Assignment -6

Objective

Creating a user management application in React involves several steps, from setting up your project to implementing CRUD operations using Axios for HTTP requests. Here's a comprehensive guide to help you develop a fully functional React app that interacts with a public API.

Project Requirement:

Fetch and Display Users:

- On component mount, fetch a list of users from the provided API.
- Display users in a table or card layout showing Name, Email, and Username.

Add User:

- Create a form with inputs for Name, Email, and Username.
- On form submit, send a POST request to add the user.
- Update the UI to include the new user.

Update User:

- Add an edit button for each user.
- When clicked, populate the form with the user's data.
- Allow editing and submitting changes via a PUT/PATCH request.
- Reflect changes in the UI after a successful update.

Delete User:

- Add a delete button for each user.
- Confirm deletion before sending a DELETE request.
- Remove the user from the UI on success.

Error Handling:

- Show appropriate error messages when API requests fail.
- Validate form inputs (non-empty and valid email format).

Loading State:

- Display a loading indicator while fetching data.

API to use:

Use the JSON Placeholder fake API:

- **GET users**: https://jsonplaceholder.typicode.com/users
- **POST user**: https://jsonplaceholder.typicode.com/users
- **PUT/PATCH user**: https://jsonplaceholder.typicode.com/users/:id
- **DELETE user**: https://jsonplaceholder.typicode.com/users/:id

Submission Guidelines

1. Deploy the React code on Netlify and Vercel.
2. If you don't know how to deploy React code on a free hosting platform, watch this YouTube video https://youtu.be/AP0fKMrmyKA?si=wKbklcCA3rGCFDb7
3. **Deploy Link:** Submit Deploy Link in Readme.md file of task-related GitHub repo.
4. **Readme.md:** Create the readme.md file like Professional Level, which includes only project-related content.
5. **Submission Link:** Submit the GitHub repo Link in the Submission Box.