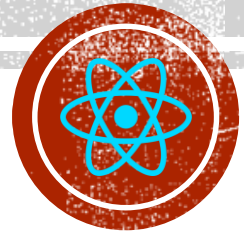
REACT NATIVE



Douglas Nassif Roma Junior

/douglasjunior

/in/douglasjunior

douglasjunior.me

nassifrroma@gmail.com

Slides: https://git.io/vdn2c

AGENDA

- Nativas vs Híbrido vs Multi-plataforma
- Introdução ao React Native
- Instalação
- Criando projetos
- Executando o projeto
- Explorando o ambiente
- Referências

NATIVO

- Aplicativo desenvolvido, compilado e empacotado na plataforma nativa do sistema operacional.
  - Android: Java, Kotlin, C++, etc.
  - iOS: Objective-C, Swift, C, C++, etc.
  - Windows: C#, C++, etc.

- Todos os recursos de software e hardware estão disponíveis naturalmente.

HÍBRIDO

- Aplicativo desenvolvido em uma única linguagem e interpretado (Web View) em diversas plataformas.
  - Cordova/Phonegap: Html, CSS, JavaScript
  - Ionic: Html, CSS, JavaScript/TypeScript, Angular
  - Meteor: Html, CSS, JavaScript/TypeScript

- Desempenho inferior ao nativo.
- Para acessar recursos de hardware ou sistema operacional é preciso criar Plugins complexos e difíceis de manter.

MULTI-PLATAFORMA

- Aplicativo desenvolvido em uma única linguagem, compilado e empacotado para a plataforma nativa.
  - React Native: JavaScript, TypeScript, React, etc.
  - Xamarin: C#.
  - Native Script: JavaScript, TypeScript, Angular, etc.

- Desempenho equivalente ao nativo.
- Para acessar recursos de hardware ou sistema operacional é possível criar módulos que, naturalmente são mais fáceis de manter devido a proximidade com a plataforma nativa.

INTRODUÇÃO AO REACT NATIVE

https://facebook.github.io/react-native/

INTRODUÇÃO AO REACT NATIVE

- Construa apps móveis nativos usando JavaScript e React.
  - React Native permite construir apps utilizando apenas JavaScript. Ele utiliza o mesmo design do React, permitindo a composição de interfaces móveis ricas, utilizando componentes

```
import React, { Component } from 'react';
import { Text, View } from 'react-native';

class WhyReactNativeIsSoGreat extends Component {
  render() {
    return (
      <View>
        <Text>
          Se você gosta do React na Web, você vai gostar do React Native.
        </Text>
        <Text>
          Basta utilizar componentes nativos, como 'View' e 'Text',
          no lugar de componentes Web como 'div' ou 'span'.
        </Text>
      </View>
    );
  }
}
```

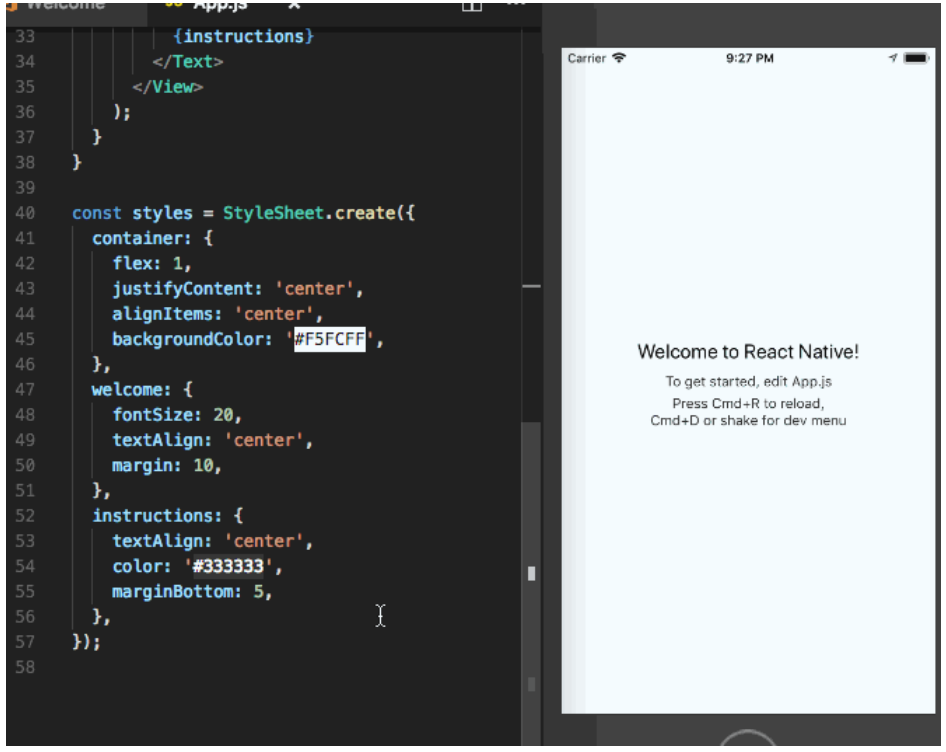
INTRODUÇÃO AO REACT NATIVE

- Um app React Native é um app realmente nativo.
  - Com React Native, você não constrói "web app", ou "HTML5 app", ou um "app híbrido". Você constrói um app mobile real que é indistinguível de qualquer app criado com Java ou Objective-C. React Native utiliza o mesmo fundamento de interfaces em bloco como no Android e iOS. Basta unir estes blocos utilizando JavaScript e React.

```
<ScrollView>
  <Image
    source={{ uri: 'https://i.chzbgr.com/full/7345954048/h7E2C65F9/' }}
    style={{ width: 320, height: 180 }}
  />
  <Text>
    No iOS, um ScrollView do React Native usa o nativo UIScrollView.
    No Android, ele usa o nativo ScrollView.

    No iOS, um Image do React Native usa o nativo UIImageView.
    No Android, ele usa o nativo ImageView.

    O React Native abstrai a maioria dos componentes fundamentais
    permitindo que você aproveite toda a performance nativa com React.
  </Text>
</ScrollView>
```

INTRODUÇÃO AO REACT NATIVE

- Não perca tempo recompilando
  - O React Native permite construir apps rapidamente. Em vez de recompilar, você pode recarregar seu app instantaneamente.

