MPL EXP 5

NAME:VAISHNAL DILIP MALI
CLASS:D15A
ROLL NO.:26

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');
```

3. 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.

4.Tap Gestures

The most common gesture is the tap, which can be handled using the GestureDetector widget or specific buttons like InkWell or ElevatedButton.

5.Long Press Gesture

For long press gestures, Flutter provides the onLongPress callback in GestureDetector or InkWell.

6.Swipe and Drag Gestures

Flutter also provides swipe and drag gesture handling. The onHorizontalDragUpdate and onVerticalDragUpdate callbacks are used for dragging gestures.

Main.dart

```
import 'package:flutter/material.dart';
import
'features/auth/screens/home_screen.dart';
import
'features/auth/screens/search_screen.dart';
import
'features/auth/screens/profile_page.dart';
import
'features/auth/screens/music_screen.dart';
import
'features/auth/screens/sign_in_screen.dart';
import
'features/auth/screens/loading_screen.dart';
import
'features/auth/screens/register_screen.dart';
import
'package:flutter_riverpod/flutter_riverpod.
dart';
import 'config/theme.dart';


void main() {
  runApp(
    const ProviderScope(
      child: SpotifyClone(),
    ),
  );
}
class SpotifyClone extends
StatelessWidget {
  const SpotifyClone({Key? key}) :
super(key: key);
// Keep instances of screens to maintain
their state
  final List<Widget> _screens = const [
    HomeScreen(),
    SearchScreen(),
    MusicScreen(),
    ProfileScreen(),
  ];
```

```
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner:
false,
      title: 'Spotify Clone',
      theme: AppTheme.lightTheme,
      home: const LoadingScreen(), //
Changed initialRoute to home
      routes: {
        '/signin': (context) => const
SignInScreen(),
        '/main': (context) => const
MainScreen(),
      },
    );
  }
}

class MainScreen extends StatefulWidget
{
  const MainScreen({Key? key}) :
super(key: key);

  @override
  State<MainScreen> createState() =>
_MainScreenState();
}

class _MainScreenState extends
State<MainScreen> {
  int _selectedIndex = 0;
 type: BottomNavigationBarType.fixed,
          backgroundColor: const
Color(0xFF282828),
          selectedItemColor: Colors.white,
          unselectedItemColor: Colors.grey,
          selectedIconTheme: const
IconThemeData(
            size: 28,
void _onItemTapped(int index) {
    setState(() {
      _selectedIndex = index;
    });
  }

  @override
  Widget build(BuildContext context) {
    return WillPopScope(
      // Handle back button press
      onWillPop: () async {
        if (_selectedIndex != 0) {
          setState(() {
            _selectedIndex = 0;
          });
          return false;
        }
        return true;
      },
      child: Scaffold(
        // Use IndexedStack to preserve state
of all screens
        body: IndexedStack(
          index: _selectedIndex,
          children: _screens,
        ),
        bottomNavigationBar: Theme(
          data: Theme.of(context).copyWith(
            canvasColor: Colors.black,
          ),
          child: BottomNavigationBar(
            currentIndex: _selectedIndex,
            onTap: _onItemTapped,

BottomNavigationBarItem(
                icon: Padding(
                  padding:
EdgeInsets.only(bottom: 4),
                  child:
Icon(Icons.library_music),
                ),
```

