EXP 5

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

Theory:

Navigation and routing are essential for managing transitions between different screens in a Flutter app. Flutter uses a stack-based navigation model, where new routes (screens) are pushed onto the stack using the Navigator.push() method and removed using Navigator.pop(). Routes can be defined inline using MaterialPageRoute or globally using named routes for better structure in larger applications.

Routing ensures that the app is organized, scalable, and easy to navigate. It also helps manage screen transitions such as going from a home page to a details page or form screen.

Gestures in Flutter enable apps to respond to user input like tapping, swiping, dragging, or long pressing. Widgets like GestureDetector, InkWell, and InkResponse are used to detect and handle these gestures. For example, wrapping a Container in a GestureDetector allows it to respond to a tap or swipe, making the interface more interactive and user-friendly.

By combining navigation, routing, and gesture handling, a Flutter app becomes more engaging, intuitive, and functional for the user.

Code:

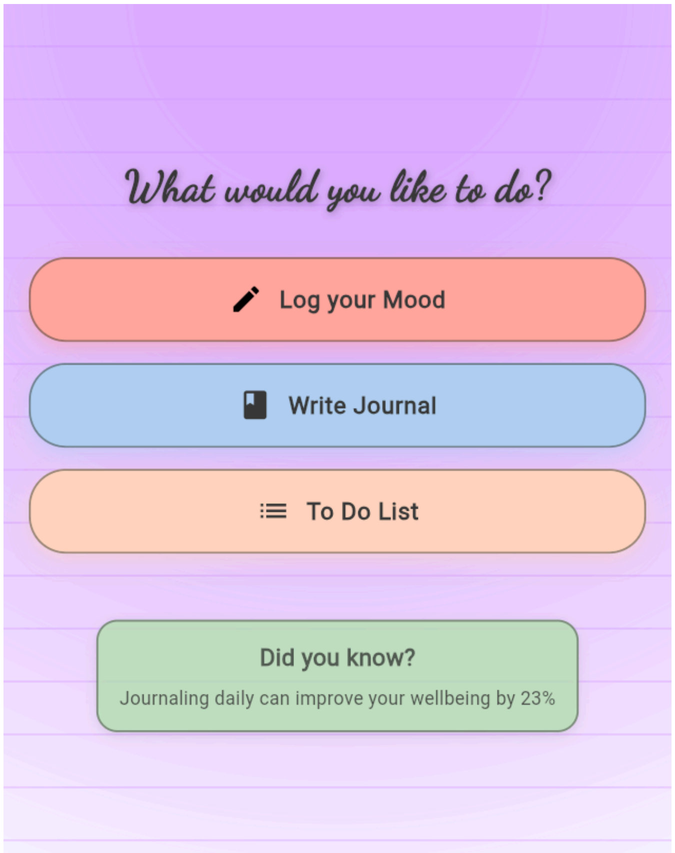
SelectionScreen.dart:

```
import 'package:flutter/material.dart';
import 'package:mood_log/TrackMood.dart';
import 'package:mood_log/models/MoodEntry.dart';
import 'package:mood_log/todo_screen.dart';
import 'package:mood_log/JournalScreen.dart';

class SelectionScreen extends StatelessWidget {
  const SelectionScreen({Key? key}) : super(key: key);
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Stack(
      children: [
        Positioned.fill(
          child: Container(
            decoration: BoxDecoration(
              gradient: RadialGradient(
                center: Alignment.topCenter,
                radius: 1.5,
                colors: [primaryColor.withOpacity(0.7), Colors.white],
                stops: const [0.2, 1.0],
              ),
            ),
          ),
        ),
        Positioned.fill(
          child: CustomPaint(
            painter: NotebookLinePainter(accentColor: primaryColor.withOpacity(0.4)),
          ),
        ),
        SafeArea(
          child: Padding(
            padding: const EdgeInsets.symmetric(horizontal: 24.0),
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              crossAxisAlignment: CrossAxisAlignment.center,
              children: [
                Text(
                  "What would you like to do?",
                  style: TextStyle(
                    fontFamily: 'DancingScript',
                    fontSize: 32,
                    color: textDark,
                    fontWeight: FontWeight.w700,
                    shadows: [
                      Shadow(
                        offset: const Offset(1.0, 1.0),
                        blurRadius: 3.0,
                        color: Colors.black.withOpacity(0.25),
                      ),
                    ],
                  ),
                ),
                const SizedBox(height: 30),
                _buildOptionButton(
  context,
  "Log your Mood",
  Icons.edit,
  softCoral,
  textColor: textDark,
  iconColor: const Color.fromARGB(255, 2, 2, 2),
  destination: TrackMoodScreen(), // Add the target screen here
),
const SizedBox(height: 16),
_buildOptionButton(
  context,
  "Write Journal",
  Icons.book,
  deepSkyBlue,
  textColor: textDark,
  destination: JournalScreen(), // Add the target screen here
),
const SizedBox(height: 16),
_buildOptionButton(
  context,
  "To Do List",
  Icons.list,
  deepPeach,
  textColor: textDark,
  destination: ToDoScreen(), // Add the target screen here
),
const SizedBox(height: 30),
const SizedBox(height: 20),
Container(
  padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 14),
  decoration: BoxDecoration(
    color: deepMint.withOpacity(0.8),
    borderRadius: BorderRadius.circular(16),
    boxShadow: [
      BoxShadow(
        color: Colors.black.withOpacity(0.08),
        blurRadius: 6,
        offset: const Offset(0, 3),
      ),
    ],
    border: Border.all(
      color: textDark.withOpacity(0.5),
      width: 1.5,
    ),
  ),
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
                    crossAxisAlignment: CrossAxisAlignment.center,
                    children: [
                      Text(
                        "Did you know?",
                        style: TextStyle(
                          fontSize: 18,
                          fontWeight: FontWeight.w600,
                          color: textDark.withOpacity(0.8),
                        ),
                      ),
                      const SizedBox(height: 8),
                      Text(
                        "Journaling daily can improve your wellbeing by 23%",
                        style: TextStyle(
                          fontSize: 14,
                          color: textDark.withOpacity(0.85),
                          fontWeight: FontWeight.w500,
                        ),
                      ),
                    ],
                  ),
                ),
              ],
            ),
          ),
        ),
      ],
    ),
  );
}

Widget _buildOptionButton(
  BuildContext context,
  String text,
  IconData icon,
  Color color, {
  Color textColor = Colors.white,
  Color? iconColor,
  required Widget destination, // Destination screen as required parameter
}) {
  return Container(
    width: double.infinity,
    height: 64,
    decoration: BoxDecoration(
      color: color,
      borderRadius: BorderRadius.circular(28),
      boxShadow: [
    BoxShadow(
      color: color.withOpacity(0.5),
      blurRadius: 10,
      offset: const Offset(0, 5),
    ),
  ],
  border: Border.all(
    color: textDark.withOpacity(0.5),
    width: 1.5,
  ),
),
child: ElevatedButton(
  onPressed: () {
    // Navigate to the corresponding screen when the button is pressed
    Navigator.push(
      context,
      MaterialPageRoute(builder: (context) => destination),
    );
  },
  style: ElevatedButton.styleFrom(
    backgroundColor: color,
    foregroundColor: textColor,
    elevation: 0,
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(28),
    ),
  ),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Icon(
        icon,
        size: 26,
        color: iconColor ?? textColor,
      ),
      const SizedBox(width: 12),
      Text(
        text,
        style: TextStyle(
          fontSize: 18,
          fontWeight: FontWeight.w600,
          letterSpacing: 0.5,
          color: textColor,
        ),
      ),
    ],
  ),
      ),
    );
  }
}


class NotebookLinePainter extends CustomPainter {
  final Color accentColor;

  const NotebookLinePainter({required this.accentColor});

  @override
  void paint(Canvas canvas, Size size) {
    final Paint paint = Paint()
      ..color = accentColor
      ..strokeWidth = 1.2
      ..style = PaintingStyle.stroke;

    double lineSpacing = 40.0;
    for (double i = 0; i < size.height; i += lineSpacing) {
      canvas.drawLine(Offset(0, i), Offset(size.width, i), paint);
    }
  }

  @override
  bool shouldRepaint(covariant NotebookLinePainter oldDelegate) {
    return oldDelegate.accentColor != accentColor;
  }
}
```

