MPL EXPERIMENT-5

Name: Ansh Sarfare **Class/Roll No :** D15A-49

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

Theory: -

Navigation, Routing, and Gesture Handling in Flutter
In Flutter, screens or pages are referred to as routes, and each route is essentially a widget. This concept is similar to Activities in Android. Navigating between pages defines an app's workflow, and the mechanism for handling this is known as routing.
Flutter provides a built-in routing system using MaterialPageRoute, along with the Navigator.push() and Navigator.pop() methods to move between routes.
Additionally, gestures allow apps to respond to user interactions like taps, swipes, and drags, making applications more dynamic and user-friendly.

Navigation and Routing in Flutter,

1. Using the Navigator Widget

Flutter's Navigator widget manages a stack of routes, enabling seamless navigation between screens.
Pushing a Route: Moves to a new screen using Navigator.push().
Popping a Route: Returns to the previous screen using Navigator.pop().

Example:

```
ElevatedButton(
 onPressed: () {
 Navigator.push(
 context,
 MaterialPageRoute(builder: (context) => SecondScreen()),
 );
 },
 child: Text('Go to Second Screen'),
);
```

2. Using Named Routes

For larger applications, named routes provide a cleaner and more structured way to manage navigation.
Step 1: Define Routes in MaterialApp

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

Step 2: Navigate Using Navigator.pushNamed()

```
Navigator.pushNamed(context, '/second');
```

Handling Gestures in Flutter

Gestures enable user interaction through taps, swipes, pinches, and drags. Flutter provides various widgets and gesture detectors to manage these interactions effectively.

1. Tap Gestures

Taps are one of the most common interactions and can be handled using:

GestureDetector
InkWell
ElevatedButton

Example (Tap Gesture using GestureDetector):

```
GestureDetector(
 onTap: () {
 print("Tapped!");
 },
 child: Container(
 padding: EdgeInsets.all(20),
 color: Colors.blue,
 child: Text('Tap Me'),
 ),
);
```

2. Long Press Gestures

Long-press interactions can be captured using the onLongPress callback in GestureDetector or InkWell.

```
InkWell(
 onLongPress: () {
 print("Long Pressed!");
 },
 child: Container(
 padding: EdgeInsets.all(20),
 color: Colors.red,
 child: Text('Long Press Me'),
 ),
);
```

3. Swipe and Drag Gestures

Flutter provides built-in methods like onHorizontalDragUpdate and onVerticalDragUpdate to detect swipe and drag actions.

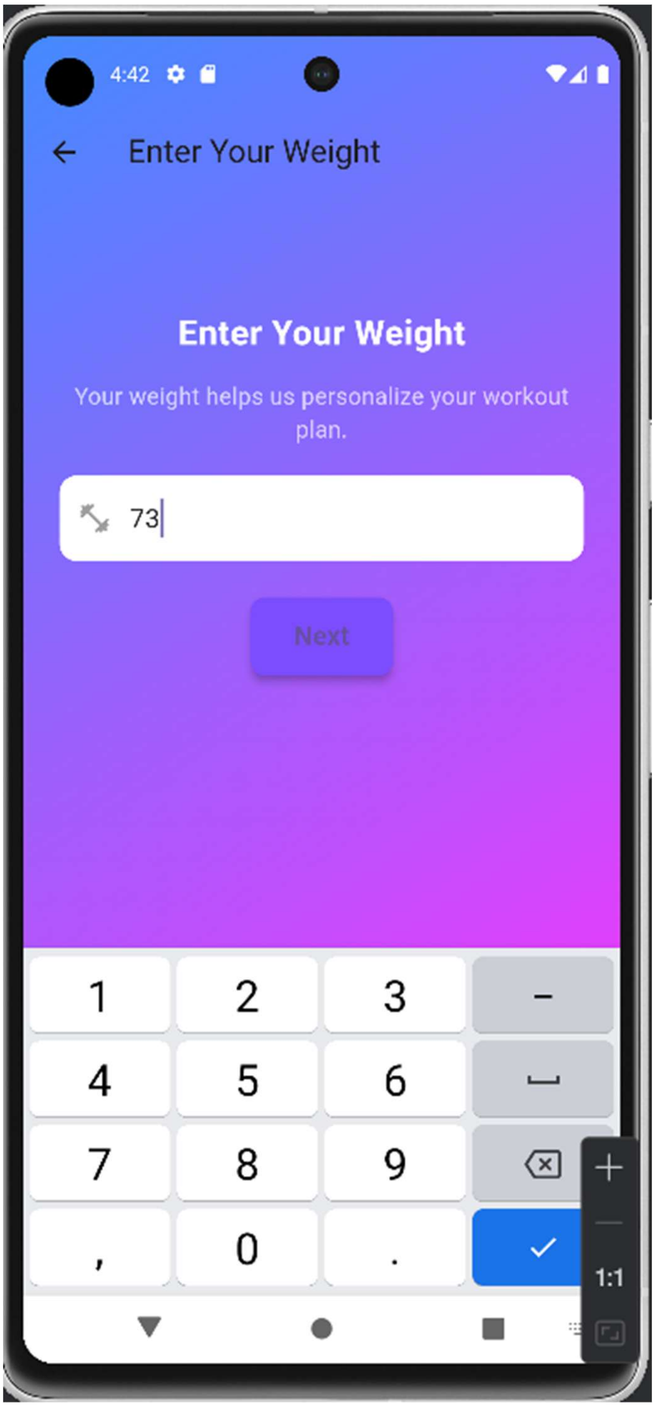
Example (Swipe Detection):

```
GestureDetector(
 onHorizontalDragUpdate: (details) {
 if (details.primaryDelta! > 0) {
 print("Swiped Right!");
 } else {
 print("Swiped Left!");
 }
 },
 child: Container(
 padding: EdgeInsets.all(20),
 color: Colors.green,
 child: Text('Swipe Me'),
 ),
);
```

