Experiment 5

Name:Mohit Tarachandani
Div: D15A
Roll no: 62

Aim: To apply navigation, routing and gestures in Flutter App

Theory:

1. Navigation:
   - Navigation refers to the process of moving between different screens or pages within a Flutter app.
   - In Flutter, navigation is typically managed using the Navigator class, which maintains a stack of routes.
   - Each route represents a screen or page in the app, and the navigator manages the navigation stack, allowing users to move forward and backward between routes.
   - Navigation can be triggered by user actions such as tapping buttons, selecting items from lists, or swiping between pages.

2. Routing:
   - Routing is the mechanism used to define and manage the routes within a Flutter app.
   - Routes are defined using route names and associated with corresponding widgets or screens.
   - Flutter provides several routing mechanisms, including named routes, on-the-fly routes, and nested routes.
   - Named routes allow developers to define routes with unique names and navigate to them using the Navigator based on these names.
   - On-the-fly routes are created dynamically at runtime and pushed onto the navigation stack as needed.
   - Nested routes involve embedding navigators within other navigators to create complex navigation structures, such as tab-based navigation or drawer navigation.

3. Gestures:
   - Gestures refer to user interactions such as tapping, dragging, swiping, pinching, and rotating on the screen.
   - Flutter provides a rich set of gesture recognition widgets and APIs to handle user gestures effectively.
   - Common gesture recognition widgets include GestureDetector, InkWell, InkResponse, Draggable, Dismissible, etc.
   - These widgets allow developers to detect various user gestures and trigger corresponding actions or animations in response.

- Gestures can be used to implement interactive UI elements, such as buttons, sliders, swipers, drag-and-drop interfaces, and more.

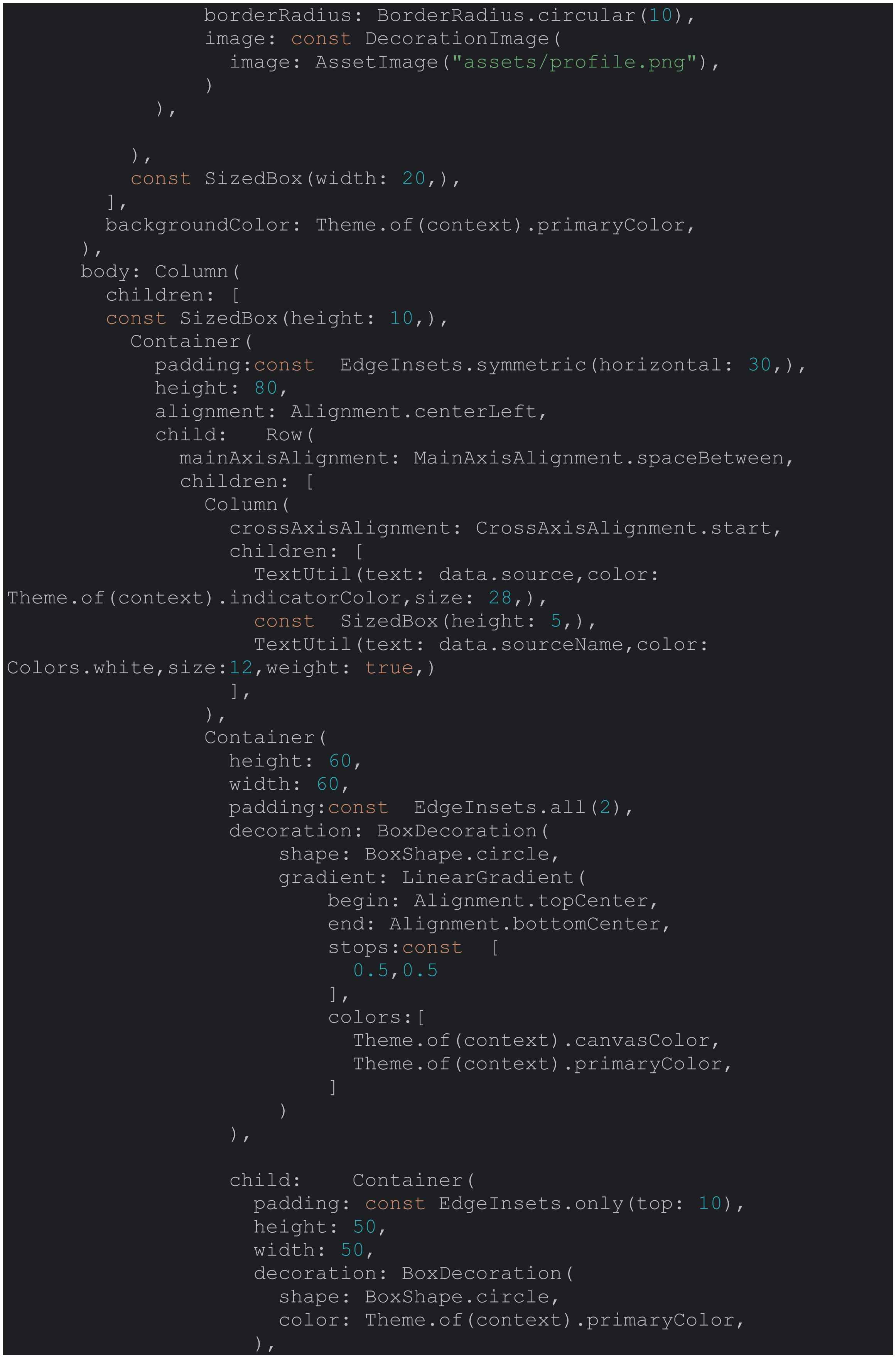
4. Gesture Detection:

    - Gesture detection in Flutter involves registering gesture recognizers on widgets to detect specific user interactions.
    - Gesture recognizers analyze touch input and determine whether a specific gesture has occurred, such as a tap, double-tap, long-press, drag, etc.
    - Once a gesture is detected, Flutter invokes the corresponding callback function associated with the gesture recognizer.
    - Developers can customize gesture detection by configuring properties such as gesture sensitivity, velocity thresholds, and touch area boundaries.

