



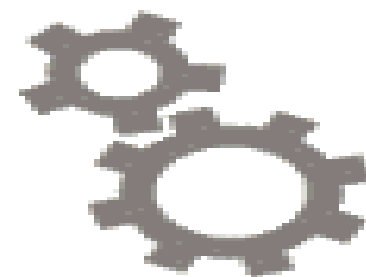
**Curso:** Desenvolvimento de Software Multiplataforma

**Disciplina:** Programação para Dispositivos Móveis I

**Professor:** André Olímpio

# Programação Para Dispositivos Móveis I

Loading...



# React Native

- O React Native (RN) é um framework de desenvolvimento de aplicativos móveis criado pelo Facebook.
- Permite a criação de aplicativos nativos para iOS e Android utilizando JavaScript (JS) e React.

**Learn once, Write anywhere**



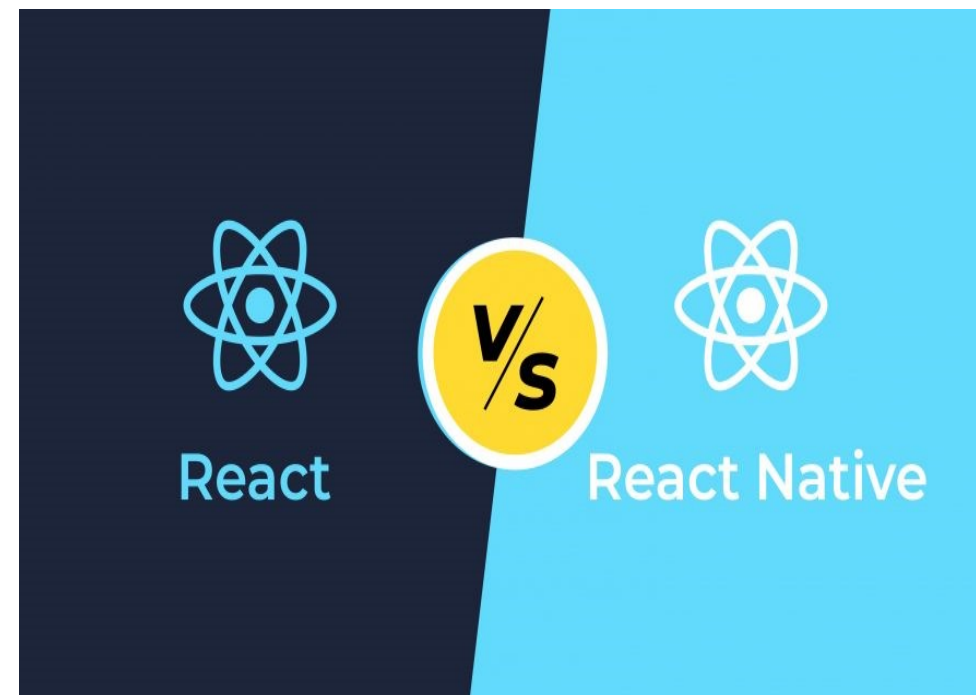
Fonte: Medium.com

# React JS X React Native

- **React JS:** É uma biblioteca baseada em JavaScript que facilita o desenvolvimento de interfaces para aplicações tanto web quanto mobile.
- Permitir criar diversos módulos da interface independente da resposta do navegador, garantindo uma renderização do código mais rápido, uma vez que o JSX (JavaScript e HTML) realiza esse procedimento.
- Ao codificar a interface, a renderização automática, gerando um código mais leve e menor.

# React JS X React Native

- **React Native:** é uma biblioteca baseada no React JS voltada para dispositivos *mobile*.
- Essa biblioteca podemos desenvolver interfaces tanto para Android quanto para iOS.
- Assim como o React JS utiliza o JSX (similiar ao XML) e DOM (Document Object Model).



Fonte: Greelow.com

# Características do React Native

- **Código único para múltiplas plataformas:**
  - ✓ O RN nos permite escrever a maior parte do código em JS, que é compartilhado entre as plataformas iOS e Android.
  - ✓ Isso reduz significativamente o esforço de desenvolvimento e manutenção.
- **Componentes nativos:**
  - ✓ São mapeados para widgets nativos, o que significa que os aplicativos têm a aparência e o desempenho de aplicativos nativos.
  - ✓ Isso inclui componentes de UI (User Interface), como botões, textos e listas.

