MAD and PWA Lab 5

Name: Bhavesh Wadhwani
Class: D15A
Roll no:66

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

THEORY:
In a Flutter app, implementing navigation, routing, and gestures effectively is crucial for creating a seamless user experience. Let's delve into a theory on how to apply these concepts in your Flutter app:

A) Navigation and Routing:

a. Navigation:
Navigation in Flutter refers to moving between different screens or "routes" within your app. It's essential to organize your app's navigation in a clear and intuitive manner.

b. Routing:
Routing involves defining routes and managing transitions between them. Flutter provides a built-in routing system that allows you to define routes using a Route object. Common types of routes include MaterialPageRoute, CupertinoPageRoute, and Custom routes.

Application:

1. Route Management:
Define routes for each screen in your app using MaterialApp or CupertinoApp's "routes" parameter. This maps route names to builder functions.

```
routes: {
'/home': (context) => HomeScreen(),
'/profile': (context) => ProfileScreen(),
// Add more routes as needed
}
```

2. Navigation:
Use Navigator.push() to navigate to a new route when a user interacts with UI elements such as buttons or gestures.

```
onPressed: () {
Navigator.pushNamed(context, '/profile');
}
```

3.Navigation with Data: Pass data between routes using arguments.

```
Navigator.pushNamed(context, '/profile', arguments: userData);
```

4. Back Navigation:
Allow users to navigate back using Navigator.pop().

```
onPressed: () {
Navigator.pop(context);
}
```

5. Named Routes vs. Route Builders:
Decide whether to use named routes or route builders based on your app's complexity and navigation requirements. Named routes are simpler, while route

builders offer more customization.

C) Gestures:

a. Gestures:

Gestures enable users to interact with your app through touch-based actions like tapping, swiping, dragging, etc. Flutter provides GestureDetector and various gesture recognizers to handle these interactions.

b. Theory/Application:

1. GestureDetector:

Wrap your UI elements with GestureDetector to detect gestures.GestureDetector(

```
onTap: () {
// Handle tap gesture
},
onDoubleTap: () {
// Handle double tap gesture
},
// Add more gesture callbacks as needed
child: YourWidget(),
)
```

CODE:

Main.dart:

```
import 'package:flutter/material.dart';
import 'package:flutter_localizations/flutter_localizations.dart';
import 'package:flutter_ecommerce_ui_kit/auth/auth.dart';
import 'package:flutter_ecommerce_ui_kit/blocks/auth_block.dart';
import 'package:flutter_ecommerce_ui_kit/cart.dart';
import 'package:flutter_ecommerce_ui_kit/categorise.dart';
import 'package:flutter_ecommerce_ui_kit/home/home.dart';
import 'package:flutter_ecommerce_ui_kit/localizations.dart';
import 'package:flutter_ecommerce_ui_kit/product_detail.dart';
import 'package:flutter_ecommerce_ui_kit/settings.dart';
import 'package:flutter_ecommerce_ui_kit/shop/shop.dart';
import 'package:flutter_ecommerce_ui_kit/wishlist.dart';
import 'package:provider/provider.dart';
import 'package:firebase_core/firebase_core.dart';



void main() async{
 WidgetsFlutterBinding.ensureInitialized();
 await Firebase.initializeApp(
   // Replace with actual values
   options: FirebaseOptions(
     apiKey: "AIzaSyClVw9QJlN2fi3_7thVm7HdmA__SWTYMtw",
     appId: "1:560299755679:android:7701dcc4e602a6bc493aaa",
     messagingSenderId: "560299755679",
     projectId: "bhaveshs-826b8",
   ),
 );

 final Locale locale = Locale('en');
 runApp(MultiProvider(
   providers: [ChangeNotifierProvider<AuthBlock>.value(value: AuthBlock())],
   child: MaterialApp(
     localizationsDelegates: [
      AppLocalizations.delegate,
      GlobalMaterialLocalizations.delegate,
      GlobalWidgetsLocalizations.delegate,
      GlobalCupertinoLocalizations.delegate,
    ],
    supportedLocales: [Locale('en'), Locale('ar')],
    locale: locale,
    debugShowCheckedModeBanner: false,
    theme: ThemeData(
      primaryColor: Colors.red,
      colorScheme: ColorScheme.fromSwatch()
          .copyWith(secondary: Colors.black),
      fontFamily: locale.languageCode == 'ar' ? 'Dubai' : 'Lato',
    ),
    initialRoute: '/',
    routes: <String, WidgetBuilder>{
      '/': (BuildContext context) => Home(),
      '/auth': (BuildContext context) => Auth(),
      '/shop': (BuildContext context) => Shop(),
      '/categorise': (BuildContext context) => Categorise(),
      '/wishlist': (BuildContext context) => WishList(),
      '/cart': (BuildContext context) => CartList(),
      '/settings': (BuildContext context) => Settings(),
      '/products': (BuildContext context) => Products()
    },
  ),
 ));
}
```

