REACT JS – COURSE OUTLINE

Since we have a shorter timeframe, here's a learning plan divided into four intensive sessions, one per week. Each session is designed to cover a broad range of concepts within React.js

Session 1: Introduction to React.js

- What is React.js? Discussing Components as its building blocks

- node package manager (npm) and creating your first React app using create-react-app (CRA)

- learning about JSX, getting familiar with the syntax and its role in React

- Create a basic app and render basic components

- Explore component's state and its importance.

- Use the useState hook to manage component's state.

- Understand component props and how to pass data between components.

Build a simple interactive UI element.

Session 2: Advanced Components and State Management

- Dive into component lifecycle

- What are hooks in React? Handling side effects with the useEffect hook.

- Understand how to perform tasks such as data fetching and event subscriptions.

- Handle component cleanup with useEffect. How to un-subscribe to events before unmounting? Why is it necessary?

- Learn about controlled components and forms in React.

- Handle form input changes and submissions. What is default behavior of forms? How to avoid in React and why? Discuss Single-Page Applications (SPA).

- Discussing form validation.

- Explore component composition and reusability.

- Implement higher-order components (HOCs) and render props.

- Class Task/Take-home assignment: Build a mini project utilizing the concepts learned so far. Could be performed in teams as well.

Session 3: Routing and API Integration

- Learn about client-side routing using react-router.

- Set up routes, navigation, and nested routes.

- Create a multi-page React app.

- Introduction to asynchronous operations and data fetching.

- Use the fetch API to retrieve data from a remote server.

- Handle loading states and error responses.

- Dive into more advanced data fetching using axios.

- Explore authentication and authorization in API calls.

Session 4: Advanced Topics and Project Work

- Q/A's regarding previous sessions

- Understanding the basics of state management using the context API.

- Implement a global state provider and consumer.

- Learn about state management libraries like Redux (optional)

- Explore performance optimization techniques in React (optional)

By the end of these four intensive sessions, students will have gained a solid grasp of React.js and be equipped to build dynamic and interactive web applications. One thing to keep in mind is that this plan is quite condensed, so students should be prepared to spend significant time practicing and experimenting with the concepts learned at class.