



Estácio

FACULDADE ESTÁCIO

CÂMPUS VOLTA REDONDA – RJ

DESENVOLVIMENTO FULL STACK

DISCIPLINA – POSSO CRIAR UM APP DE OUTRA FORMA?

TURMA – 2023.2

SEMESTRE – 4

VOLTA REDONDA, JANEIRO 2025.

DESENVOLVIMENTO FULL STACK

DISCIPLINA – POSSO CRIAR UM APP DE OUTRA FORMA?

TURMA – 2023.2

SEMESTRE – 4

ALUNO – BRUNO SAMPAIO BASTOS

TUTOR – ALTAMIRA DE SOUZA QUEIROZ

GITHUB - <https://github.com/BrunoTI-Code?tab=repositories>

VOLTA REDONDA, JANEIRO 2025.

1 OBJETIVOS DA PRÁTICA

- Instalar e configurar o Flutter SDK e o ambiente de desenvolvimento integrado (IDE) de acordo com as melhores práticas;
- Empregar Widgets fundamentais, como MaterialApp, Scaffold, AppBar, Text e RaisedButton;
- Aplicar diferentes Widgets para criar uma interface visual atraente e funcional;
- Aplicar o widget ListView para exibir e gerenciar listas de dados;
- Criar e implementar funcionalidades personalizadas para um Widget específico.

2 MISSÃO PRÁTICA | POSSO CRIAR UM APP DE OUTRA FORMA!

Nesta atividade, revisaremos todos os conceitos e práticas abordados nas microatividades anteriores. Além disso, exploraremos a colocação de um único Widget na tela, abrangendo diversos aspectos presentes em alguns dos widgets de layout mais comuns que foram abordados ao longo do curso, entre outros tópicos relevantes.

3 CONTEXTUALIZAÇÃO

A "Explore Mundo", uma Agência de Viagens, está em busca de melhorias para tornar seu aplicativo mais atrativo e funcional para os clientes. O objetivo é proporcionar uma experiência em que os usuários possam explorar destinos, consultar pacotes de viagens, efetuar reservas, entrar em contato com a equipe e obter informações detalhadas sobre a localização e as avaliações de estrelas para cada destino. As principais características desejadas para o app incluem:

- Banner de Destaque:

Um elemento visual destacado, como uma imagem ou slideshow, exibindo fotos irresistíveis dos destinos oferecidos pela agência. Cada imagem será interativa, direcionando os usuários para páginas específicas de destinos ao serem tocadas.

- Barra de Navegação:

Uma barra superior intuitiva, contendo links para diferentes seções do aplicativo, como "destinos", "pacotes de viagem", "contato" e "sobre nós". Essa navegação facilita o acesso dos usuários às áreas relevantes do app.

- Pesquisa Rápida:

Implementação de um campo de pesquisa que permitirá aos usuários inserir destinos específicos ou dados desejados, agilizando a busca por pacotes de viagem.

Além dessas melhorias para o aplicativo, a empresa tem o objetivo de ampliar seu alcance de mercado desenvolvendo um site complementar. Esse site oferecerá aos clientes acesso fácil às informações da agência e a praticidade de efetuar reservas de viagens diretamente por meio de seus dispositivos móveis. Essa estratégia visa proporcionar uma experiência integrada e conveniente para os clientes explorarem as ofertas da agência, tanto no aplicativo quanto no site.

4 ROTEIRO DE PRÁTICA

4.1 - MATERIAL NECESSÁRIO PARA A PRÁTICA

- Editor de texto ou IDE (Sugestões: IntelliJ ou VS Code)
- Flutter SDK
- Android Studio e/ou xCode
- Simulador Android ou iOS
- Navegador Web: Google Chrome, Firefox, MS Edge, Safari ou Opera

5 PROCEDIMENTOS

5.1 CONFIGURAÇÃO DO AMBIENTE:

- Certifique-se de ter seu ambiente configurado.
- Crie um novo aplicativo Flutter.

5.2 ESTRUTURA INICIAL:

- Substitua o conteúdo do arquivo lib/main.dart pelo código fornecido, o qual configura uma estrutura básica para o aplicativo.

```
import 'package:flutter/material.dart';
void main() => runApp(const MyApp());
class MyApp extends StatelessWidget {
const MyApp({super.key});
@Override
Widget build(BuildContext context) {
return MaterialApp(
title: 'Flutter layout demo',
home: Scaffold(
appBar: AppBar(
title: const Text('Flutter layout demo'),
),
body: const Center(
child: Text('Hello World'),
),
),
);
}
}
```

5.3 DIVISÃO DO LAYOUT:

- Analise o layout dividindo-o em elementos básicos.
- Identifique linhas, colunas e áreas que requerem alinhamento, preenchimento ou bordas.
- Verifique a necessidade de grade, sobreposição de elementos ou guias de interface.

5.4 DIAGRAMAÇÃO DO LAYOUT:

- Comece identificando os elementos principais do seu layout. Em nosso exemplo, temos quatro elementos dispostos em uma coluna: uma imagem, duas linhas e um bloco de texto.
- Em seguida, faça um diagrama para cada uma das linhas. A primeira linha, que chamaremos de "Seção Título", possui três elementos filhos: uma coluna de texto, um ícone de estrela e um número. A coluna de texto tem duas linhas de texto dentro dela. Para garantir que essa primeira coluna ocupe o espaço adequado, é aconselhável envolvê-la em um widget "Expandido".
- A segunda linha, que chamaremos de "Seção Button", também possui três elementos filhos. Cada um desses filhos é uma coluna que contém um ícone e um texto.
- Uma vez que você tenha diagnosticado o layout do seu aplicativo, é aconselhável adotar uma abordagem de construção "de baixo para cima" para implementá-lo. Isso significa começar pelos elementos mais internos e, gradualmente, construir a estrutura do layout de forma progressiva.
- Para manter o código organizado e evitar a confusão visual de um código de layout profundamente aninhado, considere a possibilidade de criar variáveis e funções para partes específicas da implementação. Isso ajudará a tornar o código mais legível e fácil de manter.

5.5 CONSTRUÇÃO DA SEÇÃO DE TÍTULO:

- Desenvolva a coluna esquerda na seção do título.
- Utilize o código fornecido para criar a estrutura, com destaque para o uso do Widget Expandido e do Container para gerenciar espaçamentos e estilo.
- Altere os textos e imagens de acordo com as necessidades do cliente.
- Utilize o código de exemplo a seguir para construir o seu próprio aplicativo:

```
// Passo 5: Construindo a Coluna da Seção Título
Widget titleSection = Container(
  padding: const EdgeInsets.all(32),
  child: Row(
    children: [
      Expanded(
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Container(
              padding: const EdgeInsets.only(bottom: 8),
              child: const Text(
                'Oeschinen Lake Campground',
                style: TextStyle(
                  fontWeight: FontWeight.bold,
                ),
                ),
            ),
            Text(
              'Kandersteg, Switzerland',
              style: TextStyle(
                color: Colors.grey[500],
              ),
              ),
            ],
          ),
        Icon(
          Icons.star,
          color: Colors.red[500],
        ),
        const Text('41'),
      ],
    ),
  ),
);
```

```
),  
);
```

Neste passo, estamos construindo a coluna esquerda da seção de título. Usamos um Container para envolver os elementos da coluna e aplicar um espaçamento uniforme em todas as bordas. Dentro da coluna, usamos um Expanded para ocupar todo o espaço restante na linha horizontal. Isso é importante para garantir que a coluna ocupe o espaço disponível.

A propriedade crossAxisAlignment é definida como CrossAxisAlignment.start para alinhar o conteúdo da coluna à esquerda da linha.

Dentro da coluna, temos dois elementos de texto, onde o primeiro é exibido em negrito e o segundo em cinza. Em seguida, adicionamos um ícone de estrela vermelha e o texto "41" à direita da coluna.