Shop.dart:

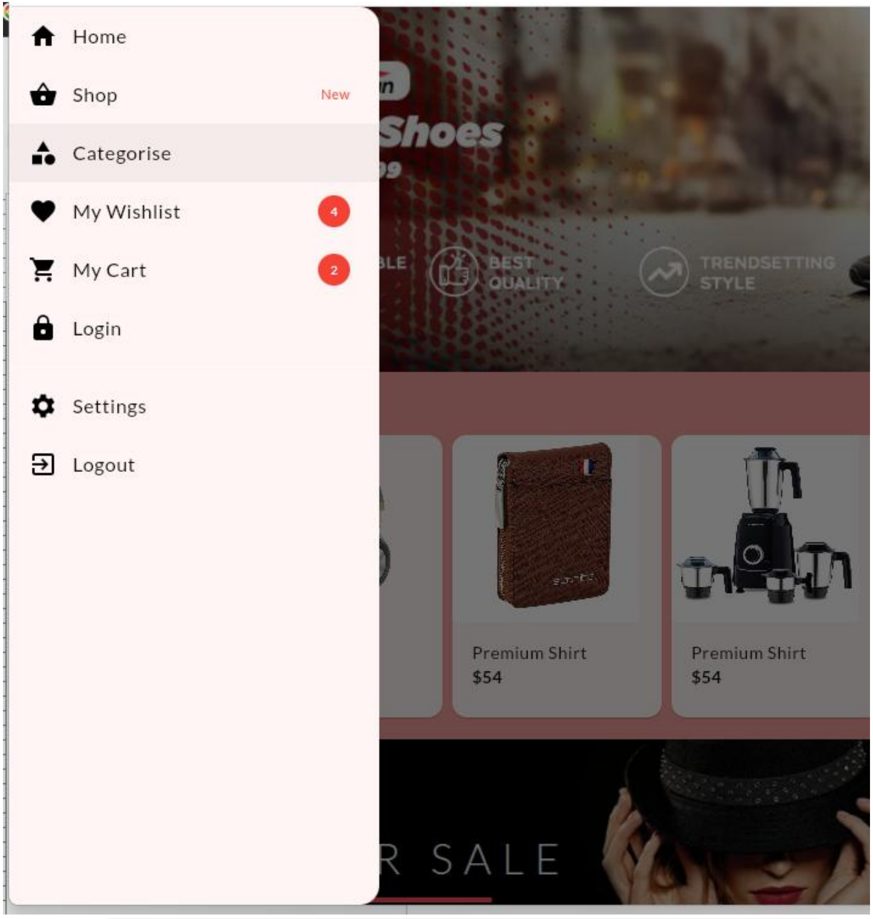
```
import 'package:flutter/material.dart';
import 'package:cached_network_image/cached_network_image.dart';
import 'package:flutter_rating_stars/flutter_rating_stars.dart';

import 'search.dart';

class Shop extends StatefulWidget {
 @override
 _ShopState createState() => _ShopState();
}

class _ShopState extends State<Shop> {
 final List<Map<dynamic, dynamic>> products = [
   {'name': 'Wallet', 'rating': 3.0, 'image': 'https://m.media-
amazon.com/images/I/71ZKFRCkdkL._SL1500_.jpg'},
   {'name': 'Mixer', 'rating': 3.0, 'image': 'https://m.media-
amazon.com/images/I/61isgOoxj8L._SL1500_.jpg'},
   {'name': 'IPhone 11', 'rating': 4.0, 'image':
'https://rukminim2.flixcart.com/image/416/416/k2jbyq80pkrrdj/mobile-
refurbished/x/j/s/iphone-11-128-d-mwm02hn-a-apple-0-original-
imafkg242ugz8hwc.jpeg?q=70&crop=false'},
   {'name': 'IPhone 11', 'rating': 4.0, 'image': 'https://images.unsplash.com/photo-
1519125323398-675f0ddb6308?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=94a1e718d89ca60a6337a6008341ca50&auto=format&fi
t=crop&w=1950&q=80'},
    {'name': 'IPhone 11', 'rating': 4.0, 'image': 'https://images.unsplash.com/photo-
1519125323398-675f0ddb6308?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=94a1e718d89ca60a6337a6008341ca50&auto=format&fi
t=crop&w=1950&q=80'},
    {'name': 'IPhone 11', 'rating': 4.0, 'image': 'https://images.unsplash.com/photo-
1519125323398-675f0ddb6308?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=94a1e718d89ca60a6337a6008341ca50&auto=format&fi
t=crop&w=1950&q=80'},
    {'name': 'IPhone 11', 'rating': 4.0, 'image': 'https://images.unsplash.com/photo-
1519125323398-675f0ddb6308?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=94a1e718d89ca60a6337a6008341ca50&auto=format&fi
t=crop&w=1950&q=80'},
    {'name': 'IPhone 11', 'rating': 4.0, 'image': 'https://images.unsplash.com/photo-
1519125323398-675f0ddb6308?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=94a1e718d89ca60a6337a6008341ca50&auto=format&fi
t=crop&w=1950&q=80'},
    {'name': 'IPhone 11', 'rating': 4.0, 'image': 'https://images.unsplash.com/photo-
1519125323398-675f0ddb6308?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=94a1e718d89ca60a6337a6008341ca50&auto=format&fi
t=crop&w=1950&q=80'},
    {'name': 'IPhone 12', 'rating': 5.0, 'image': 'https://images.unsplash.com/photo-
1523205771623-e0faa4d2813d?ixlib=rb-
0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=89719a0d55dd05e2deae4120227e6efc&auto=format&fi
t=crop&w=1953&q=80'},
  ];
  final GlobalKey<ScaffoldState> scaffoldKey = GlobalKey<ScaffoldState>();
  @override
  Widget build(BuildContext context) {
    return DefaultTabController(
      length: 2,
      child: Scaffold(
        key: scaffoldKey,
        appBar: AppBar(
          actions: <Widget>[
            IconButton(
              icon: Icon(Icons.search, color: Colors.white),
              onPressed: () {
                scaffoldKey.currentState!
                    .showBottomSheet((context) => ShopSearch());
              },
            )
          ],
          title: Text('Shop'),
        ),
        body: Builder(
          builder: (BuildContext context) {
            return DefaultTabController(
                length: 2,
                child: Column(
                  children: <Widget>[
                    Container(
                      constraints: BoxConstraints(maxHeight: 150.0),
