Experiment No: 4

Aim: Design an Interactive Form using form Widget and Navigation

LO2: Develop the App UI by incorporating widgets, layouts, gestures and animation Theory:

Theory:

What is Flutter form?

Flutter forms are a way to collect user input in a structured way. Forms in Flutter are typically composed of input widgets such as text fields, checkboxes, radio buttons, and dropdowns. Forms are useful for gathering user information such as names, email addresses, phone numbers, and more.

Step for creating form:

Here are the steps to create a form in Flutter:

- Import the necessary packages: To create a form in Flutter, you will need to import the flutter/material.dart package, which provides the TextFormField widget, and the flutter/widgets.dart package, which provides the Form widget.

- Create a Form widget: The Form widget is the container for your input fields. You can create a Form widget by instantiating the Form class.

- Create input fields: Inside the Form widget, you will need to create the input fields that you want to include in your form. For example, you can create a TextFormField widget for a text input field.

- Add validators: You can add validators to your input fields to ensure that the user enters valid input. Validators are functions that take the input value as an argument and return an error message if the input is invalid.

- Handle form submission: When the user submits the form, you can validate the input fields and then take some action based on the input. For example, you could send the input to a server, save it to a database, or display it on the screen.

- Wrap input fields in a StatefulWidget: If you want to update the UI in response to user input, you will need to wrap your input fields in a StatefulWidget. This will allow you to store the input values in the widget's state and update the UI when the state changes.

- Implement the onSaved callback: You can use the onSaved callback to save the input values to a variable or data structure when the user submits the form.

How to create a drawer in Flutter?

In Flutter, a drawer is a slide-out panel that displays navigation or other options to the user. Here's how to create a drawer in Flutter:

- Add a Scaffold widget: A Scaffold widget provides the basic structure for your app's layout, including the app bar, body, and drawer. You can add a Scaffold widget to your app by instantiating the Scaffold class.

- Add an AppBar widget: The AppBar widget provides the top bar of your app, which typically includes the app title and navigation buttons. You can add an AppBar widget to your Scaffold widget by setting the appBar property to an instance of the AppBar class.

- Add a Drawer widget: The Drawer widget provides the slide-out panel that displays navigate or other options to the user. You can add a Drawer widget to your Scaffold widget by setting the drawer property to an instance of the Drawer class.

- Add items to the drawer: You can add items to the drawer by using the ListView widget, and provides a scrolling list of items. You can add a ListView widget to the Drawer widget

Code:

Main.Dart Code:

```
import 'package:flutter/material.dart';
import 'package:form/login.dart';
void main() {
  runApp(OmkarApp());
}
class OmkarApp extends StatelessWidget {
  const OmkarApp({Key? key}) : super(key: key);
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home:LoginPage() ,
    );
  }
}
```

