**PRACTICAL:04**

**AIM: Create a login app with following features:**

**1 Successful Login message in TextViewwith Green background if Username & password is correct**

**2 Failure message in TextView with Red background if Username or password is incorrect.**

**3 Disable Login Button after three wrong login attempts.**

**4 Close application if user selects Cancel Button.**

**Source Code :**

**Main.dart:**

import 'package:flutter/material.dart';  
import 'home.dart';  
  
void main() {  
 runApp(MyApp());  
}  
class MyApp extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 debugShowCheckedModeBanner: false,  
 home: home(  
 ),  
 );  
 }  
}  
  
class MyHomePage extends StatefulWidget {  
 Color bgcolor;  
 @override  
 \_MyHomePageState createState() => \_MyHomePageState();  
}  
  
class \_MyHomePageState extends State<MyHomePage> {  
 final GlobalKey<FormState> formkey = GlobalKey<FormState>();  
  
 Color bgcolor;  
 bool \_enabled = true;  
 int counter = 3;  
 String email, password;  
  
 void enableButton() {  
 final formstate = formkey.currentState;  
 if (formstate.validate()) {  
 formstate.save();  
 }  
 if (email == 'riya@gmail.com' && password == 'riya') {  
 Navigator.*push*(  
 context,  
 MaterialPageRoute(  
 builder: (context) => second(),  
 ),  
 );  
 } else {  
 setState(() {  
 counter--;  
 });  
 }  
 if (counter == 0) {  
 setState(() {  
 \_enabled = false;  
 Navigator.*push*(  
 context,  
 MaterialPageRoute(  
 builder: (context) => third(),  
 ),  
 );  
 });  
 }  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 body: Form(  
 key: formkey,  
 child: SingleChildScrollView(  
 child: Column(  
 mainAxisAlignment: MainAxisAlignment.center,  
 crossAxisAlignment: CrossAxisAlignment.center,  
 children: <Widget>[  
 Container(  
 child: Image.asset('assets/logo1.png'),  
 ),  
 Text(  
 'Login Page',  
 style: TextStyle(  
 fontWeight: FontWeight.*w900*,  
 fontSize: 50,  
 color: Color(0xFFD7A715),  
 ),  
 ),  
 SizedBox(  
 height: 20,  
 ),  
 Padding(  
 padding: const EdgeInsets.all(16.0),  
 child: TextFormField(  
 decoration: InputDecoration(  
 contentPadding: EdgeInsets.fromLTRB(20.0, 15.0, 20.0, 15.0),  
 hintText: "Email",  
 border: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(30.0),  
 ),  
 ),  
 validator: (String value) {  
 if (value != 'riya@gmail.com') {  
 return 'Enter correct mail';  
 }  
 return null;  
 },  
 onSaved: (value) => email = value,  
 ),  
 ),  
 SizedBox(  
 height: 20,  
 ),  
 Padding(  
 padding: const EdgeInsets.all(16.0),  
 child: TextFormField(  
 obscureText: true,  
 decoration: InputDecoration(  
 contentPadding: EdgeInsets.fromLTRB(20.0, 15.0, 20.0, 15.0),  
 hintText: "Password",  
 border: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(30.0),  
 ),  
 ),  
 validator: (String value) {  
 if (value != 'riya') {  
 return 'wrong password';  
 }  
 return null;  
 },  
 onSaved: (value) => password = value,  
 ),  
 ),  
 SizedBox(  
 height: 30,  
 ),  
 Padding(  
 padding: const EdgeInsets.all(16.0),  
 child: RaisedButton(  
 onPressed: \_enabled ? enableButton : null,  
 shape: RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(20),  
 ),  
 color: Color(0xFFD7A715),  
 child: Text(  
 'Login',  
 style: TextStyle(  
 fontWeight: FontWeight.*bold*,  
 color: Colors.*white*,  
 ),  
 ),  
 ),  
 ),  
 ],  
 ),  
 ),  
 ),  
 );  
 }  
}

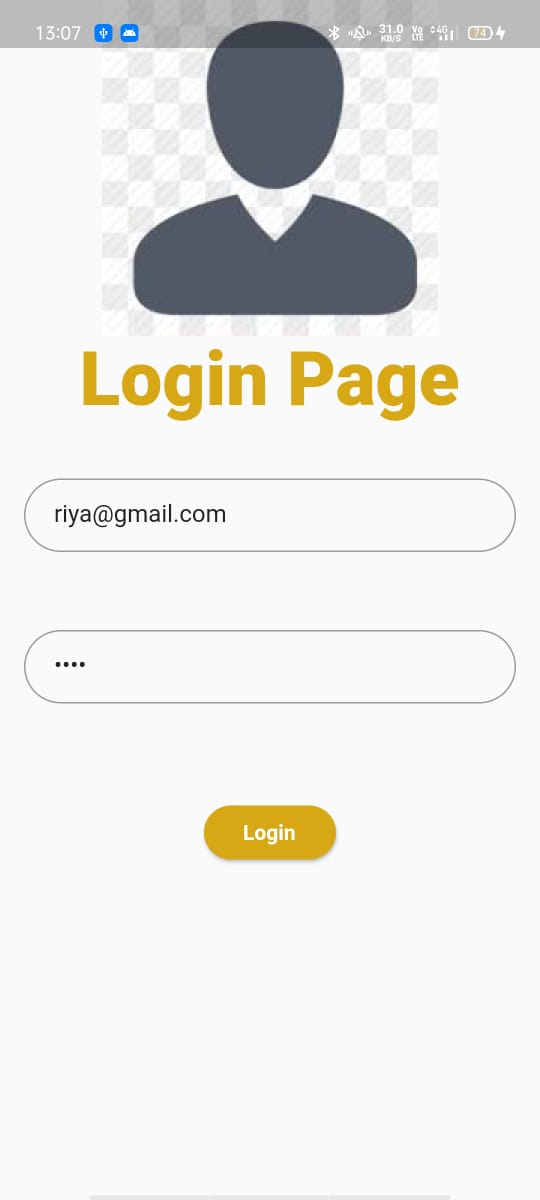
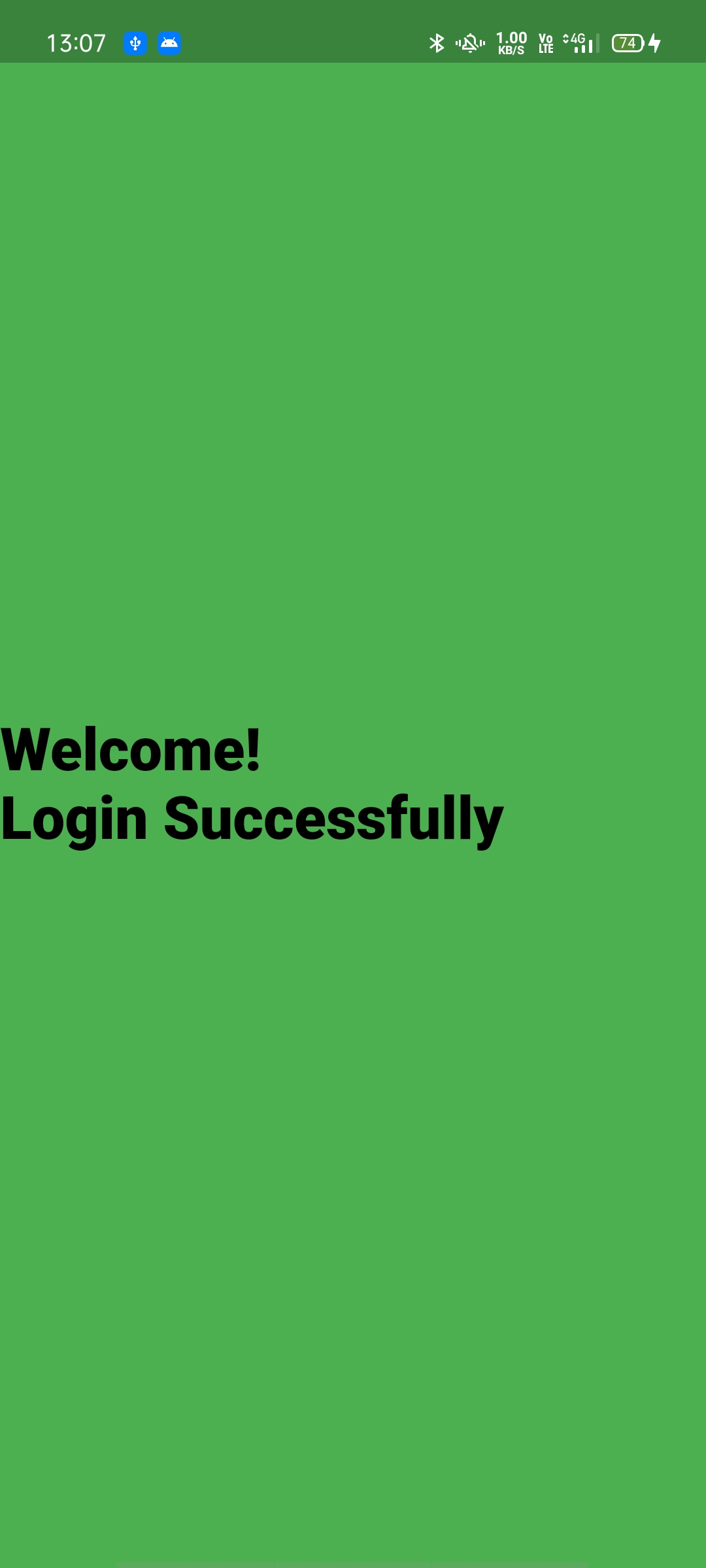
**Home.dart**

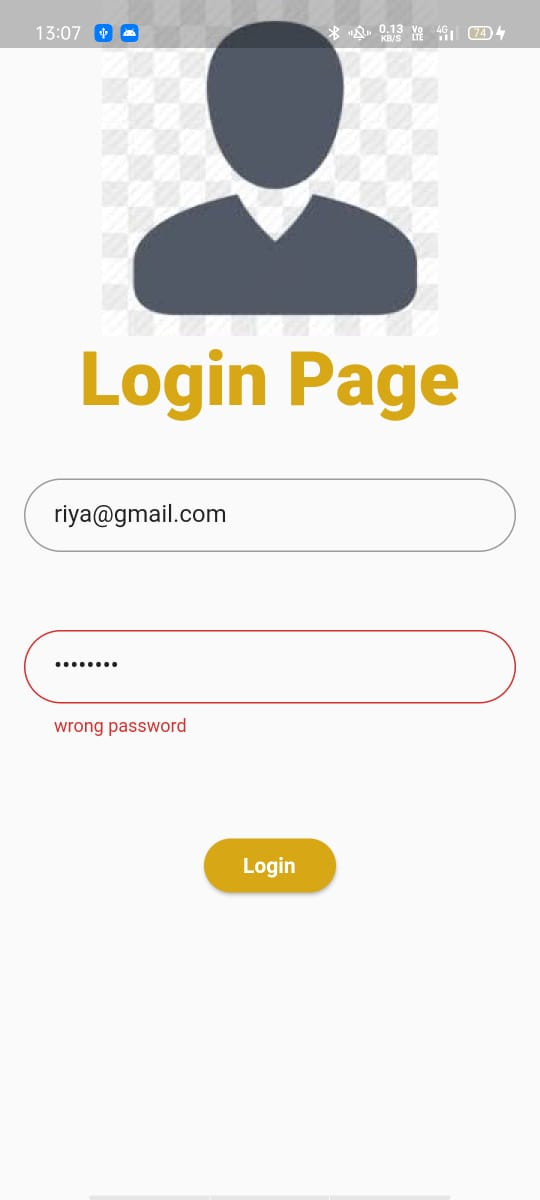
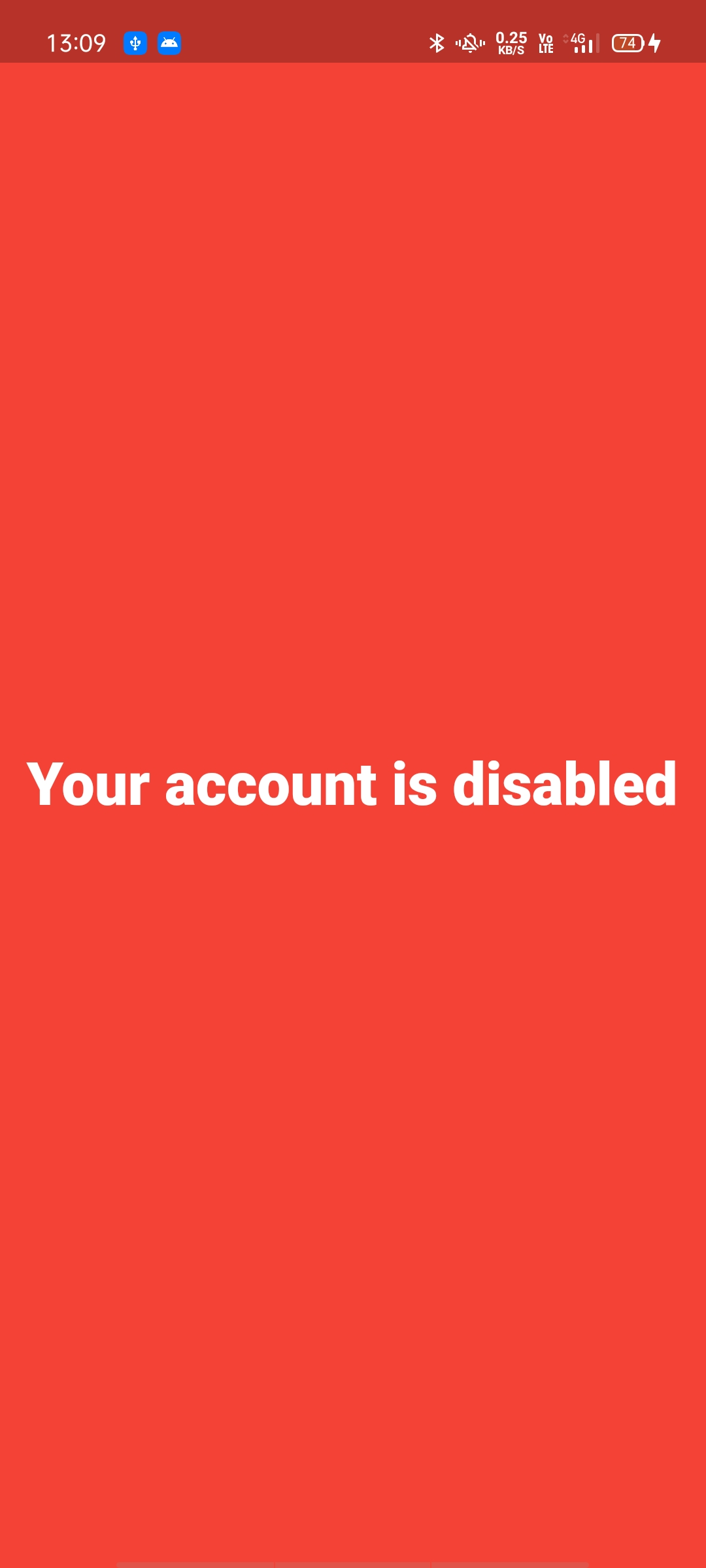
import 'package:flutter/cupertino.dart';  
import 'package:flutter/material.dart';  
import 'package:splashscreen/splashscreen.dart';  
import 'main.dart';  
  
class home extends StatefulWidget {  
 @override  
 \_homeState createState() => \_homeState();  
}  
  
class \_homeState extends State<home> {  
 @override  
 Widget build(BuildContext context) {  
  
 return SplashScreen(  
 image: Image.asset(  
 'assets/logo1.png',  
 ),  
 photoSize: 250,  
 seconds: 3,  
 loaderColor: Colors.*blue*,  
 navigateAfterSeconds: MyHomePage(),  
 );  
 }  
}  
  
class second extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 var scaffold = Scaffold(  
 backgroundColor: Colors.*green*,  
 body: Center(  
 child: Text(  
  
 'Welcome! Login Successfully',  
  
 style: TextStyle(  
 fontWeight: FontWeight.*w900*, color: Colors.*black*, fontSize: 30),  
 ),  
 ),  
 );  
 return scaffold;  
 }  
}  
  
class third extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 backgroundColor: Colors.*red*,  
 body: Center(  
 child: Text(  
 'Your account is disabled',  
 style: TextStyle(  
 fontWeight: FontWeight.*w900*, color: Colors.*white*, fontSize: 30),  
 ),  
 ),  
 );  
 }  
}

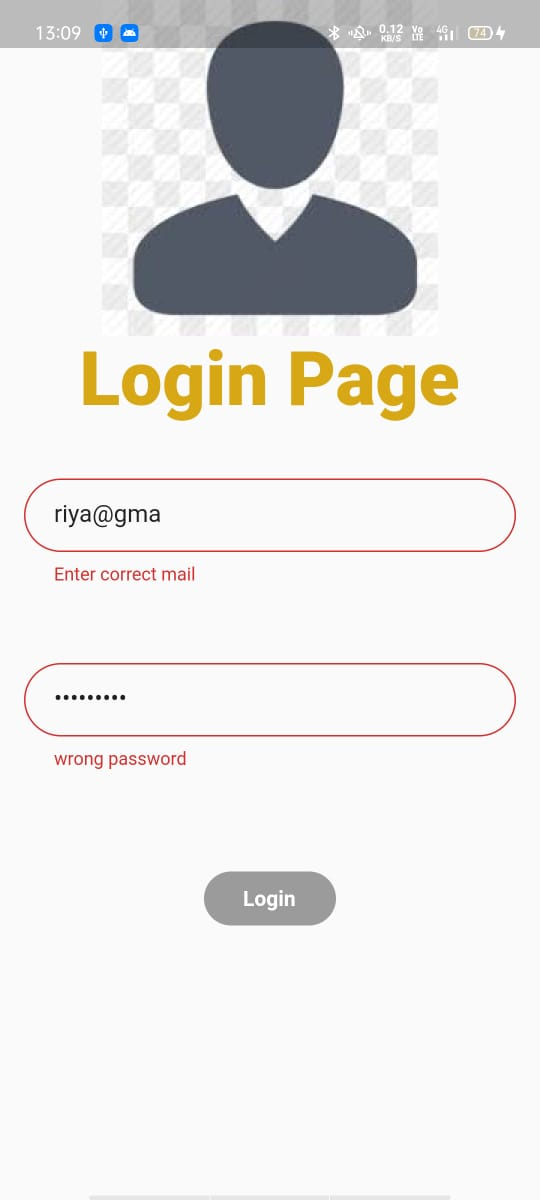
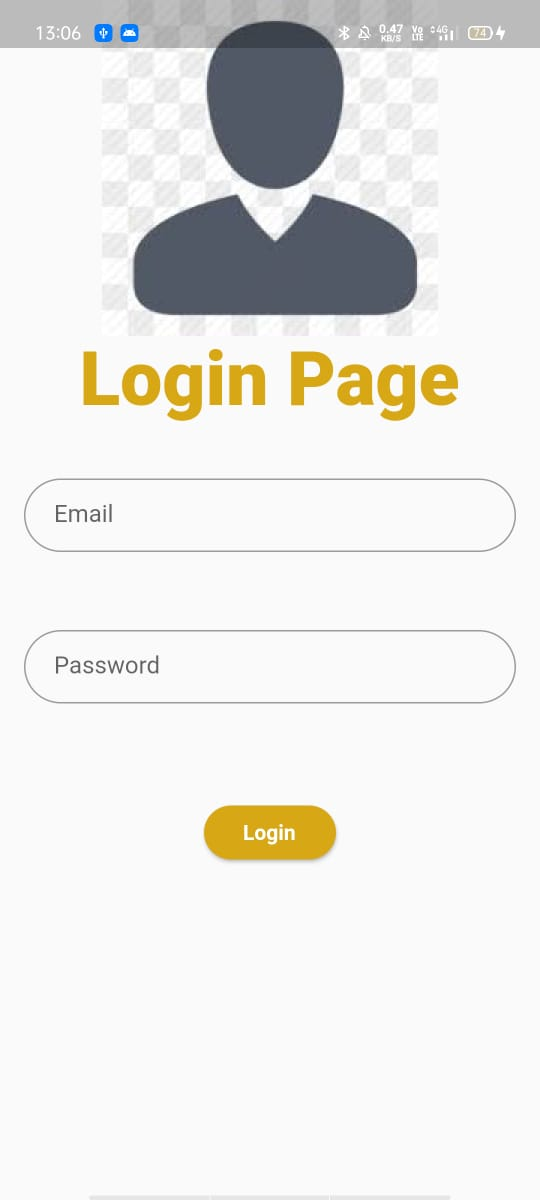
**Pubsec.yaml**

name: Practical\_5  
description: A new Login Flutter application.  
  
*# The following line prevents the package from being accidentally published to  
# pub.dev using `pub publish`. This is preferred for private packages.*publish\_to: 'none' *# Remove this line if you wish to publish to pub.dev  
  
# The following defines the version and build number for your application.  
# A version number is three numbers separated by dots, like 1.2.43  
# followed by an optional build number separated by a +.  
# Both the version and the builder number may be overridden in flutter  
# build by specifying --build-name and --build-number, respectively.  
# In Android, build-name is used as versionName while build-number used as versionCode.  
# Read more about Android versioning at https://developer.android.com/studio/publish/versioning  
# In iOS, build-name is used as CFBundleShortVersionString while build-number used as CFBundleVersion.  
# Read more about iOS versioning at  
# https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html*version: 1.0.0+1  
  
environment:  
 sdk: ">=2.7.0 <3.0.0"  
  
dependencies:  
 flutter:  
 sdk: flutter  
  
  
 *# The following adds the Cupertino Icons font to your application.  
 # Use with the CupertinoIcons class for iOS style icons.* cupertino\_icons: ^1.0.0  
 splashscreen: ^1.3.5  
  
dev\_dependencies:  
 flutter\_test:  
 sdk: flutter  
  
*# For information on the generic Dart part of this file, see the  
# following page: https://dart.dev/tools/pub/pubspec  
  
# The following section is specific to Flutter.*flutter:  
  
 *# The following line ensures that the Material Icons font is  
 # included with your application, so that you can use the icons in  
 # the material Icons class.* uses-material-design: true  
  
 *# To add assets to your application, add an assets section, like this:* assets:  
 - assets/logo1.png  
  
 *# - images/a\_dot\_ham.jpeg  
  
 # An image asset can refer to one or more resolution-specific "variants", see  
 # https://flutter.dev/assets-and-images/#resolution-aware.  
  
 # For details regarding adding assets from package dependencies, see  
 # https://flutter.dev/assets-and-images/#from-packages  
  
 # To add custom fonts to your application, add a fonts section here,  
 # in this "flutter" section. Each entry in this list should have a  
 # "family" key with the font family name, and a "fonts" key with a  
 # list giving the asset and other descriptors for the font. For  
 # example:  
 # fonts:  
 # - family: Schyler  
 # fonts:  
 # - asset: fonts/Schyler-Regular.ttf  
 # - asset: fonts/Schyler-Italic.ttf  
 # style: italic  
 # - family: Trajan Pro  
 # fonts:  
 # - asset: fonts/TrajanPro.ttf  
 # - asset: fonts/TrajanPro\_Bold.ttf  
 # weight: 700  
 #  
 # For details regarding fonts from package dependencies,  
 # see https://flutter.dev/custom-fonts/#from-packages*

**Output:**

**Learning Outcome:**

In this Practical, I have learned about how to give the validation to the login page using different functionalites.