

## Lecture Summary – Week 11

React JS is a JavaScript library that is used to build application interfaces. Most of the modern-day websites have pages that make use of the same functionality. React JS allows reusability of these features (called components) by enabling these components to be embedded wherever required. It also allows and manages the state of an application. React JS uses a virtual Document Object Model to fetch, render and modify these components. As a result, the resultant webpage loads faster and creates a better user experience. Components are the building blocks of user interface. It is a piece of code that specifies a specific part of the UI. This can be reused across not only multiple pages, but across applications. In React, there are two types of components: Functional components, Class components.

The functional components are the JavaScript functions that are usually created to show or present the data in the front-end. These functions do not have their own state. On the other hand, the class components are JavaScript classes that extend the `React.Component` class. They have their own state and can contain other components or functional components. Many times, these components are used to handle events.

The `useEffect` and `useState` are two commonly used hooks in React. Using the `useState` hook, we may include state in your functional components. It accepts an initial value as an input and outputs an array with the state's current value and an update function. React will automatically re-render our component with the modified state if we use this function to update the state. We use the `useEffect` hook to implement side effects in your functional components. Every time the component renders, it accepts an argument and calls that function. We can use it to carry out operations like obtaining data, changing the page's title, or establishing event listeners. The React Router library includes a hook called `useParams` that enables you to access parameters from the URL in your components. `useParams` can be used to obtain a parameter's value, such as `id`, when a URL contains parameters (sometimes referred to as route parameters), such as `/users/:id`.

By storing the outcomes of expensive calculations in a cache so they can be reused later, memoization is a technique that can increase performance. React's server-side rendering feature enables you to render your app more quickly, more easily, and with better SEO. By minimizing the amount of data that needs to be loaded in advance, lazy loading is a strategy that enables you to load components or data only when they are required. It also offers a robust ecosystem of third-party libraries and plugins that may assist developers in quickly and simply adding features to their apps. The Flexbox layout paradigm is the foundation for React Native, which employs the same layout engine as React for the web. Overall, React Native is a powerful framework for building high-quality native mobile apps using JavaScript and React.