```
              color: Colors.white,
            ),
            unselectedIconTheme: const
IconThemeData(
              size: 24,
              color: Colors.grey,
            ),
            selectedLabelStyle: const
TextStyle(
              fontSize: 12,
              fontWeight: FontWeight.bold,
            ),
            unselectedLabelStyle: const
TextStyle(
              fontSize: 12,
            ),
            items: const [
              BottomNavigationBarItem(
                icon: Padding(
                  padding:
EdgeInsets.only(bottom: 4),
                  child: Icon(Icons.home_filled),
                ),
                label: 'Home',
              ),
              BottomNavigationBarItem(
                icon: Padding(
                  padding:
EdgeInsets.only(bottom: 4),
                  child: Icon(Icons.search),
                ),
                label: 'Search',
              ),
```

Sign_in.dart

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

import 'register.dart';

class SignInScreen extends StatefulWidget
```

```
        label: 'Library',
       ),
       BottomNavigationBarItem(
        icon: Padding(
         padding:
EdgeInsets.only(bottom: 4),
         child: Icon(Icons.person),
        ),
        label: 'Profile',
       ),
      ],
     ),
    ),
   ),
  );
 }
}
```

```
Navigator.pop(context),
          ),
          const SizedBox(width: 12),
          Image.asset(

'assets/images/spotify_logo.png',
           height: 150,
          ),
```

```
{
 const SignInScreen({Key? key}) :
super(key: key);

 @override
 State<SignInScreen> createState() =>
_SignInScreenState();
}

class _SignInScreenState extends
State<SignInScreen> {
 bool _obscurePassword = true;

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   backgroundColor: const
Color(0xFF1C1B1B),
   body: SafeArea(
    child: SingleChildScrollView(
     child: Padding(
      padding: const
EdgeInsets.all(24.0),
      child: Column(
       crossAxisAlignment:
CrossAxisAlignment.start,
       children: [
        Row(
         children: [
          IconButton(
           icon: const
Icon(Icons.arrow_back, color:
Colors.white),
           onPressed: () =>
       hintText: 'Enter Username Or
Email',
          hintStyle: const
TextStyle(color: Colors.grey),
          filled: true,
          fillColor:
Colors.grey.withOpacity(0.1),
          border: OutlineInputBorder(
                  ],
                ),
                const SizedBox(height: 32),
                const Text(
                  'Sign In',
                  style: TextStyle(
                    color: Colors.white,
                    fontSize: 28,
                    fontWeight: FontWeight.bold,
                  ),
                ),
                const SizedBox(height: 8),
                Row(
                  children: [
                    const Text(
                      'If You Need Any Support ',
                      style: TextStyle(color:
Colors.grey),
                    ),
                    TextButton(
                      onPressed: () {},
                      child: const Text(
                        'Click Here',
                        style: TextStyle(color:
Colors.green),
                      ),
                    ),
                  ],
                ),
                const SizedBox(height: 32),
                TextField(
                  style: const TextStyle(color:
Colors.white),
                  decoration: InputDecoration(
 setState(() {
                          _obscurePassword =
!_obscurePassword;
                        });
                      },
                    ),
                  ),
                ),
```

```
                      borderRadius:
BorderRadius.circular(12),
                      borderSide:
BorderSide.none,
                    ),
                  ),
                ),
                const SizedBox(height: 16),
                TextField(
                  obscureText:
_obscurePassword,
                  style: const TextStyle(color:
Colors.white),
                  decoration: InputDecoration(
                    hintText: 'Password',
                    hintStyle: const
TextStyle(color: Colors.grey),
                    filled: true,
                    fillColor:
Colors.grey.withOpacity(0.1),
                    border: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                      borderSide:
BorderSide.none,
                    ),
                    suffixIcon: IconButton(
                      icon: Icon(
                        _obscurePassword
                            ? Icons.visibility_off
                            : Icons.visibility,
                        color: Colors.grey,
                      ),
                      onPressed: () {
                                            ),
                    ),
                    child: const Text(
                      'Sign In',
                      style: TextStyle(
                        fontSize: 16,
                        fontWeight:
FontWeight.bold,
                const SizedBox(height: 16),
                Align(
                  alignment:
Alignment.centerRight,
                  child: TextButton(
                    onPressed: () {},
                    child: const Text(
                      'Recovery Password',
                      style: TextStyle(color:
Colors.grey),
                    ),
                  ),
                ),
                const SizedBox(height: 32),
                SizedBox(
                  width: double.infinity,
                  height: 56,
                  child: ElevatedButton(
                    onPressed: () {
                      Navigator.push(
                        context,
                        MaterialPageRoute(
                            builder: (context) =>
const MainScreen()),
                      );
                    },
                    style:
ElevatedButton.styleFrom(
                      backgroundColor: const
Color(0xFF42C83C),
                      shape:
RoundedRectangleBorder(
                        borderRadius:
BorderRadius.circular(30),
Colors.grey),
                    ),
                    TextButton(
                      onPressed: () {
                        Navigator.push(
                          context,
                          MaterialPageRoute(
                              builder: (context) =>
                      ),
                    ),
                  ),
                ),
                const SizedBox(height: 16),
                const Center(
                  child: Text(
                    'Or',
                    style: TextStyle(color:
Colors.grey),
                  ),
                ),
                const SizedBox(height: 16),
                Row(
                  mainAxisAlignment:
MainAxisAlignment.center,
                  children: [

Image.asset('assets/images/google_logo.pn
g', height: 40),
                    const SizedBox(width: 20),

Image.asset('assets/images/apple_logo.png
', height: 40),
                  ],
                ),
                const SizedBox(height: 32),
                Row(
                  mainAxisAlignment:
MainAxisAlignment.center,
                  children: [
                    const Text(
                      'Not A Member? ',
                      style: TextStyle(color:
```

