

REACT NATIVE

Vamos criar um aplicativo em React Native para consumir dados através da API da ViaCEP.

O objetivo é obter recursos de uma URL (endereço de internet) remota, podemos realizar a requisição POST para consumir uma API-REST, ou trazer dados estáticos de um conteúdo estático de outro servidor.

O React Native possui uma API Fetch para realizar esta tarefa, ou podemos utilizar uma dependência chamada Axios ou outra similar.

O link de referência da API que vamos consumir é:

<https://viacep.com.br/>

Este é o material oficial da API Fetch:

https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch

Segue link da documentação oficial ReactNative.dev sobre conectividade:

<https://reactnative.dev/docs/network>

Utilizando este material como referência vamos criar um projeto novo:

Abra o Prompt de comando ou um Terminal e crie um novo projeto:

```
expo init FetchAPIExemplo
```

Escolha a primeira opção:

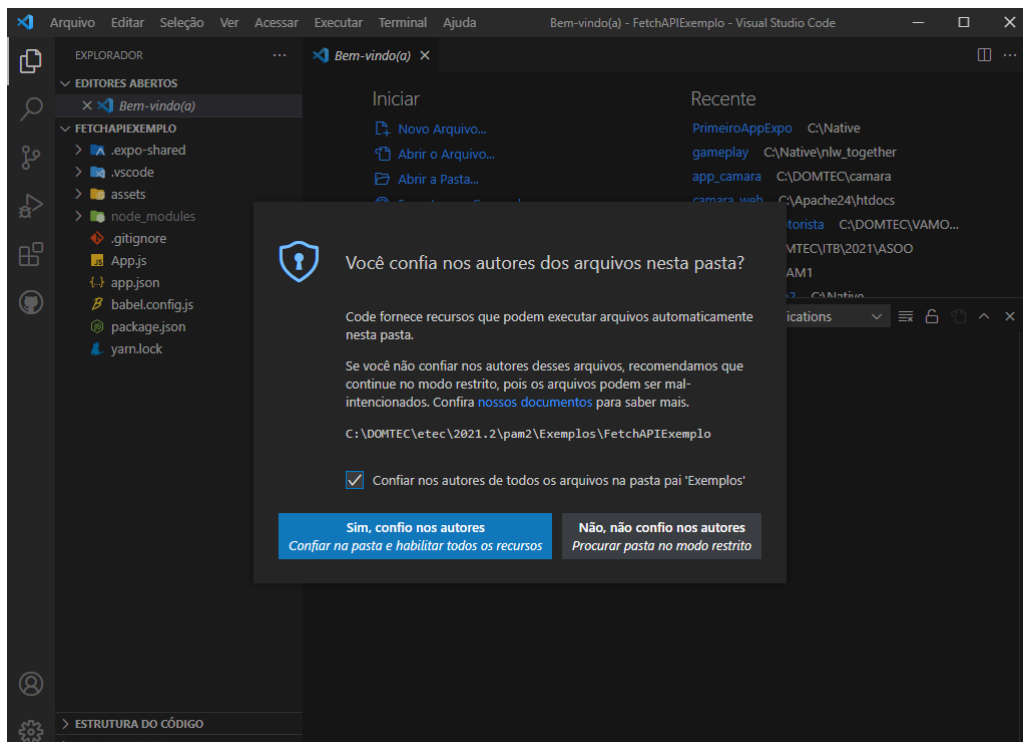
```
? Choose a template: » - Use arrow-keys. Return to submit.
  ----- Managed workflow -----
>  blank                a minimal app as clean as an empty
canvas
    blank (TypeScript)  same as blank but with TypeScript
configuration
    tabs (TypeScript)   several example screens and tabs using
react-navigation and TypeScript
  ----- Bare workflow -----
    minimal             bare and minimal, just the essentials
to get you started
```

Abra o projeto no VSCode através do comando abaixo:

```
code FetchAPIExemplo
```

No VSCode, abrir o Terminal com as teclas Alt + F12 e adicionar o pacote expo-camera no seu projeto:

```
expo install expo-camera
```



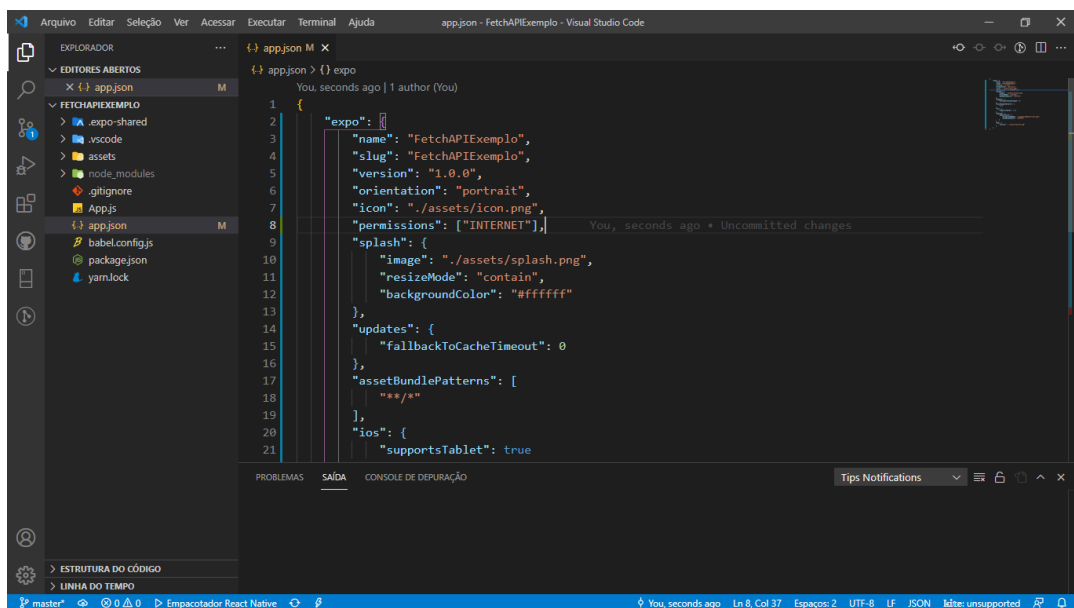
Confie nos autores, na pasta e habilite todos os recursos.

adicione as permissões no arquivo app.json:

<https://docs.expo.dev/versions/v42.0.0/config/app/#permissions>

```
"permissions": ["INTERNET"]
```

Conforme exemplo abaixo:



Vamos editar o arquivo App.js para receber os dados que serão capturados da API ViaCEP:

```
import { StatusBar } from 'expo-status-bar';
import React, { useEffect, useState } from 'react';
import { ActivityIndicator, SafeAreaView, ScrollView, StyleSheet, Text, TextInput, View, Button, Alert } from 'react-native';

export default function App() {
  const [ cep, alterarCep ] = React.useState("00000-000");
  const [ logradouro, alterarLogradouro ] = React.useState("");
  const [ complemento, alterarComplemento ] = React.useState("");
  const [ bairro, alterarBairro ] = React.useState("");
  const [ localidade, alterarLocalidade ] = React.useState("");
  const [ uf, alterarUf ] = React.useState("");
  const [ ibge, alterarIbge ] = React.useState("");
  const [ gia, alterarGia ] = React.useState("");
  const [ ddd, alterarDDD ] = React.useState("");
  const [ siafi, alterarSiafi ] = React.useState("");

  const [ isLoading, setLoading ] = React.useState(false);

  const buscarEndereco = async () => {
    try{
      const response = await fetch('https://viacep.com.br/ws/' + cep.toString().replace("-", "") + '/json/');
      const json = await response.json();
      alterarLogradouro(json.logradouro);
      alterarComplemento(json.complemento);
      alterarBairro(json.bairro);
      alterarLocalidade(json.localidade);
      alterarUf(json.uf);
      alterarIbge(json.ibge);
      alterarGia(json.gia);
      alterarDDD(json.ddd);
      alterarSiafi(json.siafi);
    } catch (error){
      Alert.alert(error);
    } finally {
      setLoading(false);
    }
  }

  useEffect(() => {
    buscarEndereco();
  })
}
```

```
useEffect(() => {
    buscarEndereco();
}, []);

return (
    <ScrollView>
    <SafeAreaView style = { styles.container }>

    <View >
    <Text style={styles.texto}> Entre com o CEP para pesquisar:
00000-000 </Text>
    <TextInput
        style = { styles.input }
        onChangeText = { alterarCep }
        value = { cep }
        keyboardType = "numeric"
        maxLength = {9}
        onFocus={()=>alterarCep("")}
    />
    {isLoading ? <ActivityIndicator/> :
    <Button style = { styles.botao }
        onPress={() => {
            //Alert.alert('Realizando busca do endereço...')

            setLoading(true);
            buscarEndereco();
        }
    }
    title="Consumindo ViaCEP API"
    color="#841584"
    accessibilityLabel="Aprenda mais sobre consumo de API RE
ST"
    />}
```

```
<TextInput
  style = { styles.inputLarge }
  onChangeText = { alterarLogradouro }
  value = { logradouro }
  keyboardType = "default"
  hintText = { "Logradouro" }
  multiline = {true} />
<TextInput
  style = { styles.input }
  onChangeText = { alterarComplemento }
  value = { complemento }
  keyboardType = "default" />
<TextInput
  style = { styles.input }
  onChangeText = { alterarBairro }
  value = { bairro }
  keyboardType = "default" />
<TextInput
  style = { styles.input }
  onChangeText = { alterarLocalidade }
  value = { localidade }
  keyboardType = "default" />
<TextInput
  style = { styles.input }
  onChangeText = { alterarUf }
  value = { uf }
  keyboardType = "default"
  maxLength = {2} />
<TextInput
  style = { styles.input }
  onChangeText = { alterarIbge }
  value = { ibge }
  keyboardType = "numeric" />
<TextInput
  style = { styles.input }
  onChangeText = { alterarGia }
  value = { gia }
  keyboardType = "numeric" />
```

```
<TextInput
  style = { styles.input }
  onChangeText = { alterarDDD }
  value = { ddd }
  keyboardType = "numeric" />
<TextInput
  style = { styles.input }
  onChangeText = { alterarSiafi }
  value = { siafi }
  keyboardType = "numeric" />
<StatusBar style = "auto" />
</View>
</SafeAreaView>
</ScrollView>

);
}
```

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#fff',
    margin: 16,
    marginTop: 48,
  },
  texto: {
    alignItems: 'center',
    justifyContent: 'center',
    alignSelf: 'center'
  },
  input: {
    height: 40,
    margin: 3,
    borderWidth: 1,
    padding: 1,
    width: "100%",
    justifyContent: "center",
    alignSelf: "center",
    maxWidth: 256
  },
  inputLarge: {
    height: 40,
    margin: 3,
    borderWidth: 1,
    padding: 1,
    width: "100%",
    justifyContent: "center",
    alignSelf: "center"
  },
  botao: {
    height: 40,
    margin: 3,
    borderWidth: 1,
    padding: 1,
    width: 50,
    borderRadius: 16
  },
});
```


No Terminal execute o código:

```
expo start
```

E teste o aplicativo.

8:14



Entre com o CEP para pesquisar: 00000-000

01010-000

CONSUMINDO VIA CEP API

Rua São Bento

até 318 - lado par

Centro

São Paulo

SP

3550308

1004

11

7107



Desafio:

Criar um projeto para consumir a API ViaCEP, melhorando o layout do professor. Acrescentar um campo para inserir o número do logradouro pesquisado e mostrar o resultado com todo o endereço completo em outra tela.

Referência Bibliográfica

[1] <https://reactnative.dev/>. Acessado em 10/08/2021.