

Aplicación Móvil SONIDO-ESTÉREO

Propuesta Técnica

Autor: Sebastián Valencia

Fecha: 28 de octubre de 2025

Resumen

SONIDO-ESTÉREO necesita una aplicación móvil (iOS y Android) para que usuarios descubran y compren entradas a conciertos según sus preferencias musicales. Este documento técnico detalla la elección tecnológica, arquitectura, estimación temporal, historias de usuario y la planificación necesaria para entregar el proyecto.

A. Elección Tecnológica

Recomendación: *Flutter* (multiplataforma) con Dart.

Por qué Flutter:

- Por qué unifica un solo código base para iOS y Android, reduciendo tiempo y coste de desarrollo y mantenimiento, permitiendo así tener una escalabilidad alta en cuanto a usuarios.
- Tiene una interfaz animada que permite una excelente experiencia de usuario (animaciones, mapas, lista) y por supuesto se adapta bien al teléfono.
- Esta actualizado constantemente con diferentes tipos de integraciones (Firebase, paquetes de pagos, deeplinks, notificaciones).

B. Arquitectura de Código

El patrón seleccionado para la ejecución de la aplicación será Clean Architecture + MVVM. Utilizaremos esta arquitectura porque la separación de la lógica de negocio con las vistas UI, hace que cada capa tenga una responsabilidad clara de sus funciones, llevando a tener u código limpio, bastante mantenible y escalable.

Capas y responsabilidades:

1. **Presentation (UI / ViewModel):** Widgets, pantallas, estados, valida entradas y expone estados para la UI.
2. **Domain (Use Cases / Entities):** Reglas de negocio, casos de uso. Independiente de frameworks.

3. **Data (Repositories / Data Sources):** Implementaciones que consumen servicios api externo para la obtención de datos necesarios.

C. Estimación Temporal del Proyecto

Estimación total: 10 semanas (2 meses y medio)

Cronograma general (4 Sprints de 2 semanas y la ultima de 2 semanas y media):

- **Sprint 1:** Infraestructura, autenticación, modelo de datos, listado básico de eventos.
- **Sprint 2:** Filtros por preferencias, visualización eventos pasados/futuros, perfil de usuario.
- **Sprint 3:** Compra de entradas (flujo de orden), integración QR de pago, límite 6 entradas.
- **Sprint 4:** Notificaciones push, visualización de entradas adquiridas, pruebas, correcciones y preparación para publicación.

Margen: En caso de retrasarse en algún sprint, se prevé aumentar 10 días adicionales a la estimación por imprevistos generales.

D. Historias de Usuario

Cada historia incluye: Descripción, criterios de aceptación, complejidad y tiempo estimado

Registro de usuario	
Descripción: Como usuario, quiero registrarme indicando mi género musical preferido para recibir eventos relevantes.	Criterios de aceptación: Formulario de registro (nombre, email, contraseña o red social), selección obligatoria de al menos 1 género, usuario creado en Auth y perfil.
Complejidad: M	Tiempo estimado: 2 días

Iniciar sesión	
Descripción: Como usuario, quiero iniciar sesión de forma segura con mi correo y contraseña.	Criterios de aceptación: Formulario de sesión (correo y contraseña), Persistencia datos de sesión.
Complejidad: S	Tiempo estimado: 1 días

Listar eventos filtrado por preferencias	
Descripción: Como usuario, quiero ver eventos recomendados según mis géneros preferidos.	Criterios de aceptación: Lista principal muestra eventos cuyo género coincide; opción para ver "otros géneros".
Complejidad: M	Tiempo estimado: 3 días

Visualización de eventos pasados y futuros	
Descripción: Como usuario, quiero ver detalles del evento (fecha/hora, lugar, imágenes) y diferenciar eventos pasados y futuros.	Criterios de aceptación: Vista de detalle; etiqueta si el evento es pasado; filtros por fecha.
Complejidad: M	Tiempo estimado: 3 días

Agregar evento al carrito y compra (máx 6 entradas)	
Descripción: Como usuario, quiero seleccionar hasta 6 entradas y completar la compra.	Criterios de aceptación: Validación de límite 6; creación de orden; reserva temporal de stock hasta pago; vista de confirmación.
Complejidad: L	Tiempo estimado: 5 días

