Shilpa Pandey
42 D15A

Experiment 2

Aim :   To design Flutter UI by including common widgets

Theory:

Text:
A Text widget holds some text to display on the screen. We can align the text widget by using **textAlign** property, and style property allow the customization of Text that includes font, font weight, font style, letter spacing, color, and many more.

Button:

This widget allows you to perform some action on click. Flutter does not allow you to use the Button widget directly; instead, it uses a type of buttons like a **FlatButton** and a **RaisedButton**.

Image

This widget holds the image which can fetch it from multiple sources like from the asset folder or directly from the URL. It provides many constructors for loading image, which are given below:

- **Image:** It is a generic image loader, which is used by **ImageProvider**.

- **asset:** It load image from your project asset folder.

- **file:** It loads images from the system folder.

- **memory:** It load image from memory.

- **network:** It loads images from the network.

Scaffold

This widget provides a framework that allows you to add common material design elements like AppBar, Floating Action Buttons, Drawers, etc.

Stack

It is an essential widget, which is mainly used for **overlapping** a widget, such as a button on a background gradient.

State Management Widget

In Flutter, there are mainly two types of widget:

- ○ StatelessWidget

- ○ StatefulWidget

StatefulWidget

A StatefulWidget has state information. It contains mainly two classes: the **state object** and the **widget**. It is dynamic because it can change the inner data during the widget lifetime. This widget does not have a **build()** method. It has **createState()** method, which returns a class that extends the Flutters State Class. The examples of the StatefulWidget are Checkbox, Radio, Slider, InkWell, Form, and TextField.

StatelessWidget

The StatelessWidget does not have any state information. It remains static throughout its lifecycle. The examples of the StatelessWidget are Text, Row, Column, Container, etc.

Code :

```
cart
import 'dart:ui';
import 'package:amazon_clone/model/product_model.dart';
import 'package:amazon_clone/model/user_details_model.dart';
```

```dart
import 'package:amazon_clone/providers/user_details_provider.dart';
import 'package:amazon_clone/resources/cloudfirestore_methods.dart';
import 'package:amazon_clone/utils/color_themes.dart';
import 'package:amazon_clone/utils/constants.dart';
import 'package:amazon_clone/utils/utils.dart';
import 'package:amazon_clone/widgets/cart_item_widget.dart';
import 'package:amazon_clone/widgets/custom_main_button.dart';
import 'package:amazon_clone/widgets/search_bar_widget.dart';
import 'package:amazon_clone/widgets/user_details_bar.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';


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


 @override
 State<CartScreen> createState() => _CartScreenState();
}


class _CartScreenState extends State<CartScreen> {
 @override
 Widget build(BuildContext context) {
  return Scaffold(
    appBar: SearchBarWidget(
      hasBackButton: false,
      isReadOnly: true,
    ),
    body: Center(
      child: Stack(
        children: [
          Column(
            children: [
              const SizedBox(
                height: kAppBarHeight / 2,
              ),
              Padding(
                  padding: const EdgeInsets.all(8.0),
                  child: StreamBuilder(
                    stream: FirebaseFirestore.instance
```

```dart
              .collection("users")
              .doc(FirebaseAuth.instance.currentUser!.uid)
              .collection("cart")
              .snapshots(),
          builder: (context,
              AsyncSnapshot<QuerySnapshot<Map<String, dynamic>>>
                  snapshot) {
            if (snapshot.connectionState ==
                ConnectionState.waiting) {
              return CustomMainButton(
                  child: const Text(
                    "Loading",
                  ),
                  color: yellowColor,
                  isLoading: true,
                  onPressed: () {});
            } else {
              return CustomMainButton(
                  child: Text(
                    "Proceed to buy (${snapshot.data!.docs.length}) items",
                    style: const TextStyle(color: Colors.black),
                  ),
                  color: yellowColor,
                  isLoading: false,
                  onPressed: () async {
                    await CloudFirestoreClass().buyAllItemsInCart(
                        userDetails:
                            Provider.of<UserDetailsProvider>(
                                    context,
                                    listen: false)
                                .userDetails);
                    Utils().showSnackBar(
                        context: context, content: "Done");
                  });
            }
          },
        )),
    Expanded(
        child: StreamBuilder(
      stream: FirebaseFirestore.instance
          .collection("users")
          .doc(FirebaseAuth.instance.currentUser!.uid)
          .collection("cart")
          .snapshots(),
```
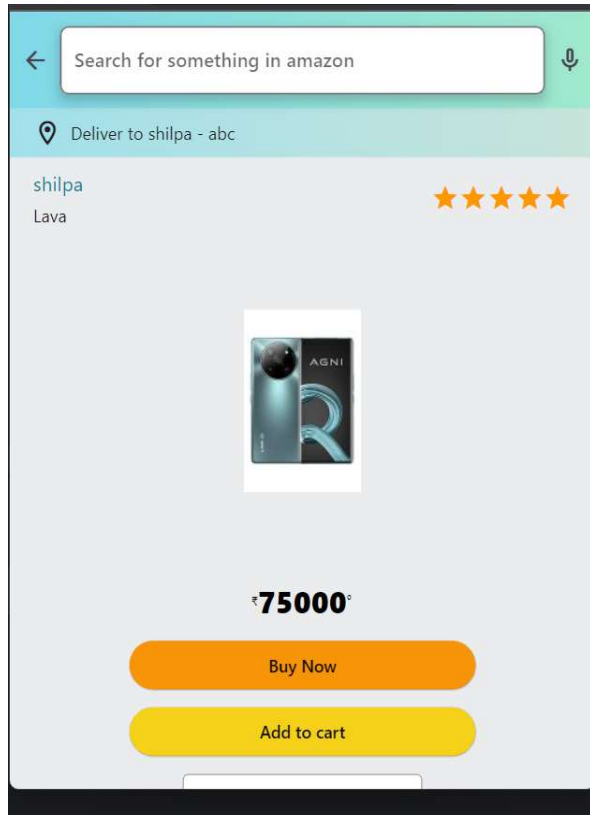
```dart
          builder: (context,
              AsyncSnapshot<QuerySnapshot<Map<String, dynamic>>>
                  snapshot) {
            if (snapshot.connectionState == ConnectionState.waiting) {
              return Container();
            } else {
              return ListView.builder(
                  itemCount: snapshot.data!.docs.length,
                  itemBuilder: (context, index) {
                    ProductModel model = ProductModel.getModelFromJson(
                        json: snapshot.data!.docs[index].data());
                    return CartItemWidget(product: model);
                  });
            }
          },
        ))
      ],
    ),
    UserDetailsBar(
      offset: 0,
    ),
  ],
),
),
);
}
}
```