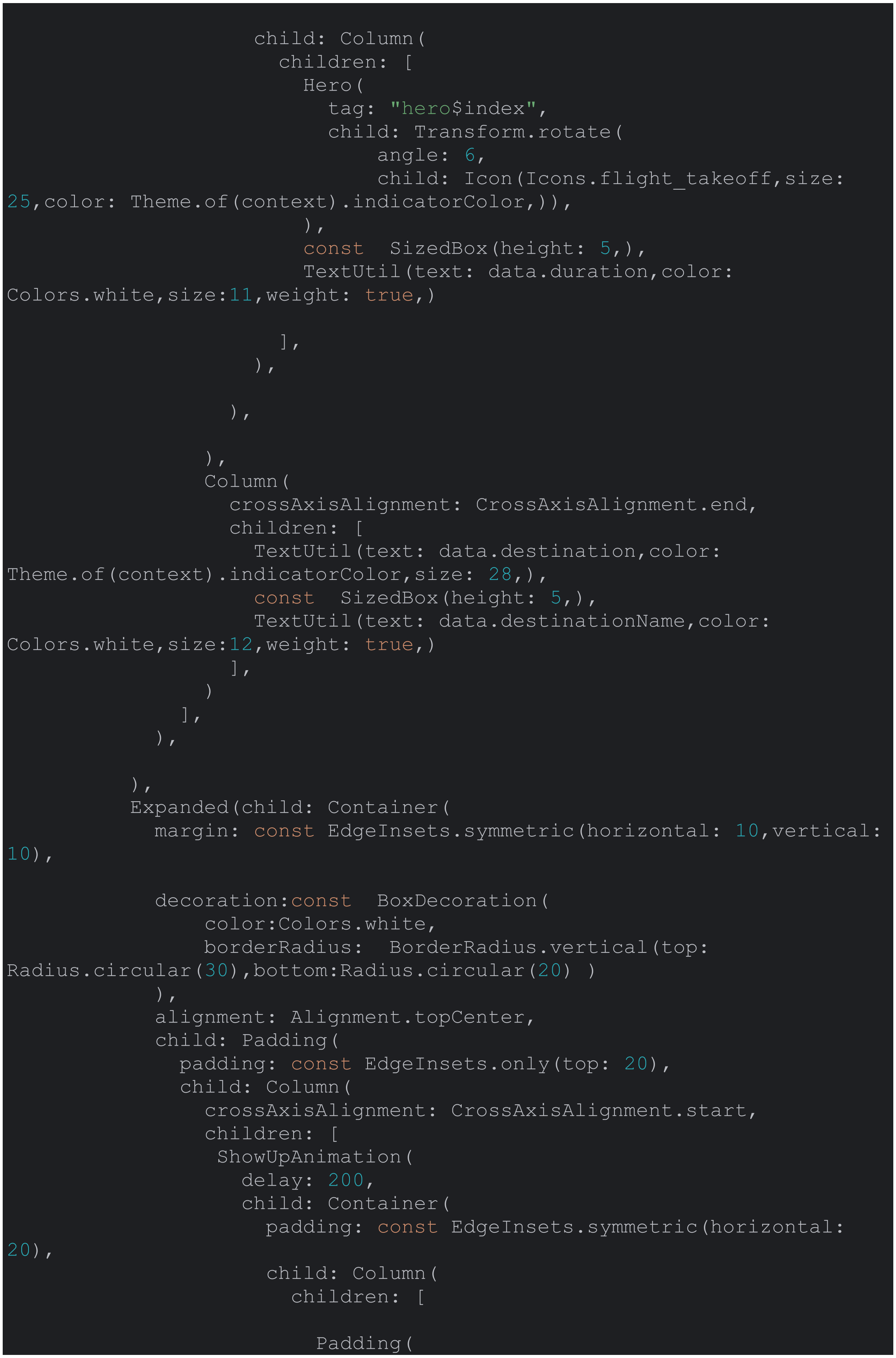
5. Gesture Handling:
    - After a gesture is detected, developers can handle it by performing various actions, such as updating UI state, navigating between screens, triggering animations, or executing business logic.
    - Gesture handling involves responding to user interactions in a way that provides feedback and enhances the user experience.
    - Flutter's declarative programming model makes it easy to update UI elements in response to user gestures, ensuring a smooth and responsive user interface.

6.

