



**Universidade Estácio de Sá**

Polo Bangu - Rio de Janeiro

- 
- **Curso:** Desenvolvimento Full-Stack
  - **Disciplina:** Vamos criar um App
  - **Semestre:** 4
  - **Turma:** 2024.4
  - **Aluna:** Clara Martins Azevedo
- 

## **1. Missão Prática - Nível 1**

### **2. Objetivos da Missão Prática:**

- Configurar o ambiente de desenvolvimento React Native;
- Implementar a funcionalidade de entrada de texto em um componente React Native;
- Implementar um Componente de Lista Dinâmica (ScrollView);
- Implementar componentes React Native para exibir informações de forma dinâmica em listas;
- Empregar elementos visuais em um aplicativo React Native.

### 3. Códigos do Projeto:

- Cadastro.js:

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

```
const Cadastro = () => {

  const [nome, setNome] = useState("");
  const [endereco, setEndereco] = useState("");
  const [contato, setContato] = useState("");
  const [categorias, setCategorias] = useState("");
  const [image, setImage] = useState(null);

  const handleCadastro = () => {
    if (nome && endereco && contato && categorias && image) {
      setNome("");
      setEndereco("");
      setContato("");
      setCategorias("");
      setImage(null);
    }
  };

  return (
    <View style={styles.container}>
      <Text style={styles.title}>Cadastro</Text>
      <Text style={styles.label}>Nome</Text>
      <TextInput
        style={styles.input}
        value={nome}
        onChangeText={setNome}
      />
      <Text style={styles.label}>Endereço</Text>
      <TextInput
        style={styles.input}
        value={endereco}
        onChangeText={setEndereco}
      />
      <Text style={styles.label}>Contato</Text>
      <TextInput
        style={styles.input}
        value={contato}
        onChangeText={setContato}
      />
      <Text style={styles.label}>Categorias</Text>
      <TextInput
        style={styles.input}
        value={categorias}
        onChangeText={setCategorias}
      />
      <TouchableOpacity style={styles.button} onPress={handleCadastro}>
        <Text style={styles.buttonText}>Cadastrar</Text>
      </TouchableOpacity>
    </View>
  );
};
```

```

};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#e7ecef',
    justifyContent: 'center',
  },
  title: {
    backgroundColor: '#274c77',
    fontSize: 50,
    fontWeight: 'bold',
    color: 'ffffff',
    padding: 40,
    marginBottom: 40,
  },
  label: {
    fontSize: 20,
    marginLeft: 30,
    marginTop: 20,
    marginBottom: 5,
    color: '#31393c',
  },
  input: {
    height: 50,
    maxWidth: 800,
    borderColor: 'ffffff',
    borderWidth: 1,
    borderRadius: 8,
    marginLeft: 30,
    marginRight: 30,
    paddingHorizontal: 10,
    marginBottom: 20,
    backgroundColor: 'ffffff',
  },
  button: {
    backgroundColor: '#ea7317',
    width: 300,
    margin: 30,
    borderRadius: 8,
    marginTop: 30,
    marginBottom: 100,
    padding: 15,
    alignItems: 'center',
  },
  buttonText: {
    color: 'ffffff',
    fontSize: 20,
    fontWeight: 'bold',
  },
});