Home
import 'package:amazon_clone/model/user_details_model.dart';
import 'package:amazon_clone/resources/cloudfirestore_methods.dart';
import 'package:amazon_clone/utils/constants.dart';
import 'package:amazon_clone/widgets/ad_banner_widget.dart';
import 'package:amazon_clone/widgets/categories_horizontal_list_view_bar.dart';
import 'package:amazon_clone/widgets/loading_widget.dart';
import 'package:amazon_clone/widgets/products_showcase_list_view.dart';
import 'package:amazon_clone/widgets/search_bar_widget.dart';
import 'package:amazon_clone/widgets/simple_product_widget.dart';
import 'package:amazon_clone/widgets/user_details_bar.dart';
import 'package:flutter/material.dart';

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

 @override

```dart
 State<HomeScreen> createState() => _HomeScreenState();
}


class _HomeScreenState extends State<HomeScreen> {
 ScrollController controller = ScrollController();
 double offset = 0;
 List<Widget>? discount70;
 List<Widget>? discount60;
 List<Widget>? discount50;
 List<Widget>? discount0;


 @override
 void initState() {
  super.initState();
  getData();
  controller.addListener(() {
    setState(() {
      offset = controller.position.pixels;
    });
  });
 }


 @override
 void dispose() {
  super.dispose();
  controller.dispose();
 }


 void getData() async {
  List<Widget> temp70 =
   await CloudFirestoreClass().getProductsFromDiscount(70);
  List<Widget> temp60 =
   await CloudFirestoreClass().getProductsFromDiscount(60);
  List<Widget> temp50 =
   await CloudFirestoreClass().getProductsFromDiscount(50);
  List<Widget> temp0 = await CloudFirestoreClass().getProductsFromDiscount(0);
  print("everything is done");
  setState(() {
    discount70 = temp70;
    discount60 = temp60;
```

```dart
      discount50 = temp50;
      discount0 = temp0;
    });
}


@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: SearchBarWidget(
      isReadOnly: true,
      hasBackButton: false,
    ),
    body: discount70 != null &&
        discount60 != null &&
        discount50 != null &&
        discount0 != null
        ? Stack(
      children: [
        SingleChildScrollView(
          controller: controller,
          child: Column(
            children: [
              SizedBox(
                height: kAppBarHeight / 2,
              ),
              CategoriesHorizontalListViewBar(),
              AdBannerWidget(),
              ProductsShowcaseListView(
                  title: "Upto 70% Off", children: discount70!),
              ProductsShowcaseListView(
                  title: "Upto 60% Off", children: discount60!),
              ProductsShowcaseListView(
                  title: "Upto 50% Off", children: discount50!),
              ProductsShowcaseListView(
                  title: "Explore", children: discount0!),
            ],
          ),
        ),
        UserDetailsBar(
          offset: offset,
        ),
      ],
    )
```

```
            : const LoadingWidget(),
   );
 }
}
```