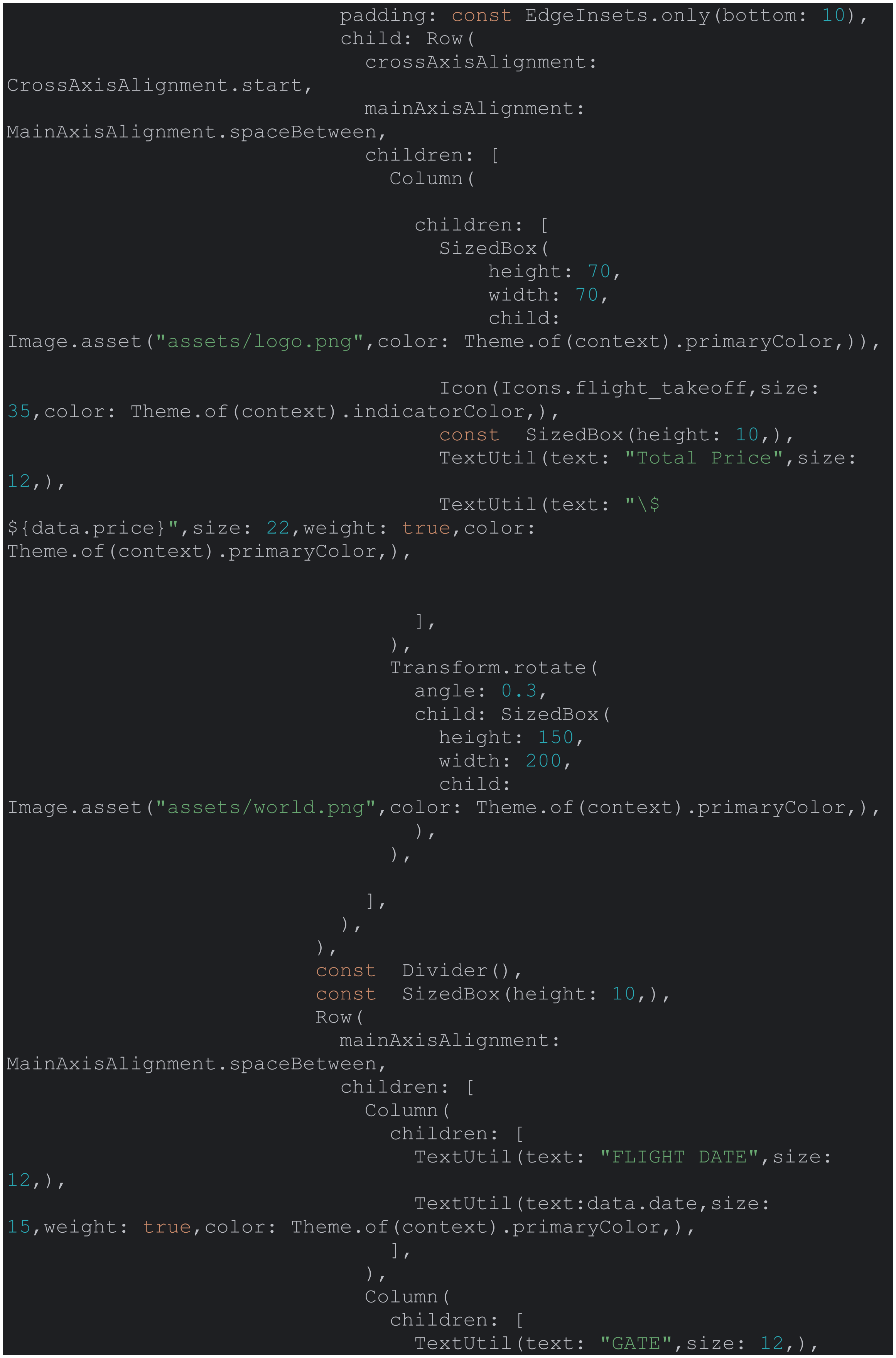
```
import 'package:flutter/material.dart';

import '../data/flight_data.dart';
import '../widgets/show_up_animation.dart';
import '../widgets/text.dart';
class DetailScreen extends StatelessWidget {
  FlightItem data;
  int index;

  DetailScreen({super.key,required this.data,required this.index});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor:  Theme.of(context).primaryColor,
      appBar: AppBar(
        elevation: 0,
        leading: GestureDetector(
          onTap: ()=>Navigator.pop(context),
            child: Icon(Icons.arrow_back_outlined,color:
Theme.of(context).canvasColor,)),
        actions: [
          Container(
            height: 50,
            width: 50,
            decoration: BoxDecoration(
```



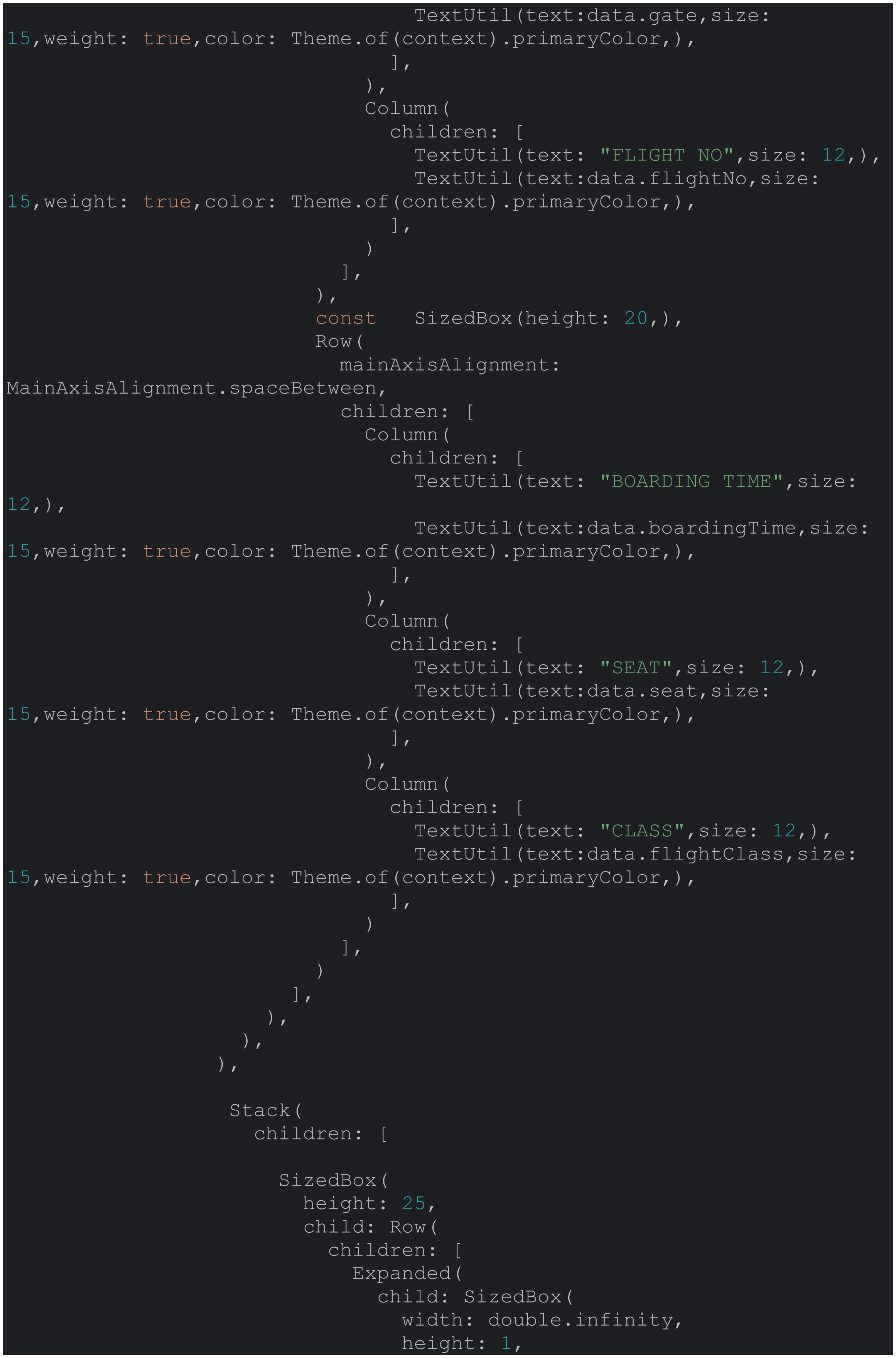
```
                borderRadius: BorderRadius.circular(10),
                image: const DecorationImage(
                  image: AssetImage("assets/profile.png"),
                )
            ),

          ),
          const SizedBox(width: 20,),
        ],
        backgroundColor: Theme.of(context).primaryColor,
      ),
      body: Column(
        children: [
        const SizedBox(height: 10,),
          Container(
            padding:const  EdgeInsets.symmetric(horizontal: 30,),
            height: 80,
            alignment: Alignment.centerLeft,
            child:   Row(
              mainAxisAlignment: MainAxisAlignment.spaceBetween,
              children: [
                Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    TextUtil(text: data.source,color:
Theme.of(context).indicatorColor,size: 28,),
                    const  SizedBox(height: 5,),
                    TextUtil(text: data.sourceName,color:
Colors.white,size:12,weight: true,)
                  ],
                ),
                Container(
                  height: 60,
                  width: 60,
                  padding:const  EdgeInsets.all(2),
                  decoration: BoxDecoration(
                      shape: BoxShape.circle,
                      gradient: LinearGradient(
                          begin: Alignment.topCenter,
                          end: Alignment.bottomCenter,
                          stops:const  [
                            0.5,0.5
                          ],
                          colors:[
                            Theme.of(context).canvasColor,
                            Theme.of(context).primaryColor,
                          ]
                      )
                  ),

                  child:    Container(
                    padding: const EdgeInsets.only(top: 10),
                    height: 50,
                    width: 50,
                    decoration: BoxDecoration(
                      shape: BoxShape.circle,
                      color: Theme.of(context).primaryColor,
                    ),
```



```
                    child: Column(
                      children: [
                        Hero(
                          tag: "hero$index",
                          child: Transform.rotate(
                              angle: 6,
                              child: Icon(Icons.flight_takeoff,size:
25,color: Theme.of(context).indicatorColor,)),
                        ),
                        const  SizedBox(height: 5,),
                        TextUtil(text: data.duration,color:
Colors.white,size:11,weight: true,)

                      ],
                    ),

                  ),

                ),
                Column(
                  crossAxisAlignment: CrossAxisAlignment.end,
                  children: [
                    TextUtil(text: data.destination,color:
Theme.of(context).indicatorColor,size: 28,),
                    const  SizedBox(height: 5,),
                    TextUtil(text: data.destinationName,color:
Colors.white,size:12,weight: true,)
                  ],
                )
              ],
            ),

          ),
          Expanded(child: Container(
            margin: const EdgeInsets.symmetric(horizontal: 10,vertical:
10),

            decoration:const  BoxDecoration(
                color:Colors.white,
                borderRadius:  BorderRadius.vertical(top:
Radius.circular(30),bottom:Radius.circular(20) )
            ),
            alignment: Alignment.topCenter,
            child: Padding(
              padding: const EdgeInsets.only(top: 20),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                 ShowUpAnimation(
                   delay: 200,
                   child: Container(
                     padding: const EdgeInsets.symmetric(horizontal:
20),
                     child: Column(
                       children: [

                         Padding(
```

