Name : Vivek Gupta
Div : D15B
Roll No : 19

MPL Practical 05

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

Theory:

Introduction
Navigation in Flutter allows users to switch between different screens (pages). It is implemented using routes and the Navigator widget. Routing can be defined using MaterialPageRoute, named routes, or the onGenerateRoute method. Gestures, such as taps and swipes, enhance user interaction using Flutter's GestureDetector.

Implementation in Our Code

- Named Routes: We defined named routes (/, /createTeam, /findTeams) in MaterialApp for structured navigation.
- Navigation: Navigator.pushNamed(context, routeName) is used to move between screens, and BottomNavigationBar provides seamless switching.
- Gestures: GestureDetector is used to handle taps on buttons (Create a Team, Find Teams) and list items in the TeamScreen.
- User Interaction: The navigation system ensures smooth movement across the Home Page, Create Team Page, and Find Teams Page with a bottom navigation bar for ease of access.

CODE:

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

void main() {
 runApp(TeamFinderApp());
}

class TeamFinderApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
  return MaterialApp(
   debugShowCheckedModeBanner: false,
   title: 'Team Finder',
   theme: ThemeData(
    primarySwatch: Colors.blue,
    fontFamily: 'Poppins',
    colorScheme: ColorScheme.fromSwatch().copyWith(
     primary: Colors.blue[900],
     secondary: Colors.blue[700],
    ),
```

```
      visualDensity: VisualDensity.adaptivePlatformDensity,
    ),
    initialRoute: '/',
    routes: {
     '/': (context) => HomePage(),
     '/createTeam': (context) => CreateTeamScreen(),
     '/findTeams': (context) => TeamScreen(),
     '/chat': (context) => ChatScreen(),
    },
   );
  }
}

class HomePage extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
  return Scaffold(
   appBar: AppBar(title: Text('Welcome to Team Finder'),
   leading: Icon(Icons.sports_soccer),
   ),
   body: HomeContent(),
   bottomNavigationBar: NavBar(currentIndex: 0),
  );
 }
}

class HomeContent extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
  return Padding(
   padding: EdgeInsets.all(16.0),
   child: Column(
    crossAxisAlignment: CrossAxisAlignment.center,
    children: [
     Center(
      child: Image.asset('assets/team_finder_logo.png', height: 150), // Display image
     ),
     Text(
      'Team Finder App',
      style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold, color: Colors.blue[900]),
     ),
```

```
SizedBox(height: 10),
Text(
 'Find and join local teams for practice, form your own team, and connect with fellow athletes.',
 textAlign: TextAlign.center,
 style: TextStyle(fontSize: 16, color: Colors.blue[700]),
),
SizedBox(height: 20),
GestureDetector(
 onTap: () {
  Navigator.pushNamed(context, '/createTeam');
 },
 child: Container(
  padding: EdgeInsets.all(15),
  decoration: BoxDecoration(
   color: Colors.lightBlue[200],
   borderRadius: BorderRadius.circular(10),
  ),
  child: Row(
   mainAxisAlignment: MainAxisAlignment.center,
   children: [
    Icon(Icons.add, color: Colors.white),
    SizedBox(width: 10),
    Text(
     'Create a Team',
     style: TextStyle(fontSize: 20, color: Colors.white),
    ),
   ],
  ),
 ),
),
SizedBox(height: 10),
GestureDetector(
 onTap: () {
  Navigator.pushNamed(context, '/findTeams');
 },
 child: Container(
  padding: EdgeInsets.all(15),
  decoration: BoxDecoration(
   color: Colors.lightBlue[200],
   borderRadius: BorderRadius.circular(10),
  ),
```

```
            child: Row(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                Icon(Icons.search, color: Colors.white),
                SizedBox(width: 10),
                Text(
                  'Find Teams',
                  style: TextStyle(fontSize: 20, color: Colors.white),
                ),
              ],
            ),
          ),
        ),
      ],
    ),
  );
 }
}

class CreateTeamScreen extends StatelessWidget {
 final _formKey = GlobalKey<FormState>();
 String? _teamName;
 String? _selectedSport;
 List<String> sports = ['Football', 'Basketball', 'Cricket', 'Tennis'];

 void _submitForm(BuildContext context) {
  if (_formKey.currentState!.validate()) {
   _formKey.currentState!.save();
   ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(content: Text('Team $_teamName for $_selectedSport created!')),
   );
  }
 }

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   appBar: AppBar(title: Text('Create a Team')),
   body: Padding(
    padding: EdgeInsets.all(16.0),
    child: Form(
```

```
key: _formKey,
child: Column(
 crossAxisAlignment: CrossAxisAlignment.start,
 children: [
  Text(
   'Create Your Team',
   style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold, color: Colors.blue[900]),
  ),
  SizedBox(height: 10),
  TextFormField(
   decoration: InputDecoration(
    labelText: 'Team Name',
    prefixIcon: Icon(Icons.group, color: Colors.blue[900]),
    border: OutlineInputBorder(),
   ),
   validator: (value) => value!.isEmpty ? 'Please enter a team name' : null,
   onSaved: (value) => _teamName = value,
  ),
  SizedBox(height: 10),
  DropdownButtonFormField<String>(
   decoration: InputDecoration(
    labelText: 'Select Sport',
    prefixIcon: Icon(Icons.sports_soccer, color: Colors.blue[900]),
    border: OutlineInputBorder(),
   ),
   items: sports.map((sport) {
    return DropdownMenuItem(
     value: sport,
     child: Text(sport),
    );
   }).toList(),
   validator: (value) => value == null ? 'Please select a sport' : null,
   onChanged: (value) => _selectedSport = value,
  ),
  SizedBox(height: 20),
  Center(
   child: ElevatedButton(
    onPressed: () => _submitForm(context),
    child: Text('Create Team'),
    style: ElevatedButton.styleFrom(
     backgroundColor: Colors.lightBlue[200],
```

```
            ),
          ),
        ),
      ],
    ),
  ),
 ),
 bottomNavigationBar: NavBar(currentIndex: 1),
 );
 }
}

class TeamScreen extends StatefulWidget {
 @override
 _TeamScreenState createState() => _TeamScreenState();
}

class _TeamScreenState extends State<TeamScreen> {
 final List<Map<String, String>> teams = [
  {'name': 'Warriors FC', 'sport': 'Football'},
  {'name': 'Thunder Hoops', 'sport': 'Basketball'},
  {'name': 'Strikers Club', 'sport': 'Cricket'}
 ];

 final Set<String> joinedTeams = {};

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   appBar: AppBar(title: Text('Find Teams')),
   body: ListView.builder(
    itemCount: teams.length,
    itemBuilder: (context, index) {
     String teamName = teams[index]['name']!;
     String sport = teams[index]['sport']!;
     bool isJoined = joinedTeams.contains(teamName);

     return Card(
      margin: EdgeInsets.all(10),
      child: ListTile(
```

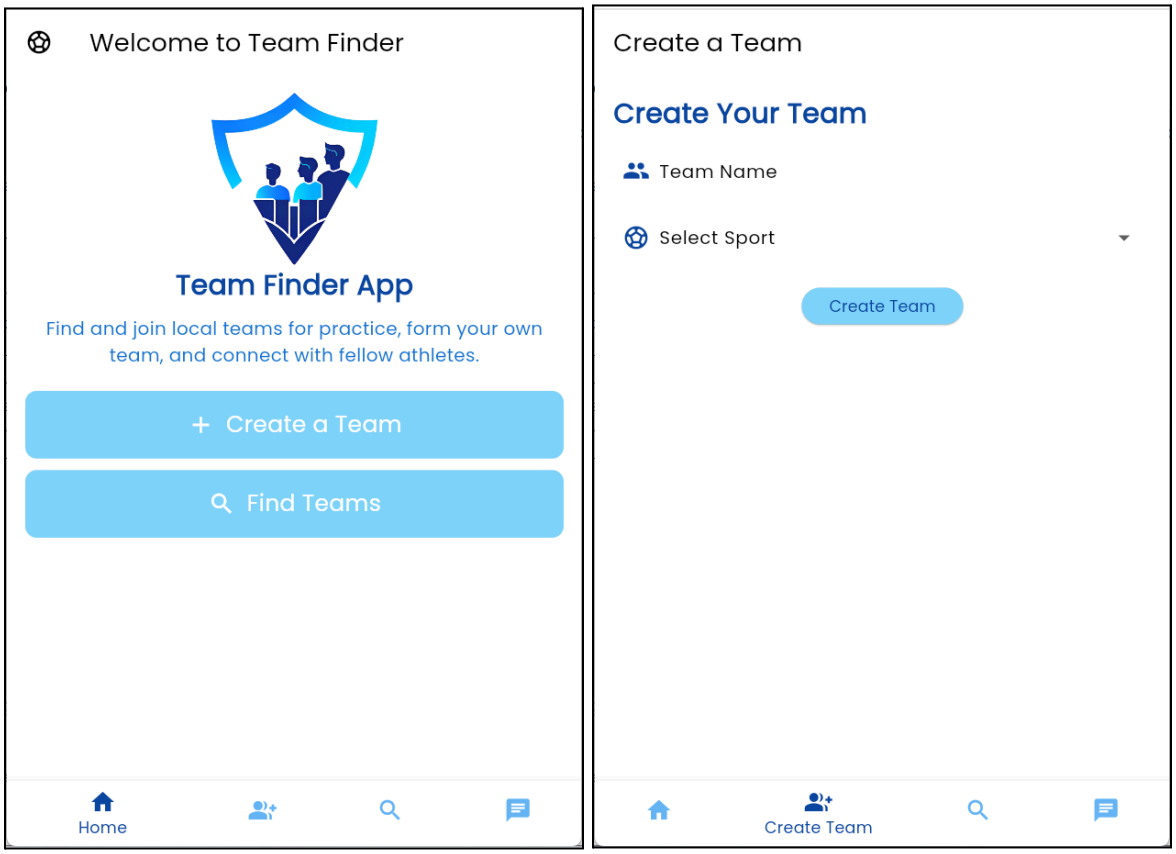
```
                    title: Text(teamName, style: TextStyle(color: Colors.blue[900], fontWeight:
FontWeight.bold)),
          subtitle: Text(sport, style: TextStyle(color: Colors.blue[700])),
          trailing: ElevatedButton(
           onPressed: () {
            setState(() {
             if (isJoined) {
              joinedTeams.remove(teamName);
             } else {
              joinedTeams.add(teamName);
             }
            });
           },
           style: ElevatedButton.styleFrom(
            backgroundColor: isJoined ? Colors.blue[900] : Colors.blue[300],
            shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(20)),
           ),
           child: Text(
            isJoined ? 'Leave' : 'Join',
            style: TextStyle(color: Colors.white),
           ),
          ),
         ),
        );
       },
      ),
      bottomNavigationBar: NavBar(currentIndex: 2),
    );
  }
}

// Chat Screen
class ChatScreen extends StatefulWidget {
 @override
 _ChatScreenState createState() => _ChatScreenState();
}