# Características do React Native

- **Desenvolvimento rápido:**

- ✓ Recursos como recarga em tempo real (hot reloading) e ferramentas de depuração integradas, o RN acelera o ciclo de desenvolvimento.
- ✓ É possível ver as mudanças instantaneamente no emulador ou no dispositivo real.

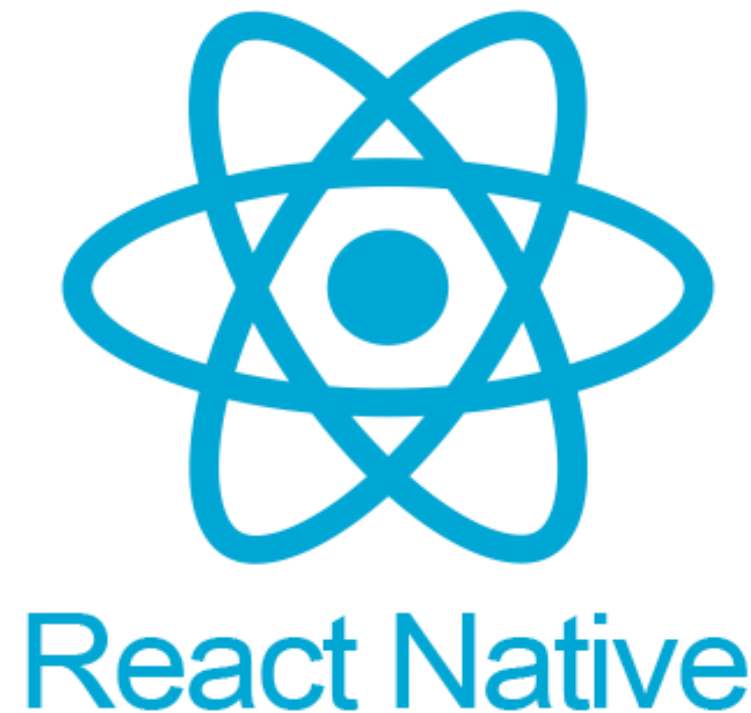
- **Comunidade e ecossistema:**

- ✓ O RN tem uma grande comunidade de desenvolvedores e uma ampla gama de bibliotecas e ferramentas de terceiros, o que facilita a adição de funcionalidades como navegação, gestão de estado, e integração com APIs externas.

# Características do React Native

- **Compatibilidade com código nativo:**

- ✓ o RN permite integrar código nativo escrito em Java (para Android) e Swift/Objective-C (para iOS), o que é útil quando precisamos de funcionalidades específicas ou desejamos otimizar partes do nosso aplicativo.



Fonte: DevTop.io



# Vantagens do React Native

- **Desempenho e uso intensivo de recursos:**
  - ✓ **Aplicativos de jogos complexos:** jogos de alta performance que exigem gráficos complexos e uso intensivo de GPU geralmente se beneficiam mais de código nativo otimizado para cada plataforma.
  - ✓ **Aplicativos de realidade aumentada/virtual:** aplicativos que utilizam AR ou VR exigem um alto nível de desempenho e integração com hardware que é mais facilmente alcançado com código nativo.

# Vantagens do React Native

- **Integração profunda com hardware:**
  - ✓ **Funcionalidades específicas de hardware:** recursos que exigem integração profunda com hardware específico, como processamento de imagens em tempo real, manipulação avançada de câmera, ou acesso a sensores específicos, são geralmente mais eficazes quando implementados em código nativo.
  - ✓ **Bluetooth e conectividade avançada:** aplicativos que necessitam de conectividade avançada, como comunicação com dispositivos via Bluetooth Low Energy (BLE), podem ter limitações ao usar RN devido à dependência de bibliotecas de terceiros.

# Vantagens do React Native

- **Acesso a APIs e funcionalidades recentes:**
  - ✓ **APIs nativas recém-lançadas:** quando novas APIs ou funcionalidades são lançadas para Android ou iOS, pode levar algum tempo para que estejam disponíveis em RN. O desenvolvimento nativo permite acesso imediato a essas funcionalidades.
  - ✓ **Funcionalidades exclusivas de plataforma:** alguns recursos exclusivos de uma plataforma (como certos serviços de Google Play no Android ou algumas funcionalidades específicas do iOS) podem não ser facilmente acessíveis ou totalmente suportados no RN.

