

Experiment 5

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

Theory: Flutter uses Navigator to manage a stack of routes (screens/pages). There are two main types:

- **Named Routing:** Predefine routes in MaterialApp.
- **Direct Routing:** Use Navigator.push() with widget constructors.

Common methods:

- Navigator.push() – Navigate to a new screen
- Navigator.pop() – Go back to the previous screen
- Navigator.pushNamed() – Navigate using a route name

Gesture Detection:

Flutter provides a GestureDetector widget to respond to user input like:

- Tap
- Double Tap
- Long Press
- Drag, etc.

These gestures help create interactive and responsive UI.

Code:

```
import 'package:flutter/material.dart';  
import 'package:monkey_app_demo/screens/changeAddressScreen.dart';  
  
import './screens/spashScreen.dart';
```

```
import './screens/landingScreen.dart';
import './screens/loginScreen.dart';
import './screens/signUpScreen.dart';
import './screens/forgetPwScreen.dart';
import './screens/sentOTPScreen.dart';
import './screens/newPwScreen.dart';
import './screens/introScreen.dart';
import './screens/homeScreen.dart';
import './screens/menuScreen.dart';
import './screens/moreScreen.dart';
import './screens/offerScreen.dart';
import './screens/profileScreen.dart';
import './screens/dessertScreen.dart';
import './screens/individualItem.dart';
import './screens/paymentScreen.dart';
import './screens/notificationScreen.dart';
import './screens/aboutScreen.dart';
import './screens/inboxScreen.dart';
import './screens/myOrderScreen.dart';
import './screens/checkoutScreen.dart';
import './const/colors.dart';
```

```
void main() {
  runApp(MyApp());
}
```

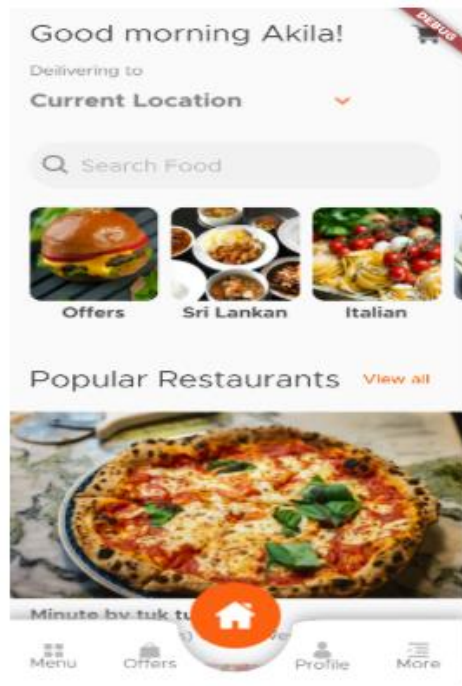
```
class MyApp extends StatelessWidget {
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        fontFamily: "Metropolis",
```

```
primarySwatch: Colors.red,
elevatedButtonTheme: ElevatedButtonThemeData(
  style: ButtonStyle(
    backgroundColor: MaterialStateProperty.all(
      AppColor.orange,
    ),
    shape: MaterialStateProperty.all(
      StadiumBorder(),
    ),
    elevation: MaterialStateProperty.all(0),
  ),
),
textButtonTheme: TextButtonThemeData(
  style: ButtonStyle(
    foregroundColor: MaterialStateProperty.all(
      AppColor.orange,
    ),
  ),
),
textTheme: TextTheme(
  headline3: TextStyle(
    color: AppColor.primary,
    fontSize: 16,
    fontWeight: FontWeight.bold,
  ),
  headline4: TextStyle(
    color: AppColor.secondary,
    fontWeight: FontWeight.bold,
    fontSize: 20,
  ),
  headline5: TextStyle(
    color: AppColor.primary,
    fontWeight: FontWeight.normal,
    fontSize: 25,
```

```
    ),  
    headline6: TextStyle(  
      color: AppColor.primary,  
      fontSize: 25,  
    ),  
    bodyText2: TextStyle(  
      color: AppColor.secondary,  
    ),  
  ),  
),  
),  
),  
//routing  
home: SplashScreen(),  
routes: {  
  LandingScreen.routeName: (context) => LandingScreen(),  
  LoginScreen.routeName: (context) => LoginScreen(),  
  SignUpScreen.routeName: (context) => SignUpScreen(),  
  ForgetPwScreen.routeName: (context) => ForgetPwScreen(),  
  SendOTPScreen.routeName: (context) => SendOTPScreen(),  
  NewPwScreen.routeName: (context) => NewPwScreen(),  
  IntroScreen.routeName: (context) => IntroScreen(),  
  HomeScreen.routeName: (context) => HomeScreen(),  
  MenuScreen.routeName: (context) => MenuScreen(),  
  OfferScreen.routeName: (context) => OfferScreen(),  
  ProfileScreen.routeName: (context) => ProfileScreen(),  
  MoreScreen.routeName: (context) => MoreScreen(),  
  DessertScreen.routeName: (context) => DessertScreen(),  
  IndividualItem.routeName: (context) => IndividualItem(),  
  PaymentScreen.routeName: (context) => PaymentScreen(),  
  NotificationScreen.routeName: (context) => NotificationScreen(),  
  AboutScreen.routeName: (context) => AboutScreen(),  
  InboxScreen.routeName: (context) => InboxScreen(),  
  MyOrderScreen.routeName: (context) => MyOrderScreen(),  
  CheckoutScreen.routeName: (context) => CheckoutScreen(),  
  ChangeAddressScreen.routeName: (context) => ChangeAddressScreen(),  
}
```

```
    },  
    );  
  }  
}
```

Screenshots:



Conclusion: In this lab, we implemented navigation using both named routes and `Navigator.push`, enabling users to navigate between screens by tapping on widgets. This enhances user experience and interactivity within the Meal Moneky app.