

Curso de Dispositivos Móviles

Prof. Gustavo Márquez Flores

Clase del 30 de Enero 2023

Requisito: Saber programar en Java

Evaluación:

Exámenes: 30% En línea Tarea examen

Tareas: 40% 5 o 6 programas en el curso

Proyecto final: 30%

Actividades de los ayudantes: Subir de calificación

Curso: Online, el link de acceso es que viene en el temario oficial en los horarios.

Presentaciones en Power Point, PDFs, notas de Word, IDE.

Se envían por mail todo el material del curso desde notas hasta los programas

Vistos en clase

Ayudantes: En los horarios vienes los mails de los ayudantes. Escribirles

Para saber el link de las sesiones de martes y jueves

Asunto: Cita de clase primera sesión

Primera tarea:

Enviar por mail: gmarquez@ciencias.unam.mx

El correo electrónico indicando su nombre y el de la materia

⇒ Objetivo es formar una lista de distribución del curso

Clase del 1° de Febrero 2023

Introducción.

¿ Que es un dispositivo Móvil ?

Se asocia con celulares, tablets.

1. ¿ Qué es un dispositivo Móvil ?

Los dispositivos móviles son aparatos electrónicos de tamaño pequeño (que caben en la palma de una mano) que realizan distintas funciones como transmitir voz y datos, almacenar y procesar información y que por su tamaño se caracterizan porque pueden ir a cualquier lado. Son aparatos móviles en el que el emisor o el receptor están en movimiento.

Algunos componentes periféricos de los dispositivos móviles.

Algunos dispositivos móviles para realizar las tareas para los cuales fueron creados, cuentan además con dispositivos periféricos, por ejemplo:

- Lector de código de barras.
- Impresora.
- Cámara fotográfica.
- Cámara de Vídeo.
- Teclado.
- Mouse.
- Pluma digital
- Bluetooth
- Pantalla táctil
- Lector de memoria externa Micro SD
- Puerto Micro USB, Mini USB
- Lector de teclado adicional
- Báscula

Algunos dispositivos móviles ya tienen incorporados estos periféricos. Por ejemplo, los teléfonos celulares cuentan además con cámara fotográfica y de video integrados.

Algunas tecnologías asociadas con los dispositivos móviles.

Algunas tecnologías asociadas con los dispositivos móviles son las siguientes:

- Reconocimiento de voz. Por ejemplo, marcación por voz.
- Procesamiento de imágenes. Por ejemplo, en deformación de imágenes y en reconocimiento de firma personal.
- Reconocimiento de patrones. Uso de pluma digital para escribir y guardar notas.
- Encriptación. Desarrollo de aplicaciones encriptadas por el tamaño limitado de espacio de memoria en el dispositivo.
- Base de datos. Control de agenda personal, contactos, pendientes, notepad, etc.
- Reproducción de sonido y video. Multimedia.
- Mensajes abreviados. Autocompletar texto

Clasificación los dispositivos móviles.

Clasificar por el tamaño X

Clasifican por el uso:

Industriales: aplicaciones en la industria, no son de uso común. Se destacan por el uso de periféricos específicos: impresora, escáner, báscula. Tamaño teclado es más grande.

Entretenimiento y en el hogar: Juego y diversión => consolas de juego Atari, Xbox 360.

Productividad: Ayudar a las personas en sus tareas cotidianas. Agendas electrónicas, calculadoras, Aplicaciones de escritorio: Procesadores de palabras, tablas, presentaciones. Comunicación: Parlante, señales (Web, mensajes escritos).

Dispositivos móviles de uso personal y del hogar :

➤ Comunicación parlante.

➤ Teléfonos Celulares y radiocomunicadores.



➤ Productividad.

➤ Palms , PDA's, Organizador y Agenda personal.



Agenda Personal



Control de Actividades

➤ Entretenimiento.



Juegos Personales



Dispositivos móviles de uso en la industria :

➤ **Localizadores vía satélite.**



➤ **Almacén : Registro de inventario.**



➤ **Comercial (*retail*) : Levantamiento de pedidos, facturación y devolución.**



➤ **Manufactura : Leer y procesar código de barras o de producto.**



Registro del UPC



Ensamble de partes
motrices



Inspección de línea
de ensamble

➤ **Aplicaciones diversas.**



Aplicaciones bancarias
de firma



Eventos
Control de participantes



Infracciones



Seguridad Reconocimiento



Farmacéutica

Lectura de muestras



Servicio Postal

Entrega de paquetes



Servicios Públicos

Lectura de consumo



Industria Militar



Seguros

Accidentes



Escuela

Biblioteca



Restaurante o cafetería

Registrar una orden



Eventos Especiales

Registro de invitados



Entretenimiento

Cine



Industria Militar

Estrategias



Industria Minera

Control de las góndolas



Construcción



Hospitales



Control de pacientes

Manejo de Mapas



Ganadería

Control de Reses

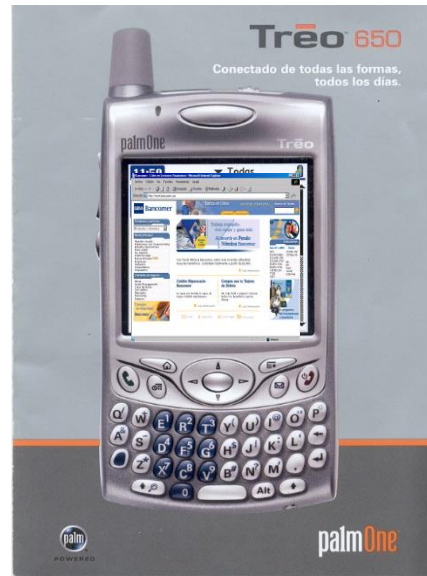


Aparatos deportivos. Control de peso y

Programas de entrenamiento.