```

```
export default Cadastro;
```

## • Listagem.js:

```

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

const ListagemFornecedores = () => {

```

```

const [fornecedores, setFornecedores] = useState([]);

const [filter, setFilter] = useState("");
const [campoSelecioneado, setCampoSelecioneado] = useState("nome");

const filteredData = fornecedores.filter((fornecedor) =>
  fornecedor[campoSelecioneado]?.toLowerCase().includes(filter.toLowerCase())
);

return (
  <View style={styles.container}>
    <Text style={styles.title}>Fornecedores Cadastrados</Text>
    <TextInput
      style={styles.input}
      placeholder="Digite para procurar"
      value={filter}
      onChangeText={setFilter}
    />
    <View style={styles.buttonContainer}>
      {[ 'nome', 'endereco', 'contato', 'categorias' ].map((campo) => (
        <TouchableOpacity
          key={campo}
          style={[
            styles.button,
            campoSelecioneado === campo && styles.buttonSelected,
          ]}
          onPress={() => setCampoSelecioneado(campo)}
        >
          <Text style={styles.buttonText}>{campo}</Text>
        </TouchableOpacity>
      ))}
    </View>
    <FlatList
      data={filteredData}
      keyExtractor={(item) => item.id.toString()}
      renderItem={({ item }) => (
        <View style={styles.item}>
          <Text style={styles.itemText}>Nome: {item.nome}</Text>
          <Text style={styles.itemText}>Endereço: {item.endereco}</Text>
          <Text style={styles.itemText}>Contato: {item.contato}</Text>
          <Text style={styles.itemText}>Categoria: {item.categorias}</Text>
        </View>
      )}
      ListEmptyComponent={<Text style={styles.emptyText}>Nenhum fornecedor encontrado.</Text>}
    />
  </View>
);
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#e7ecef',
  },
  title: {
    fontSize: 40,
    fontWeight: 'bold',
    color: '#274c77',
    marginBottom: 20,
    marginLeft: 30,
    textAlign: 'start'
  },
  input: {
    height: 50,
    maxWidth: 800,
    borderColor: '#ffffff',
    borderWidth: 1,
    borderRadius: 8,