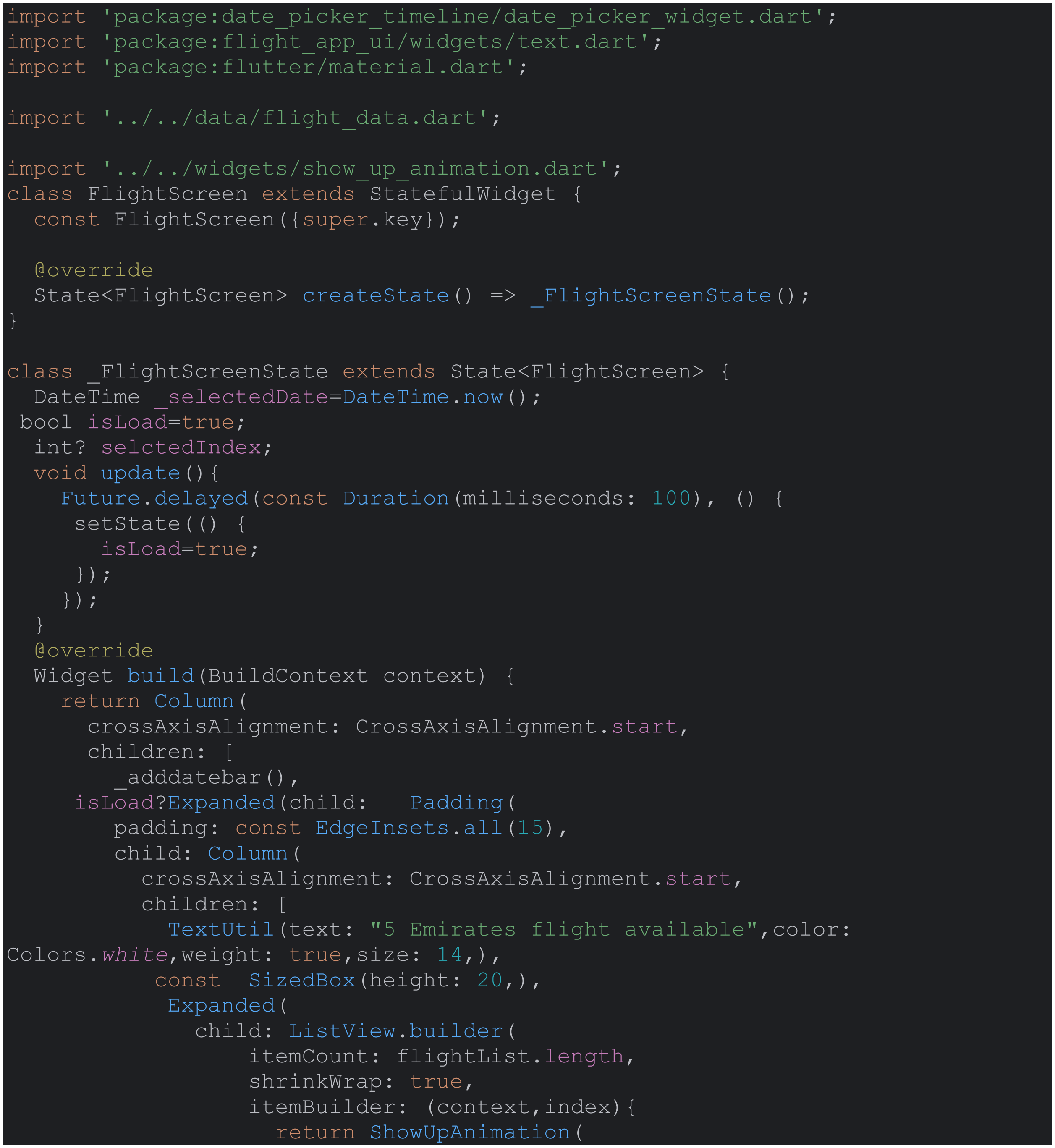


```
                           padding: const EdgeInsets.only(bottom: 10),
                           child: Row(
                             crossAxisAlignment:
CrossAxisAlignment.start,
                             mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                             children: [
                               Column(

                                 children: [
                                   SizedBox(
                                       height: 70,
                                       width: 70,
                                       child:
Image.asset("assets/logo.png",color: Theme.of(context).primaryColor,)),

                                   Icon(Icons.flight_takeoff,size:
35,color: Theme.of(context).indicatorColor,),
                                   const  SizedBox(height: 10,),
                                   TextUtil(text: "Total Price",size:
12,),

                                   TextUtil(text: "\$
${data.price}",size: 22,weight: true,color:
Theme.of(context).primaryColor,),


                                 ],
                               ),
                               Transform.rotate(
                                 angle: 0.3,
                                 child: SizedBox(
                                   height: 150,
                                   width: 200,
                                   child:
Image.asset("assets/world.png",color: Theme.of(context).primaryColor,),
                                 ),
                               ),

                             ],
                           ),
                         ),
                         const  Divider(),
                         const  SizedBox(height: 10,),
                         Row(
                           mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                           children: [
                             Column(
                               children: [
                                 TextUtil(text: "FLIGHT DATE",size:
12,),

                                 TextUtil(text:data.date,size:
15,weight: true,color: Theme.of(context).primaryColor,),
                               ],
                             ),
                             Column(
                               children: [
                                 TextUtil(text: "GATE",size: 12,),
```

