Name - Gaurang A Raorane Div - D15A Roll no - 49 Batch - C

**Experiment - 2**

**Aim :-** To design Flutter UI by including common widgets.

Theory :-

In Flutter, widgets are the basic building blocks for creating user interfaces. They represent the visual elements in the UI, such as buttons, text fields, and containers. Here's a theoretical overview of some common Flutter widgets that you might encounter in a laboratory (lab) setting:

* **Container Widget:**
  + Theory: The Container widget is a versatile and powerful widget that can contain other widgets. It allows customization of dimensions, padding, margin, decoration, and more.
  + Lab Use: Use Container to structure and style the layout of your lab interface. Adjust its properties to control spacing and alignment.
* **Text Widget:**
  + Theory: The Text widget displays a string of text with various styling options. It is used to present information to the user.
  + Lab Use: Utilize Text to display important information, labels, or instructions within your lab app.
* **Row and Column Widgets:**
  + Theory: Row and Column are layout widgets that allow you to arrange child widgets in a horizontal (row) or vertical (column) sequence.
  + Lab Use: Organize and structure your UI elements in a row or column format for better readability and organization.
* **ListView Widget:**
  + Theory: ListView is a scrollable list of widgets. It is used to display a scrolling, linear list of widgets.
  + Lab Use: Implement ListView to display a list of items, such as experiment steps, results, or any dynamic data.
* **AppBar Widget:**
  + Theory: The AppBar widget provides a top app bar with options for navigation, title, and actions.
  + Lab Use: Include an AppBar to give your lab app a consistent and recognizable navigation structure.
* **Form and TextFormField Widgets:**
  + Theory: Form and TextFormField are used for creating forms and handling user input.
  + Lab Use: Implement a form structure using Form and capture user input with TextFormField for data entry and interaction.
* **Button Widgets (ElevatedButton, TextButton, and OutlinedButton):**
  + Theory: Flutter provides various button widgets, each with its visual style and behavior.
  + Lab Use: Use buttons like ElevatedButton, TextButton, and OutlinedButton for actions such as submitting data, navigating, or triggering experiments.
* **Image Widget:**
  + Theory: The Image widget displays an image from various sources, such as assets or the internet.
  + Lab Use: Use the Image widget to show diagrams, graphs, or other visual content in your lab app.
* **AlertDialog and SnackBar Widgets:**
  + Theory: AlertDialog and SnackBar are used for displaying pop-up messages to the user.
  + Lab Use: Provide important notifications, alerts, or feedback to users through these pop-up widgets.
* **Scaffold Widget:**
  + Theory: Scaffold is a basic skeletal structure that contains the visual elements of a material design app.
  + Lab Use: Use Scaffold as the overall structure of your lab app, providing a consistent layout with an app bar, body, and other elements.

Input :-

import 'package:firebase\_core/firebase\_core.dart';

import 'package:flutter/material.dart';

import 'package:osiris/Services/auth.dart';

import 'package:osiris/routes.dart';

import 'package:provider/provider.dart';

void main() async {

WidgetsFlutterBinding.ensureInitialized();

await Firebase.initializeApp();

runApp(const App());

}

class App extends StatefulWidget {

const App({super.key});

@override

State<App> createState() => \_AppState();

}

class \_AppState extends State<App> {

@override

Widget build(BuildContext context) {

return ChangeNotifierProvider(

create: (context) => GoogleSignInProvider(),

child: MaterialApp.router(

debugShowCheckedModeBanner: false,

routerConfig: router,

),

);

}

}

**MainScreen.dart**  
import 'package:flutter/material.dart';

import 'package:flutter/rendering.dart';

import 'package:osiris/Models/PopularMovies.dart';

import 'package:osiris/Models/TvShow.dart';

import 'package:osiris/Services/API.dart';

import 'package:osiris/Services/consts.dart';

import 'package:osiris/Widgets/BottomNavBar.dart';

import 'package:osiris/Widgets/CarouselCard.dart';

import 'package:osiris/Widgets/CustomLists.dart';

import 'package:osiris/Widgets/LoadingScreen.dart';

import 'package:osiris/Widgets/SectionText.dart';

class MainScreen extends StatefulWidget {

const MainScreen({super.key});

@override

State<MainScreen> createState() => \_MainScreenState();

}

class \_MainScreenState extends State<MainScreen> {

ScrollController \_scrollController = ScrollController();

bool isVisible = true;

late List<Results> popularMovie;

late List<Results> topRatedMovie;

late List<Results> nowPLayingMovie;

late List<TvShow> popularShows;

late List<TvShow> topRatedShows;

bool isLoading = true;

@override

void initState() {

super.initState();

fetchData();

\_scrollController = ScrollController();

\_scrollController.addListener(listen);

}

@override

void dispose() {

\_scrollController.removeListener(listen);

\_scrollController.dispose();

super.dispose();

}

void listen() {

final direction = \_scrollController.position.userScrollDirection;

if (direction == ScrollDirection.forward) {

show();

} else if (direction == ScrollDirection.reverse) {

hide();

}

}

void show() {

if (!isVisible) {

(setState(

() => isVisible = true,

));

}

}

void hide() {

if (isVisible) {

(setState(

() => isVisible = false,

));

}

}

Future<void> fetchData() async {

topRatedShows = await APIService().getTopRatedShow();

popularMovie = await APIService().getPopularMovie();

topRatedMovie = await APIService().getTopRatedMovie();

popularShows = await APIService().getRecommendedTvShows("1396");

nowPLayingMovie = await APIService().getNowPLayingMovie();

setState(() {

isLoading = false;

});

}

@override

Widget build(BuildContext context) {

var size = MediaQuery.of(context).size;

return Scaffold(

bottomNavigationBar: AnimatedBuilder(

animation: \_scrollController,

builder: ((context, child) {

return AnimatedContainer(

duration: const Duration(milliseconds: 800),

curve: Curves.fastLinearToSlowEaseIn,

height: isVisible ? 75 : 0,

child: BottomNavBar(

currentIndex: 0,

),

);

})),

extendBody: true,

body: isLoading

? const LoadingScreen()

: Container(

height: size.height,

width: size.width,

color: background\_primary,

child: ListView(

padding: EdgeInsets.zero,

scrollDirection: Axis.vertical,

physics: const BouncingScrollPhysics(),

controller: \_scrollController,

shrinkWrap: true,

children: [

CustomCarouselSlider(topRatedShows),

SectionText("Popular", "Movies"),

CustomListMovie(popularMovie),

SectionText("TOP Rated", "Movies"),

CustomListMovie(topRatedMovie),

SectionText("Popular", "Shows"),

CustomListTV(popularShows),

SectionText("NoW PLAying", "Movies"),

CustomListMovie(nowPLayingMovie),

],

),

),

);

}

}

**CarouselCard.dart**  
import 'package:cached\_network\_image/cached\_network\_image.dart';

import 'package:flutter/material.dart';

import 'package:flutter/services.dart';

import 'package:go\_router/go\_router.dart';

import 'package:osiris/Models/TvShow.dart';

import 'package:osiris/Widgets/LandingCard.dart';

class CustomCarouselSlider extends StatelessWidget {

CustomCarouselSlider(this.data, {super.key});

List<TvShow> data;

@override

Widget build(BuildContext context) {

var size = MediaQuery.of(context).size;

return SizedBox(

width: size.width,

height: (size.height \* 0.33 < 300) ? 300 : size.height \* 0.33,

child: PageView.builder(

scrollDirection: Axis.horizontal,

physics: const BouncingScrollPhysics(),

pageSnapping: true,

itemCount: 20,

itemBuilder: ((context, index) {

var url = data[index].backdropPath.toString();

return GestureDetector(

onTap: () {

HapticFeedback.mediumImpact();

GoRouter.of(context).push('/tv/${data[index].id}');

},

child: LandingCard(

CachedNetworkImageProvider(

"https://image.tmdb.org/t/p/original$url"),

data[index].name.toString()),

);

}),

));

}

}

**MovieCard.dart**import 'package:flutter/material.dart';

import 'package:flutter/services.dart';

import 'package:go\_router/go\_router.dart';

class MovieCard extends StatelessWidget {

MovieCard(this.title, this.image, this.Id, this.mediaType, {super.key});

String title;

ImageProvider image;

String Id;

String mediaType;

@override

Widget build(BuildContext context) {

return GestureDetector(

onTap: () {

HapticFeedback.mediumImpact();

GoRouter.of(context).push('/$mediaType/$Id');

},

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

crossAxisAlignment: CrossAxisAlignment.center,

children: [

Container(

height: 150,

width: 100,

margin: const EdgeInsets.fromLTRB(8, 4, 8, 4),

decoration: BoxDecoration(

image: DecorationImage(fit: BoxFit.cover, image: image),

borderRadius: BorderRadius.circular(10),

),

),

SizedBox(

width: 100,

child: Text(

title,

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

maxLines: 1,

overflow: TextOverflow.ellipsis,

textAlign: TextAlign.center,

),

)

],

),

);

}

}

Pubspec.yaml

flutter:

uses-material-design: true

assets:

- assets/

flutter\_icons:

android: "launcher\_icon"

image\_path\_android: "assets/icon.png"

adaptive\_icon\_background: "assets/launcher\_icon/background.png"

adaptive\_icon\_foreground: "assets/launcher\_icon/foreground.png"

**Output:-**

