

Experiment 4

Aim: To create an interactive Form using form widget

Theory:

Form validation in Flutter ensures that the data entered by the user meets certain criteria or constraints before it's submitted or processed. It's crucial for ensuring data integrity and providing a smooth user experience. Flutter provides built-in features and widgets to implement form validation efficiently.

Key Concepts:

Form Widget: The Form widget is the root of the form hierarchy in Flutter. It provides methods to manage the state of the form and validate its fields.

TextFormField Widget: The TextFormField widget is used to capture user input in a form. It comes with properties like validator and onSave to facilitate form validation and data handling.

Validation Functions: Validators are functions assigned to validator properties of form fields. They take the user input as an argument and return an error message if the input is invalid, or null if it's valid.

GlobalKey: The GlobalKey is used to uniquely identify a Form widget. It allows us to access and interact with the form's state, such as triggering validation and saving form data.

Implementation Steps:

Wrap Fields with Form Widget: Enclose the form fields within a Form widget. This establishes the form hierarchy and enables validation.

Add Validators: Assign validation functions to the validator properties of each form field. These functions determine whether the input meets the specified criteria.

Validate Form: When the form is submitted, call the validate method of the form key. This triggers validation for all form fields. If any field fails validation, the corresponding error message is displayed.

Save Form Data: If the form passes validation, call the save method of the form key to retrieve the entered data. You can then process or submit the data as needed.

Benefits of Form Validation:

Data Integrity: Validates user input to ensure that only valid data is submitted, improving the quality and accuracy of stored data.

User Experience: Provides real-time feedback to users, guiding them to correct any errors in their input before submission, leading to a more intuitive and user-friendly experience.

Error Handling: Enables effective error handling by displaying informative error messages, helping users understand and address validation issues.

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

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Sign in',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: const MyForm(),
    );
  }
}

class MyForm extends StatefulWidget {
  const MyForm({Key? key}) : super(key: key);

  @override
  // ignore: library_private_types_in_public_api
  _MyFormState createState() => _MyFormState();
}

class _MyFormState extends State<MyForm> {
  final _formKey = GlobalKey<FormState>();
}
```

```
String? _name;
String? _email;
String? _password;

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Interactive Form Demo'),
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            TextFormField(
              decoration: const InputDecoration(labelText: 'Name'),
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter your name';
                }
                return null;
              },
              onSave: (value) {
                _name = value;
              },
            ),
            TextFormField(
              decoration: const InputDecoration(labelText: 'Email'),
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter your email';
                }
                return null;
              },
              onSave: (value) {
                _email = value;
              },
            ),
          ],
        ),
      ),
    ),
  );
}
```

```
    ),
    TextFormField(
      decoration: const InputDecoration(labelText: 'Password'),
      obscureText: true,
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Please enter your password';
        }
        return null;
      },
      onSave: (value) {
        _password = value;
      },
    ),
    Padding(
      padding: const EdgeInsets.symmetric(vertical: 16.0),
      child: ElevatedButton(
        onPressed: () {
          if (_formKey.currentState!.validate()) {
            _formKey.currentState!.save();
            // Now you can use the form data (_name, _email,
            _password)

            // ignore: avoid_print
            print('Name: $_name');
            // ignore: avoid_print
            print('Email: $_email');
            // ignore: avoid_print
            print('Password: $_password');
          }
        },
        child: const Text('Submit'),
      ),
    ),
  ],
),
),
),
),
);
}
```

Name

Aakanksha Suvarna

Email

2021.aakanksha.suvarna@ves.ac.in

Password

Please enter your password

Submit