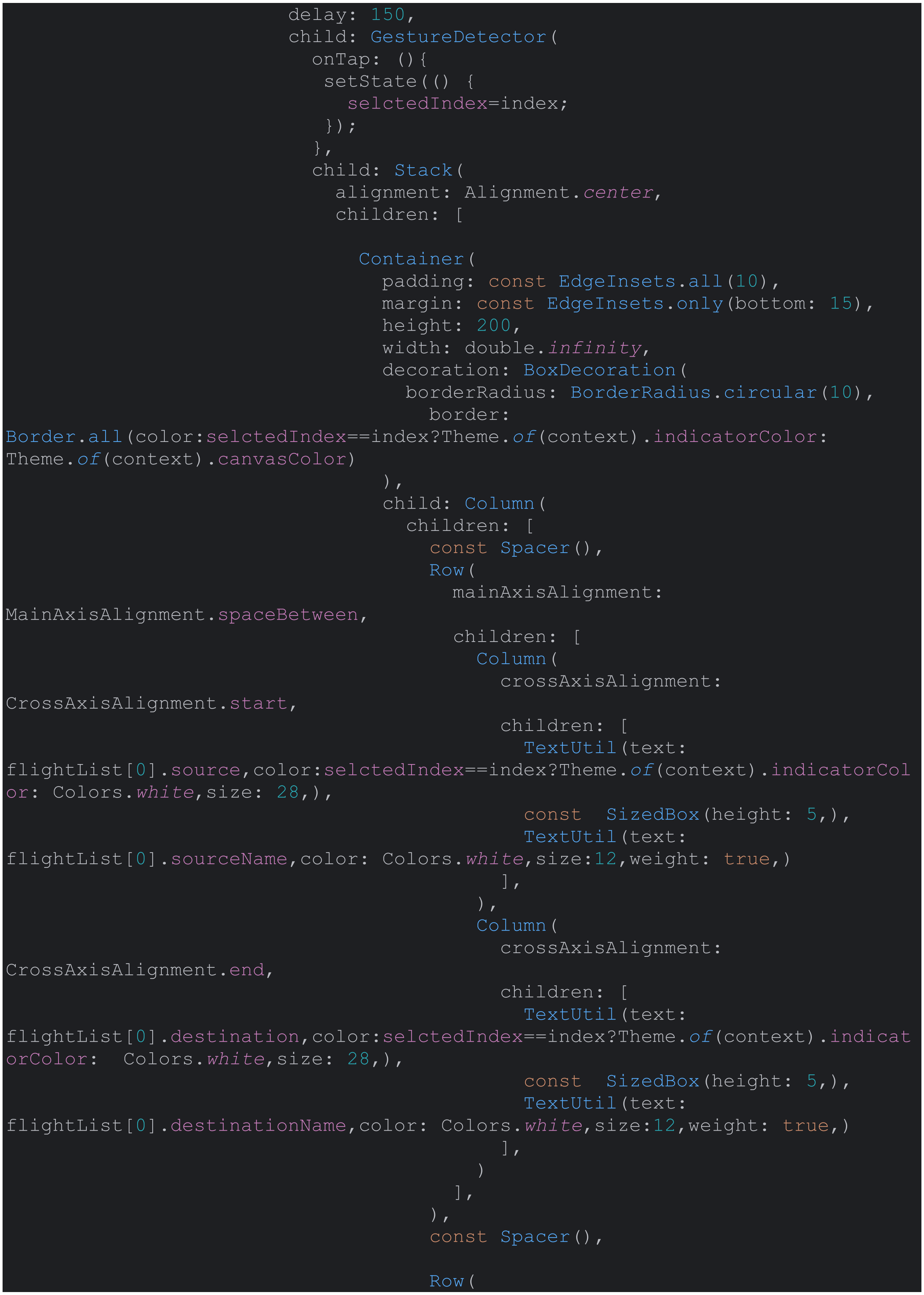


```
                                 TextUtil(text:data.gate,size:
15,weight: true,color: Theme.of(context).primaryColor,),
                               ],
                             ),
                             Column(
                               children: [
                                 TextUtil(text: "FLIGHT NO",size: 12,),
                                 TextUtil(text:data.flightNo,size:
15,weight: true,color: Theme.of(context).primaryColor,),
                               ],
                             )
                           ],
                         ),
                         const   SizedBox(height: 20,),
                         Row(
                           mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                           children: [
                             Column(
                               children: [
                                 TextUtil(text: "BOARDING TIME",size:
12,),
                                 TextUtil(text:data.boardingTime,size:
15,weight: true,color: Theme.of(context).primaryColor,),
                               ],
                             ),
                             Column(
                               children: [
                                 TextUtil(text: "SEAT",size: 12,),
                                 TextUtil(text:data.seat,size:
15,weight: true,color: Theme.of(context).primaryColor,),
                               ],
                             ),
                             Column(
                               children: [
                                 TextUtil(text: "CLASS",size: 12,),
                                 TextUtil(text:data.flightClass,size:
15,weight: true,color: Theme.of(context).primaryColor,),
                               ],
                             )
                           ],
                         )
                       ],
                     ),
                   ),
                 ),

                  Stack(
                    children: [

                      SizedBox(
                        height: 25,
                        child: Row(
                          children: [
                            Expanded(
                              child: SizedBox(
                                width: double.infinity,
                                height: 1,
                                child: Row(
                                  children: List.generate(
                                      700 ~/ 10,
                                          (index) => Expanded(
                                        child: Container(
                                          color:
                                          index % 2 == 0 ?
Colors.transparent :Theme.of(context).canvasColor,
                                          height: 2,
                                        ),
                                      )),
                                ),
                              ),
                            ),
                          ],
                        ),
                      ),
                      Positioned(
                        left: -10,
                          bottom: 0,
                          child: CircleAvatar(
                            radius: 12,
                            backgroundColor:
Theme.of(context).primaryColor,
                          )),
                      Positioned(
                          right: -10,
                          bottom: 0,
                          child: CircleAvatar(
                            radius: 12,
                            backgroundColor:
Theme.of(context).primaryColor,
                          ))
                    ],
                  ),
                  ShowUpAnimation(
                    delay: 300,
                    child: Column(

                      children: [

                       Center(child: TextUtil(text: "Boarding
pass",color: Theme.of(context).primaryColor,weight: true,)),
                        Center(
                          child: SizedBox(
                            width: 280,
                            height: 100,
                            child:
Image.asset("assets/barcode.webp",fit: BoxFit.fill,),
                          ),
                        ),

                      ],
                    ),
                  ),

                ],
              ),
```