                      child: Material(
                        color: Theme.of(context).colorScheme.secondary,
    child: TabBar(
      indicatorColor: Colors.red,
      tabs: [
        Tab(icon: Icon(Icons.view_list)),
        Tab(icon: Icon(Icons.grid_on)),
      ],
    ),
  ),
),
Expanded(
  child: TabBarView(
    children: [
      Container(
        child: ListView(
          children: products.map((product) {
            return Builder(
              builder: (BuildContext context) {
                return  InkWell(
                  onTap: () {
                    print('Card tapped.');
                  },
                  child: Column(
                    crossAxisAlignment: CrossAxisAlignment.start,
                    children: <Widget>[
                      Divider(
                        height: 0,
                      ),
                      Padding(
                        padding: const EdgeInsets.only(top: 10.0, bottom: 10.0),
                        child: ListTile(
                          trailing: Icon(Icons.navigate_next),
                          leading: ClipRRect(
                            borderRadius: BorderRadius.circular(5.0),
                            child: Container(
                              decoration: BoxDecoration(
                                  color: Colors.red
                              ),
                              child: CachedNetworkImage(
                                fit: BoxFit.cover,
                                imageUrl: product['image'],
                                placeholder: (context, url) => Center(
                                    child: CircularProgressIndicator()
                                ),
                                errorWidget: (context, url, error) => new Icon(Icons.error),
                              ),
                            ),
                          ),
                          title: Text(
                            product['name'],
                            style: TextStyle(
                                fontSize: 14
                            ),
                          ),
                          subtitle: Column(
                      crossAxisAlignment: CrossAxisAlignment.start,
                      children: <Widget>[
                        Row(
                          children: <Widget>[
                            Padding(
                              padding: const EdgeInsets.only(top: 2.0, bottom: 1),
                              child: Text('\$200', style: TextStyle(
                                color: Theme.of(context).colorScheme.secondary,
                                fontWeight: FontWeight.w700,
                              )),
                            ),
                            Padding(
                              padding: const EdgeInsets.only(left: 6.0),
                              child: Text('(\$400)', style: TextStyle(
                                  fontWeight: FontWeight.w700,
                                  fontStyle: FontStyle.italic,
                                  color: Colors.grey,
                                  decoration: TextDecoration.lineThrough
                              )),
                            )
                          ],
                        ),
                        Row(
                          children: <Widget>[
                            RatingStars(
                              value: product['rating'],
                              starSize: 16,
                              valueLabelColor: Colors.amber,
                              starColor: Colors.amber,
                            )
                          ],
                        )
                      ],
                    ),
                  ),
                ),
              ],
            ),
          );
        },
      );
  }).toList(),
  ),
),
Container(
  child: GridView.count(
    shrinkWrap: true,
    crossAxisCount: 2,
    childAspectRatio: 0.7,
    padding: EdgeInsets.only(top: 8, left: 6, right: 6, bottom: 12),
    children: List.generate(products.length, (index) {
      return Container(
        child: Card(
          clipBehavior: Clip.antiAlias,
child: InkWell(
 onTap: () {
  print('Card tapped.');
 },
 child: Column(
  crossAxisAlignment: CrossAxisAlignment.start,
  children: <Widget>[
   SizedBox(
    height: (MediaQuery.of(context).size.width / 2 - 5),
    width: double.infinity,
    child: CachedNetworkImage(
     fit: BoxFit.cover,
     imageUrl: products[index]['image'],
     placeholder: (context, url) => Center(
       child: CircularProgressIndicator()
     ),
     errorWidget: (context, url, error) => new Icon(Icons.error),
    ),
   ),
   Padding(
    padding: const EdgeInsets.only(top: 5.0),
    child: ListTile(
     title: Text(
      'Two Gold Rings',
      style: TextStyle(
        fontWeight: FontWeight.bold,
        fontSize: 16
      ),
     ),
     subtitle: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: <Widget>[
       Row(
        children: <Widget>[
         Padding(
          padding: const EdgeInsets.only(top: 2.0, bottom: 1),
          child: Text('\$200', style: TextStyle(
           color: Theme.of(context).colorScheme.secondary,
           fontWeight: FontWeight.w700,
          )),
         ),
         Padding(
          padding: const EdgeInsets.only(left: 6.0),
          child: Text('(\$400)', style: TextStyle(
            fontWeight: FontWeight.w700,
            fontStyle: FontStyle.italic,
            color: Colors.grey,
            decoration: TextDecoration.lineThrough
          )),
         )
        ],
       ),
       Row(
        children: <Widget>[
                                                     RatingStars(
                                                       value: products[index]['rating'],
                                                       starSize: 16,
                                                       valueLabelColor: Colors.amber,
                                                       starColor: Colors.amber,
                                                     )
                                                   ],
                                                 )
                                               ],
                                             ),
                                           ),
                                         )
                                       ],
                                     ),
                                   ),
                                 ),
                               );
                             }),
                           ),
                         ),
                       ],
                     ),
                   ),
                 ],
               ));
         },
       ),
     ),
   );
 }
}
```

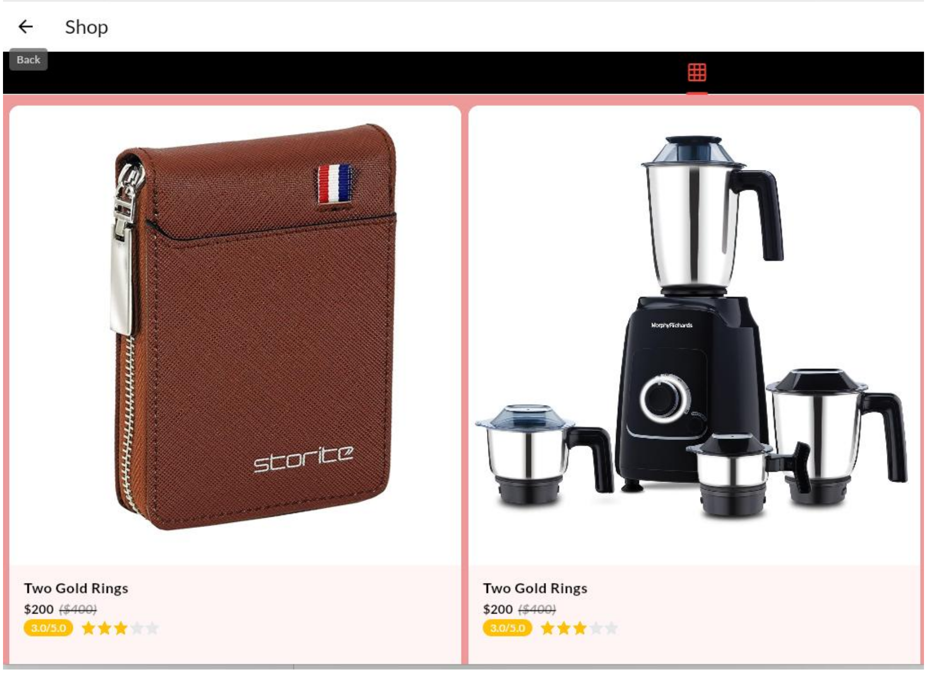
Signin.dart:

```
import 'dart:collection';
import 'dart:html';

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import 'package:flutter_ecommerce_ui_kit/blocks/auth_block.dart';
import 'package:flutter_ecommerce_ui_kit/models/user1.dart';
import 'package:provider/provider.dart';


class SignIn extends StatefulWidget {
 @override
 _SignInState createState() => _SignInState();
 static Future<List<dynamic>> _getEventsFromFirestore() async {
   var collection = FirebaseFirestore.instance.collection('users');
   var querySnapshot = await collection.get();
   var email;
   var password;
   for (var queryDocumentSnapshot in querySnapshot.docs) {
     Map<String, dynamic> data = queryDocumentSnapshot.data();
      email = data['email'];
      password = data['password'];
   }
   return [email, password];
 }
}

class _SignInState extends State<SignIn> {
 final _formKey = GlobalKey<FormState>();
 final UserCredential userCredential =
     UserCredential(usernameOrEmail: '', password: '');

 @override
 Widget build(BuildContext context) {
   return Center(
     child: Form(
       key: _formKey,
       child: SingleChildScrollView(
         child: Padding(
           padding: const EdgeInsets.only(left: 15.0, right: 15.0),
           child: Column(
             mainAxisAlignment: MainAxisAlignment.center,
             children: <Widget>[
               Padding(
                 padding: const EdgeInsets.only(bottom: 12.0),
                 child: TextFormField(
                   keyboardType: TextInputType.emailAddress,
                   validator: (value) {
                     if (value!.isEmpty) {
                       return 'Please Enter Email or Username';
                     }
                     return null;
                   },
                   onSaved: (value) {
                     setState(() {
                       userCredential.usernameOrEmail = value!;
                     });
                   },
                   decoration: InputDecoration(
                     hintText: 'Enter Username Or Email',
                     labelText: 'Email',
                   ),
                 ),
               ),
               TextFormField(
                 validator: (value) {
                   if (value!.isEmpty) {
                     return 'Please Enter Password';
                   }
                   return null;
                 },
                 onSaved: (value) {
                   setState(() {
                     userCredential.password = value!;
                   });
                 },
                decoration: InputDecoration(
                  hintText: 'Enter Password',
                  labelText: 'Password',
                ),
                obscureText: true,
              ),
              Padding(
                padding: const EdgeInsets.only(top: 25.0),
                child: SizedBox(
                  width: double.infinity,
                  height: 50,
                  child: Consumer<AuthBlock>(
                    builder: (BuildContext context, AuthBlock auth,
                        Widget? child) {
                      return ElevatedButton(
                        style: ElevatedButton.styleFrom(
                          primary: Theme.of(context).primaryColor,
                        ),
                        child: auth.loading && auth.loadingType == 'login'
                            ? CircularProgressIndicator(
                                valueColor: AlwaysStoppedAnimation<Color>(
                                    Colors.white),
                              )
                            : Text(
                                'Sign In',
                                style: TextStyle(color: Colors.white),
                              ),
                        onPressed: () {
                          // Validate form
                          if (_formKey.currentState!.validate() &&
                              !auth.loading) {
                            // Update values
                            _formKey.currentState!.save();
                            // Hit Api
                            auth.login(userCredential);
                          }
                        },
                      );
                    },
                  ),
                ),
              )
            ],
          ),
        ),
      ),
    ),
  );
 }
}
```

OUTPUT:
Location Screen:
After clicking on ‘Shop’ button

It will navigate to this pageNavigation to Shop items screen:





CONCLUSION:

We have implemented navigation ,routing and gestures in the pages.