**Name-Aditya Kundu**

**Div-D15A**

**Roll No-31**

**EXPERIMENT NO.5**

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

**THEORY:-**

**In Flutter, navigation, routing, and gestures are crucial aspects for creating dynamic and interactive mobile applications. Let's explore the theory behind each of these concepts:**

**Navigation and Routing:**

**Navigation:**

**- Definition: Navigation refers to the movement between different screens or pages within a Flutter app.**

**- Navigator Widget: Flutter uses a `Navigator` widget to manage navigation.**

**- Push and Pop: Pages are pushed onto the navigator stack when navigating forward, and popped off the stack when navigating backward.**

**- Named Routes: Named routes are commonly used for navigation, allowing you to define routes with associated screen widgets.**

**Routing:**

**- Definition: Routing is the process of determining which screen or widget should be displayed in response to a specific user action or condition.**

**- MaterialPageRoute: The `MaterialPageRoute` class is commonly used for page-based navigation in material design apps.**

**- Named Routes: Named routes are defined in the `MaterialApp` widget using the `routes` parameter.**

**Example:**

**```dart**

**// Basic navigation using MaterialPageRoute**

**Navigator.push(**

**context,**

**MaterialPageRoute(builder: (context) => SecondScreen()),**

**);**

**// Named routes in MaterialApp**

**MaterialApp(**

**routes: {**

**'/second': (context) => SecondScreen(),**

**// add more named routes as needed**

**},**

**);**

**```**

**Gestures:**

**GestureDetector:**

**- Definition: GestureDetector is a widget that recognizes gestures such as taps, drags, and swipes.**

**- onTap, onDoubleTap, onLongPress: GestureDetector provides various callback properties to handle different gestures.**

**InkWell:**

**- Definition: InkWell is a material design widget that provides a visual response to touch.**

**- onTap, onDoubleTap, onLongPress: Similar to GestureDetector, InkWell has properties to handle different tap gestures.**

**Gesture Recognition:**

**- TapGestureDetector: Recognizes single taps.**

**- DoubleTapGestureDetector: Recognizes double taps.**

**- LongPressGestureDetector: Recognizes long presses.**

**- PanGestureDetector: Recognizes dragging or panning gestures.**

**- SwipeGestureDetector: Recognizes swipe gestures.**

**Example:**

**```dart**

**GestureDetector(**

**onTap: () {**

**// Handle tap gesture**

**},**

**onDoubleTap: () {**

**// Handle double-tap gesture**

**},**

**onLongPress: () {**

**// Handle long press gesture**

**},**

**);**

**// Using InkWell**

**InkWell(**

**onTap: () {**

**// Handle tap gesture**

**},**

**onDoubleTap: () {**

**// Handle double tap gesture**

**},**

**onLongPress: () {**

**// Handle long press gesture**

**},**

**);**

**```**

**In summary, effective navigation, routing, and gestures are essential for creating a smooth and engaging user experience in Flutter apps. Named routes and the `Navigator` widget facilitate navigation, while GestureDetector and InkWell enable gesture recognition for interactive elements within your app.**

**CODE:-**

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

**void main() {**

**runApp(MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**home: VideoCallScreen(),**

**);**

**}**

**}**

**class VideoCallScreen extends StatelessWidget {**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text('Microsoft Teams'),**

**),**

**body: Center(**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**Image.network(**

**'https://i.postimg.cc/W3xf1w5k/close-up-young-businessman-having-work-video-call-while-staying-home-new-normal-lifestyle-business-c.jpg', // Replace with your image URL**

**height: 200.0,**

**width: 200.0,**

**fit: BoxFit.cover,**

**),**

**SizedBox(height: 20.0),**

**Row(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**IconButton(**

**icon: Icon(Icons.videocam),**

**onPressed: () {**

**// Toggle video on/off**

**},**

**),**

**SizedBox(width: 20.0),**

**IconButton(**

**icon: Icon(Icons.mic),**

**onPressed: () {**

**// Toggle microphone on/off**

**},**

**),**

**],**

**),**

**SizedBox(height: 20.0),**

**GestureDetector(**

**onTap: () {**

**// Open a new screen or perform a specific action on tap**

**Navigator.push(context,**

**MaterialPageRoute(builder: (context) => SecondScreen()));**

**},**

**child: Text(**

**'Click Here to navigate to secondary screen',**

**style: TextStyle(**

**fontSize: 16.0,**

**color: Colors.blue,**

**fontWeight: FontWeight.bold,**

**),**

**),**

**),**

**],**

**),**

**),**

**floatingActionButton: FloatingActionButton(**

**onPressed: () {**

**// End the call**

**},**

**child: Icon(Icons.call\_end),**

**),**

**);**

**}**

**}**

**class SecondScreen extends StatelessWidget {**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text('Secondary Screen'),**

