



Centro Universitário UNA

Usabilidade, Desenvolvimento Web,
Mobile e Jogos

Graduação – TI e Engenharias

Práticas de Laboratório

Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria
Barros, Wesley Dias Maciel

2020/02



Centro Universitário UNA
Graduação – TI e Engenharias
Usabilidade, Desenvolvimento Web, Mobile e Jogos
Prática de Laboratório
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto
de Faria Barros, Wesley Dias Maciel
2020/02

Flutter



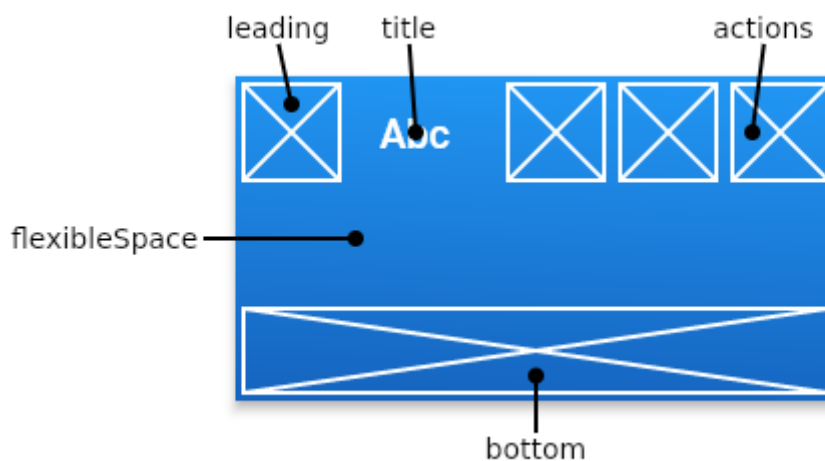
Prática 19

AppBar

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

Objetivo: exemplificar o uso do widget AppBar.

O widget AppBar é uma barra de ferramentas que contém outros widgets. Um AppBar geralmente expõe uma ou mais ações através de ícones. Além disso, são normalmente usados na propriedade Scaffold.appBar, que coloca o AppBar como um widget de altura fixa na parte superior da tela. Um AppBar exibe os widgets: leading, title, and actions. O diagrama a seguir mostra onde esses widgets aparecem:



Nesta prática, você vai:

- a) Criar um AppBar.
- b) Inserir widgets a esse AppBar.
- c) Associar funcionalidades aos widgets do AppBar.



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

AppBar

- 2) O exemplo abaixo cria duas rotas. A primeira rota possui um widget AppBar. O AppBar possui:
 - a. leading: que recebe um widget IconButton.
 - b. title: que recebe um widget Text.

Ao clicar no ícone da primeira rota, o usuário é direcionado para a segunda rota.

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

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

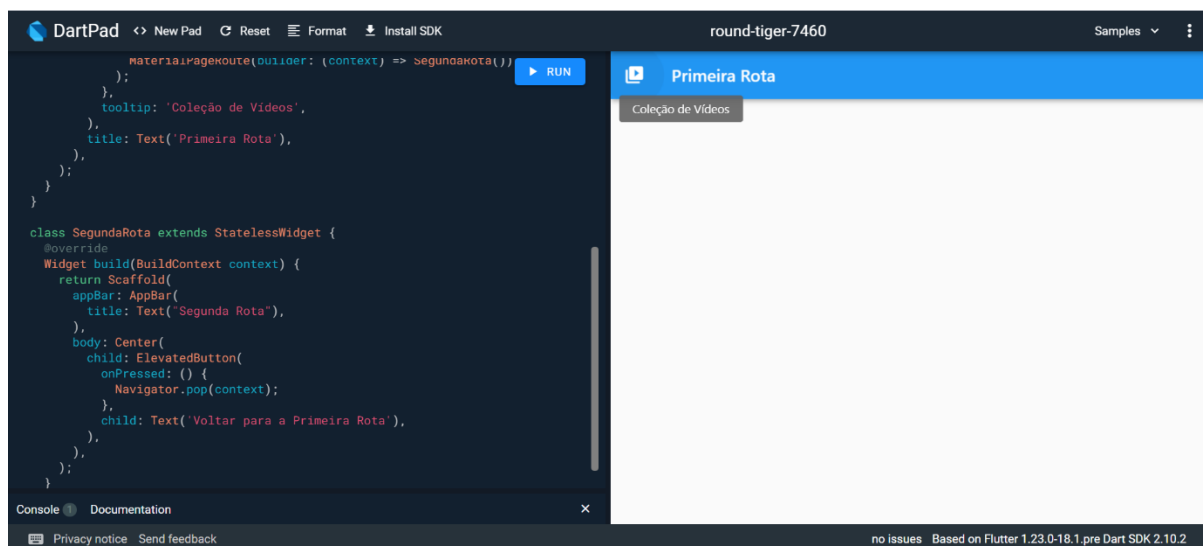
class MeuAplicativo extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: PrimeiraRota(),
      debugShowCheckedModeBanner: false,
    );
  }
}

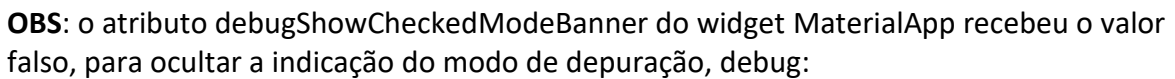
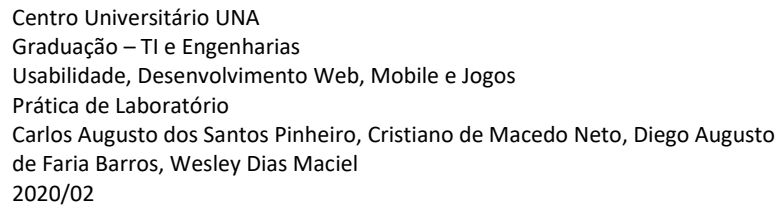
class PrimeiraRota extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        leading: IconButton(
          icon: Icon(Icons.video_collection),
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => SegundaRota()),
            );
          }
        ),
      ),
    );
  }
}
```



Centro Universitário UNA
Graduação – TI e Engenharias
Usabilidade, Desenvolvimento Web, Mobile e Jogos
Prática de Laboratório
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria Barros, Wesley Dias Maciel
2020/02

```
    },  
    tooltip: 'Coleção de Vídeos',  
  ),  
  title: Text('Primeira Rota'),  
),  
);  
}  
}  
  
class SegundaRota extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Rota"),  
      ),  
      body: Center(  
        child: ElevatedButton(  
          onPressed: () {  
            Navigator.pop(context);  
          },  
          child: Text('Voltar para a Primeira Rota'),  
        ),  
      ),  
    );  
  }  
}
```





The screenshot shows the DartPad web-based IDE. The top bar includes navigation links like 'New Pad', 'Reset', 'Format', and 'Install SDK'. The main editor area contains Dart code for a Flutter application. The code defines a `PrimeiraRota` widget that uses `Scaffold` and `AppBar`. A red circle highlights the 'run' button in the top right corner of the editor, and a red arrow points to it.

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

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

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

class PrimeiraRota extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        leading: IconButton(
          icon: Icon(Icons.video_collection),
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => SegundaRota()),
            );
          },
        ),
      ),
    );
  }
}
```



actions

- 3) O widget actions permite inserir uma lista de widgets no AppBar. No exemplo abaixo, o widget actions possui uma lista de ícones. Ao clicar num dos dois primeiros ícones, o algoritmo apresenta um cartão com o meio de transporte relacionado ao ícone. Ao clicar no terceiro ícone, o algoritmo apresenta um menu. Ao clicar numa opção do menu, o algoritmo também apresenta um cartão com o meio de transporte correspondente.