```

```

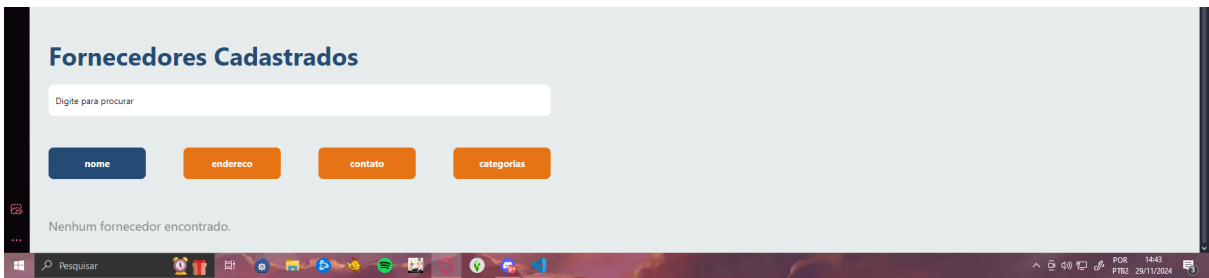
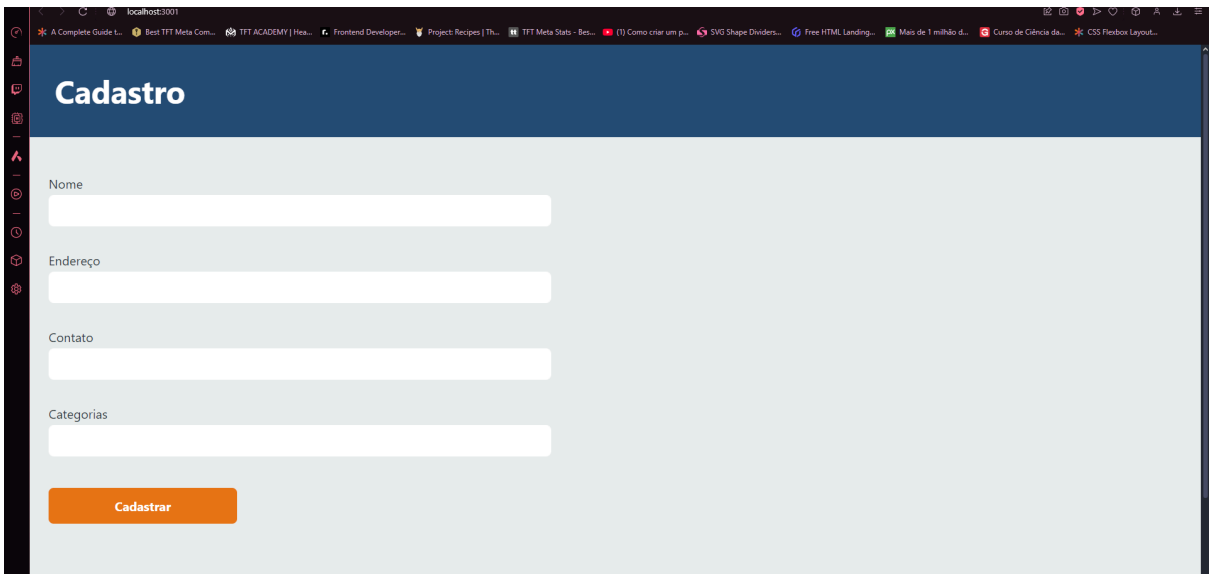
      marginLeft: 30,
      marginRight: 30,
      paddingHorizontal: 10,
      marginBottom: 20,
      backgroundColor: '#ffffff',
    },
    buttonContainer: {
      flexDirection: 'row',
      justifyContent: 'flex-start',
      marginBottom: 10
    },
    button: {
      backgroundColor: '#ea7317',
      width: 155,
      margin: 30,
      borderRadius: 8,
      marginTop: 30,
      padding: 15,
      alignItems: 'center',
    },
    buttonSelected: {
      backgroundColor: '#274c77'
    },
    buttonText: {
      color: '#fff',
      fontSize: 15,
      fontWeight: 'bold'
    },
    item: {
      backgroundColor: '#fff',
      padding: 10,
      marginBottom: 10,
      borderRadius: 8,
      borderWidth: 1,
      borderColor: '#ccc',
    },
    itemText: {
      fontSize: 20,
      color: '#31393c'
    },
    emptyText: {
      textAlign: 'start',
      color: '#888',
      fontSize: 20,
      margin: 30,
      marginTop: 20,
    },
  });

export default ListagemFornecedores;

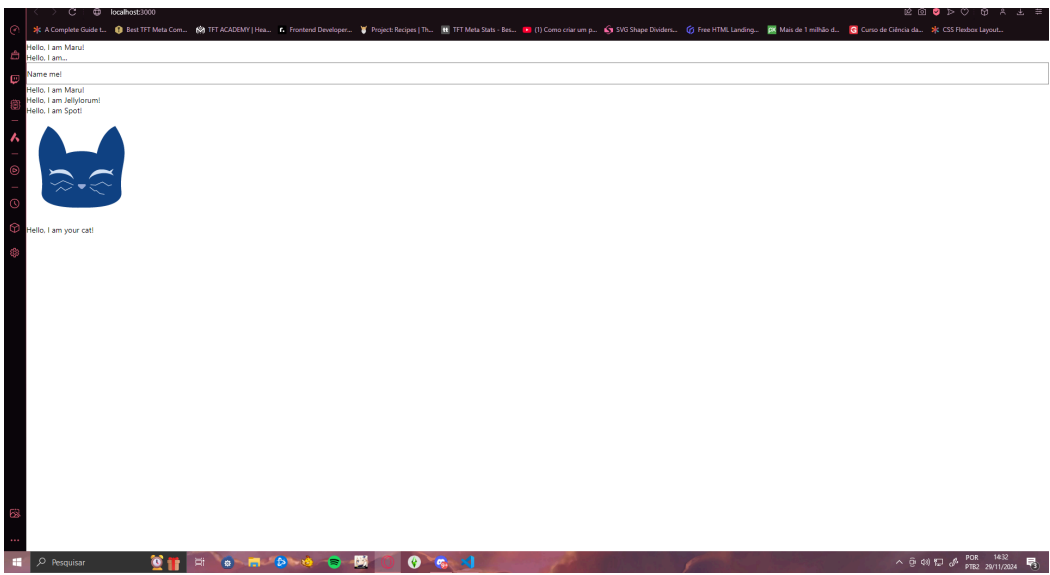
```

