

Tabla de Contenido

Contenido

[Evaluación Diagnostica 3](#_Toc191971612)

[Semana 1 – Introducción a Flutter 4](#_Toc191971613)

[Paquetes de Instalación 4](#_Toc191971614)

[Primer APP 7](#_Toc191971615)

# Evaluación Diagnostica

1. ¿Cuál de los siguientes lenguajes es más común para desarrollar aplicaciones en Android?  
   a) Swift  
   b) Kotlin ✅  
   c) JavaScript  
   d) C#
2. ¿Cuál es el lenguaje de programación principal para desarrollar aplicaciones en iOS?  
   a) Java  
   b) Dart  
   c) Swift ✅  
   d) Python
3. ¿Qué herramienta se usa para desarrollar aplicaciones móviles en Android?  
   a) Visual Studio Code  
   b) Android Studio ✅  
   c) Xcode  
   d) Eclipse
4. ¿Cuál de los siguientes frameworks permite el desarrollo de aplicaciones móviles híbridas?  
   a) Flutter ✅  
   b) SwiftUI  
   c) Android SDK  
   d) Kotlin
5. ¿Cómo se llama el archivo donde se define la interfaz de usuario en Android con XML?  
   a) activity\_main.xml ✅  
   b) index.html  
   c) style.css  
   d) ui.xml
6. Flutter usa Java como su lenguaje principal.  
   Falso (Usa Dart) ✅
7. React Native permite desarrollar aplicaciones tanto para Android como para iOS con JavaScript.  
   Verdadero ✅
8. El ciclo de vida de una actividad en Android incluye los estados: onCreate, onStart, onResume, onPause, onStop y onDestroy.  
   Verdadero ✅
9. Las aplicaciones móviles siempre necesitan conexión a internet para funcionar.  
   Falso (Pueden funcionar sin conexión si se programan adecuadamente) ✅
10. Xcode es el entorno de desarrollo oficial para aplicaciones de Android.  
    Falso (Es para iOS) ✅

# Semana 1 – Introducción a Flutter

## Paquetes de Instalación

Iniciamos con la lista de software que usaremos en esta materia:

1. Android Studio (Ultima Versión)
   1. Configuración Inicial: Plug Flutter y Dart

Captura de pantalla de computadora

Descripción generada automáticamente

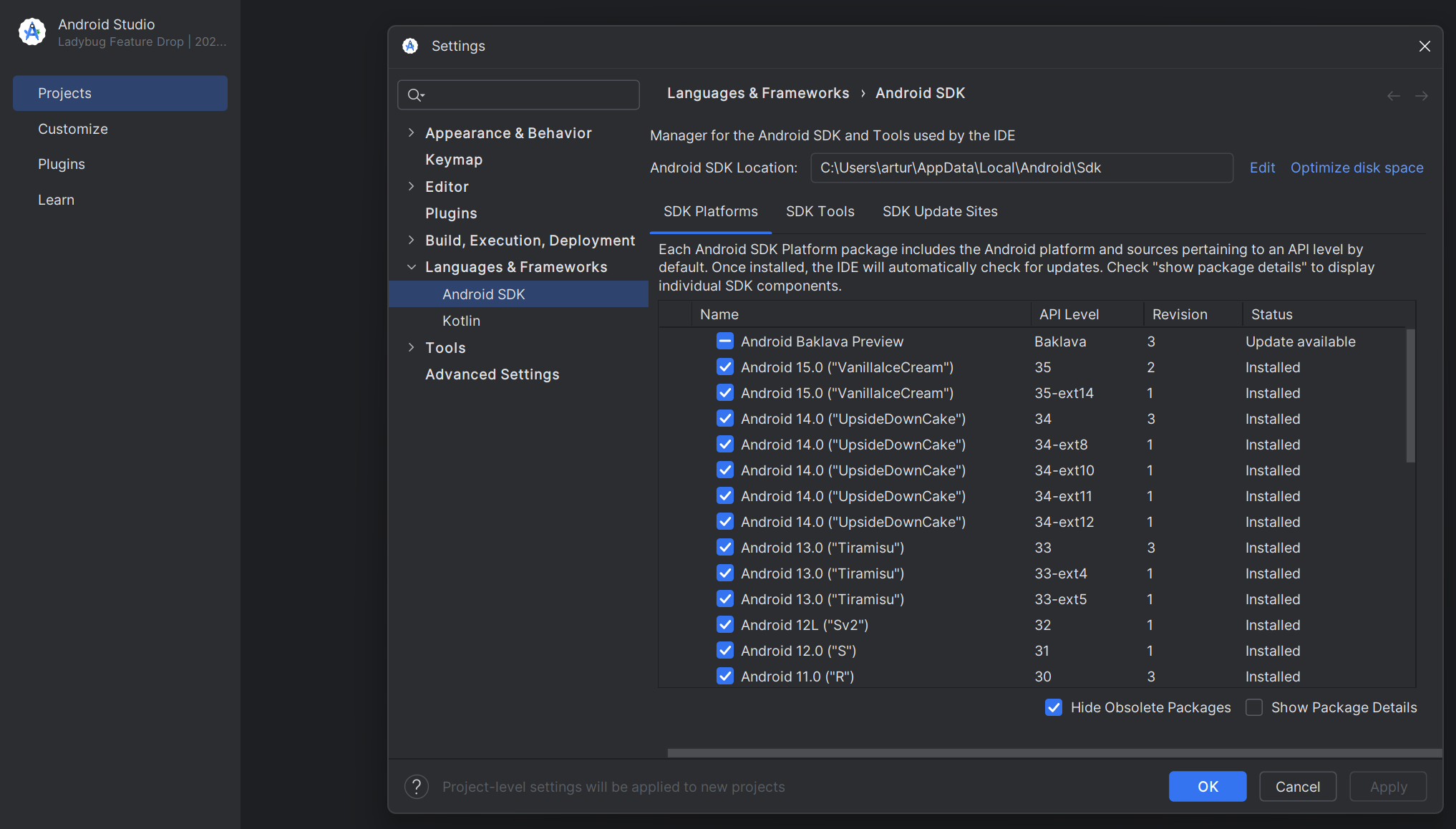
Captura de pantalla de computadora

Descripción generada automáticamente

* 1. SDK Manager instalar

Captura de pantalla de un celular

Descripción generada automáticamente

Versiones por lo menos de la 11 a la actual

Todas las SDK Tool

Captura de pantalla de computadora

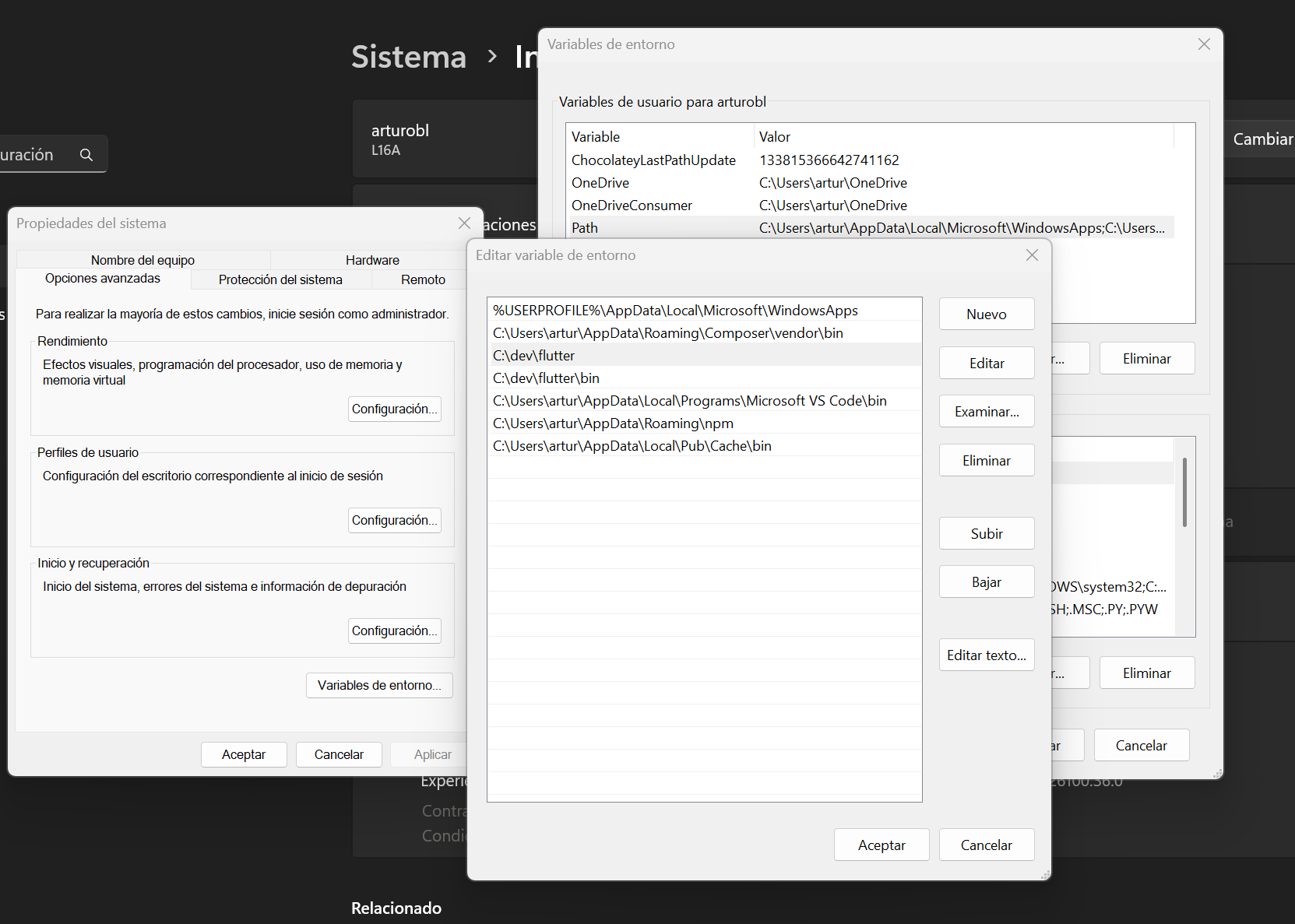
Descripción generada automáticamente

* 1. Instalar un Virtual Device Manager

Captura de pantalla de un celular

Descripción generada automáticamente

1. Instalar Visual Studio Code con las Ext y Copilot
   1. Awesome Flutter Snippets
   2. Code Runner
   3. Dart
   4. Flutter
   5. Flutter Tree
2. Bajar el SDK de Flutter
   1. [https://docs.flutter.dev/get-started/install?\_gl=1\*4b79dd\*\_gcl\_aw\*R0NMLjE3Mzc5ODgyODIuQ2p3S0NBaUEtdHk4QmhBX0Vpd0FreW9hMzJVMzhERFUwcUNjbFItcGdGcmlGOFJ0VGx5cTBYUUdlc0VKSkRKSnZSY3dRRE5DbWJfQ1d4b0NxajBRQXZEX0J3RQ..\*\_gcl\_dc\*R0NMLjE3Mzc5ODgyODIuQ2p3S0NBaUEtdHk4QmhBX0Vpd0FreW9hMzJVMzhERFUwcUNjbFItcGdGcmlGOFJ0VGx5cTBYUUdlc0VKSkRKSnZSY3dRRE5DbWJfQ1d4b0NxajBRQXZEX0J3RQ..\*\_up\*MQ..\*\_gs\*MQ..\*\_ga\*MTI5MjUxOTE5NS4xNzM3MDc1NDA4\*\_ga\_04YGWK0175\*MTczNzk4ODI4MS4yLjEuMTczNzk4ODQ2Mi4wLjAuMA..&gclid=CjwKCAiA-ty8BhA\_EiwAkyoa32U38DDU0qCclR-pgFriF8RtTlyq0XQGesEJJDJJvRcwQDNCmb\_CWxoCqj0QAvD\_BwE&gclsrc=aw.ds](https://docs.flutter.dev/get-started/install?