Register_screen.dart

```
import 'package:flutter/material.dart';
import
'package:spotify_clone/features/auth/scree
ns/sign_in_screen.dart';
import
'package:spotify_clone/features/auth/scree
ns/register.dart';
```

```
const Register()),
            );
           },
           child: const Text(
            'Register Now',
            style: TextStyle(color:
Colors.blue),
           ),
          ),
         ],
        ),
       ],
      ),
     ),
    ),
   ),
  );
 }
}
```

```
 child: Column(
          crossAxisAlignment:
CrossAxisAlignment.center,
          children: [
           Image.asset(

'assets/images/spotify_logo.png',
            width: //235
```

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

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   backgroundColor: const
Color(0xFF1C1B1B),
   body: SafeArea(
    child: Column(
     children: [
      Padding(
       padding: const
EdgeInsets.all(16.0),
       child: Row(
        children: [
         IconButton(
          icon: const
Icon(Icons.arrow_back, color:
Colors.white),
          onPressed: () =>
Navigator.pop(context),
         ),
        ],
       ),
      ),
      Expanded(
       child: SingleChildScrollView(
        child: Padding(
         padding: const
EdgeInsets.symmetric(horizontal: 24.0),
                         builder:
(context) =>
                   Register()), //
Navigate to ThirdScreen
              );
              // Handle register
             },
             style:
              500,
            height: 150,
           ),
           const SizedBox(height: 32),
           const Text(
            'Enjoy Listening To Music',
            style: TextStyle(
             color: Colors.white,
             fontSize: 24,
             fontWeight:
FontWeight.bold,
            ),
           ),
           const SizedBox(height: 16),
           const Text(
            'Spotify is a proprietary
Swedish audio\nstreaming and media
services provider',
            textAlign:
TextAlign.center,
            style: TextStyle(
             color: Colors.grey,
             fontSize: 16,
            ),
           ),
           const SizedBox(height: 32),
           SizedBox(
            width: double.infinity,
            height: 56,
            child: ElevatedButton(
             onPressed: () {
              Navigator.push(
               context,
               MaterialPageRoute(
style: TextStyle(
              color: Colors.white,
              fontSize: 16,
             ),
            ),
           ),
          ],
         ),
```

