



Aula 05.1:

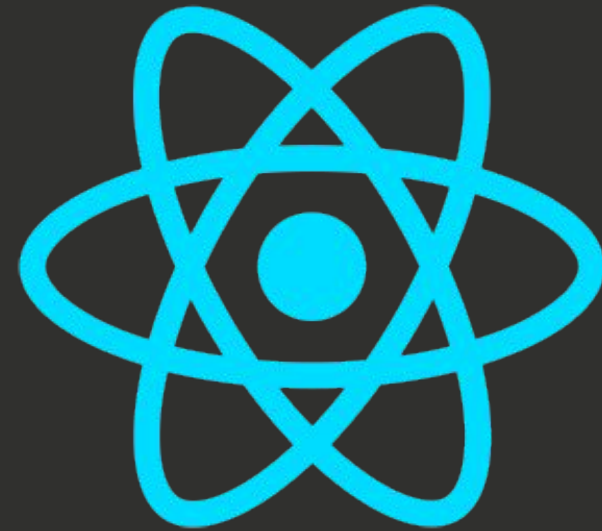
Introdução ao React
com Next.js

Introdução ao React JS

O que é? React é uma biblioteca JavaScript desenvolvida pelo Facebook para construir interfaces de usuário (UIs) interativas e reutilizáveis.

Por que usar? Facilita a criação de componentes de UI que podem ser reutilizados em diferentes partes da aplicação, melhorando a manutenção do código.

Vantagens: Performance eficiente através do Virtual DOM, uma comunidade ativa, e um grande ecossistema de ferramentas e bibliotecas.



React JS



Características principais do React



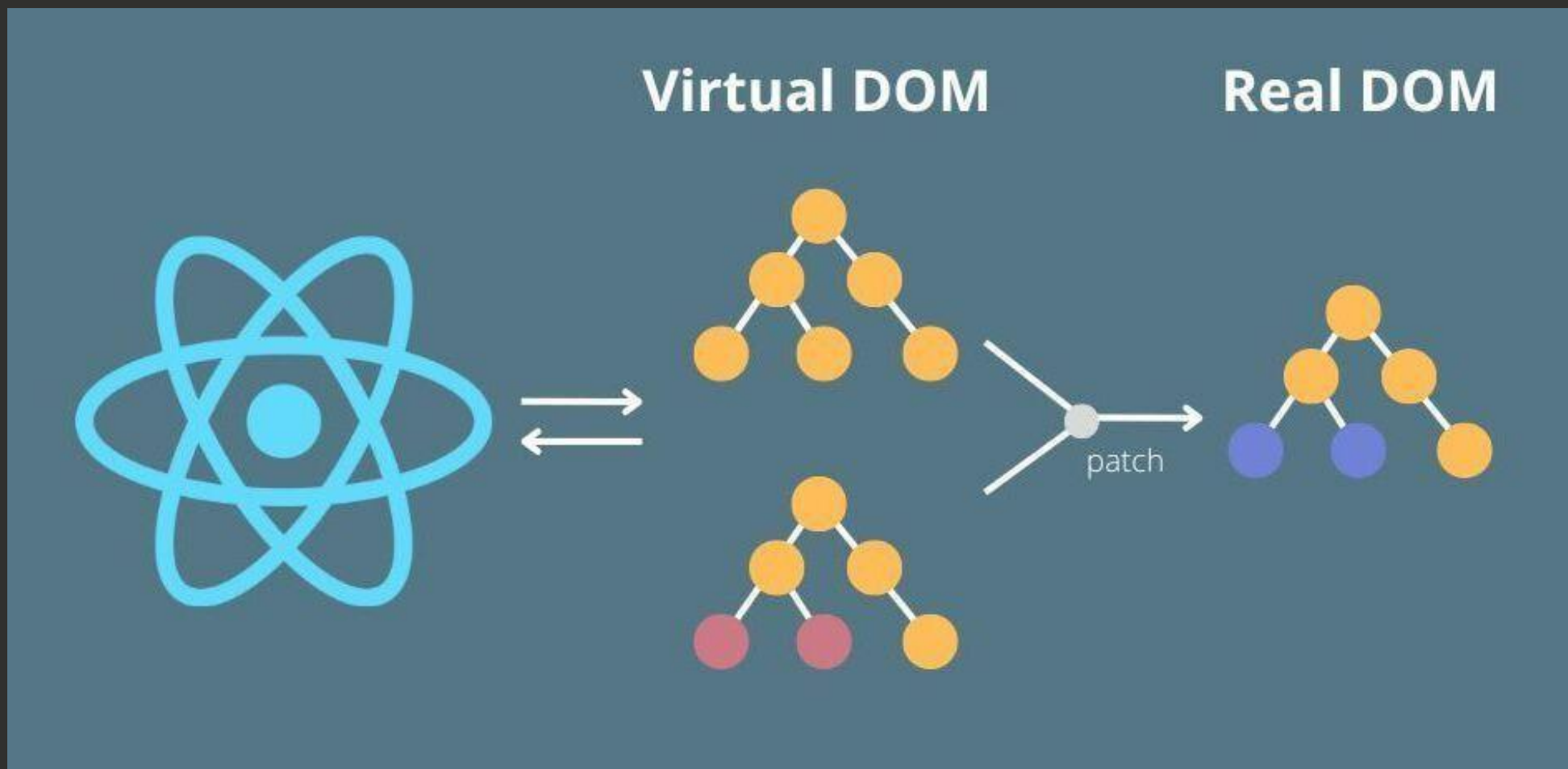
Componentes: Tudo em React é baseado em componentes, que são blocos independentes e reutilizáveis da UI.

