



ULPGC

**Escuela de
Ingeniería Informática**



Programación de Aplicaciones Móviles Nativas

Práctica 8: Recomendaciones de arquitectura para aplicaciones Android

Guillermo Marion López,
Karl Christian Deilmann,
Louka David Vanhoucke

Bloque II – S8 Recomendaciones de arquitectura para aplicaciones Android.

Guía de rendimiento de la app

Descripción: Esta guía de rendimiento para aplicaciones Android nos ofrece una visión general de bibliotecas, herramientas y prácticas recomendadas para mejorar la velocidad de inicio de nuestra aplicación. El procesamiento fluido, el consumo eficiente de batería y la gestión de memoria, proporciona información y estadísticas sobre cómo lograr un rendimiento óptimo, ayudando a los desarrolladores a optimizar sus aplicaciones para una mejor experiencia del usuario.

Decisión: Si, lo implementaremos en nuestro proyecto.

Justificación: El rendimiento de una aplicación es importante al publicarlo para los usuarios, queremos que los usuarios tengan una buena experiencia al usar nuestra aplicación. Obviamente hay muchas maneras de mejorar el rendimiento, nosotros nos focalizamos en Perfiles de inicio y optimización de poca RAM.

Cómo solicitar actualizaciones de ubicación

Descripción: Esta guía explica cómo solicitar actualizaciones periódicas de ubicación en una aplicación Android. Destaca la importancia de obtener la ubicación más reciente del dispositivo antes de iniciar las actualizaciones y muestra cómo usar el método llamado *requestLocationUpdates()*. Se aborda la influencia de la precisión, frecuencia y permisos de ubicación en las actualizaciones, con un ejemplo de código para implementar este proceso a través de un objeto *LocationRequest* y un *LocationCallback*.

Decisión: Depende del tiempo, nos gustaría

Justificación: Nos gustaría implementar un mapa en nuestra aplicación, que usará las funciones mencionadas en la descripción. Hay que añadir que no es nuestro objetivo principal implementar esta tecnología.

Cómo determinar el acceso a datos sensibles

Descripción: Esta página proporciona una visión general del funcionamiento de los permisos en Android, destacando su papel en la protección de datos y acciones restringidas. Se presenta un flujo de trabajo para el uso de permisos, tanto como descripciones de diferentes tipos de permisos, y se incluyen prácticas recomendadas. Se enfatiza la importancia de solicitar la cantidad mínima de permisos necesarios, asociar permisos con acciones específicas, considerar las dependencias de la aplicación y ser transparente con los usuarios sobre el acceso a datos sensibles y accesibles. Además, se aborda la creación de accesos explícitos al sistema y se menciona cómo los permisos también se aplican a los componentes del sistema de la aplicación.

Decisión: Si, lo implementaremos en nuestro proyecto.

Justificación: En nuestra opinión es esencial tener una buena protección de datos y un buen funcionamiento de los permisos, ya que nosotros tendremos un registro/login y guarda los datos de los usuarios.

Estilos y temas

Descripción: En Android, los estilos y temas permiten separar los detalles de diseño de la estructura y el comportamiento de la interfaz de usuario, similar a las hojas de estilo en el diseño web. Un estilo es una colección de atributos que define la apariencia de una única vista, como color, tamaño de fuente y fondo. Por otro lado, un tema es una colección de atributos aplicados a toda una app, actividad o jerarquía de vistas, influyendo en aspectos visibles y no visibles, como la barra de estado. Estos se declaran en un archivo de recursos llamado *styles.xml*.

Decisión: No, no lo implementaremos en nuestro proyecto

Justificación: En principio, no lo implementaremos en nuestro proyecto, porque tenemos otras funcionalidades más urgentes que este en concreto. Podría ser un “feature” de añadir, por ejemplo, un tema oscuro, pero por obligaciones no dará tiempo.

Introduction to animations

Descripción: En Android, las animaciones son herramientas visuales que mejoran la experiencia del usuario al indicar cambios en la aplicación. Se utilizan para notificar la carga de contenido o la disponibilidad de nuevas acciones, añadiendo un toque de calidad visual. Android ofrece diversas APIs de animación, como animación de gráficos, de visibilidad y movimiento, basada en física, y para cambios en el diseño. Los estilos y temas se definen en archivos de recursos, y se destacan prácticas recomendadas, como aplicar animaciones sutiles para mejorar la comprensión de cambios en la interfaz de usuario.

Decisión: Depende del tiempo que tendremos

Justificación: Añadir unas animaciones, podría estar muy bien para dar un poco más de vida a la aplicación o de mostrar el contenido de mejor forma, pero no será una funcionalidad principal de nuestra aplicación. Primero implementaremos todo lo necesario para que la aplicación sea funcional, si luego dará tiempo podremos implementarlo.