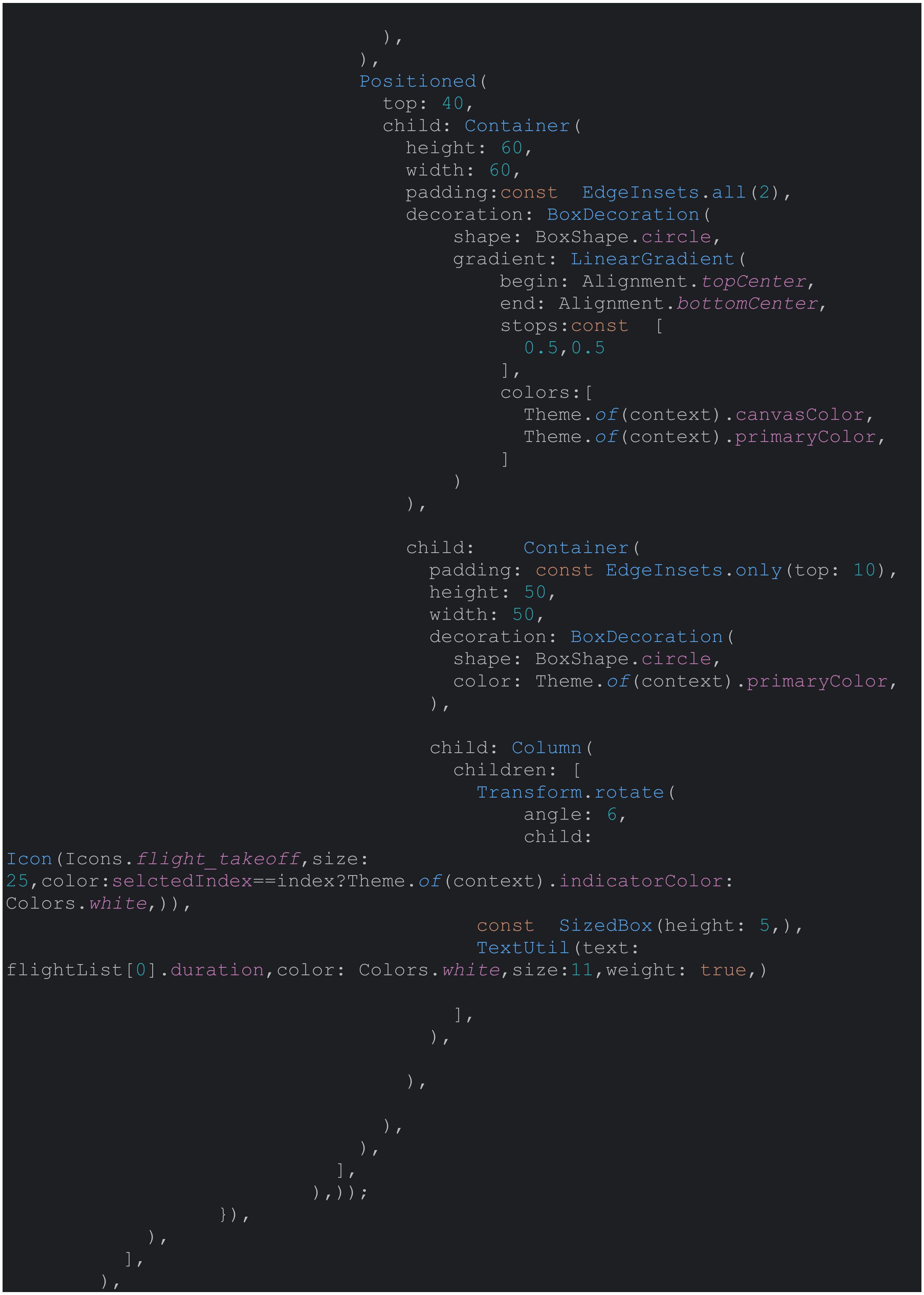
```
            ),

          ))
        ],
      ),


    );
  }
}
```



```
import 'package:date_picker_timeline/date_picker_widget.dart';
import 'package:flight_app_ui/widgets/text.dart';
import 'package:flutter/material.dart';

import '../../data/flight_data.dart';

import '../../widgets/show_up_animation.dart';
class FlightScreen extends StatefulWidget {
  const FlightScreen({super.key});

  @override
  State<FlightScreen> createState() => _FlightScreenState();
}

class _FlightScreenState extends State<FlightScreen> {
  DateTime _selectedDate=DateTime.now();
 bool isLoad=true;
  int? selctedIndex;
  void update(){
    Future.delayed(const Duration(milliseconds: 100), () {
     setState(() {
       isLoad=true;
     });
    });
  }
  @override
  Widget build(BuildContext context) {
    return Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        _adddatebar(),
     isLoad?Expanded(child:   Padding(
        padding: const EdgeInsets.all(15),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            TextUtil(text: "5 Emirates flight available",color:
Colors.white,weight: true,size: 14,),
           const  SizedBox(height: 20,),
            Expanded(
              child: ListView.builder(
                  itemCount: flightList.length,
                  shrinkWrap: true,
                  itemBuilder: (context,index){
                    return ShowUpAnimation(
```

