MODULE-2 TW-04 TEAM LEAD VERSION

TEAMWORK

CLARUSWAY
WAY TO REINVENT YOURSELF

Meeting Agenda

▶ Icebreaking

▶ Workshop Activities - Tuesday

▶ Teamwork Activities - Friday

   ▶ Questions

   ▶ Interview Questions

▶ Video of the week

▶ Retro meeting

▶ Case study / project

Teamwork Schedule

Ice-breaking   10m

- Personal Questions (Stay at home & Corona, Study Environment, Kids etc.)
- Any challenges (Classes, Coding, studying, etc.)
- Ask how they're studying, give personal advice.
- Remind that practice makes perfect.

Workshop Activities (Tuesday)   90m

- Project-006 : Sass Portfolio with React Router

Team Work Activities (Friday)   90m

Ask Questions   20m

1. Why should you use React Router's Link component instead of a basic `<a>` tag in React?

A. The link component allows the user to use the browser's `Back` button.
B. There is no difference--the `Link` component is just another name for the `<a>` tag.
C. The `<a>` tag will cause an error when used in React.
D. The `<a>` tag triggers a full page reload, while the `Link` component does not.

Answer: D

2. Consider the following code from React Router. What do you call :id in the path prop?

```
<Route path="/:id" />
```

A. This is a route modal
B. This is a route parameter
C. This is a route splitter
D. This is a route link

Answer: B

3. What is React Router used for?

A. Managing the state of a React application
B. Building a database for a React application
C. Handling routing in a React application
D. Loading a React Page

Answer: C

4. React Router is a library that provides _____ in a single-page application.

A. Routing
B. State management
C. Component rendering
D. Data fetching

Answer: A

5. React Router uses a _____ component to render a component when its path matches the current URL.

A. Link
B. Route
C. Switch
D. Redirect

Answer: B

6. The component in React Router is used to create _____ between routes.

A. Navigation
B. State management
C. Data sharing
D. Component rendering

Answer: A

7. Which React Router hook provides access to the current location object?

A. useHistory
B. useLocation
C. useRouteMatch
D. useParams

Answer: B

8. How do you handle a 404 error in React Router?

A. By creating a new component for the 404 error page and rendering it in a <Route> component with no path specified
B. By using the <Error404> component provided by React Router
C. By using the <NotFound> component provided by React Router
D. By creating a new route with a path of '*' and rendering a 404 error page component in it

Answer: D

9. What's the purpose of the BrowserRouter component?

A. To create new routes
B. To understand all of its children Route elements, and intelligently choose which ones are the best to render.
C. Allows React Router to pass the app's routing information down to any child component it needs (via context)
D. To make navigating possible in native applications

Answer: C

10. The useParams hook in React Router allows you to access URL parameters within a _____ component.

A. Class
B. Functional
C. Higher-order
D. Stateful

Answer: B

11. React Router allows you to perform client-side routing by manipulating the _____.

A. Server
B. Database
C. Browser history
D. Context API

Answer: C

12. Which of the following React Context methods is used to consume context values?

A. useContext
B. useState
C. createContext
D. useEffect

Answer: A

13. _____ is a technique of delivering data to a deeply nested component by passing it through a number of intermediate children components.

A. Context API
B. Prop drilling
C. Lifting State Up
D. Fault isolation

Answer: B

14. How do you handle passing through the component tree without having to pass props down manually at every level?

A. React Send
B. React Pinpoint
C. React Router
D. React Context

Answer: D

Interview Questions   15m

1. What is React Router?
React Router is a library that helps developers manage and navigate different routes in their React applications. React Router makes it easy to declare and render different components for each route, and also keeps your application's state updated as you navigate between routes.

2. How does browser history work in React Router?
React Router uses browser history in order to keep track of the current location within the app. This allows the user to use the back and forward buttons in their browser in order to navigate through the app.

3. What is nested routing?
Nested routing is when you have routes within other routes. This can be useful if you want to create a hierarchy of pages, for example if you have a blog with different categories and you want each category to have its own page with a list of posts. To do this, you would create a route for each category, and then nest the route for the posts within that.

4. What do you understand about the 'to' attribute in React Router?
The to attribute is used to specify the route that the user should be redirected to when they click on a link. The value of the 'to' attribute can either be a string, which will be interpreted as a relative path, or an object, which can specify the absolute path, query parameters, and so on.

5. Explain the difference between server-side rendering and client-side rendering in React.

In a React application, there are two main ways to render the components: server-side rendering (SSR) and client-side rendering (CSR).

Server-side rendering (SSR) is when the initial render of a React application is done on the server. The server generates the HTML for the initial state of the application and sends it to the browser. When the JavaScript bundle loads, React takes over and the application continues to function as a SPA (Single-Page Application) on the client side.

This approach has a few benefits such as:

- Improved performance for search engines and users on slow connections

- Faster time-to-first-byte

- Better accessibility for users who have JavaScript disabled

Client-side rendering (CSR) is when the React application is rendered entirely in the browser, using JavaScript. The browser requests the JavaScript bundle from the server and then renders the components on the client side. This approach has the benefit of faster load times for users on fast connections and a more responsive user interface.

In general, CSR is the simpler option to implement and more popular, but SSR is a good choice for certain use cases, such as when SEO is a primary concern, or when the app is targeting users on slow internet connections.

It is also worth noting that, it is possible to have a hybrid approach between SSR and CSR which is called isomorphic or universal rendering. This approach allows to leverage the benefits of both SSR and CSR.

6. Explain the concept of a Context in React.

In React, context is a way to share data that is considered "global" for a component tree. It allows you to pass data through the component tree without having to pass props down manually at every level.

A component that needs to access the context data can consume it by using the useContext hook or the Consumer component. To make the context available to a component, a parent component needs to provide it using the Provider component.

Context is often used for data that is required by many components in an application, such as the currently authenticated user, the current locale, or the theme.

It should be noted that context should be used sparingly, as it can make your components more difficult to reason about and test. If possible, it's better to pass props down the component tree manually.

☕

Coffee Break   10m

☕

Retro Meeting on a personal and team level   5m

Ask the questions below:

- What went well?
- What went wrong?
- What is the improvement areas?

Case study/Project   15m

Project will be announced during the week

Closing   5m

- Next week's plan
- QA Session