_gl=1*4b79dd*_gcl_aw*R0NMLjE3Mzc5ODgyODIuQ2p3S0NBaUEtdHk4QmhBX0Vpd0FreW9hMzJVMzhERFUwcUNjbFItcGdGcmlGOFJ0VGx5cTBYUUdlc0VKSkRKSnZSY3dRRE5DbWJfQ1d4b0NxajBRQXZEX0J3RQ..*_gcl_dc*R0NMLjE3Mzc5ODgyODIuQ2p3S0NBaUEtdHk4QmhBX0Vpd0FreW9hMzJVMzhERFUwcUNjbFItcGdGcmlGOFJ0VGx5cTBYUUdlc0VKSkRKSnZSY3dRRE5DbWJfQ1d4b0NxajBRQXZEX0J3RQ..*_up*MQ..*_gs*MQ..*_ga*MTI5MjUxOTE5NS4xNzM3MDc1NDA4*_ga_04YGWK0175*MTczNzk4ODI4MS4yLjEuMTczNzk4ODQ2Mi4wLjAuMA..&gclid=CjwKCAiA-ty8BhA_EiwAkyoa32U38DDU0qCclR-pgFriF8RtTlyq0XQGesEJJDJJvRcwQDNCmb_CWxoCqj0QAvD_BwE&gclsrc=aw.ds)
   2. Seleccionamos Windows
   3. Seleccionamos Android
   4. Descargamos: <https://storage.googleapis.com/flutter_infra_release/releases/stable/windows/flutter_windows_3.27.3-stable.zip>
   5. Creamos una carpeta en C que se llame dev dentro ponemos el contenido del zip flutter
   6. Creamos dos variable de entorno



1. Entrar a la carpeta del SDK flutter y ejecutar flutter console y el comando flutter doctor y continuar con las sugerencias.
2. Crear nuestra primera app en vscode ejecutamos en command paleete Flutter New Project

## Primer APP

Diseño simple de un layout

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

import 'package:flutter/material.dart';

void main() {

runApp(const Home());

}

class Home extends StatefulWidget {

const Home({super.key});

@override

State<Home> createState() => \_HomeState();

}

