

MAD and PWA Lab

Name: Sneha Utekar

Class: D15A

Roll no: 66

Experiment No.-4

AIM: To create an interactive Form using form widget

THEORY:

LoginScreen Widget:

Description: Represents the login screen with OLX branding and options to log in using various methods.

Used in: main.dart

Key Widgets:

1. Scaffold
2. Column
3. Container
4. Image.asset
5. Text
6. AuthUi

EmailAuthScreen Widget:

Description: Allows the user to log in or register using email and password.

Used in: login_screen.dart

Key Widgets:

1. Scaffold
2. AppBar
3. Form
4. TextFormField
5. DropdownButtonFormField
6. Checkbox

7. Radio
8. ElevatedButton

GoogleAuthentication Class:

Description: Handles Google authentication and sign-in.

Used in: login_screen.dart

Key Widgets:

SnackBar

PhoneAuthScreen Widget:

Description: Allows the user to log in using their phone number and initiates phone number verification.

Used in: login_screen.dart

Key Widgets:

- 1.Scaffold
- 2.AppBar
- 3.Padding
- 4.AbsorbPointer
- 5.ElevatedButton

PhoneAuthentication Class:

Description: Handles phone number authentication and verification.

Used in: phoneAuth.dart

Key Widgets:

ProgressDialog

OTPScreen Widget:

Description: Allows the user to enter the OTP received via SMS for phone number verification.

Used in: phoneAuth.dart

Key Widgets:

- 1.Scaffold
- 2.AppBar
- 3.Column
- 4.TextFormField
- 5.LinearProgressIndicator

AuthUi Widget:

Description: Displays various authentication options like phone, Google, Facebook, and email.

Used in: login_screen.dart

Key Widgets:

1. Center
2. Column
3. ElevatedButton
4. SignInButton
5. InkWell

CODE:

login_screen.dart:

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:olx_clone/screens/location_screen.dart';
import 'package:olx_clone/widgets/auth_ui.dart';
```

```
class LoginScreen extends StatelessWidget {
  static const String id='login-screen';
  @override
```

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

```
  return Scaffold(
    backgroundColor: Colors.cyan.shade900,
    body:Column(
      children: [
```

```

Expanded(
  child:Container(
    width:MediaQuery.of(context).size.width,//to get device width
    color: Colors.white,
    child: Column(
      children: [
        SizedBox(height: 50),
        Image.asset('assets/images/cart.png',
          color: Colors.cyan.shade900,
          height: 100,),
        SizedBox(height: 5,),
        Text('Olx',style: TextStyle(
          fontFamily: 'Horizon',
          fontSize: 50,
          fontWeight:FontWeight.bold,
          color: Colors.cyan.shade900,

        ),)

      ],
    ),
  ),),
Expanded(child:Container(
  child: AuthUi(),
),),
Padding(padding: const EdgeInsets.all(8.0),
  child:Text('If you continue,you are accepting\nTerms and Conditions and Privacy
Policy'
  ,textAlign:TextAlign.center,style:TextStyle(color: Colors.white,fontSize: 10) ),
),
],

),

);
}
}

```

emailAuthScreen.dart:

```

import 'package:email_validator/email_validator.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:olx_clone/services/emailAuth_service.dart';

class EmailAuthScreen extends StatefulWidget {
  static const String id = 'emailAuth-Screen';

  @override
  State<EmailAuthScreen> createState() => _EmailAuthScreenState();
}

class _EmailAuthScreenState extends State<EmailAuthScreen> {
  final _formKey = GlobalKey<FormState>();
  bool _validate = false;
  bool _login = true;
  bool _loading = false;
  var _emailController = TextEditingController();
  var _passwordController = TextEditingController();
  EmailAuthentication _service = EmailAuthentication();

  _validateEmail() {
    if (_formKey.currentState!.validate()) {
      setState(() {
        _validate = false;
        _loading = true;
      });
      _service.getAdminCredential(
        context: context,
        isLog: _login,
        password: _passwordController.text,
        email: _emailController.text,
      ).then((value) {
        setState(() {
          _loading = false;
        });
      });
    }
  }

  @override

```

```

Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      elevation: 1,
      iconTheme: IconThemeData(color: Colors.black),
      backgroundColor: Colors.white,
      title: Text(
        'Login',
        style: TextStyle(color: Colors.black),
      ),
    ),
    body: Form(
      key: _formKey,
      child: Padding(
        padding: const EdgeInsets.all(20.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            SizedBox(height: 40),
            CircleAvatar(
              radius: 30,
              backgroundColor: Colors.red.shade200,
              child: Icon(
                CupertinoIcons.person_alt_circle,
                color: Colors.red,
                size: 60,
              ),
            ),
            SizedBox(height: 12),
            Text(
              'Enter your ${_login ? 'Login' : 'Register'}',
              style: TextStyle(fontSize: 30, fontWeight: FontWeight.bold),
            ),
            SizedBox(height: 10),
            Text(
              'Enter Email and Password to ${_login ? 'Login' : 'Register'}',
              style: TextStyle(color: Colors.grey),
            ),
            SizedBox(
              height: 10,
            ),
          ],
        ),
      ),
    ),
  );
}

```