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

void main() {
  runApp(MeuAplicativo());
}

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

class PrimeiraRota extends StatefulWidget {
  @override
  PrimeiraRotaState createState() => PrimeiraRotaState();
}

class PrimeiraRotaState extends State<PrimeiraRota> {
  Transporte transporte = transportes[0];

  void selecionar(Transporte transporteEscolhido) {
    setState(
      () {
        this.transporte = transporteEscolhido;
      },
    );
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
```

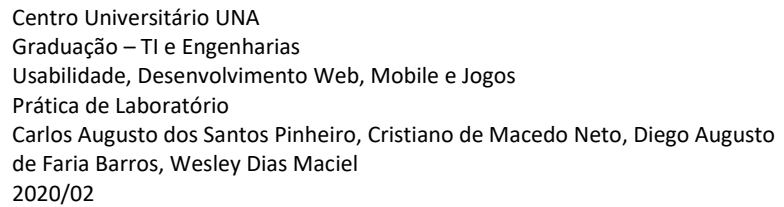


```
appBar: AppBar(  
  leading: IconButton(  
    icon: Icon(Icons.video_collection),  
    onPressed: () {  
      Navigator.push(  
        context,  
        MaterialPageRoute(builder: (context) => SegundaRota()),  
      );  
    },  
    tooltip: 'Coleção de Vídeos',  
  ),  
  title: Text('Primeira Rota'),  
  actions: <Widget>[  
    IconButton(  
      icon: Icon(transportes[0].icone),  
      onPressed: () {  
        selecionar(transportes[0]);  
      },  
    ),  
    IconButton(  
      icon: Icon(transportes[1].icone),  
      onPressed: () {  
        selecionar(transportes[1]);  
      },  
    ),  
    PopupMenuButton<Transporte>(  
      onSelected: selecionar,  
      itemBuilder: (BuildContext context) {  
        return transportes.skip(2).map(  
          (Transporte transporte) {  
            return PopupMenuItem<Transporte>(  
              value: transporte,  
              child: Text(transporte.titulo),  
            );  
          },  
        ).toList();  
      },  
    ),  
  ],  
,  
  body: Padding(  
    padding: EdgeInsets.all(16),  
    child: Cartao(transporte: this.transporte),  
  ),  
,
```




```
);  
}  
}  
  
class SegundaRota extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Rota"),  
      ),  
      body: Center(  
        child: ElevatedButton(  
          onPressed: () {  
            Navigator.pop(context);  
          },  
          child: Text('Voltar para a Primeira Rota'),  
        ),  
      ),  
    );  
  }  
}  
  
class Transporte {  
  final String titulo;  
  final IconData icone;  
  const Transporte({this.titulo, this.icone});  
}  
  
const List<Transporte> transportes = <Transporte>[  
  Transporte(titulo: 'Carro', icone: Icons.directions_car),  
  Transporte(titulo: 'Bicicleta', icone: Icons.directions_bike),  
  Transporte(titulo: 'Barco', icone: Icons.directions_boat),  
  Transporte(titulo: 'Ônibus', icone: Icons.directions_bus),  
  Transporte(titulo: 'Trem', icone: Icons.directions_railway),  
];  
  
class Cartao extends StatelessWidget {  
  final Transporte transporte;  
  Cartao({this.transporte});  
  
  @override  
  Widget build(BuildContext context) {  
    return Card(  

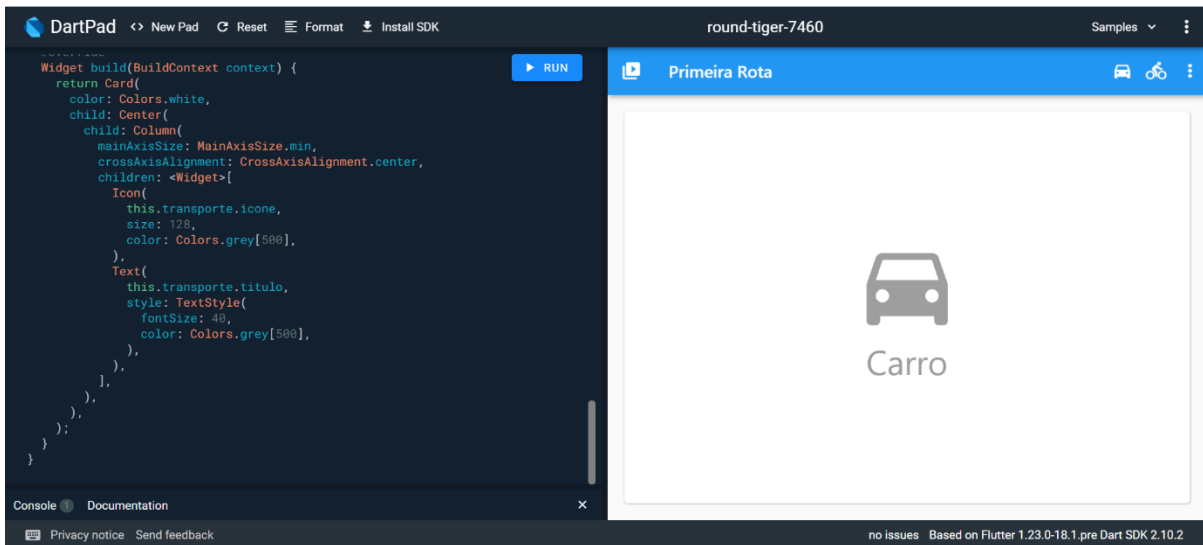
```



```

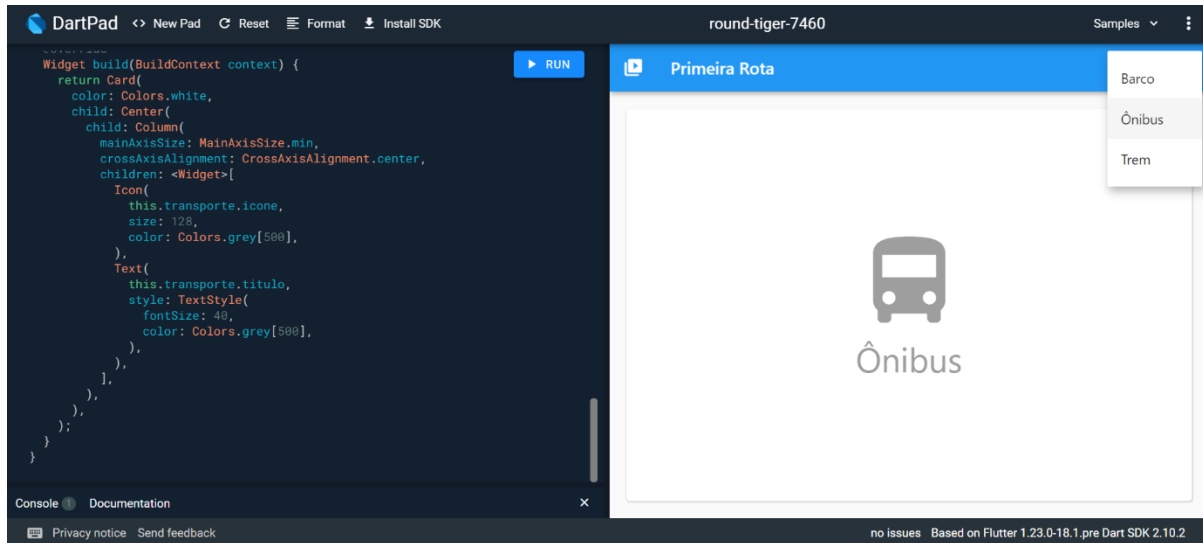
color: Colors.white,
child: Center(
  child: Column(
    mainAxisAlignment: MainAxisAlignment.min,
    crossAxisAlignment: CrossAxisAlignment.center,
    children: <Widget>[
      Icon(
        this.transporte.icone,
        size: 128,
        color: Colors.grey[500],
      ),
      Text(
        this.transporte.titulo,
        style: TextStyle(
          fontSize: 40,
          color: Colors.grey[500],
        ),
      ),
    ],
  ),
),
);
}
}

```





Centro Universitário UNA
Graduação – TI e Engenharias
Usabilidade, Desenvolvimento Web, Mobile e Jogos
Prática de Laboratório
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria Barros, Wesley Dias Maciel
2020/02



Exercício

- 1) Altere o algoritmo apresentado nesta prática, para que ao clicar num item do menu, o usuário seja direcionado para uma nova rota. A nova rota deve apresentar o cartão correspondente à opção selecionada no menu. A nova rota deve ser gerada por uma rota genérica.

Exemplo:

