Experiment No. 2

**Aim:-**

To design Flutter UI by including common widgets.

**Input Widgets:-**

1.TextField: Allows text input.

TextField(

decoration: InputDecoration(labelText: 'Enter Name', border: OutlineInputBorder()),

)

2. Button Widgets:

a. ElevatedButton: Raised button with elevation.

ElevatedButton(

onPressed: () {}, child: Text('Click Me'),

)

b. TextButton: Flat button with no elevation.

TextButton( onPressed: () {}, child: Text('Click Me'),

)

c. IconButton: A button with an icon.

IconButton(

icon: Icon(Icons.add), onPressed: () {},

)  
D. Navigation & Routing Widgets

1. Navigator: Handles screen transitions.

Navigator.push(context, MaterialPageRoute(builder: (context) => NewScreen())); 2. BottomNavigationBar: Used for switching between different screens. BottomNavigationBar(

items: [ BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),

BottomNavigationBarItem(icon: Icon(Icons.settings), label: 'Settings'),

],

)

E. Interactive & Gestures GestureDetector: Detects taps, swipes, and other gestures. GestureDetector(

onTap: () => print('Tapped!'),

child: Container(color: Colors.blue, height: 100, width: 100),

)

**Code:-**

*Profile*

import 'package:flutter/material.dart';

import 'package:flutter\_svg/flutter\_svg.dart';

import 'package:foodie/theme/text\_styles.dart';

class ProfileTitleRow extends StatelessWidget {

  const ProfileTitleRow({

    super.key,

    required this.icon,

    required this.label,

  });

  final String icon;

  final String label;

  @override

  Widget build(BuildContext context) {

    return Padding(

      padding: const EdgeInsets.symmetric(vertical: 15),

      child: Row(

        mainAxisAlignment: MainAxisAlignment.spaceBetween,

        children: [

          SvgPicture.asset(icon),

          const SizedBox(

            width: 15,

          ),

          Expanded(

            child: Text(

              label,

              style: TextStyles.titleProfile,

            ),

          ),

        ],

      ),

    );

  }

}

*Homepage*

import 'package:flutter/material.dart';

import 'package:flutter\_riverpod/flutter\_riverpod.dart';

import 'package:foodie/router/router\_context\_extension.dart';

import 'package:foodie/services/points/points.dart';

import '../../../../../common/is\_saved\_material\_button.dart';

import '../../../domain/recipe.dart';

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

import 'package:shimmer/shimmer.dart';

import '../../../../../theme/theme.dart';

class RecipeWidget extends ConsumerWidget {

  const RecipeWidget({

    super.key,

    required this.recipe,

  });

  final Recipe recipe;

  @override

  Widget build(BuildContext context, WidgetRef ref) {

    return GestureDetector(

      onTap: () => context.pushDetailPage(recipe: recipe),

      child: Stack(

        alignment: Alignment.bottomCenter,

        children: [

          Container(

            height: 120,

            alignment: Alignment.bottomCenter,

            decoration: BoxDecoration(

                borderRadius: BorderRadius.circular(

                  16,

                ),

                color: ThemeColors.main),

          ),

          Padding(

            padding: const EdgeInsets.symmetric(horizontal: 15),

            child: Column(

              crossAxisAlignment: CrossAxisAlignment.start,

              mainAxisAlignment: MainAxisAlignment.spaceAround,

              children: [

                Align(

                  alignment: Alignment.topCenter,

                  child: Stack(

                    children: [

                      ClipRRect(

                        borderRadius: BorderRadius.circular(16),

                        child: CachedNetworkImage(

                          width: 150,

                          height: 150,

                          fit: BoxFit.cover,

                          imageUrl: recipe.imageUrl,

                          placeholder: (context, url) => Shimmer.fromColors(

                            baseColor: Colors.black26,

                            highlightColor: Colors.black12,

                            child: Container(

                              color: ThemeColors.white,

                            ),

                          ),

                        ),

                      ),

                      Padding(

                        padding: const EdgeInsets.all(10),

                        child: IsSavedMaterialButton(

                          recipe: recipe,

                        ),

                      ),

                    ],

                  ),

                ),

                Text(

                  recipe.name,

                  maxLines: 2,

                  style: TextStyles.recipeName,

                ),

                Padding(

                  padding: const EdgeInsets.only(bottom: 5),

                  child: Row(

                    mainAxisAlignment: MainAxisAlignment.spaceBetween,

                    children: [

                      Text(

                        '${Points.recipePoints} points',

                        style: TextStyles.points,

                      ),

                      Text(

                        '${recipe.time != null ? (recipe.time != 0 ? recipe.time : '') : ''} ${recipe.time != null ? (recipe.time != 0 ? 'mins' : '') : ''}',

                        style: TextStyles.time,

                      ),

                    ],

                  ),

                ),

              ],

            ),

          )

        ],

      ),

    );

  }

}

*Saved*

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

import 'package:flutter/material.dart';

import 'package:flutter\_riverpod/flutter\_riverpod.dart';

import 'package:foodie/constants/storage\_box\_constants.dart';

import 'package:foodie/providers/providers.dart';

import 'package:foodie/router/router\_context\_extension.dart';

import 'package:shimmer/shimmer.dart';

import '../../../../constants/string\_constants.dart';

import '../../../../services/points/points.dart';

import '../../../../theme/theme.dart';

import '../../domain/recipe.dart';

class SavedRecipesPage extends ConsumerWidget {

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

  @override

  Widget build(BuildContext context, WidgetRef ref) {

    final storageService = ref.watch(storageServiceProvider);

    final imageSize = MediaQuery.of(context).size.width \* 0.3;

    final favoriteBoxSize =

        storageService.getLength<Recipe>(StorageBox.favoritesBox);

    return WillPopScope(

      onWillPop: () async => false,

      child: Scaffold(

        appBar: AppBar(

          automaticallyImplyLeading: false,

          title: Text(

            StringConstants.savedRecipes,

            style: TextStyles.title,

          ),

        ),

        body: SafeArea(

          child: favoriteBoxSize == 0

              ? Center(

                  heightFactor: MediaQuery.sizeOf(context).height \* 0.5,

                  child: Text(

                    StringConstants.noSavedRecipes,

                    style: TextStyles.subtitle,

                  ),

                )

              : Column(

                  children: [

                    Expanded(

                      child: ListView.builder(

                          shrinkWrap: true,

                          itemCount: favoriteBoxSize,

                          itemBuilder: (context, index) {

                            final recipe = storageService

                                .getAll<Recipe>(StorageBox.favoritesBox)[index];

                            return Padding(

                              padding: const EdgeInsets.fromLTRB(20, 20, 20, 0),

                              child: GestureDetector(

                                onTap: () =>

                                    context.pushDetailPage(recipe: recipe),

                                child: Row(

                                  children: [

                                    ClipRRect(

                                      borderRadius: BorderRadius.circular(16),

                                      child: CachedNetworkImage(

                                        width: imageSize,

                                        height: imageSize,

                                        fit: BoxFit.cover,

                                        imageUrl: recipe.imageUrl,

                                        placeholder: (context, url) =>

                                            Shimmer.fromColors(

                                          baseColor: Colors.black26,

                                          highlightColor: Colors.black12,

                                          child: Container(

                                            color: ThemeColors.white,

                                          ),

                                        ),

                                      ),

                                    ),

                                    const SizedBox(

                                      width: 20,

                                    ),

                                    Expanded(

                                      child: Column(

                                        crossAxisAlignment:

                                            CrossAxisAlignment.start,

                                        children: [

                                          Text(

                                            recipe.name,

                                            style: TextStyles.title,

                                          ),

                                          const SizedBox(

                                            height: 20,

                                          ),

                                          Text('${Points.recipePoints} points',

                                              style: TextStyles.points)

                                        ],

                                      ),

                                    )

                                  ],

                                ),

                              ),

                            );

                          }),

                    )

                  ],

                ),

        ),

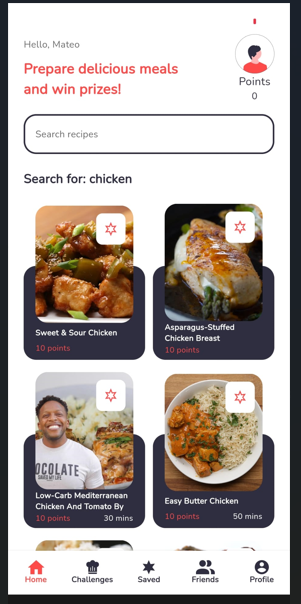
      ),

    );

  }

}

**Output:-**

****

