

MPL EXPERIMENT 2

NAME-ARYAN ANIL PATANKAR

CLASS-D15A

ROLL NO-33

AIM:To design Flutter UI by including common widgets

1,Introduction

Flutter is an open-source UI framework by Google that enables developers to build cross-platform applications with a single codebase. The UI in Flutter is built using **widgets**, which are the fundamental building blocks of the application.

Widgets in Flutter can be classified into various categories, such as structural widgets, layout widgets, interactive widgets, and styling widgets. These widgets help in designing complex UI components efficiently and responsively.

2.Types of widgets

Flutter widgets are classified into the following types based on their functionality:

1. Structural Widgets (Scaffolding Widgets)

These widgets define the basic structure of an application.

- **Scaffold:** Provides a framework with an app bar, body, floating action button, drawer, etc.
- **AppBar:** Displays the top navigation bar with titles, actions, and menus.
- **Drawer:** A side navigation panel for navigating through different screens.

2. Layout Widgets

These widgets help in arranging other widgets on the screen.

- **Container:** A flexible widget for holding and styling child widgets.
- **Column:** Arranges widgets vertically.
- **Row:** Arranges widgets horizontally.
- **Stack:** Allows widgets to be placed on top of each other.
- **Expanded:** Expands widgets to fill available space.

- **SizedBox:** Provides spacing between widgets.

3. Interactive Widgets

These widgets accept user input and interactions.

- **TextField:** Allows users to enter text input.
- **ElevatedButton:** A clickable button with elevation.
- **OutlinedButton:** A button with an outlined border.
- **IconButton:** A clickable icon button.
- **GestureDetector:** Detects gestures like tap, swipe, and drag.
- **Switch:** A toggle button for enabling/disabling options.

4. Styling & Display Widgets

These widgets help in styling and displaying content.

- **Text:** Displays text content.
- **Image:** Displays an image from assets or a URL.
- **Icon:** Displays an icon from Material or custom icons.
- **Card:** A box with elevation and rounded corners for content display.
- **Chip:** A small, compact element that represents information.

5. Scrolling & List Widgets

These widgets handle large lists and scrolling content.

- **ListView:** Displays a scrollable list of items.
- **GridView:** Displays items in a grid format.
- **SingleChildScrollView:** Makes a child widget scrollable.
- **PageView:** Creates swipeable pages.

3.Commonly used widgets:

1. Scaffold (Basic Page Structure)

A *Scaffold* provides a basic structure for a screen, including an **AppBar, Floating Action Button, Drawer, and Body**.

2.Container (A Flexible Box for UI Design)

A *Container* is used for styling, padding, and positioning elements.

3.Row and Column (Arranging Widgets)

A *Row* arranges widgets horizontally, while a *Column* arranges them vertically.

4.Stack (Overlapping Widgets)

A *Stack* places widgets on top of each other.

5.ListView (Scrollable List)

A *ListView* is used for displaying a scrollable list of items.

6.ElevatedButton (Clickable Button)

Buttons provide user interactions in Flutter.

7.TextField (User Input Field)

A *TextField* allows users to enter data.

CODE

Genre_card.dart

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

class GenreCard extends StatelessWidget {

  final String title;

  final Color color;

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

  @override
  Widget build(BuildContext context) {
    return Container(
      decoration: BoxDecoration(
        color: color,
        borderRadius:
          BorderRadius.circular(4),
      ),
      child: Center(
        child: Text(
          title,
          style: const TextStyle(
            fontSize: 16,
            fontWeight: FontWeight.bold,
          ),
        ),
      ),
    );
  }
}
```

```
Widget build(BuildContext context) {
  return Container(
    decoration: BoxDecoration(
      color: color,
      width: 48,
      height: 48,
```

Now_playing_bar.dart

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

class NowPlayingBar extends
StatelessWidget {

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

  @override
Widget build(BuildContext context) {

  return Container(

    color: Colors.grey[900],

    padding: const EdgeInsets.all(8),

    child: SafeArea(

      top: false,

      child: Row(

        children: [

          ClipRRect(

            borderRadius:
BorderRadius.circular(4),

            child: Image.asset(

              'assets/album_cover.jpg',
```

```
fit: BoxFit.cover,

            ),

          ),

          const SizedBox(width: 12),

          const Expanded(

            child: Column(

              crossAxisAlignment:
CrossAxisAlignment.start,

              mainAxisAlignment:
MainAxisSize.min,

              children: [

                Text(

                  'Perfect',

                  style:
TextStyle(fontWeight:
FontWeight.bold),

                ),

                Text(

                  'One Direction',
```

Playlist_card.dart

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

```

        style: TextStyle(color:
Colors.grey, fontSize: 12),

    ),

  ],

),

IconButton(

  icon: const
Icon(Icons.play_arrow),

  onPressed: () {},

),

],

),

),

);

}

}

```

```

child: Row(

  children: [

```

```

class PlaylistCard extends
StatelessWidget {

  final String title;

  final String saves;

  const PlaylistCard({

    Key? key,

    required this.title,

    required this.saves,

  }) : super(key: key);

  @override

  Widget build(BuildContext context) {

    return Container(

      padding: const EdgeInsets.all(16),

      decoration: BoxDecoration(

        color: Colors.grey[900],

        borderRadius:

BorderRadius.circular(4),

      ),

      style: const TextStyle(fontWeight:

FontWeight.bold),

    ),

```

```

Container(
  width: 48,
  height: 48,
  decoration: BoxDecoration(
    color: Colors.grey[800],
    borderRadius:
BorderRadius.circular(4),
  ),
  child: const
Icon(Icons.music_note),
),
const SizedBox(width: 16),
Expanded(
  child: Column(
    crossAxisAlignment:
CrossAxisAlignment.start,
    children: [

```

```

      Text(
        title,

```

```

      Text(
        saves,
        style: const TextStyle(color:
Colors.grey),
      ),
    ],
  ),
),
],
),
);
}
}

```

```

padding: const EdgeInsets.all(12),
decoration: BoxDecoration(
  border: Border.all(color:
Colors.grey[300]!),

```

Social_login.dart

```
//
lib/features/auth/widgets/social_login_
button.dart

import 'package:flutter/material.dart';

class SocialLoginButton extends
StatelessWidget {

  final String assetPath;

  final VoidCallback onTap;

  const SocialLoginButton({
    super.key,
    required this.assetPath,
    required this.onTap,
  });

  @override
  Widget build(BuildContext context) {
    return InkWell(
      onTap: onTap,
      child: Container(
        borderRadius:
        BorderRadius.circular(8),
      ),
      child: Image.asset(
        assetPath,
        height: 24,
        width: 24,
      ),
    ),
  );
}

bodyLarge: TextStyle(color:
Colors.white),

bodyMedium: TextStyle(color:
Colors.white),
```


Home_screen.dart

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

```
void main() {
```

```
  runApp(const MyApp());
```

```
}
```

```
class MyApp extends StatelessWidget
```

```
{
```

```
  const MyApp({Key? key}) :
```

```
  super(key: key);
```

```
  @override
```

```
  Widget build(BuildContext context) {
```

```
    return MaterialApp(
```

```
      title: 'Spotify Clone',
```

```
      theme: ThemeData(
```

```
        scaffoldBackgroundColor:  
Colors.black,
```

```
        brightness: Brightness.dark,
```

```
        textTheme: const TextTheme(
```

```
          titleLarge: TextStyle(color:  
Colors.white),
```

```
          titleMedium: TextStyle(color:  
Colors.white),
```

```
        ),
```

```
      ),
```

```
      home: const HomeScreen(),
```

```
    );
```

```
  }
```

```
}
```

```
class HomeScreen extends  
StatelessWidget {
```

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

```
  @override
```

```
  Widget build(BuildContext context) {
```

```
    return Scaffold(
```

```
      children: [
```

```
        Padding(
```

```

body: Stack(
  children: [
    Container(
      decoration: BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end:
Alignment.bottomCenter,
          colors: [
Colors.black.withOpacity(0.8),
          Colors.black,
        ],
      ),
    ),
  ),
  SingleChildScrollView(
    child: SafeArea(
      child: Column(
        crossAxisAlignment:
CrossAxisAlignment.start,
        padding: const
EdgeInsets.all(16.0),
        child: Row(
          children: [
            const CircleAvatar(
              backgroundColor:
Color(0xFF282828),
              child: Text('A', style:
TextStyle(color: Colors.white)),
            ),
            const SizedBox(width:
16),
            Container(
              padding: const
EdgeInsets.symmetric(horizontal: 16,
vertical: 8),
              decoration:
BoxDecoration(
                color: const
Color(0xFF1DB954),
                borderRadius:
BorderRadius.circular(20),
              ),
            Container(

```

```

        ),
        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),

```

```

        padding: const
EdgeInsets.symmetric(horizontal: 16,
vertical: 8),
        decoration:
BoxDecoration(
            color: const
Color(0xFF282828),
            borderRadius:
BorderRadius.circular(20),
        ),
        child: const
Text('Podcasts', style: TextStyle(color:
Colors.white)),
    ),
],
),
const PlaylistSection(
    title: 'On Repeat',
    image:
'assets/images/on_repeat.jpg',
    isLarge: true,
    children: const [
        MixCard(

```

```

    ),
    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(
        ),
        title: 'Arijit Singh Mix',
        subtitle: 'Pritam,
        Vishal-Shekhar and Atif Aslam',
        image:
        'assets/images/arijit.jpg',
      ),
      SizedBox(width: 16),
      MixCard(
        title: 'Chill Mix',
        subtitle: 'Kendrick
        Lamar, The Weeknd, Future',
        image:
        'assets/images/chill.jpg',
      ),
      SizedBox(width: 16),
      MixCard(
        title: 'R&B Mix',
        subtitle: 'Kali Uchis,
        SZA',
        image:
        'assets/images/rnb.jpg',
      type: BottomNavigationBarType.fixed,
      selectedItemColor: Colors.white,

```

```

        ],
      ),
    ),
    const SizedBox(height:
100), // Space for bottom player
    ],
  ),
),
const Positioned(
  left: 0,
  right: 0,
  bottom: 0,
  child: NowPlayingBar(),
),
],
),
bottomNavigationBar:
BottomNavigationBar(
  backgroundColor: Colors.black,
    ],

```

```

unselectedItemColor: Colors.grey,
items: const [
  BottomNavigationBarItem(
    icon: Icon(Icons.home_filled),
    label: 'Home',
  ),
  BottomNavigationBarItem(
    icon: Icon(Icons.search),
    label: 'Search',
  ),
  BottomNavigationBarItem(
    icon: Icon(Icons.library_music),
    label: 'Your Library',
  ),
  BottomNavigationBarItem(
    icon: Icon(Icons.person),
    label: 'Profile',
  ),
return Padding(
  padding: const EdgeInsets.all(16.0),

```

```

    ),
  );
}
}

```

```

class PlaylistSection extends
StatelessWidget {

```

```

  final String title;

  final String image;

  final bool isLarge;

```

```

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

```

```

  @override

```

```

  Widget build(BuildContext context) {

```

```

    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: const TextStyle(
              fontSize: 16,
            ),
            required this.title,
            required this.subtitle,

```



```

    },
  ),
const SizedBox(height: 8),
Text(
  title,
  style: const TextStyle(
    fontWeight: FontWeight.w600,
    color: Colors.white,
  ),
),
const SizedBox(height: 4),
Text(
  subtitle,
  style: const TextStyle(
    color: Colors.grey,
    fontSize: 12,
  ),
),
],

    },
  },
),
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: [
          Image.asset(
            'assets/images/perfect_1d.jpg',
            width: 40,
            height: 40,
          ),

```



```

    ),
    const SizedBox(width: 12),
    const Expanded(
      child: Column(
        crossAxisAlignment:
CrossAxisAlignment.start,
        mainAxisAlignment:
MainAxisSize.min,
        children: [
          Text(
            'Perfect',
            style: TextStyle(
              fontWeight:
FontWeight.w600,
              color: Colors.white,
            ),
          ),
          Text(
            'One Direction',
            style: TextStyle(
              color: Colors.grey,
              fontSize: 12,

