CSS, and json-server. By following modern development practices, the project ensures scalability, maintainability, and user-friendliness.