```
ElevatedButton.styleFrom(
              backgroundColor: const
Color(0xFF42C83C),
              shape:
RoundedRectangleBorder(
               borderRadius:
BorderRadius.circular(30),
              ),
             ),
             child: const Text(
              'Register',
              style: TextStyle(
               color: Colors.white,
               fontSize: 16,
               fontWeight:
FontWeight.bold,
              ),
             ),
            ),
           ),
           const SizedBox(height: 16),
           TextButton(
            onPressed: () {
             Navigator.push(
               context,
               MaterialPageRoute(
                 builder: (context) =>
                   SignInScreen()), //
Navigate to ThirdScreen
              );
             // Handle sign in
            },
            child: const Text(
             'Sign in',
```

Home_screen.dart

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

// First, add this typography class at the
top of your file
```

```
              ),
            ),
          ),
        ],
      ),
    ),
  );
 }
}
```

```
body: Stack(
      children: [
        Container(
          decoration: BoxDecoration(
            gradient: LinearGradient(
              begin: Alignment.topCenter,
              end: Alignment.bottomCenter,
              colors: [
```

```
class AppTypography {
 static final TextStyle displayLarge =
GoogleFonts.poppins(
   fontSize: 24,
   fontWeight: FontWeight.bold,
   letterSpacing: -0.3,
   color: Colors.white,
 );

 static final TextStyle titleMedium =
GoogleFonts.poppins(
   fontSize: 16,
   fontWeight: FontWeight.w600,
   letterSpacing: -0.2,
   color: Colors.white,
 );

 static final TextStyle bodySmall =
GoogleFonts.poppins(
   fontSize: 12,
   fontWeight: FontWeight.normal,
   letterSpacing: 0.1,
   color: Colors.grey[400],
 );
}

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

 @override
 Widget build(BuildContext context) {
   return Scaffold(
    8),
                    decoration:
BoxDecoration(
                      color: const
Color(0xFF1DB954),
                      borderRadius:
BorderRadius.circular(20),
                    ),
                  Colors.black.withOpacity(0.8),
                  Colors.black,
                ],
              ),
            ),
          ),
          SingleChildScrollView(
            child: SafeArea(
              child: Column(
                crossAxisAlignment:
CrossAxisAlignment.start,
                children: [
                  Padding(
                    padding: const
EdgeInsets.all(16.0),
                    child: Row(
                      children: [
                        // Updated CircleAvatar
with specific size and styling
                        const CircleAvatar(
                          radius: 20,
                          backgroundColor:
Color(0xFF282828),
                          child: Icon(
                            Icons.person,
                            color: Colors.white,
                            size: 24,
                          ),
                        ),
                        const SizedBox(width: 16),
                        Container(
                          padding: const
EdgeInsets.symmetric(
                              horizontal: 16, vertical:
 child: const Text('Podcasts',
                              style: TextStyle(color:
Colors.white)),
                        ),
                      ],
                    ),
                  ),
                  const PlaylistSection(
```

```
                          child: const Text('All',
                              style: TextStyle(color:
Colors.white)),
                        ),
                        const SizedBox(width: 8),
                        Container(
                          padding: const
EdgeInsets.symmetric(
                              horizontal: 16, vertical:
8),
                          decoration:
BoxDecoration(
                            color: const
Color(0xFF282828),
                            borderRadius:
BorderRadius.circular(20),
                          ),
                          child: const Text('Music',
                              style: TextStyle(color:
Colors.white)),
                        ),
                        const SizedBox(width: 8),
                        Container(
                          padding: const
EdgeInsets.symmetric(
                              horizontal: 16, vertical:
8),
                          decoration:
BoxDecoration(
                            color: const
Color(0xFF282828),
                            borderRadius:
BorderRadius.circular(20),
                          ),

                        SizedBox(width: 16),
                        MixCard(
                          title: 'Chill Mix',
                          subtitle: 'Kendrick Lamar,
The Weeknd, Future',
                          image:
'assets/images/chill.jpg',
                    title: 'On Repeat',
                    image:
'assets/images/on_repeat.jpg',
                    isLarge: true,
                  ),
                  const PlaylistSection(
                    title: 'Trending Now India',
                    image:
'assets/images/trending_india.jpg',
                    isLarge: true,
                  ),
                  const PlaylistSection(
                    title: 'Liked Songs',
                    image:
'assets/images/liked_songs.jpg',
                    isLarge: true,
                  ),
                  const SectionTitle(title: 'Your
top mixes'),
                  SingleChildScrollView(
                    scrollDirection:
Axis.horizontal,
                    padding: const
EdgeInsets.symmetric(horizontal: 16),
                    child: Row(
                      children: const [
                        MixCard(
                          title: 'Arijit Singh Mix',
                          subtitle: 'Pritam,
Vishal-Shekhar and Atif Aslam',
                          image:
'assets/images/arijit.jpg',
                        ),

const PlaylistSection({
    Key? key,
    required this.title,
    required this.image,
    this.isLarge = false,
  }) : super(key: key);

  @override
```

```
                        ),
                        SizedBox(width: 16),
                        MixCard(
                          title: 'R&B Mix',
                          subtitle: 'Kali Uchis,
SZA',
                          image:
'assets/images/rnb.jpg',
                        ),
                      ],
                    ),
                  ),
                  const SizedBox(height: 100),
                ],
              ),
            ),
          ),
          const Positioned(
            left: 0,
            right: 0,
            bottom: 0,
            child: NowPlayingBar(),
          ),
        ],
      ),
    );
  }
}

class PlaylistSection extends
StatelessWidget {
  final String title;
  final String image;
  final bool isLarge;

                  size: 24,
              ),
              onPressed: () {},
            ),
          ),
        ],
      ),
 Widget build(BuildContext context) {
  return Padding(
   padding: const EdgeInsets.all(16.0),
   child: Row(
    children: [
     ClipRRect(
      borderRadius:
BorderRadius.circular(4),
      child: Image.asset(
       image,
       width: isLarge ? 80 : 60,
       height: isLarge ? 80 : 60,
       fit: BoxFit.cover,
      ),
     ),
     const SizedBox(width: 16),
     Expanded(
      child: Text(
       title,
       style:
AppTypography.titleMedium, // Updated
typography
       overflow: TextOverflow.ellipsis,
// Added overflow handling
       maxLines: 2,
      ),
     ),
     Material(
      color: Colors.transparent,
      child: IconButton(
       icon: const Icon(
        Icons.more_vert,
        color: Colors.white,

const SizedBox(height: 8),
     Text(
      title,
      style:
AppTypography.titleMedium, // Updated
typography
      overflow: TextOverflow.ellipsis, //
Added overflow handling
```

```
  );
 }
}

class MixCard extends StatelessWidget {
 final String title;
 final String subtitle;
 final String image;

 const MixCard({
  Key? key,
  required this.title,
  required this.subtitle,
  required this.image,
 }) : super(key: key);

 @override
 Widget build(BuildContext context) {
  return SizedBox(
   width: 160,
   child: Column(
    crossAxisAlignment:
CrossAxisAlignment.start,
    children: [
     ClipRRect(
      borderRadius:
BorderRadius.circular(4),
       child: Image.asset(
        image,
        width: 160,
        height: 160,
        fit: BoxFit.cover,
      ),
     ),
            width: 40,
       height: 40,
       fit: BoxFit.cover,
      ),
     ),
     const SizedBox(width: 12),
     Expanded(
      child: Column(
            maxLines: 1,
          ),
          const SizedBox(height: 4),
          Text(
            subtitle,
            style: AppTypography.bodySmall,
// Updated typography
            overflow: TextOverflow.ellipsis, //
Added overflow handling
            maxLines: 2,
          ),
        ],
      ),
    );
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Container(
      color: const Color(0xFF282828),
      padding: const EdgeInsets.all(8),
      child: Row(
        children: [
          ClipRRect(
            borderRadius:
BorderRadius.circular(4),
            child: Image.asset(
              'assets/images/perfect_1d.jpg',
 ),
          Material(
            color: Colors.transparent,
            child: IconButton(
              icon: const Icon(
                Icons.favorite_border,
                color: Colors.white,
                size: 24,
              crossAxisAlignment:
CrossAxisAlignment.start,
              mainAxisSize:
MainAxisSize.min,
              children: [
                Text(
                  'Perfect',
                  style:
AppTypography.titleMedium, // Updated
typography
                  overflow:
TextOverflow.ellipsis,
                ),
                Text(
                  'One Direction',
                  style:
AppTypography.bodySmall, // Updated
typography
                  overflow:
TextOverflow.ellipsis,
                ),
              ],
            ),
          ),
          Material(
            color: Colors.transparent,
            child: IconButton(
              icon: const Icon(
                Icons.devices_outlined,
                color: Colors.white,
                size: 24,
              ),
              onPressed: () {},
            ),
                 style:
AppTypography.displayLarge, // Updated
typography
      ),
    );
  }
}
```

```
              ),
              onPressed: () {},
            ),
          ),
          Material(
            color: Colors.transparent,
            child: IconButton(
              iconSize: 32,
              icon: const Icon(
                Icons.play_circle_filled,
                color: Colors.white,
              ),
              onPressed: () {},
            ),
          ),
        ],
      ),
    );
  }
}

class SectionTitle extends StatelessWidget
{
  final String title;

  const SectionTitle({Key? key, required
this.title}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(16.0),
      child: Text(
        Title,
```

