

# Ampliación de Conocimientos - Puntos 5 y 6

## 5. Desarrollo de aplicaciones móviles en Android

El desarrollo en Android se apoya en el uso del SDK (Software Development Kit), que proporciona APIs y herramientas para acceder a las funcionalidades del sistema. Permite trabajar con hardware como GPS, cámara, sensores o Bluetooth, así como gestionar procesos en segundo plano y almacenamiento en SQLite.

### Ejemplos prácticos de uso de APIs:

- GPS y mapas: una app de turismo puede usar la API de localización para mostrar la posición actual y recomendar lugares cercanos.
- Cámara y multimedia: una aplicación de escaneo de documentos que activa la cámara, convierte la imagen en PDF y la guarda en el dispositivo.
- Procesos en segundo plano: una app de música que sigue reproduciendo aunque el usuario abra otras aplicaciones.
- SQLite: una aplicación de notas que guarda información localmente y permite acceder sin conexión a internet.
- Notificaciones: una app de recordatorios que muestra avisos programados usando NotificationManager.

### Capas del sistema operativo Android:

- Aplicaciones: apps de usuario (ej. WhatsApp, Gmail).
- Framework de aplicaciones: clases para actividades, servicios, vistas y proveedores de contenido.
- Android Runtime: máquina virtual ART/Dalvik que ejecuta el código Java/Kotlin.
- Librerías nativas: OpenGL para gráficos, SQLite para bases de datos.
- Núcleo Linux: gestiona memoria, procesos, red y drivers de hardware.

## 6. Ciclo de vida de una aplicación en Android

El ciclo de vida de una Activity describe cómo responde una aplicación a los cambios de estado, como abrirse, minimizarse o cerrarse. Comprender estos estados permite optimizar recursos y evitar errores.

### Estados principales de una Activity:

- Running: la aplicación está en primer plano y es totalmente interactiva.
- Paused: visible parcialmente (ej. un cuadro de diálogo encima), pero sin interacción.
- Stopped: no es visible, pero mantiene su estado en memoria.
- Destroyed: liberada de memoria, requiere reinicio desde cero.

### Ejemplo de ciclo de vida:

Una aplicación de reproductor de vídeo: - Al abrirla: se ejecutan onCreate(), onStart() y onResume(), mostrando la interfaz y comenzando el vídeo. - Si el usuario recibe una llamada: se ejecuta onPause(), pausando el vídeo. - Al colgar: vuelve a onResume(), reanudando el vídeo. - Si el usuario cierra la app: se ejecutan onPause(), onStop() y onDestroy(), liberando recursos.

## **Métodos clave del ciclo de vida:**

- onCreate(): inicializa componentes y recursos (ej. cargar interfaz).
- onStart(): la Activity es visible, aún no interactiva.
- onResume(): la Activity entra en primer plano y es interactiva.
- onPause(): liberar recursos temporales (ej. detener animaciones).
- onStop(): la Activity ya no es visible, guardar datos pesados.
- onDestroy(): limpieza final de recursos.
- onRestart(): se ejecuta cuando se relanza después de estar detenida.

## **Conclusión**

Dominar el SDK de Android y el ciclo de vida de una Activity es esencial para crear aplicaciones eficientes, seguras y optimizadas. Estos conocimientos permiten gestionar recursos limitados, aprovechar al máximo el hardware y garantizar una buena experiencia de usuario.