Experiment 4

Aim: To design flutter ui by including common widgets

Theory: Forms are essential in collecting user input in apps. Flutter provides interactive form widgets such as:

- **TextField**: Takes text input.
- **DropdownButton**: Lets users select from a list of options.
- ElevatedButton: Triggers actions like form submission.
- Form & GlobalKey: Used for validation and managing form state.

Flutter also supports input validation and displaying feedback to users.

Common Widgets Used:

- Form & GlobalKey<FormState>
- TextFormField
- DropdownButtonFormField
- DatePicker
- ElevatedButton

Code:

```
import 'package:flutter/material.dart';
import 'package:monkey_app_demo/const/colors.dart';
import 'package:monkey_app_demo/screens/changeAddressScreen.dart';
import 'package:monkey_app_demo/screens/homeScreen.dart';
import 'package:monkey_app_demo/utils/helper.dart';
import 'package:monkey_app_demo/widgets/customNavBar.dart';
import 'package:monkey_app_demo/widgets/customTextInput.dart';
```

```
class CheckoutScreen extends StatelessWidget {
 static const routeName = "/checkoutScreen";
 @override
 Widget build(BuildContext context) {
  return Scaffold(
   body: Stack(
    children: [
     SafeArea(
       child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
         Row(
          children: [
            IconButton(
             onPressed: () {
              Navigator.of(context).pop();
             },
             icon: Icon(
              Icons.arrow back ios rounded,
             ),
            ),
            Expanded(
             child: Text(
              "Checkout",
              style: Helper.getTheme(context).headline5,
             ),
            ),
          ],
```

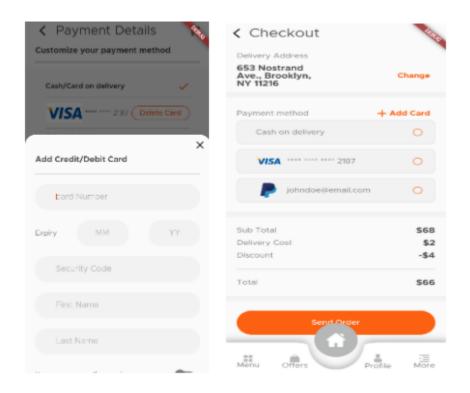
```
),
SizedBox(
 height: 20,
),
Padding(
 padding: const EdgeInsets.symmetric(horizontal: 20),
 child: Text("Delivery Address"),
),
SizedBox(
 height: 10,
),
Padding(
 padding: const EdgeInsets.symmetric(horizontal: 20),
 child: Row(
  mainAxisAlignment: MainAxisAlignment.spaceBetween,
  children: [
   SizedBox(
    width: Helper.getScreenWidth(context) * 0.4,
    child: Text(
      "653 Nostrand Ave., Brooklyn, NY 11216",
      style: Helper.getTheme(context).headline3,
    ),
   ),
   TextButton(
    onPressed: () {
     Navigator.of(context)
        .pushNamed(ChangeAddressScreen.routeName);
    },
    child: Text(
```

```
"Change",
      style: TextStyle(
       fontWeight: FontWeight.bold,
      ),
     ),
  ],
 ),
),
SizedBox(
 height: 10,
),
Container(
 height: 10,
 width: double.infinity,
 color: AppColor.placeholderBg,
),
SizedBox(
 height: 20,
),
Padding(
 padding: const EdgeInsets.symmetric(horizontal: 20),
 child: Row(
  mainAxisAlignment: MainAxisAlignment.spaceBetween,
  children: [
   Text("Payment method"),
   TextButton(
    onPressed: () {
      showModalBottomSheet(
```

```
shape: RoundedRectangleBorder(
 borderRadius: BorderRadius.circular(20),
),
isScrollControlled: true,
isDismissible: false,
context: context,
builder: (context) {
 return Container(
  height: Helper.getScreenHeight(context) * 0.7,
  child: Column(
   children: [
    Row(
      mainAxisAlignment:
        MainAxisAlignment.end,
      children: [
       IconButton(
        onPressed: () {
         Navigator.of(context).pop();
        },
        icon: Icon(
         Icons.clear,
        ),
     ),
     Padding(
      padding: const EdgeInsets.symmetric(
        horizontal: 20.0),
     child: Row(
```

```
children: [
   Text(
     "Add Credit/Debit Card",
     style: Helper.getTheme(context)
       .headline3,
   )
  ],
 ),
),
Padding(
 padding: const EdgeInsets.symmetric(
   horizontal: 20.0),
 child: Divider(
  color: AppColor.placeholder
     .withOpacity(0.5),
  thickness: 1.5,
  height: 40,
 ),
),
Padding(
 padding: const EdgeInsets.symmetric(
   horizontal: 20.0),
 child: CustomTextInput(
   hintText: "card Number"),
),
```

Screenshots:



Conclusion: In this lab, we explored how to create and manage interactive forms in Flutter using widgets such as TextFormField, DropdownButtonFormField, and ElevatedButton. We designed a payment and checkout form that includes card details input, payment method selection, and dynamic validation. This hands-on experience demonstrates how forms can be used to securely collect and manage user payment information, making them essential for building real-world e-commerce applications.