class \_HomeState extends State<Home> {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: Scaffold(

appBar: AppBar(

title: const Text('Flutter layout demo'),

centerTitle: true,

),

body:

//Nodo Padre principal

Column(

//mainAxisAlignment: MainAxisAlignment.center,

crossAxisAlignment: CrossAxisAlignment.center,

children: [

//Hijo 1 imagen

Image.asset('assets/suiza.jpg'),

SizedBox(

height: 30,

),

//Hio 2 Fila

Row(

mainAxisAlignment: MainAxisAlignment.spaceEvenly,

children: [

//Hijo 1 de la fila

Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

//Hijo 1 de la Columna

Text("Oeschinen Lake Campground",

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

Text(

"Kandersteg, Suiza",

style: TextStyle(color: Colors.black54),

),

],

),

//Hijo 2 de la fila

Row(

children: [

Icon(

Icons.star,

color: Colors.red,

),

Text("41"),

],

),

],

),

SizedBox(height: 30),

//Hijo 3 de la columna

Row(

mainAxisAlignment: MainAxisAlignment.spaceEvenly,

children: [

Column(

children: [

Icon(

Icons.king\_bed,

color: Colors.purple,

),

Text(

"Hotel",

style: TextStyle(color: Colors.purple),

),

],

),

Column(

children: [

Icon(

Icons.restaurant,

color: Colors.purple,

),

Text(

"Comida",

style: TextStyle(color: Colors.purple),

),

],

),

Column(

children: [

Icon(

Icons.directions\_car,

color: Colors.purple,

),

Text(

"Transporte",

style: TextStyle(color: Colors.purple),

),

],

)

],

),

SizedBox(height: 30),

//Hijo 4 de la columna

Padding(

padding: const EdgeInsets.only(left: 30.0, right: 30.0),

child: Text(

"Descripcion de Suiza Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam nec purus ac libero ultricies tincidunt. Nullam nec purus ac libero ultricies tincidunt. Nullam nec purus ac libero",

textAlign: TextAlign.justify,

),

),

Container(

margin: EdgeInsets.all(30),

child: Text(

"Descripcion de Suiza Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam nec purus ac libero ultricies tincidunt. Nullam nec purus ac libero ultricies tincidunt. Nullam nec purus ac libero",

textAlign: TextAlign.justify,

),

),

],

)),

);

}

}

## SEMANA 4

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

//import 'package:firebase/pages/container\_page.dart';

import 'package:firebase/pages/home\_notes.dart';

//import 'package:firebase/pages/home\_page.dart';

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

import 'package:flutter/material.dart';

void main() async {

WidgetsFlutterBinding.ensureInitialized();

await Firebase.initializeApp(options: DefaultFirebaseOptions.currentPlatform);

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

title: 'Flutter Demo',

theme: ThemeData.dark(),

home: HomeNotes(),

);

}

}

### FIREBASE\_OPTIONS

// File generated by FlutterFire CLI.

// ignore\_for\_file: type=lint

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

import 'package:flutter/foundation.dart'

show defaultTargetPlatform, kIsWeb, TargetPlatform;

/// Default [FirebaseOptions] for use with your Firebase apps.

///

/// Example:

/// ```dart

/// import 'firebase\_options.dart';

/// // ...

/// await Firebase.initializeApp(

/// options: DefaultFirebaseOptions.currentPlatform,

/// );

/// ```

class DefaultFirebaseOptions {

static FirebaseOptions get currentPlatform {

if (kIsWeb) {

return web;

}

switch (defaultTargetPlatform) {

case TargetPlatform.android:

return android;

case TargetPlatform.iOS:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for ios - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

case TargetPlatform.macOS:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for macos - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

case TargetPlatform.windows:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for windows - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

case TargetPlatform.linux:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for linux - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

default:

throw UnsupportedError(

'DefaultFirebaseOptions are not supported for this platform.',

);

}

}

static const FirebaseOptions web = FirebaseOptions(

apiKey: 'AIzaSyBX7gPgknyj9XH5Z4QgvIXCSsZpISR0qPE',

appId: '1:97142501820:web:e36a2696dcd0ab23ca7bb9',

messagingSenderId: '97142501820',

projectId: 'myfiredata-6f51e',

authDomain: 'myfiredata-6f51e.firebaseapp.com',

storageBucket: 'myfiredata-6f51e.firebasestorage.app',

);

static const FirebaseOptions android = FirebaseOptions(

apiKey: 'AIzaSyCI4\_cWBpX4Sj8BEqNBHwuJpvF2geQEIWQ',

appId: '1:97142501820:android:6c68d3f1a7dbd1acca7bb9',

messagingSenderId: '97142501820',

projectId: 'myfiredata-6f51e',

storageBucket: 'myfiredata-6f51e.firebasestorage.app',

);

}

## Semana 5 FireBase

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un videojuego

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Imagen que contiene Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un video juego