Search_screen.dart

```
import 'package:flutter/material.dart';
import
'package:flutter_riverpod/flutter_riverpod.
dart';
import
'package:spotify_clone/models/search_res
ult.dart';
```

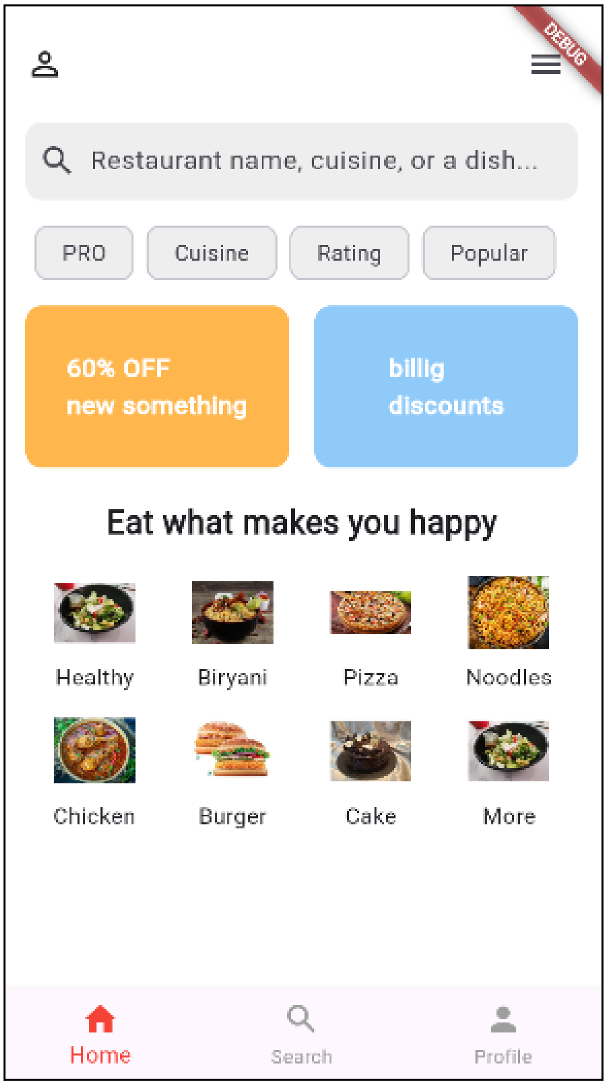
```
Future<void> _performSearch(String
query) async {
    if (query.isEmpty) return;

    setState(() {
      _isLoading = true;
    });
```

```
import
'package:spotify_clone/services/youtube_s
ervice.dart';
import
'package:spotify_clone/providers/audio_pr
ovider.dart';
import
'package:spotify_clone/services/youtube_e
xplode_service.dart';

final youtubeServiceProvider =
Provider((ref) => YouTubeService());
final audioPlayerProvider =
StateNotifierProvider<AudioPlayerNotifie
r, AudioPlayerState>((ref) =>
AudioPlayerNotifier());

class SearchScreen extends
ConsumerStatefulWidget {
  const SearchScreen({Key? key}) :
super(key: key);

  @override
  ConsumerState<SearchScreen>
createState() => _SearchScreenState();
}

class _SearchScreenState extends
ConsumerState<SearchScreen> {
  final TextEditingController
_searchController =
TextEditingController();
  List<SearchResult> _searchResults = [];
  bool _isLoading = false;

        body: SafeArea(
        child: Column(
          children: [
            Padding(
              padding: const
EdgeInsets.all(16.0),
              child: TextField(
```

