Portada

Desarrollo de Aplicaciones II

Tabla de Contenido

Contenido

[Evaluación Diagnostica 3](#_Toc191885383)

[Semana 1 – Introducción a Flutter 4](#_Toc191885384)

[Paquetes de Instalación 4](#_Toc191885385)

[Primer APP 7](#_Toc191885386)

# Evaluación Diagnostica

# Semana 1 - Características del desarrollo de aplicaciones híbridas

# Semana 2 - Frameworks para el desarrollo hibrido (Dart, Sencha, Kendo, Vue)

# Semana 3 - Introducción a FLUTTER

# Semana 4 - Import y Export

# Evaluación Unidad 1 - Introducción al Desarrollo de aplicaciones móviles híbridas

**1.** ¿Qué es una aplicación híbrida?  
**Respuesta:**  
Es una aplicación desarrollada con tecnologías web (HTML, CSS, JavaScript, etc.) que se ejecuta dentro de un contenedor nativo, lo que le permite funcionar en múltiples plataformas (iOS, Android) desde una sola base de código.

**2.** Menciona **dos características** principales del desarrollo de aplicaciones híbridas.  
**Respuesta:**

* Uso de una sola base de código para múltiples plataformas.
* Acceso a funcionalidades del dispositivo mediante plugins o APIs nativas.

**3.** ¿Cuál es la función de los frameworks en el desarrollo híbrido?  
**Respuesta:**  
Proveen herramientas, bibliotecas y componentes que facilitan la creación de aplicaciones móviles híbridas, mejorando la productividad y reduciendo el tiempo de desarrollo.

**4.** Relaciona cada framework con su principal característica:  
a) Dart  
b) Sencha  
c) Kendo  
d) Vue

A. Orientado a interfaces visuales empresariales  
B. Lenguaje base de Flutter  
C. Framework progresivo para interfaces web  
D. Biblioteca de componentes UI basada en jQuery

**5.** ¿Qué es Flutter y quién lo desarrolló?  
**Respuesta:**  
Flutter es un framework de desarrollo de interfaces móviles, web y escritorio con una sola base de código, creado por Google.

**6.** Menciona **dos ventajas** del ecosistema de Flutter.  
**Respuesta:**

* Hot Reload para ver cambios al instante.
* Uso de widgets personalizables para construir interfaces nativas.

**7.** ¿Cuál es el lenguaje de programación que se utiliza en Flutter?  
**Respuesta:**  
Dart

**8.** ¿Qué clase se utiliza para navegar entre páginas (pantallas) en Flutter?  
**Respuesta:**  
Navigator

**9.** ¿Cómo se recibe información (input) del usuario en Flutter? Menciona un widget.  
**Respuesta:**  
Con el widget TextField, que permite recibir texto del usuario.

**10.** ¿Cuál es el propósito de usar import y export en Flutter?  
**Respuesta:**  
import se usa para incluir bibliotecas o archivos necesarios en un archivo Dart. export se usa para hacer accesibles ciertos archivos o funciones a otros módulos del proyecto.

# Semana 5 - Desarrollo de interfaces

# Semana 6 – Widgets Basicos Flutter

# Semana 7 - Servicios Y Plugins Native

# Evaluación Unidad 2 - Desarrollo de interfaces

**1.** ¿Qué widget de Flutter permite seleccionar una o varias opciones marcando una casilla?  
**Respuesta:**  
Checkbox

**2.** ¿Cuál es la diferencia principal entre los widgets ListView y GridView?  
**Respuesta:**  
ListView muestra elementos en una sola columna o fila (lista), mientras que GridView organiza los elementos en una cuadrícula de filas y columnas.

**3.** ¿Qué widget se utiliza para mostrar un botón flotante sobre la interfaz, comúnmente para ejecutar una acción principal?  
**Respuesta:**  
FloatingActionButton

**4.** ¿Qué widgets permiten al usuario seleccionar una fecha y una hora en Flutter?  
**Respuesta:**  
showDatePicker() y showTimePicker()

**5.** Menciona un componente de navegación en Flutter y su función.  
**Respuesta:**  
BottomNavigationBar: permite cambiar entre diferentes secciones o vistas principales de una app mediante una barra de navegación en la parte inferior.

**6.** ¿Para qué sirve el comando flutter build?  
**Respuesta:**  
Compila la aplicación en una versión lista para producción (APK, iOS, web, etc.).

**7.** ¿Qué comando se utiliza para iniciar una aplicación Flutter en un emulador o dispositivo?  
**Respuesta:**  
flutter run o flutter start

**8.** ¿Qué hace el comando flutter add?  
**Respuesta:**  
Agrega un paquete o plugin al proyecto (por ejemplo, desde pub.dev).

**9.** ¿Qué se entiende por "Service Integration" en el desarrollo híbrido?  
**Respuesta:**  
La capacidad de integrar servicios externos como bases de datos, autenticación, APIs REST, entre otros, dentro de la aplicación híbrida.

**10.** ¿Qué son los "Premier Plugins" en Flutter o desarrollo híbrido?  
**Respuesta:**  
Son plugins oficiales y altamente recomendados por la comunidad o desarrolladores del framework, que ofrecen integraciones nativas con alta calidad y soporte activo (como camera, firebase\_core, google\_maps\_flutter).

# Semana 8 - Integración de Base de datos, Introducción a Firebase

# Semana 9 – Autentificación

# Semana 10 - Conexión a base de datos, Operaciones CRUD

# Evaluación Unidad 3 - Integración de Base de datos

**1.** ¿Qué es Firebase y quién lo desarrolló?  
**Respuesta:**  
Firebase es una plataforma de desarrollo de aplicaciones móviles y web creada por Google que proporciona servicios backend como autenticación, base de datos en tiempo real, almacenamiento y más.

**2.** Menciona **dos ventajas** de usar Firebase en una aplicación híbrida.  
**Respuesta:**

* Fácil integración con aplicaciones móviles.
* Servicios escalables como autenticación, bases de datos y notificaciones push.

**3.** ¿Qué tipos de autenticación ofrece Firebase? Menciona al menos tres.  
**Respuesta:**  
Correo electrónico y contraseña, autenticación con Google, autenticación con Facebook, autenticación anónima, entre otros.

**4.** ¿Qué función tiene FirebaseAuth en Flutter?  
**Respuesta:**  
Gestiona el inicio de sesión, registro y cierre de sesión de usuarios en la aplicación utilizando los métodos de autenticación de Firebase.

**5.** ¿Qué es Cloud Firestore?  
**Respuesta:**  
Es una base de datos en la nube NoSQL de Firebase que permite almacenar, sincronizar y consultar datos de forma eficiente y en tiempo real.

**6.** ¿Qué significa que Firestore sea una base de datos **NoSQL**?  
**Respuesta:**  
Que no usa tablas ni relaciones tradicionales como en una base de datos SQL, sino que almacena los datos en forma de documentos y colecciones jerárquicas.

**7.** ¿Cómo se llama la colección de documentos en Firestore?  
**Respuesta:**  
Colección (Collection)

**8.** ¿Qué operación realiza el método .add() en Firestore?  
**Respuesta:**  
Crea un nuevo documento dentro de una colección (operación **Create**).

**9.** Menciona las siglas CRUD y lo que significa cada una.  
**Respuesta:**

* **C**: Create (Crear)
* **R**: Read (Leer)
* **U**: Update (Actualizar)
* **D**: Delete (Eliminar)

**10.** ¿Qué pasos básicos se deben seguir para conectar Flutter con Firebase?  
**Respuesta:**

1. Registrar el proyecto en Firebase.
2. Agregar el archivo de configuración (google-services.json o GoogleService-Info.plist).
3. Agregar dependencias en pubspec.yaml.
4. Inicializar Firebase en el main.dart usando Firebase.initializeApp().

# Semana 11 - API Development

# Semana 12 - Desarrollo mediante API

# Semana 13 - Bibliotecas y marcos

# Semana 14 – REST y SOAP

# Evaluación Unidad 4 - API Development

**1.** ¿Qué es una API en el contexto del desarrollo de software?  
**Respuesta:**  
Una API (Interfaz de Programación de Aplicaciones) es un conjunto de definiciones y protocolos que permiten la comunicación entre diferentes aplicaciones o servicios.

