

# WebDev - Módulo 3

## Roteiro de aula

---

### Aula 5 - Hard: Ancorados



#### Tópicos da aula:

- Virtual DOM
- Ciclo de Vida dos Componentes
- Hooks
- useState
- axios(get, post)



#### Objetivos da aula:

1. Relatar noções básicas sobre hooks
2. Apresentar o projeto que será desenvolvido em aula
3. Introduzir a biblioteca Axios
4. Relembrar e introduzir de forma prática o verbo GET no Axios
5. Relembrar e introduzir de forma prática o verbo POST no Axios



#### Atividades da Aula

→ Checkin da aula: Link do vídeo <https://www.youtube.com/watch?v=ILDvhChQzEk>

→ TODO: Criação de estado

◆ Crie o componente NovaTarefa.jsx

◆ Dentro dele iremos:

- Criar um estado para receber o valor do input de título
- Usar o evento de change para capturar esse valor
- A cada alteração do valor do input, ele deve ser salvo no estado

◆ Dica: usar o atributo onChange

→ TODO: GET tarefas

◆ Crie um componente Tarefas.jsx

◆ Dentro dele:

- Execute um GET ao nosso Json-server (criado anteriormente);
- Salvar as tarefas em um estado;

- Mostrar os títulos das tarefas na tela.
- ◆ Dica: Para mostrar na tela utilize o `map()`



## 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

- **React Hooks** são **funções JavaScript simples que podemos usar para isolar a parte reutilizável de um componente funcional**. **Hooks** é o nome dado ao novo recurso introduzido na versão React 16.8. Ele permite que você use o estado e outros recursos do React sem escrever uma classe. **Ganchos** são as funções que se "engancham" no estado do React e nos recursos do ciclo de vida dos componentes de função. Não funcionam dentro das classes. Os ganchos são compatíveis com versões anteriores, o que significa que não contém nenhuma alteração significativa. Além disso, não substitui seu conhecimento dos conceitos do React.
- **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. Assim, concluímos que ele faz algo que já podemos fazer e que recentemente foi significativamente melhorado. Na raiz da pasta do seu projeto, abra o terminal, execute `npm install axios` e está tudo pronto. Este trecho de código envia uma solicitação GET para a API JSON. Como a solicitação retorna uma promessa, você usará o bloco `.then()` para manipular as respostas. Você

também precisará usar o método `.catch()` para registrar quaisquer erros no console. Adicione o trecho de código anterior à função `getUser()` no arquivo `users.js` e salve-o. Em seguida, vá para o navegador e clique no botão OBTER. Abaixo dele, um novo conteúdo deve aparecer, exibindo os detalhes da resposta.

- A **solicitação POST do Axios** usa um objeto após a URL da solicitação para definir as propriedades que você deseja criar para seu usuário. Uma vez concluída a operação, haverá uma resposta do servidor. Para verificar se seu código está funcionando, volte para a janela do navegador do aplicativo e clique no botão POST. Este snippet faz parte da função `postUser()` no arquivo `users.js`.



### Para ajudar



#### Links interessantes:

- [Virtual DOM and Internals – React \(reactjs.org\)](https://reactjs.org/)
- [ReactJS - Component Life Cycle \(tutorialspoint.com\)](https://tutorialspoint.com/)
- [POST Requests | Axios Docs \(axios-http.com\)](https://axios-http.com/)
- [Como usar o Axios com o React: o guia definitivo \(2021\) \(freecodecamp.org\)](https://freecodecamp.org/)