dependencies:

flutter:

sdk: flutter

firebase\_core: ^3.0.0

cloud\_firestore: ^5.0.0

import 'package:flutter/material.dart';

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

import 'image\_card\_page.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: 'Firestore Image Cards',

theme: ThemeData(primarySwatch: Colors.blue),

home: const ImageCardPage(),

);

}

}

import 'package:flutter/material.dart';

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

class ImageCardPage extends StatelessWidget {

const ImageCardPage({super.key});

@override

Widget build(BuildContext context) {

final Stream<QuerySnapshot> imageStream =

FirebaseFirestore.instance.collection('images').snapshots();

return Scaffold(

appBar: AppBar(

title: const Text('Firestore Image Cards'),

),

body: StreamBuilder<QuerySnapshot>(

stream: imageStream,

builder: (context, snapshot) {

if (snapshot.connectionState == ConnectionState.waiting) {

return const Center(child: CircularProgressIndicator());

}

if (!snapshot.hasData || snapshot.data!.docs.isEmpty) {

return const Center(child: Text('No images found.'));

}

final docs = snapshot.data!.docs;

return ListView.builder(

padding: const EdgeInsets.all(12.0),

itemCount: docs.length,

itemBuilder: (context, index) {

final data = docs[index].data() as Map<String, dynamic>;

final imageUrl = data['image'] ?? '';

final text = data['text'] ?? 'No description';

// Add slide animation for each card

return TweenAnimationBuilder(

tween: Tween<Offset>(

begin: const Offset(-1, 0), // start off-screen to the left

end: Offset.zero, // slide into position

),

duration: const Duration(milliseconds: 600),

curve: Curves.easeOut,

builder: (context, Offset offset, child) {

return Transform.translate(

offset: offset \* 100, // smooth left-to-right motion

child: child,

);

},

child: Card(

elevation: 5,

margin: const EdgeInsets.symmetric(vertical: 10),

shape: RoundedRectangleBorder(

borderRadius: BorderRadius.circular(12),

),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

ClipRRect(

borderRadius: const BorderRadius.vertical(

top: Radius.circular(12)),

child: Image.network(

imageUrl,

height: 200,

width: double.infinity,

fit: BoxFit.cover,

errorBuilder: (context, error, stackTrace) =>

const Icon(Icons.broken\_image, size: 100),

),

),

Padding(

padding: const EdgeInsets.all(12.0),

child: Text(

text,

style: const TextStyle(

fontSize: 16,

fontWeight: FontWeight.w600,

),

),

),

],

),

),

);

},

);

},

),

);

}

}