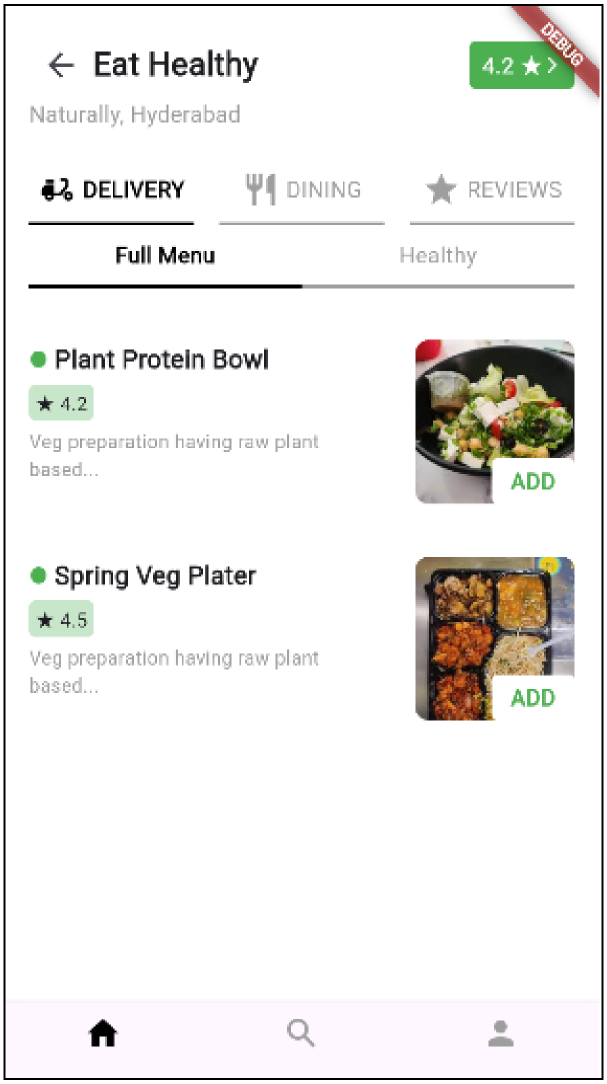
```
    final youtubeService =
ref.read(youtubeServiceProvider);
    final results = await
youtubeService.searchSongs(query);

    setState(() {
      _searchResults = results;
      _isLoading = false;
    });
  }

  Future<void> _playSong(SearchResult
song) async {
    final youtubeService =
ref.read(youtubeServiceProvider);
    final audioUrl = await
youtubeService.getAudioUrl(song.id);

    if (audioUrl != null) {
      await
ref.read(audioPlayerProvider.notifier).play
Song(audioUrl, song);
    }
  }

  @override
  Widget build(BuildContext context) {
    final audioState =
ref.watch(audioPlayerProvider);

    return Scaffold(
      backgroundColor: const
Color(0xFF1C1B1B),

_searchResults[index];
                    final isPlaying =
audioState.currentSong?.id == song.id;

                    return ListTile(
                      leading:
Image.network(song.thumbnail),
                      title: Text(
                controller: _searchController,
                style: const TextStyle(color:
Colors.white),
                decoration: InputDecoration(
                  hintText: 'Search for songs...',
                  hintStyle: const
TextStyle(color: Colors.grey),
                  filled: true,
                  fillColor:
Colors.grey.withOpacity(0.1),
                  border: OutlineInputBorder(
                    borderRadius:
BorderRadius.circular(12),
                    borderSide: BorderSide.none,
                  ),
                  suffixIcon: IconButton(
                    icon: const Icon(Icons.search,
color: Colors.grey),
                    onPressed: () =>
_performSearch(_searchController.text),
                  ),
                ),
                onSubmitted: _performSearch,
              ),
            ),
            if (_isLoading)
              const CircularProgressIndicator()
            else
              Expanded(
                child: ListView.builder(
                  itemCount:
_searchResults.length,
                  itemBuilder: (context, index) {
                    final song =
EdgeInsets.all(16),
                color: Colors.black,
                child: Row(
                  children: [
                    Image.network(

audioState.currentSong!.thumbnail,
                      height: 50,
                        song.title,
                        style: const
TextStyle(color: Colors.white),
                      ),
                      subtitle: Text(
                        song.channelTitle,
                        style: const
TextStyle(color: Colors.grey),
                      ),
                      trailing: IconButton(
                        icon: Icon(
                          isPlaying &&
audioState.isPlaying
                              ? Icons.pause
                              : Icons.play_arrow,
                          color: Colors.white,
                        ),
                        onPressed: () {
                          if (isPlaying) {

ref.read(audioPlayerProvider.notifier).togg
lePlay();
                          } else {
                            _playSong(song);
                          }
                        },
                      ),
                    );
                  },
                ),
              ),
            if (audioState.currentSong != null)
              Container(
                padding: const
lePlay();
                      },
                    ),
                  ],
                ),
              ),
          ],
        ),
                    ),
                    const SizedBox(width: 16),
                    Expanded(
                      child: Column(
                        crossAxisAlignment:
CrossAxisAlignment.start,
                        children: [
                          Text(

audioState.currentSong!.title,
                            style: const
TextStyle(color: Colors.white),
                            maxLines: 1,
                            overflow:
TextOverflow.ellipsis,
                          ),
                          Text(

audioState.currentSong!.channelTitle,
                            style: const
TextStyle(color: Colors.grey),
                          ),
                        ],
                      ),
                    ),
                    IconButton(
                      icon: Icon(
                        audioState.isPlaying ?
Icons.pause : Icons.play_arrow,
                        color: Colors.white,
                      ),
                      onPressed: () {

ref.read(audioPlayerProvider.notifier).togg
```

```
      ),
    );
  }
}
```

OUTPUT:



DEBUG
Restaurant name, cuisine, or a dish...
PRO
Cuisine
Rating
Popular
60% OFF
new something
billig
discounts
Eat what makes you happy
Healthy
Biryani
Pizza
Noodles
Chicken
Burger
Cake
More
Home
Search
Profile



← Eat Healthy

4.2 ★ >

DEBUG

Naturally, Hyderabad

DELIVERY | DINING | REVIEWS

Full Menu | Healthy

● Plant Protein Bowl

★ 4.2

Veg preparation having raw plant based...

