|  |  |
| --- | --- |
| Disciplina: | RPG0023 - Vamos criar um App! |
| Nome: | Iggor Motta de Oliveira |
| Turma: | 2025.1 |
| Campus: | EAD |
| Curso: | Desenvolvimento Full stack |
| Link: | https://github.com/Iggor-motta/Mundo4-RPG0023 |

A black and white logo

AI-generated content may be incorrect.

**Missão Prática / Nível 1 / Mundo 4**

**RPG0023 “ Vamos criar um App!”**

1. **Requisitos Funcionais:**

* **Cadastro de Fornecedores:** O aplicativo deve permitir o cadastro de fornecedores, incluindo informações detalhadas, como nome, endereço, contato e categorias de produtos fornecidos. Essas informações serão exibidas utilizando componentes como <Text>, <TextInput>, e <Image>.
* **Listagem de Fornecedores:** Deve ser possível visualizar uma lista de fornecedores cadastrados, com opções de pesquisa e filtragem com base em critérios como categoria ou localização. A lista de fornecedores será exibida utilizando componentes como <Text> e <Image>.
* **Associação de Imagens:** O aplicativo deve permitir a associação de imagens aos perfis dos fornecedores. Os usuários devem poder fazer o upload de logotipos ou fotos relacionadas ao fornecedor, utilizando o componente <Image>.
* **Experiência de Usuário Intuitiva:** A interface do aplicativo deve ser intuitiva e fácil de usar, garantindo que os usuários possam navegar, adicionar e editar informações de forma eficiente. Isso será alcançado usando componentes como <Text>, <TextInput>, e <Image>.

2. **Objetivo da Prática:**

* Busca criar um aplicativo móvel eficaz para o cadastro

de fornecedores, com listas e imagens de alta qualidade, economizando

recursos e proporcionando uma excelente experiência ao usuário. A escolha da tecnologia React Native é crucial para estabelecer uma presença sólida no mercado móvel.

3. **Resultado da execução dos códigos:**

4. **Análise e Conclusão**

Todos os códigos solicitados neste roteiro esta no link abaixo:

<https://github.com/Iggor-motta/Mundo4-RPG0023>

**Códigos:**

import React, { useState } from 'react';

import {

View, Text, TextInput,Image, ScrollView, TouchableOpacity, StyleSheet,} from 'react-native';

interface Fornecedor {

id: string;

nome: string;

endereco: string;

contato: string;

categorias: string[];

}

const CadastroFornecedores = () => {

const [fornecedores, setFornecedores] = useState<Fornecedor[]>([]);

const [nome, setNome] = useState('');

const [endereco, setEndereco] = useState('');

const [contato, setContato] = useState('');

const [categoriaInput, setCategoriaInput] = useState('');

const [categorias, setCategorias] = useState<string[]>([]);

const [searchText, setSearchText] = useState('');

const [filterCategory, setFilterCategory] = useState('');

const adicionarCategoria = () => {

if (categoriaInput.trim() !== '') {

setCategorias([...categorias, categoriaInput.trim()]);

setCategoriaInput('');

}

};

const removerCategoria = (index: number) => {

const novasCategorias = [...categorias];

novasCategorias.splice(index, 1);

setCategorias(novasCategorias);

};

const cadastrarFornecedor = () => {

if (

nome.trim() === '' ||

endereco.trim() === '' ||

contato.trim() === '' ||

categorias.length === 0

) {

alert('Por favor, preencha todos os campos.');

return;

}

const novoFornecedor: Fornecedor = {

id: Date.now().toString(),

nome,

endereco,

contato,

categorias,

};

setFornecedores([...fornecedores, novoFornecedor]);

setNome('');

setEndereco('');

setContato('');

setCategorias([]);

};

const filtrarFornecedores = () => {

return fornecedores.filter((fornecedor) => {

const nomeLower = fornecedor.nome.toLowerCase();

const searchTextLower = searchText.toLowerCase();

const categoriaLower = filterCategory.toLowerCase();

const categoriasLower = fornecedor.categorias.map((cat) => cat.toLowerCase());

const nomeMatch = searchTextLower === '' || nomeLower.includes(searchTextLower);

const categoriaMatch = filterCategory === '' || categoriasLower.includes(categoriaLower);

return nomeMatch && categoriaMatch;

});

};

const fornecedoresFiltrados = filtrarFornecedores();

return (

<ScrollView style={styles.container}>

<Text style={styles.titleProjeto}>Projeto RGP0023</Text>

<Image style={styles.logo} source={require('../components/imagens/logo.png')}/>

<Text style={styles.title}>Cadastro de Fornecedores</Text>

<Text style={styles.label}>Nome:</Text>

<TextInput

style={styles.input}

value={nome}

onChangeText={setNome}

placeholder="Digite o nome do fornecedor"

/>

<Text style={styles.label}>Endereço:</Text>

<TextInput

style={styles.input}

value={endereco}

onChangeText={setEndereco}

placeholder="Digite o endereço do fornecedor"