# Vantagens do React Native

- **Desempenho de animações complexas:**
  - ✓ **Animações e interações complexas:** animações muito complexas ou interações que exigem uma renderização suave e altamente responsiva podem não ter o mesmo desempenho em RN, especialmente em dispositivos mais antigos ou menos poderosos.
- **Tamanho do aplicativo:**
  - ✓ **Aplicativos com muitas dependências:** o uso de muitas bibliotecas de terceiros pode aumentar o tamanho do aplicativo. Embora isso possa acontecer com aplicativos nativos também, as dependências de bibliotecas de terceiros em RN podem exacerbar o problema.

# Vantagens do React Native

- **Consistência de UI:**
  - ✓ **UI altamente personalizada:** aplicativos que requerem uma interface de usuário altamente personalizada e que seguem as diretrizes de design de cada plataforma, podem ser mais facilmente desenvolvidas em código nativo, garantindo uma aparência e comportamento que atendam precisamente às expectativas dos usuários de cada plataforma.
- **Suporte e manutenção de bibliotecas:**
  - ✓ **Manutenção de bibliotecas de terceiros:** algumas bibliotecas externas podem não ser tão bem mantidas quanto o código nativo. Problemas de compatibilidade ou bugs podem surgir, exigindo investimento de tempo para resolver ou encontrar alternativas.

## Desvantagens do React Native

- **Desempenho:** embora seja muito próximo ao nativo, ainda pode haver casos em que o desempenho não é tão bom quanto os aplicativos nativos puros, especialmente para aplicativos muito complexos ou que requerem uso intensivo de recursos.
- **Dependência de bibliotecas de terceiros:** muitas funcionalidades podem depender de bibliotecas externas, com a possibilidade não ser tão bem mantidas quanto o código nativo.

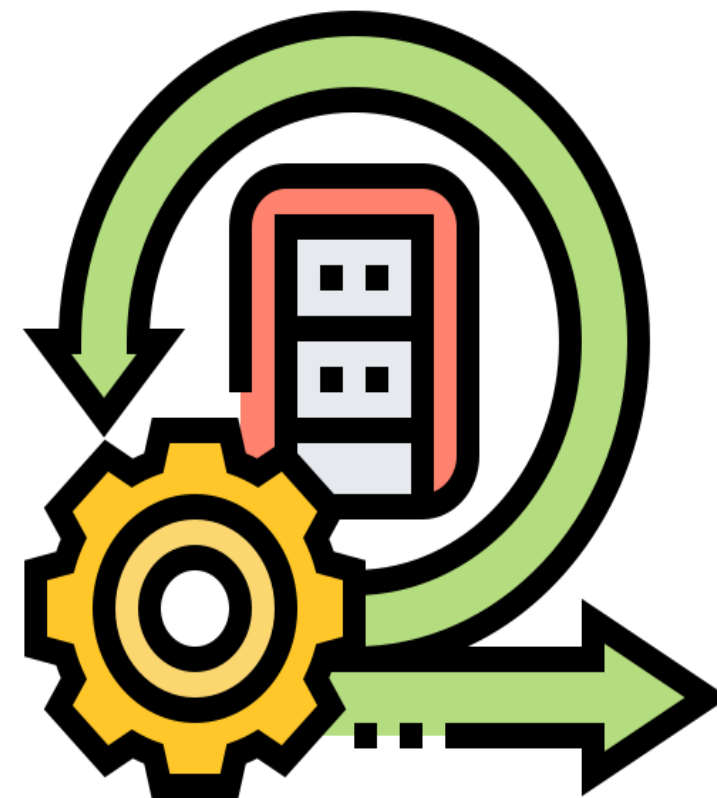
# Criando o Ambiente de Trabalho



Fonte: DreamsTime.com

# Criando o Ambiente de Trabalho

- Node.JS
- NPM (Gerenciador de Pacotes do Node.JS)
- Chocolatey (Gerenciador de Pacotes para o Windows)
- Expo CLI (Command Line Interface)
- Expo Go
  - Android Emulator
  - iOS Emulator