Pago dentro de la App por QR	
Descripción: Como usuario, quiero pagar mi orden mediante un pago por QR integrado.	Criterios de aceptación: Generación de QR (integración con API del proveedor), ver estado de pago (pendiente / confirmado), Algoritmo que esta observando el pago con éxito para generar la orden.
Complejidad: L	Tiempo estimado: 7 días

Visualización de entradas compradas	
Descripción: Como usuario, quiero ver mis entradas con código QR para acceso al evento.	Criterios de aceptación: Listado de entradas, detalle con QR único, posibilidad de re-enviarlas por correo.
Complejidad: M	Tiempo estimado: 3 días

Notificaciones push publicitarias	
Descripción: Como administrador, quiero enviar notificaciones push a todos los usuarios con mensajes publicitarios.	Criterios de aceptación: Módulo en backend para crear y enviar notificaciones, funciones para enviar notificaciones a tokens; los usuarios reciben notificación; usuario puede desactivar notificaciones en perfil.
Complejidad: L	Tiempo estimado: 7 días

Perfil de usuario y configuración	
Descripción: Como usuario, quiero editar mi perfil y cambiar géneros preferidos.	Criterios de aceptación: Edición de perfil, correo bloqueado a edición, actualización de género.
Complejidad: M	Tiempo estimado: 5 días

E. Planificación de Sprints (Detalle)

Sprint 1 (2 semanas)

Infraestructura & Autenticación

- *Objetivo:* Tener una base de proyecto, base de datos creado y bien estructurado.
- *Historias:* Para este sprint se va abordar el registro de usuario con su género preferente, iniciar sesión con la cuenta previamente registrada y ver el Perfil a nivel básico, solo informativo (Historia: 1 y 2).
- *Entregables:* Proyecto Flutter inicial, flujo de inicio sesión y registro de usuario, guardado de datos permanente.

Sprint 2 (2 semanas)

Eventos de la aplicación y edición de usuario

- *Objetivo:* Implementar listado y detalle de eventos, filtros por preferencia y fijar fechas de las actividades posteriores., como también editar datos del usuario como la preferencia del género musical
- *Historias:* Listado de eventos y detalle del mismo ya sea uno pasado o nuevo, perfil del usuario y configuración (Historia: 3, 4 y 9).
- *Entregables:* Pantallas de eventos, filtros, tests unitarios básicos.

Sprint 3 (2 semanas)

Compra & Pagos QR

- *Objetivo:* Flujo de compra completo y pago por QR.
- *Historias:* Compra de usuario hasta 6 entradas, pagar Qr para la creación de la orden. (Historia 5 y 6)
- *Entregables:* Flujo de carrito, generación de QR, integración con proveedor o simulador de pago, Observador para confirmar pago exitoso.

Sprint 4 (2 semanas y media)

Entradas, Notificaciones, QA y Publicación

- *Objetivo:* Mostrar entradas compradas, notificaciones push, pruebas y preparación para lanzar a producción
- *Historias:* Visualizar las entradas compradas del usuario, creación y envío de notificaciones a los usuarios para publicidad. (Historia 7 y 8)
- *Entregables:* APK de prueba, documentación final, acceso a github para proyecto completo y/o actualización del flujo.

F. Comandos Git utilizados

Instrucciones para crear el repositorio local y subir el PDF final

1.- Crear repositorio local con el archivo pdf adentro, entrar al repositorio y una vez adentro ejecutar los siguientes comandos.

Ejemplo (E:\CarreraSistemas\Diplomado-Apps-moviles-multiplataforma\Fundamentos-basicos\SONIDO-STEREO-APP-SVS\Evaluacion-sebastianValencia.pdf):

```
# 1. git init
# 2. git add .
# 3. git commit -m "Commit inicial"
```

2.- Ahora creamos el repositorio público en github para poder subir el repositorio local, siguiendo estos comandos:

```
# 1. git remote add origin https://github.com/sebastian246007/sonido-
stereo-app-svs.git
```

```
# 2. git branch main
```

```
# 3. git checkout main
```

```
# 4. git branch --set-upstream-to=origin/main main
```

```
# 5. git push
```

Nota: En mi caso para poder subir, tuve que iniciar sesión, pero de una manera peculiar, ingresando mi correo, y de contraseña poniendo el token personal clásico, y funciono para poder subir al repo.

Fin de propuesta Técnica.