import 'package:flutter/material.dart';

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

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

void main() async {

WidgetsFlutterBinding.ensureInitialized();

await Firebase.initializeApp();

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'Email & Password Auth',

home: const AuthScreen(),

);

}

}

class AuthScreen extends StatefulWidget {

const AuthScreen({super.key});

@override

State<AuthScreen> createState() => \_AuthScreenState();

}

class \_AuthScreenState extends State<AuthScreen> {

final \_formKey = GlobalKey<FormState>();

final TextEditingController \_emailController = TextEditingController();

final TextEditingController \_passwordController = TextEditingController();

bool \_isLoading = false;

String? \_errorMessage;

Future<void> \_signUp() async {

setState(() {

\_isLoading = true;

\_errorMessage = null;

});

try {

await FirebaseAuth.instance.createUserWithEmailAndPassword(

email: \_emailController.text.trim(),

password: \_passwordController.text.trim());

// Successful signup - you can navigate or show success message

ScaffoldMessenger.of(context).showSnackBar(

const SnackBar(content: Text('Sign-up successful!')));

} on FirebaseAuthException catch (e) {

setState(() {

\_errorMessage = e.message;

});

} finally {

setState(() {

\_isLoading = false;

});

}

}

Future<void> \_signIn() async {

setState(() {

\_isLoading = true;

\_errorMessage = null;

});

try {

await FirebaseAuth.instance.signInWithEmailAndPassword(

email: \_emailController.text.trim(),

password: \_passwordController.text.trim());

ScaffoldMessenger.of(context).showSnackBar(

const SnackBar(content: Text('Sign-in successful!')));

// Navigate to another screen if needed

} on FirebaseAuthException catch (e) {

setState(() {

\_errorMessage = e.message;

});

} finally {

setState(() {

\_isLoading = false;

});

}

}

@override

void dispose() {

\_emailController.dispose();

\_passwordController.dispose();

super.dispose();

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(title: const Text('Email & Password Auth')),

body: Padding(

padding: const EdgeInsets.all(16),

child: Form(

key: \_formKey,

child: Column(

children: [

if (\_errorMessage != null)

Text(

\_errorMessage!,

style: const TextStyle(color: Colors.red),

),

TextFormField(

controller: \_emailController,

decoration: const InputDecoration(

labelText: 'Email',

border: OutlineInputBorder(),

),

keyboardType: TextInputType.emailAddress,

validator: (value) {

if (value == null ||

value.isEmpty ||

!value.contains('@')) {

return 'Enter a valid email';

}

return null;

},

),

const SizedBox(height: 16),

TextFormField(

controller: \_passwordController,

decoration: const InputDecoration(

labelText: 'Password',

border: OutlineInputBorder(),

),

obscureText: true,

validator: (value) {

if (value == null || value.length < 6) {

return 'Password should be at least 6 characters';

}

return null;

},

),

const SizedBox(height: 20),

if (\_isLoading)

const CircularProgressIndicator()

else

Row(

mainAxisAlignment: MainAxisAlignment.spaceEvenly,

children: [

ElevatedButton(

onPressed: () {

if (\_formKey.currentState!.validate()) {

\_signIn();

}

},

child: const Text('Sign In'),

),

ElevatedButton(

onPressed: () {

if (\_formKey.currentState!.validate()) {

\_signUp();

}

},

child: const Text('Sign Up'),

),

],

)

],

),

),

),

);

}

}