## 4. Resultado da execução do código:

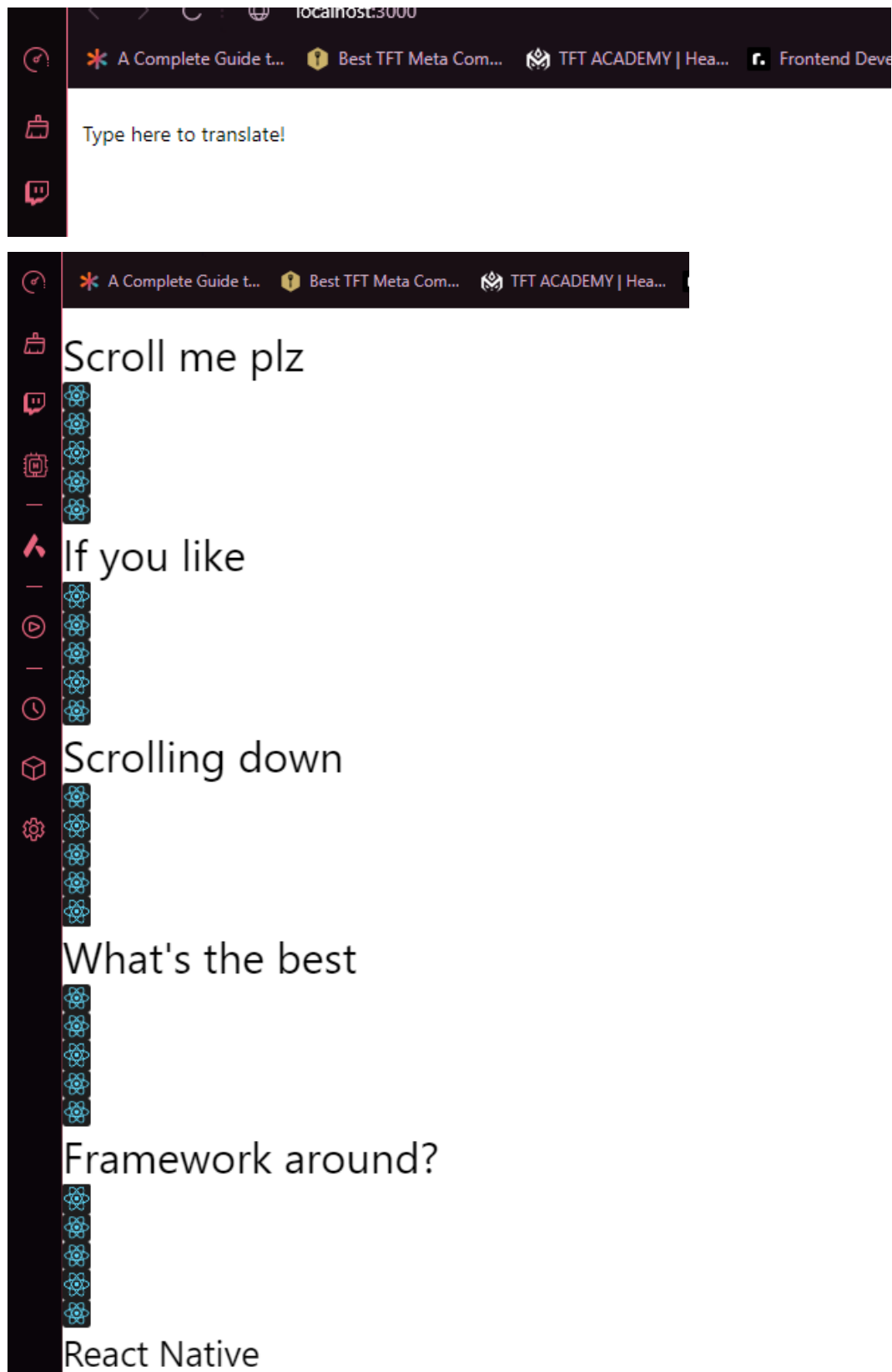
## 4.1 - App-Cadastro:



## 4.2 - Cat:



### 4.3 - Microatividades:



- 
- ▶ Devin
- 
- 🕒 Dan
- 📦 Dominic
- ⚙️ Jackson
- James
- Joel
- John
- Jillian
- Jimmy
- Julie

- 
- 🕒 **D**
- Devin
- 📦 Dan
- ⚙️ Dominic
- J**
- Jackson
- James
- Jillian
- Jimmy
- Joel
- John
- Julie