Weather Locations		
Oslo	33°F	☁
Seattle	40°F	☀
Chicago	44°F	☁
New York	41°F	☁
Paris	71°F	☀
Beijing	55°F	☀
Back		



Navegador WAP

```
<?xml version="1.0"?>
```

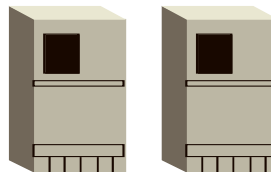
```
<!DOCTYPE wml PUBLIC "-//WAPFORM//DTD WML 1.3//EN"
"http://www.wapforum.org/DTD/wml13.dtd">
```

```
<wml>
```

```
<card id="card1" title=Descarga de Aplicacion"
```

```
Bienvenido a mi sitio movil <br>
```

<http://www.bancomer.com>



Clase del 3 de Febrero 2023

+ Términos que se asocian con los dispositivos móviles

+ Smartphone:

+ Teléfono celular con pantalla táctil, que permite al usuario conectarse a internet, gestionar cuentas de correo electrónico e instalar otras aplicaciones y recursos a modo de pequeña computadora. Sinónimo: Teléfono inteligente

+ Tablet:

+ Una Tablet es una computadora con forma rectangular de 6, 8, 10, 12 pulgadas y muy delgada, un cuarto de pulgada aproximadamente, sin teclado y con una gran pantalla sensible al tacto. El término "Slate" pasó a ser usado hace poco también para referirse a una Tablet, después de que los rumores de que la tablet de Apple podría llamarse iSlate (la empresa registró el dominio islate.com). Incluso Steve Ballmer, CEO de Microsoft, también usó el término "Slate" cuando se refirió a la Tablet de HP con Windows 7.

+ iPhone:

+ El **iPhone** es un **teléfono móvil** con herramientas multimedia, desarrollado por la empresa estadounidense **Apple Inc.** Este tipo de dispositivo es conocido como **smartphone** o **teléfono inteligente**, ya que cuenta con conexión a **Internet**, pantalla táctil, cámara de fotos, la capacidad de reproducir música y películas, entre otras características. Salió a la venta el **29 de junio de 2007**

+ iPad

+ Es la versión de la Tablet de Apple Inc.

+ iPod

+ Es un reproductor de música portátil y de pequeño tamaño creado por Apple Computer. En realidad, es un reproductor de audio digital y un disco duro portátil con capacidad de 10GB, 15GB o 30GB. Puede ser conectado con una computadora a través de un puerto Firewire o USB.

+ Teléfono Celular

+ Aparato telefónico de pequeño tamaño, portátil, sin cables externos, para poder hablar desde cualquier lugar, siempre que sea dentro del área de cobertura del servicio que lo facilita. Aparato electrónico que recibe y envía comunicaciones sonoras a larga distancia y está conectado a una red telefónica. Además de tener otras funcionalidades como agenda, calculadora, reproducción de sonido, video, señales de radio y de televisión.

+ SmartSwatch

+ Un smartwatch es un dispositivo electrónico portátil que es colocado en la muñeca de una persona, el cual le ofrece capacidades y funcionalidades similares a las de un Smartphone. Es un teléfono inteligente: ofrece capacidades y funcionalidades similares a las de un Smartphone.

+ Phablets

+ La palabra phablet viene de la mezcla entre smartphone y Tablet. Por tanto, **un híbrido** que busca combinar elementos de las dos partes: las funciones de un teléfono inteligente, pero a la vez un tamaño similar al de una tableta, pero más pequeño por supuesto.

Normalmente son fáciles de identificar a simple vista pero cualquier teléfono de cinco pulgadas o más entraría dentro de esta categoría. A fin de cuentas, el tamaño de la pantalla es el principal criterio.

Con un phablet tendremos batería para todo el día. Son un poco más grandes y pesados pero una vez que los pruebas, no hay vuelta atrás. Ejemplo:

Samsung Galaxy Note 2



Huawei Ascend Mate



Samsung Galaxy Mega



Sony Xperia Z



+ FonBlet

+ Término comercial acuñado por Samsung para referirse a sus teléfonos móviles con características más avanzadas y más "grandes". Tamaño, resolución y procesador.

+ Gadget

+ Un Gadget es un dispositivo electrónico que tiene un propósito y una función específica, generalmente de pequeñas proporciones, práctico y a la vez novedoso. Los gadgets suelen tener un diseño más ingenioso que el de la tecnología corriente.

