**Technical Specification:**

**1. Creating a Project Using Create React App**  
◦ Create a new React project using Create React App with TypeScript (--template typescript).  
◦ Set up a clean project structure: separate folders for components, pages, store, and services.  
◦ Install required packages:  
    react-router-dom, axios, @reduxjs/toolkit, react-redux.

**2. Building Components**  
◦ Create modular and reusable functional components using TypeScript.  
◦ Develop components such as: Navbar, Footer, UserCard, UserList, UserForm, etc.  
◦ Use props and children effectively for reusability and flexibility.

**3. Props and Local State Management**  
◦ Manage internal component state using useState and useEffect.  
◦ Lift state when needed to share data between components.  
◦ Pass data via props from parent to child components.

**4. Event Handling and Forms**  
◦ Create forms using input elements (input, select, textarea).  
◦ Handle form events: onChange, onSubmit.  
◦ Implement basic form validation (e.g., required fields, email format).

**5. Routing with React Router**  
◦ Configure routes using react-router-dom v6+.  
◦ Create pages: /, /users, /users/:id, /add-user, /about.  
◦ Use useNavigate and useParams for navigation and dynamic route handling.

**6. Connecting to a REST API**  
◦ Use axios to fetch user data from a public API such as:  
    https://jsonplaceholder.typicode.com/users  
◦ Load data using useEffect and display in UserList.  
◦ Implement functionality to add and delete users (mocked via local or global state).

**7. Styling the Application**  
◦ Use CSS Modules or styled-components for component-scoped styling.  
◦ Ensure the app is responsive using flexbox/grid and media queries.

**8. Global State Management with Redux Toolkit**  
◦ Set up the Redux store using @reduxjs/toolkit and react-redux.  
◦ Create slices using createSlice() for managing user data (e.g., userSlice.ts).  
◦ Use createAsyncThunk for API requests (e.g., fetching users).  
◦ Dispatch actions and select data in components using useDispatch and useSelector.  
◦ Organize state logic clearly: store/index.ts, slices/userSlice.ts.