

## **EXPERIMENT NO: - 05**

**Name:-** Ronak Katariya

**Class:-** D15A

**Roll:No: -** 23

**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(  

```

```
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.

**Code: -**

main.dart

```
import 'package:flutter/material.dart';
import
'package:quiz/screens/login_screen.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner:
false,
      title: "Login & SignUp",
      home: const LoginScreen(),
    );
  }
}
initialRoute: '/login', // Set initial page
routes: {
  '/login': (context) => LoginPage(),
  '/register': (context) => RegistrationPage(),
  '/home': (context) => HomePage(),
},
);
}
```

## Login\_page.dart

```
import 'package:flutter/material.dart';
import 'signup_screen.dart';
import 'landing_page.dart'; // Import the
LandingPage

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

  @override
  State<LoginScreen> createState() =>
  _LoginScreenState();
}

class _LoginScreenState extends
State<LoginScreen> {
  final TextEditingController _emailController =
TextEditingController();
  final TextEditingController _passwordController =
TextEditingController();

  @override
  void dispose() {
    _emailController.dispose();
    _passwordController.dispose();
    super.dispose();
  }

  void _login(BuildContext context) {
    String email = _emailController.text;
    String password = _passwordController.text;

    if (email == 'abc@gmail.com' && password ==
'abc') {
      // Simulate successful login
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text('Login
Successful!')),
      );
      // Navigate to the LandingPage
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(builder: (context) => const
LandingPage()),
      );
    } else {
      // Show error message for invalid credentials
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text('Invalid email or
password.')),
      );
    }
  }
}
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Container(
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [Color(0xFF4A00E0),
Color(0xFF8E2DE2)],
        ),
      ),
    child: Center(
      child: Padding(
        padding: const EdgeInsets.all(20.0),
        child: Column(
          mainAxisAlignment:
MainAxisAlignment.center,
          children: [
            const Text(
              "Welcome Back",
              style: TextStyle(fontSize: 28, fontWeight:
FontWeight.bold, color: Colors.white),
            ),
            const SizedBox(height: 10),
            const Text(
              "Enter your credentials to login",
              style: TextStyle(fontSize: 16, color:
Colors.white70),
            ),
            const SizedBox(height: 30),
            _buildTextField(Icons.email, "Email",
controller: _emailController),
            const SizedBox(height: 15),
            _buildTextField(Icons.lock, "Password",
isPassword: true, controller: _passwordController),
            const SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                _login(context);
              },
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.white,
                foregroundColor: Colors.deepPurple,
                minimumSize: const Size(double.infinity,
50),
                shape:
RoundedRectangleBorder(borderRadius:
BorderRadius.circular(30)),
              ),
              child: const Text("Login", style:
TextStyle(fontSize: 18)),
            ),
          ],
        ),
      ),
    ),
  );
}
```

```

const SizedBox(height: 10),
      TextButton(
        onPressed: () {},
        child: const Text("Forgot password?",
style: TextStyle(color: Colors.white)),
      ),
      const SizedBox(height: 10),
      Row(
        mainAxisAlignment:
MainAxisAlignment.center,
        children: [
          const Text("Don't have an account?",
style: TextStyle(color: Colors.white)),
          TextButton(
            onPressed: () {
              Navigator.push(context,
MaterialPageRoute(builder: (_) => const
SignUpScreen()));
            },
            child: const Text("Sign Up", style:
TextStyle(fontWeight: FontWeight.bold, color:
Colors.white)),
          ),
        ],
      ),
    ],
  ),
),
},
}

```

```

Widget _buildTextField({
  required String label,
  required String hint,

```

```

    ),
  ),
  ),
);
}

Widget _buildTextField(IconData icon, String hint,
{bool isPassword = false, required
TextEditingController controller}) {
  return TextField(
    controller: controller,
    obscureText: isPassword,
    style: const TextStyle(color: Colors.white),
    decoration: InputDecoration(
      prefixIcon: Icon(icon, color: Colors.white),
      hintText: hint,
      hintStyle: const TextStyle(color: Colors.white70),
      filled: true,
      fillColor: Colors.white.withOpacity(0.2),
      border: OutlineInputBorder(borderRadius:
BorderRadius.circular(30), borderSide:
BorderSide.none),
    ),
  );
}
}

```

### **Landing\_page.dart**

```
import 'package:flutter/material.dart';
import 'package:quiz/widgets/bottom_navbar.dart';
import 'package:quiz/screens/category_selection_screen.dart';
import 'package:quiz/screens/leaderboard.dart';
import 'package:quiz/screens/profile.dart';
```

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

  @override
  _LandingPageState createState() => _LandingPageState();
}
```

```
class _LandingPageState extends State<LandingPage> {
  int _selectedIndex = 0;
```

```
  final List<Widget> _pages = [
    const HomeScreen(), // Landing Page UI
    const CategorySelectionScreen(),
    const LeaderboardPage(),
    const ProfilePage(),
  ];
```

```
  void _onItemTapped(int index) {
    setState(() {
      _selectedIndex = index;
    });
  }
```

```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: _pages[_selectedIndex],
      bottomNavigationBar: BottomNavBar(
        currentIndex: _selectedIndex,
        onTap: _onItemTapped,
      ),
    );
  }
}
```

```
class HomeScreen extends StatelessWidget {
  const HomeScreen({super.key});
```

```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: HomeContent(),
    );
  }
}
```

```
class HomeContent extends StatelessWidget {
  const HomeContent({super.key});
```

```

@override
Widget build(BuildContext context) {
  return Container(
    decoration: const BoxDecoration(
      gradient: LinearGradient(
        colors: [Color(0xFF6A11CB), Color(0xFF2575FC)],
        begin: Alignment.topCenter,
        end: Alignment.bottomCenter,
      ),
    ),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        const Padding(
          padding: EdgeInsets.symmetric(horizontal: 20),
          child: Text(
            "Test Your Knowledge 🚀",
            style: TextStyle(
              fontSize: 26,
              fontWeight: FontWeight.bold,
              color: Colors.white,
            ),
            textAlign: TextAlign.center,
          ),
        ),
        const SizedBox(height: 20),
        _buildCategoryCards(context),
        const SizedBox(height: 20),
        ElevatedButton(
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => const CategorySelectionScreen()),
            );
          },
          style: ElevatedButton.styleFrom(
            backgroundColor: Colors.white,
            foregroundColor: Colors.blueAccent,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(30),
            ),
            padding: const EdgeInsets.symmetric(horizontal: 40, vertical: 15),
          ),
          child: const Text("Start Quiz", style: TextStyle(fontSize: 18)),
        ),
      ],
    ),
  );
}

```

/// ✅ Function to build category cards

```

Widget _buildCategoryCards(BuildContext context) {
  List<Map<String, String>> categories = [
    {"title": "CNS", "image": "assets/cns.jpg"},
    {"title": "OS", "image": "assets/os.jpg"},
  ];
}

```

```


    {"title": "DSA", "image": "assets/dsa.jpg"},
    {"title": "SQL", "image": "assets/sql.png"},
  ];

```

```

return SizedBox(
  height: 200,
  child: ListView.builder(
    scrollDirection: Axis.horizontal,
    itemCount: categories.length,
    itemBuilder: (context, index) {
      return _categoryCard(categories[index]["title"]!, categories[index]["image"]!, context);
    },
  ),
);
}

```

///  *Category Card Widget*

```

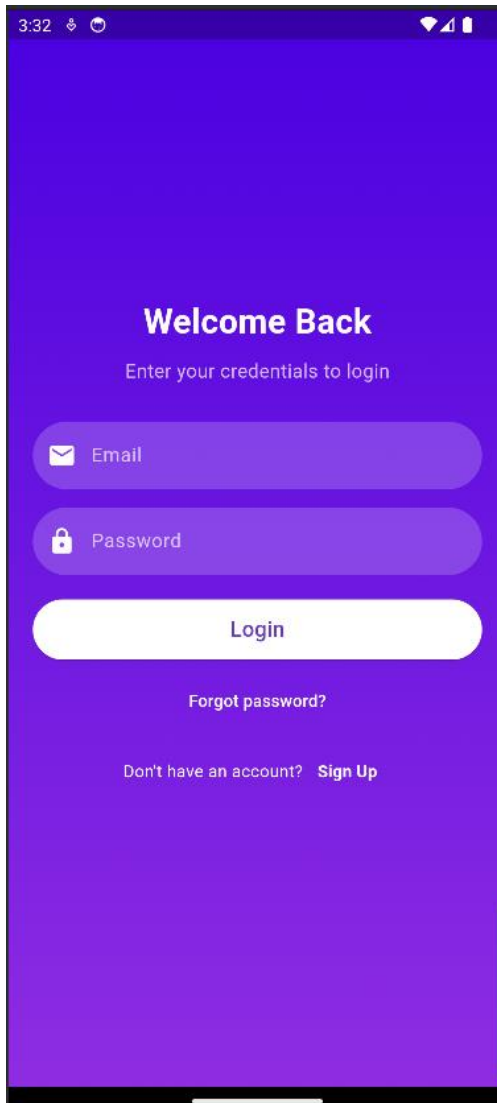
Widget _categoryCard(String title, String imagePath, BuildContext context) {
  return GestureDetector(
    onTap: () {
      Navigator.push(
        context,
        MaterialPageRoute(builder: (context) => const CategorySelectionScreen()),
      );
    },
    child: Card(
      margin: const EdgeInsets.symmetric(horizontal: 10),
      shape: RoundedRectangleBorder(
        borderRadius: BorderRadius.circular(20),
      ),
      elevation: 5,
      child: Container(
        width: 150,
        decoration: BoxDecoration(
          borderRadius: BorderRadius.circular(20),
          image: DecorationImage(
            image: AssetImage(imagePath),
            fit: BoxFit.cover,
          ),
        ),
        child: Center(
          child: Text(
            title,
            style: const TextStyle(
              fontSize: 20,
              fontWeight: FontWeight.bold,
              color: Colors.white,
              backgroundColor: Colors.black54,
            ),
          ),
        ),
      ),
    ),
  );
}

```



## OUTPUT: -

After clicking on Don't have an account? it navigates to the registration page.



3:36

Sign Up

Create your account

Email

Password

Sign Up

Sign In with Google

Already have an account? [Login](#)

In home page, after clicking on Start Quiz it navigates to the Quiz categories.