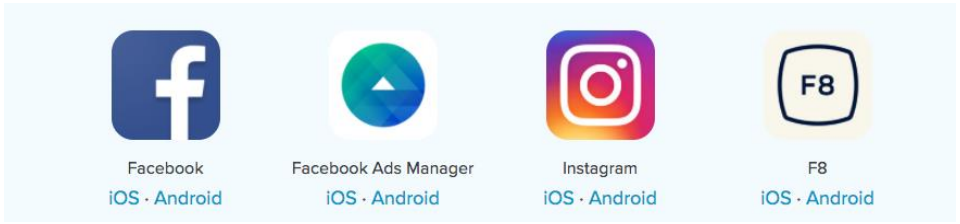


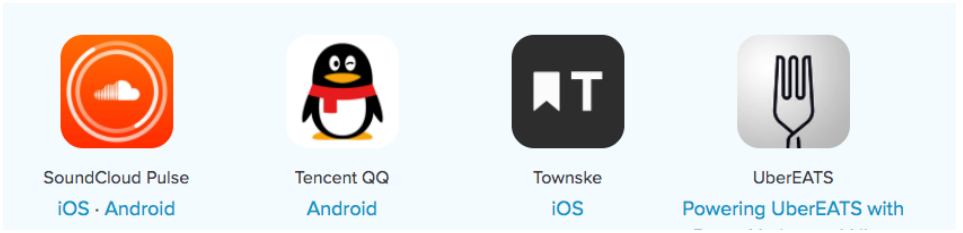
INTRODUÇÃO AO REACT NATIVE

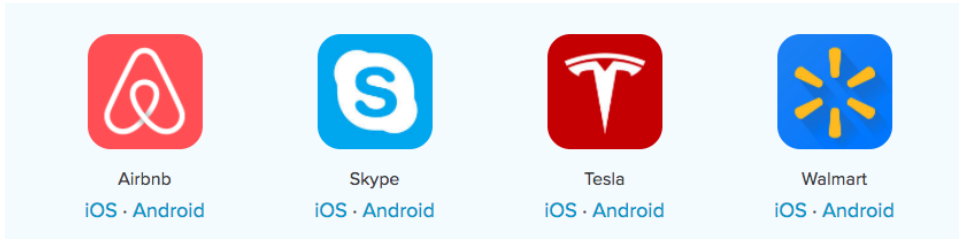
- Use código nativo quando precisar.
  - É simples acessar recursos nativos se você precisar otimizar sua aplicação. Assim como é fácil de construir partes de seu app em React Native, e outros partes usando código nativo diretamente. É assim que o app do Facebook funciona.

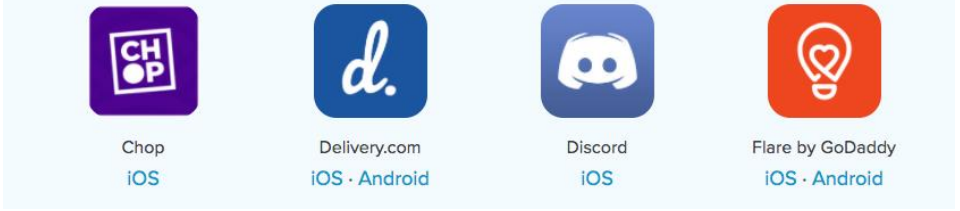
```
import { TheGreatestComponentInTheWorld } from './your-native-code';

<View>
  <TheGreatestComponentInTheWorld />
  <Text>
    TheGreatestComponentInTheWorld permite utilizar Objective-C,
    Java, ou Swift - o processo de desenvolvimento é o mesmo.
  </Text>
</View>
```

INTRODUÇÃO AO REACT NATIVE





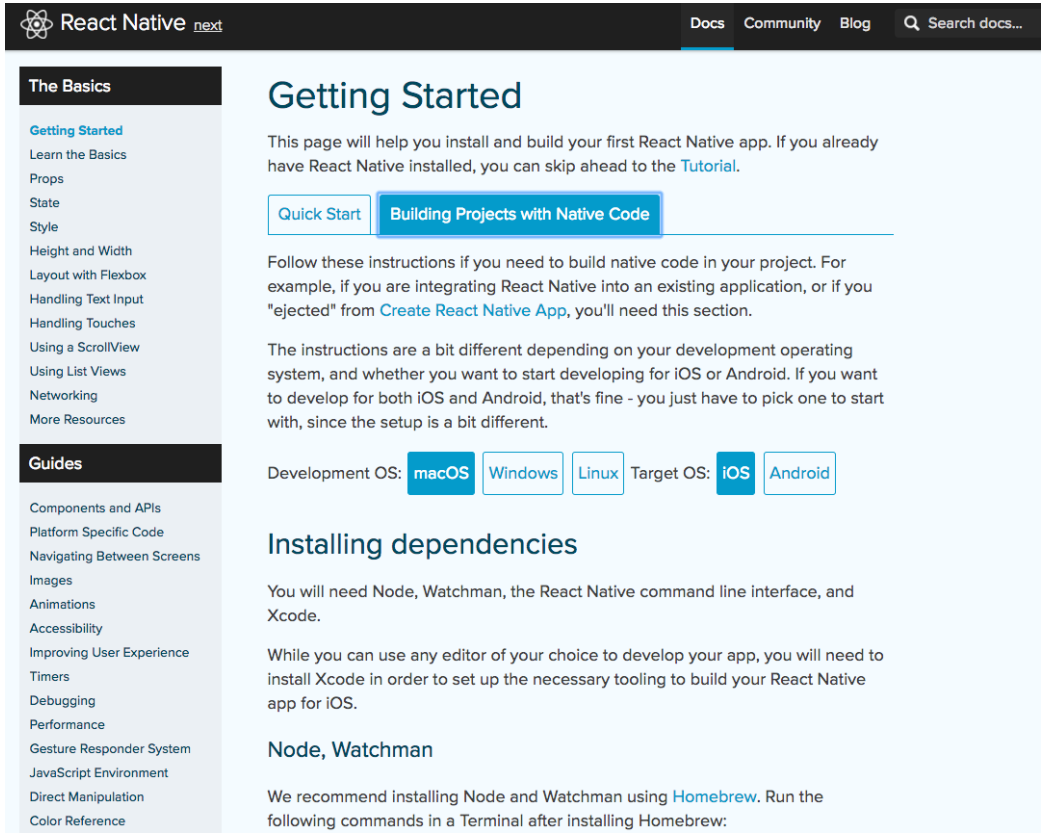




https://facebook.github.io/react-native/showcase.html

INTRODUÇÃO AO REACT NATIVE

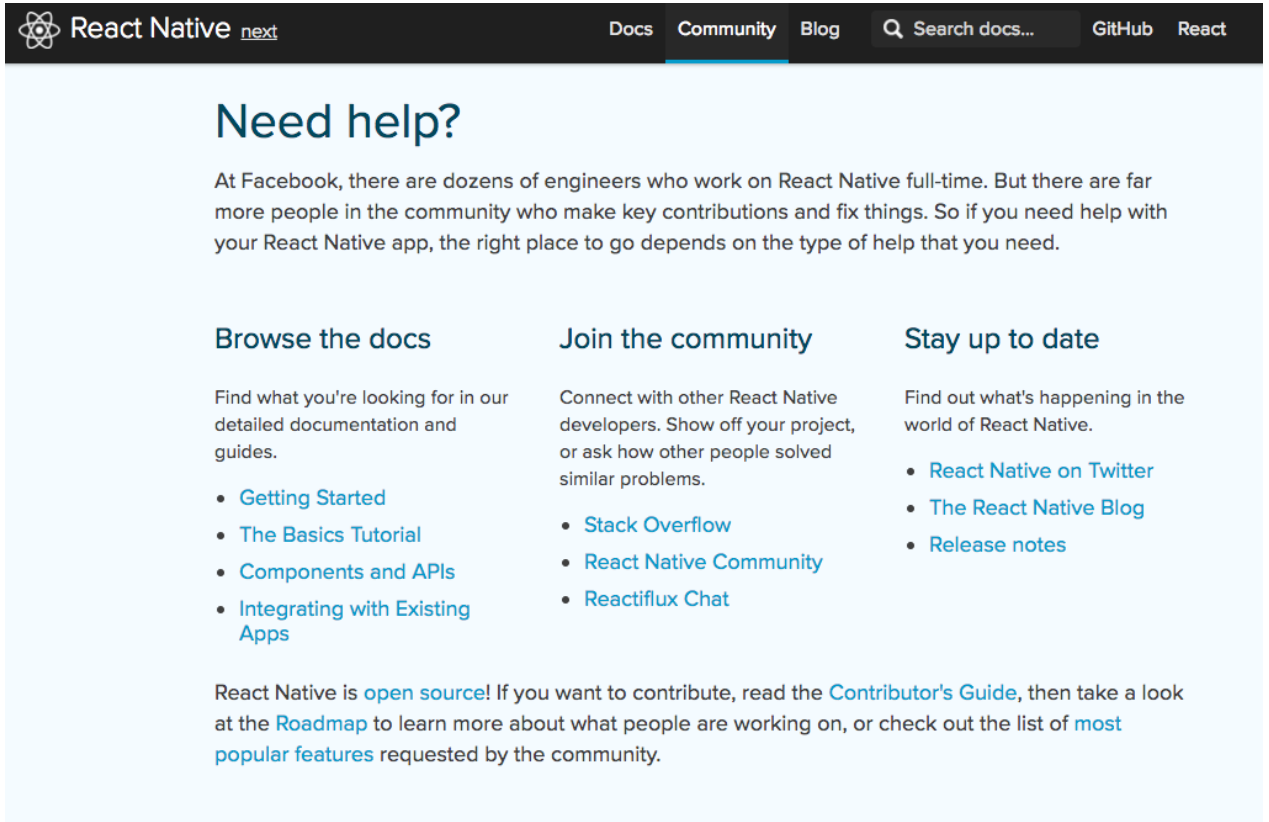
- Documentação rica



https://facebook.github.io/react-native/releases/next/docs/getting-started.html

INTRODUÇÃO AO REACT NATIVE

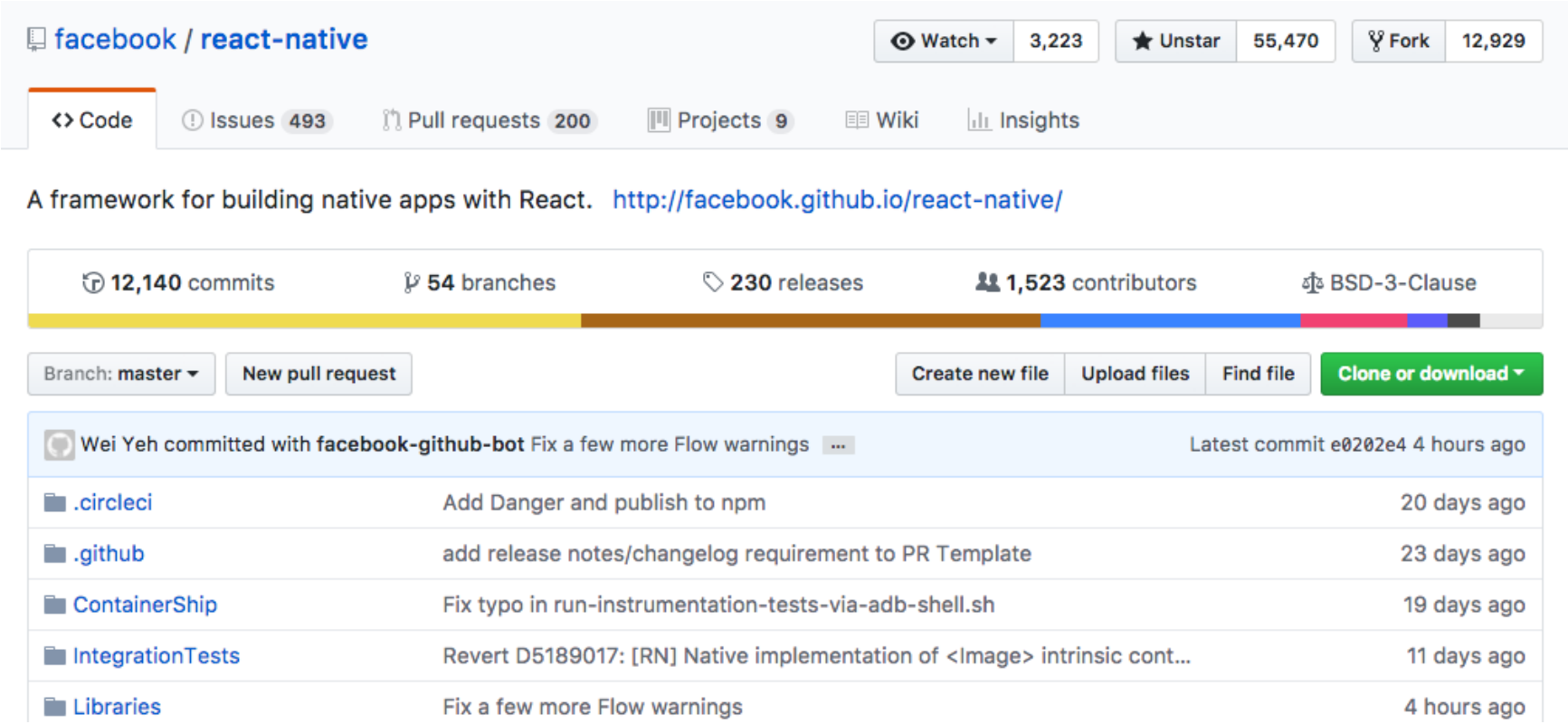
- Comunidade ativa



https://facebook.github.io/react-native/support.html

INTRODUÇÃO AO REACT NATIVE

- Desenvolvimento constante



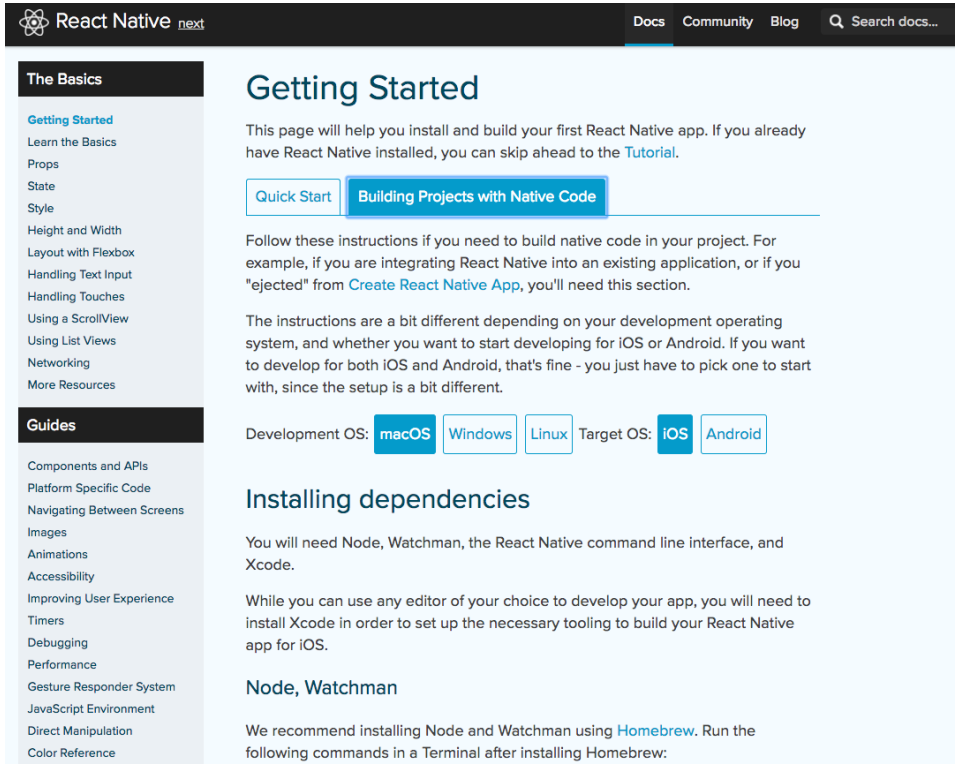
https://github.com/facebook/react-native

INSTALAÇÃO E CONFIGURAÇÃO

- Pré-requisitos
  - Node 6.x
  - Python 2.x