/>

<Text style={styles.label}>Contato:</Text>

<TextInput

style={styles.input}

value={contato}

onChangeText={setContato}

placeholder="Digite o contato do fornecedor"

keyboardType="phone-pad"

/>

<Text style={styles.label}>Categorias de Produtos:</Text>

<View style={styles.categoriaInputContainer}>

<TextInput

style={styles.categoriaInput}

value={categoriaInput}

onChangeText={setCategoriaInput}

placeholder="Adicionar categoria"

/>

<TouchableOpacity style={styles.addButton} onPress={adicionarCategoria}>

<Text style={styles.addButtonText}>Adicionar</Text>

</TouchableOpacity>

</View>

{categorias.length > 0 && (

<View style={styles.categoriasList}>

{categorias.map((categoria, index) => (

<View key={index} style={styles.categoriaItem}>

<Text>{categoria}</Text>

<TouchableOpacity onPress={() => removerCategoria(index)}>

<Text style={styles.removeButton}>X</Text>

</TouchableOpacity>

</View>

))}

</View>

)}

<TouchableOpacity style={styles.cadastrarButton} onPress={cadastrarFornecedor}>

<Text style={styles.cadastrarButtonText}>Cadastro</Text>

</TouchableOpacity>

<Text style={styles.title}>Lista de Fornecedores</Text>

<TextInput

style={styles.input}

value={searchText}

onChangeText={setSearchText}

placeholder="Pesquisar por nome"

/>

<TextInput

style={styles.input}

value={filterCategory}

onChangeText={setFilterCategory}

placeholder="Filtrar por categoria"

/>

{fornecedoresFiltrados.length === 0 ? (

<Text>Nenhum fornecedor cadastrado.</Text>

) : (

fornecedoresFiltrados.map((fornecedor) => (

<View key={fornecedor.id} style={styles.fornecedorItem}>

<View style={styles.fornecedorInfo}>

<Text style={styles.fornecedorNome}>{fornecedor.nome}</Text>

<Text>Endereço: {fornecedor.endereco}</Text>

<Text>Contato: {fornecedor.contato}</Text>

<Text>Categorias: {fornecedor.categorias.join(', ')}</Text>

</View>

</View>

))

)}

</ScrollView>

);

};

const styles = StyleSheet.create({

logo: {

display:'flex',

alignItems:'center',

justifyContent:'center',

marginLeft:'auto',

marginRight:'auto',

},

container: {

flex: 1,

padding: 20,

backgroundColor: '#f5f5f5',

},

titleProjeto: {

margin: 24,

marginTop: 20,

fontSize: 20,

textAlign: 'center',

},

title: {

fontSize: 24,

fontWeight: 'bold',

marginBottom: 20,

textAlign: 'center',

},

label: {

fontSize: 16,

marginBottom: 5,

fontWeight: 'bold',

},

input: {

borderWidth: 1,

borderColor: '#ccc',

borderRadius: 5,

padding: 10,

marginBottom: 15,

backgroundColor: '#fff',

},

categoriaInputContainer: {

flexDirection: 'row',

marginBottom: 15,

},

categoriaInput: {

flex: 1,

borderWidth: 1,

borderColor: '#ccc',

borderRadius: 5,

padding: 10,

marginRight: 10,

backgroundColor: '#fff',

},

addButton: {

backgroundColor: '#007bff',

padding: 10,

borderRadius: 5,

justifyContent: 'center',

alignItems: 'center',

},

addButtonText: {

color: '#fff',

fontWeight: 'bold',

},

categoriasList: {

flexDirection: 'row',

flexWrap: 'wrap',

marginBottom: 15,

},

categoriaItem: {

backgroundColor: '#e0e0e0',

borderRadius: 5,

padding: 8,

marginRight: 8,

marginBottom: 5,

flexDirection: 'row',

alignItems: 'center',

},

removeButton: {

marginLeft: 8,

color: 'red',

fontWeight: 'bold',

},

imagePicker: {

backgroundColor: '#ddd',

padding: 15,

borderRadius: 5,

marginBottom: 15,

alignItems: 'center',

},

previewImage: {

width: 100,

height: 100,

borderRadius: 5,

marginBottom: 15,

},

cadastrarButton: {

backgroundColor: 'green',

padding: 15,

borderRadius: 5,

alignItems: 'center',

marginBottom: 20,

},

cadastrarButtonText: {

color: '#fff',

fontWeight: 'bold',

fontSize: 18,

},

fornecedorItem: {

flexDirection: 'row',

padding: 15,

marginBottom: 10,

borderWidth: 1,

borderColor: '#ddd',

borderRadius: 5,

backgroundColor: '#fff',

alignItems: 'center',

},

fornecedorImage: {

width: 80,

height: 80,

borderRadius: 40,

marginRight: 15,

},

fornecedorInfo: {

flex: 1,

},

fornecedorNome: {

fontSize: 18,

fontWeight: 'bold',

marginBottom: 5,

},

});

export default CadastroFornecedores;