

Webdev - Módulo 3

Roteiro de aula

Aula 8 - Hard: Na Fotossíntese



Tópicos da aula:

- Virtual DOM
- Ciclo de vida dos Componentes
- Revisão de axios



Objetivos da aula:

1. Conhecer o virtual DOM
2. Utilizar a documentação como base dos estudos
3. Compreender o ciclo de vida de componentes e saber diferenciar componentes estruturais de componentes funcionais
4. Rever os conceitos de axios



Atividades da Aula

→ Atividade de checkin: [Dicionário de Nomes Próprios](#)

→ Atividade: Mergulhando na documentação:

- ◆ [Virtual dom e elementos internos](#)
- ◆ Tempo: 15 minutos

→ Atividade: Analisando ciclos:

- ◆ Acesse o projeto disponibilizado no [replit](#) e análise o App.jsx:
 - constructor()
 - componentDidMount()
 - componentWillUnmount()
 - render()
 - componentDidUpdate()
- ◆ Tempo: 25 minutos



Momento Aprendizagem em Pares

Esse momento é dedicado para vocês desenvolverem suas demandas e entregas para o curso.

- Utilize esse tempo da maneira que preferir, mas atente-se às aulas que você deve realizar as entregas.
 - ◆ **Dica:** Nas Propostas dos projetos, vocês encontram uma sugestão de organização para a realização das atividades.
 - ◆ **Lembre-se:** O momento de Aprendizagem em Pares é justamente para fazer trocas e aprender em comunidade, aproveite seus colegas e se desenvolvam juntos!
- **Entregas:**
 - ◆ Projeto individual: aula 5 - HARD
 - ◆ Projeto em grupo: aula 10 - HARD
 - ◆ Apresentação do projeto: aula 10 - HARD



Revisão da aula

- Normalmente, sempre que uma página da Web é solicitada por um usuário, o navegador recebe um documento HTML para essa página do servidor. O navegador então constrói uma **estrutura lógica semelhante a uma árvore a partir do HTML para que o usuário veja a página solicitada no cliente**. Essa estrutura semelhante a uma árvore é chamada de **Document Object Model, também conhecido como DOM**. É uma representação estrutural do documento da web — neste caso, um documento HTML — como nós e objetos. O DOM serve como uma interface para o documento da Web para que o JavaScript e outras linguagens de script possam acessar, interagir programaticamente e manipular o conteúdo do documento.
- A **documentação de software** fornece informações sobre um programa de software para todos os envolvidos em sua criação, implantação e uso. A documentação orienta e registra o processo de desenvolvimento. Ela também auxilia em tarefas básicas, como instalação e solução de problemas. A documentação do software pode ser um documento vivo que é atualizado ao longo do ciclo de vida do desenvolvimento do software. Seu uso e a comunicação que ele incentiva com os usuários fornecem aos desenvolvedores informações sobre os problemas que os usuários têm com o software e quais

recursos adicionais eles precisam. Os desenvolvedores podem responder com atualizações de software, melhorando a satisfação do cliente e a experiência do usuário.

→ Cada componente React tem um **ciclo de vida** próprio, o ciclo de vida de um componente pode ser definido como **a série de métodos que são invocados em diferentes estágios da existência do componente**. A definição é bastante direta, mas o que queremos dizer com estágios diferentes? Um React Component pode passar por quatro estágios de sua vida, como segue.

- **Inicialização:** Este é o estágio em que o componente é construído com os Props fornecidos e o estado padrão. Isso é feito no construtor de uma Classe de Componente.
- **Montagem:** A montagem é a etapa de renderização do JSX retornado pelo próprio método render.
- **Atualização:** A atualização é a etapa em que o estado de um componente é atualizado e o aplicativo é redesenhado.
- **Desmontagem:** como o nome sugere, a desmontagem é a etapa final do ciclo de vida do componente em que o componente é removido da página

→ **Axios** é uma biblioteca Javascript usada para fazer solicitações HTTP de node.js ou XMLHttpRequests do navegador que também oferece suporte à ES6 Promise API.



Para ajudar



Links interessantes:

- [Virtual DOM and Internals – React \(reactjs.org\)](https://reactjs.org/docs/virtual-dom-internals.html)
- [ReactJS - Component Life Cycle \(tutorialspoint.com\)](https://www.tutorialspoint.com/reactjs/reactjs_component_life_cycle.htm)
- [POST Requests | Axios Docs \(axios-http.com\)](https://axios-http.com/docs/post-requests)
- [Como usar o Axios com o React: o guia definitivo \(2021\) \(freecodecamp.org\)](https://www.freecodecamp.org/news/using-axios-with-react-the-definitive-guide-2021/)