  - Java JDK (Open JDK) >= 1.7
  - Git
  - Android SDK 23.0.1
  - Xcode
  - Windows, Linux ou Mac

INSTALAÇÃO E CONFIGURAÇÃO

- Acesse a página de documentação, selecione seu sistema operacional e a plataforma desejada.



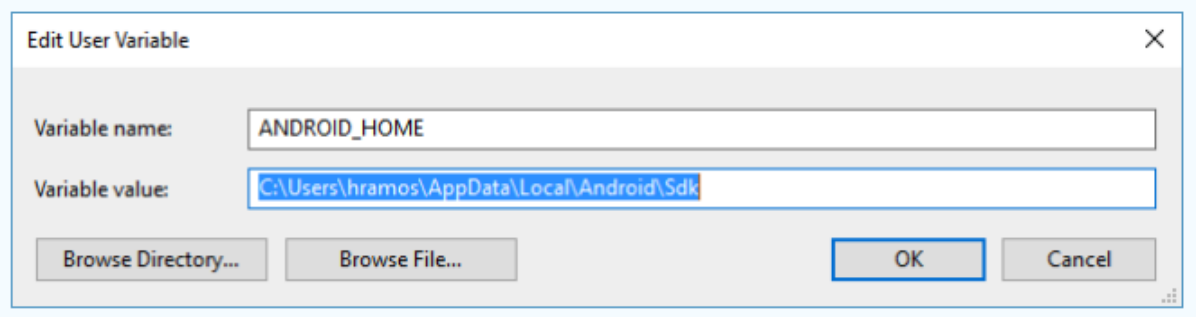
https://facebook.github.io/react-native/releases/next/docs/getting-started.html