Este trecho de código representa a construção da seção de título do aplicativo Flutter. Para incorporá-lo ao corpo do aplicativo, você pode fazer o seguinte:

```
return MaterialApp(  
    title: 'Demonstração de layout Flutter',  
    home: Scaffold(  
        appBar: AppBar(  
            title: const Text('Demonstração de layout Flutter'),  
        ),  
        body: Column(  
            children: [  
                titleSection,  
                // Outros elementos do corpo do aplicativo podem ser adicionados aqui.  
            ],  
        ),  
    ),  
);
```

Assim, a seção de título é adicionada como parte do corpo do aplicativo.

5.6 CONSTRUÇÃO DA SEÇÃO DE BOTÕES:

- Crie um método auxiliar, `_buildButtonColumn()`, para gerar colunas de botões de forma eficiente.
- Utilize esse método para construir a linha de botões, alinhando as colunas uniformemente com `MainAxisAlignment.spaceEvenly`.
- Utilize o código de exemplo a seguir para construir o seu próprio aplicativo:

```
// Passo 6: Construindo a Seção de Botões
class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);
  @override
  Widget build(BuildContext context) {
    // ...
    Column _buildButtonColumn(Color color, IconData icon, String label) {
      return Column(
        mainAxisSize: MainAxisSize.min,
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          Icon(icon, color: color),
          Container(
            margin: const EdgeInsets.only(top: 8),
            child: Text(
              label,
              style: TextStyle(
                fontSize: 12,
                fontWeight: FontWeight.w400,
                color: color,
              ),
            ),
          ),
        ],
      );
    }
}
```

```
}

Color color = Theme.of(context).primaryColor;
Widget buttonSection = Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    _buildButtonColumn(color, Icons.call, 'CALL'),
    _buildButtonColumn(color, Icons.near_me, 'ROUTE'),
    _buildButtonColumn(color, Icons.share, 'SHARE'),
  ],
);
// ...
return MaterialApp(
  title: 'Demonstração de layout Flutter',
  home: Scaffold(
    appBar: AppBar(
      title: const Text('Demonstração de layout Flutter'),
    ),
    body: Column(
      children: [
        titleSection,
        buttonSection,
        // Outros elementos do corpo do aplicativo podem ser adicionados aqui.
      ],
    ),
  ),
);
}
```

Neste passo, estamos construindo a seção de botões do aplicativo Flutter. Criamos um método auxiliar privado `_buildButtonColumn` que aceita uma cor, um ícone e um rótulo, e retorna uma coluna com esses elementos.

Em seguida, chamamos esse método para construir três colunas de botões usando o mesmo layout, mas com diferentes ícones e rótulos. Essas colunas são organizadas em uma linha (Row) e alinhadas uniformemente ao longo do eixo principal usando `MainAxisAlignment.spaceEvenly`.

Por fim, adicionamos a seção de botões como parte do corpo do aplicativo, abaixo da seção de título. O restante do código permanece inalterado.

5.7 DEFINIÇÃO DA SEÇÃO DE TEXTO:

- Estabeleça a seção de texto como uma variável.
- Utilize um Container para gerenciar o preenchimento e adicione o texto desejado, configurando `softWrap` para quebras de linha automáticas.
- Utilize o código de exemplo a seguir para construir o seu próprio aplicativo:

```
// Passo 7: Definindo a Seção de Texto
Widget textSection = Container(
  padding: const EdgeInsets.all(32),
  child: const Text(
    'O Lago Oeschinen fica aos pés do Blüemlisalp nos Alpes Berneses. Situado a 1.578 metros acima do nível do mar, é um dos lagos alpinos mais amplos. Um passeio de teleférico a partir de Kandersteg, seguido por meia hora de caminhada por pastagens e floresta de pinheiros, leva você ao lago, que aquece até 20 graus Celsius no verão. As atividades desfrutadas aqui incluem remo e andar no tobogã de verão.',
    softWrap: true,
  ),
);
// ...
return MaterialApp(
  title: 'Demonstração de layout Flutter',
  home: Scaffold(
    appBar: AppBar(
      title: const Text('Demonstração de layout Flutter'),
    ),
    body: Column(
      children: [
        titleSection,
        buttonSection,
        textSection,
        // Outros elementos do corpo do aplicativo podem ser adicionados aqui.
      ],
    ),
  ),
);
});
```