+ Ver 20 Gadgets increíbles.jpeg

+ [20+ Gadgets increíblemente geniales que se agotaron en el último mes |](#)

[TrendingBoom](#)

Algunos ejemplos.

Muchos aparatos tecnológicos de tamaño reducido pueden ser considerados gadgets, como es el caso de los móviles, mandos a distancia (Joystick), iPod, PDA, calculadoras, relojes, mouse, lentes 3D, etc.



+ Futuro de los móviles, tendencia actual.

➤ Ver documentos:

- El móvil del Futuro
- Intel teléfono plegable.pdf
- *Video Galaxy Z Fold2 Official Introduction Film Flex mode Samsung*

➤ Las 5 tendencias tecnológicas que marcarán 2020

<https://hipertextual.com/2020/01/tendencias-tecnologicas-2020-tesla-telefonos-plegables-5g-vehiculos-electricos>

Cinco tendencias tecnológicas 2020.pdf

Clase del 3 de Febrero 2023

+ Plataformas de desarrollo.

Nos interesa la programación. Cual o que plataformas existen para programar. Existen muchos tipos, características, modelos de dispositivos móviles. Han aparecido muchas plataformas de desarrollo => lenguajes hasta Sistemas operativos.

- Que Sistemas Operativos para los dispositivos móviles existen, que ventajas tienen
 - Android es uno de los más populares y ampliamente usado
- Ver artículo:
 - Sistemas Operativos Dispositivos Móviles.doc

Clase del 8 de Febrero 2023

Programación.

Plataforma Android: Sistema Operativo – Framework => plataforma de desarrollo de aplicaciones =>

IDE => ambiente de programación para programar aplicaciones, compilar y ejecutar.

Características generales de Android.

El Sistema Operativo Android.

Origen

La historia de Android empieza en octubre de 2003, cuando Andy Rubin, Rich Miner, Nick Sears y Chris White crearon la empresa Android en Palo Alto (California).

Google compró la empresa en agosto de 2005. Dos años más tarde, se anunció la Open Handset Alliance y Android pasa a ser oficialmente open source.

La primera versión del SDK Android 1.0 aparece en 2008 con el primer teléfono con Android (HTC Dream).

En abril de 2009, la versión 1.5 (API 3) de Android vio la luz. Esta versión, bautizada como **Cupcake** (Pastelito), inaugura los nuevos nombres de versión de Android.

Las siguientes versiones siguieron este patrón:

- **Donut** (Donut) 1.6 (API 4): septiembre de 2009
- **Eclair** (Rayo) 2.0/2.1 (API 7): octubre de 2009
- **Froyo** (Yogur helado) 2.2.x (API 8): mayo de 2010
- **Gingerbread** (Pan de jengibre) 2.3.x (API 10): diciembre de 2010
- **Honeycomb** (Panal) 3.x (API 11 - 12 - 13): febrero de 2011
- **Ice Cream Sandwich** (Sandwich de helado) 4.0.x (API 14 - 15): octubre de 2011. Esta versión unifica la versión para tabletas (Honeycomb) y la versión para smartphones (Gingerbread).
- **Jelly Bean** (Gominola) 4.1.x hasta 4.3.1 (API 16, 17 y 18): junio de 2012.
- **KitKat** 4.4.x (API 19): octubre 2013.
- **Lollipop** (Piruleta) 5.x (API 21): octubre 2014. :

El icono de Android se llama **Bugdroid**. Este pequeño robot verde tiene como origen, según rumores, el personaje de un videojuegos de los años 90 de Atari: **Gauntlet: The Third Encounter**.



Version ↕	Code name ↕	Release date ↕	API level ↕	Runtime ↕	First devices to run version ↕
10	10	September 3, 2019	29	ART	Pixel, Pixel XL, Pixel 2, Pixel 2 XL, Pixel 3, Pixel 3 XL, Pixel 3a, Pixel 3a XL ^[377]
9	Pie	August 6, 2018	28	ART	Essential Phone, Pixel, Pixel XL, Pixel 2, Pixel 2 XL, Nokia 7 Plus, OnePlus 6, Oppo R15 Pro, Sony Xperia XZ2, Vivo X21UD, Vivo X21, Xiaomi Mi Mix 2S ^[378]
8.1	Oreo	December 5, 2017	27	ART	Pixel, Pixel XL, Nexus 6P, Nexus 5X
8.0		August 21, 2017	26	ART	N/A
7.1	Nougat	October 4, 2016	25	ART	Pixel, Pixel XL
7.0		August 22, 2016	24	ART	Nexus 5X, Nexus 6P
6.0	Marshmallow	October 5, 2015	23	ART	
5.1	Lollipop	March 9, 2015	22	ART	Android One
5.0		November 3, 2014	21	ART 2.1.0	Nexus 6, Nexus 9
4.4	KitKat	October 31, 2013	19	Dalvik (and ART 1.6.0)	Nexus 5
4.3	Jelly Bean	July 24, 2013	18	Dalvik	Nexus 7 2013
4.2		November 13, 2012	17	Dalvik	Nexus 4, Nexus 10
4.1		July 9, 2012	16	Dalvik	Nexus 7
4.0	Ice Cream Sandwich	October 19, 2011	15	Dalvik	Galaxy Nexus
2.3	Gingerbread	February 9, 2011	10	Dalvik 1.4.0	Nexus S
Legend: Old version Older version, still supported Latest version Latest preview version Future release					

La versión actual es la 13.