INSTALAÇÃO E CONFIGURAÇÃO

- Em todas as plataformas é preciso criar as variáveis de ambiente que apontam para o diretório do Android SDK, e em seguida adicionar ao PATH.
  - Linux ou Mac

```
export ANDROID_HOME=$HOME/Library/Android/sdk
export PATH=$PATH:$ANDROID_HOME/tools
export PATH=$PATH:$ANDROID_HOME/platform-tools
```

  - Windows



https://facebook.github.io/react-native/releases/next/docs/getting-started.html

INSTALAÇÃO E CONFIGURAÇÃO

- No Linux pode ser necessário instalar alguns pacotes 32 bits para compilar para Android.

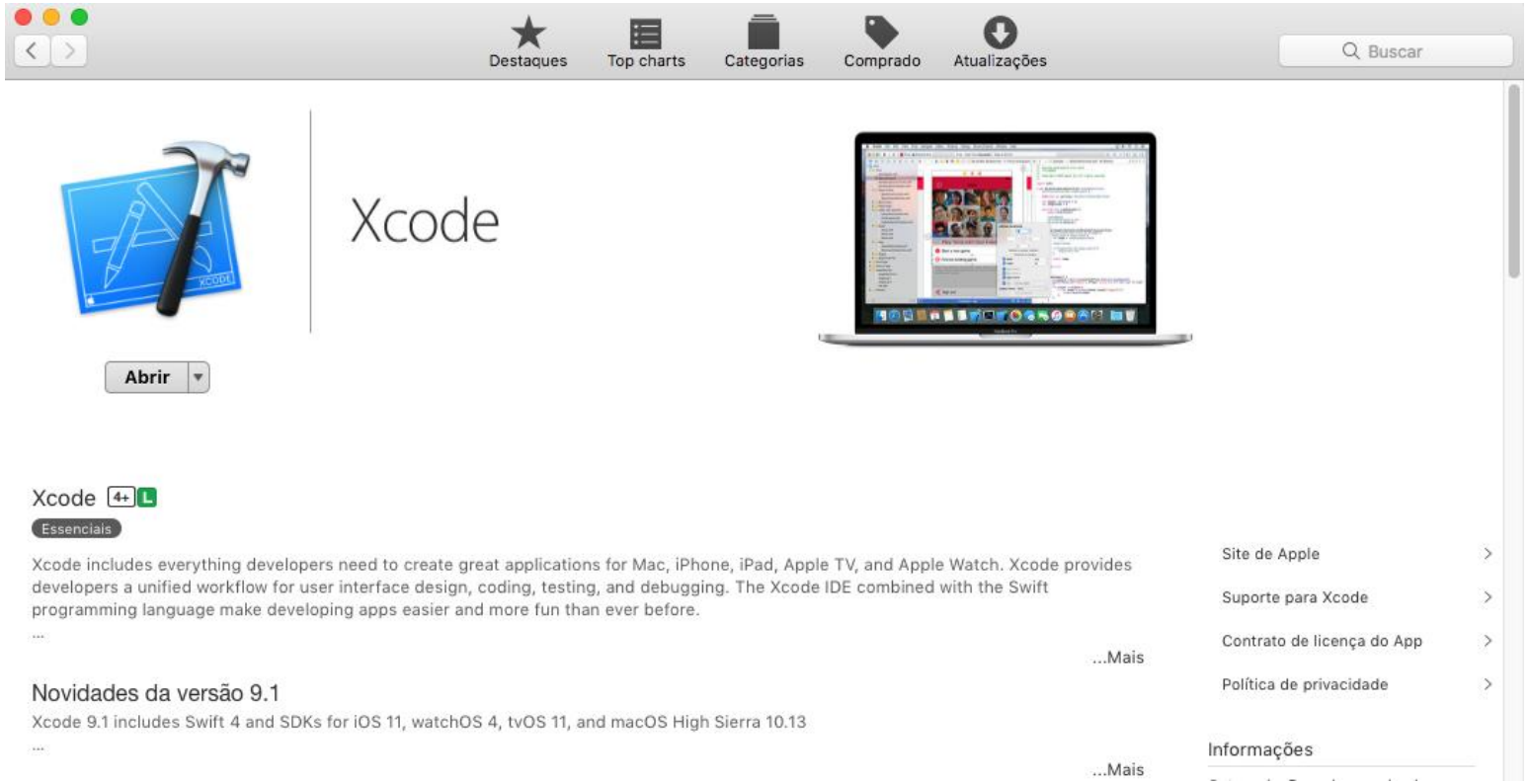
- Instruções do site do Android Studio.
  - https://developer.android.com/studio/install.html

