

Tema 1: Desarrollo de aplicaciones para dispositivos móviles

¿Qué aprenderás?

- La evolución experimentada por los dispositivos móviles.
- Cómo instalar y preparar un entorno de desarrollo para programar aplicaciones móviles.
- Los distintos componentes que conforman la arquitectura del sistema operativo Android.

¿Sabías que...?

- Uno de los mercados con mayor crecimiento en los últimos años ha sido el de la telefonía y los dispositivos móviles.
- Es posible crear aplicaciones de forma libre y gratuita que se pueden ejecutar en multitud de dispositivos basados en el sistema operativo Android.



1.1. Evolución de los dispositivos móviles

La variedad de dispositivos móviles disponibles en el mercado nos permite realizar fácilmente todas nuestras tareas diarias. De hecho, la línea que separa unos dispositivos de otros cada vez se hace más difícil de definir: los teléfonos tienen cada vez pantallas más grandes, al tiempo que las tabletas táctiles disponen de diferentes tamaños, de forma que la diferencia entre un teléfono

1.2. Limitaciones al ejecutar aplicaciones en dispositivos móviles

El reducido tamaño de los smartphones implica inevitablemente una serie de limitaciones de hardware que los mantienen claramente en desventaja frente a los ordenadores convencionales.

Además de disponer de pantallas de menor tamaño, las limitaciones se reflejan principalmente en la memoria RAM y la memoria persistente (almacenamiento interno, también llamada memoria ROM), una menor capacidad del procesador y la necesidad de adaptar el consumo de energía a la capacidad de una batería pequeña.



Si bien en el pasado estos factores suponían graves inconvenientes que limitaban el tipo de aplicaciones que se podían desarrollar, la tecnología ha evolucionado notablemente con el transcurso de los años. La deficiencia de las conexiones y la alta latencia se han visto solventadas con el desarrollo de las tecnologías 4G y 5G. La velocidad de proceso ha aumentado de manera considerable con los microprocesadores de varios núcleos y la introducción de procesadores



gráficos especializados. Sin embargo y a pesar de los avances en materia de hardware, los programadores deben tener presente el tamaño reducido de las pantallas de los dispositivos en los que se van a ejecutar las aplicaciones.

1.3. Entornos integrados de trabajo

Para facilitar el desarrollo de aplicaciones móviles existen diferentes herramientas de desarrollo, desde lenguajes de programación hasta emuladores. En la actualidad, el lenguaje de programación Java y el entorno de desarrollo Android Studio es la combinación más utilizada a la hora de desarrollar aplicaciones móviles.

Para desarrollar programas para dispositivos móviles Android es necesario instalar el software siguiente:

1. El lenguaje de programación Java.
2. El Android SDK (Software Development Kit, librerías de desarrollo de software)
3. Preparar un entorno de trabajo por primera vez.

Es posible desarrollar aplicaciones para dispositivos móviles con Android desde cualquier sistema operativo (GNU / Linux, MacOS, MS Windows) y con varios entornos de desarrollo; el oficial y más popular actualmente es el Android Studio aunque en el pasado muchos programadores utilizaban Eclipse y en Netbeans.

Actualmente únicamente hay que descargar de la web oficial de Desarrolladores de Android <https://developer.android.com/studio> el paquete de software de Android Studio que contiene todo lo necesario para poder empezar a desarrollar aplicaciones móviles.

En el siguiente video se detalla el proceso de instalación de Android Studio.



Video 1: Instalación de Android Studio

1.4. Módulos para el desarrollo de aplicaciones móviles

Google recomienda la instalación de ciertos componentes para comenzar con el desarrollo de aplicaciones Android. En la siguiente tabla se muestra algunos de los componentes que forman parte de un entorno básico de desarrollo

Componente	Descripción
SDK Tools	La instalación de Android Studio ya incorpora librerías y herramientas como el AVD (Android Virtual Device) que permite ejecutar y compilar una aplicación en un emulador.
Documentation	La documentación es útil ya que permite trabajar offline y disponer de referencias a la API desde el propio entorno de desarrollo.
Samples	Código fuente de varias aplicaciones de ejemplo que se puede cargar como proyectos y facilitan el aprendizaje.
USB Driver	Componente necesario para poder ejecutar aplicaciones desde Android Studio en entorno Windows en nuestro propio dispositivo móvil. Este componente no es necesario si ejecutamos Android Studio en Linux o Mac.



Google API	Librerías externas de Google requeridas por ciertas aplicaciones que hacen uso de componentes de Google, como por ejemplo Google Maps.
Plataformas SDK	Es posible descargar versiones de distintas plataformas de Android (versiones como Android 4.4 Kit Kat, Android 5.0 Lollipop, Android 7.0 Nougat, ...). De esta forma es posible compilar, ejecutar y probar nuestra aplicación en las distintas versiones de Android.

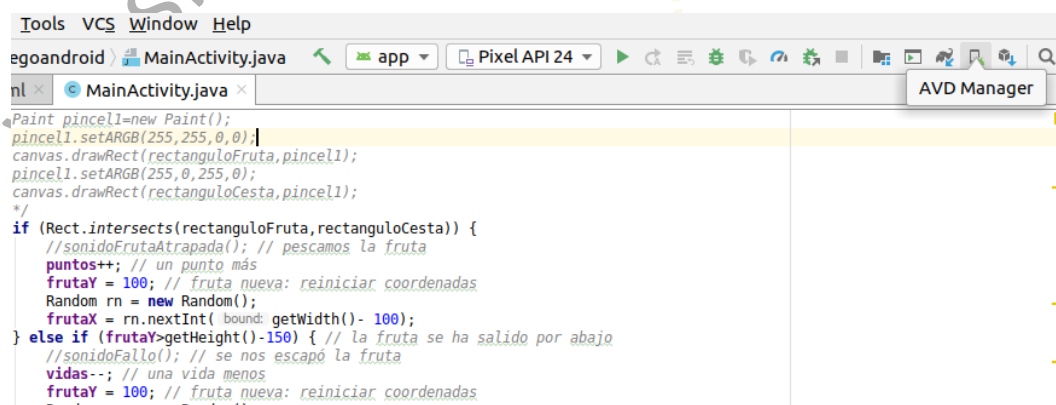
Tabla: Componentes adicionales de Android Studio

1.5.Emuladores

El AVD Manager es una herramienta que forma parte del Android SDK y que ejecuta emuladores de dispositivos móviles Android. Se pueden crear nuevos dispositivos a partir de las imágenes disponibles en la instalación de SDK y configurar cada emulador que se quiera utilizar con una arquitectura determinada y sus características: versión de Android, memoria SD disponible, tamaño de la pantalla, etcétera. Para mejorar el rendimiento del emulador, si nuestro microprocesador es compatible, se recomienda instalar el Intel® Hardware Accelerated Execution Manager siguiendo las instrucciones del siguiente enlace: [Intel HAXM](#).

1.5.1. Ejecución del AVD Manager

Podemos iniciar la ejecución del AVD Manager para gestionar los emuladores que necesitemos desde la opción del menú de Android Studio.



Ejecución del AVD Manager



1.5.2. Creación de un nuevo emulador

Podemos crear tantos emuladores como queramos, eligiendo distintas plataformas ya sean teléfonos, tablets, gadgets Android o incluso Android TV.

Cada tipo de dispositivo Android aparece listado con unas determinadas características técnicas, destacando el tamaño de la pantalla y la resolución. Por último, seleccionamos la imagen del sistema operativo que ejecutará, es decir, la versión de Android que tendrá instalado ese emulador.



AVD Manager: listado de emuladores

El AVD Manager nos mostrará una lista de todos los emuladores que hemos creado con la posibilidad de añadir, borrar o clonar los existentes.



1.6. Configuraciones. Tipos y características

Android tiene una licencia de código libre y está disponible de forma gratuita para los fabricantes de forma que pueda realizar personalizaciones y adaptaciones para sus dispositivos. No existen configuraciones fijas de hardware ni de software, aunque en general Android soporta las siguientes características:

- Mensajería: soporta SMS (Short Message Service, servicio de mensajes cortos) y MMS (Multimedia Messaging Service, servicio de mensajería multimedia).
- Soporte de hardware: acelerómetros, cámara, brújula, sensor de proximidad y GPS.
- Almacenamiento: utiliza SQLite, una base de datos relacional ultraligera para el almacenamiento de datos.
- Formatos: soporta multitud de formatos multimedia (imágenes, sonido, vídeo...).
- Conectividad: soporta redes GSM / EDGE, HSDPA, LTE, CDMA, UMTS, Bluetooth, Wifi y WiMAX.
- Android está pensado para poder utilizarse en multitud de dispositivos, tales como teléfonos inteligentes, tabletas, lectores de libros electrónicos, netbooks, reproductores de MP3 y MP4, televisores, relojes. . .

