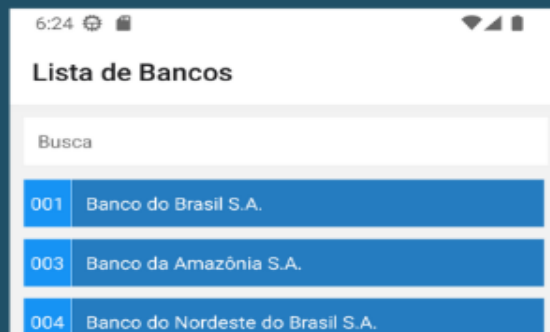


# 1. Criando uma busca com Array.filter

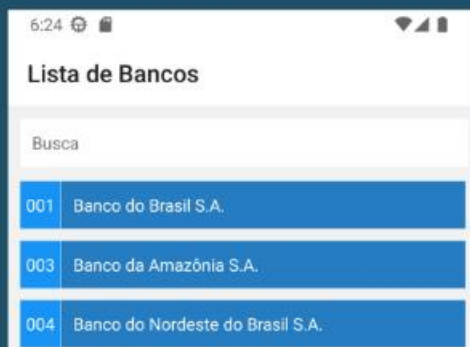
## Criando uma busca com filter

Apresentação

O exemplo desse flow é uma aplicação que **lista** e permite **buscar** por bancos.



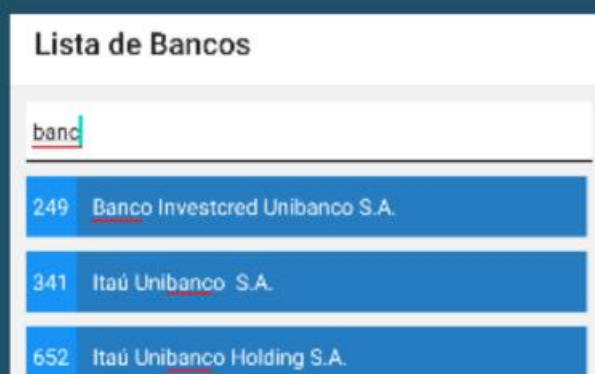
O App lista **diversos bancos** a partir de uma coleção de dados e os exibe com uso do **array.map**:



O App conta também com um **campo de busca**, onde o usuário pode **digitar o que quiser** para buscar:



Além de buscar por **nomes inteiros**, a busca também funciona com **textos parciais**, por exemplo:



Note que o texto 'banc' é encontrado em qualquer posição. Veja também que letras maiúsculas e minúsculas não fazem diferença.

Lista de Bancos	
<input type="text" value="banc"/>	
249	Banco Investcred Unibanco S.A.
341	Itaú Unibanco S.A.

## Criando uma busca com filter

Funcionamento do App em detalhes

O App inicia mostrando uma listagem de bancos.

Os dados dessa listagem vêm da importação de uma coleção de dados.

```
import bancos from "../../dados/bancos";
```

ListaBancos/index.js

Essa coleção é armazenada em uma **variável de estado** chamada **listaBancos**, para que possa ser manipulada posteriormente pela **busca**.

```
import bancos from "../..//dados/bancos";

export default function ListaBancos() {

  const [textoBusca, setTextoBusca] = useState("");
  const [listaBancos, setListaBancos] = useState(bancos);
}
```

ListaBancos/index.js

Essa **variável de estado** é utilizada pelo **Array.map** para listar os **bancos** na tela:

```
const [listaBancos, setListaBancos] = useState(bancos)
```

```
<ScrollView>
{
  listaBancos.map((banco) => {
    return (
      <ItemBanco
        codigo={banco.codigo}
        nome={banco.nome}
        key={banco.codigo}
      />
    )
  })
}
</ScrollView>
```



Quando o usuário digita algo no **campo de busca** uma função chamada **handleBusca** é chamada.

```
<TextInput
  style={styles.CampoBusca}
  placeholder="Busca"
  value={textoBusca}
  onChangeText={({termoDigitado}) => handleBusca(termoDigitado)}
/>
```

Lista de Bancos

Itaú

Essa função recebe o **texto digitado** como **parâmetro** e o utiliza para fazer a filtragem dos dados, buscando pelo **termoDigitado**.

```
function handleBusca(termoDigitado) {  
  const resultadoBusca = bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));  
  setListaBancos(resultadoBusca)  
  
  setTextoBusca(termoDigitado)  
}
```

Em seguida, o **resultado da busca** é utilizado para definir a variável de estado **listaBancos**, fazendo com que a lista seja **atualizada na tela**.

```
function handleBusca(termoDigitado) {  
  const resultadoBusca = bancos.filter((banco) =>  
    setListaBancos(resultadoBusca)  
  )  
  
  setTextoBusca(termoDigitado)  
}
```

Lista de Bancos	
Itaú	
029	Banco Itaú Consignado S.A.
184	Banco Itaú BBA S.A.
341	Itaú Unibanco S.A.