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

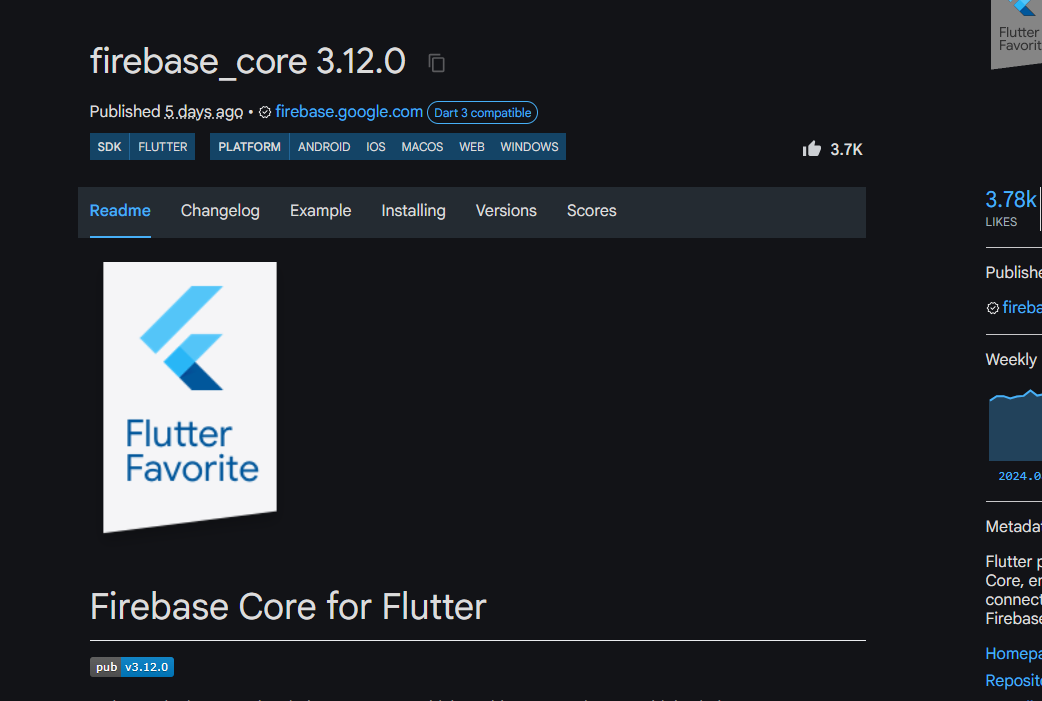
El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.



Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Imagen que contiene Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

## SEMANA 6

### Main.dart

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

//import 'package:firebase/pages/container\_page.dart';

import 'package:firebase/pages/home\_notes.dart';

//import 'package:firebase/pages/home\_page.dart';

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

import 'package:flutter/material.dart';

void main() async {

WidgetsFlutterBinding.ensureInitialized();

await Firebase.initializeApp(options: DefaultFirebaseOptions.currentPlatform);

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

title: 'Flutter Demo',

theme: ThemeData.dark(),

home: HomeNotes(),

);

}

}

### Firebase\_options

// File generated by FlutterFire CLI.

// ignore\_for\_file: type=lint

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

import 'package:flutter/foundation.dart'

show defaultTargetPlatform, kIsWeb, TargetPlatform;

/// Default [FirebaseOptions] for use with your Firebase apps.

///

/// Example:

/// ```dart

/// import 'firebase\_options.dart';

/// // ...

/// await Firebase.initializeApp(

/// options: DefaultFirebaseOptions.currentPlatform,

/// );

/// ```

class DefaultFirebaseOptions {

static FirebaseOptions get currentPlatform {

if (kIsWeb) {

return web;

}

switch (defaultTargetPlatform) {

case TargetPlatform.android:

return android;

case TargetPlatform.iOS:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for ios - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

case TargetPlatform.macOS:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for macos - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

case TargetPlatform.windows:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for windows - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

case TargetPlatform.linux:

throw UnsupportedError(

'DefaultFirebaseOptions have not been configured for linux - '

'you can reconfigure this by running the FlutterFire CLI again.',

);

default:

throw UnsupportedError(

'DefaultFirebaseOptions are not supported for this platform.',

);

}

}

static const FirebaseOptions web = FirebaseOptions(

apiKey: 'AIzaSyBX7gPgknyj9XH5Z4QgvIXCSsZpISR0qPE',

appId: '1:97142501820:web:e36a2696dcd0ab23ca7bb9',

messagingSenderId: '97142501820',

projectId: 'myfiredata-6f51e',

authDomain: 'myfiredata-6f51e.firebaseapp.com',

storageBucket: 'myfiredata-6f51e.firebasestorage.app',

);

static const FirebaseOptions android = FirebaseOptions(

apiKey: 'AIzaSyCI4\_cWBpX4Sj8BEqNBHwuJpvF2geQEIWQ',

appId: '1:97142501820:android:6c68d3f1a7dbd1acca7bb9',

messagingSenderId: '97142501820',

projectId: 'myfiredata-6f51e',

storageBucket: 'myfiredata-6f51e.firebasestorage.app',

);

}