Code:

```
import 'package:flutter/material.dart';
import 'workouts_page.dart'; // Import
the next page

class WeightPage extends
StatefulWidget {
  final String selectedGender;
  const WeightPage({super.key, required
this.selectedGender});

  @override
  State<WeightPage> createState() =>
_WeightPageState();
}

class _WeightPageState extends
State<WeightPage> {
  TextEditingController weightController
= TextEditingController();

  void proceedToWorkout() {
    if (weightController.text.isEmpty) {

ScaffoldMessenger.of(context).showSna
ckBar(
        SnackBar(content: Text("Please
enter your weight")),
      );
      return;
    }

    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => WorkoutPage(
          selectedGender:
widget.selectedGender,
          weight:
double.parse(weightController.text),
        ),
      ),
    );
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      extendBodyBehindAppBar: true,
      appBar: AppBar(
        title: Text("Enter Your Weight"),
        backgroundColor:
Colors.transparent,
        elevation: 0,
      ),
      body: Container(
        decoration: BoxDecoration(
          gradient: LinearGradient(
            colors: [Colors.blueAccent,
Colors.purpleAccent],
            begin: Alignment.topLeft,
            end: Alignment.bottomRight,
          ),
        ),
        child: Center(
          child: Padding(
            padding: const
EdgeInsets.symmetric(horizontal: 25.0),
            child: Column(
              mainAxisSize:
MainAxisSize.min,
              children: [
                Text(
                  "Enter Your Weight",
                  style: TextStyle(
                    fontSize: 24,
                    fontWeight: FontWeight.bold,
                    color: Colors.white,
                  ),
                ),
                SizedBox(height: 15),
                Text(
                  "Your weight helps us
personalize your workout plan.",
                  style: TextStyle(fontSize: 16,
color: Colors.white70),
         textAlign: TextAlign.center,
        ),
        SizedBox(height: 20),
        TextField(
         controller: weightController,
         decoration: InputDecoration(
          filled: true,
          fillColor: Colors.white,
          border: OutlineInputBorder(
           borderRadius:
BorderRadius.circular(12),
           borderSide:
BorderSide.none,
          ),
          hintText: "Weight in kg",
          prefixIcon:
Icon(Icons.fitness_center, color:
Colors.grey),
         ),
         keyboardType:
TextInputType.number,
         style: TextStyle(fontSize: 18),
        ),
        SizedBox(height: 25),
        ElevatedButton(
         onPressed:
proceedToWorkout,
         style:
ElevatedButton.styleFrom(
          padding:
EdgeInsets.symmetric(vertical: 14,
horizontal: 30),
          shape:
RoundedRectangleBorder(
           borderRadius:
BorderRadius.circular(10),
          ),
          backgroundColor:
Colors.deepPurpleAccent,
          elevation: 5,
         ),
         child: Text(
          "Next",
          style: TextStyle(fontSize: 18,
fontWeight: FontWeight.bold),
         ),
        ),
       ],
      ),
     ),
    ),
   ),
  );
 }
}
```

Output:



4:42

Enter Your Weight

Enter Your Weight

Your weight helps us personalize your workout plan.

73

Next