Características.

Android es un OS (*Operating System* o sistema operativo) open source pensado para teléfonos móviles y desarrollado por la **Open Handset Alliance** (OHA) bajo autorización de Google. La OHA se compone de alrededor de 80 empresas, tales como Samsung, HTC, SFR, Orange, Asus, Qualcomm...

Android se basa en un kernel Linux y se distribuye bajo una licencia **Apache License 2.0**. Separa la capa hardware de la capa lógica, es decir, cualquier teléfono Android puede ejecutar la misma aplicación y, de este modo, se puede crear un amplio abanico de posibilidades para los fabricantes, los usuarios y los desarrolladores.

Android es también un framework y, como desarrollador, tendrá acceso al SDK (*Software Development Kit* - kit de desarrollo) y a todo el código fuente de la plataforma, lo que permitirá comprender su funcionamiento, crear versiones personalizadas del OS y, por supuesto, desarrollar sus propias aplicaciones.

Esto ha hecho que varios dispositivos exóticos hayan visto la luz con Android: radios de coche, relojes, cascos de esquí, etc.

Ver documento: Android cumple 10 años.doc

El entorno de desarrollo Android Studio

Ver el documento: Instalación de Android Studio.doc

- Directorio de instalación de Android Studio:
- C:\Program Files\Android\Android Studio

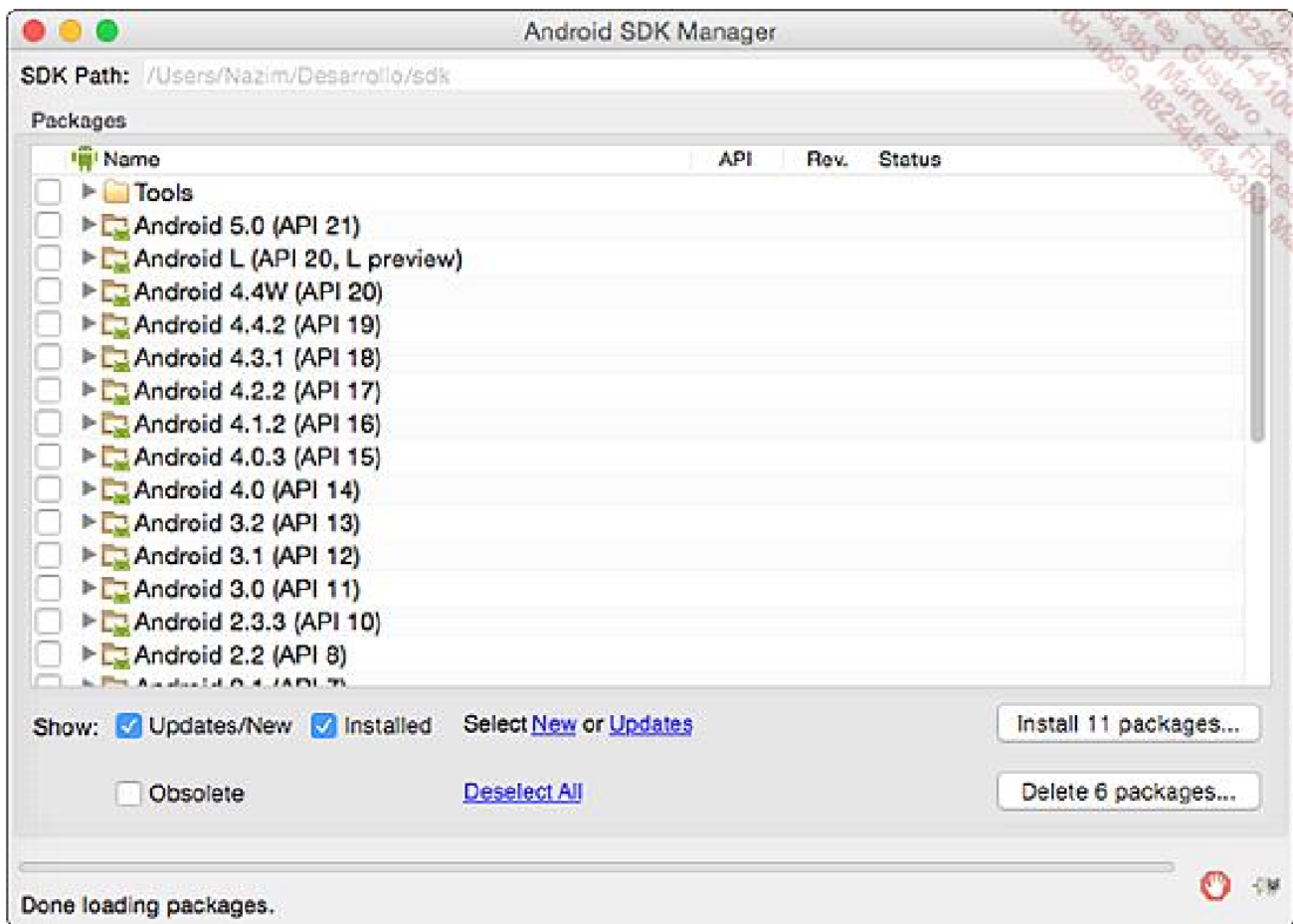
La siguiente etapa permite configurar el SDK Android. Este SDK contiene todas las herramientas necesarias para crear una aplicación Android. Está disponible para Windows, Mac OS y Linux.

Cada versión del SDK contiene:

- **aapt - Android Asset Packaging Tool:** esta herramienta sirve para crear y analizar archivos *.apk (Principios de programación - Arquitectura Android). Estos archivos contienen el programa de la aplicación.
- **adb - Android Debug Bridge:** el objetivo de esta herramienta es establecer conexiones con un teléfono Android o un emulador facilitando el acceso a la terminal. Todo ello para listar su contenido o ejecutar comandos. Esta herramienta también sirve para transferir una aplicación o archivos a un teléfono Android o a un emulador.
- **ddms - Dalvik Debug Monitor Service** (Depuración y gestión de errores - Principios): esta herramienta se utiliza para la depuración de aplicaciones y permite:
 - Realizar captura de pantallas.
 - Ver los threads en ejecución.
 - Ver el Logcat.
 - Averiguar la lista de procesos en ejecución en el dispositivo.
 - Simular el envío de mensajes y llamadas.
 - Simular una localización, etc.
- El SDK también contiene un sistema que permite crear y gestionar emuladores así como la documentación para cada versión de Android y ejemplos para cada una de las API.

Configuración Para la versión 2019 esta configuración ya no es necesaria, sólo para aspectos específicos

Ejecute el administrador de SDK Android (con ayuda del acceso directo especificado en Android Studio). Debería abrirse la ventana siguiente:



Esta herramienta, **Android SDK Manager**, permitirá actualizar su SDK, instalar nuevas versiones de Android o actualizar las versiones ya instaladas.

Instalaremos, como mínimo, los siguientes elementos (útiles para los ejemplos utilizados).

- En la sección **Tools**:
 - **SDK Tools** (instalado por defecto): contiene los elementos necesarios para desarrollar, probar y depurar su aplicación. Se instala por defecto con el SDK Android y se actualizará cuando aparezcan nuevas versiones.
 - **SDK Platform-tools**: contiene herramientas de desarrollo, de pruebas y de depuración para su aplicación. Estas herramientas se adaptan a la versión de Android utilizada por su aplicación.
 - **Android SDK Build Tools** (instalada a partir de la versión 19 hasta la última versión): contiene herramientas que le permiten construir y configurar sus aplicaciones Android.
- En la sección **Android 4.0, 4.1.2, 4.4.2 y 5.0.1**:
 - **Documentation Android**: la documentación de la versión de Android seleccionada.
 - **SDK Platform**: la versión del SDK.
 - **Sample**: ejemplos de las API.
 - **Google APIs**: la versión del SDK Android incluyendo las APIs de Google.
 - **Source**: los fuentes de esta versión de Android.

- En la sección **Extras**, únicamente:

- **Android Support Library**: anteriormente llamado "compatibility package" (paquete de compatibilidad), representa una biblioteca que se incluirá en el proyecto Android para utilizar funcionalidades no incluidas en el framework Android (tales como el **ViewPager**) o las funcionalidades no disponibles en todas las versiones de Android (tales como los **Fragments**).
- **Google Repository**: almacén Maven local para las bibliotecas de compatibilidad Android.
- **Google USB Driver Package**: necesario si se dispone a desarrollar en Windows con un teléfono Google (Galaxy Nexus, Nexus 4/5 o 6, etc). Contiene los drivers para desarrollar, probar y depurar en estos teléfonos.
- **Google Play Services**: útil para varios servicios de Google tales como Google Maps y la localización del usuario.

→ A continuación, clic en el botón **Install Packages** (Instalar paquetes) para finalizar la instalación.

Contenido del SDK

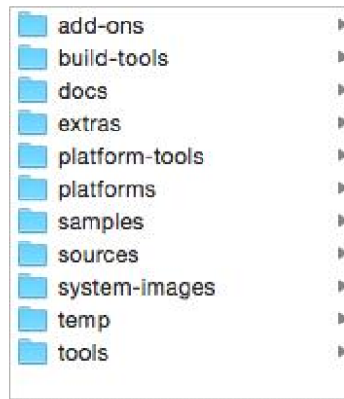
Una vez se ha instalado el SDK, explorar su estructura de directorios:

Para la versión 2019 ver el directorio:

C:\Users\GustavoMarquez\AppData\Local\Android\Sdk

Directorio del SDK versión 2021:

C:\Users\gmarq\AppData\Local\Android\Sdk (HP)



- La carpeta **add-ons** contiene las distintas extensiones o SDK que ha descargado fuera del SDK principal (Google APIs, por ejemplo).
- La carpeta **build-tools** contiene diferentes herramientas que sirven para crear y exportar una aplicación.
- La carpeta **docs** contiene la documentación de Android.
- La carpeta **extras** contiene las herramientas que ha descargado (sección Extras en el SDK Manager).
- La carpeta **platform-tools** contiene herramientas que le permiten desarrollar y depurar sus aplicaciones.
- La carpeta **platforms** contiene las distintas versiones del SDK que se ha descargado, organizadas por número de versión.
- La carpeta **samples** contiene distintos ejemplos de las APIs de Android.
- La carpeta **sources** contiene el código fuente de Android en función de las versiones del SDK.
- La carpeta **tools** contiene herramientas que facilitan el uso del SDK Android.
- **AVDManager** es el gestor de emuladores (únicamente presente en Windows).
- **SDKManager** es el gestor del SDK (únicamente presente en Windows).
- La carpeta **samples** contiene distintos ejemplos de las APIs de Android.
- La carpeta **sources** contiene el código fuente de Android en función de las versiones del SDK.
- La carpeta **tools** contiene herramientas que facilitan el uso del SDK Android.
- **AVDManager** es el gestor de emuladores (únicamente presente en Windows).
- **SDKManager** es el gestor del SDK (únicamente presente en Windows).

Crear un emulador en Android

⇒ Simular un dispositivo móvil en la laptop para ver los programas correr

Aplicación Hola a Todos.

Ver los siguientes documentos:

- Crear HolaTodos Android.doc => Proyecto \AndroidStudioProjects\ProyectosEscuela\HolaTodos
- Aplicación HolaTodosAndroid.doc => Proyecto \AndroidStudioProjects\ProyectosEscuela\HolaTodosAndroid

Directorio de trabajo:

C:\Users\GustavoMarquez\AndroidStudioProjects (ubicación por default de Android)

Ver documento *Análisis de la aplicación Hola a Todos en Andorid.doc*

Documento impreso con anotaciones

Tarea 1: Hacer un dibujo de caracteres en Android. No imágenes ni digital

%#D |

| //

Mostrarlo en el layout, en vez de un hola a todos.

Objetivo es probar el ambiente. Entregar la carpeta completa del proyecto en drive

Enviar la tarea al ayudante Jesús Ivan Saavedra con copia para mi. Es individual.

Fecha de entrega: ¿?

Ver con Jesus la entrega si no tienen Drive y crear un drive o sitio donde subirlas

Clase del 13 de Febrero 2023

Programación con Android Studio.

Arquitectura Android

La elección de la arquitectura Android se basa en la idea de controlar los recursos y el consumo. Las aplicaciones Android se ejecutan en un sistema con restricciones (memoria disponible, consumo de batería, diferencias en la visualización, almacenamiento disponible...).

Como desarrollador, . prestar especial atención a los siguientes puntos:

- La creación de nuevos objetos.
- El uso de recursos (procesador, RAM, almacenamiento, etc.).
- El consumo de la batería.
- La diversidad de tamaños y resoluciones de pantalla y de configuraciones de hardware.
- La diversidad de versiones de Android disponibles en el mercado.

La arquitectura Android se compone de cinco partes diferenciadas:

- **Aplicación:** representa el conjunto de aplicaciones proporcionadas con Android.
- **Framework Android:** representa el framework que permite a los desarrolladores crear aplicaciones accediendo al conjunto de API y funcionalidades disponibles en el teléfono (fuentes de contenido, gestor de recursos, gestor de notificaciones, gestor de actividades, etc.).
- **Librerías:** Android dispone de un conjunto de librerías que utilizan los distintos componentes del sistema.
- **Android Runtime:** contiene, entre otros, la máquina virtual ART.
- **Linux Kernel:** el núcleo Linux (2.6) que proporciona una interfaz con el hardware y que gestiona la memoria, los recursos y los procesos Android.

ART (Android RunTime)

Android se basa en una nueva máquina virtual (desde la versión 5.0 Lollipop) particular, denominada **ART** (*Android RunTime*). Remplaza a Dalvik (disponible en las versiones anteriores de Android) y posee varias características particulares.

Una máquina virtual es un programa que tiene la propiedad de simular (emular) las características de ejecución de un procesador.

Por ejemplo, la **máquina virtual de Java** (**Java Virtual Machine, JVM**) es un programa que interpreta y ejecuta instrucciones expresadas en un código binario especial (el bytecode Java, class), el cual es generado por el compilador del lenguaje **Java** y que es ejecutado en un procesador de una computadora específica (INTEL, AMD, Motorola 6000, etc.)

En particular en Java esta máquina virtual se llama **JRE** (Java Runtime Environment). Su función es de intermediario entre una aplicación programada en **Java** y el sistema operativo donde se esté ejecutando el programa. De este modo, cualquier aplicación puede funcionar en cualquier sistema operativo que disponga del **JRE**.