```
TextFormField(
  controller: _emailController,
  validator: (value) {
    final bool isValid =
      EmailValidator.validate(_emailController.text);
    if (value == null || value.isEmpty) {
      return 'Enter Email';
    }
    if (value.isNotEmpty && isValid == false) {
      return 'Enter Valid Email';
    }
    return null;
  },
  decoration: InputDecoration(
    contentPadding: EdgeInsets.only(left: 10),
    labelText: 'Email',
    filled: true,
    fillColor: Colors.grey.shade300,
    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(4),
    ),
  ),
),
),
SizedBox(
  height: 10,
),
TextFormField(
  controller: _passwordController,
  decoration: InputDecoration(
    suffixIcon: _validate
      ? IconButton(
        icon: Icon(Icons.clear),
        onPressed: () {
          _passwordController.clear();
          setState() {
            _validate = false;
          };
        },
      ),
  ),
),
): null,
contentPadding: EdgeInsets.only(left: 10),
```

```

        labelText: 'Password',
        filled: true,
        fillColor: Colors.grey.shade300,
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(4),
        ),
      ),
    ),
    onChanged: (value) {
      if (_emailController.text.isNotEmpty) {
        if (value.length > 3) {
          setState(() {
            _validate = true;
          });
        }
      } else {
        setState(() {
          _validate = false;
        });
      }
    },
  ),
  Row(
    children: [
      Text(_login ? 'New Account?' : 'Already has an Account?'),
      TextButton(
        child: Text(
          _login ? 'Register' : 'Login',
          style: TextStyle(
            color: Theme.of(context).primaryColor,
            fontWeight: FontWeight.bold,
          ),
        ),
        onPressed: () {
          setState(() {
            _login = !_login;
          });
        },
      ),
    ],
  ),
],
),
],

```



```

    ),
  ),
),
bottomNavigationBar: SafeArea(
  child: Padding(
    padding: const EdgeInsets.all(12.0),
    child: AbsorbPointer(
      absorbing: _validate ? false : true,
      child: ElevatedButton(
        style: ButtonStyle(
          backgroundColor: _validate
            ? MaterialStateProperty.all(
                Theme.of(context).primaryColor)
            : MaterialStateProperty.all(Colors.grey),
        ),
        onPressed: () {
          _validateEmail();
        },
        child: Padding(
          padding: const EdgeInsets.all(12.0),
          child: _loading
            ? SizedBox(
                height: 24,
                width: 24,
                child: CircularProgressIndicator(
                  valueColor:
                    AlwaysStoppedAnimation<Color>(Colors.white),
                ),
              )
            : Text(
                '$_login ? 'Login' : 'Register'',
                style: TextStyle(
                  color: Colors.white,
                  fontWeight: FontWeight.bold),
              ),
            ),
          ),
        ),
      ),
    ),
  ),
);

```

```
}  
}
```

googleAuth.dart:

```
import 'package:firebase_auth/firebase_auth.dart';  
import 'package:flutter/material.dart';  
import 'package:google_sign_in/google_sign_in.dart';  
  
class GoogleAuthentication {  
  static SnackBar customSnackBar({required String content}) {  
    return SnackBar(  
      backgroundColor: Colors.black,  
      content: Text(  
        content,  
        style: TextStyle(color: Colors.redAccent, letterSpacing: 0.5),  
      ),  
    );  
  }  
  
  static Future<User?> signInWithGoogle({required BuildContext context}) async {  
    FirebaseAuth auth = FirebaseAuth.instance;  
    User? user;  
  
    final GoogleSignIn googleSignIn = GoogleSignIn();  
  
    final GoogleSignInAccount? googleSignInAccount =  
      await googleSignIn.signIn();  
  
    if (googleSignInAccount != null) {  
      final GoogleSignInAuthentication googleSignInAuthentication =  
        await googleSignInAccount.authentication;  
  
      final AuthCredential credential = GoogleAuthProvider.credential(  
        accessToken: googleSignInAuthentication.accessToken,  
        idToken: googleSignInAuthentication.idToken,  
      );  
  
      try {  
        final UserCredential userCredential =  
          await auth.signInWithCredential(credential);  
      }  
    }  
  }  
}
```

```

        user = userCredential.user;
    } on FirebaseAuthException catch (e) {
        if (e.code == 'account-exists-with-different-credential') {
            ScaffoldMessenger.of(context).showSnackBar(
                customSnackBar(
                    content: 'account-exists-with-different-credential',
                ),
            );
        } else if (e.code == 'invalid-credential') {
            ScaffoldMessenger.of(context).showSnackBar(
                customSnackBar(
                    content: 'invalid-credential',
                ),
            );
        }
    } catch (e) {
        ScaffoldMessenger.of(context).showSnackBar(
            customSnackBar(
                content: 'Login Failed',
            ),
        );
    }
}

return user;
}
}

```

phoneAuth.dart:

```

import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:legacy_progress_dialog/legacy_progress_dialog.dart';
import 'package:olx_clone/services/phoneauth_service.dart';

class PhoneAuthScreen extends StatefulWidget {
    static const String id = 'phone-auth-screen';

    @override
    _PhoneAuthScreenState createState() => _PhoneAuthScreenState();
}

```

```

class _PhoneAuthScreenState extends State<PhoneAuthScreen> {
  bool validate = false;
  TextEditingController countryCodeController = TextEditingController(text: '+91');
  TextEditingController phoneNumberController = TextEditingController();
  String counterText = '0';

```

```

PhoneAuthService _service= PhoneAuthService();

```

```

@override
Widget build(BuildContext context) {

  //Create an instance of ProgressDialog
  ProgressDialog progressDialog = ProgressDialog(
    context: context,
    backgroundColor: Colors.white,
    textColor: Colors.black,
    loadingText:'Please wait' ,
    progressIndicatorColor:Theme.of(context).primaryColor ,

  );

  return Scaffold(
    appBar: AppBar(
      elevation: 1,
      backgroundColor: Colors.white,
      iconTheme: IconThemeData(
        color: Colors.black,
      ),
      title: Text('Login', style: TextStyle(color: Colors.black)),
    ),
    body: Padding(
      padding: const EdgeInsets.all(20.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [

```

```

    SizedBox(height: 40),
    CircleAvatar(
      radius: 30,
      backgroundColor: Colors.red.shade200,
      child: Icon(
        CupertinoIcons.person_alt_circle,
        color: Colors.red,
        size: 60,
      ),
    ),
    SizedBox(height: 12),
    Text(
      'Enter your Phone',
      style: TextStyle(fontSize: 30, fontWeight: FontWeight.bold),
    ),
    SizedBox(height: 10),
    Text(
      'We will send the confirmation code to your phone',
      style: TextStyle(color: Colors.grey),
    ),
    Row(
      children: [
        Expanded(
          flex: 1,
          child: TextFormField(
            controller: countryCodeController,
            enabled: false,
            style: TextStyle(color: Colors.black), // Set the text color here
            decoration: InputDecoration(
              labelText: 'Country',
            ),
          ),
        ),
        SizedBox(width: 10),
        Expanded(
          flex: 3,
          child: TextFormField(
            onChanged: (value) {
              setState(() {
                counterText = value.length.toString();
                if (value.length == 10) {

```

```

        validate = true;
      }
      if (value.length < 10) {
        setState(() {
          validate = false;
        });
      }
    });
  },
  autofocus: true,
  maxLength: 10,
  keyboardType: TextInputType.phone,
  controller: phoneNumberController,
  decoration: InputDecoration(
    contentPadding: EdgeInsets.only(bottom: 22, top: 22),
    counterText: '$counterText / 10',
    counterStyle: TextStyle(fontSize: 10),
    labelText: 'Number',
    hintText: 'Enter your phone number',
    hintStyle: TextStyle(fontSize: 10, color: Colors.grey),
  ),
),
),
),
],
),
],
),
),
bottomNavigationBar: SafeArea(
  child: Padding(
    padding: const EdgeInsets.all(12.0),
    child: AbsorbPointer(
      absorbing: validate ? false : true,
      child: ElevatedButton(
        style: ButtonStyle(
          backgroundColor: validate
            ? MaterialStateProperty.all(Theme.of(context).primaryColor)
            : MaterialStateProperty.all(Colors.grey),
        ),
        onPressed: () {

```

```

        String number = '${countryCodeController.text}${phoneNumberController.text}';
        progressDialog.show();
        _service.verifyPhoneNumber(context, number);

    },

    child: Padding(
      padding: const EdgeInsets.all(12.0),
      child: Text(
        'Next',
        style: TextStyle(color: Colors.white, fontWeight: FontWeight.bold),
      ),
    ),
  ),
),
),
),
),
),
),
);
}
}

```

otpscreen.dart:

```

import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/services.dart';
import 'package:olx_clone/screens/authentication/phoneauth_screen.dart';
import 'package:olx_clone/screens/location_screen.dart';
import 'package:olx_clone/services/phoneauth_service.dart';

```

```

class OTPScreen extends StatefulWidget {
  final String number,verId;

```

```

  OTPScreen({this.number="",this.verId=""});

```

```

  @override
  State<OTPScreen> createState() => _OTPScreenState();
}

```

```

class _OTPScreenState extends State<OTPScreen> {
  bool _loading=false;

```

```
String error="";
```

```
PhoneAuthService _services=PhoneAuthService();
```

```
var _text1 = TextEditingController();
```

```
var _text2 = TextEditingController();
```

```
var _text3 = TextEditingController();
```

```
var _text4 = TextEditingController();
```

```
var _text5 = TextEditingController();
```

```
var _text6 = TextEditingController();
```

```
Future<void>phoneCredential(BuildContext context,String otp,)async{
```

```
  FirebaseAuth _auth=FirebaseAuth.instance;
```

```
  try{
```

```
    PhoneAuthCredential credential = PhoneAuthProvider.credential(  
      verificationId:widget.verId, smsCode:otp );
```

```
    final User? user=(await _auth.signInWithCredential(credential)).user;
```

```
    if(user!=null){
```

```
      _services.addUser(context,user.uid);
```

```
  }else {
```

```
    print("Login Failed");
```

```
    if (mounted) {
```

```
      setState() {
```

```
        error = "Login Failed";
```

```
      });
```

```
    }
```

```
  }
```

```
}catch(e){
```

```
  print(e.toString());
```

```
  if(mounted){
```

```
    setState() {
```

```
      error="Invalid OTP";
```

```
    });
```

```
  }
```

```
}
```

```
}
```

```
@override
```

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

```
  final node=FocusScope.of(context);
```

```
  return Scaffold(
```



```

appBar: AppBar(
  elevation: 1,
  backgroundColor: Colors.white,
  title: Text('Login', style: TextStyle(color: Colors.black)),
  automaticallyImplyLeading: false ,
),
body: Padding(
  padding: const EdgeInsets.only(left: 20, right: 20),
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start ,
    children: [
      SizedBox(height: 40,),
      CircleAvatar(
        radius: 30,
        backgroundColor: Colors.red.shade200,
        child: Icon(CupertinoIcons.person_alt_circle,
          size: 60,
          color: Colors.red,
        ),
      ),
      SizedBox(height: 10,),
      Text(
        "Welcome Back",
        style: TextStyle(fontWeight: FontWeight.bold,
          fontSize: 25,
        ),
      ),
      SizedBox(height: 10,),
      Row(
        children: [
          Expanded(
            child: RichText(
              text: TextSpan(text: 'We sent a 6-digit code to ',
                style: TextStyle(color: Colors.grey,
                  fontSize: 12),
              children: [
                TextSpan(
                  text: widget.number,
                  style: TextStyle(fontWeight: FontWeight.bold,
                    color: Colors.black,

```

```

        fontSize: 12)
    ),
  ],),

),
),
InkWell(
  onTap:(){
    Navigator.push(
      context, MaterialPageRoute(builder: (context) => PhoneAuthScreen(),),);

  },
  child: Icon(Icons.edit)),

],
),
 SizedBox(height:12,),
Row(
  children: [
    Expanded(
      child: TextFormField(
        textAlign: TextAlign.center,
        controller:_text1 ,
        keyboardType: TextInputType.number,
        textInputAction: TextInputAction.next,
        decoration: InputDecoration(
          border:OutlineInputBorder()
        ),
        onChanged:(value){
          if(value.length==1){
            node.nextFocus();
          }
        },
      ),
    ),
    ),
    SizedBox(width: 10,),
    Expanded(
      child: TextFormField(
        textAlign: TextAlign.center,
        controller:_text2 ,
        keyboardType: TextInputType.number,

```

```
textInputAction: TextInputAction.next,  
decoration: InputDecoration(  
    border:OutlineInputBorder()  
),  
onChanged:(value){  
    if(value.length==1){  
        node.nextFocus();  
    }  
},  
),  
),  
SizedBox(width: 10,),  
Expanded(  
    child: TextFormField(  
        textAlign: TextAlign.center,  
        controller:_text3,  
        keyboardType: TextInputType.number,  
        textInputAction: TextInputAction.next,  
        decoration: InputDecoration(  
            border:OutlineInputBorder()  
        ),  
        onChanged:(value){  
            if(value.length==1){  
                node.nextFocus();  
            }  
        },  
    ),  
),  
SizedBox(width: 10,),  
Expanded(  
    child: TextFormField(  
        textAlign: TextAlign.center,  
        controller:_text4 ,  
        keyboardType: TextInputType.number,  
        textInputAction: TextInputAction.next,  
        decoration: InputDecoration(  
            border:OutlineInputBorder()  
        ),  
        onChanged:(value){  
            if(value.length==1){  
                node.nextFocus();
```

```
}  
},  
) ,  
) ,  
SizedBox(width: 10,),  
Expanded(  
  child: TextFormField(  
    textAlign: TextAlign.center,  
    controller:_text5 ,  
    keyboardType: TextInputType.number,  
    textInputAction: TextInputAction.next,  
    decoration: InputDecoration(  
      border:OutlineInputBorder()  
    ),  
    onChanged:(value){  
      if(value.length==1){  
        node.nextFocus();  
      }  
    },  
  ),  
) ,  
SizedBox(width: 10,),  
Expanded(  
  child: TextFormField(  
    textAlign: TextAlign.center,  
    controller:_text6 ,  
    keyboardType: TextInputType.number,  
    textInputAction: TextInputAction.next,  
    decoration: InputDecoration(  
      border:OutlineInputBorder(  
  
    )  
  
  ),  
  onChanged:(value){  
    if(value.length==1){  
      if (_text1.text.length==1){  
        if (_text2.text.length==1){  
          if (_text3.text.length==1){  
            if (_text4.text.length==1){  
              if (_text5.text.length==1){
```

```

        String _otp='${_text1.text}${_text2.text}'
        '${_text3.text}${_text4.text}${_text5.text}${_text6.text}';
        setState(() {
          _loading=true;
        });
        //login
        phoneCredential(context, _otp);
      }
    }
  }
}

}

}else{
  setState() {
    _loading=false;
  });
}
},
),
),

],
),
  SizedBox(height: 18,),
  if(_loading)
  Align(
    alignment: Alignment.center,
    child: SizedBox(
      width: 50,
      child: LinearProgressIndicator(
        backgroundColor:Colors.grey.shade200 ,
valueColor:AlwaysStoppedAnimation<Color>(Theme.of(context).primaryColor) ,
      ),
    ),
  ),
  SizedBox(height: 18,),
  Text(error,style: TextStyle(color: Colors.red,fontSize: 12),)

```

```

    ],
  ),
),
);
}
}

```

AuthUI.dart:

```

import 'package:flutter/material.dart';
import 'package:flutter_signin_button/button_list.dart';
import 'package:flutter_signin_button/button_view.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:olx_clone/screens/authentication/google_auth.dart';
import 'package:olx_clone/screens/authentication/phoneauth_screen.dart';
import 'package:olx_clone/screens/authentication/email_auth_screen.dart';
import 'package:olx_clone/services/phoneauth_service.dart';

```

```

class AuthUi extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.min,
        children: [
          SizedBox(
            width: 220,
            child: ElevatedButton(
              style: ElevatedButton.styleFrom(
                primary: Colors.white,
                shape: new RoundedRectangleBorder(
                  borderRadius: new BorderRadius.circular(3.0),
                ),
              ),
              onPressed: () {
                Navigator.pushNamed(context, PhoneAuthScreen.id);
              },
            child: Row(
              children: [
                Icon(Icons.phone_android_outlined, color: Colors.black),
                SizedBox(width: 8),

```

```

        Text('Continue with Phone', style: TextStyle(color: Colors.black)),
      ],
    ),
  ),
  SignInButton(
    Buttons.Google,
    text: ('Continue with Google'),
    onPressed: () async {
      User? user = await GoogleAuthentication.signInWithGoogle(context: context);
      if (user != null) {
        PhoneAuthService _authentication = PhoneAuthService();
        _authentication.addUser(context, user.uid);
      }
    },
  ),
  SignInButton(
    Buttons.FacebookNew,
    text: ('Continue with Facebook'),
    onPressed: () {},
  ),
  InkWell(
    onTap: () {
      Navigator.pushNamed(context, EmailAuthScreen.id);
    },
    child: Container(
      child: Text(
        'Login with Email',
        style: TextStyle(
          color: Colors.white,
          fontWeight: FontWeight.bold,
          fontSize: 18,
        ),
      ),
    ),
    decoration: BoxDecoration(
      border: Border(bottom: BorderSide(color: Colors.white)),
    ),
  ),
),
],
),

```

```
);  
}  
}
```

main.dart:

```
import 'package:firebase_core/firebase_core.dart';  
import 'package:flutter/material.dart';  
import 'package:olx_clone/screens/authentication/email_auth_screen.dart';  
import 'package:olx_clone/screens/authentication/phoneauth_screen.dart';  
import 'package:olx_clone/screens/home_screen.dart';  
import 'package:olx_clone/screens/login_screen.dart';  
import 'package:olx_clone/screens/splash_screen.dart';  
import 'package:olx_clone/screens/location_screen.dart';  
import 'dart:io';
```

```
void main() async {  
  WidgetsFlutterBinding.ensureInitialized();  
  Platform.isAndroid ?await Firebase.initializeApp(  
    options: const FirebaseOptions(  
      apiKey: "AIzaSyDuFtAm59795zlrpp16kTJTbd1Lac95KkI",  
      appId: "1:232516563209:android:d2dea587c60de17bac63d8",  
      messagingSenderId:"232516563209" ,  
      projectId: "olxclone-12618"  
    )  
  ):await Firebase.initializeApp();  
  runApp(MyApp());  
}
```

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      debugShowCheckedModeBanner: false,  
      theme: ThemeData(  
        primaryColor: Colors.cyan.shade900,  
        fontFamily: 'Horizon',  
      ),  
      initialRoute: SplashScreen.id,  
    );  
  }  
}
```



```
routes: {  
  SplashScreen.id: (context) => SplashScreen(),  
  LoginScreen.id: (context) => LoginScreen(),  
  PhoneAuthScreen.id: (context) => PhoneAuthScreen(),  
  LocationScreen.id: (context) => LocationScreen(),  
  HomeScreen.id: (context) => HomeScreen(locationData: null),  
  EmailAuthScreen.id: (context) => EmailAuthScreen(),  
},  
  
);
```

pubsec.yaml:

name: olx_clone

description: "A new Flutter project."

publish_to: 'none' # Remove this line if you wish to publish to pub.dev

version: 1.0.0+1

environment:

sdk: '>=3.2.5 <4.0.0'

dependencies:

flutter:

sdk: flutter

cupertino_icons: ^1.0.2

cloud_firestore: ^4.15.3

animated_text_kit: ^4.2.2

flutter_signin_button: ^2.0.0

firebase_core: ^2.24.2

firebase_auth: ^4.16.0

legacy_progress_dialog: ^0.1.2

location: ^5.0.3

flutter_geocoder: ^0.2.2-nullsafety

google_sign_in: ^6.2.1

email_validator: ^2.1.17

flutter_neumorphic_plus: ^3.3.0

dev_dependencies:

flutter_test:
 sdk: flutter

flutter_lints: ^2.0.0

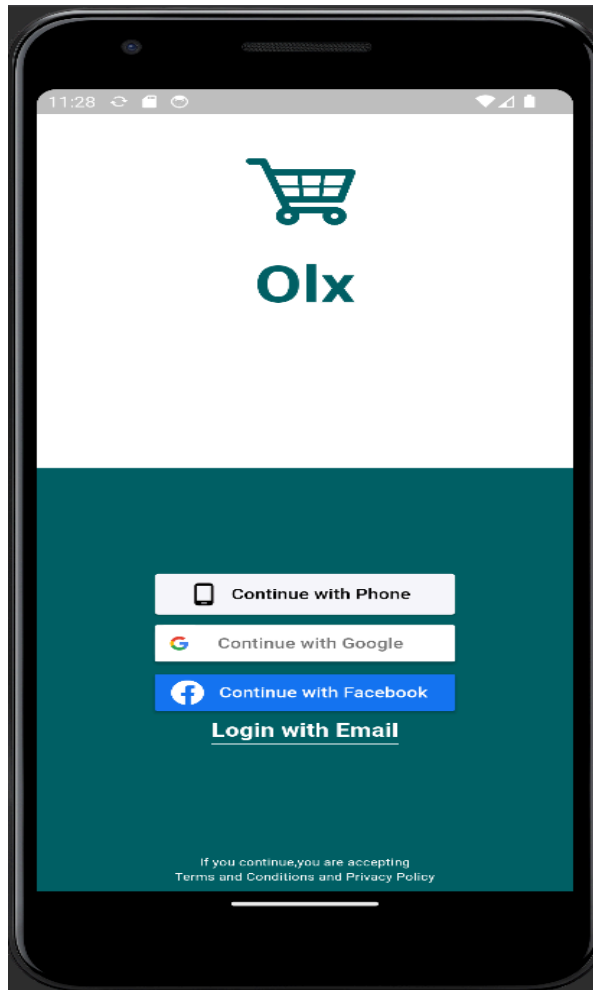
flutter:

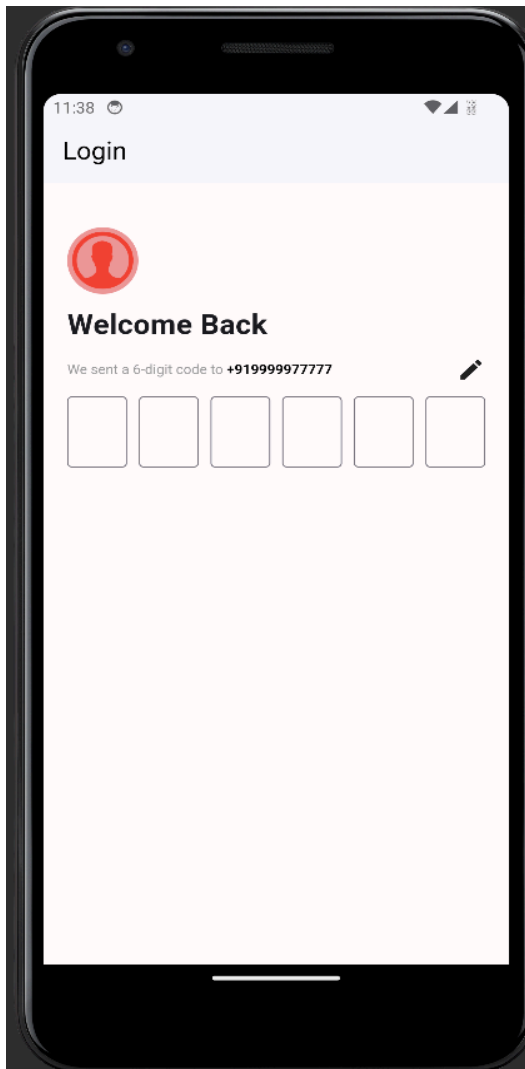
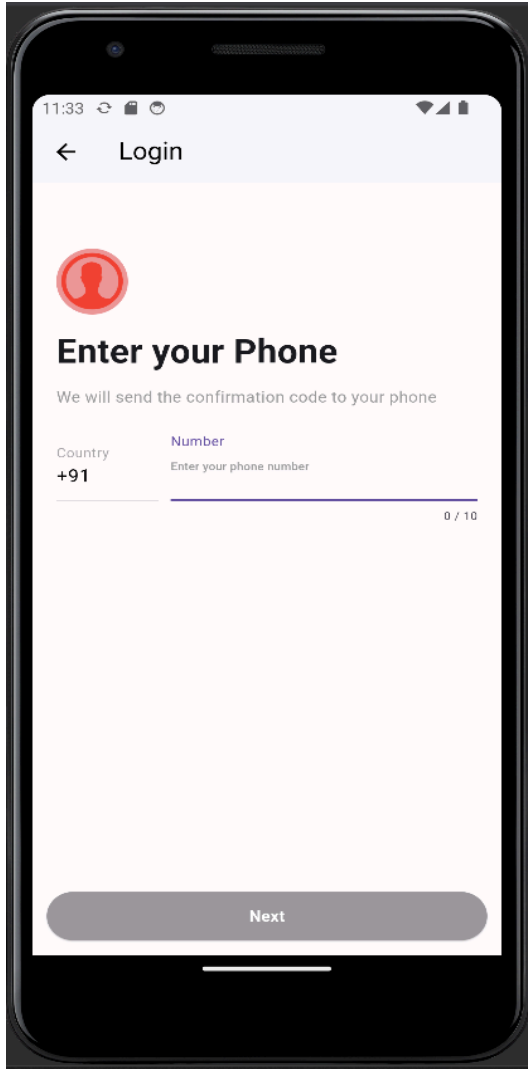
uses-material-design: true

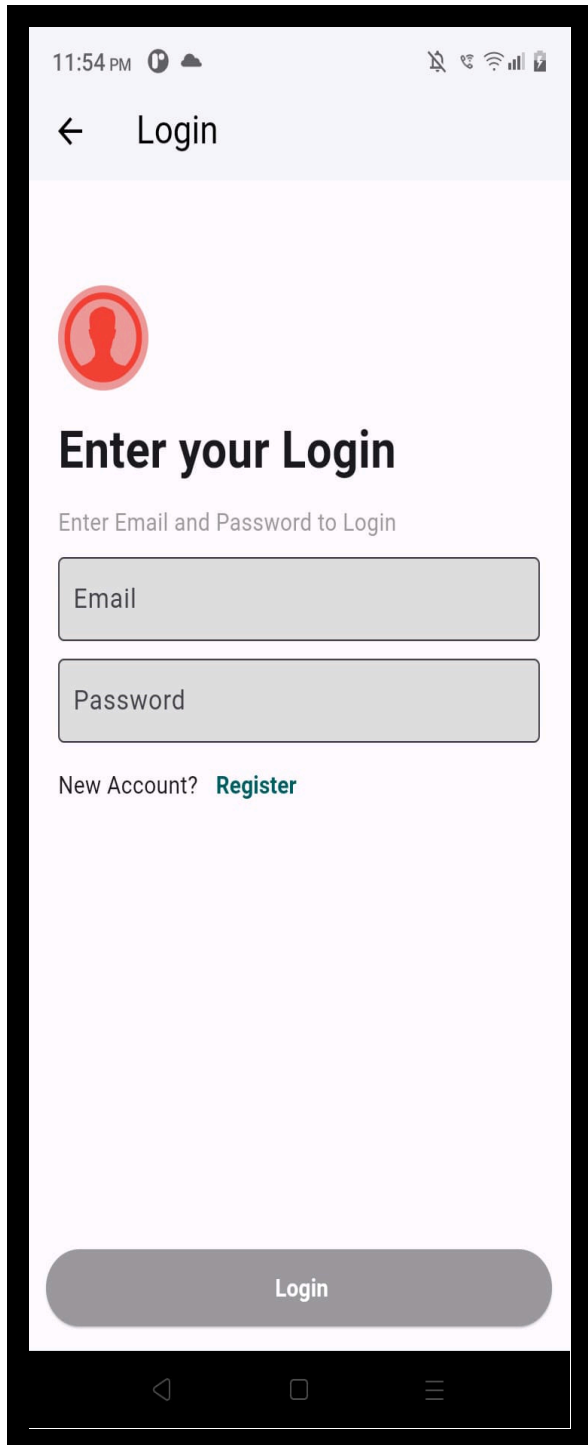
assets:

- assets/images/cart.png
- assets/images/map.png

OUTPUT:








11:54 PM

← Login



Enter your Login

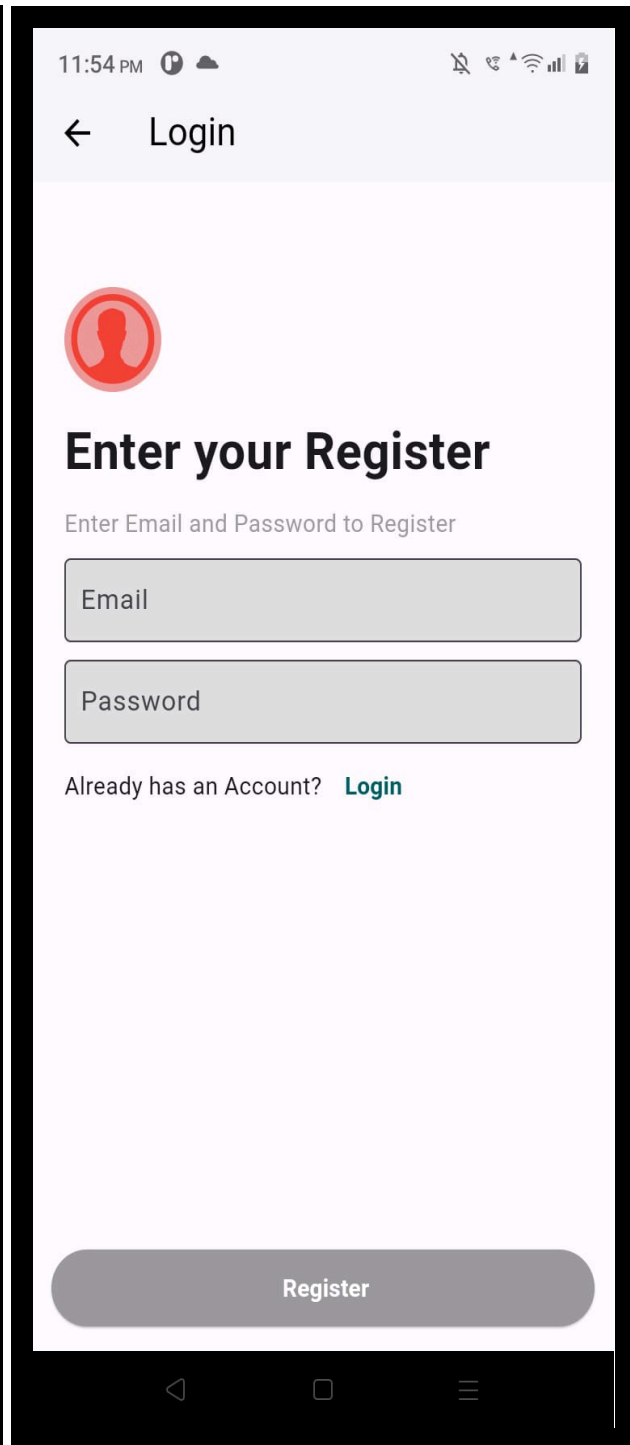
Enter Email and Password to Login

Email

Password


New Account? [Register](#)

Login



11:54 PM

← Login



Enter your Register

Enter Email and Password to Register

Email

Password

Already has an Account? [Login](#)

Register

CONCLUSION:

We have successfully implemented the form in Flutter UI using widgets.

