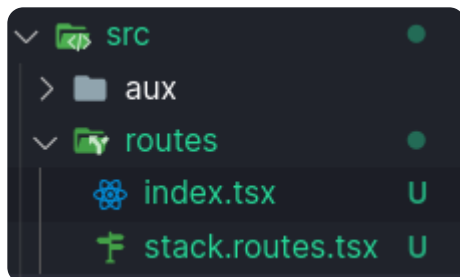


# React Native Navigation

## Criando nossas Rotas

---

Vamos criar um modelo inicial de pastas para seguirmos um exemplo igual e deixar as coisas mais simples. Vamos usar esse exemplo:



Vamos criar também duas telas para usarmos de exemplo

Dentro de src crie uma pasta chamada screens e dentro dela dois arquivos: `screenA.tsx` e `screenB.tsx`, dentro dos arquivo vamos criar duas telas bem simples:

SCREENA:

```
import React from 'react';
import { View } from 'react-native';
export function ScreenA() {

  return (
    <View style={{flex: 1, backgroundColor: '#ff0000', justifyContent:
      'center'}}>

    </View>
  );
}
```

SCREENB:

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

```
export function ScreenB() {

  return (
    <View style={{flex: 1, backgroundColor: '#0ff0ff'}}>

    </View>

  );

}
```

Feito isso vamos voltar para dentro do arquivo `stack.routes.tsx` e nós vamos importar o `createNativeStackNavigator`

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

TYPESCRIPT

Em seguida desestruturamos ele em Screen e Navigator

```
const { Screen, Navigator } = createNativeStackNavigator()
```

TYPESCRIPT

Após isso podemos criar nossas screens, veja:

```
import { createNativeStackNavigator } from '@react-navigation/native-stack'
import { ScreenA } from '../screens/ScreenA'
import { ScreenB } from '../screens/ScreenB'

const { Screen, Navigator } = createNativeStackNavigator()

export function StackRoutes() {

  return (

    <Navigator>

      <Screen name='screenA' component={ScreenA} />
      <Screen name='screenB' component={ScreenB} />

    </Navigator>

  )

}
```

# Criando a navegação

---

Dentro do nosso arquivo `index.tsx` nós importamos o `NavigationContainer` do `react-navigation/native` e o nosso `StackRoutes` que acabamos de criar, depois disso nós usamos o `NavigationContainer` para que nossa navegação funcione com as rotas que criamos.

```
import { NavigationContainer } from '@react-navigation/native'
import { StackRoutes } from './stack.routes'

export function Routes() {
  return (
    <NavigationContainer>
      <StackRoutes />
    </NavigationContainer>
  )
}
```

e o nosso `App.tsx` fica dessa forma:

```
import { Routes } from './src/routes'

export default function App() {
  return (
    <Routes />
  )
}
```

Agora vamos fazer a navegação entre telas funcionar, primeiro precisamos cuidar dos tipos, afinal estamos usando TypeScript, dentro da sua pasta `src` crie outra pasta com o nome `@types`, e dentro do `@types` crie um arquivo chamado `navigation.d.ts`. Muito bem, nesse arquivo vamos fazer um export de forma global para que ele seja visto por toda nossa aplicação, vamos criar um namespace de `ReactNavigation` e uma interface chamada `RootParamList`, dentro delas nós vamos tipar nossas screens, veja:

```
export declare global {
  namespace ReactNavigation {
    interface RootParamList {
      screenA: undefined
      screenB: undefined
    }
  }
}
```

```
}  
}
```

Nesse momento não estamos passando nenhuma propriedade entre uma interface e outra, por isso nossas screens são tipadas como `undefined`

Após criar esse arquivo de tipos nós vamos criar um botão na nossa screenA e criar uma função para que nossa navegação aconteça vamos também importar o `useNavigation` para nos ajudar, nossa screenA fica assim:

```
import React from 'react';  
import { View, Button } from 'react-native';  
import { useNavigation } from '@react-navigation/native'  
  
export function ScreenA() {  
  const navigation = useNavigation()  
  
  function openScreen() {  
  
    navigation.navigate('screenB')  
  
  }  
  
  return (  
    <View style={{flex: 1, backgroundColor: '#ff0000', justifyContent:  
      'center'}}>  
      <Button title='Ir para a tela B' onPress={openScreen} />  
    </View>  
  );  
}
```

Assim que pressionarmos o botão "Ir para tela B" nossa navegação estará funcionando.