



Aplicações
Mobile
Aula 06

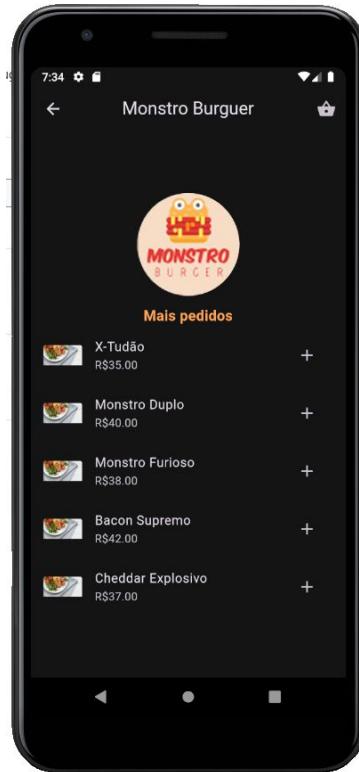
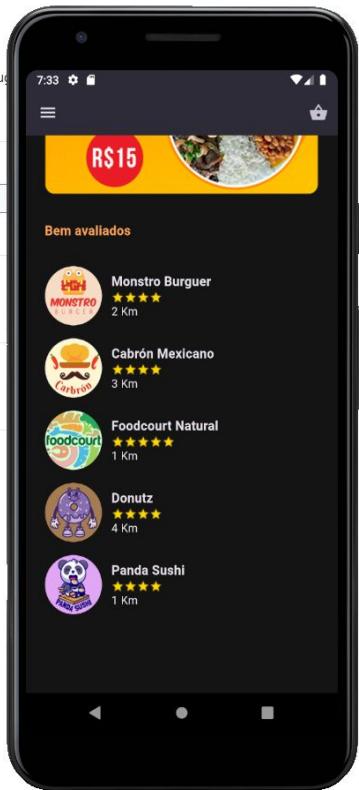
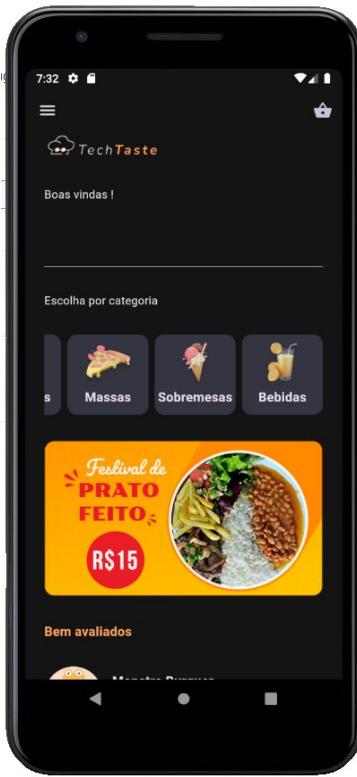
Prof. Me Daniel Vieira



Agenda

- 1 - Aplicativo Delivery
- 2 - Organização do projeto
- 3 - Provider

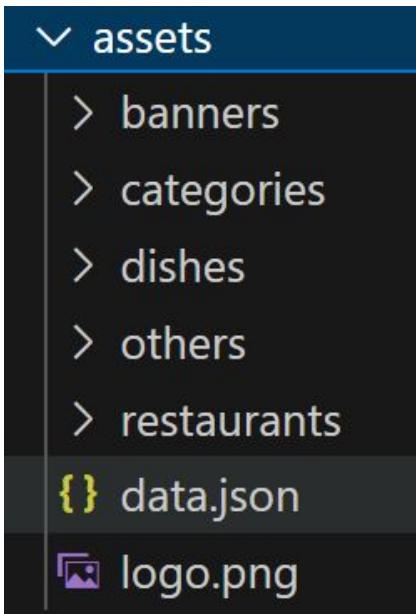
Aplicativo Delivery



Organização do projeto

1- Criar a pasta assets

1.1 - Criar as subpastas banners, categories, dishes, others, restaurants, colocar o arquivo data.json e o arquivo logo.png na pasta assets



Organização do projeto

2 - Entrar no arquivo pubspec.yaml e adicionar a pasta assets com os arquivos e pastas no projeto

```
61 # To add assets to your application, add an assets section, like this:  
62 assets:  
63     - assets/  
64     - assets/banners/  
65     - assets/categories/  
66     - assets/dishes/  
67     - assets/others/  
68     - assets/restaurants/
```

3 - No terminal do VS CODE digitar flutter pub get e todas dependências e arquivos serão adicionados ao projeto

