

Experiment : 04

AIM : To create an interactive Form using form widget

Code:

```
import 'dart:io';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:firebase_storage/firebase_storage.dart';
import 'package:gmail_clone/models/user.dart' as CustomUser;
import 'package:flutter/material.dart';
import 'package:gmail_clone/variables.dart';
import 'package:image_picker/image_picker.dart';

class SignUp extends StatefulWidget {
  const SignUp({super.key});
  @override
  State<SignUp> createState() => _SignUpState();
}

class _SignUpState extends State<SignUp> {
  final GlobalKey<ScaffoldMessengerState> scaffoldMessengerKey =
    GlobalKey<ScaffoldMessengerState>();
  File? imagepath;
  var usernamecontroller = TextEditingController();
  var passwordcontroller = TextEditingController();
  pickimage(ImageSource imageSource) async {
    final image = await ImagePicker().pickImage(
      source: imageSource,
      maxWidth: 670,
      maxHeight: 800,
    );
    if (image != null) {
      setState(() {
        imagepath = File(image.path);
      });
      Navigator.pop(context);
    }
  }

  pickimagedialog() {
    return showDialog(
      context: context,
      builder: (context) {
        return SimpleDialog(
          children: <Widget>[
```

```

SimpleDialogOption(
  onPressed: () => pickimage(ImageSource.gallery),
  child:
    Text("From gallery", style: mystyle(20, Colors.lightBlue)),
),
SimpleDialogOption(
  onPressed: () => pickimage(ImageSource.camera),
  child:
    Text("From camera", style: mystyle(20, Colors.lightBlue)),
),
SimpleDialogOption(
  onPressed: () => Navigator.pop(context),
  child: Text("Cancel", style: mystyle(20, Colors.lightBlue)),
),
],
);
});
}

uploadimage() async {
  //Store image
  UploadTask storage =
    profilepics.child(usernamecontroller.text).putFile(imagepath!);
  //complete storage
  TaskSnapshot storageTaskSnapshot = await storage;
  //download pic
  String downloadedpic = await storageTaskSnapshot.ref.getDownloadURL();
  return downloadedpic;
}

registeruser() async {
  try {
    String downloadpic = imagepath == null
      ? 'https://fdlc.org/157-1578186_user-profile-default-image-png-clipart-png/'
      : await uploadimage();
    FirebaseAuth.instance
      .createUserWithEmailAndPassword(
        email: usernamecontroller.text + '@gmail.com',
        password: passwordcontroller.text)
      .then((signeduser) {
        CustomUser.User().storeuser(usernamecontroller.text + '@gmail.com',
          usernamecontroller.text, passwordcontroller.text, downloadpic);
      });
    Navigator.pop(context);
  } catch (e) {
    print(e);
    Snackbar snackBar = Snackbar(content: Text(e.toString()));
    scaffoldMessengerKey.currentState!.showSnackBar(snackBar);
  }
}

```

```

}
@override
Widget build(BuildContext context) {
  return Scaffold(
    key: scaffoldMessengerKey,
    body: SingleChildScrollView(
      child: Center(
        child: Container(
          width: MediaQuery.of(context).size.width,
          height: MediaQuery.of(context).size.height,
          child: Card(
            elevation: 10,
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              children: <Widget>[
                Text("Google",
                  style: mystyle(
                    25,
                    Color.fromARGB(255, 4, 20, 243),
                  )),
                SizedBox(height: 5.0),
                Text("Create Account", style: mystyle(18, Colors.black)),
                SizedBox(height: 10.0),
                Text("Continue to gmail", style: mystyle(18, Colors.black)),
                SizedBox(height: 20),
                Container(
                  width: MediaQuery.of(context).size.width,
                  margin: EdgeInsets.only(left: 20, right: 20),
                  child: TextField(
                    controller: usernamecontroller,
                    decoration: InputDecoration(
                      filled: true,
                      hintText: "Username",
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(15.0))),
                  ),
                ),
                SizedBox(height: 15.0),
                Container(
                  width: MediaQuery.of(context).size.width,
                  margin: EdgeInsets.only(left: 20, right: 20),
                  child: TextField(
                    controller: passwordcontroller,
                    decoration: InputDecoration(
                      filled: true,
                      hintText: "Password",
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(15.0))),
                  ),
                ),
              ],
            ),
          ),
        ),
      ),
    ),
  );
}

```

```

    ),
    SizedBox(height: 15.0),
    Container(
      width: MediaQuery.of(context).size.width,
      margin: EdgeInsets.only(left: 20, right: 20),
      child: TextField(
        decoration: InputDecoration(
          filled: true,
          hintText: "Repeat Password",
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(15.0))),
      ),
    ),
    SizedBox(height: 10.0),
    Row(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: <Widget>[
        InkWell(
          onTap: () => pickimagedialog(),
          child: Container(
            width: 64,
            height: 64,
            decoration: BoxDecoration(
              color: Colors.lightBlue,
              borderRadius: BorderRadius.circular(20.0)),
            child: Center(
              child: imagepath == null
                ? Icon(Icons.add)
                : Image(image: FileImage(imagepath!)),
            ),
          ),
        ),
        InkWell(
          onTap: () => pickimagedialog(),
          child:
            Text("Choose profile image", style: mystyle(20))),
      ],
    ),
    SizedBox(height: 15.0),
    ElevatedButton(
      onPressed: () => registeruser(),
      style: ElevatedButton.styleFrom(
        backgroundColor: Colors.lightBlue[200],
      ),
      child: Text("Register", style: mystyle(20)),
    )
  ],
),
),
),

```

```
    ));  
    },  
    );  
}  
}
```

Output:


