Guia Completo: Criando um App React Native com Expo e Múltiplas Páginas

# 1. Criar o Projeto

Abra o terminal e execute:  
  
npx create-expo-app appMenuSimples --template blank  
cd appMenuSimples  
  
Isso cria um projeto do zero chamado `appMenuSimples`.

# 2. Instalar Dependências do React Navigation

No terminal, dentro da pasta do projeto:  
  
npm install @react-navigation/native  
npx expo install react-native-screens react-native-safe-area-context  
npm install @react-navigation/native-stack  
  
Essas bibliotecas permitem criar a navegação entre telas.

# 3. Estrutura de Pastas

Você pode criar as telas direto na pasta `screens`:  
  
appMenuSimples/  
 ├─ App.js  
 └─ screens/  
 ├─ HomeScreen.js  
 └─ AboutScreen.js

# 4. Criar as Telas

## 4.1 HomeScreen.js

```javascript  
import React from 'react';  
import { View, Text, Button, StyleSheet } from 'react-native';  
  
export default function HomeScreen({ navigation }) {  
 return (  
 <View style={styles.container}>  
 <Text style={styles.title}>Tela Inicial</Text>  
 <Button  
 title="Ir para Sobre"  
 onPress={() => navigation.navigate('About')}  
 />  
 </View>  
 );  
}  
  
const styles = StyleSheet.create({  
 container: { flex: 1, justifyContent: 'center', alignItems: 'center' },  
 title: { fontSize: 24, marginBottom: 20 }  
});  
```

## 4.2 AboutScreen.js

```javascript  
import React from 'react';  
import { View, Text, Button, StyleSheet } from 'react-native';  
  
export default function AboutScreen({ navigation }) {  
 return (  
 <View style={styles.container}>  
 <Text style={styles.title}>Sobre o App</Text>  
 <Button  
 title="Voltar para Home"  
 onPress={() => navigation.goBack()}  
 />  
 </View>  
 );  
}  
  
const styles = StyleSheet.create({  
 container: { flex: 1, justifyContent: 'center', alignItems: 'center' },  
 title: { fontSize: 24, marginBottom: 20 }  
});  
```

# 5. Configurar o Stack Navigator em App.js

```javascript  
import React from 'react';  
import { NavigationContainer } from '@react-navigation/native';  
import { createNativeStackNavigator } from '@react-navigation/native-stack';  
  
import HomeScreen from './screens/HomeScreen';  
import AboutScreen from './screens/AboutScreen';  
  
const Stack = createNativeStackNavigator();  
  
export default function App() {  
 return (  
 <NavigationContainer>  
 <Stack.Navigator initialRouteName="Home">  
 <Stack.Screen name="Home" component={HomeScreen} />  
 <Stack.Screen name="About" component={AboutScreen} />  
 </Stack.Navigator>  
 </NavigationContainer>  
 );  
}  
```

# 6. Rodar o Projeto

```bash  
npx expo start  
```  
  
- Escaneie o QR code no celular ou use um emulador.  
- Teste os botões para navegar entre as telas.

## ****7️⃣ Adicionar Formulário e Tela de Resultado****

### ****7.1 Criar as telas****

Na pasta screens/, crie:

* FormScreen.js – tela com o formulário
* ResultScreen.js – tela que exibirá os valores enviados

#### ****FormScreen.js****

import React, { useState } from 'react';

import { View, Text, TextInput, Button, StyleSheet } from 'react-native';

export default function FormScreen({ navigation }) {

const [name, setName] = useState('');

const [email, setEmail] = useState('');

const handleSubmit = () => {

// Navega para a tela de resultado passando os valores

navigation.navigate('Result', { name, email });

};

return (

<View style={styles.container}>

<Text style={styles.title}>Formulário</Text>

<Text>Nome:</Text>

<TextInput

style={styles.input}

value={name}

onChangeText={setName}

placeholder="Digite seu nome"

/>

<Text>Email:</Text>

<TextInput

style={styles.input}

value={email}

onChangeText={setEmail}

placeholder="Digite seu email"

keyboardType="email-address"

/>

<Button title="Enviar" onPress={handleSubmit} />

</View>

);

}

const styles = StyleSheet.create({

container: { flex: 1, justifyContent: 'center', padding: 20 },

title: { fontSize: 24, marginBottom: 20, textAlign: 'center' },

input: { borderWidth: 1, borderColor: '#ccc', padding: 10, marginBottom: 15, borderRadius: 5 }

});

#### ****ResultScreen.js****

import React from 'react';

import { View, Text, Button, StyleSheet } from 'react-native';

export default function ResultScreen({ route, navigation }) {

const { name, email } = route.params; // Recebe os valores do formulário

return (

<View style={styles.container}>

<Text style={styles.title}>Resultado</Text>

<Text>Nome: {name}</Text>

<Text>Email: {email}</Text>

<Button title="Voltar para Formulário" onPress={() => navigation.goBack()} />

</View>

);

}

const styles = StyleSheet.create({

container: { flex: 1, justifyContent: 'center', padding: 20 },

title: { fontSize: 24, marginBottom: 20, textAlign: 'center' }

});

### ****7.2 Atualizar o App.js para incluir as novas telas****

import React from 'react';

import { NavigationContainer } from '@react-navigation/native';

import { createNativeStackNavigator } from '@react-navigation/native-stack';

import HomeScreen from './screens/HomeScreen';

import AboutScreen from './screens/AboutScreen';

import FormScreen from './screens/FormScreen';

import ResultScreen from './screens/ResultScreen';

const Stack = createNativeStackNavigator();

export default function App() {

return (

<NavigationContainer>

<Stack.Navigator initialRouteName="Home">

<Stack.Screen name="Home" component={HomeScreen} />

<Stack.Screen name="About" component={AboutScreen} />

<Stack.Screen name="Form" component={FormScreen} />

<Stack.Screen name="Result" component={ResultScreen} />

</Stack.Navigator>

</NavigationContainer>

);

}

### ****7.3 Como testar****

1. No HomeScreen.js, adicione um botão para abrir o formulário:

<Button

title="Ir para Formulário"

onPress={() => navigation.navigate('Form')}

/>

1. Rode o projeto (npx expo start), abra no celular ou emulador.
2. Preencha o formulário e clique em **Enviar** → será redirecionado para a tela de **Resultado**, mostrando os valores digitados.