Neste passo, estamos definindo a seção de texto do aplicativo Flutter. O texto é colocado em um Container com preenchimento ao longo de todas as bordas. Usamos o widget Text para exibir o texto e configuramos softWrap como verdadeiro para permitir que as linhas de texto preencham a largura da coluna antes de serem quebradas no limite da palavra.

Depois, adicionamos a seção de texto ao corpo do aplicativo, abaixo das seções de título e botões. Isso completa a construção da interface do aplicativo, e você pode continuar adicionando outros elementos ao corpo conforme necessário. O restante do código permanece inalterado.

5.8 ADICIONANDO UMA IMAGEM:

- Crie um diretório images e adicione a imagem lake.jpg.
- Atualize o arquivo pubspec.yaml para incluir a imagem como um recurso.
- Referencie a imagem no código, ajustando seu tamanho e comportamento com BoxFit.cover.
- Utilize os passos e o código de exemplo a seguir para construir o seu próprio aplicativo:

Para adicionar uma imagem ao seu aplicativo, siga estas etapas:

- Crie um diretório chamado "images" no diretório principal do seu projeto, se ainda não existir.
- Adicione o arquivo da imagem que você deseja colocar no aplicativo dentro da pasta "images".
- Atualize o arquivo "pubspec.yaml" para incluir uma seção "assets", que disponibiliza a imagem para o seu código. Certifique-se de que o arquivo "pubspec.yaml" fique semelhante ao seguinte exemplo (a imagem de exemplo foi chamada de lake.jpg):

```
flutter:
```

```
assets:
```

```
- images/lake.jpg
```

Agora que a imagem está disponível para o seu aplicativo, você pode referenciá-la em seu código da seguinte maneira:

```
// ...  
body: Column(  
    children: [  
        Image.asset(  
            'images/lake.jpg',  
            width: 600,  
            height: 240,  
            fit: BoxFit.cover,  
        ),  
        titleSection,  
        buttonSection,  
        textSection,  
        // Outros elementos do corpo do aplicativo podem ser adicionados aqui.  
    ],  
,  
// ...
```

Neste código, usamos `Image.asset` para carregar a imagem selecionada (no exemplo acima o arquivo "lake.jpg") do diretório de ativos "images". Configuramos a largura e a altura da imagem e usamos `BoxFit.cover` para garantir que a imagem seja dimensionada para cobrir toda a caixa de renderização.

Com essas etapas, você adicionou com sucesso uma imagem ao seu aplicativo Flutter e a exibiu na interface do usuário. Você pode personalizar o posicionamento e o tamanho da imagem conforme necessário para atender aos requisitos do seu aplicativo.

Lembre-se de adaptar o código e as imagens de acordo com o seu projeto.

5.9 ORGANIZANDO OS ELEMENTOS EM UMA LISTVIEW:

- Substitua o uso de Column por ListView para suportar rolagem em dispositivos menores.
- Utilize os passos e o código de exemplo a seguir para construir o seu próprio aplicativo:

Nesta etapa final, organizaremos todos os elementos em uma ListView em vez de uma Column. Isso é útil quando o aplicativo precisa de suporte à rolagem do corpo em dispositivos com telas menores.

```
return MaterialApp(  
    title: 'Demonstração de layout Flutter',  
    home: Scaffold(  
        appBar: AppBar(  
            title: const Text('Demonstração de layout Flutter'),  
        ),  
        body: ListView(  
            children: [  
                Image.asset(  
                    'images/lake.jpg',  
                    width: 600,  
                    height: 240,  
                    fit: BoxFit.cover,  
                ),  
                titleSection,  
                buttonSection,  
                textSection,  
                // Outros elementos do corpo do aplicativo podem ser adicionados aqui.  
            ],  
        ),  
    ),  
);
```

Neste código, substituímos a Column por uma ListView. A ListView permite que os elementos sejam rolados verticalmente quando a tela do dispositivo é pequena, garantindo que todos os conteúdos sejam acessíveis. Os elementos, como a imagem, a seção do título, a seção de botões e a seção de texto, são adicionados como filhos da ListView. Agora, o corpo do aplicativo oferece suporte à rolagem quando necessário, proporcionando uma experiência de usuário mais flexível em diferentes tamanhos de tela.