Fonte: Flaticon.com



# Criando o Ambiente de Trabalho

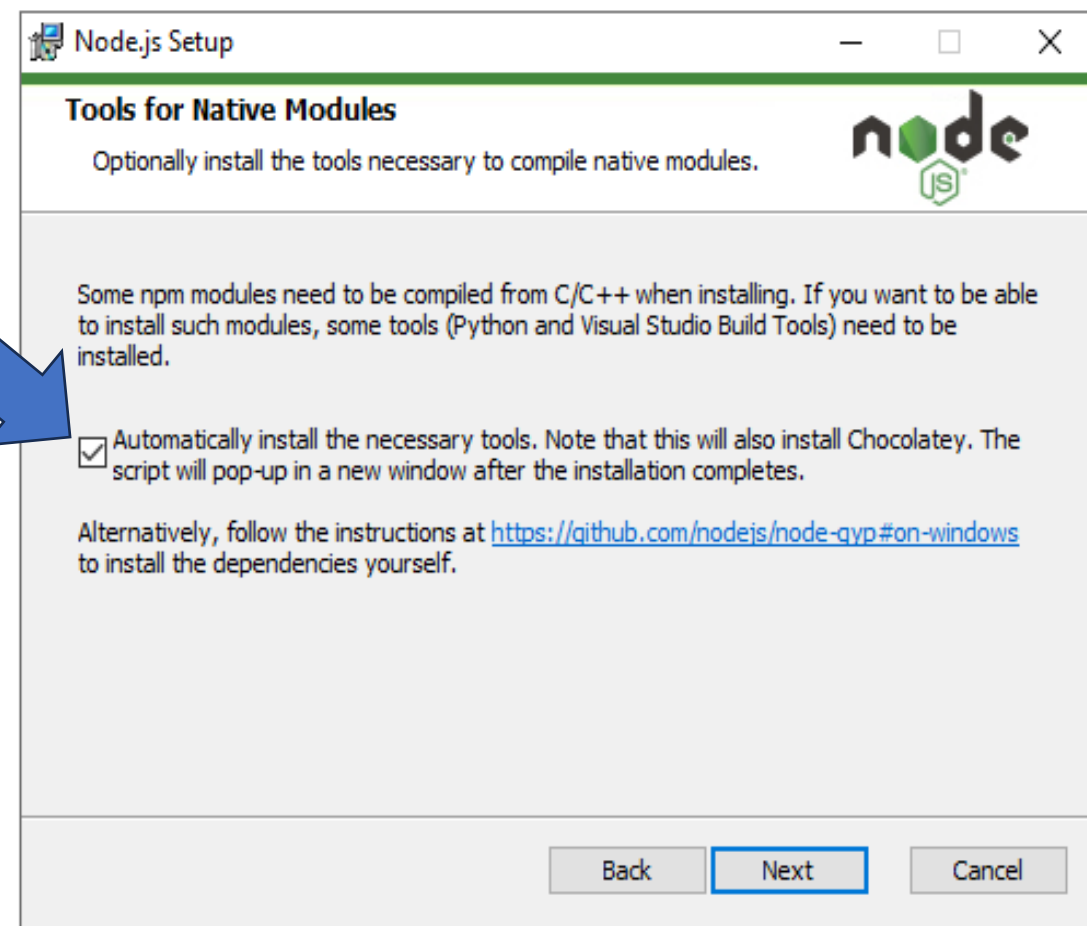
- **Node.JS:** é um ambiente de execução JavaScript baseado no engine do Google Chrome.
- Permite que o JS seja executado no servidor, possibilitando o desenvolvimento de aplicações escaláveis e de alto desempenho.
- Com um modelo assíncrono e orientado a eventos, o Node.js é ideal para aplicações em tempo real, como **APIs, chats e streaming de dados**.
- Além disso, seu ecossistema conta com o **npm (Node Package Manager)**, que facilita a instalação de bibliotecas e frameworks.

# Criando o Ambiente de Trabalho

- Instalação do pacote Chocolatey.

É um gerenciador de pacotes, semelhante ao npm, voltado para a instalação de softwares no ambiente Windows.

Embora não seja diretamente um pacote utilizado em **Node.js**, podendo ser utilizado para instalar o próprio Node.js e outras dependências necessárias para o desenvolvimento.



Fonte: NodeJS.org



## Expo Go

Expo Go is a sandbox that enables you to quickly experiment with building native Android and iOS apps. It's the fastest way to get started. Download any version of it below.

### SDK Version

SDK 52

Latest



### Install Expo Go on your device

Android

 Install

iOS

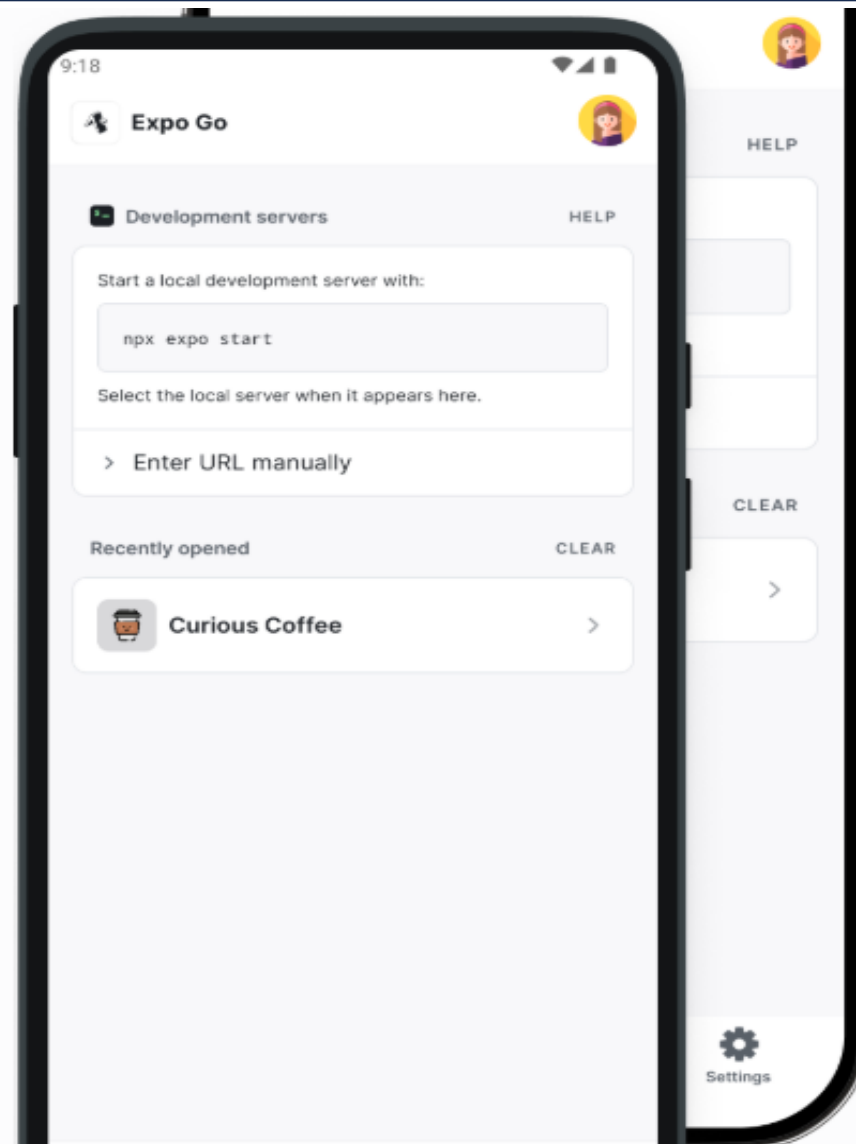
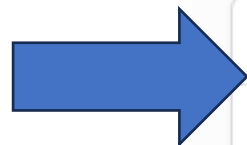
 Install

### Install Expo Go on your emulator/simulator

Android Emulator

 Install

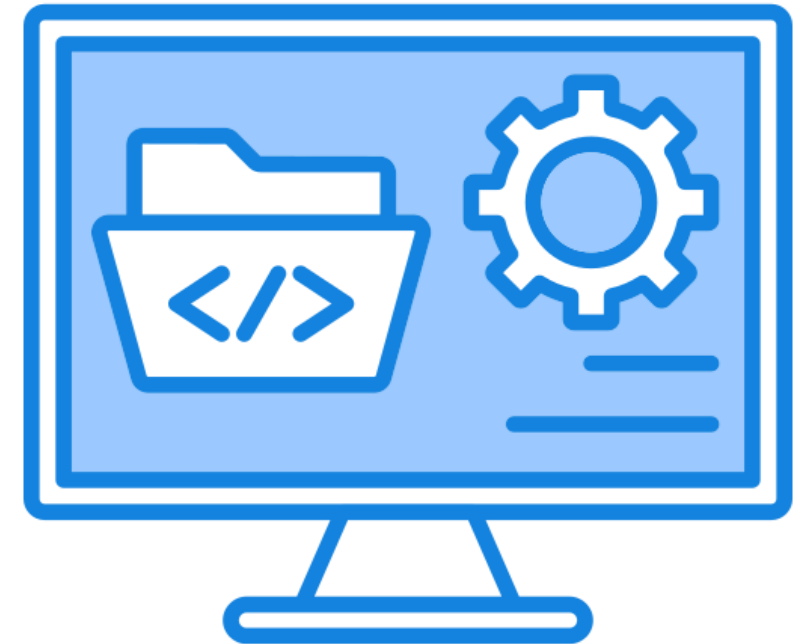
iOS Simulator

 Install

Fonte: Expo.dev

# Criando o Ambiente de Trabalho

- No **VS Code**, instalar as seguintes extensões:
  - Expo Tools
  - React Native & Expo Code Snippers
  - React Native Tools



Fonte: PNGegg.com

# Criando o Ambiente de Trabalho

- **Expo CLI:**

- Ferramenta de linha de comando para criar e gerenciar projetos Expo.

- **Expo SDK:**

- Conjunto de bibliotecas e APIs que fornecem acesso a funcionalidades nativas do dispositivo, como câmera, localização, notificações push, sensores entre outros.
- É integrado diretamente ao Expo e facilita o uso dessas funcionalidades sem necessidade de escrever código nativo.

# Criando o Ambiente de Trabalho

- **Expo Go:**

- Aplicativo disponível na App Store e Google Play que permite executar aplicativos Expo em dispositivos reais.
- Facilita a visualização de mudanças de código em tempo real sem a necessidade de recompilar ou reinstalar o aplicativo.

- **Expo Snack (<https://snack.expo.dev>):**

- Editor de código online que permite experimentar e compartilhar rapidamente código RN/Expo diretamente no navegador.
- Integração com Expo Go para testar o código em dispositivos reais.

# Criar Projeto Expo

- Instalação do pacote create-expo-app: a ferramenta create-expo-app é uma CLI (Command Line Interface) projetada para facilitar a criação e configuração de novos projetos Expo.
- Oferece uma maneira rápida e simplificada de começar a desenvolver aplicativos RN com Expo, substituindo o antigo expo-cli para a criação de projetos:
  - ✓ `npm i create-expo-app -g`
  - ✓ `npm list -g`

# Criar Projeto Expo

- **Criação do projeto:** o comando a seguir criar um projeto Expo com uma configuração.
  - ✓ `expo-template-blank-typescript` é um template que configura o projeto com suporte a TS e um ambiente inicial vazio:

```
npx create-expo-app app --template expo-template-blank-typescript
```
  - ✓ Será criada uma pasta de nome `app`. Abra essa pasta no VS Code.



## Criar Projeto Expo

- Assim como definido na propriedade `scripts` do `package.json`, é possível executar o projeto no Android, iOS e também no navegador (web):

```
"scripts": {  
  "start": "expo start",  
  "android": "expo start --android",  
  "ios": "expo start --ios",  
  "web": "expo start --web"  
}
```

## Criar Projeto Expo

- Inicialmente testaremos o aplicativo no navegador.
- Ao digitar o comando `npm run web` será mostrada a mensagem dizendo que precisaremos instalar alguns pacotes.
- A seguir tem-se um exemplo do comando sugerido, porém, recomenda-se usar o comando sugerido por ele, ou seja, não copie esse aqui:

```
npx expo install react-native-web react-dom @expo/metro-runtime
```

## Criar Projeto Expo

- Execute o comando `npm run web` novamente após instalar os pacotes sugeridos.
- O aplicativo pode ser testado no navegador utilizando a URL sugerida (<http://localhost:8081>) ou abra o aplicativo no Expo Go escaneando o QR Code.

# Leitura Complementar

- Diferenças entre React e React Native

<https://www.youtube.com/watch?v=2-OAUXA4lcE>

- React Native: o que é e tudo sobre o Framework?

[https://www.alura.com.br/artigos/react-native?srsId=AfmBOoq9dSY\\_U2MCL1zCHnanVYguHzRa-ROtlvdsFIHleDcN3qUGPdjI](https://www.alura.com.br/artigos/react-native?srsId=AfmBOoq9dSY_U2MCL1zCHnanVYguHzRa-ROtlvdsFIHleDcN3qUGPdjI)

- Core Components and Native Components

<https://reactnative.dev/docs/intro-react-native-components#:~:text=React%20Native%20is%20an%20open,the%20app%20platform's%20native%20capabilities>



Fonte: Flaticon.com