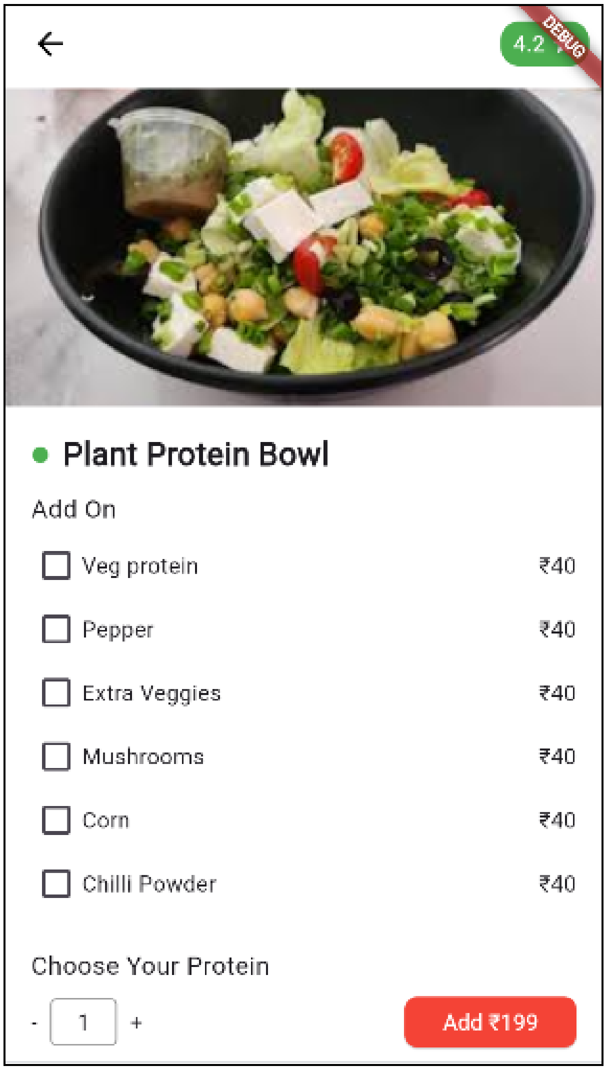
ADD

● Spring Veg Plater

★ 4.5

Veg preparation having raw plant based...

ADD



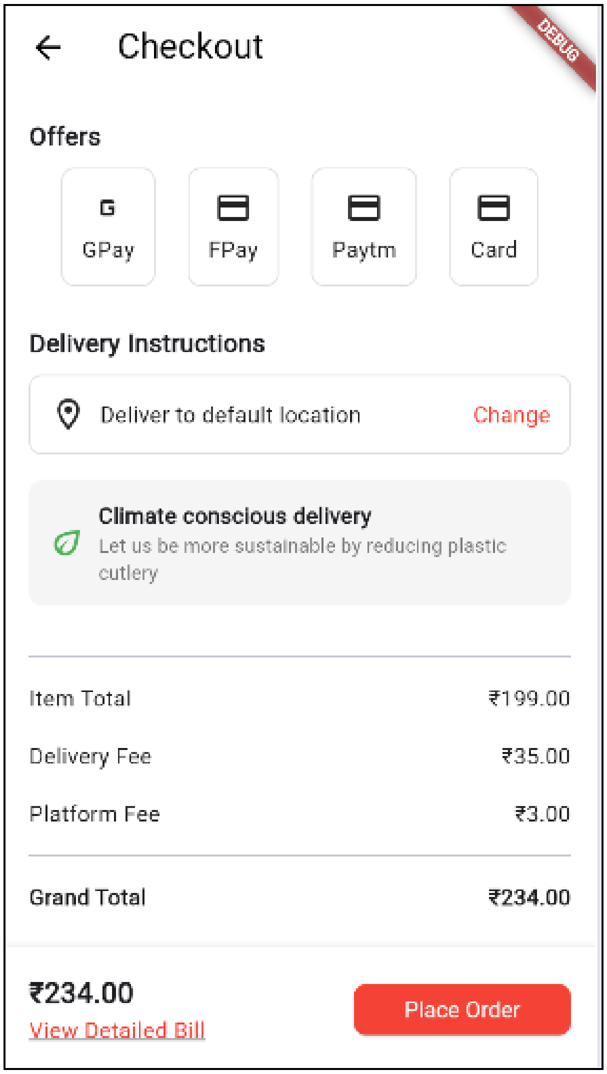
● Plant Protein Bowl

Add On

□ Veg protein	₹40
□ Pepper	₹40
□ Extra Veggies	₹40
□ Mushrooms	₹40
□ Corn	₹40
□ Chilli Powder	₹40

Choose Your Protein

- 1 +



Checkout

Offers

Delivery Instructions

Deliver to default location

Climate conscious delivery

Let us be more sustainable by reducing plastic cutlery

Item Total	₹199.00
Delivery Fee	₹35.00
Platform Fee	₹3.00
Grand Total	**₹234.00**

₹234.00

View Detailed Bill