**Training plan for ReactJS, Redux, and unit testing with Jest & Enzyme:**

Duration: 14-days

Here's a 10-day training plan focusing exclusively on functional components based on hooks, for ReactJS, Redux, and unit testing with Jest & Enzyme:

Dinesh Suggestion:

Create a session so that we can create a pixcel perfect react application

**03rd, Oct  - Day 1: Introduction to ReactJS with Hooks**

**04th, Oct - Day 2: Advanced React Hooks**

**05th, Oct - Break for revision**

**06th, Oct - Day 3: React Router with Hooks**

**09th, Oct - Day 4: Introduction to Redux with Hooks**

**10th, Oct - Break for revision**

**11th, Oct - Day 5: Advanced Redux with Redux Toolkit & Async Operations**

**12th, Oct - Day 6: Further into Redux & Middleware**

**13th, Oct - Break for revision**

**16th, Oct - Day 7: Introduction to Jest & Enzyme for Functional Components – Unit Test**

**17th, Oct - Day 8: Testing Redux Logic**

**18th, Oct - Day 9: Deep Dive into Testing React Components**

**19th, Oct - Break for revision**

**20th, Oct - Day 10: Wrap Up & Best Practices**

Each day a mix of theoretical sessions, hands-on coding, and assignments.

By the end of the training, the participants should feel adept at building and testing applications using React, Redux with hooks, Jest, and Enzyme.

**Training plan for ReactJS, Redux, and unit testing with Jest & Enzyme:**

Duration: 16-days

<https://react.dev/>

<https://reactrouter.com/en/main>

Here's a 16-day training plan focusing exclusively on functional components based on hooks, for ReactJS, Redux, and unit testing with Jest & Enzyme:

### **Day 1: Introduction to ReactJS with Hooks**

* **What is React and why use it?**
* **Setting up a new React project** with Vite
* **Understanding JSX syntax**
* **Functional Components**
* **State with useState**
* **Effects with useEffect**
* **Assignment**: Create a simple counter application using functional components and hooks

### **Day 2: Advanced React Hooks**

* **Context with useContext**
* **Ref with useRef**
* **Callback with useCallback**
* **Memoization with useMemo**
* **Custom Hooks**
* **Assignment**: Enhance the counter application, add a feature to store counter history.

### **Day 3: React Router with Hooks**

* **What is React Router?**
* **Setting up React Router** in a project
* **useParams, useHistory, and useLocation hooks**
* **Route protection and authentication**
* **Assignment**: Add routes to the counter app. Each counter has its own route.

### **Day 4: Introduction to Redux with Hooks**

* **Why use Redux?**
* **Core concepts**: Actions, Reducers, Store
* **Setting up Redux Toolkit**
* **Redux Hooks**: useSelector, useDispatch, useStore
* **Assignment**: Create a simple to-do app using React and Redux with hooks.

### **Day 5: Advanced Redux with Redux Toolkit & Async Operations**

* **Slice concept** in Redux Toolkit
* **createAsyncThunk** for async actions
* **ExtraReducers** to handle async states
* **Assignment**: Enhance the to-do app to fetch data from an API using Redux Toolkit.

### **Day 6: Further into Redux & Middleware**

* **Middleware in Redux**
* **React-Redux hooks**: useStore, useActions
* **Normalization of state**
* **Assignment**: Implement a feature in the to-do app where tasks have categories, and they can be filtered.

### **Day 7: Introduction to Jest & Enzyme for Functional Components – Unit Test**

* **What is Jest** and why use it?
* **Setting up Jest** with Create React App
* **Introduction to Enzyme**
* **Testing functional components** with hooks
* **Assignment**: Write tests for the counter app components.

### **Day 8: Testing Redux Logic**

* **Testing Redux reducers and actions**
* **Mocking modules using Jest**
* **Testing async actions and middleware**
* **Assignment**: Write tests for the to-do app actions and reducers.

### **Day 9: Deep Dive into Testing React Components**

* **Shallow vs. full rendering in Enzyme**
* **Testing component interactions** and effects
* **Testing custom hooks**
* **Assignment**: Write tests for the to-do app components and its interactions.

### **Day 10: Wrap Up & Best Practices**

* **Performance optimization** in React and Redux with hooks.
* **Best practices**: code splitting, lazy loading.
* **Jest & Enzyme best practices**.
* **Final project**: Create a simple e-commerce app using the concepts learned and write tests for it.

Each day a mix of theoretical sessions, hands-on coding, and assignments.

By the end of the training, the participants should feel adept at building and testing applications using React, Redux with hooks, Jest, and Enzyme.