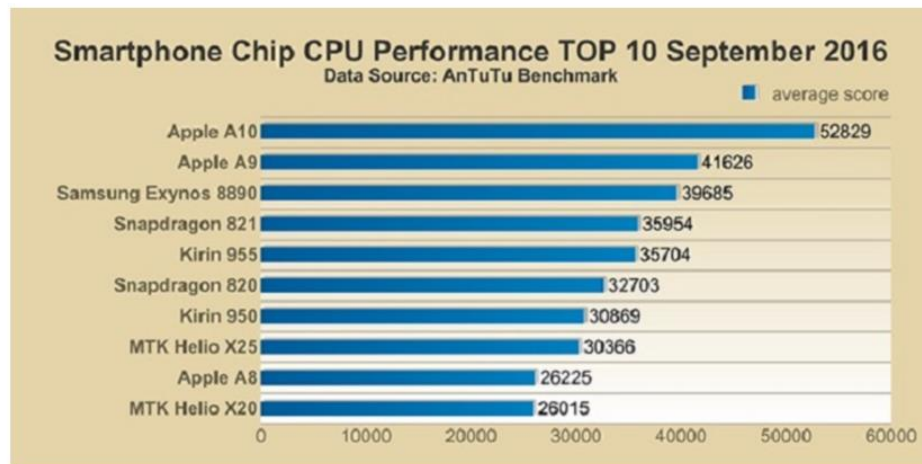
Entre las características principales que tiene la máquina virtual ART, están las siguientes:

- AOT Compilation (*Ahead of Time* - compilación anticipada) que permite compilar una aplicación en tiempo de instalación y no en tiempo de ejecución (como hacía Dalvik). Esto permite mejorar enormemente el rendimiento de las aplicaciones, puesto que no se compilan con cada ejecución.
- Mejora del Garbage Collector (recolector de basura).
- Mejora en el desarrollo y depuración de aplicaciones (los fallos ofrecen mensajes más detallados).

Dalvik, la máquina virtual utilizada en las versiones anteriores de Android, se basa en la compilación en tiempo de ejecución (*Just In Time*), lo que significa que su aplicación se compilará con cada ejecución. El paso a ART aumenta el rendimiento, pero también el espacio necesario para almacenar una aplicación (el almacenamiento de la versión compilada de la aplicación).

La gran mayoría de los teléfonos Android tienen alguna variante de los **procesadores Snapdragon de Qualcomm**. El 810 de ocho núcleos (dos **procesadores** de 4 núcleos cada uno) y 808 son los más populares hoy en día y lo utilizan los mejores teléfonos que hay en el mercado actualmente.

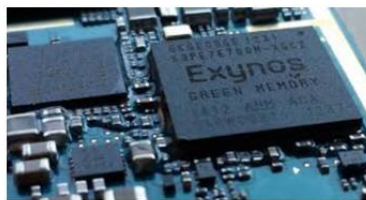
Los procesadores para celulares más potentes en cuanto a rendimiento.



18

Ver los siguientes documentos:

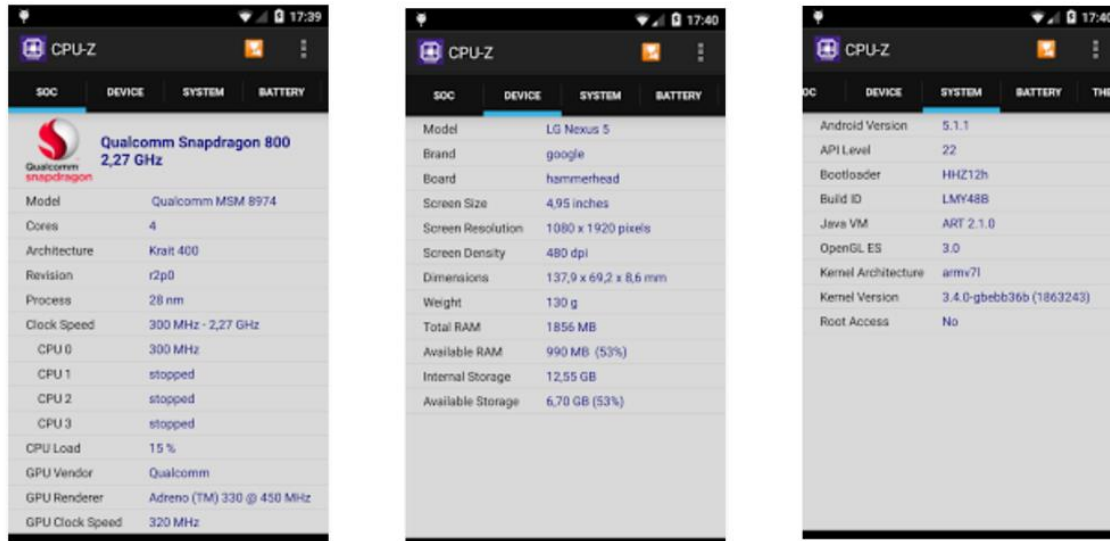
- Los 6 mejores procesadores para celulares.pdf
- 10 procesadores de móviles Android más usados en 2017.pdf



Algunos procesadores de dispositivos móviles

20

Para conocer que procesador y máquina virtual tiene un celular, instalar la aplicación CPU Info o CPU-Z, que son gratuitas de Google-Play.



21

NDK (Native Development Kit)

Es posible desarrollar de forma nativa en Android utilizando el NDK (*Native Development Kit* - kit de desarrollo nativo), que se basa en C/C++.

También puede utilizar un mecanismo Java que . permitirá llamar a código nativo desde una clase JNI (*Java Native Interface*).

