
PRESENTACIÓN DE PROYECTO

PROPÓSITO

El proyecto final tiene como objetivo que cada alumno aplique los conocimientos adquiridos durante el curso desarrollando una aplicación móvil completa, ya sea en **Android (Kotlin/Jetpack Compose)**, **iOS (Swift/SwiftUI)** o **Flutter (Dart)**.

La app debe demostrar un flujo funcional, integración con servicios externos o API, persistencia local y un diseño basado en buenas prácticas de arquitectura y manejo de estado.

FORMATO DE PRESENTACIÓN

Cada alumno tiene entre 5 a 10 minutos para su presentación.

- La exposición se realizará a través de Zoom.
- Se debe compartir pantalla para demostrar el funcionamiento de la aplicación.
- Opcionalmente, pueden presentar una diapositiva breve (1 o 2 diapositivas) con el nombre del proyecto y objetivo de la aplicación.

CONTENIDO DE LA PRESENTACIÓN

Cada alumno deberá presentar una aplicación móvil funcional y documentada para **uno de los siguientes casos de uso**:

A. **Tracking/Ubicación**: mostrar en mapa la ubicación del usuario u objetos, con listado y detalle.

B. **Mensajería básica**: chat simple con envío/recepción de mensajes.

REQUISITOS TÉCNICOS MÍNIMOS

Estructura del proyecto

- Código en Kotlin (Android), Swift (iOS) o Dart (Flutter).
- Arquitectura basada en buenas prácticas (MVVM, separación de capas, manejo de estado adecuado por plataforma).

- Navegación clara entre pantallas (lista ↔ detalle).

Consumo de datos

- Integración con un API (propia, pública o mock).
- Manejo de errores, estados de carga y conectividad.
- Persistencia local (Room, CoreData/SwiftData, sqflite/Drift, etc.).

Seguridad

- Manejo seguro de credenciales (almacenamiento en Keystore/Keychain/Secure Storage).
- Autenticación básica (por ejemplo, login con email/password o token simulado).
- Manejo correcto de permisos del dispositivo (ubicación, cámara, notificaciones según corresponda).

Documentación

- **README** claro con:
 - Objetivo del proyecto.
 - Pasos de instalación y ejecución.
 - Dependencias principales.
- Capturas de pantalla o gifs opcionales para ilustrar los flujos principales.

Despliegue / Build

- Generación de build de desarrollo:
 - Android: APK/AAB.
 - iOS: build para simulador o TestFlight.
 - Flutter: build para al menos una plataforma (Android o iOS).

EXTRAS VALORADOS

- Tema claro/oscuro persistente.
- Integración con mapas, cámara o notificaciones.

-
- Internacionalización (i18n).
 - Accesibilidad básica (labels, contrastes).
 - Pruebas unitarias (al menos de capa de lógica/estado).
 - CI/CD básico para build automatizado.

CRITERIOS DE EVALUACIÓN

El proyecto será evaluado con el siguiente peso:

- **Funcionalidad y cumplimiento de requisitos técnicos (40%)**
Flujo principal completo, navegación correcta, API integrada, persistencia local.
- **Calidad del código y buenas prácticas (20%)**
Arquitectura clara, separación de capas, uso adecuado de estado, manejo de errores.
- **Seguridad, autenticación y documentación (20%)**
Manejo de permisos y tokens, README completo, uso responsable de secretos.
- **Presentación y explicación del proyecto (20%)**
Demostración fluida, claridad en decisiones técnicas, storytelling del proyecto.