

EXPERIMENT 4

AIM : To create an interactive Form using a form widget.

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.

Code:

LOGIN.DART:

```
import 'package:flutter/material.dart';
import '../services/auth_service.dart';
import '../widgets/custom_text_field.dart';
import '../widgets/small_button.dart';

class LoginPage extends StatelessWidget {
  final TextEditingController _usernameController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();
  final AuthService _authService = AuthService();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Stack(
        children: [
          // Background Image
          Container(
            decoration: BoxDecoration(
              image: DecorationImage(
                image: AssetImage('assets/Connected.jpg'),
                fit: BoxFit.cover,
              ),
            ),
          ),
          // Dark overlay
          Container(
            color: Colors.black.withOpacity(0.3),
          ),
          // Login Form Positioned at Bottom
          Align(
            alignment: Alignment.bottomCenter,
            child: Container(
              padding: EdgeInsets.fromLTRB(16, 12, 16, 8),
              width: double.infinity,
              decoration: BoxDecoration(
                color: Color(0xFF1B2228),
                borderRadius: BorderRadius.only(
                  topLeft: Radius.circular(12),
                  topRight: Radius.circular(12),
                ),
              ),
              child: Column(
                mainAxisAlignment: MainAxisAlignment.min,
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  Text(
                    "Sign in to Letterboxd",
                    style: TextStyle(
                      fontSize: 20,
                      fontWeight: FontWeight.w500,
                      color: Colors.white,
                    ),
                  ),
                  SizedBox(height: 12),
                ],
              ),
            ),
          ),
        ],
      ),
    );
  }
}
```

```
CustomTextField(hintText: "Username", controller: _usernameController),
CustomTextField(hintText: "Password", isPassword: true, controller: _passwordController),
SizedBox(height: 8),
Row(
  children: [
    SmallButton(
      text: "JOIN",
      color: Color(0xFF38444E),
      onPressed: () {
        Navigator.pushNamed(context, '/signup');
      },
    ),
    SizedBox(width: 8),
    SmallButton(
      text: "RESET PASSWORD",
      color: Color(0xFF38444E),
      onPressed: () {
        // Add reset password functionality
      },
    ),
    SizedBox(width: 8),
    SmallButton(
      text: "GO",
      color: Colors.green,
      onPressed: () async {
        final user = await _authService.signIn(
          _usernameController.text,
          _passwordController.text,
        );
        if (user != null) {
          // Fetch the username from Firestore
          final username = await _authService.getUsername(user.uid);
          if (username != null) {
            Navigator.pushReplacementNamed(context, '/home');
          } else {
            ScaffoldMessenger.of(context).showSnackBar(
              SnackBar(content: Text('Failed to fetch user data.')),
            );
          }
        } else {
          ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(content: Text('Login failed. Please try again.')),
          );
        }
      },
    ),
  ],
),
SizedBox(height: 8),
Center(
  child: Text(
    "Artwork from Mitchell vs the Machines",
    style: TextStyle(
      color: Colors.grey[400],
      fontSize: 11,
    ),
  ),
),
),
),
),
),
```

```

    ],
  ),
);
}
}

```

SIGNUP.DART:

```

import 'package:flutter/material.dart';
import '../services/auth_service.dart';
import '../widgets/custom_text_field.dart';
import '../widgets/small_button.dart';
import '../widgets/toggle_switch.dart';

```

```

class SignupPage extends StatelessWidget {
  final TextEditingController _emailController = TextEditingController();
  final TextEditingController _usernameController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();
  final AuthService _authService = AuthService();

```

@override

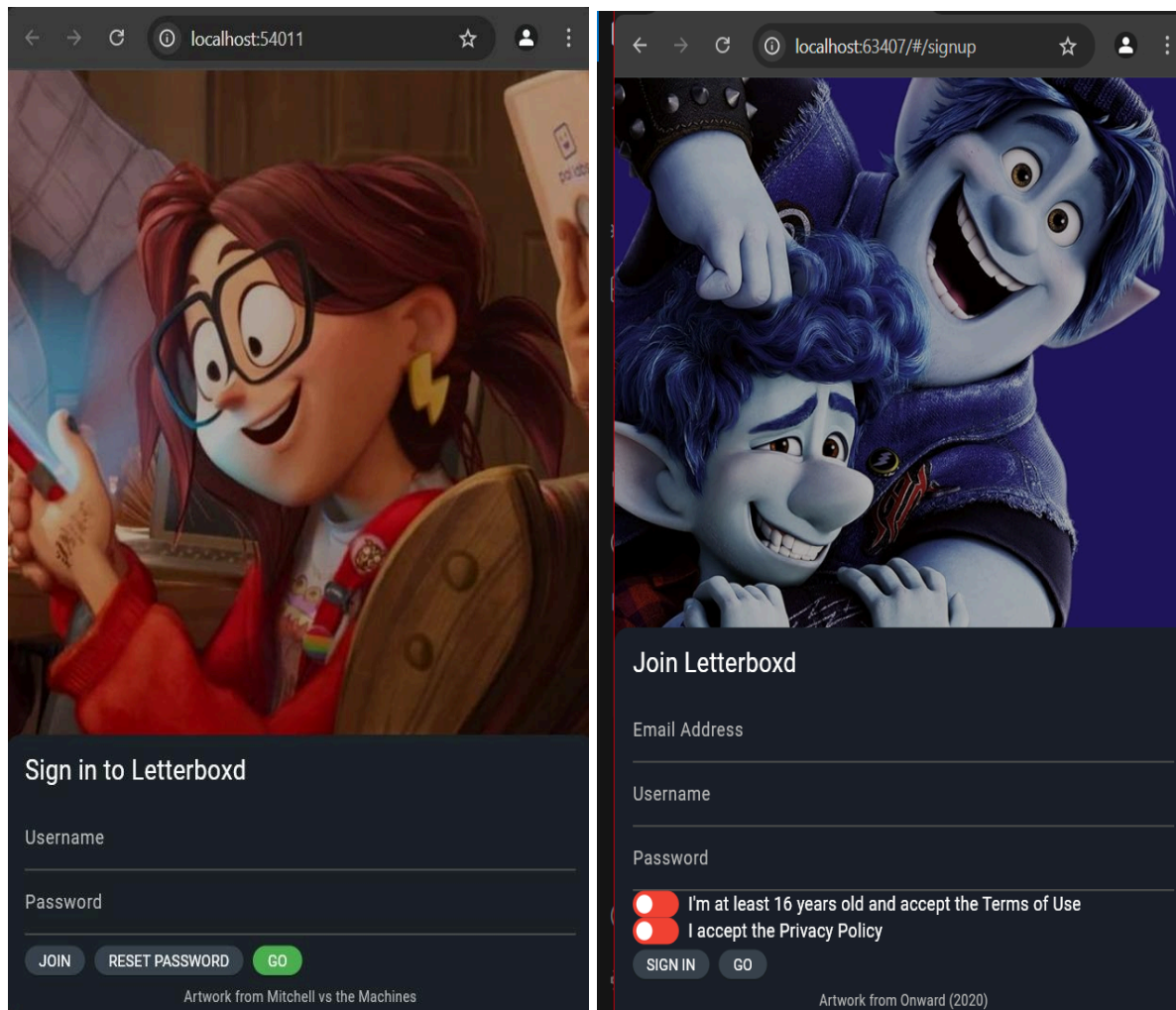
```

Widget build(BuildContext context) {
  return Scaffold(
    body: Stack(
      children: [
        // Background Image
        Container(
          decoration: BoxDecoration(
            image: DecorationImage(
              image: AssetImage('assets/onward.jpg'),
              fit: BoxFit.cover,
            ),
          ),
        ),
        // Dark overlay
        Container(
          color: Colors.black.withOpacity(0.3),
        ),
        // Signup Form Positioned at Bottom
        Align(
          alignment: Alignment.bottomCenter,
          child: Container(
            padding: EdgeInsets.fromLTRB(16, 12, 16, 8),
            decoration: BoxDecoration(
              color: Color(0xFF1B2228),
              borderRadius: BorderRadius.only(
                topLeft: Radius.circular(12),
                topRight: Radius.circular(12),
              ),
            ),
            child: Column(
              mainAxisAlignment: MainAxisAlignment.min,
              crossAxisAlignment: CrossAxisAlignment.start,
              children: [
                Text(
                  "Join Letterboxd",
                  style: TextStyle(
                    fontSize: 20,
                    fontWeight: FontWeight.w500,
                    color: Colors.white,
                  ),
                ),
                SizedBox(height: 12),

```

```
CustomTextField(hintText: "Email Address", controller: _emailController),  
CustomTextField(hintText: "Username", controller: _usernameController),  
CustomTextField(hintText: "Password", isPassword: true, controller: _passwordController),  
ToggleSwitch(text: "I'm at least 16 years old and accept the Terms of Use"),  
ToggleSwitch(  
    text: "I accept the Privacy Policy "  
),  
SizedBox(height: 4),  
Row(  
    children: [  
        SmallButton(  
            text: "SIGN IN",  
            color: Color(0xFF38444E),  
            onPressed: () {  
                Navigator.pushNamed(context, '/login');  
            },  
        ),  
        SizedBox(width: 8),  
        SmallButton(  
            text: "GO",  
            color: Color(0xFF38444E),  
            onPressed: () async {  
                final user = await _authService.signUp(  
                    _emailController.text,  
                    _usernameController.text,  
                    _passwordController.text,  
                );  
                if (user != null) {  
                    Navigator.pushReplacementNamed(context, '/home');  
                } else {  
                    ScaffoldMessenger.of(context).showSnackBar(  
                        SnackBar(content: Text('Sign-up failed. Please try again.')),  
                    );  
                }  
            },  
        ),  
    ],  
),  
SizedBox(height: 8),  
Center(  
    child: Text(  
        "Artwork from Onward (2020)",  
        style: TextStyle(  
            color: Colors.grey[400],  
            fontSize: 11,  
        ),  
    ),  
),  
),  
),  
),  
),  
),  
),  
),  
);  
}
```

Screenshots:



Conclusion:

In this experiment, we successfully created an interactive form using Flutter's Form and TextFormField widgets. The form included multiple input fields such as full name, email, password, and confirm password. We used a GlobalKey<FormState> to manage the form's state and implemented custom validators for each field to ensure data integrity and proper user input.

The experiment demonstrated how Flutter allows developers to build responsive and user-friendly forms with built-in validation, easy data handling, and clean UI design. This approach is highly efficient for applications that require user registration, login, or any kind of data input.

Overall, this experiment enhanced our understanding of form validation, form state management, and user interaction handling in Flutter.

