

Vitor Amadeu Souza

Introdução ao

React Native

No ambiente Expo Snack

Parte XIII

© 2024 by Cerne Tecnologia e Treinamento Ltda.

© 2024 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

Outubro de 2024

Direitos reservados por:

Cerne Tecnologia e Treinamento Ltda

Produção: Cerne Tecnologia e Treinamento

E-mail da Empresa: cerne@cerne-tec.com.br

Home Page: www.cerne-tec.com.br.com.br

Atendimento ao Consumidor: sac@cerne-tec.com.br

Contato com o Autor: vitor@cerne-tec.com.br



FEITO NO BRASIL

***“Venha também sobre mim a tua benignidade, ó Senhor, e a tua
salvação, segundo a tua palavra.”***

Sl 119:41

Cerne Tecnologia

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos eletrônicos em diversas áreas: Médica, Entretenimento, Industrial, Robótica, Científica, Automobilística, Aeronáutica, etc. Trabalhamos com tecnologia microcontrolada usando o PIC, ARM, AVR, 8051, dsPIC, PIC24, PIC32 além do Arduino, Raspberry, Beaglebone etc. Desenvolvemos o projeto desde sua concepção até a entrega do produto final, passando pelas etapas de esquema elétrico, protótipo e desenvolvimento de circuito impresso.

Desenvolvemos aplicativos para smartphones/tablets Android, iOS, Blackberry, Windows Phone e no desenvolvimento de softwares a nível PC para plataforma Windows, usando ferramentas como o Visual Basic, C# e C++.

Atuamos na parte de montagem de placas, onde podemos fornecer ambos os serviços de desenvolvimento de projetos e produção ou apenas um destes.

Desenvolvemos esquemas elétricos e layout de PCI, tanto em tecnologia convencional como SMD.

Temos a flexibilidade de customizar um de nossos produtos, de modo a atender a uma necessidade específica do cliente, tornando o custo de desenvolvimento menor se comparado a construção de um projeto desde a sua fase inicial.

Desenvolvemos e fornecemos kits didáticos para diversos microcontroladores além de apostilas, livros e e-books.

Na hora de desenvolver um projeto ou equipar seu laboratório não hesite em nos contatar. Entre em contato conosco através do endereço cerne-tec.com.br para obter mais informações.



Sumário

Capítulo I – Metodologia de desenvolvimento.....	6
1. Introdução.....	6
Capítulo II – Programação em React Native.....	7
1. React-navigation.....	7
2. Vector icons.....	13
3. Jogo de adivinhação.....	17

Capítulo I

Metodologia de desenvolvimento

1. Introdução

Neste livro, você encontrará uma parte da série dedicada ao desenvolvimento de projetos com React Native. Ao longo desta obra, diversos experimentos serão apresentados conforme o sumário, permitindo um aperfeiçoamento contínuo no uso deste framework para a construção de interfaces responsivas com foco em dispositivos móveis.

Capítulo II

Programação em React Native

1. React-navigation

Este código é uma aplicação React Native que implementa uma navegação de pilha (stack navigation) usando a biblioteca `@react-navigation/native`. A aplicação apresenta quatro telas: "Home", "Details", "About" e "Settings", e permite que o usuário navegue entre elas. Para este exemplo, teremos vários arquivos em que cada um será responsável por uma tela da aplicação.

No arquivo `App.js`, a navegação é configurada utilizando o componente `NavigationContainer`, que é o contêiner principal para a navegação, e o `createNativeStackNavigator`, que cria uma pilha de telas. O `Stack.Navigator` contém as quatro telas mencionadas, cada uma com um nome (`name`) e o respectivo componente que será exibido quando essa tela for acessada.

A tela inicial, `HomeScreen`, contém três botões, que levam o usuário para as telas de detalhes (`DetailScreen`), sobre

(AboutScreen), e configurações (SettingsScreen). Cada botão chama a função `navigation.navigate()` para ir para a respectiva tela.

Na tela `DetailScreen`, o usuário vê um texto "Tela de Detalhes" e um botão "Voltar", que usa o método `navigation.goBack()` para retornar à tela anterior na pilha.

A tela `SettingsScreen` exibe um título "Configurações" e um texto descritivo sobre a funcionalidade, além de um botão "Voltar", também usando `navigation.goBack()`.

A navegação entre essas telas é gerenciada automaticamente pela biblioteca de navegação, permitindo uma experiência de usuário suave e fácil, sem a necessidade de implementar roteamento manualmente.

```
// App.js
import React from 'react';
import {NavigationContainer} from '@react-navigation/native';
import {createNativeStackNavigator} from '@react-navigation/native-stack';
import HomeScreen from './HomeScreen';
import DetailScreen from './DetailScreen';
import AboutScreen from './AboutScreen'; // Importando a nova tela
import SettingsScreen from './SettingsScreen'; // Importando a nova tela

const Stack = createNativeStackNavigator();

const App = () => {
  return (
    <NavigationContainer>
      <Stack.Navigator initialRouteName="Home">
```

```

        <Stack.Screen name="Home" component={HomeScreen} />
        <Stack.Screen name="Details" component={DetailScreen} />
        <Stack.Screen name="About" component={AboutScreen} />
        <Stack.Screen name="Settings" component={SettingsScreen} />
      </Stack.Navigator>
    </NavigationContainer>
  );
};

```

```
export default App;
```

```
// DetailScreen.js
```

```

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

const DetailScreen = ({ navigation }) => {
  return (
    <View style={styles.container}>
      <Text style={styles.title}>Tela de Detalhes</Text>
      <Button title="Voltar" onPress={() => navigation.goBack()} />
    </View>
  );
};

```

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
  title: {
    fontSize: 24,
  },
});

```

```
export default DetailScreen;
```

```
// HomeScreen.js
```

```

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

const HomeScreen = ({ navigation }) => {

```

```

return (
  <View style={styles.container}>
    <Text style={styles.title}>Tela Inicial</Text>

    {/* Botão para Tela de Detalhes */}
    <View style={styles.buttonContainer}>
      <Button
        title="Ir para a Tela de Detalhes"
        onPress={() => navigation.navigate('Details')}
      />
    </View>

    {/* Botão para Tela Sobre */}
    <View style={styles.buttonContainer}>
      <Button
        title="Ir para Sobre"
        onPress={() => navigation.navigate('About')}
      />
    </View>

    {/* Botão para Tela de Configurações */}
    <View style={styles.buttonContainer}>
      <Button
        title="Ir para Configurações"
        onPress={() => navigation.navigate('Settings')}
      />
    </View>
  </View>
);
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
  title: {
    fontSize: 24,
  },
  buttonContainer: {
    marginVertical: 10,
    width: '80%',
  }
});

```