EXPERIMENT NO: - 05

Name:- Gunjan Chandnani Class:- D15A Roll:No: - 06

AIM: - To apply navigation, routing and gestures in Flutter App.

Theory: -

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. 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. Gestures enable the app to respond to user interactions, making the application more dynamic and responsive.

> Navigation and Routing in Flutter Navigation is the process of moving between different screens or pages in an app.

Flutter provides a simple and effective way to handle this through the use of the Navigator widget and routes.

- **1. Using Navigator Widget** The Navigator widget manages a stack of routes, allowing for pushing and popping routes on the stack.
- **Pushing a Route:** To navigate to a new screen, use Navigator.push().
- **Popping a Route:** To go back to the previous screen, use Navigator.pop().

ElevatedButton(onPressed: () { Navigator.push(Gunjan Chandnani– D15A / 06 context, MaterialPageRoute(builder: (context) => SecondScreen()),);},); 2. Named Routes Flutter also allows the use of named routes to navigate, which can make the routing process cleaner, especially in larger applications.

```
MaterialApp( initialRoute: '/', routes: { '/': (context) => HomeScreen(), '/second': (context) => SecondScreen(), }, );
```

Navigate to the route using Navigator.pushNamed() Navigator.pushNamed(context, '/second');

Handling Gestures in Flutter Gestures refer to user interactions with the app, such as taps, swipes, pinches, and drags.

Flutter provides several widgets and gesture detectors to handle these interactions. Tap Gestures The most common gesture is the tap, which can be handled using the GestureDetector widget or specific buttons like InkWell or ElevatedButton.

Long Press Gesture For long press gestures, Flutter provides the onLongPress callback in GestureDetector or InkWell.

Swipe and Drag Gestures Flutter also provides swipe and drag gesture handling. The onHorizontalDragUpdate and onVerticalDragUpdate callbacks are used for dragging gestures

2. Named Routes:

Flutter also allows the use of named routes to navigate, which can make the routing process cleaner, especially in larger applications.

MaterialApp(initialRoute: '/', routes: { '/': (context) => HomeScreen(), '/second': (context) => SecondScreen(), },);

Navigate to the route using Navigator.pushNamed() Navigator.pushNamed(context, '/second');

Handling Gestures in Flutter

Gestures refer to user interactions with the app, such as taps, swipes, pinches, and drags.

Flutter provides several widgets and gesture detectors to handle these interactions.

Tap Gestures

The most common gesture is the tap, which can be handled using the GestureDetector widget or specific buttons like InkWell or ElevatedButton.

Long Press Gesture For long press gestures, Flutter provides the onLongPress callback in GestureDetector or InkWell.

```
Code:
main.dart:
// ignore for file: unnecessary null comparison, prefer const constructors
import 'package:cloud firestore/cloud firestore.dart';
import 'package:firebase auth/firebase auth.dart';
import 'package:firebase core/firebase core.dart';
import 'package:flutter/material.dart';
import 'package:flutter riverpod/flutter riverpod.dart';
import 'package:youtube clone/cores/screens/loader.dart';
import 'package:youtube clone/features/auth/pages/login page.dart';
import 'package:youtube clone/features/auth/pages/username page.dart';
import
'package:youtube clone/features/upload/long video/video details page.dart';
import 'package:youtube_clone/firebase options.dart';
import 'package:youtube clone/home page.dart';
void main() async {
 WidgetsFlutterBinding.ensureInitialized();
 await Firebase.initializeApp(
  options: DefaultFirebaseOptions.currentPlatform,
 );
 runApp(ProviderScope(child: MyApp()));
class MyApp extends ConsumerWidget {
 const MyApp({super.key});
 @override
 Widget build(BuildContext context, WidgetRef ref) {
  return MaterialApp(
   debugShowCheckedModeBanner: false,
   // we check wthether we are signed in or not
```

