



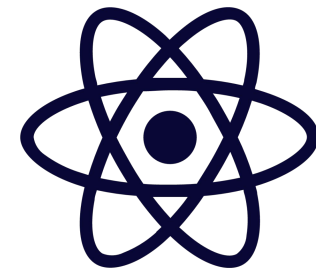
***Mobile***

## **Aula 03 - React Native**

Prof. MSc. Kelson Almeida

# Navegação no React Native

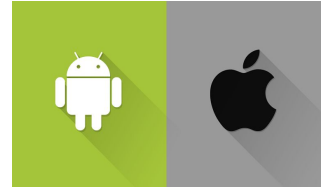
- Navegação refere-se à capacidade de mover-se entre diferentes telas ou seções dentro de um aplicativo.
- O React Native não inclui um sistema de navegação integrado.
- Bibliotecas populares: React Navigation, React Native Navigatio



React Native



# Configurando React Navigation

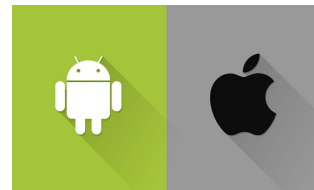


```
bash
```

```
npm install @react-navigation/native  
npm install react-native-screens react-native-safe-area-context  
npm install @react-navigation/native-stack
```



# Criando um Stack Navigator



- Um stack navigator permite a navegação entre telas com a opção de voltar à tela anterior.

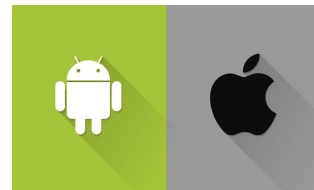
```
jsx
import * as React from 'react';
import { NavigationContainer } from '@react-navigation/native';
import { createNativeStackNavigator } from '@react-navigation/native-stack';
import HomeScreen from './HomeScreen';
import DetailsScreen from './DetailsScreen';

const Stack = createNativeStackNavigator();

function App() {
  return (
    <NavigationContainer>
      <Stack.Navigator>
        <Stack.Screen name="Home" component={HomeScreen} />
        <Stack.Screen name="Details" component={DetailsScreen} />
      </Stack.Navigator>
    </NavigationContainer>
  );
}

export default App;
```

# Passando Dados entre Telas



- Como passar dados de uma tela para outra usando parâmetros de navegação.

```
jsx Copy code

// Em HomeScreen.js
function HomeScreen({ navigation }) {
  return (
    <Button
      title="Go to Details"
      onPress={() => navigation.navigate('Details', { itemId: 86 })}
    />
  );
}

// Em DetailsScreen.js
function DetailsScreen({ route, navigation }) {
  const { itemId } = route.params;
  return <Text>Item ID: {itemId}</Text>;
}
```

# Estrutura de Projeto Recomendada



- Organização de arquivos e pastas.
- Separação de componentes, telas, navegação, e recursos.

```
bash

/app
  /assets
  /components
  /screens
  /navigation
  App.js
```

# Vamos praticar? [1] [com o prof]

Objetivo: Configurar o React Navigation em um novo projeto React Native.

Inicie um novo projeto React Native.

Instale o React Navigation e suas dependências.

Configure um StackNavigator simples.

```
bash
```

```
npx react-native init NavigationDemo  
cd NavigationDemo  
npm install @react-navigation/native  
npm install react-native-screens react-native-safe-area-context  
npm install @react-navigation/native-stack
```

# Vamos praticar? [2] [com o prof]

Objetivo: Implementar a funcionalidade para navegar do HomeScreen para o DetailsScreen ao pressionar um botão.



# Vamos praticar? [3] [com o prof]

Objetivo: Modificar o  
HomeScreen para passar  
parâmetros para o  
DetailsScreen e exibi-los.

# Vamos praticar? [desafio] [com o prof]

Desafio: Aplicativo de Perfil de Usuário

Objetivo: Construir um aplicativo com duas telas:

**Tela Home (Lista de Usuários):** Mostra uma lista de usuários. Ao clicar em um usuário, navega para a tela de detalhes, passando informações do usuário como parâmetro.

**Tela de Detalhes (Perfil do Usuário):** Exibe informações detalhadas do usuário selecionado.

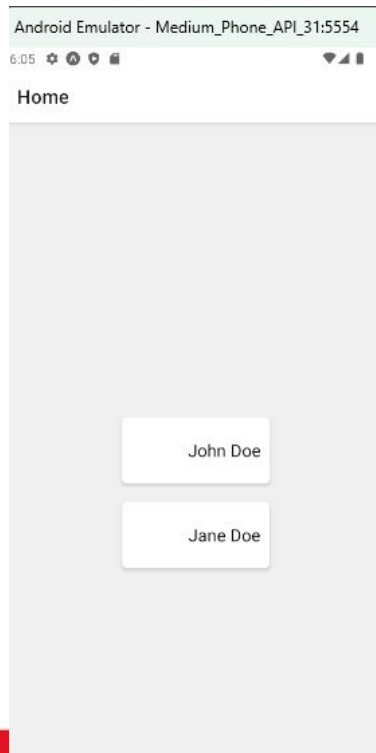
Passos:

Configuração do Navegador: Defina um StackNavigator no seu projeto React Native.

Tela Home:

Crie uma lista de usuários. Você pode usar um array de objetos com informações como nome e idade para cada usuário.

Use o componente Button ou TouchableOpacity para cada usuário na lista. Quando um usuário é pressionado, navegue para a tela de detalhes e passe os dados do usuário como parâmetros.



# Vamos praticar? [desafio] [com o prof]

← Details

## Tela de Detalhes:

Receba os parâmetros passados pela tela Home e exiba-os.  
Por exemplo, mostre o nome e a idade do usuário em Text views.



**Name: Jane Doe**

Age: 25