```

```

    ],
    ),
    ),
    IconButton(
      icon: const
Icon(Icons.devices_outlined, color:
Colors.white),
      onPressed: () {},
    ),
    IconButton(
      icon: const
Icon(Icons.favorite_border, color:
Colors.white),
      onPressed: () {},
    ),
    IconButton(
      icon: const
Icon(Icons.play_circle_filled, color:
Colors.white),
      onPressed: () {},
    ),
    fontWeight: FontWeight.bold,
    color: Colors.white,

```

```

        ],
    ),
),
);
}
}

```

```

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,

```

```

            style: const TextStyle(

```

```

                fontSize: 24,

```

```

            child: Column(

```

```

                crossAxisAlignment:
CrossAxisAlignment.start,

```

Profile_page.dart

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

class ProfileScreen extends
StatelessWidget {

  const ProfileScreen({Key? key}) :
super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.black,
      body: Column(
        children: [
          const Expanded(flex: 2, child:
_TopPortion()),
          Expanded(
            flex: 3,
            child: Padding(
              padding: const
EdgeInsets.all(16.0),
```

```
        children: [
          Text(
            "Aryan",
            style:
Theme.of(context).textTheme.headline
Small?.copyWith(
              color: Colors.white,
              fontWeight:
FontWeight.bold,
            ),
          ),
          const SizedBox(height: 8),
          const _ProfileInfoRow(),
          const SizedBox(height: 24),
          Text(
            "Playlists",
            style:
Theme.of(context).textTheme.titleLarg
e?.copyWith(
              color: Colors.white,
            );
        ]
      )
    );
  }
}
```

```

fontWeight:      }
FontWeight.bold,

    ),
    ),
const SizedBox(height: 16),
const _PlaylistCard(
    title: "My playlist #3",
    saves: "0 saves",
),
const SizedBox(height: 12),
const _PlaylistCard(
    title: "CHILL",
    saves: "0 saves",
),
],
),
),
),
const _BottomNavBar(),
],
),

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

@override
Widget build(BuildContext context) {
    return Padding(
        padding: const
EdgeInsets.symmetric(vertical: 8.0),
        child: Row(
            children: [
                Text(
                    "1 follower",
                    style:
Theme.of(context).textTheme.bodyMe
dium?.copyWith(
                        color: Colors.grey,
                    ),
            ],
        ),
    ),
const _PlaylistCard({
    Key? key,

```

```

        ),
    ),
    const Text(" • ", style:
TextStyle(color: Colors.grey)),
    Text(
        "19 following",
        style:
Theme.of(context).textTheme.bodyMe
dium?.copyWith(
            color: Colors.grey,
        ),
    ),
],
),
);
}
}

```

```

class _PlaylistCard extends
StatelessWidget {

```

```

    final String title;

```

```

    final String saves;

```

```

        required this.title,
        required this.saves,
    }) : super(key: key);

@override
Widget build(BuildContext context) {
    return Container(
        padding: const EdgeInsets.all(12),
        decoration: BoxDecoration(
            color: Colors.grey[900],
            borderRadius:
BorderRadius.circular(8),
        ),
        child: Row(
            children: [
                Container(
                    width: 60,
                    height: 60,
                ),
                const SizedBox(height: 4),
            ],
        ),
    );
}

```

```

        decoration: BoxDecoration(
          color: Colors.grey[800],
          borderRadius:
BorderRadius.circular(4),
        ),
        child: const
Icon(Icons.music_note, color:
Colors.white),
      ),
      const SizedBox(width: 12),
      Column(
        crossAxisAlignment:
CrossAxisAlignment.start,
        children: [
          Text(
            title,
            style: const TextStyle(
              color: Colors.white,
              fontWeight:
FontWeight.bold,
            ),

```

```

Text(
  saves,
  style: TextStyle(color:
Colors.grey[400]),
),
],
),
],
),
);
}
}

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

  @override
  Widget build(BuildContext context) {
    IconButton(

```

```

        icon: const
        Icon(Icons.arrow_back, color:
Colors.white),
        onPressed: () {
            Navigator.pop(context);
        },
    ),
    const Spacer(),
    IconButton(
        icon: const
        Icon(Icons.more_vert, color:
Colors.white),
        onPressed: () {},
    ),
],
),
const SizedBox(height: 20),
Container(
    width: 100,
    height: 100,
    decoration: BoxDecoration(
return Stack(
    children: [
        Container(
            decoration: const
BoxDecoration(
                gradient: LinearGradient(
                    begin: Alignment.topCenter,
                    end: Alignment.bottomCenter,
                    colors: [Color(0xFF9370DB),
Colors.black],
                ),
            ),
        ),
        SafeArea(
            child: Padding(
                padding: const
EdgeInsets.all(16.0),
                child: Column(
                    children: [
                        Row(
                            children: [

```

```

        color:
Colors.purple[200],
    shape: BoxShape.circle,
),
child: Center(
    child: Text(
        "A",
        style:
Theme.of(context).textTheme.displayLarge?.copyWith(
        color: Colors.black,
        fontWeight:
FontWeight.bold,
    ),
),
),
),
],
),
),
),
],
        }
    }

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

    @override
    Widget build(BuildContext context) {
        return Container(
            padding: const
EdgeInsets.symmetric(vertical: 12),
            color: Colors.black,
            child: Row(
                mainAxisAlignment:
MainAxisAlignment.spaceEvenly,
                children: [
                    _NavBarItem(
                        icon: Icons.home,
                        label: 'Home',
                    )
                ]
            )
        )
    }
}

```



```

        onTap: () {},
      ),
    _NavBarItem(
      icon: Icons.search,
      label: 'Search',
      onTap: () {},
    ),
    _NavBarItem(
      icon: Icons.library_music,
      label: 'Your Library',
      onTap: () {},
    ),
    _NavBarItem(
      icon: Icons.person,
      label: 'Profile',
      onTap: () {},
    ),
  ],
),
);

class _NavBarItem extends
StatelessWidget {
  final IconData icon;
  final String label;
  final VoidCallback onTap;

  const _NavBarItem({
    Key? key,
    required this.icon,
    required this.label,
    required this.onTap,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return InkWell(
      onTap: onTap,
      child: Column(

```

```
      mainAxisAlignment:
MainAxisSize.min,

      children: [

        Icon(icon, color: Colors.white),

        const SizedBox(height: 4),

        Text(

          label,

          style: const TextStyle(

            color: Colors.white,

            fontSize: 12,

          ),

        ),

      ],

    ),

  );

}

}
```

Search_page:

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

import
'package:flutter_riverpod/flutter_riverp
od.dart';

import
'package:spotify_clone/models/search_
result.dart';

import
'package:spotify_clone/services/youtub
e_service.dart';

import
'package:spotify_clone/providers/audio
_provider.dart';

import
'package:spotify_clone/services/youtub
e_explode_service.dart';

final youtubeServiceProvider =
Provider((ref) => YouTubeService());

final audioPlayerProvider =
StateNotifierProvider<AudioPlayerNot
ifier, 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;

  Future<void> _performSearch(String
query) async {

    if (query.isEmpty) return;
```

```

        _isLoading = true;

    });

    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) {

```

```

        setState(() {

            await
ref.read(audioPlayerProvider.notifier).p
laySong(audioUrl, song);

        }

    }

    @override

    Widget build(BuildContext context) {

        final audioState =
ref.watch(audioPlayerProvider);

        return Scaffold(

            backgroundColor: const
Color(0xFF1C1B1B),

            body: SafeArea(

                child: Column(

                    children: [

                        Padding(

                            padding: const
EdgeInsets.all(16.0),

                            child: TextField(

```

```

controller: _searchController,

style: const TextStyle(color: Colors.white),

onSubmitted:
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)
  ),

CircularProgressIndicator()
else
Expanded(
  child: ListView.builder(
    itemCount: _searchResults.length,
    itemBuilder: (context, index) {
      final song = _searchResults[index];
      final isPlaying = audioState.currentSong?.id == song.id;

      return ListTile(

```

```

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

```

```

          leading:
Image.network(song.thumbnail),
          ref.read(audioPlayerProvider.notifier).t
ogglesPlay();
        } else {
          _playSong(song);
        }
      },
    ),
  );
},
),
),
),
if (audioState.currentSong !=
null)
Container(
  padding: const
EdgeInsets.all(16),
  color: Colors.black,
  child: Row(
    children: [
      Image.network(

```

```

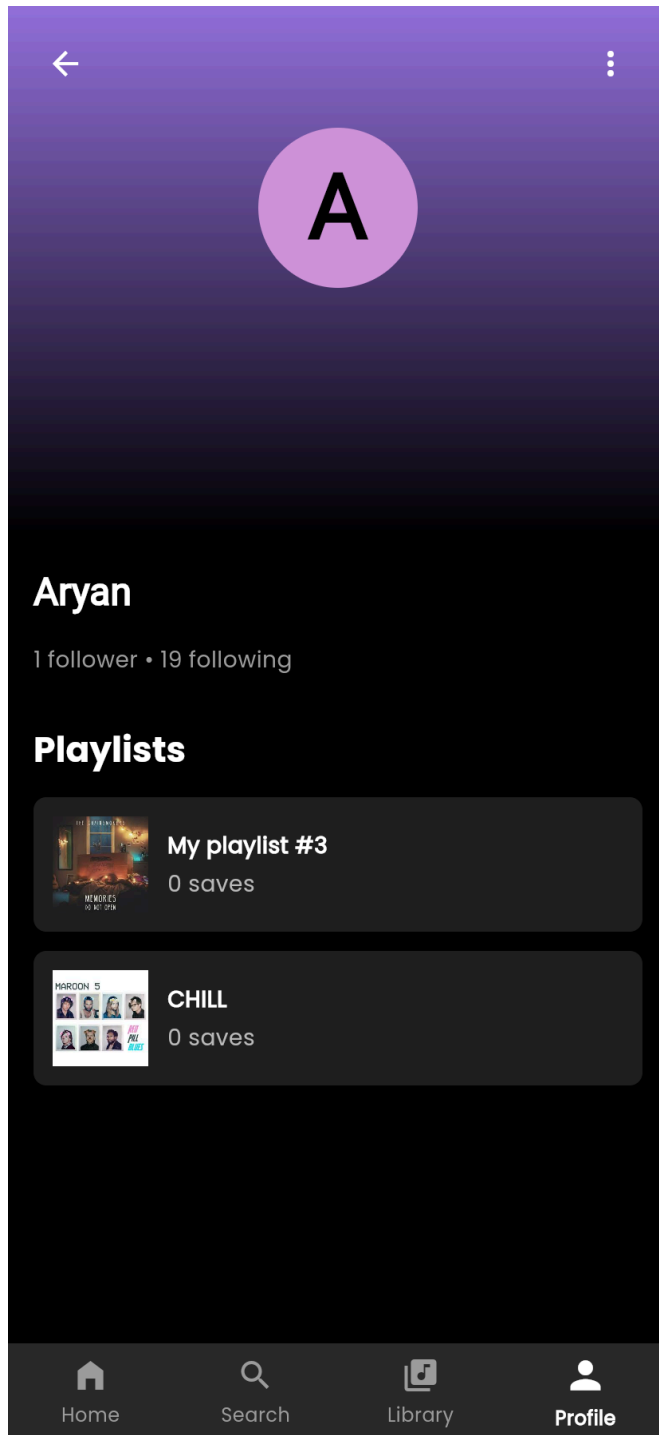
        height: 50,
      ),
    ],
    const SizedBox(width: 16),
  ),
  Expanded(
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Text(
          audioState.currentSong!.title,
          style: const TextStyle(color: Colors.white),
          maxLines: 1,
          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).togglePlay();
      },
    ),
  ],
),
);

```

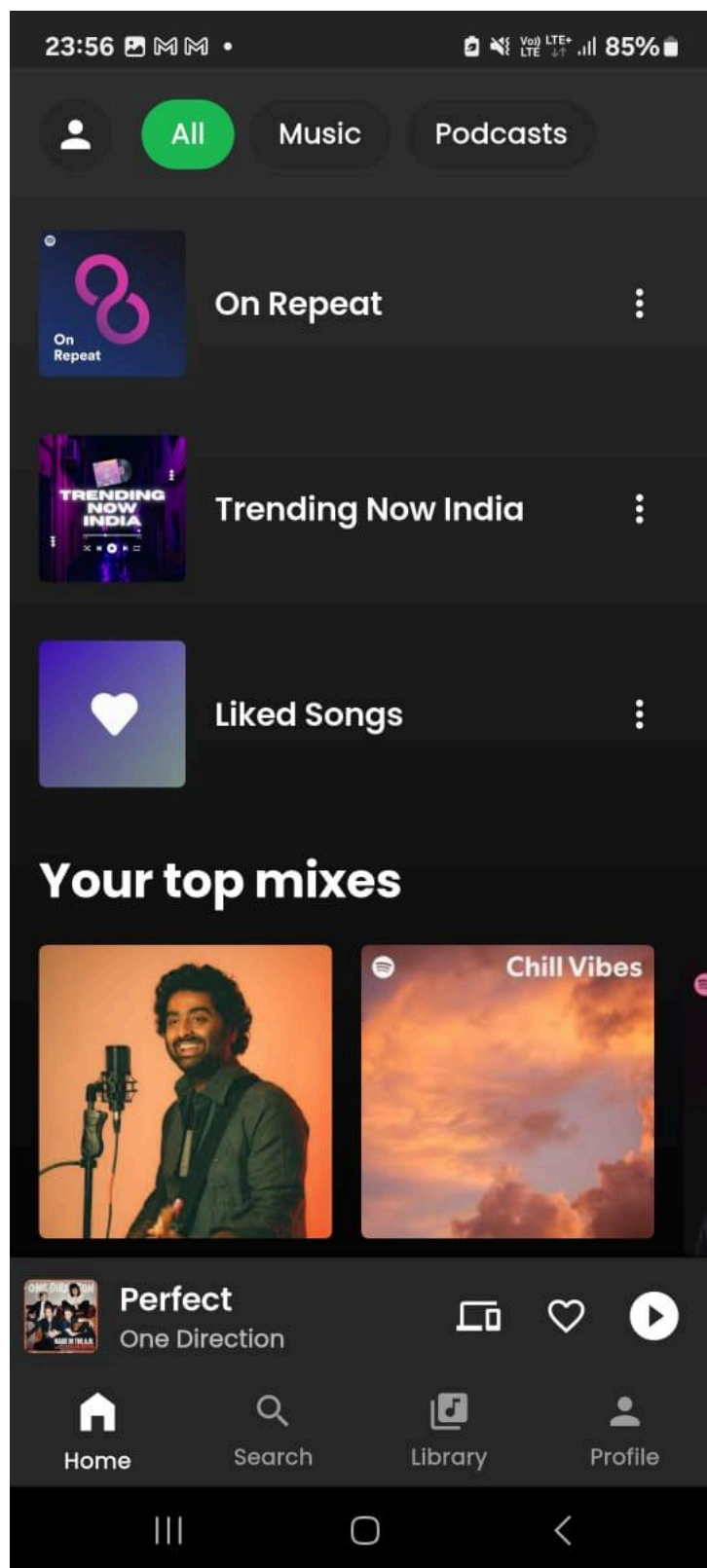
}

OUTPUT

Profile Page



Homepage



Search Page

23:58

VoLTE+ 85%

perfect 1d



One Direction -
Perfect (Official
Video)



OneDirectionVEVO



One Direction -
Perfect (Lyrics)



Authentic Music



One Direction -
Perfect (Audio)



OneDirectionVEVO



One Direction -
Perfect (Lyrics)



VinnyTalks



One Direction
Performs
'Perfect'



Jimmy Kimmel Live



Home



Search



Library



Profile