A função também define a variável de estado **TextoBusca**, que é usada para atualizar o valor do **campo de busca**:

```
function handleBusca(termoDigitado) {  
  const resultadoBusca = bancos.filter(  
    setListaBancos(resultadoBusca)  
  )  
  
  setTextoBusca(termoDigitado)  
}
```

```
<TextInput  
  style={styles.CampoBusca}  
  placeholder="Busca"  
  value={textoBusca}  
  onChangeText={({termoDigitado)  
  />
```

Lista de Bancos	
Itaú	
029	Banco Itaú Consignado S.A.
184	Banco Itaú BBA S.A.
341	Itaú Unibanco S.A.
652	Itaú Unibanco Holding S.A.

# Criando uma busca com filter

## Regras da busca

O filter busca pelo **termo digitado** dentro da **coleção de dados** de bancos e retorna o **resultado correspondente**.

```
function handleBusca(termoDigitado) {  
  const resultadoBusca = bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));  
  setListaBancos(resultadoBusca)  
  setTextoBusca(termoDigitado)  
}
```

A screenshot of a web application's search interface. At the top, there is a search bar containing the text 'Itaú'. Below the search bar, a list of search results is displayed. The first result is '029 Banco Itaú Consignado S.A.' and the second is '184 Banco Itaú BBA S.A.'. The word 'Itaú' in both the search bar and the search results is highlighted with a red box.

029	Banco Itaú Consignado S.A.
184	Banco Itaú BBA S.A.

Porém, note que a **filtragem** possui algumas regras como **toLowerCase** e **includes**.

```
const resultadoBusca = bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));  
setListaBancos(resultadoBusca)
```

`toLowerCase` serve para transformar todas as letras da `string` que a chamou em minúsculas.

```
const resultadoBusca = bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));  
setListaBancos(resultadoBusca)
```

Ex:

Itaú -> itaú

Banco do Brasil -> banco do brasil

Isso é feito para que o `termo digitado` pelo usuário seja `encontrado` independente de letras maiúsculas ou minúsculas:

Lista de Bancos	
Itaú	
029	Banco Itaú Consignado S.A.
184	Banco Itaú BBA S.A.
341	Itaú Unibanco S.A.

Em seguida usamos o `includes`, que serve para verificar se um determinado `trecho` de texto `existe` dentro de uma `string`.

```
const resultadoBusca = bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));  
setListaBancos(resultadoBusca)
```

Ex:

Busca por: `It` -> Encontra `Itau`, `Capital`

Uma dúvida que pode surgir é: Por que usar **includes** e não **===** ?

Veja o motivo nas imagens abaixo:

Lista de Bancos	
Ita	
029	Banco Itaú Consignado S.A.
180	CM Capital Markets Corretora de Câmbio, Títulos e Valores Mobiliários Ltda.
184	Banco Itaú BBA S.A.
341	Itaú Unibanco S.A.

Lista de Bancos	
Ita	

Isso ocorre pois o **===** verifica a palavra inteira, já o **includes** verifica também **trechos**.

Lista de Bancos	
Ita	
029	Banco Itaú Consignado S.A.
180	CM Capital Markets Corretora de Câmbio, Títulos e Valores Mobiliários Ltda.
184	Banco Itaú BBA S.A.
341	Itaú Unibanco S.A.
412	Banco Capital S. A.

Lista de Bancos	
Ita	



Includes

===

Fazendo isso, o app vai **retornar os bancos** que contenham o **termo digitado**, não importa a posição ou a forma que a palavra for escrita.

Lista de Bancos	
band	
249	Banco Investcred Unibanco S.A.
341	Itaú Unibanco S.A.



#### Códigos 4 a 7.

```
1  const bancos = [  
2    {  
3      codigo: "001",  
4      nome: "Banco do Brasil S.A."  
5    },  
6    {  
7      codigo: "003",  
8      nome: "Banco da Amazônia S.A."  
9    },  
10   {  
11     codigo: "004",  
12     nome: "Banco do Nordeste do Brasil S.A."  
13   },  
14   {  
15     codigo: "007",  
16     nome: "Banco Nacional de Desenvolvimento Econômico e Social BNDES"  
17   }  
18 ]  
19 export default bancos;
```

Código 4. dados/bancos.js (conteúdo parcial)

```
1  import bancos from "../dados/bancos";  
2  
3  export function retornaBancos() {  
4    return bancos;  
5  }  
6  
7  export function buscaBanco(termoDigitado) {  
8    return bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigi  
9  )
```