6 RESULTADOS ESPERADOS

Os passos anteriores visam a construção de um aplicativo completo com a estrutura básica do Flutter, garantindo que as interfaces de usuário sejam responsivas e acessíveis em dispositivos móveis, tablets e desktops. Essa plataforma permitirá que os clientes compartilhem avaliações e comentários sobre suas experiências de viagem. Além disso, a Agência de viagens utilizará widgets interativos do Flutter para exibir as avaliações dos clientes e interagir com eles de forma envolvente, compartilhando suas opiniões e proporcionando uma experiência interativa e informativa aos usuários do site. A imagem a seguir ilustra um exemplo de aplicativo desenvolvido. Adapte os códigos apresentados para que você possa criar a sua solução.

7 REFERÊNCIAS

- https://pt.wikipedia.org/wiki/Balneário_Camboriú
- https://pt.wikipedia.org/wiki/Ouro_Preto
- <https://pt.wikipedia.org/wiki/Gramado>

8 RESULTADOS DA EXECUÇÃO DOS CÓDIGOS

Viagem ao Mundo DEBUG

Destinos	Pacotes	Contato	Sobre Nós
 Balneário Camboriú Santa Catarina			★ 10
 Ouro Preto Minas Gerais			★ 10
 Gramado Rio Grande do Sul			★ 10

← Gramado DEBUG



Gramado ★ 41
Rio Grande do Sul

📞 CONTATO 📍 ROTA 🔗 COMPARTILHAR

Gramado é um município do estado do Rio Grande do Sul, no Brasil. Localiza-se na Serra Gaúcha, mais precisamente na Região das Hortênsias. O município de Gramado foi criado pela Lei 2 522, de 15 de dezembro de 1954, após ser emancipado da cidade de Taquara. A região era habitada por índios caingangues e, posteriormente, foi colonizada por descendentes de açorianos, imigrantes alemães e italianos. Sua demografia é etnicamente variada, com forte influência alemã e italiana, o que se reflete especialmente na culinária e na arquitetura urbana e rural. Gramado também foi colonizado, em menor número, por portugueses, sírios e libaneses. Com uma economia voltada ao turismo, a cidade recebe anualmente milhões de turistas nacionais e estrangeiros.

```
class Destination {  
    final String name;  
    final String location;  
    final String imagePath;  
    final String description;  
  
    Destination(this.name, this.location, this.imagePath, this.description);  
  
    static List<Destination> getDestinations() {  
        return [  
            Destination('Balneário Camboriú', 'Santa Catarina', 'assets/balneario.jpg',  
                'Balneário Camboriú é um município brasileiro localizado no litoral norte do estado de Santa Catarina, Região Sul do país. Pertence à Região Metropolitana de Florianópolis.'),  
            Destination('Ouro Preto', 'Minas Gerais', 'assets/ouropreto.jpg',  
                'Ouro Preto é um município brasileiro localizado no estado de Minas Gerais, na Região Sudeste do país. Ouro Preto localiza-se em uma das principais áreas de mineração do Brasil.'),  
            Destination('Gramado', 'Rio Grande do Sul', 'assets/gramado.jpg',  
                'Gramado é um município do estado do Rio Grande do Sul, no Brasil. Localiza-se na Serra Gaúcha, mais precisamente na Região das Hortênsias. O município de Gramado é conhecido por sua paisagem montanhosa, clima suave e cultura alemã.'),  
        ];  
    }  
}
```

