

## Experiment 4

Name: Khushi Jeswani

Div: D15A

Roll no: 26

Aim: To create an interactive form using the form widget Theory:

- **Form Widget:**
- The Form widget is a container used to group multiple form fields together.
- It helps manage the state of the form, including validation, submission, and resetting.
- The Form widget maintains a `FormState` object that holds the current state of the form fields.
- Form widgets facilitate form submission, validation, and error handling.
- **FormField Widget:**
- A `FormField` widget represents a single form field within a `Form`.
- Flutter provides various subclasses of the `FormField` widget for different types of input fields, such as `TextFormField`, `CheckboxFormField`, `RadioFormField`, `DropdownButtonFormField`, etc.
- Each form field widget encapsulates the logic for validating user input and managing its state.
- Form fields can be customized with properties to specify validation rules, error messages, initial values, input formatting, and more.
- Form fields automatically register themselves with the `Form` widget and handle validation and state management transparently.
- **Validation:**
- Flutter's form widgets include built-in support for validation to ensure that user input meets specific criteria.
- Form fields can be configured with validation functions or validators to check the correctness of user input.
- Validators can be synchronous or asynchronous functions that return error messages if the input is invalid.
- Flutter provides a `FormFieldState` class associated with each form field, which exposes methods to validate the field's value and retrieve validation errors if any.
- **Submission:**
- The `Form` widget provides a mechanism to submit the form data once it's been filled out by the user.
- Developers can define an `onSaved` callback for each form field to specify how the field's value should be processed when the form is submitted.

- When the form is submitted, the onSaved callbacks for all form fields are invoked, allowing developers to collect, process, and submit the form data to a backend server or perform other actions.

Signup:

```
[10:06 AM, 3/20/2024] Khushi Vesit: import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
// import 'package:slicing_snapchat/page/home_page.dart';
// import 'package:slicing_snapchat/widgets/custom_text_field_widget.art.dart';
import 'package:snapchatfinal/page/home_page.dart';
import 'package:snapchatfinal/widgets/custom_text_field_widget.art.dart';
```

```
import '../widgets/reusable_button.dart';
```

```
class RegisterPage extends StatefulWidget {
  @override
  _RegisterPageState createState() => _RegisterPageState();
}
```

```
class _RegisterPageState extends State<RegisterPage> {
```

```
  TextEditingController emailTextEditingController = TextEditingController();
  TextEditingController passwordTextEditingController = TextEditingController();
  TextEditingController confirmPasswordTextEditingController =
    TextEditingController();
  TextEditingController UsernameTextEditingController = TextEditingController();
  TextEditingController ageTextEditingController = TextEditingController();
  TextEditingController phoneNoEditingController = TextEditingController();
  // TextEditingController cityTextEditingController = TextEditingController();
  // TextEditingController countryTextEditingController = TextEditingController();
  // TextEditingController profileHeadingTextEditingController =
  //   TextEditingController();
  //
  // TextEditingController heightTextEditingController = TextEditingController();
  // TextEditingController weightTextEditingController = TextEditingController();
  // TextEditingController bodytypeTextEditingController = TextEditingController();
  // TextEditingController drinkTextEditingController = TextEditingController();
  // TextEditingController smokeTextEditingController = TextEditingController();
  // TextEditingController haveChildrenTextEditingController =
  //   TextEditingController();
  // TextEditingController noOfChildrenTextEditingController =
  //   TextEditingController();
```

```

// TextEditingController professionTextEditingController =
//   TextEditingController();
// TextEditingController incomeTextEditingController = TextEditingController();
// TextEditingController livingSituationTextEditingController =
//   TextEditingController();
// TextEditingController willingToRelocateTextEditingController =
//   TextEditingController();
// TextEditingController relationshipYouAreLookingForTextEditingController =
//   TextEditingController();
// TextEditingController nationalityTextEditingController =
//   TextEditingController();
// TextEditingController educationTextEditingController =
//   TextEditingController();
// TextEditingController languageSpokenTextEditingController =
//   TextEditingController();
// TextEditingController religionTextEditingController = TextEditingController();
// TextEditingController ethnicityTextEditingController =
//   TextEditingController();
// final FirebaseFirestore _firestore = FirebaseFirestore.instance;
final FirebaseFirestore _firestore = FirebaseFirestore.instance;
void createAccount() async {
  String email = emailTextEditingController.text.trim();
  String password = passwordTextEditingController.text.trim();
  String cpassword = confirmpasswordTextEditingController.text.trim();

  if (email == "" || password == "" || cpassword == "") {
    print('Please fill all details');
  } else if (password != cpassword) {
    print('Password do not match');
  } else {
    try {
      UserCredential userCredential = await FirebaseAuth.instance
        .createUserWithEmailAndPassword(email: email, password: password);

      if (userCredential.user != null) {
        Navigator.push(
          context,
          MaterialPageRoute(builder: (context) => HomePage()),
        );
      }
    } catch (e) {
      print(e);
    }
  }
}

```

[10:06 AM, 3/20/2024] Khushi Vesit: signup

[10:06 AM, 3/20/2024] Khushi Vesit: import 'package:cloud\_firestore/cloud\_firestore.dart';

import 'package:firebase\_auth/firebase\_auth.dart';

```

import 'package:flutter/material.dart';
// import 'package:slicing_snapchat/page/home_page.dart';
// import 'package:slicing_snapchat/widgets/custom_text_field_widget.art.dart';
import 'package:snapchatfinal/page/home_page.dart';
import 'package:snapchatfinal/widgets/custom_text_field_widget.art.dart';
// import 'package:font_awesome_flutter/font_awesome_flutter.dart';

import '../widgets/reusable_button.dart';

class LoginPage extends StatefulWidget {
  @override
  _LoginPageState createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  bool _obscureText = true;
  TextEditingController emailTextEditingController = TextEditingController();
  TextEditingController passwordTextEditingController = TextEditingController();
  // IconData eyelcon = FontAwesomeIcons.eyeSlash;
  final FirebaseFirestore _firestore = FirebaseFirestore.instance;

  // void _togglePasswordVisibility() {
  //   setState(() {
  //     _obscureText = !_obscureText;
  //     // eyelcon = _obscureText ? FontAwesomeIcons.eyeSlash : FontAwesomeIcons.eye;
  //   });
  // }

  void login() async {
    String email = emailTextEditingController.text.trim();
    String password = passwordTextEditingController.text.trim();

    if (email == "" || password == "") {
      print('please fill all details');
    } else {
      try {
        UserCredential userCredential = await FirebaseAuth.instance
          .signInWithEmailAndPassword(email: email, password: password);

        if (userCredential != null) {
          Navigator.push(
            context,
            MaterialPageRoute(builder: (context) => HomePage()),
          );
        }
      }
    }
  }
}

```

```

        _firestore.collection('users').doc(userCredential.user!.uid).set({
          'uid': userCredential.user!.uid,
          'email': email,
        }, SetOptions(merge: true));
      }
    } on FirebaseAuthException catch (ex) {
      print(ex.code.toString());
    }
  }
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      leading: GestureDetector(
        onTap: () {
          Navigator.pop(context);
        },
        child: Icon(
          Icons.arrow_back_ios,
          color: Colors.grey,
        ),
      ),
      elevation: 0,
      backgroundColor: Colors.white,
    ),
    body: Center(
      child: SingleChildScrollView(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
          children: [
            Text(
              "Log in",
              style: TextStyle(fontSize: 40, color: Colors.black),
            ),
            SizedBox(height: 40),
            CustomTextFieldWidget(
              // label: "USERNAME OR EMAIL",
              // isPasswordField: false,
              // autoFocus: true,
              editingController: emailTextEditingController,
              labelText: 'Email',
            ),
          ],
        ),
      ),
    ),
  );
}

```

```

        iconData: Icons.email_outlined,
        isObscure: false,
    ),
    SizedBox(height: 20),
    Column(
      children: [
        Container(
          alignment: Alignment.centerLeft,
          margin: EdgeInsets.symmetric(horizontal: 50),
          child: CustomTextFieldWidget(

            // label: "USERNAME OR EMAIL",
            // isPasswordField: false,
            // autoFocus: true,
            editingController: passwordTextEditingController,
            labelText: 'password',
            iconData: Icons.email_outlined,
            isObscure: false,
          ),
          // child: Text(
          //   "PASSWORD",
          //   style: TextStyle(
          //     fontSize: 18,
          //     fontWeight: FontWeight.bold,
          //     color: Color(0xFF51B5E5),
          //   ),
          // ),
        ),
        Padding(
          padding: const EdgeInsets.symmetric(horizontal: 50),
          // child: TextField(
          //   obscureText: _obscureText,
          //   autofocus: false,
          //   cursorHeight: 33,
          //   cursorWidth: 2,
          //   decoration: InputDecoration(
          //     suffixIcon: GestureDetector(
          //       onTap: _togglePasswordVisibility,
          //       child: Text('Show Password')
          //     // child: Falcon(
          //     //   eyeIcon,
          //     //   color: Colors.grey,
          //     // ),
          //   ),
        ),
      ],
    ),
  ),
)

```

```

        // floatingLabelBehavior: FloatingLabelBehavior.never,
        // contentPadding: EdgeInsets.all(6),
        // ),
        // cursorColor: Color(0xFF69B77D),
        // ),
    ),
],
),
 SizedBox(height: 60),
 GestureDetector(
  onTap: () {
    //forgot your password
  },
  child: Text(
    "Forgot your password?",
    style: TextStyle(
      fontSize: 17,
      fontWeight: FontWeight.bold,
      color: Color(0xFF51B5E5),
    ),
  ),
),
),
 SizedBox(height: 90),
 Padding(
  padding: const EdgeInsets.symmetric(horizontal: 80),
  child: GestureDetector(
    onTap: () {
      Navigator.pushNamed(context, '/home_page');
    },
    child: Container(
      margin: EdgeInsets.only(top: 20),
      child: Text(
        "Log in",
        style: TextStyle(
          fontSize: 25,
          color: Colors.white,
          fontWeight: FontWeight.bold,
        ),
      ),
    ),
    alignment: Alignment.center,
    height: 55,
    width: double.infinity,
    decoration: BoxDecoration(
      color: Color(0xFFADB6BD),

```

```

        borderRadius: BorderRadius.circular(80),
      ),
    ),
  ),
],
),
),
),
);
}
}

```

Login:

```

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
// import 'package:slicing_snapchat/page/home_page.dart';
// import 'package:slicing_snapchat/widgets/custom_text_field_widget.art.dart';
import 'package:snapchatfinal/page/home_page.dart';
import 'package:snapchatfinal/widgets/custom_text_field_widget.art.dart';
// import 'package:font_awesome_flutter/font_awesome_flutter.dart';

import '../widgets/reusable_button.dart';

class LoginPage extends StatefulWidget {
  @override
  _LoginPageState createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  bool _obscureText = true;
  TextEditingController emailTextEditingController = TextEditingController();
  TextEditingController passwordTextEditingController = TextEditingController();
  // IconData eyelcon = FontAwesomeIcons.eyeSlash;
  final FirebaseFirestore _firestore = FirebaseFirestore.instance;

  // void _togglePasswordVisibility() {
  //   setState(() {
  //     _obscureText = !_obscureText;
  //     // eyelcon = _obscureText ? FontAwesomeIcons.eyeSlash : FontAwesomeIcons.eye;
  //   });
  // }

```



```
// }
```

```
void login() async {  
  String email = emailTextEditingController.text.trim();  
  String password = passwordTextEditingController.text.trim();  
  
  if (email == "" || password == "") {  
    print('please fill all details');  
  } else {  
    try {  
      UserCredential userCredential = await FirebaseAuth.instance  
        .signInWithEmailAndPassword(email: email, password: password);  
  
      if (userCredential != null) {  
        Navigator.push(  
          context,  
          MaterialPageRoute(builder: (context) => HomePage()),  
        );  
        _firestore.collection('users').doc(userCredential.user!.uid).set({  
          'uid': userCredential.user!.uid,  
          'email': email,  
        }, SetOptions(merge: true));  
      }  
    } on FirebaseAuthException catch (ex) {  
      print(ex.code.toString());  
    }  
  }  
}
```

```
@override  
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      leading: GestureDetector(  
        onTap: () {  
          Navigator.pop(context);  
        },  
        child: Icon(  
          Icons.arrow_back_ios,  
          color: Colors.grey,  
        ),  
      ),  
      elevation: 0,  
      backgroundColor: Colors.white,
```

```

),
body: Center(
  child: SingleChildScrollView(
    child: Column(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: [
        Text(
          "Log in",
          style: TextStyle(fontSize: 40, color: Colors.black),
        ),
        SizedBox(height: 40),
        CustomTextFieldWidget(

          // label: "USERNAME OR EMAIL",
          // isPasswordField: false,
          // autoFocus: true,
          editingController: emailTextEditingController,
          labelText: 'Email',
          iconData: Icons.email_outlined,
          isObscure: false,
        ),
        SizedBox(height: 20),
        Column(
          children: [
            Container(
              alignment: Alignment.centerLeft,
              margin: EdgeInsets.symmetric(horizontal: 50),
              child: CustomTextFieldWidget(

                // label: "USERNAME OR EMAIL",
                // isPasswordField: false,
                // autoFocus: true,
                editingController: passwordTextEditingController,
                labelText: 'password',
                iconData: Icons.email_outlined,
                isObscure: false,
              ),
              // child: Text(
              //   "PASSWORD",
              //   style: TextStyle(
              //     fontSize: 18,
              //     fontWeight: FontWeight.bold,
              //     color: Color(0xFF51B5E5),
              //   ),
            ),

```

```

        // ),
      ),
      Padding(
        padding: const EdgeInsets.symmetric(horizontal: 50),
        // child: TextField(
        //   obscureText: _obscureText,
        //   autofocus: false,
        //   cursorHeight: 33,
        //   cursorWidth: 2,
        //   decoration: InputDecoration(
        //     suffixIcon: GestureDetector(
        //       onTap: _togglePasswordVisibility,
        //       child :Text('Show Password')
        //       // child: Falcon(
        //       //   eyelcon,
        //       //   color: Colors.grey,
        //       // ),
        //     ),
        //     floatingLabelBehavior: FloatingLabelBehavior.never,
        //     contentPadding: EdgeInsets.all(6),
        //   ),
        //   cursorColor: Color(0xFF69B77D),
        // ),
      ),
    ],
  ),
  SizedBox(height: 60),
  GestureDetector(
    onTap: () {
      //forgot your password
    },
    child: Text(
      "Forgot your password?",
      style: TextStyle(
        fontSize: 17,
        fontWeight: FontWeight.bold,
        color: Color(0xFF51B5E5),
      ),
    ),
  ),
  ),
  ),
  SizedBox(height: 90),
  Padding(
    padding: const EdgeInsets.symmetric(horizontal: 80),
    child: GestureDetector(

```

```
onTap: () {  
  Navigator.pushNamed(context, '/home_page');  
},  
child: Container(  
  margin: EdgeInsets.only(top: 20),  
  child: Text(  
    "Log in",  
    style: TextStyle(  
      fontSize: 25,  
      color: Colors.white,  
      fontWeight: FontWeight.bold,  
    ),  
  ),  
  alignment: Alignment.center,  
  height: 55,  
  width: double.infinity,  
  decoration: BoxDecoration(  
    color: Color(0xFFADB6BD),  
    borderRadius: BorderRadius.circular(80),  
  ),  
),  
),  
),  
),  
],  
),  
),  
),  
);  
}  
}
```

App UI: Login page

10:04 83%

<

## Log in

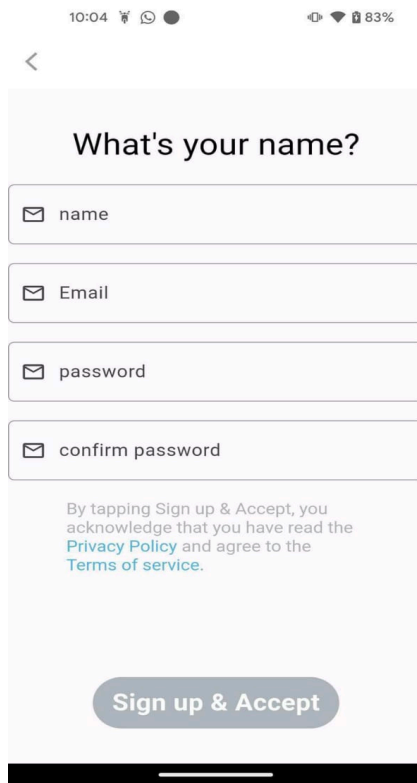
✉ Email

✉ password

[Forgot your password?](#)

Log in

## Signup page



The screenshot shows a mobile app interface for a signup page. At the top, the status bar displays the time 10:04, signal strength, Wi-Fi, and 83% battery. Below the status bar is a back arrow icon. The main heading is "What's your name?". There are four input fields, each with an envelope icon on the left: "name", "Email", "password", and "confirm password". Below the input fields is a paragraph of text: "By tapping Sign up & Accept, you acknowledge that you have read the [Privacy Policy](#) and agree to the [Terms of service](#)." At the bottom is a large, rounded button labeled "Sign up & Accept". The bottom of the screen shows a black bar with a white horizontal line, indicating a mobile device home indicator.

Widgets used: Form Widget, Form Widget Fields

Conclusion: Therefore understood the use of form widget in Flutter. Implemented signup and login page using form widget in my Flutter application.