Experiment 4

Aim: Guidelines To create an interactive Form using form widget

Code:

login page

import 'package:cloud\_firestore/cloud\_firestore.dart';

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

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

// import 'package:';

// import 'sign\_up\_page.dart';

// import 'package:flutter\_firebase/features/user\_auth/presentation/pages/sign\_up\_page.dart';

// import 'package:flutter\_firebase/features/user\_auth/presentation/widgets/form\_container\_widget.dart';

// import 'package';

// import 'package:flutter\_firebase/global/common/toast.dart';

import 'package:font\_awesome\_flutter/font\_awesome\_flutter.dart';

import 'package:login\_auth\_new/features/user\_auth/firebase\_auth\_implementation/firebase\_auth\_services.dart';

import 'package:login\_auth\_new/features/user\_auth/presentation/pages/login\_page.dart';

import 'package:login\_auth\_new/features/user\_auth/presentation/widgets/form\_container\_widget.dart';

import 'package:login\_auth\_new/global/common/toast.dart';

// import 'package:google\_sign\_in/google\_sign\_in.dart';

import 'package:login\_auth\_new/features/user\_auth/presentation/pages/sign\_up\_page.dart';

import 'package:login\_auth\_new/global/common/toast.dart';

import '../../firebase\_auth\_implementation/firebase\_auth\_services.dart';

class LoginPage extends StatefulWidget {

const LoginPage({super.key});

@override

State<LoginPage> createState() => \_LoginPageState();

}

class \_LoginPageState extends State<LoginPage> {

bool \_isSigning = false;

final FirebaseAuthService \_auth = FirebaseAuthService();

final FirebaseAuth \_firebaseAuth = FirebaseAuth.instance;

TextEditingController \_emailController = TextEditingController();

TextEditingController \_passwordController = TextEditingController();

@override

void dispose() {

\_emailController.dispose();

\_passwordController.dispose();

super.dispose();

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

automaticallyImplyLeading: false,

title: Text("Login"),

),

body: Center(

child: Padding(

padding: const EdgeInsets.symmetric(horizontal: 15),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

"Login",

style: TextStyle(fontSize: 27, fontWeight: FontWeight.bold),

),

SizedBox(

height: 30,

),

FormContainerWidget(

controller: \_emailController,

hintText: "Email",

isPasswordField: false,

),

SizedBox(

height: 10,

),

FormContainerWidget(

controller: \_passwordController,

hintText: "Password",

isPasswordField: true,

),

SizedBox(

height: 30,

),

GestureDetector(

onTap: () {

\_signIn();

},

child: Container(

width: double.infinity,

height: 45,

decoration: BoxDecoration(

color: Colors.blue,

borderRadius: BorderRadius.circular(10),

),

child: Center(

child: \_isSigning ? CircularProgressIndicator(

color: Colors.white,) : Text(

"Login",

style: TextStyle(

color: Colors.white,

fontWeight: FontWeight.bold,

),

),

),

),

),

SizedBox(height: 10,),

// GestureDetector(

// onTap: () {

// \_signInWithGoogle();

// },

// child: Container(

// width: double.infinity,

// height: 45,

// decoration: BoxDecoration(

// color: Colors.red,

// borderRadius: BorderRadius.circular(10),

// ),

// child: Center(

// child: Row(

// mainAxisAlignment: MainAxisAlignment.center,

// children: [

// Icon(FontAwesomeIcons.google, color: Colors.white,),

// SizedBox(width: 5,),

// Text(

// "Sign in with Google",

// style: TextStyle(

// color: Colors.white,

// fontWeight: FontWeight.bold,

// ),

// ),

// ],

// ),

// ),

// ),

// ),

SizedBox(

height: 20,

),

Row(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text("Don't have an account?"),

SizedBox(

width: 5,

),

GestureDetector(

onTap: () {

Navigator.pushAndRemoveUntil(

context,

MaterialPageRoute(builder: (context) => SignUpPage()),

(route) => false,

);

},

child: Text(

"Sign Up",

style: TextStyle(

color: Colors.blue,

fontWeight: FontWeight.bold,

),

),

),

],

),

],

),

),

),

);

}

void \_signIn() async {

setState(() {

\_isSigning = true;

});

String email = \_emailController.text;

String password = \_passwordController.text;

User? user = await \_auth.signInWithEmailAndPassword(email, password);

setState(() {

\_isSigning = false;

});

if (user != null) {

showToast(message: "User is successfully signed in");

Navigator.pushNamed(context, "/home");

} else {

showToast(message: "some error occured");

}

}

// \_signInWithGoogle()async{

// final GoogleSignIn \_googleSignIn = GoogleSignIn();

// try {

// final GoogleSignInAccount? googleSignInAccount = await \_googleSignIn.signIn();

// if(googleSignInAccount != null ){

// final GoogleSignInAuthentication googleSignInAuthentication = await

// googleSignInAccount.authentication;

// final AuthCredential credential = GoogleAuthProvider.credential(

// idToken: googleSignInAuthentication.idToken,

// accessToken: googleSignInAuthentication.accessToken,

// );

// await \_firebaseAuth.signInWithCredential(credential);

// Navigator.pushNamed(context, "/home");