home: StreamBuilder(

builder: (context, snapshot) {
 if (!snapshot.hasData) {

stream: FirebaseAuth.instance.authStateChanges(),

```
return LoginPage();
 } else if (snapshot.connectionState == ConnectionState.waiting) {
   return Loader();
 return StreamBuilder(
   stream: FirebaseFirestore.instance
     .collection("users")
     .doc(FirebaseAuth.instance.currentUser!.uid)
     .snapshots(),
  builder: (context, snapshot) {
    final user = FirebaseAuth.instance.currentUser;
    if (!snapshot.hasData | !snapshot.data!.exists) {
     return UsernamePage(
      displayName: user!.displayName!,
      profilePic: user.photoURL!,
      email: user.email!,
    } else if (snapshot.connectionState == ConnectionState.waiting) {
     return Loader();
    return HomePage();
  },
},
```

login.dart:

```
import 'package:flutter/material.dart';
import 'package:flutter_riverpod/flutter_riverpod.dart';
import 'package:youtube_clone/features/auth/repository/auth_service.dart';

class LoginPage extends ConsumerWidget {
  const LoginPage({super.key});

  @override

Widget build(BuildContext context, WidgetRef ref) {
    return Scaffold(
```

```
body: SafeArea(
    child: Center(
      child: Column (
        children: [
          Padding (
            padding: const EdgeInsets.only(
              top: 20,
              bottom: 25,
            child: Image.asset(
              "assets/images/youtube-signin.jpg",
              height: 150,
            ),
          ),
          const Text(
            "Welcome To Youtube",
            style: TextStyle(
              fontSize: 30,
              fontWeight: FontWeight.bold,
              color: Colors.blueGrey,
            ),
          ),
          const Spacer(),
          Padding(
            padding: const EdgeInsets.only(bottom: 55),
            child: GestureDetector(
              onTap: () async {
                await ref.read(authServiceProvider).signInWithGoogle();
              },
              child: Image.asset(
                "assets/images/signinwithgoogle.png",
                height: 60,
            ),
          ),
        ],
     ),
 ),
);
```

home_page.dart:

```
// ignore_for_file: invalid_use_of_visible_for_testing_member
import 'package:cached_network_image/cached_network_image.dart';
import 'package:flutter/material.dart';
import 'package:flutter_riverpod/flutter_riverpod.dart';
```

```
import 'package:path/path.dart';
import 'package:youtube clone/cores/screens/error page.dart';
import 'package:youtube clone/cores/screens/loader.dart';
import 'package:youtube clone/cores/widgets/image button.dart';
import 'package:youtube clone/features/account/account page.dart';
import 'package:youtube clone/features/auth/provider/user provider.dart';
import 'package:youtube clone/features/content/bottom navigation.dart';
import 'package:youtube clone/features/upload/upload bottom sheet.dart';
import 'package:youtube clone/pages list.dart';
class HomePage extends StatefulWidget {
const HomePage({super.key});
@override
State<HomePage> createState() => HomePageState();
class HomePageState extends State<HomePage> {
int currentIndex = 0;
@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: const Color(0xffFFFFFF),
    body: SafeArea(
       child: Column (
        children: [
          Row (
             children: [
               Image.asset(
                 "assets/images/youtube.jpg",
                 height: 36,
               const SizedBox(width: 4),
               const Spacer(),
               Padding (
                 padding: const EdgeInsets.only(right: 12),
                 child: SizedBox(
                   height: 42,
                   child: ImageButton(
                     image: "cast.png",
                    onPressed: () {},
                     haveColor: false,
                   ),
                 ),
               ),
               SizedBox(
                 height: 38,
                 child: ImageButton(
                   image: "notification.png",
```

```
onPressed: () {},
        haveColor: false,
      padding: const EdgeInsets.only(left: 12, right: 15),
      child: SizedBox(
        height: 41.5,
        child: ImageButton(
          image: "search.png",
          onPressed: () {},
          haveColor: false,
        ),
    Consumer (
      builder: (context, ref, child) {
        return ref.watch(currentUserProvider).when(
              data: (currentUser) => Padding(
                padding: const EdgeInsets.only(right: 12),
                child: GestureDetector(
                  onTap: () {
                    Navigator.push(
                      context,
                      MaterialPageRoute(
                        builder: (context) => AccountPage(
                          user: currentUser,
                        ),
                      ),
                    );
                  },
                  child: CircleAvatar(
                    radius: 14,
                    backgroundColor: Colors.grey,
                    backgroundImage: CachedNetworkImageProvider(
                      currentUser.profilePic,
                    ),
                  ),
                ),
              ),
              error: (error, stackTrace) => const ErrorPage(),
              loading: () => const Loader(),
            );
      },
Expanded (
  child: pages[currentIndex],
```

upload_bottom_sheet.dart:

```
// ignore for file: use build context synchronously
import 'package:flutter/material.dart';
import 'package:youtube clone/cores/methods.dart';
import 'package:youtube clone/cores/widgets/image item.dart';
class CreateBottomSheet extends StatelessWidget {
const CreateBottomSheet({super.key});
 @override
 Widget build(BuildContext context) {
  return Container(
    color: const Color(0xffFFFFFF),
    child: Padding(
      padding: const EdgeInsets.only(left: 7, top: 12),
      child: SizedBox(
         height: 270,
         child: Column (
           crossAxisAlignment: CrossAxisAlignment.start,
          children: [
             const Padding(
               padding: EdgeInsets.only(left: 10),
               child: Text(
                 "Create",
                 style: TextStyle(
                   fontSize: 20,
                   fontWeight: FontWeight.w500,
```

```
),
  const SizedBox(height: 28),
  SizedBox(
    height: 38,
    child: ImageItem(
      itemText: "Create a Short",
      itemClicked: () async {
        await pickShortVideo(context);
      imageName: "short-video.png",
      haveColor: true,
  ),
  const SizedBox(height: 12),
  SizedBox(
    height: 38,
    child: ImageItem(
      itemText: "Upload a Video",
      itemClicked: () async {
        await pickVideo(context);
      },
      imageName: "upload.png",
      haveColor: true,
    ),
  ),
  const SizedBox(height: 12),
  SizedBox(
    height: 38,
    child: ImageItem(
      itemText: "Go Live",
      itemClicked: () {},
      imageName: "go-live.png",
      haveColor: true,
  ),
  const SizedBox(height: 12),
  SizedBox(
    height: 38,
    child: ImageItem(
      itemText: "Create a post",
      itemClicked: () {},
      imageName: "create-post.png",
      haveColor: true,
],
```

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

OutPut:



Sign in

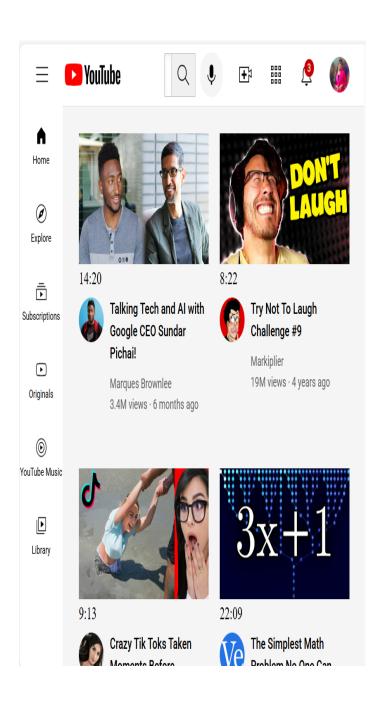
Use your Google Account

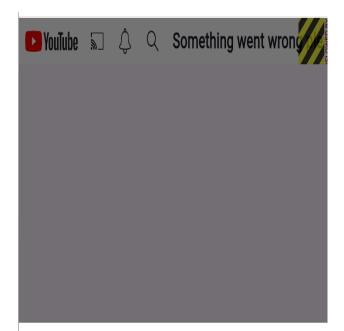
gunjanchandnani18@gmail.com

Forgot password?

Next

Create account





Create

- © Create a Short
- ↑ Upload a Video
- (**(•)**) Go Live
- Create a post