Hoy en día Android se utiliza en multitud de dispositivos, tales como teléfonos inteligentes, tabletas, lectores de libros electrónicos, netbooks, reproductores de MP3 y MP4, televisores, relojes, etcétera.

1.7. Arquitectura y requerimientos

La arquitectura del sistema operativo Android consta de cinco capas. El núcleo es un kernel basado en Linux 2.6 sobre el cual se han añadido librerías preinstaladas que apoyan las operaciones más habituales (conectividad de red, multimedia, base de datos).



Junto a estas librerías encontramos el Android Runtime, responsable de ejecutar las aplicaciones Android. El Android Runtime contiene la máquina virtual Dalvik que es responsable de ejecutar el *bytecode* o código binario de las aplicaciones Android.

Por un lado, esta estructura de máquina virtual permite aislarnos de las características de los dispositivos físicos, que en el caso de los dispositivos móviles es muy variable y depende de las diferentes tecnologías que utilizan los fabricantes de esos dispositivos.

Por otro lado, también permite aumentar la seguridad. Cada aplicación se ejecuta en una instancia independiente de la máquina virtual, usando un usuario Linux independiente. Las máquinas virtuales actúan como un Sandbox o sistema de aislamiento, de forma que cuando una app deja de funcionar adecuadamente, no debe alterar el estado del teléfono. De esta forma se persigue que un dispositivo móvil no deje nunca de responder, aunque una app esté "colgada".



Arquitectura del sistema operativo Android

El framework de aplicaciones proporciona a los desarrolladores las diferentes características y funcionalidades de Android para que las puedan utilizar en sus aplicaciones.



Finalmente, en la capa más alta encontramos las aplicaciones que se distribuyen con el dispositivo Android (funcionalidad de teléfono, contactos, navegador, etcétera), así como otras aplicaciones que se descargan y se instalan desde la tienda de Google, la Google Play Store.

1.8. Ciclo de vida de una aplicación

Independientemente del fabricante y el modelo de móvil, el código fuente es el mismo y por tanto podemos hablar de un lenguaje multiplataforma. Dos dispositivos móviles de fabricantes diferentes, con tamaños de pantalla diferentes, con o sin GPS, con diferente tamaño y resolución de pantalla, se programan de forma idéntica siempre y cuando la versión de Android que utilizan sea la misma.

El programador únicamente debe preocuparse de escribir código diferenciado en el caso de acceder a funcionalidades adicionales que no siempre estarán disponibles en todos los terminales.

Por ejemplo, si queremos realizar una aplicación para realizar pagos utilizando la tecnología NFC (Near Field Communications), la aplicación contiene código para indagar si ese servicio está o no disponible en el teléfono.

El ciclo de vida de una aplicación de Android se podría definir de la siguiente manera:

1. Descubrimiento de la aplicación, tanto desde Google Play (el más habitual) como de otra forma.
2. Descarga e instalación de la aplicación.
3. Ejecución de la aplicación.
4. Actualización de la aplicación si existen versiones más modernas.
5. Desinstalación de la aplicación.



Recursos y enlaces

- [Guías para desarrolladores de Android](#)



- [Manual “Crear tu primera app con Android Studio”](#)



Conceptos clave

- **Android:** Es un sistema operativo adquirido por Google en el año 2005 que ostenta la mayor cuota de mercado en cuanto a sistemas operativos instalados en teléfonos inteligentes. Entre las razones de su éxito destacan ser un proyecto de código libre e incorporar un mercado de aplicaciones que permite a los usuarios buscar e instalar aplicaciones para sus dispositivos.
- **Android Studio:** Entorno de desarrollo que facilita el desarrollo, la compilación y el testeo de aplicaciones móviles utilizando el lenguaje de programación Java.
- **Kotlin:** Lenguaje de programación desarrollado por la empresa rusa JetBrains pensado para ser compilado en cualquier máquina virtual Java. En el año 2017 Google establece Kotlin como lenguaje oficial para Android al mismo nivel que Java.
- **Máquina virtual:** Entorno que permite a cada aplicación ejecutarse en un proceso propio. A partir de la versión 5.0 de Android, la máquina virtual Dalvik evoluciona a una nueva máquina virtual llamada Art, que incrementa el rendimiento general de las aplicaciones.



Test de autoevaluación

1. ¿Cuál de las siguientes limitaciones habría que tener en cuenta a la hora de desarrollar aplicaciones móviles?
 - a) Velocidad del procesador.
 - b) Tamaño de la pantalla.
 - c) Deficiencia de las conexiones.
 - d) Falta de funcionalidades multimedia.
2. ¿Qué es un AVD en Android?
 - a) Un dispositivo virtual o emulador.
 - b) Una máquina virtual en la que se ejecuta el código Java compilado.
 - c) Un dispositivo móvil conectado por USB
 - d) Una librería que forma parte del SDK Tools.
3. ¿Qué nombre recibe la capa del sistema operativo Android que se encarga de ejecutar las aplicaciones?
 - a) Application Framework.
 - b) Kernel.
 - c) Android Runtime.
 - d) Ninguna de las respuestas anteriores es correcta.
4. ¿Cuál es el componente Windows que nos permite ejecutar directamente en nuestro propio teléfono móvil las aplicaciones que estamos desarrollando en Android Studio?
 - a) Máquina virtual.
 - b) Emulador.
 - c) Servicio Remoto.
 - d) USB Driver.



Ponlo en práctica

Actividad 1

Realiza la instalación de entorno de desarrollo Android Studio en tu ordenador. Recuerda, que debes descargar el programa ejecutable de la web oficial de Android; de esta forma, se instalarán automáticamente en tu ordenador el entorno de desarrollo, el lenguaje de programación Java y las librerías de Android necesarias.





Solucionarios

Test de autoevaluación

1. ¿Cuál de las siguientes limitaciones habría que tener en cuenta a la hora de desarrollar aplicaciones móviles?
 - a) Velocidad del procesador.
 - b) Tamaño de la pantalla.**
 - c) Deficiencia de las conexiones.
 - d) Falta de funcionalidades multimedia.
2. ¿Qué es un AVD en Android?
 - a) Un dispositivo virtual o emulador.**
 - b) Una máquina virtual en la que se ejecuta el código Java compilado.
 - c) Un dispositivo móvil conectado por USB
 - d) Una librería que forma parte del SDK Tools.
3. ¿Qué nombre recibe la capa del sistema operativo Android que se encarga de ejecutar las aplicaciones?
 - a) Application Framework.
 - b) Kernel.
 - c) Android Runtime.**
 - d) Ninguna de las respuestas anteriores es correcta.
4. ¿Cuál es el componente Windows que nos permite ejecutar directamente en nuestro propio teléfono móvil las aplicaciones que estamos desarrollando en Android Studio?
 - a) Máquina virtual.
 - b) Emulador.
 - c) Servicio Remoto.
 - d) USB Driver.**



Ponlo en práctica

Actividad 1

Realiza la instalación de entorno de desarrollo Android Studio en tu ordenador. Recuerda, que debes descargar el programa ejecutable de la web oficial de Android; de esta forma, se instalarán automáticamente en tu ordenador el entorno de desarrollo, el lenguaje de programación Java y las librerías de Android necesarias.

Solución:

En primer lugar, es necesario descargar el programa de la web oficial de desarrolladores de Android: <https://developer.android.com/studio/?hl=ES>

A continuación, inicia el proceso de instalación de Android Studio en Windows siguiendo los siguientes pasos:

1. Haz doble clic en el archivo .exe descargado para iniciarlo.
2. Sigue los pasos del asistente de configuración en Android Studio y asegúrate de instalar los paquetes de SDK que recomiende.
3. Cuando haya nuevas herramientas y otras API disponibles, Android Studio te informará por medio de una ventana emergente. También puedes buscar actualizaciones si haces clic en *Help > Check for Update*