

Name:Mayank Lulla

Dlv:D15A

Roll.no:33

Experiment 4

Aim: To create an interactive Form using form widget

Theory:

A typical form in Flutter consists of the following components:

Form Widget:

The Form widget is a container for form fields and manages the form state, validation, and submission. It provides methods to validate and save the form data.

Form Fields:

Form fields are input elements where users can enter data. Flutter provides various form field widgets like TextFormField, DropdownButtonFormField, CheckboxFormField, etc., to handle different types of user inputs.

GlobalKey:

To uniquely identify and manage the form state, we use a GlobalKey<FormState> instance. This key is assigned to the Form widget and allows us to interact with the form state from anywhere in the widget tree.

Validation:

Form validation ensures that the data entered by the user meets certain criteria or constraints. In Flutter, form validation is performed using the validator property of form field widgets. This property expects a function that returns a string message if the input data is invalid, or null if it's valid.

Saving Form Data:

After validation, the form data needs to be saved for further processing, such as submitting it to a server or updating the local state. The onSave property of form field widgets is used to define functions that save the user input into variables or objects.

Code:

```
import 'package:flutter/material.dart';
```

```
void main() {  
  runApp(LoginPage());  
}
```

```
class LoginPage extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      home: Scaffold(  
        appBar: AppBar(  
          title: Text('Login Page'),  
        ),  
        body: LoginForm(),  
      ),  
    );  
  }  
}
```

```
class LoginForm extends StatefulWidget {  
  @override  
  _LoginFormState createState() => _LoginFormState();  
}
```

```
class _LoginFormState extends State<LoginForm> {  
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>();  
  final TextEditingController _emailController =  
    TextEditingController();  
}
```

```
final TextEditingController _passwordController =  
TextEditingController();
```

```
@override
```

```
Widget build(BuildContext context) {  
  return Padding(  
    padding: EdgeInsets.all(20.0),  
    child: Form(  
      key: _formKey,  
      child: Column(  
        mainAxisAlignment: MainAxisAlignment.center,  
        children: [  
          TextFormField(  
            controller: _emailController,  
            keyboardType: TextInputType.emailAddress,  
            decoration: InputDecoration(  
              labelText: 'Email',  
              border: OutlineInputBorder(),  
            ),  
            validator: (value) {  
              if (value?.isEmpty ?? true) {  
                return 'Please enter your email';  
              }  
              return null;  
            },  
          ),  
          SizedBox(height: 20.0),  
          TextFormField(  
            controller: _passwordController,  
            obscureText: true,  
            decoration: InputDecoration(  
              labelText: 'Password',
```

```

        border: OutlineInputBorder(),
      ),
      validator: (value) {
        if (value?.isEmpty ?? true) {
          return 'Please enter your password';
        }
        return null;
      },
    ),
    SizedBox(height: 20.0),
    ElevatedButton(
      onPressed: () {
        if (_formKey.currentState?.validate() ?? false) {
          // Form is valid, perform login logic here
          String email = _emailController.text;
          String password = _passwordController.text;
          print('Email: $email, Password: $password');
        }
      },
      child: Text('Login'),
    ),
  ],
),
),
),
);
}
}

```

Output:

