

Tarea 1 : Android

Tarea 1: Programación en Android

El estudiante resolverá un problema del mundo real donde el hardware del dispositivo va a ser el protagonista. En Android nativo (Kotlin/Jetpack Compose), la autenticidad radica en usar los sensores, la persistencia de datos y el ciclo de vida de la aplicación.

Actividad: "SafeWalk: El Centinela de Bolsillo"

1. El Desafío (Problema Real)

Muchos estudiantes universitarios caminan solos por el campus de noche. Necesitan una herramienta que detecte situaciones de riesgo (como una caída o una corrida repentina) y envíe una alerta automática.

2. Misión Técnica

Los estudiantes deben construir una aplicación funcional en Android Nativo que cumpla con:

1. Uso de Sensores: Implementar el acelerómetro para detectar un "impacto" (caída).
 2. Persistencia: Guardar un número de contacto de emergencia usando Preferences DataStore o Room.
 3. Intenciones (Intents): Lanzar una llamada o enviar un SMS automático ante la emergencia.
 4. UI Moderna: Construir la interfaz con Jetpack Compose.
-

3. Fases de la Actividad

Fase 1: La Interfaz Reactiva (Jetpack Compose)

- Crear una pantalla con un botón de "Pánico" grande y un interruptor de "Modo Vigilancia".
- Concepto auténtico: Manejo de estados (remember, MutableState). Si el modo vigilancia está activo, el botón debe cambiar de color y estado.

Fase 2: El Sensor de Gravedad

- Acceder al SensorManager y escuchar los cambios en el acelerómetro.
- Cálculo Técnico: Implementar una fórmula sencilla para medir la magnitud del vector fuerza G:

$$\sqrt{x^2 + y^2 + z^2}$$

- Desafío: Si la magnitud supera un umbral (ej. 30 m/s²), disparar el evento de alerta.

Fase 3: El Ciclo de Vida y Permisos

- Solicitar permisos en tiempo de ejecución (Manifest.permission.SEND_SMS).
- Concepto auténtico: Gestionar qué pasa si la app se va a segundo plano. ¿Sigue funcionando el sensor? (Introducción a WorkManager o Foreground Services).

4. Evaluación de Aprendizaje Auténtico

Prueba de Campo:

- El estudiante debe colocar el teléfono en su bolsillo y simular un tropiezo (sobre una colchoneta).
- La app debe:
 1. Detectar la caída.
 2. Iniciar una cuenta regresiva de 10 segundos (para evitar falsos positivos).

3. Vibrar y, si no se cancela, mostrar el mensaje: *"Alerta enviada al contacto: [Nombre]"*.
-

5. Guía de Conexión Técnica

Mapa de navegación del proyecto:

| Componente | Responsabilidad |
|---------------------|--|
| MainActivity.kt | Punto de entrada y manejo de permisos. |
| SensorViewModel.kt | Procesa los datos del acelerómetro y mantiene el estado de la UI. |
| DataStoreManager.kt | Guarda y recupera el teléfono de emergencia. |
| AlarmScreen.kt | UI construida con Compose que reacciona a los cambios del ViewModel. |