**2.** Menciona **dos ventajas** de desarrollar aplicaciones mediante el uso de APIs.  
**Respuesta:**

* Permite la reutilización de servicios existentes sin reescribir código.
* Facilita la integración con servicios externos como pagos, mapas, autenticación, etc.

**3.** ¿Qué diferencia hay entre una **biblioteca** y un **framework** en desarrollo de software?  
**Respuesta:**  
Una **biblioteca** ofrece funciones que el desarrollador llama cuando las necesita.  
Un **framework** impone una estructura y el flujo del programa, y el desarrollador escribe el código que se ajusta a él.

**4.** Menciona un ejemplo de biblioteca y un ejemplo de framework comúnmente usados.  
**Respuesta:**

* Biblioteca: jQuery
* Framework: Angular

**5.** ¿Qué significa REST y cuál es su propósito principal?  
**Respuesta:**  
REST (Representational State Transfer) es un estilo de arquitectura que usa HTTP para realizar operaciones CRUD sobre recursos representados como URLs.

**6.** Menciona **dos métodos HTTP** utilizados en una API REST y su función.  
**Respuesta:**

* GET: Obtener información.
* POST: Enviar o crear nuevos datos.

**7.** ¿Qué estructura de datos se usa comúnmente para intercambiar información en APIs REST?  
**Respuesta:**  
JSON (JavaScript Object Notation)

**8.** ¿Qué es SOAP y en qué se diferencia de REST?  
**Respuesta:**  
SOAP (Simple Object Access Protocol) es un protocolo estándar basado en XML para intercambiar mensajes estructurados. A diferencia de REST, SOAP es más estricto, usa XML exclusivamente y tiene un formato más complejo.

**9.** Menciona una **ventaja** de usar SOAP sobre REST.  
**Respuesta:**  
SOAP ofrece mayor seguridad y transacciones más complejas, lo que lo hace ideal para servicios bancarios o empresariales críticos.

**10.** ¿En qué casos se recomienda usar REST sobre SOAP?  
**Respuesta:**  
Cuando se busca simplicidad, rapidez y eficiencia en aplicaciones móviles o web modernas que no requieren estructuras de datos complejas ni protocolos estrictos.

# Semana 15 - Configuración de una PWA, PWA Share API

# Semana 16 - Push Storage, Inicio de sesión y Token

# Examen final Unidad Publicar aplicación como PWA

1. ¿Cuál de los siguientes archivos es obligatorio para que una aplicación web sea reconocida como una PWA?

A) .htaccess

B) service-worker.js

C) manifest.json

D) robots.txt

2. Menciona dos ventajas de convertir una aplicación web en una PWA.

3. ¿Cuál es un requisito esencial del servidor donde se alojará una PWA para que funcione correctamente?

A) Soporte para PHP 8

B) Acceso SSH

C) Soporte para HTTPS

D) Redireccionamiento 301 activo

4. ¿Qué debe configurarse en el archivo .htaccess (o en la configuración del servidor) para garantizar que las rutas de una PWA funcionen correctamente en aplicaciones tipo SPA?

5. ¿Cuál es la función principal de la Web Share API en una PWA?

A) Enviar datos al servidor usando WebSockets

B) Compartir contenido desde la app a otras apps del dispositivo

C) Encriptar datos de usuario antes de almacenarlos

D) Reproducir multimedia de forma offline

6. Describe brevemente el flujo general para implementar notificaciones Push en una PWA sobre Android.

7. (Opción múltiple) ¿Cuál de los siguientes elementos NO es necesario para enviar una notificación Push en la Web?

A) Un Service Worker registrado

B) Una suscripción Push válida

C) Un archivo robots.txt

D) Un servidor que envíe el mensaje

8. ¿Dónde y cómo se almacena generalmente la información de suscripción para notificaciones push en una aplicación PWA?

9. ¿Qué componente es necesario para garantizar que las notificaciones Push sean privadas y seguras?

A) Una API REST

B) Claves públicas y privadas para cifrado

C) Un archivo .env

D) Un CDN

10. Explica brevemente cómo funciona un inicio de sesión mediante tokens en una PWA. Menciona al menos los tres pasos clave.