Organização do projeto

4 - Entrar no arquivo pubspec.yaml e adicionar as dependências provider e badges ao projeto

```
34  # The following adds the Cupertino Icons font to your application.
35  # Use with the CupertinoIcons class for iOS style icons.
36  cupertino_icons: ^1.0.8
37  provider: ^6.1.4
38  badges: ^3.1.2
39  dev_dependencies:
40    flutter_test:
41      sdk: flutter
```

5 - No terminal do VS CODE digitar flutter pub get e todas dependências serão adicionados ao projeto

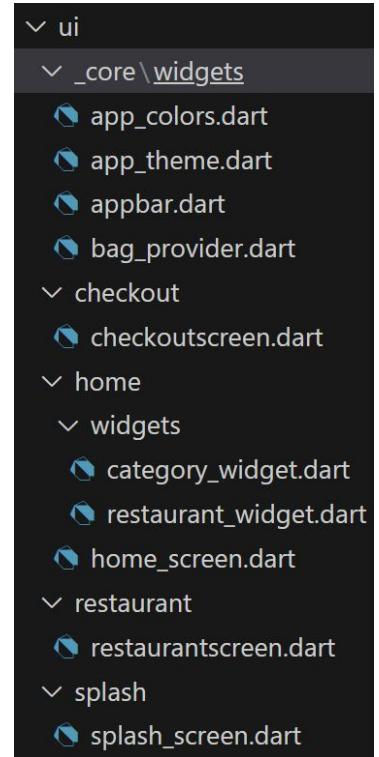
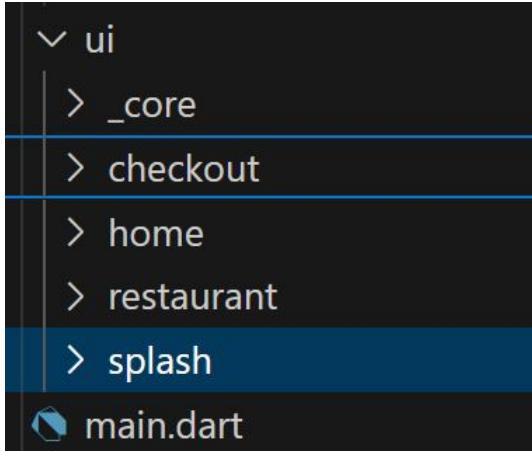
Organização do projeto

```
✓ lib
  ✓ data
    categories_data.dart
    restaurant_data.dart
  ✓ model
    dish.dart
    restaurant.dart
  ✓ ui
    _core\widgets
      app_colors.dart
      app_theme.dart
      appBar.dart
      bag_provider.dart
    checkout
      checkoutscreen.dart
    home
      ✓ widgets
        category_widget.dart
        restaurant_widget.dart
        home_screen.dart
    restaurant
      restaurantscreen.dart
    splash
      splash_screen.dart
  main.dart
```

6 - Dentro da pasta lib criar as pastas data, model, ui

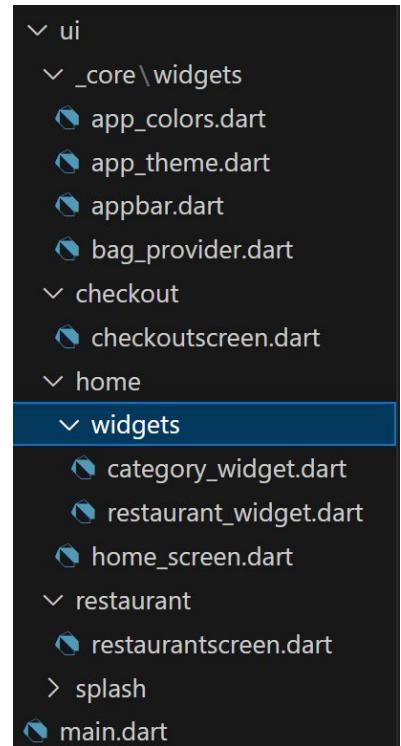
Organização do projeto

7 - Dentro da pasta ui criar a pasta _core, widgets, home e dentro da pasta home criar a pasta widgets, criar a pasta restaurant, splash



Organização do projeto

8 - Dentro da pasta widgets criar a pasta checkout, home, widgets, restaurant, splash



Código

9 - Criar o arquivo Dish.dart em lib - model

```
1 class Dish {  
2   final String id;  
3   final String name;  
4   final String description;  
5   final int price;  
6   final String imagePath;  
7  
8   Dish({  
9     required this.id,  
10    required this.name,  
11    required this.description,  
12    required this.price,  
13    required this.imagePath  
14  });  
15  Map<String, dynamic> toMap() {  
16    return {  
17      'id': id,  
18      'name': name,  
19      'description': description,  
20      'price': price,  
21      'imagePath': imagePath,  
22    };  
23  }  
24  
25  factory Dish.fromMap(Map<String, dynamic> map) {  
26    return Dish(  
27      id: map['id'],  
28      name: map['name'],  
29      description: map['description'],  
30      price: map['price'],  
31      imagePath: map['imagePath'],  
32    );  
33  }  
34 }  
35  
36  @override  
37  String toString(){  
38    return 'Dish(id: $id, name: $name, description: $description, price: $price, imagePath: $imagePath)';  
39  }  
40 }  
41 }
```

Código

10 - Criar o arquivo restaurant.dart em lib - model

```
1 import 'package:appaula06prep/model/dish.dart';
2 class Restaurant {
3   | | String id;
4   String imagePath;
5   String name;
6   String description;
7   double stars;
8   int distance;
9   List<String> categories;
10  List<Dish> dishes;
11
12 Restaurant({
13   required this.id,
14   required this.imagePath,
15   required this.name,
16   required this.description,
17   required this.stars,
18   required this.distance,
19   required this.categories,
20   required this.dishes,
21 });
22 // cria mapa para converter elementos json para o app consumir
23 Map<String, dynamic> toMap() {
24   return {
25     'id': id,
26     'imagePath': imagePath,
27     'name': name,
28     'description': description,
29     'stars': stars,
30     'distance': distance,
31     'categories': categories,
32     // função arrow função de uma linha só que transforma em uma lista
33     'dishes': dishes.map((dish) => dish.toMap()).toList(),
34   };
35 }
36
```

```
37 // queremos receber o mapa para cria o restaurante
38 factory Restaurant.fromMap(Map<String, dynamic> map) {
39   return Restaurant(
40     id: map['id'],
41     imagePath: map['imagePath'],
42     name: map['name'],
43     description: map['description'],
44     stars: map['stars'],
45     distance: map['distance'],
46     categories: List<String>.from(map['categories']),
47     dishes: List<Dish>.from(map['dishes'].map((dish) => Dish.fromMap(dish))),
48   ); // Restaurant
49 }
50
51 @override
52 String toString() {
53   return '''Restaurant(
54     id: $id,
55     imagePath: $imagePath,
56     name: $name,
57     description: $description,
58     stars: $stars,
59     distance: $distance,
60     categories: $categories,
61   )'''';
62 }
63 }
64
```

Código

11 - Criar o arquivo categories_data.dart em lib - data

```
lib > data > categories_data.dart > ...
1 abstract class CategoriesData {
2     // Lista do tipo string
3     static List<String> listCategories = [
4         "Petiscos",
5         "Principais",
6         "Massas",
7         "Sobremesas",
8         "Bebidas",
9     ];
10 }
11
```

Código

12 - Criar o arquivo restaurant_data.dart em lib - data

```
1 import 'dart:convert';
2 import 'package:appaula06prep/model/restaurant.dart';
3 import 'package:flutter/material.dart';
4 import 'package:flutter/services.dart';
5
6 class RestaurantData extends ChangeNotifier{
7   List<Restaurant> _listRestaurant =[];
8   List<Restaurant> get listRestaurant=> _listRestaurant;
9
10 Future<List<Restaurant>> getRestaurant() async {
11   if (_listRestaurant.isNotEmpty) {
12     return _listRestaurant; // Evita recarregar se já tiver carregado
13   }
14
15   try {
16     final String jsonString = await rootBundle.loadString('assets/data.json');
17     final Map<String, dynamic> data = json.decode(jsonString);
18     final List<dynamic> restaurantData = data['restaurants'];
19
20     _listRestaurant.addAll(
21       restaurantData.map((e) => Restaurant.fromMap(e)).toList(),
22     );
23
24     notifyListeners(); // Notifica caso esteja usando com Provider
25   } catch (e) {
26     debugPrint('Erro ao carregar restaurantes: $e');
27   }
28
29   return _listRestaurant;
30 }
31
32
33
34
35 }
```

Código

13 - Criar o arquivo app_colors.dart em lib - ui _core - widgets

```
lib > ui > _core > widgets > app_colors.dart > AppColors
1 import 'package:flutter/material.dart';
2
3 abstract class AppColors {
4     static const Color backgroundColor = Color(0xFF202123);
5     static const Color mainColor = Color(0xFFffa559);
6     static const Color lightBackgroundColor = Color(0xFF343541);
7     static const Color fundoCards = Color(0xFF343541);
8 }
```

Código

14 - Criar o arquivo app_theme.dart em lib - ui _core - widgets

```
lib > ui > _core > widgets > app_theme.dart > ...
1 import 'package:appaula06prep/ui/_core/widgets/app_colors.dart';
2 import 'package:flutter/material.dart';
3
4 abstract class AppTheme {
5     static ThemeData appTheme = ThemeData.dark().copyWith(
6         elevatedButtonTheme: ElevatedButtonThemeData(
7             style: ButtonStyle(
8                 foregroundColor: WidgetStatePropertyAll(Colors.black),
9                 backgroundColor: WidgetStateColor.resolveWith((states) {
10                     if (states.contains(WidgetState.disabled)) {
11                         return Colors.grey;
12                     } else if (states.contains(WidgetState.pressed)) {
13                         return Color.fromARGB(171, 255, 164, 89);
14                     }
15                     return AppColors.mainColor;
16                 }), // WidgetStateColor.resolveWith
17             ), // ButtonStyle
18         ), // ElevatedButtonThemeData
19     );
20 }
21 }
```

Código

15 - Criar o arquivo `appbar.dart` em `lib - ui _core - widgets`

```
lib > ui > _core > widgets >  appbar.dart >  getAppBar
1 import 'package:appaula06prep/ui/_core/widgets/bag_provider.dart';
2 import 'package:appaula06prep/ui/checkout/checkoutscreen.dart';
3 import 'package:flutter/material.dart';
4
5 import 'package:provider/provider.dart';
6 import 'package:badges/badges.dart' as badges;
7
8 AppBar getAppBar({required BuildContext context, String? title}) {
9   BagProvider bagProvider = Provider.of<BagProvider>(context);
10  return AppBar(
11    title: title != null ? Text(title) : null,
12    centerTitle: true,
13    actions: [
14      badges.Badge(
15        showBadge: bagProvider.dishesOnBag.isNotEmpty,
16        position: badges.BadgePosition.bottomStart(start: 0, bottom: 0),
17        badgeContent: Text(
18          bagProvider.dishesOnBag.length.toString(),
19          style: TextStyle(fontSize: 10, color: Colors.white),
20        ), // Text
21        child: IconButton(
22          onPressed: () {
23            Navigator.push(
24              context,
25              MaterialPageRoute(builder: (context) => Checkoutscreen()),
26            );
27          },
28          icon: Icon(Icons.shopping_basket),
29        ), // IconButton
30        ), // badges.Badge
31    ],
32  ); // AppBar
33 }
34 }
```

Código

16 - Criar o arquivo bag_provider.dart em lib - ui _core - widgets

```
lib > ui > _core > widgets > bag_provider.dart ...
1 import 'package:appaula06prep/model/dish.dart';
2 import 'package:flutter/material.dart';
3
4
5 class BagProvider extends ChangeNotifier {
6   List<Dish> dishesOnBag = [];
7
8   addAllDishes(List<Dish> dishes) {
9     dishesOnBag.addAll(dishes);
10    notifyListeners();
11  }
12
13 removeDish(Dish dish) {
14   dishesOnBag.remove(dish);
15   notifyListeners();
16 }
17
18 @override
19 String toString() {
20   return 'BagProvider(dishesOnBag: $dishesOnBag)';
21 }
22
23 clearBag() {
24   dishesOnBag.clear();
25   notifyListeners();
26 }
27
```

```
28   Map<Dish, int> getMapByAmount() {
29     Map<Dish, int> mapResult = {};
30     for (Dish dish in dishesOnBag) {
31       if (mapResult[dish] == null) {
32         mapResult[dish] = 1;
33       } else {
34         mapResult[dish] = mapResult[dish]! + 1;
35       }
36     }
37     return mapResult;
38   }
39 }
40 |
```

Código

17 - Criar o arquivo checkout.dart em lib - ui - checkout

```
lib > ui > checkout > checkoutscreen.dart > Checkoutscreen > build
1 import 'package:appaula06prep/model/dish.dart';
2 import 'package:appaula06prep/ui/_core/widgets/app_colors.dart';
3 import 'package:appaula06prep/ui/_core/widgets/bag_provider.dart';
4 import 'package:flutter/material.dart';
5
6
7 import 'package:provider/provider.dart';
8
9 class Checkoutscreen extends StatelessWidget {
10   const Checkoutscreen({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     BagProvider bagProvider = Provider.of<BagProvider>(context);
15
16     // calcula o valor total
17     double total = 0;
18     bagProvider.getMapByAmount().forEach((dish, amount) {
19       total += dish.price * amount;
20     });
21
22     return Scaffold(
23       appBar: AppBar(
24         title: const Text('Sacola'),
25         actions: [
26           TextButton(
27             onPressed: () {
28               bagProvider.clearBag();
29             },
30             child: const Text('Limpar'),
31           ), // TextButton
32         ],
33       ), // AppBar
34   }
```

```
34   body: Padding(
35     padding: const EdgeInsets.symmetric(horizontal: 16.0),
36     child: SingleChildScrollView(
37       child: Column(
38         crossAxisAlignment: CrossAxisAlignment.stretch,
39         children: [
40           const Text('Pedidos', textAlign: TextAlign.center),
41           Column(
42             children: List.generate(
43               bagProvider.getMapByAmount().keys.length,
44               (index) {
45                 Dish dish =
46                   bagProvider.getMapByAmount().keys.toList()[index];
47                 return ListTile(
48                   leading: Image.asset(
49                     'assets/dishes/default.png',
50                     width: 48,
51                     height: 48,
52                   ), // Image.asset
53                   title: Text(dish.name),
54                   subtitle: Text('R\$ ${dish.price.toStringAsFixed(2)}'),
55                   trailing: Row(
56                     mainAxisSize: MainAxisSize.min,
57                     children: [
58                       IconButton(
59                         onPressed: () {
60                           bagProvider.removeDish(dish);
61                         },
62                         icon: const Icon(Icons.remove),
63                       ), // IconButton
64                     Text(bagProvider.getMapByAmount()[dish].toString()),
65                   ],
66                 );
67               }
68             ),
69           ],
70         ),
71       ),
72     ),
73   ),
74 }
```

Código

18 - Criar o arquivo checkout.dart em lib - ui - checkout

```
65           IconButton(
66             onPressed: () {
67               bagProvider.addAllDishes([dish]);
68             },
69             icon: const Icon(Icons.add),
70           ), // IconButton
71         ],
72       ), // Row
73     ); // ListTile
74   },
75   ), // List.generate
76 ), // Column
77 const SizedBox(height: 16),
78 const Text(
79   'Pagamento',
80   style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
81 ), // Text
82 Stack(
83   alignment: Alignment.centerLeft,
84   children: [
85     Container(
86       width: double.infinity,
87       height: 100,
88       color: AppColors.fundoCards,
89     ), // Container
90     Container(
91       width: 100,
92       height: 80,
93       color: AppColors.fundoCards,
94       child: Padding(
95         padding: const EdgeInsets.all(12.0),
96         child: Image.asset(
```

```
96           child: Image.asset(
97             'assets/others/visa.png',
98             fit: BoxFit.fill,
99           ), // Image.asset
100         ), // Padding
101       ), // Container
102     Row(
103       mainAxisSize: MainAxisSize.spaceAround,
104       children: [
105         const Positioned(
106           left: 100,
107           child: Text(
108             'Visa Classic',
109             style: TextStyle(fontSize: 18, color: Colors.white),
110           ), // Text
111         ), // Positioned
112       ],
113     ), // Row
114   ],
115 ), // Stack
116 const SizedBox(height: 24),
117 // total da compra
118 Text(
119   'Total: R\$ ${total.toStringAsFixed(2)}',
120   style: const TextStyle(
121     fontSize: 18,
122     fontWeight: FontWeight.bold,
123   ), // TextStyle
124   textAlign: TextAlign.end,
125 ), // Text
126 const SizedBox(height: 12).
```

Código

19 - Criar o arquivo checkout.dart em lib - ui - checkout

```
127 // botão pedir
128 ElevatedButton(
129   style: ElevatedButton.styleFrom(
130
131     backgroundColor: Colors.orange,
132     padding: const EdgeInsets.symmetric(vertical: 16),
133
134   ),
135   onPressed: () {
136     // aqui você define a ação do pedido
137     ScaffoldMessenger.of(context).showSnackBar(
138       const SnackBar(content: Text("Pedido realizado!")),
139     );
140   },
141   child: const Text(
142     'Pedir',
143     style: TextStyle(fontSize: 18, color: Colors.white),
144   ), // Text
145   ), // ElevatedButton
146 ],
147 ), // Column
148 ), // SingleChildScrollView
149 ), // Padding
150 ); // Scaffold
151 }
152 }
153 }
```

Código

20 - Criar o arquivo checkout.dart em lib - ui - checkout

```
127 // botão pedir
128 ElevatedButton(
129   style: ElevatedButton.styleFrom(
130
131     backgroundColor: Colors.orange,
132     padding: const EdgeInsets.symmetric(vertical: 16),
133
134   ),
135   onPressed: () {
136     // aqui você define a ação do pedido
137     ScaffoldMessenger.of(context).showSnackBar(
138       const SnackBar(content: Text("Pedido realizado!")),
139     );
140   },
141   child: const Text(
142     'Pedir',
143     style: TextStyle(fontSize: 18, color: Colors.white),
144   ), // Text
145   ), // ElevatedButton
146 ],
147 ), // Column
148 ), // SingleChildScrollView
149 ), // Padding
150 ); // Scaffold
151 }
152 }
153 }
```

Código

21 - Criar o arquivo category_widget.dart em lib - ui - home

```
lib > ui > home > widgets > category_widget.dart > CategoryWidget > build
1 import 'package:appaula06prep/ui/_core/widgets/app_colors.dart';
2 import 'package:flutter/material.dart';
3
4 class CategoryWidget extends StatelessWidget {
5   final String category;
6   const CategoryWidget({super.key, required this.category});
7
8   @override
9   Widget build(BuildContext context) {
10    return Container(
11      width: 100,
12      height: 100,
13      decoration: BoxDecoration(
14        color: AppColors.lightBackgroundColor,
15        borderRadius: BorderRadius.circular(12),
16      ), // BoxDecoration
17      child: Column(
18        spacing: 8,
19        mainAxisSize: MainAxisSize.min,
20        mainAxisAlignment: MainAxisAlignment.center,
21        crossAxisAlignment: CrossAxisAlignment.center,
22        children: [
23          Image.asset(
24            'assets/categories/${category.toLowerCase()}.png',
25            height: 48,
26          ), // Image.asset
27          Text(
28            category,
29            style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
30          ), // Text
31        ],
32      ), // Column
33    ); // Container
34  }
35}
36
```

Código

22 - Criar o arquivo restaurant_widget.dart em lib - ui - home

```
lib > ui > home > widgets > restaurant_widget.dart ...
1 import 'package:appaula06prep/model/restaurant.dart';
2 import 'package:appaula06prep/ui/restaurant/restaurantscreen.dart';
3 import 'package:flutter/material.dart';
4
5 class RestaurantWidget extends StatelessWidget {
6   final Restaurant restaurant;
7   const RestaurantWidget({super.key, required this.restaurant});
8
9   @override
10  Widget build(BuildContext context) {
11    return Material(
12      child: InkWell(
13        onTap: () {
14          Navigator.push(
15            context,
16            MaterialPageRoute(
17              builder: (context) {
18                return Restaurantscreen(restaurant: restaurant);
19              },
20            ), // MaterialPageRoute
21          );
22        },
23      ),
24      child: Row(
25        spacing: 12,
26        children: [
27          Image.asset('assets/${restaurant.imagePath}', width: 72),
28          Column(
29            crossAxisAlignment: CrossAxisAlignment.start,
30            mainAxisAlignment: MainAxisAlignment.start,
31            children: [
32              Text(
33                restaurant.name,
34                style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
35              ), // Text
36              Row(
37                children: List.generate(restaurant.stars.toInt(), (index) {
```

```
35               Row(
36                 children: List.generate(restaurant.stars.toInt(), (index) {
37                   return Image.asset('assets/others/star.png', width: 16);
38                 }), // List.generate
39               ), // Row
40               Text('${restaurant.distance} Km'),
41             ],
42           ), // Column
43         ],
44       ), // Row
45     ), // InkWell
46   ); // Material
47 }
48 }
49 }
```

Código

23 - Criar o arquivo home_screen.dart em lib - ui - home

```
lib > ui > home > home_screen.dart > HomeScreen > build
1 import 'package:appaula06prep/data/categories_data.dart';
2 import 'package:appaula06prep/data/restaurant_data.dart';
3 import 'package:appaula06prep/model/restaurant.dart';
4 import 'package:appaula06prep/ui/_core/widgets/app_colors.dart';
5 import 'package:appaula06prep/ui/_core/widgets/appbar.dart';
6 import 'package:appaula06prep/ui/home/widgets/category_widget.dart';
7 import 'package:appaula06prep/ui/home/widgets/restaurant_widget.dart';
8 import 'package:flutter/material.dart';
9
10 import 'package:provider/provider.dart';
11
12 // palvra chave aula 2Widgets
13 class HomeScreen extends StatelessWidget {
14   const HomeScreen({super.key});
15
16   @override
17   Widget build(BuildContext context) {
18     RestaurantData restaurantData = Provider.of<RestaurantData>(context);
19     return Scaffold(
20       drawer: Drawer(),
21       appBar: getAppBar(context: context),
22       body: Padding(
23         padding: EdgeInsets.symmetric(horizontal: 24.0),
24         child: Center(
25           child: SingleChildScrollView(
26             child: Column(
27               spacing: 32,
28               crossAxisAlignment: CrossAxisAlignment.start,
29               children: [
30                 Image.asset('assets/logo.png', width: 147),
31                 Text('Boas vindas !'),
32                 TextFormField(),
33                 Text('Escolha por categoria'),
34                 SingleChildScrollView(
35                   scrollDirection: Axis.horizontal,
36                   child: Row(
```

```
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
```

```
// Lista que vai ser gerada das categories
spacing: 8,
children: List.generate(
  CategoriesData.listCategories.length,
  (index) {
    return CategoryWidget(
      category: CategoriesData.listCategories[index],
    ); // CategoryWidget
  },
), // List.generate
), // Row
), // SingleChildScrollView
Image.asset('assets/banners/banner_promo.png'),
Text(
  'Bem avaliados',
  style: TextStyle(
    color: AppColors.mainColor,
    fontSize: 16,
    fontWeight: FontWeight.bold,
  ), // TextStyle
), // Text
Column(
  spacing: 16,
  children: List.generate(
    restaurantData.listRestaurant.length,
    (index) [
      Restaurant restaurant = restaurantData.listRestaurant[index];
      return RestaurantWidget(restaurant: restaurant);
    ],
  ),
)
```

Código

24 - Criar o arquivo home_screen.dart em lib - ui - home

```
66     ), // List.generate
67     ), // Column
68     SizedBox(height: 64),
69   ],
70   ), // Column
71   ), // SingleChildScrollView
72   ), // Center
73   ), // Padding
74   ); // Scaffold
75 }
76 }
77 }
```

Código

25 - Criar o arquivo restaurant_screen.dart em lib - ui - restaurant

```
lib > ui > restaurant > restaurantscreen.dart > Restaurantscreen > build
1 import 'package:appaula06prep/model/dish.dart';
2 import 'package:appaula06prep/model/restaurant.dart';
3 import 'package:appaula06prep/ui/_core/widgets/app_colors.dart';
4 import 'package:appaula06prep/ui/_core/widgets/appbar.dart';
5 import 'package:appaula06prep/ui/_core/widgets/bag_provider.dart';
6 import 'package:flutter/material.dart';
7
8 import 'package:provider/provider.dart';
9
10 class Restaurantscreen extends StatelessWidget {
11   final Restaurant restaurant;
12   const Restaurantscreen({super.key, required this.restaurant});
13
14   @override
15   Widget build(BuildContext context) {
16     return Scaffold(
17       appBar: getAppBar(context: context, title: restaurant.name),
18       body: Center(
19         child: Column(
20           mainAxisAlignment:
21             MainAxisAlignment.min, // Deixa os filhos centralizados verticalmente
22           mainAxisAlignment: MainAxisAlignment.start,
23           crossAxisAlignment: CrossAxisAlignment.center,
24           children: [
25             Image.asset('assets/${restaurant.imagePath}', width: 128),
26             const SizedBox(height: 12), // espaçamento
27             Text(
28               'Mais pedidos',
29               style: TextStyle(
30                 color: AppColors.mainColor,
31                 fontSize: 18,
32                 fontWeight: FontWeight.bold,
33               ), // TextStyle
34             ), // Text
35             Column(
36               children: List.generate(restaurant.dishes.length, (index) {
```

```
35       Column(
36         children: List.generate(restaurant.dishes.length, (index) {
37           Dish dish = restaurant.dishes[index];
38           return ListTile(
39             leading: Image.asset(
40               'assets/dishes/default.png',
41               width: 48,
42               height: 48,
43             ), // Image.asset
44             title: Text(dish.name),
45             subtitle: Text('R\${dish.price.toStringAsFixed(2)}'),
46             trailing: IconButton(
47               onPressed: () {
48                 context.read<BagProvider>().addAllDishes([dish]);
49               },
50               icon: Icon(Icons.add),
51             ), // IconButton
52           ); // ListTile
53         }, // List.generate
54       ), // Column
55     ], // Column
56   ), // Center
57 ); // Scaffold
58 }
59 }
60 }
61 }
```