**),**

**body: Center(**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**Icon(**

**Icons.star,**

**size: 50.0,**

**color: Colors.yellow,**

**),**

**SizedBox(height: 20.0),**

**Text(**

**'Custom Font Example',**

**style: TextStyle(**

**fontSize: 20.0,**

**fontFamily: 'CustomFont', // Replace with your custom font**

**fontWeight: FontWeight.bold,**

**),**

**),**

**],**

**),**

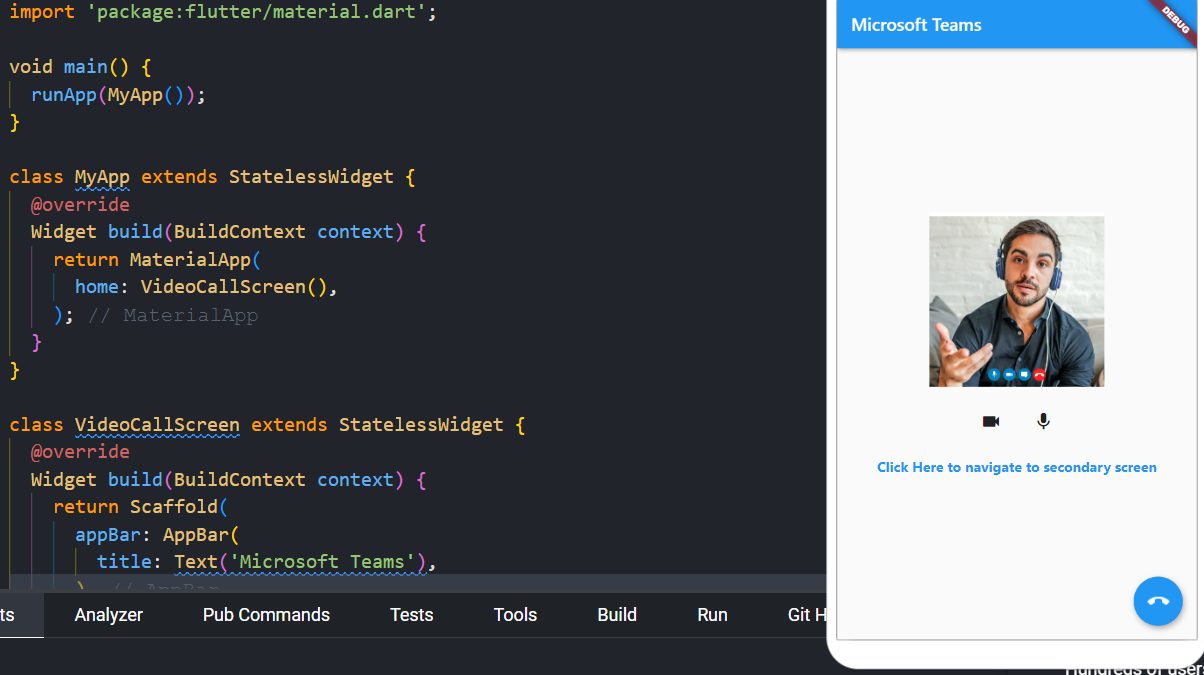
**),**

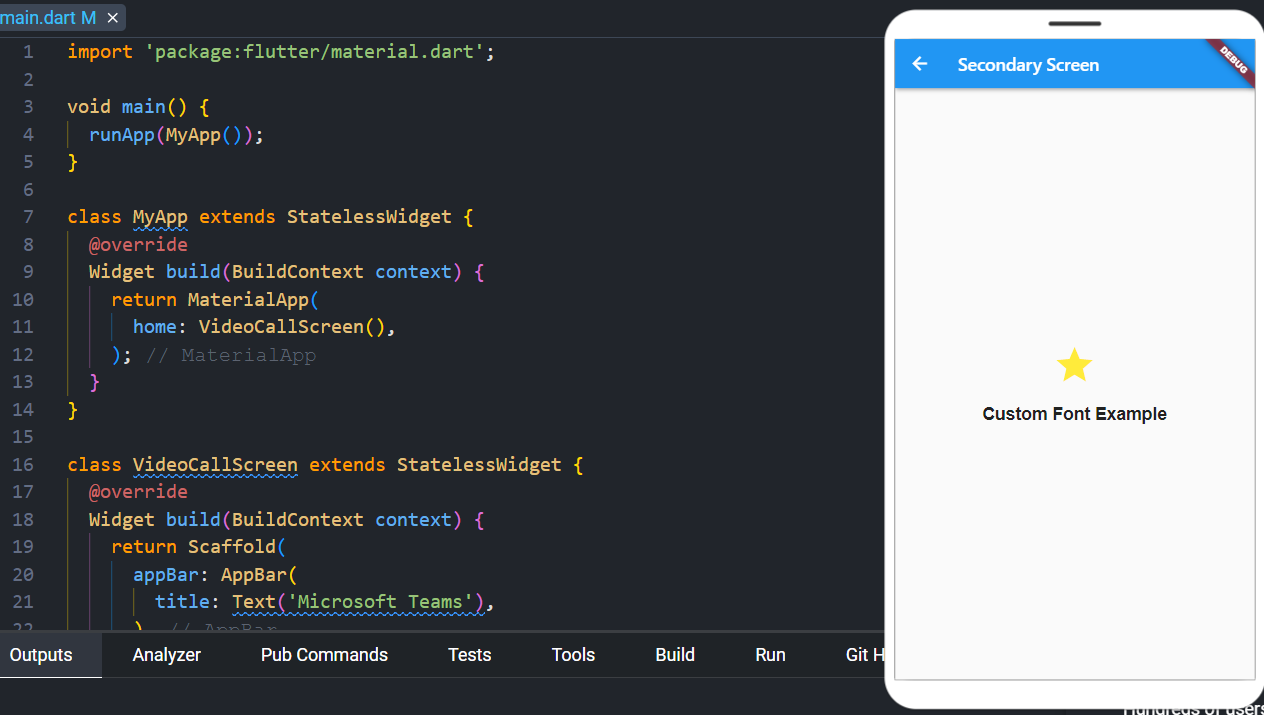
**);**

**}**

**}**

**OUTPUT:-**

****

****

**CONCLUSION:-**

**Learned to apply navigation, routing, and gestures in the Flutter App.**