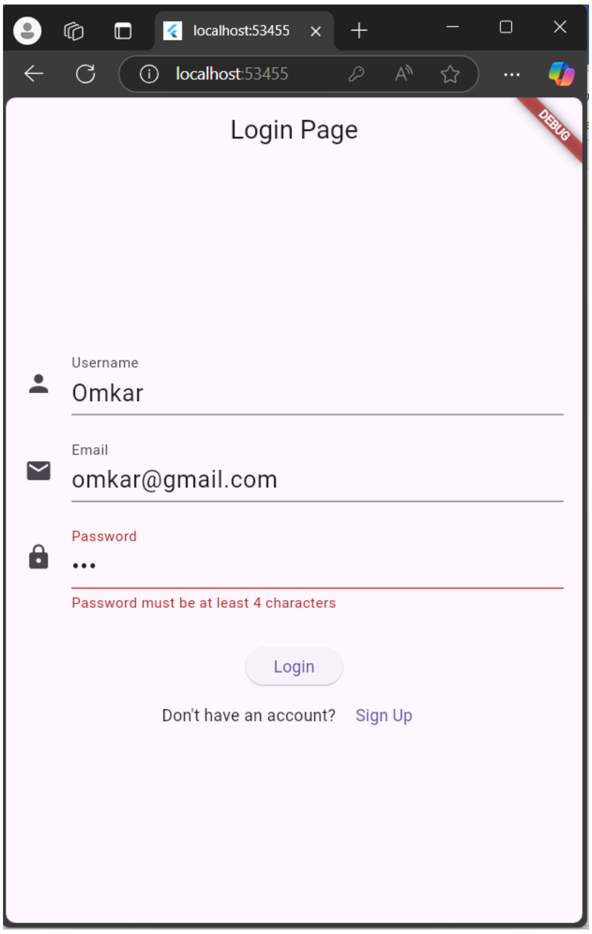
Login Code:

```
import 'package:flutter/material.dart';
import 'package:form/signup.dart';
class LoginPage extends StatefulWidget {
  const LoginPage({Key? key}) : super(key: key);
  @override
  State<LoginPage> createState() => _LoginPageState();
}
class _LoginPageState extends State<LoginPage> {
  final _formKey = GlobalKey<FormState>();
  String _username = "";
  String _email = "";
  String _password = "";
  void _validateAndSubmit() {
    if (_formKey.currentState!.validate()) {
      if (_username.isEmpty || _email.isEmpty || _password.isEmpty) {
        ScaffoldMessenger.of(context).showSnackBar(
          const SnackBar(
            content: Text("All fields are required!"),
            backgroundColor: Colors.red,
          ),
        );
      } else {
        Navigator.push(
          context,
          MaterialPageRoute(builder: (context) => const SignUpPage()),
        );
      }
    }
  }
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: const Text("Login Page"),
          centerTitle: true,
        ),
        body: Container(
          decoration: const BoxDecoration(
            image: DecorationImage(
              image: AssetImage('assest/image/pic.jpg'), // Background image
              fit: BoxFit.cover,
            ),
          ),
          child: Padding(
            padding: const EdgeInsets.all(16.0),
            child: Form(
              key: _formKey,
              child: Column(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
                  TextFormField(
                    style: const TextStyle(fontSize: 20, fontFamily: "Times New Roman"),
                    decoration: const InputDecoration(
                      labelText: "Username",
                      icon: Icon(Icons.person),
                      hintText: "Enter your username",
                    ),
                    validator: (value) {
                      if (value == null || value.isEmpty) {
                        return "Username cannot be empty";
                      } else if (value.length < 4) {
                        return "Enter a valid username (min 4 chars)";
                      }
                      return null;
                    },
                    onChanged: (value) => _username = value,
                  ),
                  const SizedBox(height: 20),
                  TextFormField(
                    style: const TextStyle(fontSize: 20, fontFamily: "Times New Roman"),
                    decoration: const InputDecoration(
                      labelText: "Email",
                      icon: Icon(Icons.email),
                      hintText: "Enter your email",
                    ),
                    validator: (value) {
                      if (value == null || value.isEmpty) {
                        return "Email cannot be empty";
                      } else if (!RegExp(r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$')
                          .hasMatch(value)) {
                        return "Enter a valid email address";
                      }
                      return null;
                    },
                    onChanged: (value) => _email = value,
                  ),
                  const SizedBox(height: 20),
                  TextFormField(
                    obscureText: true,
                    style: const TextStyle(fontSize: 20, fontFamily: "Times New Roman"),
                    decoration: const InputDecoration(
                      labelText: "Password",
                      icon: Icon(Icons.lock),
                      hintText: "Enter your password",
                    ),
                    validator: (value) {
                      if (value == null || value.isEmpty) {
                        return "Password cannot be empty";
                      } else if (value.length < 4) {
                        return "Password must be at least 4 characters";
                      }
                      return null;
                    },
                    onChanged: (value) => _password = value,
                  ),
                  const SizedBox(height: 30),
                  ElevatedButton(
                    onPressed: _validateAndSubmit,
                    child: const Text("Login"),
                  ),
                  const SizedBox(height: 10),
                  Row(
                    mainAxisAlignment: MainAxisAlignment.center,
                    children: [
                      const Text("Don't have an account?"),
                      const SizedBox(width: 5),
                                        TextButton(
                                          onPressed: () {
                                            Navigator.push(
                                              context,
                                              MaterialPageRoute(builder: (context) => const SignUpPage()),
                                            );
                                          },
                                          child: const Text("Sign Up"),
                                        ),
                                      ],
                                    ),
                                  ],
                                ),
                              ),
                            ),
                          ),
                        ),
                      );
                    }
                  }
```

SignUp Code:

```
import 'package:flutter/material.dart';
import 'package:form/login.dart';
class SignUpPage extends StatefulWidget {
  const SignUpPage({Key? key}) : super(key: key);
  @override
  State<SignUpPage> createState() => _SignUpPageState();
}
class _SignUpPageState extends State<SignUpPage> {
  final _formKey = GlobalKey<FormState>();
  String _username = "";
  String _email = "";
  String _password = "";
  void _signUp() {
    if (_formKey.currentState!.validate()) {
      if (_username.isEmpty || _email.isEmpty || _password.isEmpty) {
        ScaffoldMessenger.of(context).showSnackBar(
          const SnackBar(
            content: Text("All fields are required!"),
            backgroundColor: Colors.red,
          ),
        );
      } else {
        Navigator.push(
        context,
        MaterialPageRoute(builder: (context) => const LoginPage()),
      );
    }
  }
}
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text("Sign Up Page"),
      centerTitle: true,
    ),
    body: Padding(
      padding: const EdgeInsets.all(20.0),
      child: Form(
        key: _formKey,
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            TextFormField(
              decoration: const InputDecoration(
                labelText: 'Username',
                hintText: "Enter your username",
                icon: Icon(Icons.person),
              ),
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter a username';
                } else if (value.length < 4) {
                  return 'Username must be at least 4 characters';
                }
                return null;
              },
              onChanged: (value) {
                setState(() {
                  _username = value;
                });
              },
            ),
            const SizedBox(height: 20),
            TextFormField(
              decoration: const InputDecoration(
                labelText: 'Email',
                hintText: "Enter your email ID",
                icon: Icon(Icons.email),
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter an email';
    } else if (!RegExp(r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$')
        .hasMatch(value)) {
      return 'Enter a valid email address';
    }
    return null;
  },
  onChanged: (value) {
    setState(() {
      _email = value;
    });
