



# Flutter

Wesley Dias Maciel

2021/02



Centro Universitário UNA  
Flutter  
Wesley Dias Maciel  
2021/02

# Flutter

# Prática 20

## BottomNavigationBar

Documentação: <https://api.flutter.dev/flutter/material/BottomNavigationBar-class.html>

**Objetivo:** exemplificar o uso do widget BottomNavigationBar.

O widget BottomNavigationBar é exibido na parte inferior de um aplicativo. Ele permite selecionar um pequeno número de telas, visualizações (normalmente entre três e cinco).

O widget BottomNavigationBar apresenta itens na forma de rótulos de texto, ícones ou ambos. Geralmente, é usado em conjunto com um widget Scaffold, em que é passado como argumento do atributo Scaffold.bottomNavigationBar.

Nesta prática, você vai:

- a) Criar um BottomNavigationBar.

- 1) Crie um novo projeto Flutter, usando:
  - a. Visual Studio Code, ou;
  - b. <https://dartpad.dev/>, ou;
  - c. <https://flutlab.io/ide>, ou;
  - d. <https://flutterstudio.app/>, ou;
  - e. <https://codemagic.io/>.

## Transição de Telas

- 2) O exemplo abaixo cria um aplicativo que permite realizar uma transição de telas através de um widget BottomNavigationBar. O exemplo mostra como um widget BottomNavigationBar é usado em um widget Scaffold. O widget BottomNavigationBar tem três widgets BottomNavigationBarItem. O inteiro “índice” é usado para indicar o BottomNavigationBarItem selecionado pelo usuário. Ele é iniciado com o valor 0, indicando a primeira tela do aplicativo. Além disso, o item selecionado é apresentado na cor âmbar. A função “selecionar” muda o índice do item selecionado e exibe uma

mensagem no corpo do widget Scaffold. Essa função é chamada quando o usuário pressiona os widgets BottomNavigationBarItem.

```
import 'package:flutter/material.dart';

void main() => runApp(Aplicativo());

class Aplicativo extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Home(),
      debugShowCheckedModeBanner: false,
    );
  }
}

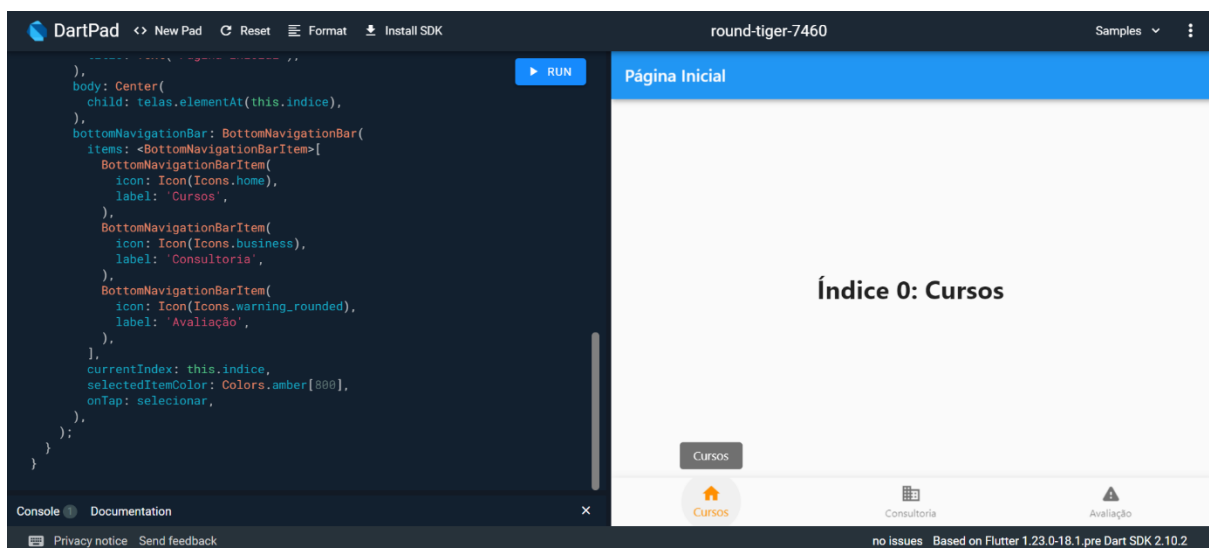
class Home extends StatefulWidget {
  @override
  HomeState createState() => HomeState();
}

class HomeState extends State<Home> {
  int indice = 0;
  static const TextStyle estiloParaTexto =
    TextStyle(fontSize: 30, fontWeight: FontWeight.bold);
  static const List<Widget> telas = <Widget>[
    Text(
      'Índice 0: Cursos',
      style: estiloParaTexto,
    ),
    Text(
      'Índice 1: Consultoria',
      style: estiloParaTexto,
    ),
    Text(
      'Índice 3: Avaliação',
      style: estiloParaTexto,
    ),
  ];

  void selecionar(int indice) {
    setState(() => this.indice = indice);
  }

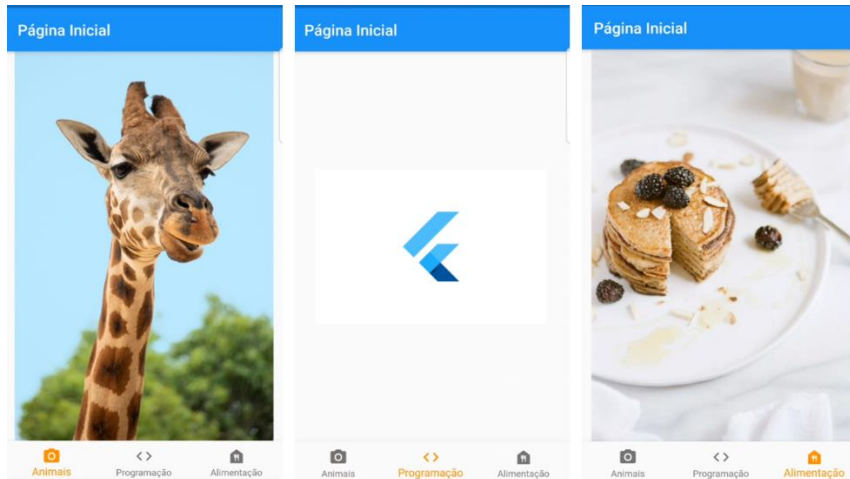
  @override
```

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: Text('Página Inicial'),  
    ),  
    body: Center(  
      child: telas.elementAt(this.indice),  
    ),  
    bottomNavigationBar: BottomNavigationBar(  
      items: <BottomNavigationBarItem>[  
        BottomNavigationBarItem(  
          icon: Icon(Icons.home),  
          label: 'Cursos',  
        ),  
        BottomNavigationBarItem(  
          icon: Icon(Icons.business),  
          label: 'Consultoria',  
        ),  
        BottomNavigationBarItem(  
          icon: Icon(Icons.warning_rounded),  
          label: 'Avaliação',  
        ),  
      ],  
      currentIndex: this.indice,  
      selectedItemColor: Colors.amber[800],  
      onTap: selecionar,  
    ),  
  );  
}
```



## Exercício

1) Altere o algoritmo apresentado nesta prática, para que ele apresente as telas abaixo.



Dica:

a)

```
static const List<Widget> telas = <Widget>[  
  Image(  
    image: NetworkImage(  
      'https://images.unsplash.com/photo-1547721064-da6cfb341d50'),  
    ),  
  Image(  
    image: NetworkImage(  
      'https://flutter.io/images/catalog-widget-placeholder.png'),  
    ),  
  Image(  
    image: NetworkImage(  
      'https://images.pexels.com/photos/213780/pexels-photo-  
213780.jpeg?auto=compress&cs=tinysrgb&dpr=1&w=500'),  
    ),  
];
```