Text

Description automatically generated

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Assignment No** | Assignment No 05 |

Assignment Number - 05

**Title :** Design Flutter Application for Login page

**Theory :**

**Form Validation**

Sample code for validation

TextFormField(  
 *// The validator receives the text that the user has entered.* validator: (value) {  
 if (value == null || value.isEmpty) {  
 return 'Please enter some text';  
 }  
 return null;  
 },  
),

**Email Validation**

**Regular expression for an email address :**

**^([a-zA-Z0-9\_\-\.]+)@([a-zA-Z0-9\_\-\.]+)\.([a-zA-Z]{2,5})$**

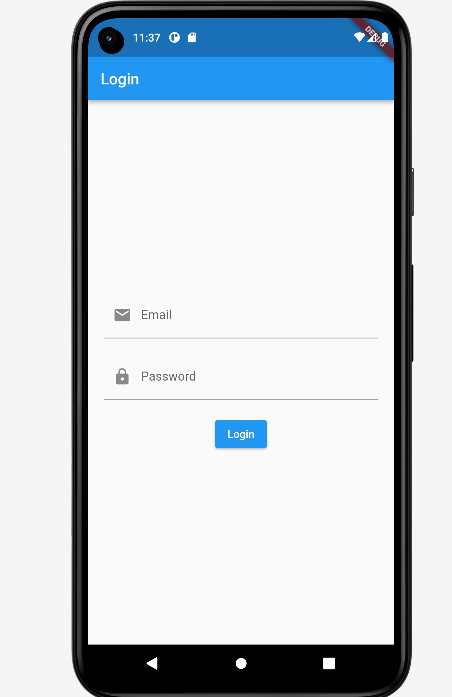
**Flutter Login Screen**

The login screen we are going to build is simple in its visual aspects. First there is a widget for the company/organization/app name. Then about the screen itself, Sign in. Now, we have two text fields, user name and password, to get login/sign-in credentials from user. Then we have a TextButton widget for the Forgot Password. After that, there is an ElevatedButton widget for the Login button. If the user does not have an account, there should be a provision for the Sign-up process, hence a Sign-up TextButton.

**Source Code:-**

import 'package:flutter/material.dart';  
  
void main() => runApp(LoginApp());  
  
class LoginApp extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 title: 'Login Page',  
 theme: ThemeData(primarySwatch: Colors.*blue*),  
 home: LoginPage(),  
 );  
 }  
}  
  
class LoginPage extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(title: Text('Login')),  
 body: Container(  
 padding: EdgeInsets.all(20.0),  
 child: Column(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 TextField(  
 decoration: InputDecoration(  
 labelText: 'Email',  
 prefixIcon: Icon(Icons.*email*),  
 ),  
 ),  
 SizedBox(height: 20.0),  
 TextField(  
 obscureText: true,  
 decoration: InputDecoration(  
 labelText: 'Password',  
 prefixIcon: Icon(Icons.*lock*),  
 ),  
 ),  
 SizedBox(height: 20.0),  
 ElevatedButton(  
 onPressed: () {},  
 child: Text('Login'),  
 ),  
 ],  
 ),  
 ),  
 );  
 }  
}

**Output:**



**Conclusion :**

**In this assignment I have learn about , a basic login page UI is created using the Flutter framework. The LoginPage class is a StatelessWidget that defines the UI elements for the login page.**

**The login page consists of an AppBar at the top with a title, and a Container in the body that holds the login form. The form includes two TextField widgets for entering the email and password. The email field has an email icon as a prefix, and the password field has a lock icon. The SizedBox widget is used to create spacing between the form elements.**

**Finally, an ElevatedButton is added for the login button. Currently, the button's onPressed callback is empty, but you can add your own logic for handling the login functionality.**