  },
),
const SizedBox(height: 20),
TextFormField(
  obscureText: true,
  decoration: const InputDecoration(
    labelText: 'Password',
    hintText: "Enter password",
    icon: Icon(Icons.lock),
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter a password';
    } else if (value.length < 6) {
      return 'Password must be at least 6 characters';
    }
    return null;
  },
  onChanged: (value) {
    setState(() {
      _password = value;
    });
  },
),
const SizedBox(height: 20),
ElevatedButton(
  onPressed: _signUp,
  child: const Text('Sign Up'),),
const SizedBox(height: 10),
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: [
                  const Text("Already have an account?"),
                  TextButton(
                    onPressed: () {
                      Navigator.push(
                        context,
                        MaterialPageRoute(builder: (context) => const LoginPage()),
                      );
                    },
                    child: const Text("Login"),
                  ),
                ],
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

Output:



Login Page

Username
Omkar

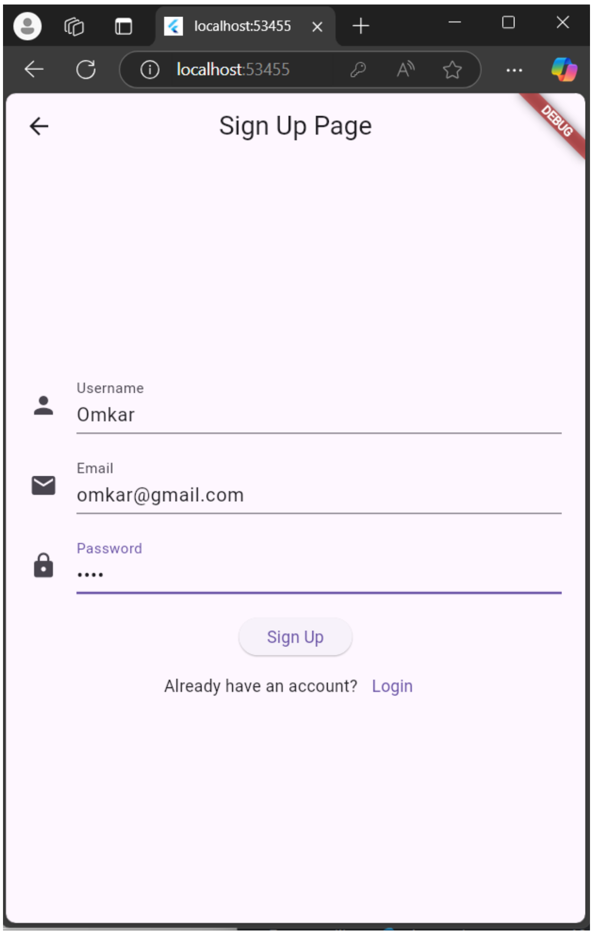
Email
omkar@gmail.com

Password
•••

Password must be at least 4 characters

Login

Don't have an account? Sign Up



Sign Up Page

Username
Omkar

Email
omkar@gmail.com

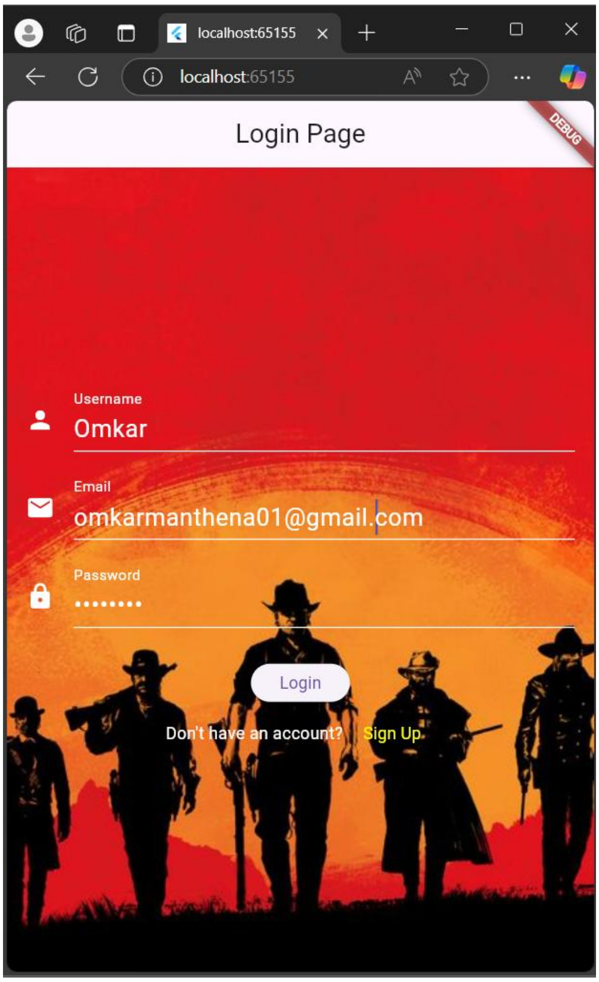
Password
••••

Sign Up

Already have an account? Login

Task:

Create your own design for login page similar to below UI



localhost:65155

Login Page

Username

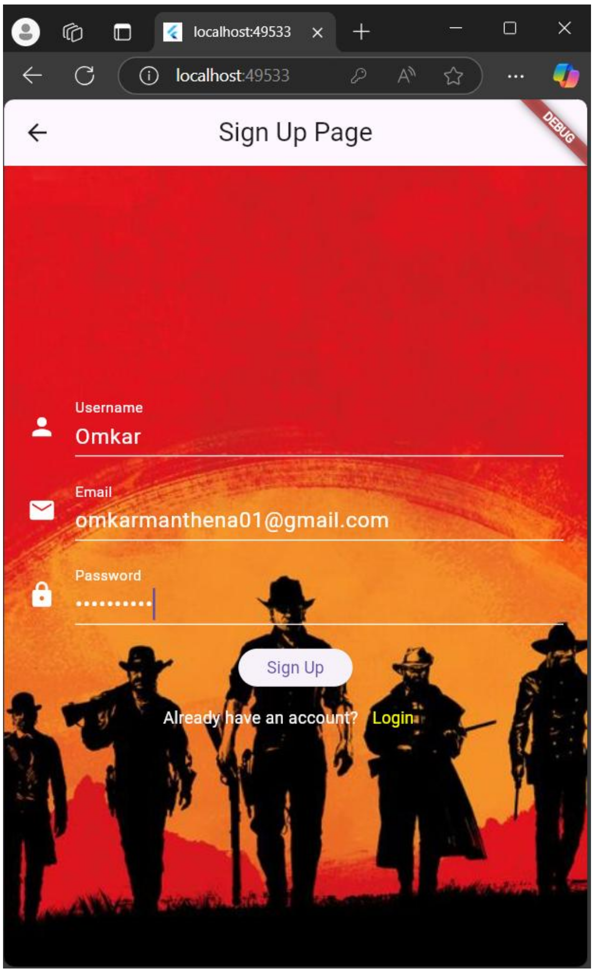
Omkar

Email

omkarmanthena01@gmail.com

Password

••••••••



localhost:49533

Sign Up Page

Username

Omkar

Email

omkarmanthena01@gmail.com

Password

•••••••••

Conclusion:

Through this lab experiment, we built an **interactive form** using Flutter's **Form widget** with proper validation and navigation. This experience helped us understand how to create user-friendly forms that ensure smooth data entry and error handling. Adding navigation made the form feel more dynamic and intuitive, making it easier for users to move between screens. Overall, this experiment showed how simple yet powerful Flutter is for creating seamless user interactions.

LO's achieved: LO2

PO's achieved: PO1, PO3, PO5, PO6, PO8, PO9, PO10, PO12.