

MPL Experiment 5

Name: Prathamesh Palve

Class: D15A

Roll no:31

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

Theory:

Navigation & Routing:

Flutter provides multiple ways to navigate between screens (pages):

1. **Using Navigator.push() and Navigator.pop()**
Push a new screen onto the stack and pop it to return to the previous one.
2. **Named Routes**
Define routes in MaterialApp and navigate using Navigator.pushNamed().
3. **GoRouter Package**
A declarative approach to handle navigation more efficiently.

Gestures:

Flutter's GestureDetector and InkWell widgets help in detecting user interactions like taps, swipes, and long presses. Some common gestures include:

- **onTap:** Detects a tap event.
- **onDoubleTap:** Recognizes double taps.
- **onLongPress:** Detects a long press on a widget.
- **onHorizontalDrag & onVerticalDrag:** Detects drag/swipe motions.

Steps:

Step 1: Create a Flutter project and define multiple screens.

Step 2: Set up named routes in MaterialApp.

Step 3: Implement navigation using Navigator.push() and Navigator.pushNamed().

Step 4: Use GestureDetector to detect taps, swipes, and long presses.

Step 5: Add buttons or swipes to navigate between screens.

Code:

```
import 'package:flutter/material.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter_dotenv/flutter_dotenv.dart'; // For accessing Mistral
API key
import '../services/api_service.dart';
import '../services/firestore_service.dart';
import '../widgets/chat_bubble.dart';
import '../widgets/empty_state.dart';
import '../widgets/bottom_bar.dart';
```

```

import 'setting.dart';
import 'left_slidder.dart';

class HomeScreen extends StatefulWidget {
  final String chatId; // Accepts chatId dynamically

  HomeScreen({required this.chatId});

  @override
  _HomeScreenState createState() => _HomeScreenState();
}

class _HomeScreenState extends State<HomeScreen> {
  final TextEditingController _messageController = TextEditingController();
  final FirebaseAuth _auth = FirebaseAuth.instance;
  final FirestoreService _firestoreService = FirestoreService();
  final ApiService _apiService = ApiService();
  final ScrollController _scrollController = ScrollController();

  void _sendMessage() async {
    final user = _auth.currentUser;
    if (user == null || _messageController.text.trim().isEmpty) return;

    String userId = user.uid;
    String chatId = widget.chatId;
    String userMessage = _messageController.text.trim();

    try {
      // Store user message in Firestore
      await _firestoreService.addMessage(userId, chatId, userMessage, true);

      setState(() {
        _messageController.clear();
      });

      // Scroll to bottom after sending message
      Future.delayed(Duration(milliseconds: 300), () {
        _scrollToBottom();
      });

      // Get response from Mistral API
      String aiResponse = await _apiService.getMistralResponse(userMessage);

      // Store AI response in Firestore
      await _firestoreService.addMessage(userId, chatId, aiResponse, false);
    }
  }
}

```

```

        print("✅ AI Response stored successfully!");
    } catch (e) {
        print("❌ Error sending message: $e");
    }
}

void _scrollToBottom() {
    if (_scrollController.hasClients) {
        _scrollController.animateTo(
            0, // Since we reversed ListView, scroll to position 0
            duration: Duration(milliseconds: 300),
            curve: Curves.easeOut,
        );
    }
}

@override
Widget build(BuildContext context) {
    final user = _auth.currentUser;
    if (user == null) {
        return Scaffold(
            body: Center(child: Text("User not logged in")),
        );
    }

    return Scaffold(
        backgroundColor: Colors.black,
        appBar: AppBar(
            backgroundColor: Colors.black,
            elevation: 0,
            leading: Builder(
                builder: (context) => IconButton(
                    icon: Icon(Icons.menu, color: Colors.white),
                    onPressed: () {
                        Scaffold.of(context).openDrawer();
                    },
                ),
            ),
        actions: [
            TextButton(
                onPressed: () {},
                child: Text("Get Plus ✨", style: TextStyle(color: Colors.white)),
            ),
            IconButton(

```

```

        icon: Icon(Icons.more_vert, color: Colors.white),
        onPressed: () {},
      ),
    ],
  ),
  drawer: LeftSlider(currentChatId: widget.chatId),
  body: Column(
    children: [
      Expanded(
        child: StreamBuilder<QuerySnapshot>(
          stream: _firestoreService.getMessages(user.uid, widget.chatId),
          builder: (context, snapshot) {
            if (snapshot.connectionState == ConnectionState.waiting) {
              return Center(child: CircularProgressIndicator());
            }

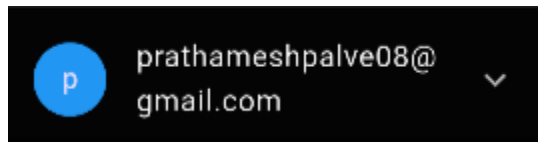
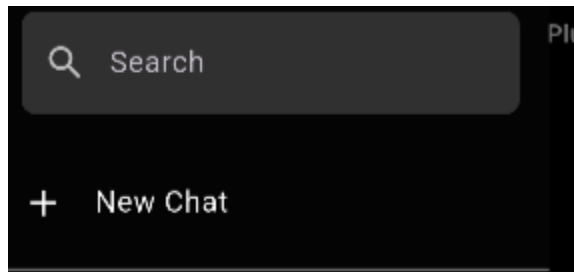
            if (!snapshot.hasData || snapshot.data!.docs.isEmpty) {
              return EmptyState(); // Show empty state if no messages
            }

            return ListView(
              controller: _scrollController,
              padding: EdgeInsets.all(16),
              reverse: true, // 🔥 Messages appear from bottom to top
              children: snapshot.data!.docs.map((doc) {
                Map<String, dynamic> data = doc.data() as Map<String,
dynamic>;

                bool isUser = data['isUser'] ?? false;
                return ChatBubble(message: data['message'] ?? '', isUser:
isUser);
              }).toList(),
            );
          },
        ),
      ),
      BottomBar(
        messageController: _messageController,
        onSendPressed: _sendMessage,
      ),
    ],
  ),
);
}
}

```

Output:



8:38



Settings



prathmzz

prathameshpalve08@gmail.com



Email

prathameshpalve08@gmail.com



Phone number

Not linked



Upgrade to Plus



Customize



Data Controls



Voice



About



Sign out

8:38




Get Plus ✨



What can I help with?

 Create Image

 Summarize Text

 Surprise Me

 Make a Plan

... More

Message



8:28



Sign Up



Sign Up

Already have an account? [Sign In](#)