

Experiment No. 4

AIM : To create an interactive Form using form widget

Theory:

1. Form Widget:

- The Form widget is a container for managing form-related interactions. It allows for validation and saving the form data.
- It keeps track of the state of all the fields within the form (like TextFormField widgets) through a GlobalKey<FormState>.

2. TextFormField:

- This is the main widget used for collecting user input (such as text). It integrates easily with form validation and submission.
- You can apply validators to the TextFormField to ensure the input meets specific criteria (e.g., required fields, correct format).

3. GlobalKey:

- A GlobalKey<FormState> is essential for managing the form's state (e.g., validating fields, saving data). It's assigned to the Form widget and can be used to trigger actions like form validation or saving the data.

4. Validation:

- You can define validation rules on each form field. The TextFormField widget has a validator property, which allows you to write logic that will run whenever the form is validated.
- A validator checks whether the input meets the required format (such as checking for valid email format or a non-empty field).

5. Form Submission:

- When you're ready to submit the form, you can call formKey.currentState?.save() to trigger the save method for all fields or formKey.currentState?.validate() to check if all the fields pass the validation checks.

6. Saving Data:

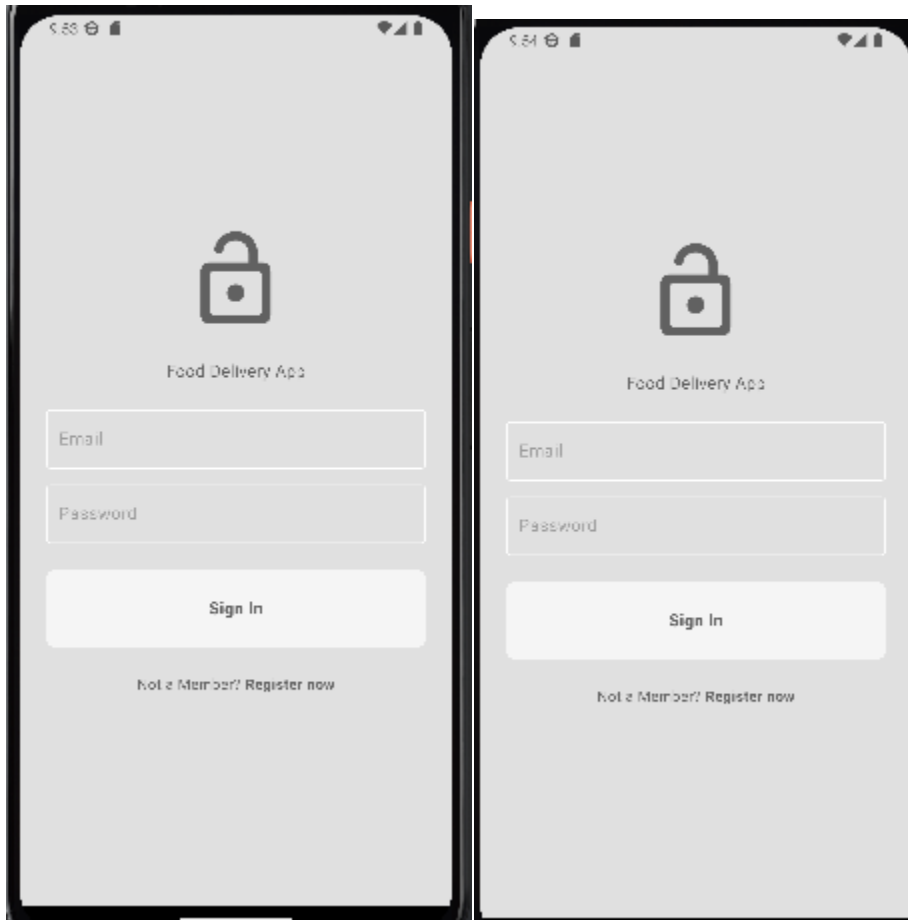
- After validation, data entered in the form fields can be saved to variables or used for further processing, such as sending it to a server.

Form Workflow:

1. **Create a form:** Use the Form widget to wrap all input fields.
2. **Add form fields:** Use widgets like TextFormField to collect data.

3. **Set up validation:** Add validation logic to the form fields.
4. **Submit the form:** Trigger validation and handle the form submission.
5. **Save data:** Capture the entered data once validation passes.

Flutter's form management tools are powerful, enabling you to handle complex forms with various input types, validations, and submissions effectively.



Code Snippets:

Login Page:

```
import 'package:flutter/material.dart';
import 'package:flutter_eats/components/my_button.dart';
import 'package:flutter_eats/components/my_textfield.dart';
import 'home_page.dart';

class LoginPage extends StatefulWidget {
```

```

final void Function()? onTap;
const LoginPage( {super.key, required this.onTap});

@override
State<LoginPage> createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  // Text editing controllers
  final TextEditingController emailController = TextEditingController();
  final TextEditingController passwordController = TextEditingController();

  void login() {
    /*
    fill out authentication here
    */

    // navigate to home page
    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => const HomePage(),
      ),
    );
  }

  @override
  void dispose() {
    emailController.dispose();
    passwordController.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Theme.of(context).colorScheme.background,
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            // Logo
            Icon(
              Icons.lock_open_rounded,
              size: 100,
              color: Theme.of(context).colorScheme.inversePrimary,
            ),
          ],
        ),
      ),
    );
  }
}

```

```
),

const SizedBox(height: 25),

// App Slogan
Text(
  "Food Delivery App",
  style: TextStyle(
    fontSize: 16,
    color: Theme.of(context).colorScheme.inversePrimary,
  ),
),

const SizedBox(height: 25),

// Email Text Field
MyTextField(
  controller: emailController,
  hintText: "Email",
  obscureText: false,
),

const SizedBox(height: 15),

// Password Text Field
MyTextField(
  controller: passwordController,
  hintText: "Password",
  obscureText: true,
),

const SizedBox(height: 25),

// Sign In Button
MyButton(
  text: "Sign In",
  onTap: login
),

const SizedBox(height: 25),

// Register Row
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: [
    Text(
      "Not a Member?",
```

```

        style: TextStyle(color: Theme.of(context).colorScheme.inversePrimary),
      ),
      const SizedBox(width: 4),
      GestureDetector(
        onTap: widget.onTap, // Fix onTap
        child: Text(
          "Register now",
          style: TextStyle(
            color: Theme.of(context).colorScheme.inversePrimary,
            fontWeight: FontWeight.bold,
          ),
        ),
      ),
    ],
  ),
],
),
),
);
}
}

```

Register Page:

```

import 'package:flutter/material.dart';
import 'package:flutter_eats/components/my_button.dart';
import 'package:flutter_eats/components/my_textfield.dart';

class RegisterPage extends StatefulWidget {
  final void Function()? onTap;
  const RegisterPage({super.key, required this.onTap});

  @override
  State<RegisterPage> createState() => _RegisterPageState();
}

class _RegisterPageState extends State<RegisterPage> {
  // Text editing controllers
  final TextEditingController emailController = TextEditingController();
  final TextEditingController passwordController = TextEditingController();
  final TextEditingController confirmPasswordController = TextEditingController();

  @override
  void dispose() {
    emailController.dispose();
    passwordController.dispose();
    super.dispose();
  }
}

```

```
}
```

```
@override
```

```
Widget build(BuildContext context) {  
  return Scaffold(  
    backgroundColor: Theme.of(context).colorScheme.background,  
    body: Center(  
      child: Column(  
        mainAxisAlignment: MainAxisAlignment.center,  
        children: [  
          // Logo  
          Icon(  
            Icons.lock_open_rounded,  
            size: 100,  
            color: Theme.of(context).colorScheme.inversePrimary,  
          ),  
  
          const SizedBox(height: 25),  
  
          // App Slogan  
          Text(  
            " Lets create an account for you",  
            style: TextStyle(  
              fontSize: 16,  
              color: Theme.of(context).colorScheme.inversePrimary,  
            ),  
          ),  
  
          const SizedBox(height: 25),  
  
          // Email Text Field  
          MyTextField(  
            controller: emailController,  
            hintText: "Email",  
            obscureText: false,  
          ),  
  
          const SizedBox(height: 15),  
  
          // Password Text Field  
          MyTextField(  
            controller: passwordController,  
            hintText: "Password",  
            obscureText: true,  
          ),  
  
          const SizedBox(height: 25),  
        ],  
      ),  
    ),  
  );  
}
```

```

// Confirm Password Text Field
MyTextField(
  controller: confirmPasswordController,
  hintText: "Confirm Password",
  obscureText: true,
),

// Sign up Button
MyButton(
  onTap: () {
    // Implement sign-in functionality
  },
  text: "Sign up",
),

const SizedBox(height: 25),

// already have an account? Login here
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: [
    Text(
      "already have an account?",
      style: TextStyle(color: Theme.of(context).colorScheme.inversePrimary),
    ),
    const SizedBox(width: 4),
    GestureDetector(
      onTap: widget.onTap, // Fix onTap
      child: Text(
        "Login now",
        style: TextStyle(
          color: Theme.of(context).colorScheme.inversePrimary,
          fontWeight: FontWeight.bold,
        ),
      ),
    ),
  ],
),
],
),
),
);
}
}

```