Código 5. servicos/bancos.js

```
1  import React from 'react'  
2  import { View, Text } from 'react-native'  
3  import styles from './styles'  
4  
5  export default function ItemBanco({codigo, nome}) {  
6    return (  
7      <View style={styles.Item}>  
8        <Text style={styles.Ano}></Text>  
9        <Text style={styles.Texto}></Text>  
10     </View>  
11   )  
12 }
```

```

1  import React, { useState } from "react";
2  import { ScrollView, View, TextInput } from 'react-native'
3
4  import { buscaBanco, retornaBancos } from "../../servicos/busca";
5  import ItemBanco from "../ItemBanco";
6
7  import styles from "./styles";
8
9  export default function ListaBancos() {
10
11     const [textoBusca, setTextoBusca] = useState("");
12     const [listaBancos, setListaBancos] = useState(retornaBancos())
13
14     function handleBusca(termoDigitado) {
15         setTextoBusca(termoDigitado)
16
17         const resultadoBusca = buscaBanco(termoDigitado);
18         setListaBancos(resultadoBusca)
19     }
20
21     return (
22         <View style={styles.Container}>
23
24             <TextInput
25                 style={styles.CampoBusca}
26                 placeholder="Busca"
27                 value=
28                 onChangeText={({termoDigitado}) => handleBusca(termoDigitado)}
29             />
30
31             <ScrollView>
32                 {
33                     listaBancos.map((banco) => {
34                         return (
35                             <ItemBanco
36                                 codigo={banco.codigo}
37                                 nome={banco.nome}
38                                 key={banco.codigo}
39                             />
40                         )
41                     })
42                 }
43             </ScrollView>
44         </View>
45     );
46 }

```

Código 7. ListaBancos/index.js

# Criando uma busca com filter

## Refatorando o código - Camada de Serviço

Vimos anteriormente que o **filtro da busca** fica dentro do componente **ListaBancos**.

```
function handleBusca(termoDigitado) {  
  const resultadoBusca = bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));  
  setListaBancos(resultadoBusca)  
  setTextoBusca(termoDigitado)  
}
```

O que faremos nesse flow é criar uma **Camada de Serviço** para conter toda a lógica da aplicação, deixando o **código mais organizado**.

Para isso, criamos uma pasta chamada **servicos** e colocamos dentro dela um arquivo **busca.js** com o seguinte código:

```
import bancos from "../dados/bancos";

export function retornaBancos() {
  return bancos;
}

export function buscaBanco(termoDigitado) {
  return bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));
}
```

A primeira função serve para retornar a **lista completa** de bancos.

```
import bancos from "../dados/bancos";

export function retornaBancos() {
  return bancos;
}

export function buscaBanco(termoDigitado) {
  return bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));
}
```

Já a **segunda** é o mesmo **filtro de busca** que vimos anteriormente no componente **ListaBancos**.

```
import bancos from "../dados/bancos";

export function retornaBancos() {
  return bancos;
}

export function buscaBanco(termoDigitado) {
  return bancos.filter((banco) => banco.nome.toLowerCase().includes(termoDigitado.toLowerCase()));
}
```

Com isso feito, começaremos utilizando a função **retornaBancos** para definir a variável de estado **listaBancos**.

```
import bancos from "../dados/bancos";  
  
export function retornaBancos() {  
  return bancos;  
}
```

servicos/busca.js

```
import { buscaBanco, retornaBancos } from "../../servicos/busca";  
export default function ListaBancos() {  
  const [textoBusca, setTextoBusca] = useState("");  
  const [listaBancos, setListaBancos] = useState(retornaBancos());  
}
```

ListaBancos/index.js



Note que como não acessamos mais a coleção diretamente podemos **remover a linha abaixo**:

```
import bancos from "../../dados/bancos";
```

ListaBancos/index.js

Com isso, o **código de interface** não precisa mais importar a **camada de dados**, o que deixa o **código** melhor dividido e **organizado**

A **segunda** e última alteração é substituir o **filter** dentro de **handleBusca** pela função **buscaBanco** que criamos na **Camada de Serviço**:

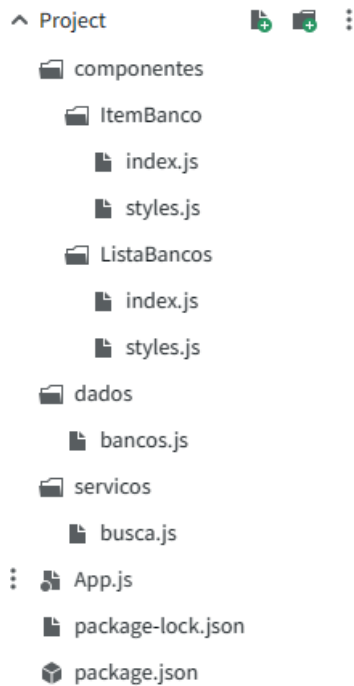
```
import { buscaBanco, retornaBancos } from "../../servicos/busca";  
import ItemBanco from "../ItemBanco";
```

```
function handleBusca(termoDigitado) {  
  setTextoBusca(termoDigitado)  
  
  const resultadoBusca = buscaBanco(termoDigitado);  
  setListaBancos(resultadoBusca)  
}
```

Pronto! Perceba que sua aplicação continua funcionando **igual antes**, mas o **código** ficou **mais organizado**:

O **código de lógica** e o **código de interface** separados.

