

Vitor Amadeu Souza

Introdução ao

React Native

No ambiente Expo Snack

Parte XII

© 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. Login.....	7
2. Header, Rodapé e Scroll.....	11
3. Componente Botão.....	15

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. Login

Este exemplo trata de uma aplicação de autenticação de usuário utilizando React Native. A aplicação permite que um usuário insira seu e-mail e senha, verificando se as credenciais estão corretas. A estrutura principal do código inclui estados que armazenam o e-mail e a senha inseridos, além de um indicador de autenticação (`isAuthenticated`) e uma variável para controlar a cor de fundo da tela (`bgColor`). Quando o usuário pressiona o botão "Entrar", a função `handleLogin` é chamada. Essa função verifica se o e-mail e a senha correspondem a um conjunto pré-definido (neste caso, 'admin' e '1234'). Se as credenciais estiverem corretas, o estado `isAuthenticated` é definido como verdadeiro, e a cor de fundo muda para verde. Caso contrário, o estado é definido como falso, e a cor de fundo muda para vermelho.

A renderização é condicional: se o usuário não estiver autenticado, são exibidos os campos de entrada para o e-mail e a

senha, além do botão "Entrar". Se o usuário estiver autenticado, aparece uma mensagem de boas-vindas. Se as credenciais forem incorretas, uma mensagem de erro informa que o usuário não está cadastrado.

Os estilos são definidos usando StyleSheet, proporcionando uma interface básica mas funcional. Este código serve como uma introdução à implementação de autenticação simples em aplicações móveis com React Native, demonstrando conceitos fundamentais como gerenciamento de estado, manipulação de eventos e renderização condicional.

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

export default function App() {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [isAuthenticated, setIsAuthenticated] = useState(false);
  const [bgColor, setBgColor] = useState('white');

  const handleLogin = () => {
    if (email === 'admin' && password === '1234') {
      setIsAuthenticated(true);
      setBgColor('green');
    } else {
      setIsAuthenticated(false);
      setBgColor('red');
    }
  };

  return (
    <View style={ [styles.container, { backgroundColor: bgColor } ]}>
```

```

    {!isAuthenticated ? (
      <>
        <TextInput
          style={styles.input}
          placeholder="E-mail"
          value={email}
          onChangeText={setEmail}
        />
        <TextInput
          style={styles.input}
          placeholder="Senha"
          secureTextEntry
          value={password}
          onChangeText={setPassword}
        />
        <Button title="Entrar" onPress={handleLogin} />
      </>
    ) : (
      <Text style={styles.welcome}>Bem-vindo, {email}!</Text>
    )}
    {!isAuthenticated && bgColor === 'red' && (
      <Text style={styles.errorMessage}>Usuário não cadastrado.</Text>
    )}
  </View>
);
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    padding: 20,
  },
  input: {
    height: 40,
    borderColor: 'gray',
    borderWidth: 1,
    marginBottom: 20,
    paddingHorizontal: 10,
  },
  welcome: {
    fontSize: 24,
    textAlign: 'center',
    marginVertical: 20,
  },
});

```

```
    color: 'white',  
  },  
  errorMessage: {  
    textAlign: 'center',  
    color: 'white',  
    marginTop: 10,  
  },  
},  
});
```

