

EXP 5

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

Theory:

Navigation routing in Flutter refers to the process of defining and managing the routes or screens within your app. Flutter's navigation routing system allows you to move between different screens or pages in a structured and organized way. Here's an overview:

Route Definition: In Flutter, each screen or page in your app is typically represented by a widget. To define routes, you create a mapping between route names and the corresponding widget classes. This mapping is often done in the MaterialApp widget using the routes parameter.

Named Routes: Named routes are a common approach in Flutter for defining routes using unique names. Each route is associated with a name, making it easier to navigate to specific screens by referencing these names. For example, you can define routes like '/home', '/profile', '/settings', etc.

Navigation Stack: Flutter's navigation system maintains a stack of route objects. When you navigate to a new screen, the new route is pushed onto the stack. When you navigate back, the top route is popped off the stack, returning you to the previous screen.

Navigator: The Navigator class in Flutter is responsible for managing the navigation stack and transitioning between routes. You can use methods like Navigator.push() to navigate to a new route, Navigator.pop() to return to the previous route, and Navigator.pushNamed() to navigate to a named route.

MaterialPageRoute and CupertinoPageRoute: Flutter provides two types of page route classes - MaterialPageRoute for Android-style navigation and CupertinoPageRoute for iOS-style navigation. These classes define transitions and behaviors specific to each platform.

Route Parameters: You can pass parameters between routes in Flutter using the arguments parameter of the navigation methods (Navigator.pushNamed() or Navigator.push()). This allows you to send data to the next screen or receive data from the previous screen.

Modal Routes: Modal routes are routes that appear on top of other routes and typically block interaction with the underlying routes until they are dismissed. You can create modal routes using methods like `showDialog()` or by using `Navigator.push()` with a `PageRouteBuilder` and setting the `fullscreenDialog` parameter to `true`.

Code:

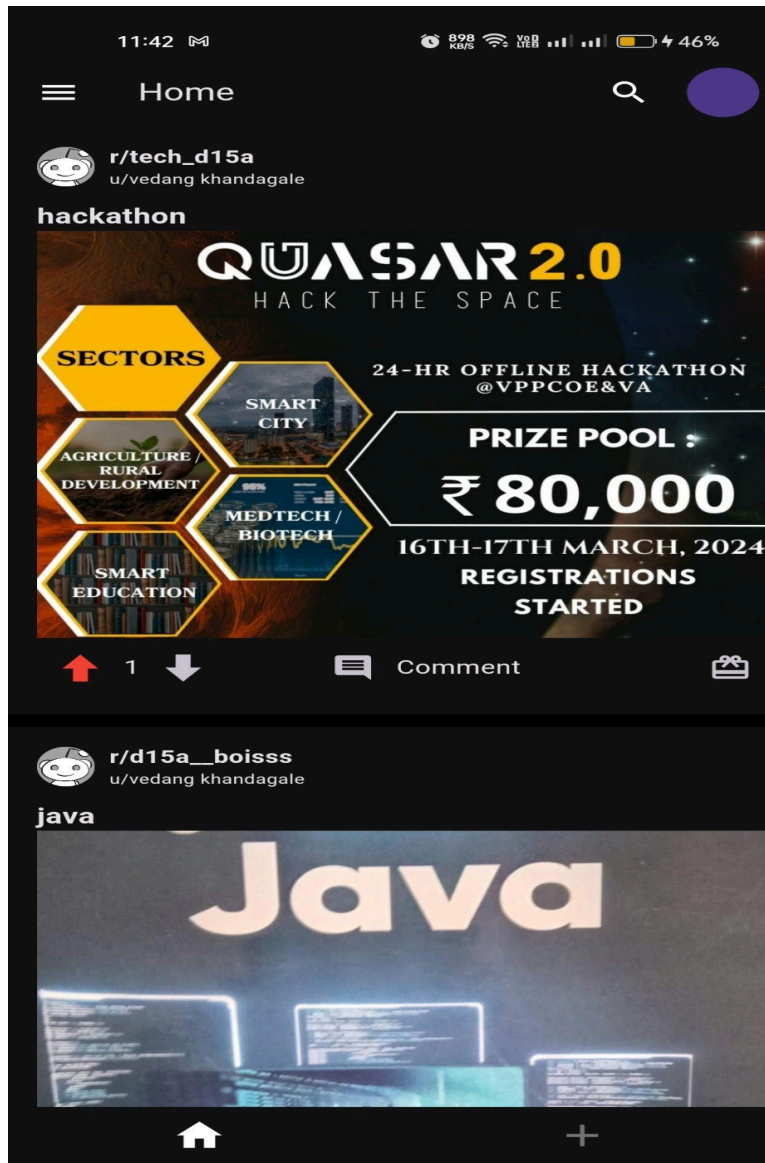
```
import 'package:flutter/material.dart';
import 'package:reddit_clone/features/auth/screens/login_screen.dart';
import 'package:reddit_clone/features/community/screens/add_mods_screen.dart';
import 'package:reddit_clone/features/community/screens/community_screen.dart';
import 'package:reddit_clone/features/community/screens/create_community_screen.dart';
import 'package:reddit_clone/features/community/screens/edit_community_screen.dart';
import 'package:reddit_clone/features/community/screens/mod_tools_screen.dart';
import 'package:reddit_clone/features/home/screens/home_screen.dart';
import 'package:reddit_clone/features/post/screens/add_post_type_screen.dart';
import 'package:reddit_clone/features/post/screens/comments_screen.dart';
import 'package:reddit_clone/features/user_profile/screens/edit_profile_screen.dart';
import 'package:reddit_clone/features/user_profile/screens/user_profile_screen.dart';
import 'package:routemaster/routemaster.dart';
```

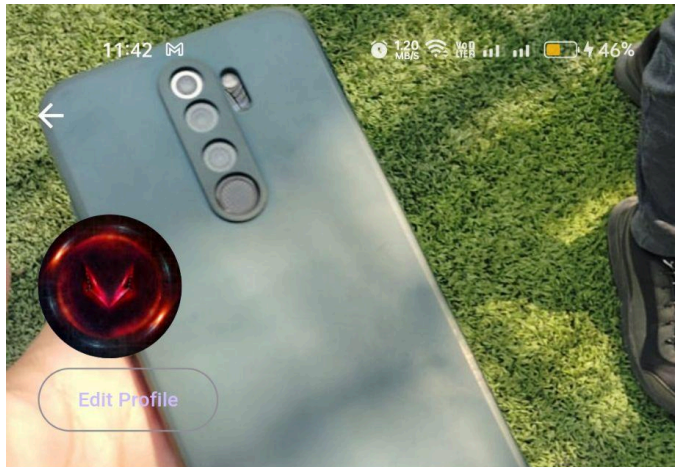
```
final loggedInRoute =
  RouteMap(routes: {'/': (_) => const MaterialPage(child: LoginScreen());});
```

```
final loggedInRoute = RouteMap(routes: {
  '/': (_) => const MaterialPage(child: HomeScreen()),
  '/create_community': (_) =>
    const MaterialPage(child: CreateCommunityScreen()),
  '/r/:name': (route) => MaterialPage(
    child: CommunityScreen(
      name: route.pathParameters['name']!,
    )),
  '/mod-tools/:name': (routeData) => MaterialPage(
    child: ModsToolsScreen(
      name: routeData.pathParameters['name']!,
    )),
  '/edit-community/:name': (routeData) => MaterialPage(
    child: EditCommunityScreen(
      name: routeData.pathParameters['name']!,
    )),
  '/add-mods/:name': (routeData) => MaterialPage(
    child: AddModsScreen(
      name: routeData.pathParameters['name']!,
    )),
});
```

```
'/u/:uid': (routeData) => MaterialPage(  
  child: UserProfileScreen(  
    uid: routeData.pathParameters['uid']!,  
  )),  
'/edit-profile/:uid': (routeData) => MaterialPage(  
  child: EditProfileScreen(  
    uid: routeData.pathParameters['uid']!,  
  )),  
'/add-post/:type': (routeData) => MaterialPage(  
  child: AddPostTypeScreen(  
    type: routeData.pathParameters['type']!,  
  )),  
'/post/:postId/comments': (route) => MaterialPage(  
  child: CommentsScreen(  
    postId: route.pathParameters['postId']!,  
  ),  
,  
// '/add-post': (routeData) => const MaterialPage(  
//   child: AddPostScreen(),  
//   ),  
});
```

Output:





u/vedang khandagale

16 karma



r/tech_d15a

u/vedang khandagale

hackathon

QUASAR 2.0
HACK THE SPACE

SECTORS

- SMART CITY
- AGRICULTURE RURAL DEVELOPMENT
- MEDTECH / BIOTECH
- SMART EDUCATION

24-HR OFFLINE HACKATHON
@VPPCOE&VA

PRIZE POOL :
₹ 80,000

16TH-17TH MARCH, 2024
REGISTRATIONS STARTED



1



Comment