- Stack Overflow
  - https://stackoverflow.com/questions/2710499/android-sdk-on-a-64-bit-linux-machine

https://facebook.github.io/react-native/releases/next/docs/getting-started.html

INSTALAÇÃO E CONFIGURAÇÃO

- Para instalar o Xcode, basta acessar a loja de aplicativos do Mac OSX.



CRIANDO PROJETOS

- Antes de criar o projeto, recomenda-se, utilizar o Yarn no lugar do NPM.

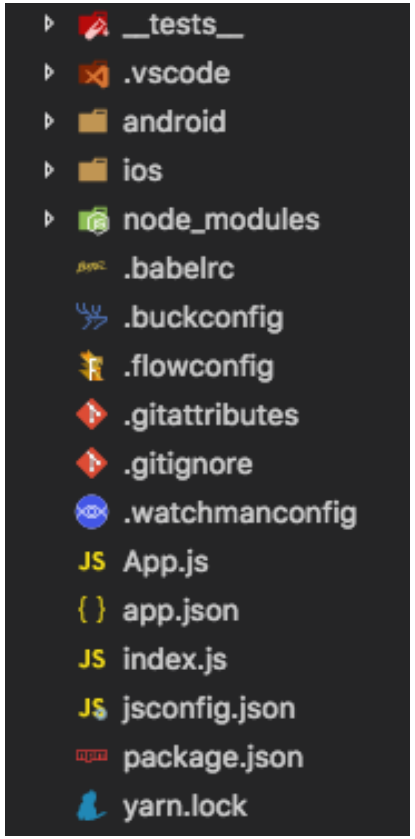
```
$ npm install -g yarn
```

- Após todo ambiente instalado e configurado, você pode criar seu projeto utilizando o comando.

```
$ react-native init PrimeiroProjeto
```

CRIANDO PROJETOS

- Estrutura de diretórios



EXECUTANDO O PROJETO ANDROID

- Antes de executar o projeto a primeira vez, recomenda-se abri-lo com o Android Studio. Este processo vai garantir que todas as dependências necessárias estão instaladas, e se não estiver, o Android Studio irá lhe guiar para a instalação.

- Para executar o projeto Android, basta ter aberto um emulador ou dispositivo plugado ao USB, e então:

```
$ react-native run-android
```

- Caso o `packager` não inicie automaticamente, execute:

```
$ npm start
```

EXECUTANDO O PROJETO IOS

- O projeto iOS pode ser executado diretamente pelo Xcode, ou através do comando:

```
$ react-native run-ios
```

- Caso o `packager` não inicie automaticamente, execute:

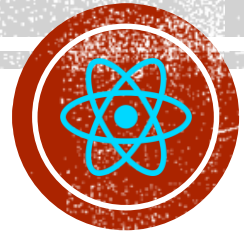
```
$ npm start
```

VAMOS EXPLORAR O AMBIENTE

REFERÊNCIAS

- React Native - https://facebook.github.io/react-native/
- Showcase - https://facebook.github.io/react-native/showcase.html
- Documentação - https://facebook.github.io/react-native/releases/next/docs/getting-started.html

DÚVIDAS?



Douglas Nassif Roma Junior

/douglasjunior

/in/douglasjunior

douglasjunior.me

nassifrroma@gmail.com

Slides: https://git.io/vdn2c