// }

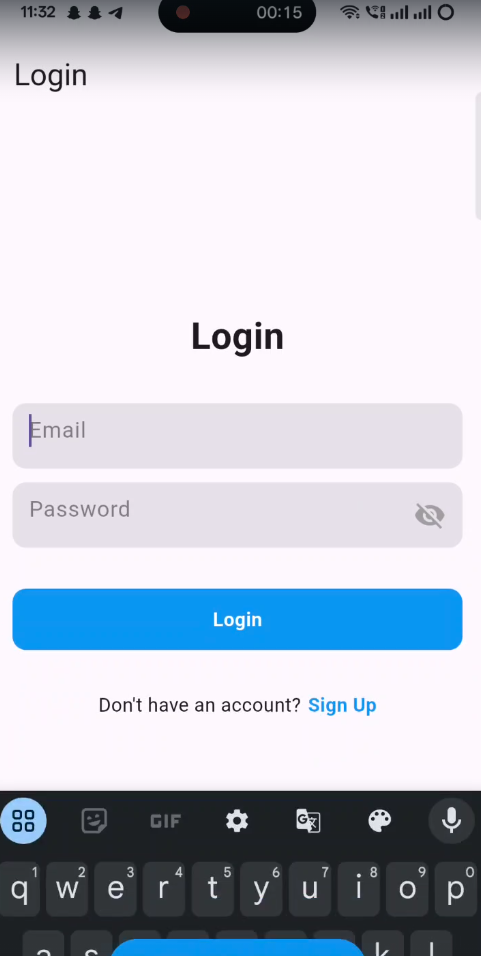
// }catch(e) {

// showToast(message: "some error occured $e");

// }

// }

}



signup page

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

import 'package:flutter/material.dart';

// import 'package:flutter\_firebase/features/user\_auth/firebase\_auth\_implementation/firebase\_auth\_services.dart';

// import 'package:flutter\_firebase';

// import 'package:flutter\_firebase/features/user\_auth/presentation/pages/login\_page.dart';

// import 'package:flutter\_firebase/features/user\_auth/presentation/widgets/form\_container\_widget.dart';

// import 'package:flutter\_firebase/global/common/toast.dart';

import 'package:login\_auth\_new/features/user\_auth/firebase\_auth\_implementation/firebase\_auth\_services.dart';

import 'package:login\_auth\_new/features/user\_auth/presentation/pages/login\_page.dart';

import 'package:login\_auth\_new/features/user\_auth/presentation/widgets/form\_container\_widget.dart';

import 'package:login\_auth\_new/global/common/toast.dart';

import 'package:login\_auth\_new/features/user\_auth/firebase\_auth\_implementation/firebase\_auth\_services.dart';

import 'package:login\_auth\_new/features/user\_auth/presentation/pages/login\_page.dart';

import 'package:login\_auth\_new/global/common/toast.dart';

class SignUpPage extends StatefulWidget {

const SignUpPage({super.key});

@override

State<SignUpPage> createState() => \_SignUpPageState();

}

class \_SignUpPageState extends State<SignUpPage> {

final FirebaseAuthService \_auth = FirebaseAuthService();

TextEditingController \_usernameController = TextEditingController();

TextEditingController \_emailController = TextEditingController();

TextEditingController \_passwordController = TextEditingController();

bool isSigningUp = false;

@override

void dispose() {

\_usernameController.dispose();

\_emailController.dispose();

\_passwordController.dispose();

super.dispose();

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

automaticallyImplyLeading: false,

title: Text("SignUp"),

),

body: Center(

child: Padding(

padding: const EdgeInsets.symmetric(horizontal: 15),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

"Sign Up",

style: TextStyle(fontSize: 27, fontWeight: FontWeight.bold),

),

SizedBox(

height: 30,

),

FormContainerWidget(

controller: \_usernameController,

hintText: "Username",

isPasswordField: false,

),

SizedBox(

height: 10,

),

FormContainerWidget(

controller: \_emailController,

hintText: "Email",

isPasswordField: false,

),

SizedBox(

height: 10,

),

FormContainerWidget(

controller: \_passwordController,

hintText: "Password",

isPasswordField: true,

),

SizedBox(

height: 30,

),

GestureDetector(

onTap: (){

\_signUp();

},

child: Container(

width: double.infinity,

height: 45,

decoration: BoxDecoration(

color: Colors.blue,

borderRadius: BorderRadius.circular(10),

),

child: Center(

child: isSigningUp ? CircularProgressIndicator(color: Colors.white,):Text(

"Sign Up",

style: TextStyle(

color: Colors.white, fontWeight: FontWeight.bold),

)),

),

),

SizedBox(

height: 20,

),

Row(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text("Already have an account?"),

SizedBox(

width: 5,

),

GestureDetector(

onTap: () {

Navigator.pushAndRemoveUntil(

context,

MaterialPageRoute(

builder: (context) => LoginPage()),

(route) => false);

},

child: Text(

"Login",

style: TextStyle(

color: Colors.blue, fontWeight: FontWeight.bold),

))

],

)

],

),

),

),

);

}

void \_signUp() async {

setState(() {

isSigningUp = true;

});

String username = \_usernameController.text;

String email = \_emailController.text;

String password = \_passwordController.text;

User? user = await \_auth.signUpWithEmailAndPassword(email, password);

setState(() {

isSigningUp = false;

});

if (user != null) {

showToast(message: "User is successfully created");

Navigator.pushNamed(context, "/home");

} else {

showToast(message: "Some error happend");

}

}

}