Viagem ao Mundo

DEBUG

Destinos

Pacotes

Contato

Sobre Nós

Viagem ao Mundo: Venha Viver essa Experiência

Explore o mundo com a Viagem ao Mundo ! Transforme seus sonhos de viagem em realidade com experiências personalizadas e destinos inesquecíveis. De praias paradisíacas a aventuras nas montanhas, oferecemos pacotes sob medida para cada tipo de viajante. Com conforto, segurança e os melhores preços, levamos você para onde o coração deseja explorar. Planeje sua próxima jornada conosco e crie memórias para a vida toda! Viagem ao Mundo – Seu próximo destino começa aqui!

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

class AboutScreen extends StatelessWidget {
  const AboutScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return const Scaffold(
      body: Padding(
        padding: EdgeInsets.all(16.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.start,
          children: [
            Text(
              'Viagem ao Mundo: Venha Viver essa Experiência',
              style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
            ),
            SizedBox(height: 20),
            Text(
              'Explore o mundo com a Viagem ao Mundo ! Transforme seus sonhos de viagem em realidade com experiências personalizadas e destinos inesquecíveis. De praias paradisíacas a aventuras nas montanhas, oferecemos pacotes sob medida para cada tipo de viajante. Com conforto, segurança e os melhores preços, levamos você para onde o coração deseja explorar. Planeje sua próxima jornada conosco e crie memórias para a vida toda! Viagem ao Mundo – Seu próximo destino começa aqui!',
              style: TextStyle(fontSize: 16),
            ),
          ],
        ),
      );
  }
}
```

Viagem ao Mundo

DEBUG

Destinos

Pacotes

Contato

Sobre Nós

Entre em contato conosco:

E-mail: ViagemaoMundo@contato.com.br

Telefone: (58) 92357-9853



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

class ContactScreen extends StatelessWidget {
  const ContactScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.start,
          children: [
            const Text(
              'Entre em contato conosco:',
              style: TextStyle(fontSize: 18),
            ),
            const SizedBox(height: 20),
            const Text(
              'E-mail: ViagemaoMundo@contato.com.br',
              style: TextStyle(fontSize: 16),
            ),
            const SizedBox(height: 10),
            const Text(
              'Telefone: (58) 92357-9853',
              style: TextStyle(fontSize: 16),
            ),
            const SizedBox(height: 20),
            Row(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                IconButton(
                  onPressed: () {},
                  icon: const Icon(Icons.facebook),
                ),
              ],
            ),
          ],
        );
  }
}
```

Viagem ao Mundo

DEBUG

Destinos

Pacotes

Contato

Sobre Nós



Balneário Camboriú
Santa Catarina

\$ 1520



Ouro Preto
Minas Gerais

\$ 1520



Gramado
Rio Grande do Sul

\$ 1520

```
import 'package:flutter/material.dart';
import 'package:exploremundo/models/destination.dart';

class PackagesScreen extends StatelessWidget {
  const PackagesScreen({super.key});

  @override
  Widget build(BuildContext context) {
    List<Destination> destinations = Destination.getDestinations();

    return ListView.builder(
      itemCount: destinations.length,
      itemBuilder: (context, index) {
        var destination = destinations[index];
        return Card(
          child: ListTile(
            leading: Container(
              width: 100,
              height: 100,
              decoration: BoxDecoration(
                image: DecorationImage(
                  image: AssetImage(destination.imagePath),
                  fit: BoxFit.cover,
                ),
              ),
            ),
            title: Row(
              children: [
                Expanded(
                  child: Text(destination.name),
                ),
                const Icon(Icons.attach_money, color: Colors.green),
                const Text("1520"),
              ],
            ),
            subtitle: Text(destination.location),
          );
      );
  }
}
```


← Balneário Camboriú

DEBUG

Balneário Camboriú

← natal

DEBUG

Destino não encontrado.

```
import 'package:flutter/material.dart';
import 'package:exploremundo/models/destination.dart';
import 'package:exploremundo/screens/destination_detail_screen.dart';

class CustomSearchDelegate extends SearchDelegate {
  List<Destination> destinations = Destination.getDestinations();
  @override
  Widget buildSuggestions(BuildContext context) {
    final suggestions = query.isEmpty
        ? []
        : destinations
            .where((destination) =>
                destination.name.toLowerCase().contains(query.toLowerCase()))
            .toList();

    return ListView.builder(
      itemCount: suggestions.length,
      itemBuilder: (context, index) {
        var destination = suggestions[index];
        return ListTile(
          title: Text(destination.name ?? 'Destino Desconhecido'),
          onTap: () {
            query = destination.name ?? '';
            showResults(context);
          },
        );
      },
    );
  }

  @override
  Widget buildResults(BuildContext context) {
    final results = destinations
        .where((destination) =>
            destination.name.toLowerCase().contains(query.toLowerCase()))
        .toList();

    if (results.isEmpty) {
      return const Center(
        child: Text('Destino não encontrado.'),
      );
    }
  }
}
```

```
}

return ListView.builder(
  itemCount: results.length,
  itemBuilder: (context, index) {
    var destination = results[index];
    return Card(
      child: ListTile(
        leading: Container(
          width: 100,
          height: 100,
          decoration: BoxDecoration(
            image: DecorationImage(
              image: AssetImage(destination.imagePath),
              fit: BoxFit.cover,
            ),
          ),
        ),
        title: Row(
          children: [
            Expanded(
              child: Text(destination.name),
            ),
            const Icon(Icons.star, color: Colors.amber),
            const Text("10"),
          ],
        ),
        subtitle: Text(destination.location),
        onTap: () {
          Navigator.push(
            context,
            MaterialPageRoute(
              builder: (context) =>
                DestinationDetailScreen(destination: destination),
            ),
          );
        },
      );
    );
  },
)
```

```
}

@Override
Widget buildLeading(BuildContext context) {
    return IconButton(
        icon: const Icon(Icons.arrow_back),
        onPressed: () => close(context, null),
    );
}

@Override
List<Widget> buildActions(BuildContext context) {
    return [
        IconButton(
            icon: const Icon(Icons.clear),
            onPressed: () {
                query = '';
                showSuggestions(context);
            },
        ),
    ];
}
```

```
import 'package:flutter/material.dart';
import 'package:exploremundo/models/destination.dart';

class DestinationDetailScreen extends StatelessWidget {
  final Destination destination;

  const DestinationDetailScreen({super.key, required this.destination});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(destination.name),
        leading: IconButton(
          icon: const Icon(Icons.arrow_back),
          onPressed: () => Navigator.of(context).pop(),
        ),
      ),
      body: ListView(
        children: [
          ImageSection(imagePath: destination.imagePath),
          TitleSection(name: destination.name, location: destination.location),
          const ButtonSection(),
          TextSection(description: destination.description),
        ],
      ),
    );
  }
}

class ImageSection extends StatelessWidget {
  final String imagePath;
  const ImageSection({super.key, required this.imagePath});

  @override
  Widget build(BuildContext context) {
    return Image.asset(
      imagePath,
      width: MediaQuery.of(context).size.width,
      height: 240,
      fit: BoxFit.cover,
    );
  }
}
```

```
| );
}
}

class TitleSection extends StatelessWidget {
    final String name;
    final String location;

    const TitleSection({super.key, required this.name, required this.location});

    @override
    Widget build(BuildContext context) {
        return Container(
            padding: const EdgeInsets.all(32),
            child: Row(
                children: [
                    Expanded(
                        child: Column(
                            crossAxisAlignment: CrossAxisAlignment.start,
                            children: [
                                Container(
                                    padding: const EdgeInsets.only(bottom: 8),
                                    child: Text(
                                        name,
                                        style: const TextStyle(fontWeight: FontWeight.bold),
                                    ),
                                ),
                                Text(
                                    location,
                                    style: TextStyle(