## ESTRUTURA DO PROJETO



### Dependências

```
{
  "dependencies": {
    "expo-status-bar": "~1.4.2",
    "react-native-screens": "~3.18.0",
    "@react-navigation/stack": "^6.3.16",
    "@react-navigation/native": "^6.1.6",
    "react-native-gesture-handler": "~2.8.0",
    "react-native-safe-area-context": "4.4.1"
  }
}
```

### Bancos.js

```
const bancos = [
  {
    "codigo": "001",
    "nome": "Banco do Brasil S.A."
  },
  {
    "codigo": "003",
    "nome": "Banco da Amazônia S.A."
  },
]
```

```
{
  "codigo": "004",
  "nome": "Banco do Nordeste do Brasil S.A."
},
{
  "codigo": "007",
  "nome": "Banco Nacional de Desenvolvimento Econômico e Social BNDES"
},
{
  "codigo": "010",
  "nome": "Credicoamo Crédito Rural Cooperativa"
},
{
  "codigo": "011",
  "nome": "Credit Suisse Hedging-Griffo Corretora de Valores S.A."
},
{
  "codigo": "012",
  "nome": "Banco Inbursa S.A."
},
{
  "codigo": "014",
  "nome": "Natixis Brasil S.A. Banco Múltiplo"
},
{
  "codigo": "015",
  "nome": "UBS Brasil Corretora de Câmbio, Títulos e Valores
Mobiliários S.A."
},
{
  "codigo": "016",
  "nome": "Coop de Créd. Mútuo dos Despachantes de Trânsito de SC e Rio
Grande do Sul"
},
{
  "codigo": "017",
  "nome": "BNY Mellon Banco S.A."
},
{
  "codigo": "018",
  "nome": "Banco Tricury S.A."
},
{
  "codigo": "021",
  "nome": "Banestes S.A. Banco do Estado do Espírito Santo"
},
{
  "codigo": "024",
  "nome": "Banco Bandepe S.A."
}
```



```
},
{
  "codigo": "025",
  "nome": "Banco Alfa S.A."
},
{
  "codigo": "029",
  "nome": "Banco Itaú Consignado S.A."
},
{
  "codigo": "033",
  "nome": "Banco Santander (Brasil) S. A."
},
{
  "codigo": "036",
  "nome": "Banco Bradesco BBI S.A."
},
{
  "codigo": "037",
  "nome": "Banco do Estado do Pará S.A."
},
{
  "codigo": "040",
  "nome": "Banco Cargill S.A."
},
{
  "codigo": "041",
  "nome": "Banco do Estado do Rio Grande do Sul S.A."
},
{
  "codigo": "047",
  "nome": "Banco do Estado de Sergipe S.A."
},
{
  "codigo": "060",
  "nome": "Confidence Corretora de Câmbio S.A."
},
{
  "codigo": "062",
  "nome": "Hipercard Banco Múltiplo S.A."
},
{
  "codigo": "063",
  "nome": "Banco Bradescard S.A."
},
{
  "codigo": "064",
  "nome": "Goldman Sachs do Brasil Banco Múltiplo S. A."
},
}
```

```
{
  "codigo": "065",
  "nome": "Banco AndBank (Brasil) S.A."
},
{
  "codigo": "066",
  "nome": "Banco Morgan Stanley S. A."
},
{
  "codigo": "069",
  "nome": "Banco Crefisa S.A."
},
{
  "codigo": "070",
  "nome": "Banco de Brasília S.A."
},
{
  "codigo": "074",
  "nome": "Banco J. Safra S.A."
},
{
  "codigo": "075",
  "nome": "Banco ABN Amro S.A."
},
{
  "codigo": "076",
  "nome": "Banco KDB do Brasil S.A."
},
{
  "codigo": "077",
  "nome": "Banco Inter S.A."
},
{
  "codigo": "078",
  "nome": "Haitong Banco de Investimento do Brasil S.A."
},
{
  "codigo": "079",
  "nome": "Banco Original do Agronegócio S.A."
},
{
  "codigo": "080",
  "nome": "BT Corretora de Câmbio Ltda."
},
{
  "codigo": "081",
  "nome": "BBN Banco Brasileiro de Negocios S.A."
}
```

```
"codigo": "082",
"nome": "Banco Topazio S.A."
},
{
  "codigo": "083",
  "nome": "Banco da China Brasil S.A."
},
{
  "codigo": "084",
  "nome": "Uniprime Norte do Paraná - Cooperativa de Crédito Ltda."
},
{
  "codigo": "085",
  "nome": "Cooperativa Central de Crédito Urbano - Cecred"
},
{
  "codigo": "089",
  "nome": "Cooperativa de Crédito Rural da Região da Mogiana"
},
{
  "codigo": "091",
  "nome": "Central de Cooperativas de Economia e Crédito Mútuo do Est
RS - Unicred"
},
{
  "codigo": "092",
  "nome": "BRK S.A. Crédito, Financiamento e Investimento"
},
{
  "codigo": "093",
  "nome": "Pólocred Sociedade de Crédito ao Microempreendedor e à
Empresa de Pequeno Porte"
},
{
  "codigo": "094",
  "nome": "Banco Finaxis S.A."
},
{
  "codigo": "095",
  "nome": "Banco Confidence de Câmbio S.A."
},
{
  "codigo": "096",
  "nome": "Banco BMFBovespa de Serviços de Liquidação e Custódia S/A"
},
{
  "codigo": "097",
  "nome": "Cooperativa Central de Crédito Noroeste Brasileiro Ltda -
CentralCredi"
```

```
},
{
  "codigo": "098",
  "nome": "Credialiança Cooperativa de Crédito Rural"
},
{
  "codigo": "099",
  "nome": "Uniprime Central - Central Interestadual de Cooperativas de
Crédito Ltda."
},
{
  "codigo": "100",
  "nome": "Planner Corretora de Valores S.A."
},
{
  "codigo": "101",
  "nome": "Renascença Distribuidora de Títulos e Valores Mobiliários
Ltda."
},
{
  "codigo": "102",
  "nome": "XP Investimentos Corretora de Câmbio Títulos e Valores
Mobiliários S.A."
},
{
  "codigo": "104",
  "nome": "Caixa Econômica Federal"
},
{
  "codigo": "105",
  "nome": "Lecca Crédito, Financiamento e Investimento S/A"
},
{
  "codigo": "107",
  "nome": "Banco Bocom BBM S.A."
},
{
  "codigo": "108",
  "nome": "PortoCred S.A. Crédito, Financiamento e Investimento"
},
{
  "codigo": "111",
  "nome": "Oliveira Trust Distribuidora de Títulos e Valores
Mobiliários S.A."
},
{
  "codigo": "113",
  "nome": "Magliano S.A. Corretora de Cambio e Valores Mobiliarios"
},
}
```

```
{
  "codigo": "114",
  "nome": "Central Cooperativa de Crédito no Estado do Espírito Santo - CECOOP"
},
{
  "codigo": "117",
  "nome": "Advanced Corretora de Câmbio Ltda."
},
{
  "codigo": "118",
  "nome": "Standard Chartered Bank (Brasil) S.A. Banco de Investimento"
},
{
  "codigo": "119",
  "nome": "Banco Western Union do Brasil S.A."
},
{
  "codigo": "120",
  "nome": "Banco Rodobens SA"
},
{
  "codigo": "121",
  "nome": "Banco Agibank S.A."
},
{
  "codigo": "122",
  "nome": "Banco Bradesco BERJ S.A."
},
{
  "codigo": "124",
  "nome": "Banco Woori Bank do Brasil S.A."
},
{
  "codigo": "125",
  "nome": "Brasil Plural S.A. Banco Múltiplo"
},
{
  "codigo": "126",
  "nome": "BR Partners Banco de Investimento S.A."
},
{
  "codigo": "127",
  "nome": "Codepe Corretora de Valores e Câmbio S.A."
},
{
  "codigo": "128",
  "nome": "MS Bank S.A. Banco de Câmbio"
},
}
```

```
{
  "codigo": "129",
  "nome": "UBS Brasil Banco de Investimento S.A."
},
{
  "codigo": "130",
  "nome": "Caruana S.A. Sociedade de Crédito, Financiamento e
Investimento"
},
{
  "codigo": "131",
  "nome": "Tullett Prebon Brasil Corretora de Valores e Câmbio Ltda."
},
{
  "codigo": "132",
  "nome": "ICBC do Brasil Banco Múltiplo S.A."
},
{
  "codigo": "133",
  "nome": "Confederação Nacional das Cooperativas Centrais de Crédito e
Economia Familiar e"
},
{
  "codigo": "134",
  "nome": "BGC Liquidez Distribuidora de Títulos e Valores Mobiliários
Ltda."
},
{
  "codigo": "135",
  "nome": "Gradual Corretora de Câmbio, Títulos e Valores Mobiliários
S.A."
},
{
  "codigo": "136",
  "nome": "Confederação Nacional das Cooperativas Centrais Unicred Ltda
- Unicred do Brasil"
},
{
  "codigo": "137",
  "nome": "Multimoney Corretora de Câmbio Ltda"
},
{
  "codigo": "138",
  "nome": "Get Money Corretora de Câmbio S.A."
},
{
  "codigo": "139",
  "nome": "Intesa Sanpaulo Brasil S.A. - Banco Múltiplo"
},
}
```

```
{
  "codigo": "140",
  "nome": "Easynvest - Título Corretora de Valores SA"
},
{
  "codigo": "142",
  "nome": "Broker Brasil Corretora de Câmbio Ltda."
},
{
  "codigo": "143",
  "nome": "Treviso Corretora de Câmbio S.A."
},
{
  "codigo": "144",
  "nome": "Bexs Banco de Câmbio S.A."
},
{
  "codigo": "145",
  "nome": "Levycom - Corretora de Câmbio e Valores Ltda."
},
{
  "codigo": "146",
  "nome": "Guitta Corretora de Câmbio Ltda."
},
{
  "codigo": "149",