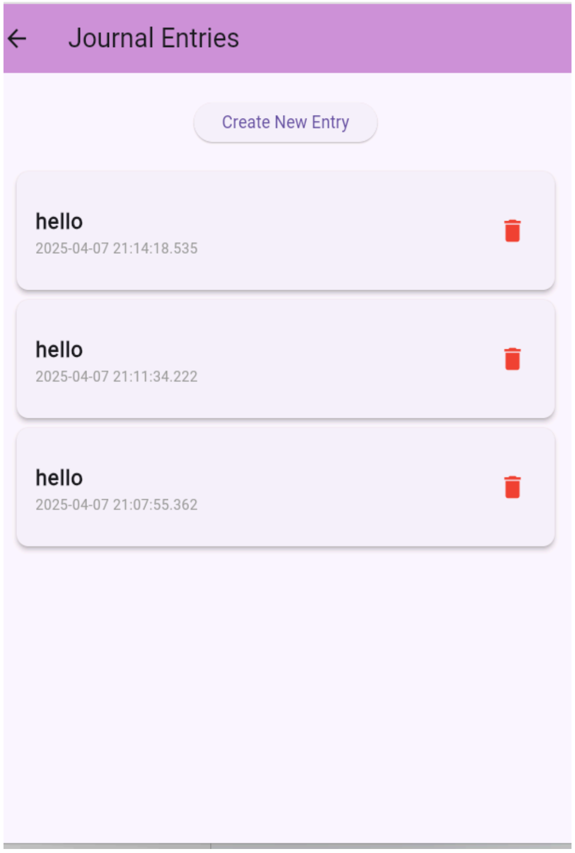
Output:

Clicking the write journal screen opens the Journal entries page:



Did you know?

Journaling daily can improve your wellbeing by 23%



hello
2025-04-07 21:14:18.535

hello
2025-04-07 21:11:34.222

hello
2025-04-07 21:07:55.362

Conclusion:

In conclusion, applying navigation, routing, and gestures in a Flutter app enhances user experience by enabling smooth transitions between screens, organized app structure through defined routes, and interactive features through gesture detection. These elements are essential for building intuitive and responsive mobile applications.