DOM Virtual: React utiliza o conceito de DOM Virtual para atualizar a UI de forma eficiente, minimizando o número de operações diretas no DOM real e aumentando o desempenho.

JSX: Uma sintaxe semelhante ao HTML, usada dentro do JavaScript para descrever como a interface deve ser renderizada.



Virtual DOM (Document Object Model)



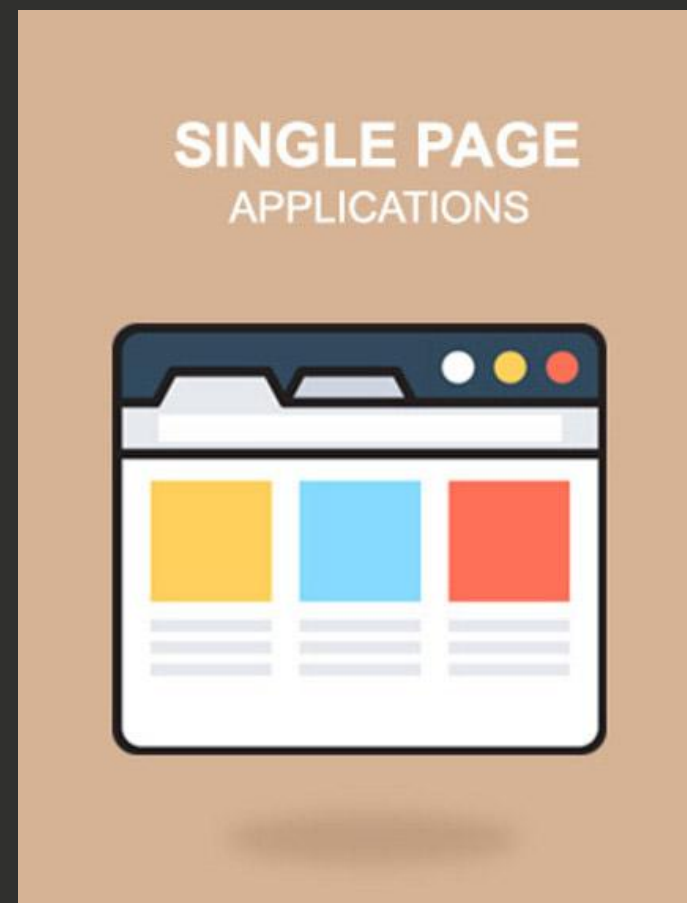
Vantagens do React

- Alta performance devido ao uso do DOM Virtual.
- Reutilização de componentes, o que facilita a manutenção e escalabilidade de projetos.
- Comunidade ativa e vasta disponibilidade de bibliotecas e ferramentas.
- Integração fácil com outras bibliotecas e frameworks JavaScript.