                                        color: Colors.grey[500],
                                    ),
                                ),
                            ],
                        ),
                    ),
                    const Icon(Icons.star, color: Colors.amber),
                    const Text('41'),
                ],
            ),
        );
    }
}
```

```
    );
}

}

class ButtonSection extends StatelessWidget {
  const ButtonSection({super.key});

  @override
  Widget build(BuildContext context) {
    Color color = Theme.of(context).primaryColor;
    return Row(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: [
        _buildButtonColumn(color, Icons.call, 'CONTATO'),
        _buildButtonColumn(color, Icons.near_me, 'ROTA'),
        _buildButtonColumn(color, Icons.share, 'COMPARTILHAR'),
      ],
    );
  }

  Column _buildButtonColumn(Color color, IconData icon, String label) {
    return Column(
      mainAxisSize: MainAxisSize.min,
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Icon(icon, color: color),
        Container(
          margin: const EdgeInsets.only(top: 8),
          child: Text(
            label,
            style: TextStyle(
              fontSize: 12,
              fontWeight: FontWeight.w400,
              color: color,
            ),
          ),
        ),
      ],
    );
  }
}
```

```
class TextSection extends StatelessWidget {
  final String description;
  const TextSection({super.key, required this.description});

  @override
  Widget build(BuildContext context) {
    return Container(
      padding: const EdgeInsets.all(32),
      child: Text(
        description,
        softWrap: true,
      ),
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'package:exploremundo/models/destination.dart';
import 'package:exploremundo/screens/destination_detail_screen.dart';

class DestinationsTab extends StatelessWidget {
  const DestinationsTab({super.key});

  @override
  Widget build(BuildContext context) {
    List<Destination> destinations = Destination.getDestinations();

    return ListView.builder(
      itemCount: destinations.length,
      itemBuilder: (context, index) {
        var destination = destinations[index];
        return Card(
          child: ListTile(
            leading: Container(
              width: 100,
              height: 100,
              decoration: BoxDecoration(
                image: DecorationImage(
                  image: AssetImage(destination.imagePath),
                  fit: BoxFit.cover,
                ),
              ),
            ),
            title: Row(
              children: [
                Expanded(
                  child: Text(destination.name),
                ),
                const Icon(Icons.star, color: Colors.amber),
                const Text("10"),
              ],
            ),
            subtitle: Text(destination.location),
            onTap: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (context) =>
                    MaterialPageRoute(
                      builder: (context) =>
                        DestinationDetailScreen(destination: destination),
                    ),
                );
            };
          );
        );
      },
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'package:exploremundo/screens/about_screen.dart';
import 'package:exploremundo/screens/contact_screen.dart';
import 'package:exploremundo/screens/destinations_tab.dart';
import 'package:exploremundo/screens/packages_screen.dart';
import 'package:exploremundo/widgets/custom_search_delegate.dart';

class HomeScreen extends StatelessWidget {
  const HomeScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return DefaultTabController(
      length: 4,
      child: Scaffold(
        appBar: AppBar(
          title: const Text('Viagem ao Mundo'),
          actions: [
            IconButton(
              icon: const Icon(Icons.search),
              onPressed: () {
                showSearch(context: context, delegate: CustomSearchDelegate());
              },
            ),
          ],
        ),
        bottom: const TabBar(
          tabs: [
            Tab(text: 'Destinos'),
            Tab(text: 'Pacotes'),
            Tab(text: 'Contato'),
            Tab(text: 'Sobre Nós'),
          ],
        ),
        body: const TabBarView(
          children: [
            DestinationsTab(),
            PackagesScreen(),
            ContactScreen(),
            AboutScreen(),
          ],
        ),
      ),
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'package:exploremundo/screens/home_screen.dart';

void main() => runApp(const MyApp());

class MyApp extends StatelessWidget {
    const MyApp({super.key});

    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            title: 'Viajem Mundo',
            theme: ThemeData(
                primarySwatch: Colors.blue,
                visualDensity: VisualDensity.adaptivePlatformDensity,
            ),
            home: const HomeScreen(),
        );
    }
}
```