Código

26 - Criar o arquivo splash_screen.dart em lib - ui - splashscreen

```
lib > ui > splash > splash_screen.dart > ...
1 import 'package:appaula06prep/ui/_core/widgets/app_colors.dart';
2 import 'package:appaula06prep/ui/home/home_screen.dart';
3 import 'package:flutter/material.dart';
4
5
6 class SplashScreen extends StatelessWidget {
7   const SplashScreen({super.key});
8
9   @override
10  Widget build(BuildContext context) {
11    return Scaffold(
12      backgroundColor: AppColors.backgroundColor,
13      body: Stack(
14        children: [
15          Image.asset('assets/banners/banner_splash.png'),
16          Center(
17            child: Padding(
18              padding: const EdgeInsets.symmetric(horizontal: 25),
19              child: Column(
20                mainAxisAlignment: MainAxisAlignment.center,
21                spacing: 32,
22                children: [
23                  Image.asset('assets/logo.png', width: 192),
24                  Column(
25                    children: [
26                      Text(
27                        "Um parceiro inovador para sua",
28                        style: TextStyle(color: Colors.white, fontSize: 22),
29                      ), // Text
30                      Text(
31                        "melhor experiência culinária",
32                        style: TextStyle(
33                          color: AppColors.mainColor,
34                          fontSize: 22,
35                          fontWeight: FontWeight.bold,
```

```
36                               ), // TextStyle
37                               ), // Text
38                           ],
39                         ),
40                         ), // Column
41                         SizedBox(
42                           width: double.infinity,
43                           child: ElevatedButton(
44                             onPressed: () {
45                               Navigator.pushReplacement(
46                                 context,
47                                 MaterialPageRoute(
48                                   builder: (context) {
49                                     return HomeScreen();
50                                   },
51                                 ),
52                               );
53                               child: Text('Bora !'),
54                             ),
55                           ), // ElevatedButton
56                         ), // SizedBox
57                         ],
58                         ), // Column
59                         ), // Padding
60                         ),
61                         ],
62                         ), // Stack
63                       ); // Scaffold
64                     }
65                   }
```

Código

27 - Criar o arquivo main.dart em lib

```
lib > main.dart > ...
1 import 'package:appaula06prep/data/restaurant_data.dart';
2 import 'package:appaula06prep/ui/_core/widgets/app_theme.dart';
3 import 'package:appaula06prep/ui/_core/widgets/bag_provider.dart';
4 import 'package:appaula06prep/ui/splash/splash_screen.dart';
5 import 'package:flutter/material.dart';
6 import 'package:provider/provider.dart';
7
8 |
Run | Debug | Profile
9 void main() async {
10 WidgetsFlutterBinding.ensureInitialized();
11 RestaurantData restaurantsData = RestaurantData();
12 await restaurantsData.getRestaurant();
13 //List<Restaurant> listRestaurant = await RestaurantsData().getRestaurant();
14 //print(listRestaurant);
15 runApp(
16 MultiProvider(
17 providers: [
18 ChangeNotifierProvider(
19 create: (context) {
20 return restaurantsData;
21 },
22 ), // ChangeNotifierProvider
23 ChangeNotifierProvider(create: (context) => BagProvider(),
24 ],
25 child: MyApp(),
26 ), // MultiProvider
27 );
28 }
29
30 // Cria um novo widget
31
32 class MyApp extends StatelessWidget {
33 const MyApp({super.key});
34
35 @override
```

```
35     @override
36     Widget build(BuildContext context) {
37         return MaterialApp(
38             debugShowCheckedModeBanner: false,
39             theme: AppTheme.appTheme,
40             home: SplashScreen(),
41         ); // MaterialApp
42     }
43 }
44 }
```

Provider

No Flutter, **Provider** é um padrão de gerenciamento de estado.

Ele serve para:

- **Compartilhar dados** entre diferentes partes da aplicação (sem precisar ficar passando via construtor ou parâmetros).
- **Notificar widgets** automaticamente quando o estado mudar.

Provider → É o mecanismo do Flutter para gerenciar e compartilhar estado.

BagProvider → É a sua classe personalizada que usa ChangeNotifier para representar o **estado da sacola** (carrinho).

Widgets podem acessar o estado assim:

```
BagProvider bagProvider = Provider.of<BagProvider>(context);  
E automaticamente serão reconstruídos quando você chamar notifyListeners().
```

Obrigado!

Prof. Me Daniel Vieira

Email: danielvieira2006@gmail.com

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95