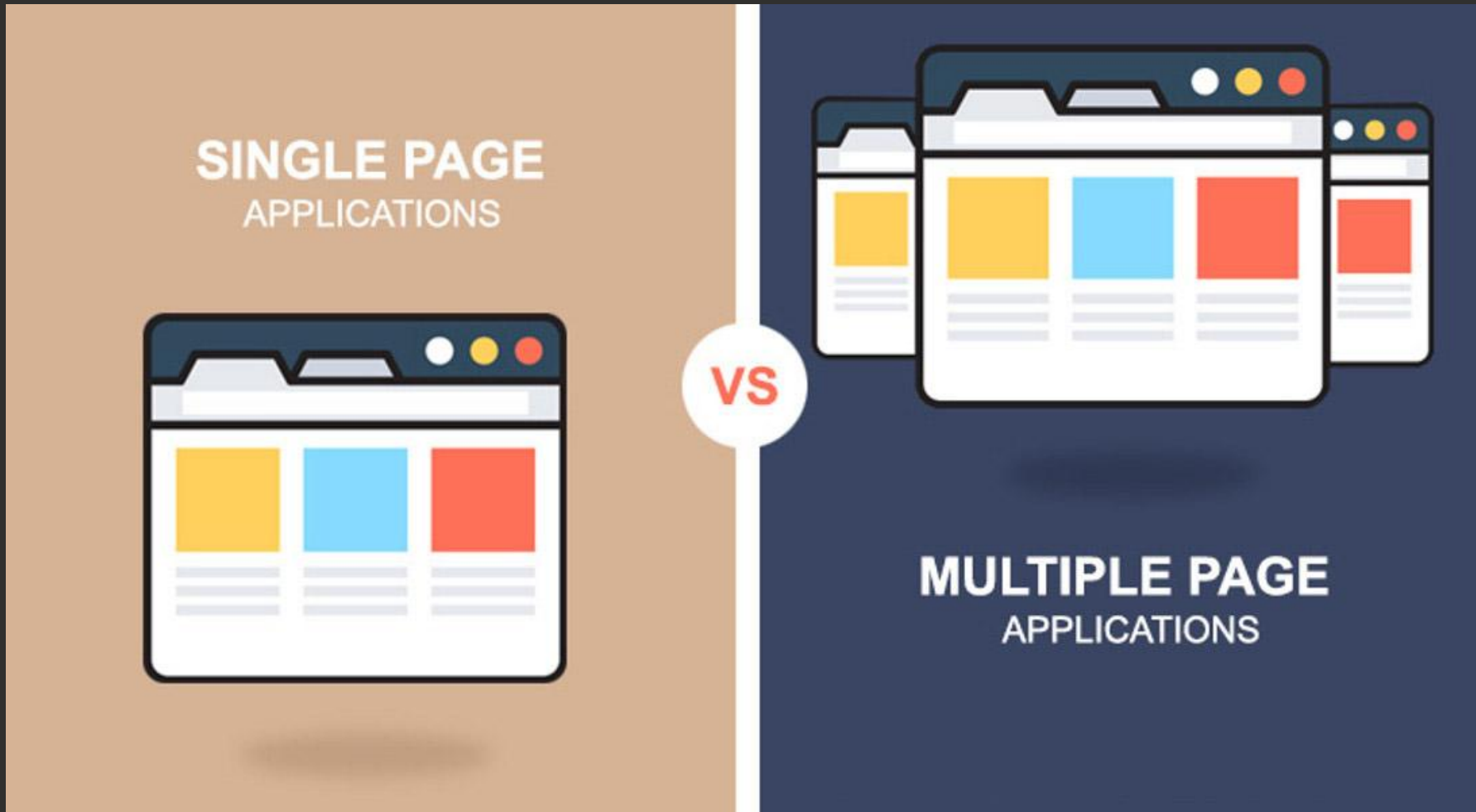


Quando usar React?

- Aplicações de página única (SPA) onde a experiência do usuário é crucial.
- Projetos que precisam de alto desempenho e interatividade.
- Quando há necessidade de reutilizar componentes em várias partes da aplicação.



Single Page Application

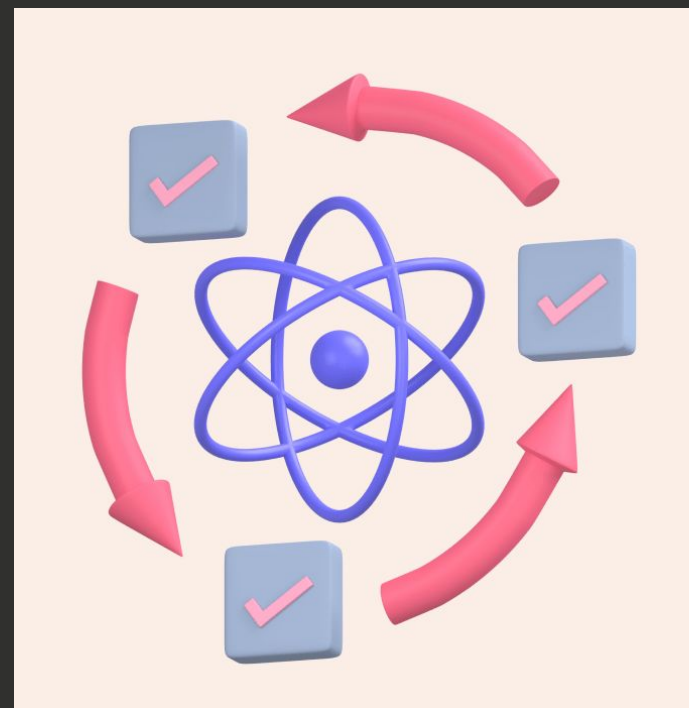


Estados no React



O estado em React representa dados que podem mudar ao longo do tempo, em resposta a interações do usuário ou a mudanças no ambiente (como dados vindos de uma API).

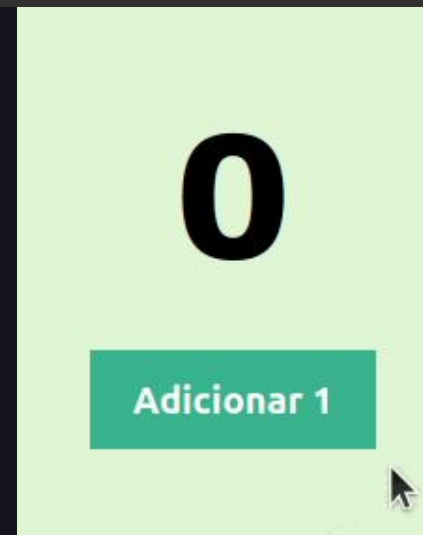
Cada componente em React pode ter seu próprio estado, o que o torna independente e **reativo**.



Exemplo de Estado em um componente



```
1  import React, { useState } from 'react';
2
3  function Contador() {
4    // Inicialização do estado 'count' com valor 0
5    const [count, setCount] = useState(0);
6
7    return (
8      <div>
9        <p>Você clicou {count} vezes</p>
10       <button onClick={() => setCount(count + 1)}>Clique aqui</button>
11     </div>
12   );
13 }
```



Framework Next.js

Next.js é um framework de desenvolvimento React que permite a criação de aplicativos React com funcionalidades adicionais, como renderização do lado do servidor (**SSR**) e geração estática de páginas (**SSG**).

Criado pela **Vercel**, ele combina a simplicidade do React com funcionalidades avançadas que facilitam o desenvolvimento de aplicações web modernas.



Características principais do Next.js



Renderização do Lado do Servidor (SSR):

Ao contrário do React puro, que renderiza a aplicação no navegador, o Next.js pode renderizar páginas no servidor antes de enviá-las para o cliente, melhorando o tempo de carregamento inicial e o SEO.

Geração Estática (SSG):

Gera páginas HTML estáticas no momento da construção do projeto, o que pode aumentar significativamente a performance para conteúdos que não mudam com frequência.



Características principais do Next.js



Roteamento Automático:

Next.js tem um sistema de roteamento baseado em arquivos, eliminando a necessidade de bibliotecas externas como o React Router.

Suporte a API Routes:

Permite criar rotas de API diretamente no projeto, facilitando o desenvolvimento de back-ends simples sem precisar de um servidor externo.



Vantagens do Next.js



- **Otimização para SEO (Search Engine Optimization):**

A renderização do lado do servidor permite que as páginas sejam indexadas mais facilmente pelos motores de busca.

- **Tempo de carregamento inicial mais rápido:**

Com SSR e SSG, o tempo que o usuário espera para ver o conteúdo pode ser drasticamente reduzido.



Vantagens do Next.js

- **Suporte a CSS e Sass nativo:**

Facilita a estilização, sem necessidade de configurações complexas.

- **Deploy simplificado:**

Pode ser integrado facilmente a serviços de hospedagem, como o Vercel, para implementação contínua.



Quando usar Next.js?

- Em projetos que necessitam de uma boa performance inicial e SEO, como blogs ou e-commerce.
- Quando há necessidade de páginas estáticas e dinâmicas ao mesmo tempo.
- Aplicações que demandam SSR para otimizar o desempenho ou gerar conteúdo dinâmico.



React vs Next.js: Comparação

Característica	React	Next.js
Renderização	Somente no lado do cliente	Suporte a SSR e SSG
Roteamento	Necessita de bibliotecas externas (React Router)	Roteamento automático com base em arquivos
SEO	Menor suporte para SEO devido ao carregamento no cliente	Melhor otimização para SEO devido à renderização no servidor
Complexidade	Mais flexível, mas requer mais configurações	Opiniado e com configurações prontas, facilitando o início
Deploy	Depende da configuração	Integração nativa com Vercel e fácil deploy



Conclusão

- React é excelente para criar aplicações front-end dinâmicas e interativas, especialmente quando se deseja flexibilidade e modularidade.
- Next.js aproveita o poder do React, adicionando funcionalidades avançadas como SSR e geração estática, que são ideais para melhorar o desempenho e o SEO.
- Ambas as tecnologias são fundamentais para o desenvolvimento web-moderno e se complementam para criar experiências de usuário eficientes e otimizadas.





Aula 05.1:

Introdução ao React
com Next.js