Experiment no. 5

Name: Aniket MAhajan
Div: D15B
Roll no.: 33

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

Theory:
Navigation and Routing Navigation and routing are some of the core concepts of all mobile applications, which allows the user to move between different pages. We know that every mobile application contains several screens for displaying different types of information. For example, an app can have a screen that contains various products. When the user taps on that product, immediately it will display detailed information about that product.

In Flutter, the screens and pages are known as routes, and these routes are just a widget.

In Android, a route is similar to an Activity, whereas, in iOS, it is equivalent to a ViewController In any mobile app, navigating to different pages defines the workflow of the application, and the way to handle the navigation is known as routing. Flutter provides a basic routing class MaterialPageRoute and two methods Navigator.push() and Navigator.pop() that shows how to navigate between two routes. The following steps are required to start navigation in your application. Navigation: Navigation in Flutter refers to the ability to move between different screens or pages within an app. Flutter provides a Navigator widget to manage the navigation stack and perform common navigation operations. Navigation Operations:

Pushing a Screen: Use Navigator.push to navigate to a new screen.
Example:
```
Navigator.push(context, MaterialPageRoute(builder: (context) => DetailsScreen()));
```

Popping a Screen: Use Navigator.pop to go back to the previous screen.
Example:
```
Navigator.pop(context);
```

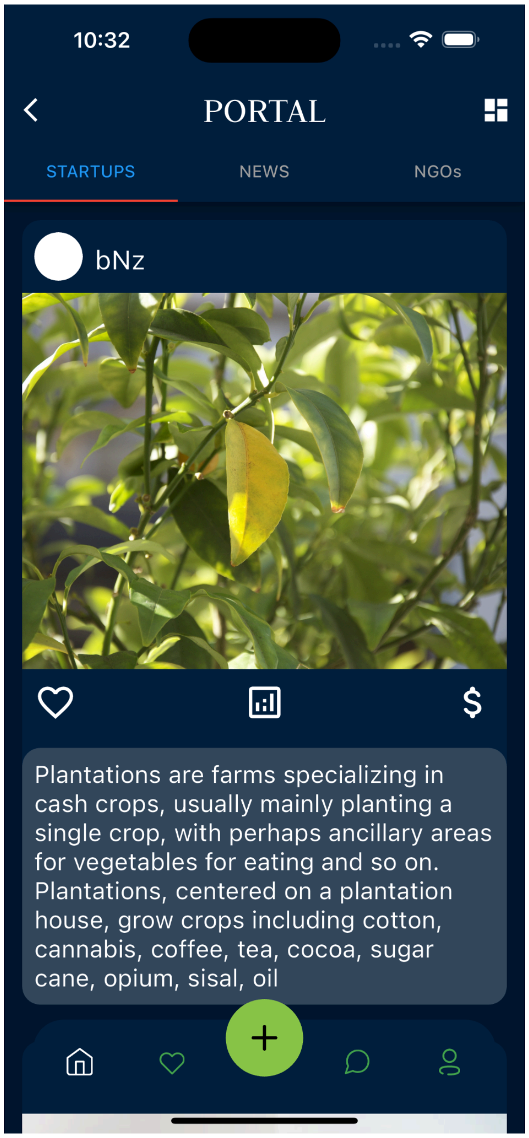
Routing:
In the context of Flutter, it involves defining the paths or routes that lead to different screens in your app. It allows you to organize and structure the flow of your application. Flutter supports both named routes and unnamed (or default) routes.

Named Routes:
Named routes are routes identified by a unique string identifier. They provide a more organized and maintainable way to navigate between screens. You can define named routes in the MaterialApp widget using the routes property.

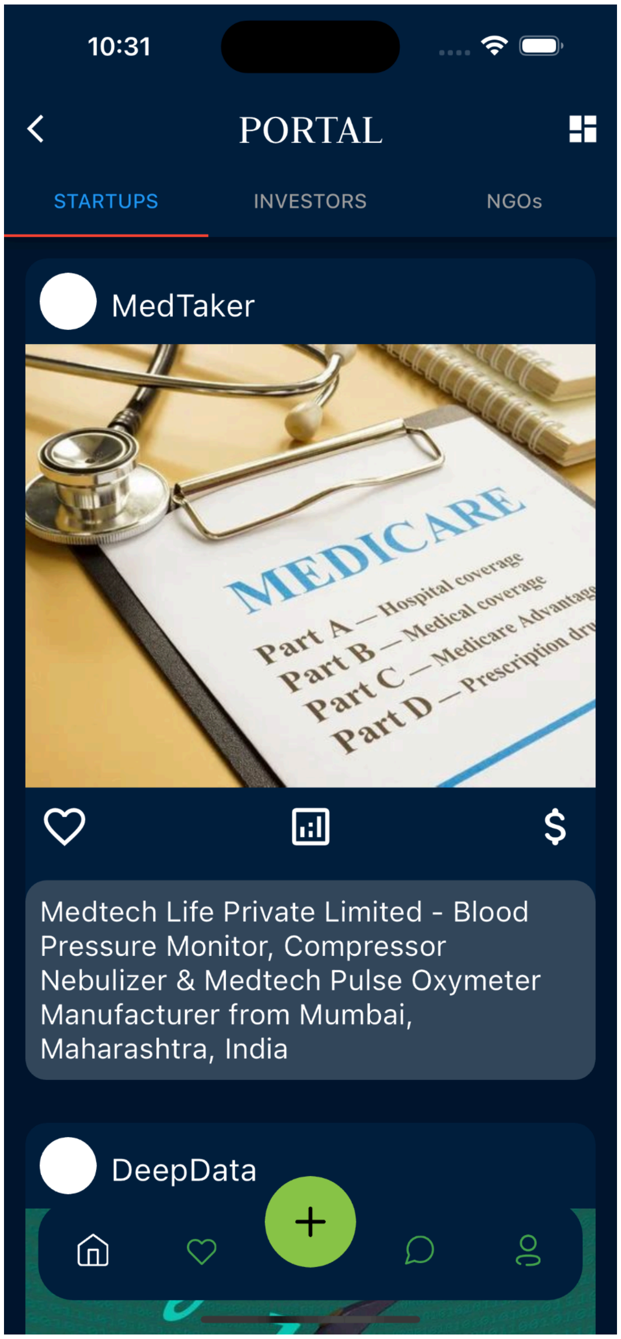
Example:

```dart
import 'package:flutter/material.dart';
import 'package:flutter_svg/svg.dart';
import 'package:portal_app/InvestorScreen/Investor_homepage.dart';
import 'package:portal_app/constants/appicons.dart';
import 'package:portal_app/screens/addpost.dart';

class InvestorNavigation extends StatefulWidget {
  const InvestorNavigation({super.key});

  @override
  State<InvestorNavigation> createState() => _MainPageState();
}

class _MainPageState extends State<InvestorNavigation> {
  Menus currentIndex = Menus.home;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      extendBody: true,
      body: pages[currentIndex.index],
      bottomNavigationBar: InvestorBottomNavBar(
        currentIndex: currentIndex,
        onTap: (value) {
          if (value == Menus.add) {
            AddPost();
          }
          setState(() {
            currentIndex = value;
          });
        },
      ),
    );
  }

  final pages = [
    InvestorHomePage(),
    Center(
      child: Text('Favorite'),
    ),
    AddPost(),
    Center(
      child: Text('Favorite'),
    ),
    Center(
      child: Text('Favorite'),
    ),
  ];
}

enum Menus {
  home,
  favorite,
  add,
  messages,
  user,
}

class InvestorBottomNavBar extends StatelessWidget {
  final Menus currentIndex;
  final ValueChanged<Menus> onTap;
  const InvestorBottomNavBar({
    super.key,
    required this.currentIndex,
    required this.onTap,
  });

  @override
  Widget build(BuildContext context) {
    return Container(
      height: 87,
      margin: EdgeInsets.all(24),
      child: Stack(
        children: [
          Positioned(
            right: 0,
            left: 0,
            top: 17,
            child: Container(
              height: 70,
              decoration: BoxDecoration(
                  color: Color(0xFF001F3F),
                  borderRadius: BorderRadius.all(Radius.circular(25))),
              child: Row(
                children: [
                  Expanded(
                    child: BottomNavigationItem(
                        onPressed: () => onTap(Menus.home),
                        icon: AppIcons.home,
                        current: currentIndex,
                        name: Menus.home),
                  ),
                  Expanded(
                    child: BottomNavigationItem(
                        onPressed: () => onTap(Menus.favorite),
                        icon: AppIcons.favorite,
                        current: currentIndex,
                        name: Menus.favorite),
                  ),
                  Spacer(),
                  Expanded(
                    child: BottomNavigationItem(
                        onPressed: () => onTap(Menus.messages),
                        icon: AppIcons.message,
                        current: currentIndex,
                        name: Menus.messages),
                  ),
                  Expanded(
                    child: BottomNavigationItem(
                        onPressed: () => onTap(Menus.user),
                        icon: AppIcons.user,
                        current: currentIndex,
                        name: Menus.user),
                  ),
                ],
              ),
            ),
          ),
          Positioned(
            left: 0,
            right: 0,
            top: 0,
            child: GestureDetector(
              onTap: () => onTap(Menus.add),
              child: Container(
                width: 64,
                height: 64,
                padding: const EdgeInsets.all(16),
                decoration: BoxDecoration(
                  color: Colors.lightGreen,
                  shape: BoxShape.circle,
                ),
                child: SvgPicture.asset(AppIcons.add),
              ),
            ),
          )
        ],
      ),
    );
  }
}

class BottomNavigationItem extends StatelessWidget {
  final VoidCallback onPressed;
  final String icon;
  final Menus current;
  final Menus name;
  const BottomNavigationItem(
      {super.key,
      required this.onPressed,
      required this.icon,
      required this.current,
      required this.name});

  @override
  Widget build(BuildContext context) {
    return IconButton(
        onPressed: onPressed,
        icon: SvgPicture.asset(
          icon,
          colorFilter: ColorFilter.mode(
              current == name ? Colors.white :
Colors.green.withOpacity(0.9),
              BlendMode.srcIn),
        ));
  }
}
```

OUTPUT-



Plantations are farms specializing in cash crops, usually mainly planting a single crop, with perhaps ancillary areas for vegetables for eating and so on. Plantations, centered on a plantation house, grow crops including cotton, cannabis, coffee, tea, cocoa, sugar cane, opium, sisal, oil



Medtech Life Private Limited - Blood Pressure Monitor, Compressor Nebulizer & Medtech Pulse Oxymeter Manufacturer from Mumbai, Maharashtra, India

Conclusion: Hence, we have successfully implemented the required functionalities for our app.