Experiment 5

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

Theory: Navigation in Flutter allows movement between different screens using routes. There are two types of routing: named and unnamed. Named routes are managed centrally and make large apps more scalable. Unnamed routes are directly pushed using MaterialPageRoute. Gestures in Flutter are detected using the GestureDetector widget, which allows interaction like tapping, swiping, or long-pressing on widgets. These are key to creating interactive, user-friendly apps.

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

```
void main() {
  runApp(MyApp());
}
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Food Delivery App',
      debugShowCheckedModeBanner: false,
      theme: ThemeData.dark(),
      home: HomeScreen(),
    );
  }
}
class HomeScreen extends StatelessWidget {
  final List<Map<String, String>> categories = [
    {"name": "Fast Food", "image": "assets/icons/fast_food.png"},
```

```
  {"name": "Asian", "image": "assets/icons/asian.png"},
  {"name": "Mexican", "image": "assets/icons/mexican.png"},
  {"name": "Pizza", "image": "assets/icons/pizza.png"},
  {"name": "Chinese", "image": "assets/icons/chinese.png"},
 ];
@override
 Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Color(0xFF1E1E1E),
    appBar: AppBar(
      title: Text('Food Delivery'),
      backgroundColor: Colors.black,
    ),
    body: Column(
      children: [
        SizedBox(height: 10),
        SingleChildScrollView(
          scrollDirection: Axis.horizontal,
          padding: EdgeInsets.symmetric(horizontal: 12),
          child: Row(
            children: categories.map((category) {
              return GestureDetector(
                onTap: () {
                  Navigator.push(
                    context,
                    MaterialPageRoute(
```

```
                builder: (_) => FastFoodScreen(title: category["name"]!),
              ),
            );
          },
          child: Container(
            margin: EdgeInsets.symmetric(horizontal: 8),
            child: Column(
              children: [
                Image.asset(
                  category["image"]!,
                  width: 50,
                ),
                SizedBox(height: 5),
                Text(
                  category["name"]!,
                  style: TextStyle(color: Colors.white),
                ),
              ],
            ),
          ),
        );
      }).toList(),
    ),
  ),
  // Add other UI like national picks below
],
```

```
      ),
      bottomNavigationBar: BottomNavigationBar(
        backgroundColor: Colors.black,
        selectedItemColor: Colors.orange,
        unselectedItemColor: Colors.grey,
        items: [
          BottomNavigationBarItem(icon: Icon(Icons.home), label: "Home"),
          BottomNavigationBarItem(icon: Icon(Icons.local_offer), label: "Offers"),
          BottomNavigationBarItem(icon: Icon(Icons.receipt_long), label: "Orders"),
          BottomNavigationBarItem(icon: Icon(Icons.person), label: "Account"),
        ],
      ),
    );
  }
}

class FastFoodScreen extends StatelessWidget {
  final String title;
  FastFoodScreen({required this.title});

  final List<Map<String, String>> fastFoodItems = [
    {
      "name": "McDonald's",
      "image": "assets/images/mcdonalds.png",
      "desc": "4.0 ★ · 25-40 mins · \$1.99 delivery"
    },
```

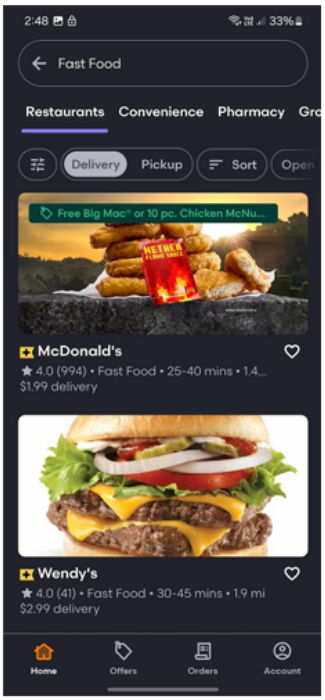
```
  {
    "name": "Wendy's",
    "image": "assets/images/wendys.png",
    "desc": "4.0 ★ · 30-45 mins · \$2.99 delivery"
  },
];

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Color(0xFF1E1E1E),
    appBar: AppBar(
      title: Text(title),
      backgroundColor: Colors.black,
    ),
    body: ListView.builder(
      itemCount: fastFoodItems.length,
      itemBuilder: (context, index) {
        var item = fastFoodItems[index];
        return Card(
          color: Colors.grey[900],
          margin: EdgeInsets.symmetric(horizontal: 12, vertical: 8),
          child: ListTile(
            leading: Image.asset(item["image"]!, width: 60),
            title: Text(item["name"]!, style: TextStyle(color: Colors.white)),
            subtitle: Text(item["desc"]!, style: TextStyle(color: Colors.grey)),
```

```
            trailing: Icon(Icons.favorite_border, color: Colors.white),
        ),
```

Output:

On pressing the fast food button



Conclusion: .This experiment showed how screens can be linked and how user interactions like taps or swipes can trigger events. These features improve the user experience and make the app feel more responsive and interactive.