

## Experiment No:- 05

Name:- Mayank Pahuja

Class:- D15-A RollNo:- 40

### Aim: To apply routing in Flutter Application

#### Theory:

In Flutter, routing refers to the navigation system that allows users to move between different screens or pages within an app. Flutter uses a widget-based approach for navigation, where each screen is represented by a widget. The 'Navigator' class manages the stack of routes and facilitates transitions between them.

Routes are typically defined using the 'MaterialPageRoute' class, providing a seamless and platform-aware transition between screens. Developers can use the 'Navigator' to push new routes onto the stack or pop existing routes off it. Named routes help in easily identifying and navigating to specific screens.

Flutter also supports route arguments, allowing developers to pass data between screens. Additionally, the 'Navigator' provides a flexible set of transitions, such as slide, fade, or custom animations, enhancing the user experience during navigation.

Overall, Flutter's routing system provides a structured and intuitive way to handle navigation within mobile and web applications.

#### Code & Implementation:

```
ignore_for_file: prefer_const_constructors,
prefer_const_literals_to_create_immutables
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:rapido/AllWidgets/Divider.dart';
class Homepage extends StatefulWidget {
  const Homepage({Key? key}) : super(key: key);
  @override
  State<Homepage> createState() =>
  HomePageState();
}
```

```
class HomePageState extends State<Homepage> {
  final User? user =
  FirebaseAuth.instance.currentUser;
  GlobalKey<ScaffoldState> scaffoldKey =
  GlobalKey<ScaffoldState>();
  Future<void> signOut() async {
    await FirebaseAuth.instance.signOut();
  }
  bool _isDragging = false;
```

```
Offset _offset = Offset(0, 0);
@override
Widget build(BuildContext context) {
  return Scaffold(
    key: scaffoldKey,
    appBar: AppBar(
      title: Text("Homepage"),
    ),
    drawer: Drawer(
      child: ListView(
        children: [
          GestureDetector(
            onTap: () {
              scaffoldKey.currentState?.openDrawer();
            },
            child: Container(
              height: 165.0,
              child: DrawerHeader(
                decoration: BoxDecoration(color:
                Colors.white),
                child: Row(
```