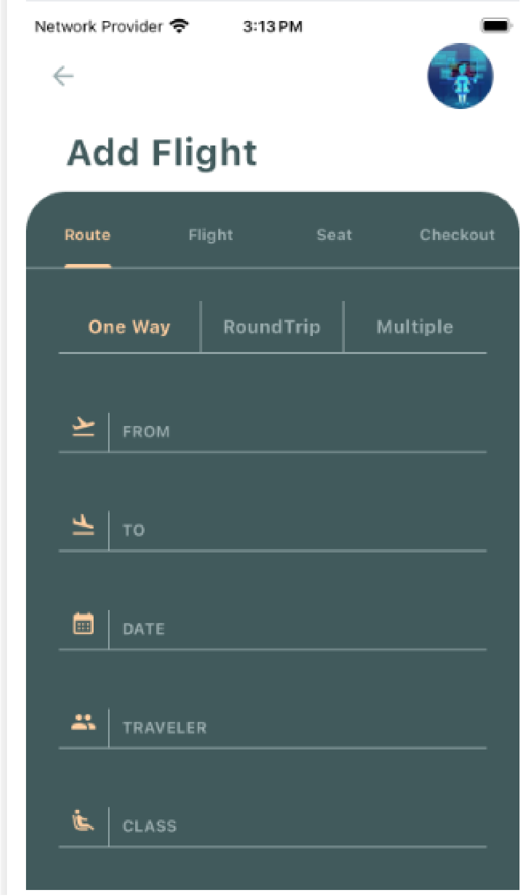


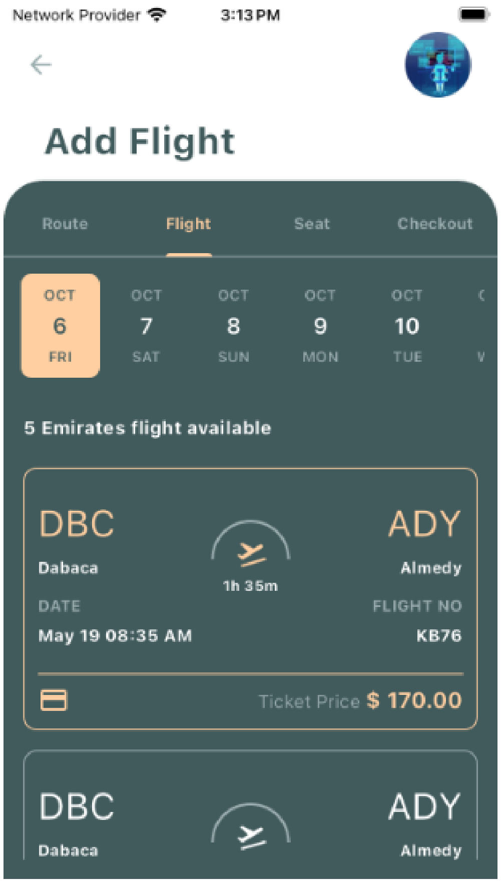
```
                        delay: 150,
                        child: GestureDetector(
                          onTap: (){
                           setState(() {
                             selctedIndex=index;
                           });
                          },
                          child: Stack(
                            alignment: Alignment.center,
                            children: [

                              Container(
                                padding: const EdgeInsets.all(10),
                                margin: const EdgeInsets.only(bottom: 15),
                                height: 200,
                                width: double.infinity,
                                decoration: BoxDecoration(
                                  borderRadius: BorderRadius.circular(10),
                                    border:
Border.all(color:selctedIndex==index?Theme.of(context).indicatorColor:
Theme.of(context).canvasColor)
                                ),
                                child: Column(
                                  children: [
                                    const Spacer(),
                                    Row(
                                      mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                                      children: [
                                        Column(
                                          crossAxisAlignment:
CrossAxisAlignment.start,
                                          children: [
                                            TextUtil(text:
flightList[0].source,color:selctedIndex==index?Theme.of(context).indicatorCol
or: Colors.white,size: 28,),
                                            const  SizedBox(height: 5,),
                                            TextUtil(text:
flightList[0].sourceName,color: Colors.white,size:12,weight: true,)
                                          ],
                                        ),
                                        Column(
                                          crossAxisAlignment:
CrossAxisAlignment.end,
                                          children: [
                                            TextUtil(text:
flightList[0].destination,color:selctedIndex==index?Theme.of(context).indicat
orColor:  Colors.white,size: 28,),
                                            const  SizedBox(height: 5,),
                                            TextUtil(text:
flightList[0].destinationName,color: Colors.white,size:12,weight: true,)
                                          ],
                                        )
                                      ],
                                    ),
                                    const Spacer(),

                                    Row(
                                      mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                                      children: [
                                        Column(
                                          crossAxisAlignment:
CrossAxisAlignment.start,
                                          children: [
                                            TextUtil(text: "DATE",color:
Theme.of(context).canvasColor,size: 12,weight: true,),
                                            const  SizedBox(height: 5,),
                                            TextUtil(text:
"${flightList[0].date} ${flightList[0].boardingTime}",color:
Colors.white,size:13,weight: true,)
                                          ],
                                        ),
                                        Column(
                                          crossAxisAlignment:
CrossAxisAlignment.end,
                                          children: [
                                            TextUtil(text: "FLIGHT NO",color:
Theme.of(context).canvasColor,size: 12,weight: true,),
                                            const  SizedBox(height: 5,),
                                            TextUtil(text:
flightList[0].flightNo,color: Colors.white,size:13,weight: true,)
                                          ],
                                        )
                                      ],
                                    ),
                                    const Spacer(),
                                    Divider(color:
selctedIndex==index?Theme.of(context).indicatorColor: Colors.white,),
                                    Row(
                                      mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                                      children: [

Icon(Icons.credit_card,color:selctedIndex==index?Theme.of(context).indicatorC
olor:  Colors.white,),
                                        RichText(
                                          text: TextSpan(
                                            text: 'Ticket Price ',
                                            style:  TextStyle(color:
Theme.of(context).canvasColor,fontSize: 14),
                                            children:  <TextSpan>[
                                              TextSpan(text: "\$
${flightList[0].price}", style: TextStyle(fontWeight:
FontWeight.bold,color:selctedIndex==index?Theme.of(context).indicatorColor:
Colors.white,fontSize: 17),

                                              ),

                                            ],
                                          ),
                                        ),

                                      ],
                                    )
                                  ],
```



```
                                ),
                              ),
                              Positioned(
                                top: 40,
                                child: Container(
                                  height: 60,
                                  width: 60,
                                  padding:const  EdgeInsets.all(2),
                                  decoration: BoxDecoration(
                                      shape: BoxShape.circle,
                                      gradient: LinearGradient(
                                          begin: Alignment.topCenter,
                                          end: Alignment.bottomCenter,
                                          stops:const   [
                                            0.5,0.5
                                          ],
                                          colors:[
                                            Theme.of(context).canvasColor,
                                            Theme.of(context).primaryColor,
                                          ]
                                      )
                                  ),

                                  child:    Container(
                                    padding: const EdgeInsets.only(top: 10),
                                    height: 50,
                                    width: 50,
                                    decoration: BoxDecoration(
                                      shape: BoxShape.circle,
                                      color: Theme.of(context).primaryColor,
                                    ),

                                    child: Column(
                                      children: [
                                        Transform.rotate(
                                            angle: 6,
                                            child:
Icon(Icons.flight_takeoff,size:
25,color:selctedIndex==index?Theme.of(context).indicatorColor:
Colors.white,)),
                                        const  SizedBox(height: 5,),
                                        TextUtil(text:
flightList[0].duration,color: Colors.white,size:11,weight: true,)

                                      ],
                                    ),

                                  ),

                                ),
                              ),
                            ],
                          ),));
                  }),
            ),
          ],
        ),
      )):const  SizedBox(),

      ],
    );
  }

_adddatebar(){
  return  Container(
    margin:const  EdgeInsets.all(10),
    child: DatePicker(
      DateTime.now(),
      height: 85,
      width: 60,
      initialSelectedDate: DateTime.now(),
      selectedTextColor: Theme.of(context).primaryColor,
      selectionColor:Theme.of(context).indicatorColor,
      dateTextStyle:const  TextStyle(
          fontWeight: FontWeight.w600,
          color: Colors.white,
        fontSize: 17

      ),

      dayTextStyle: TextStyle(
          fontWeight: FontWeight.w600,
          color: Theme.of(context).canvasColor,
          fontSize: 11
      ),
      monthTextStyle: TextStyle(
          fontWeight: FontWeight.w600,

          color: Theme.of(context).canvasColor,
          fontSize: 11
      ),


      onDateChange: (date){
        setState(() {
          _selectedDate=date;
          isLoad=false;
          selctedIndex=null;
          update();
        });

      },

    ),
  );
}
}
```

App UI:


Network Provider
3:13 PM
Add Flight
Route
Flight
Seat
Checkout
One Way
RoundTrip
Multiple
FROM
TO
DATE
TRAVELER
CLASS



Add Flight

OCT	OCT	OCT	OCT	OCT
6	7	8	9	10
FRI	SAT	SUN	MON	TUE

5 Emirates flight available

DBC
Dabaca

1h 35m

ADY
Almedy

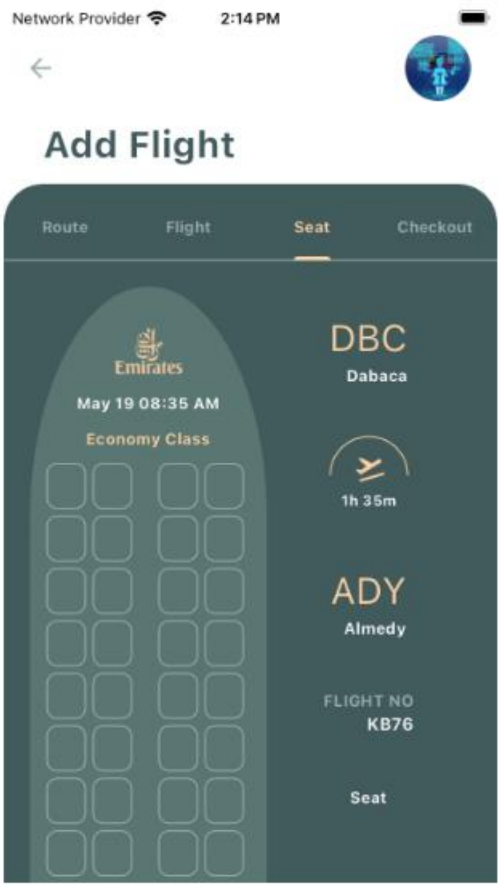
DATE
May 19 08:35 AM

FLIGHT NO
KB76

Ticket Price $ 170.00

DBC
Dabaca

ADY
Almedy



Conclusion: Therefore understood navigation, routing, gesture detection and gesture handling in Flutter and implemented the same in my Flutter application to route different pages.