  "nome": "Facta Financeira S.A. - Crédito Financiamento e
Investimento"
},
{
  "codigo": "157",
  "nome": "ICAP do Brasil Corretora de Títulos e Valores Mobiliários
Ltda."
},
{
  "codigo": "159",
  "nome": "Casa do Crédito S.A. Sociedade de Crédito ao
Microempreendedor"
},
{
  "codigo": "163",
  "nome": "Commerzbank Brasil S.A. - Banco Múltiplo"
},
{
  "codigo": "169",
  "nome": "Banco Olé Bonsucesso Consignado S.A."
},
{
  "codigo": "172",
```

```
    "nome": "Albatross Corretora de Câmbio e Valores S.A"
  },
  {
    "codigo": "173",
    "nome": "BRL Trust Distribuidora de Títulos e Valores Mobiliários
S.A."
  },
  {
    "codigo": "174",
    "nome": "Pernambucas Financiadora S.A. Crédito, Financiamento e
Investimento"
  },
  {
    "codigo": "177",
    "nome": "Guide Investimentos S.A. Corretora de Valores"
  },
  {
    "codigo": "180",
    "nome": "CM Capital Markets Corretora de Câmbio, Títulos e Valores
Mobiliários Ltda."
  },
  {
    "codigo": "182",
    "nome": "Dacasa Financeira S/A - Sociedade de Crédito, Financiamento
e Investimento"
  },
  {
    "codigo": "183",
    "nome": "Socred S.A. - Sociedade de Crédito ao Microempreendedor"
  },
  {
    "codigo": "184",
    "nome": "Banco Itaú BBA S.A."
  },
  {
    "codigo": "188",
    "nome": "Ativa Investimentos S.A. Corretora de Títulos Câmbio e
Valores"
  },
  {
    "codigo": "189",
    "nome": "HS Financeira S/A Crédito, Financiamento e Investimentos"
  },
  {
    "codigo": "190",
    "nome": "Cooperativa de Economia e Crédito Mútuo dos Servidores
Públicos Estaduais do Rio"
  },
  {
```



```
    "codigo": "191",
    "nome": "Nova Futura Corretora de Títulos e Valores Mobiliários
Ltda."
  },
  {
    "codigo": "194",
    "nome": "Parmetal Distribuidora de Títulos e Valores Mobiliários
Ltda."
  },
  {
    "codigo": "196",
    "nome": "Fair Corretora de Câmbio S.A."
  },
  {
    "codigo": "197",
    "nome": "Stone Pagamentos S.A."
  },
  {
    "codigo": "204",
    "nome": "Banco Bradesco Cartões S.A."
  },
  {
    "codigo": "208",
    "nome": "Banco BTG Pactual S.A."
  },
  {
    "codigo": "212",
    "nome": "Banco Original S.A."
  },
  {
    "codigo": "213",
    "nome": "Banco Arbi S.A."
  },
  {
    "codigo": "217",
    "nome": "Banco John Deere S.A."
  },
  {
    "codigo": "218",
    "nome": "Banco BS2 S.A."
  },
  {
    "codigo": "222",
    "nome": "Banco Credit Agrícola Brasil S.A."
  },
  {
    "codigo": "224",
    "nome": "Banco Fibra S.A."
  },
  },
```

```
{
  "codigo": "233",
  "nome": "Banco Cifra S.A."
},
{
  "codigo": "237",
  "nome": "Banco Bradesco S.A."
},
{
  "codigo": "241",
  "nome": "Banco Clássico S.A."
},
{
  "codigo": "243",
  "nome": "Banco Máxima S.A."
},
{
  "codigo": "246",
  "nome": "Banco ABC Brasil S.A."
},
{
  "codigo": "249",
  "nome": "Banco Investcred Unibanco S.A."
},
{
  "codigo": "250",
  "nome": "BCV - Banco de Crédito e Varejo S/A"
},
{
  "codigo": "253",
  "nome": "Bexs Corretora de Câmbio S/A"
},
{
  "codigo": "254",
  "nome": "Parana Banco S. A."
},
{
  "codigo": "260",
  "nome": "Nu Pagamentos S.A."
},
{
  "codigo": "265",
  "nome": "Banco Fator S.A."
},
{
  "codigo": "266",
  "nome": "Banco Cédula S.A."
},
}
```

```
"codigo": "268",
"nome": "Barigui Companhia Hipotecária"
},
{
  "codigo": "269",
  "nome": "HSBC Brasil S.A. Banco de Investimento"
},
{
  "codigo": "271",
  "nome": "IB Corretora de Câmbio, Títulos e Valores Mobiliários Ltda."
},
{
  "codigo": "300",
  "nome": "Banco de la Nacion Argentina"
},
{
  "codigo": "318",
  "nome": "Banco BMG S.A."
},
{
  "codigo": "320",
  "nome": "China Construction Bank (Brasil) Banco Múltiplo S/A"
},
{
  "codigo": "341",
  "nome": "Itaú Unibanco S.A."
},
{
  "codigo": "366",
  "nome": "Banco Société Générale Brasil S.A."
},
{
  "codigo": "370",
  "nome": "Banco Mizuho do Brasil S.A."
},
{
  "codigo": "376",
  "nome": "Banco J. P. Morgan S. A."
},
{
  "codigo": "389",
  "nome": "Banco Mercantil do Brasil S.A."
},
{
  "codigo": "394",
  "nome": "Banco Bradesco Financiamentos S.A."
},
{
  "codigo": "399",
```

```
    "nome": "Kirtton Bank S.A. - Banco Múltiplo"
  },
  {
    "codigo": "412",
    "nome": "Banco Capital S. A."
  },
  {
    "codigo": "422",
    "nome": "Banco Safra S.A."
  },
  {
    "codigo": "456",
    "nome": "Banco MUFG Brasil S.A."
  },
  {
    "codigo": "464",
    "nome": "Banco Sumitomo Mitsui Brasileiro S.A."
  },
  {
    "codigo": "473",
    "nome": "Banco Caixa Geral - Brasil S.A."
  },
  {
    "codigo": "477",
    "nome": "Citibank N.A."
  },
  {
    "codigo": "479",
    "nome": "Banco ItauBank S.A."
  },
  {
    "codigo": "487",
    "nome": "Deutsche Bank S.A. - Banco Alemão"
  },
  {
    "codigo": "488",
    "nome": "JPMorgan Chase Bank, National Association"
  },
  {
    "codigo": "492",
    "nome": "ING Bank N.V."
  },
  {
    "codigo": "494",
    "nome": "Banco de La Republica Oriental del Uruguay"
  },
  {
    "codigo": "495",
    "nome": "Banco de La Provincia de Buenos Aires"
```

```
},
{
  "codigo": "505",
  "nome": "Banco Credit Suisse (Brasil) S.A."
},
{
  "codigo": "545",
  "nome": "Senso Corretora de Câmbio e Valores Mobiliários S.A."
},
{
  "codigo": "600",
  "nome": "Banco Luso Brasileiro S.A."
},
{
  "codigo": "604",
  "nome": "Banco Industrial do Brasil S.A."
},
{
  "codigo": "610",
  "nome": "Banco VR S.A."
},
{
  "codigo": "611",
  "nome": "Banco Paulista S.A."
},
{
  "codigo": "612",
  "nome": "Banco Guanabara S.A."
},
{
  "codigo": "613",
  "nome": "Omni Banco S.A."
},
{
  "codigo": "623",
  "nome": "Banco Pan S.A."
},
{
  "codigo": "626",
  "nome": "Banco Ficsa S. A."
},
{
  "codigo": "630",
  "nome": "Banco Intercap S.A."
},
{
  "codigo": "633",
  "nome": "Banco Rendimento S.A."
},
}
```

```
{
  "codigo": "634",
  "nome": "Banco Triângulo S.A."
},
{
  "codigo": "637",
  "nome": "Banco Sofisa S. A."
},
{
  "codigo": "641",
  "nome": "Banco Alvorada S.A."
},
{
  "codigo": "643",
  "nome": "Banco Pine S.A."
},
{
  "codigo": "652",
  "nome": "Itaú Unibanco Holding S.A."
},
{
  "codigo": "653",
  "nome": "Banco Indusval S. A."
},
{
  "codigo": "654",
  "nome": "Banco A. J. Renner S.A."
},
{
  "codigo": "655",
  "nome": "Banco Votorantim S.A."
},
{
  "codigo": "707",
  "nome": "Banco Daycoval S.A."
},
{
  "codigo": "712",
  "nome": "Banco Ourinvest S.A."
},
{
  "codigo": "719",
  "nome": "Banif - Bco Internacional do Funchal (Brasil) S.A."
},
{
  "codigo": "735",
  "nome": "Banco Neon S.A."
}
```

```
"codigo": "739",
"nome": "Banco Cetelem S.A."
},
{
  "codigo": "741",
  "nome": "Banco Ribeirão Preto S.A."
},
{
  "codigo": "743",
  "nome": "Banco Semear S.A."
},
{
  "codigo": "745",
  "nome": "Banco Citibank S.A."
},
{
  "codigo": "746",
  "nome": "Banco Modal S.A."
},
{
  "codigo": "747",
  "nome": "Banco Rabobank International Brasil S.A."
},
{
  "codigo": "748",
  "nome": "Banco Cooperativo Sicredi S. A."
},
{
  "codigo": "751",
  "nome": "Scotiabank Brasil S.A. Banco Múltiplo"
},
{
  "codigo": "752",
  "nome": "Banco BNP Paribas Brasil S.A."
},
{
  "codigo": "753",
  "nome": "Novo Banco Continental S.A. - Banco Múltiplo"
},
{
  "codigo": "754",
  "nome": "Banco Sistema S.A."
},
{
  "codigo": "755",
  "nome": "Bank of America Merrill Lynch Banco Múltiplo S.A."
},
{
  "codigo": "756",
```

```
    "nome": "Banco Cooperativo do Brasil S/A - Bancoob"
  },
  {
    "codigo": "757",
    "nome": "Banco Keb Hana do Brasil S. A."
  }
]

export default bancos;
```

#### busca.js

```
import bancos from "../dados/bancos";

export default function retornaBusca(termo) {
  const resultado = bancos.filter((banco) =>
    banco.nome.toLowerCase().includes(termo.toLowerCase()) ||
    banco.codigo.includes(termo));

  return resultado
}
```

#### ItemBanco\index.js

```
import React from 'react'
import { View, Text } from 'react-native'
import styles from './styles'

export default function ItemBanco({codigo, nome}) {
  return (
    <View style={styles.Item}>
      <Text style={styles.Codigo}>{codigo}</Text>
      <Text style={styles.Banco}>{nome}</Text>
    </View>
  )
}
```



# ItemBanco\styles.js

```
import { StyleSheet } from "react-native";

const styles = StyleSheet.create({

  Item : {
    flexDirection: 'row',
    marginBottom: 10,
  },
  Codigo : {
    backgroundColor: '#1693f4',
    color: 'ffffff',
    padding: 5,
    textAlign: 'center',
    textAlignVertical: 'center'
  },
  Banco : {
    width: '90%',
    padding: 10,
    flexWrap: 'wrap',

    textAlignVertical: 'center',
    color: 'ffffff',
    backgroundColor: '#257cc0',
  }
})

export default styles
```

## ListaBancos\Index.js

```
import React, { useState } from "react";
import { ScrollView, TextInput, View } from 'react-native'
import bancos from "../dados/bancos";
import ItemBanco from "../ItemBanco";
import styles from "../styles";
import retornaBusca from "../servicos/busca";

function ListaBancos() {
  const [termo, setTermo] = useState("");
  const [listabancos, setListabancos] = useState(bancos)

  function handleBusca(termo) {
    const resultadoBusca = retornaBusca(termo)

    setTermo(termo)
    setListabancos(resultadoBusca);
  }

  return (
    <View style={styles.Container}>
      <TextInput
        style={styles.Busca}
        placeholder="Buscar Banco"
        onChangeText={(termoDigitado) => {
          handleBusca(termoDigitado);
        }}
        value={termo}
      />
      <ScrollView>
        {
          listabancos.map((banco, index) => {
            return (
              <ItemBanco key={index} codigo={banco.codigo}
nome={banco.nome} />
            );
          })
        }
      </ScrollView>
    </View>
  );
}

export default ListaBancos;
```

ListaBancos\styles.js

```
import { StyleSheet } from "react-native";

const styles = StyleSheet.create({

  Container : {
    padding: 10,
  },
  Busca : {
    marginBottom: 10,
    padding: 5,
    height: 40,
    backgroundColor: '#fff',
    borderBottomColor: '#000',
    borderBottomWidth: 1
  }
})

export default styles
```

**Requisito Funcional: Aplicativo de Busca de Carros**

**Descrição:**

O aplicativo de busca de carros permite que os usuários pesquisem e visualizem anúncios de carros usados com informações sobre Modelo, Marca, Ano e Valor.

#### **Critérios de Aceitação:**

O aplicativo deve permitir que os usuários pesquisem anúncios de carros com base nas seguintes informações:

Modelo: O modelo específico do carro (por exemplo, "Corolla", "Fiesta", "Cruze").

Marca: O nome da marca do veículo (por exemplo, "Toyota", "Ford", "Chevrolet").

Ano: O ano de fabricação do veículo.

Valor: O preço de venda do carro.

O aplicativo deve exibir os resultados da pesquisa de forma clara e ordenada, permitindo que os usuários visualizem as informações sobre Modelo, Marca, Ano e Valor dos carros disponíveis.

Os usuários não podem cadastrar novos anúncios de carros no aplicativo. A função principal do aplicativo é a busca e listagem de carros existentes.

Os usuários podem refinar sua pesquisa usando filtros, como faixa de preço, marca específica, modelo e ano, para encontrar carros que atendam às suas preferências.

O aplicativo deve ser compatível com dispositivos móveis e oferecer uma interface de usuário amigável para facilitar a pesquisa e a visualização dos anúncios.

Este requisito especifica os critérios para o aplicativo de busca de carros, focando na funcionalidade de pesquisa e listagem de carros usados com base em Modelo, Marca, Ano e Valor, enquanto esclarece que o aplicativo não permite o cadastro de novos anúncios. Certifique-se de desenvolver o aplicativo de acordo com esses critérios para atender às necessidades dos usuários que desejam encontrar informações sobre carros usados.