class _ChatScreenState extends State<ChatScreen> {
 final List<String> messages = [];
 final TextEditingController _controller = TextEditingController();
```

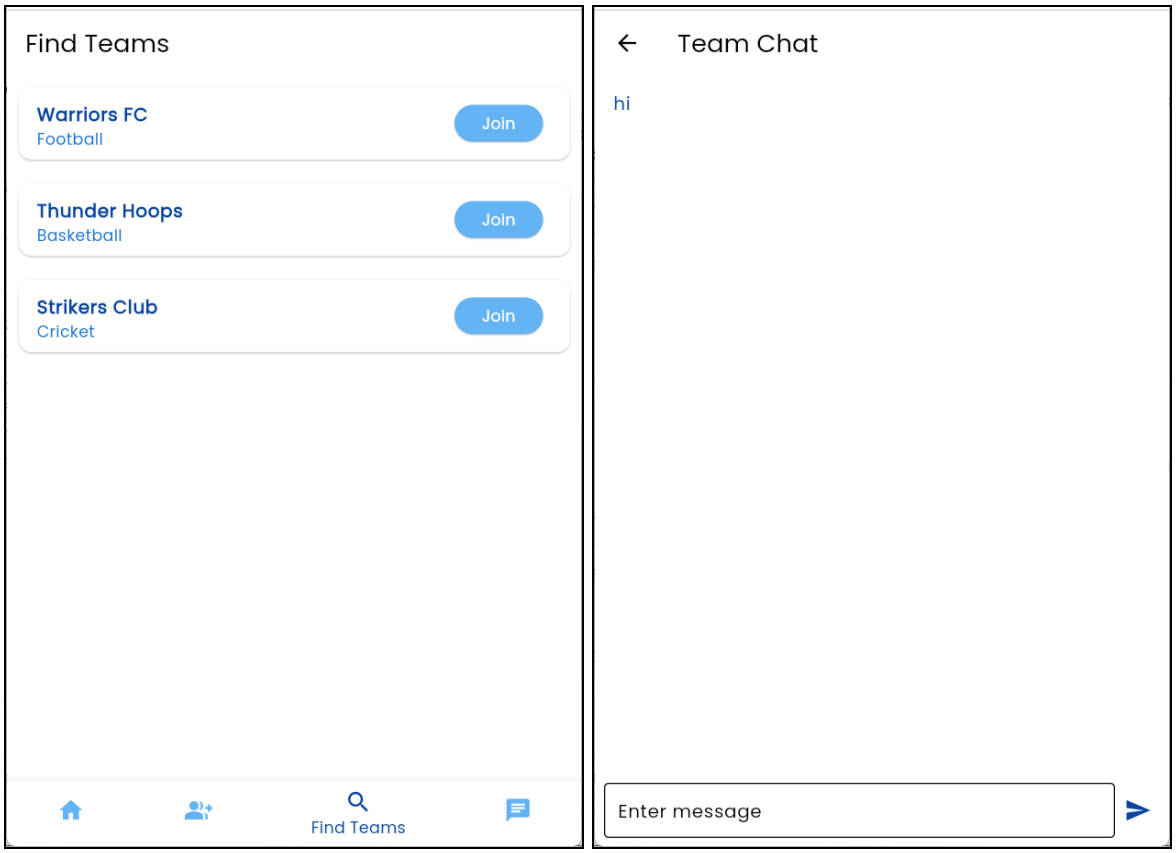
```
void _sendMessage() {
 if (_controller.text.isNotEmpty) {
  setState(() {
   messages.add(_controller.text);
  });
  _controller.clear();
 }
}

@override
Widget build(BuildContext context) {
 return Scaffold(
  appBar: AppBar(
   title: Text("Team Chat"),
   leading: IconButton(
    icon: Icon(Icons.arrow_back, color: Colors.black), // Back button
    onPressed: () {
     Navigator.pop(context); // Navigate back
    },
   ),
  ),
  body: Column(
   children: [
    Expanded(
     child: ListView.builder(
      itemCount: messages.length,
      itemBuilder: (context, index) => ListTile(
       title: Text(messages[index], style: TextStyle(color: Colors.blue[900])),
      ),
     ),
    ),
    Padding(
     padding: const EdgeInsets.all(8.0),
     child: Row(
      children: [
       Expanded(
        child: TextField(
         controller: _controller,
         decoration: InputDecoration(
          hintText: "Enter message",
          border: OutlineInputBorder(),
```

```
              ),
            ),
          ),
          IconButton(
            icon: Icon(Icons.send, color: Colors.blue[900]),
            onPressed: _sendMessage,
          ),
        ],
      ),
    ),
  ],
 ),
);
}
}

class NavBar extends StatelessWidget {
 final int currentIndex;
 NavBar({required this.currentIndex});

 @override
 Widget build(BuildContext context) {
  return BottomNavigationBar(
   currentIndex: currentIndex,
   selectedItemColor: Colors.blue[900],
   unselectedItemColor: Colors.blue[300],
   onTap: (index) {
    if (index == 0) {
     Navigator.pushReplacementNamed(context, '/');
    } else if (index == 1) {
     Navigator.pushReplacementNamed(context, '/createTeam');
    } else if (index == 2) {
     Navigator.pushReplacementNamed(context, '/findTeams');
    } else if (index == 3) { // Chat navigation added
     Navigator.pushReplacementNamed(context, '/chat');
    }
   },
   items: [
    BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),
    BottomNavigationBarItem(icon: Icon(Icons.group_add), label: 'Create Team'),
    BottomNavigationBarItem(icon: Icon(Icons.search), label: 'Find Teams'),
```

```
      BottomNavigationBarItem(icon: Icon(Icons.chat), label: "Chat"), // Added Chat button
    ],
   );
  }
}
```

OUTPUT:





Conclusion:

In this experiment, we successfully implemented navigation using named routes and integrated gestures for user interaction in the **Team Finder** app. Initially, we faced issues with incorrect route navigation and unresponsive gestures, which we resolved by debugging route names and ensuring GestureDetector was properly wrapped around interactive widgets.