

# WebDev - Módulo 3

## Roteiro de aula

---

### Aula 4 - Hard: Luz, Câmera, React!



#### Tópicos da aula:

- React básico;
- Instalação com Create React App;
- Instalação com Vite;
- JSX;
- Componentes;
- Props.



#### Objetivos da aula:

1. Aprender as noções básicas da biblioteca React.js;
2. Compreender os possíveis meios de instalação com o CRA e Vite;
3. Conhecer a sintaxe do JSX;
4. Conhecer as noções básicas de componentes;
5. Conhecer as noções básicas de props;



#### Atividades da Aula

→ **Experiência de Aprendizagem [Create react App, Babel, Webpack]** : link para acesso à documentação sugerida:

[https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Strict\\_mode](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Strict_mode)

→ **Atividade 1: Criar componente de formulário**

- Em grupos de 3 a 4 pessoas, vocês terão 20 min para criar um componente de formulário. Esse componente deverá conter 2 inputs, um para nome e outro para e-mail. Além disso, deve conter um botão para envio das informações e você deve chamá-lo no App.jsx
  - ◆ Criar uma pasta components na pasta src
  - ◆ Criar um arquivo Form.jsx
  - ◆ Esse arquivo terá:
    - 2 inputs (nome e e-mail)

- Uma área de texto
  - Um botão para envio do formulário
- ◆ O componente deve ser chamado no componente App.js

## → Atividade 2: Criar componente de botão

Em grupos de 4 a 5 pessoas, vocês terão 15 minutos para cada um desenvolver, individualmente, um componente de botão no projeto de vocês. Ele deverá ter uma prop chamada color e a cor deve ser recebida como uma palavra, como por exemplo 'vermelho' deve renderizar um botão na cor vermelho.

### Criar um componente de botão

- ◆ Dentro da pasta components criar um arquivo Button.jsx
- ◆ Criar um objeto com 10 cores escolhidas pelo grupo
- ◆ O componente Button terá uma prop chamada color
- ◆ A cor de fundo (background color) deve ser alterada de acordo com o valor da prop color
- ◆ Substituir o botão do componente Form pelo novo componente Button



## 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** é uma biblioteca para ajudar os desenvolvedores a construir interfaces de usuário (UIs) como uma árvore de pequenos pedaços, chamados componentes. Um **componente** é uma mistura de HTML e JavaScript que captura toda a lógica necessária para exibir uma pequena seção de uma interface do usuário maior. Cada um desses componentes pode ser construído em partes sucessivamente

complexas de um aplicativo. O React oferece várias extensões para todo o suporte arquitetônico do aplicativo, como Flux e React Native, além da mera interface do usuário.

- O **Create React App (CRA)** tem sido a **ferramenta** preferida para muitos desenvolvedores de todos os níveis de habilidade **quando se trata de construir um aplicativo React (iniciantes, intermediários e até especialistas)**. No entanto, tem algumas desvantagens significativas, que são velocidade e desempenho.
- **Vite** é a palavra francesa para rápido. É uma **ferramenta de construção moderna para desenvolvimento web front-end que utiliza módulos ES, ou ESM**. O Vite apresenta um empacotador mais enxuto e rápido que vem com um servidor de desenvolvimento pré-configurado. O Vite usa recursos HMR para acompanhar as alterações em seu aplicativo sem recarregar a página inteira. Com a API HMR, o navegador carregará apenas a seção modificada da página e ainda reterá o estado do aplicativo. Você não precisa configurá-los manualmente - quando você cria um aplicativo por meio do create-vite, os modelos selecionados já os têm pré-configurados para você.
- O **JavaScript XML / Extension** é chamado JSX e é uma extensão de sintaxe para JavaScript. Recomendamos usar JSX com o React para descrever como a UI deveria parecer. JSX pode lembrar uma linguagem de template, mas que vem com todo o poder do JavaScript.
- Ao invés de separar *tecnologias* artificialmente colocando markup e lógica em arquivos separados, o React separa *conceitos* com unidades pouco acopladas chamadas "componentes" que contém ambos.
- **Props** significa simplesmente propriedades. São eles que **tornam os componentes reutilizáveis**. Porque eles executam uma função essencial - eles passam dados de um componente para outro.



## Para ajudar



### Links interessantes:

- [Criando um app React usando vite](#)
- [Guia - Vite](#)

- [Vite - Funcionalidades](#)
- [Componentes e Props - React](#)
- [Introduzindo JSX](#)