➤ El NDK está disponible en la siguiente dirección: <https://developer.android.com/tools/sdk/ndk/index.html>

El NDK . da acceso a:

- Un conjunto de herramientas que permiten generar código nativo desde archivos fuente C/C++.
- Una herramienta de creación de archivos apk (. APK (Android Package)) a partir de código nativo.
- La documentación, con ejemplos y tutoriales.

APK (Android Package)

Un APK es el archivo binario que representa una aplicación. Este formato se utiliza para distribuir e instalar aplicaciones.

Para crear un APK, se debe compilar y empaquetar en un archivo una aplicación Android. Este archivo contendrá:

- El código de la aplicación compilada (.dex).
- Los recursos.
- Los assets. Recursos extra que no son de Android, por ejemplo imágenes, archivos multimedia, etc.
- Los certificados.
- El archivo de manifiesto.

Para poder publicar una aplicación en **Google Play**, necesita crear un archivo APK.

Para generar un APK desde **Android Studio**, Android proporciona una herramienta que se llama **aapt** (*Android Asset Packaging Tool* - herramienta que permite crear, visualizar y modificar los archivos Android-APK), incluida en el SDK Android e integrada en el entorno de desarrollo.

23

Puede generar dos tipos de APK:

- Un APK firmado.
- Un APK sin firmar. Este formato puede usarse para probar una aplicación pero no puede publicarse en Google Play.

1. Creación de un keystore

Para generar un archivo APK firmado, necesitará un **keystore** (almacén de claves), que sirve para albergar las claves utilizadas para firmar aplicaciones. Puede usar un keystore que ya usaba previamente, o bien crear uno nuevo.

Para crear un keystore hay que:

- Elegir una ubicación para guardar el keystore.
- Indicar una contraseña para el uso del keystore.

Un keystore se compone de varias claves de firma, cada clave puede servir, por ejemplo, para firmar una aplicación distinta.

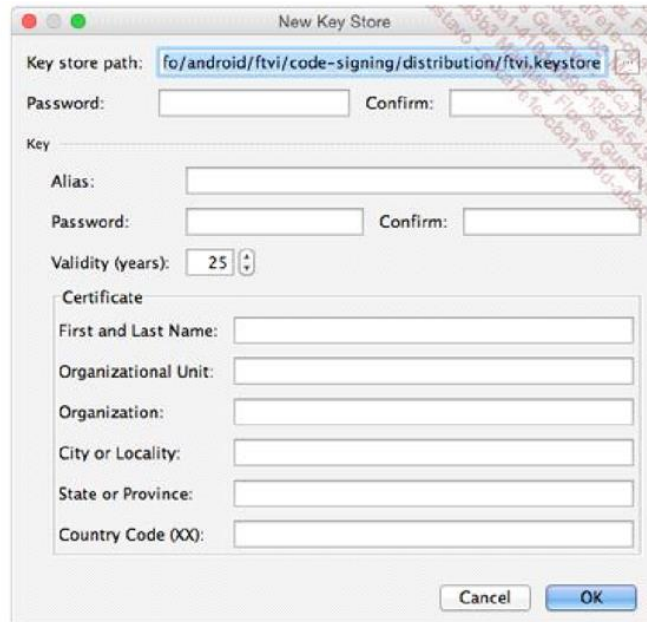
24

Durante la firma de una aplicación, puede elegir entre utilizar un keystore existente o bien crear uno nuevo.



→ En la ventana anterior, clic en la opción **Create new** (Crear uno nuevo) para crear un nuevo keystore.

25



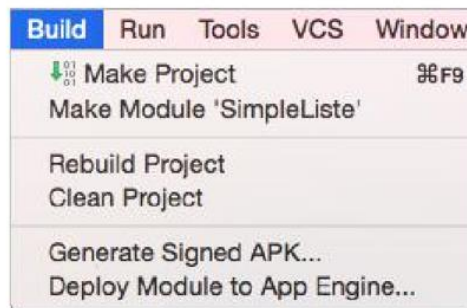
Esta pantalla le permite introducir los datos siguientes sobre clave:

26

- **Key store path:** ubicación del keystore.
- **Password:** contraseña del keystore y su confirmación.
- **Alias:** identificador de la clave que servirá para firmar la aplicación.
- **Password:** contraseña de la clave de firma y su confirmación.
- **Validity (years):** duración de validez, en años.
- Introducir información de al menos uno de los campos adicionales (nombre y apellidos, población, país, etc.).

2. Creación de APK firmado

→ Para generar un APK firmado en Android Studio, clic en el menú **Build** (Construir) y, a continuación, seleccione la opción **Generate Signed APK...** (Generar APK firmado).



- Seleccione la aplicación correspondiente para generar el APK y, a continuación, clic en **Next** (Siguiente).
- Seleccione, a continuación, el keystore creado en la sección anterior mediante el botón **Choose existing...** (seleccionar un keystore existente).



→ Para terminar, seleccione la ubicación del APK generado y clic en **Finish** (Finalizar).

- Debe conservar clave de firma cuando publique su aplicación en Google Play. Si la pierde, no podrá actualizar aplicación nunca más.

También puede crear una clave de firma por línea de comandos mediante el ejecutable **keytool**.

```
[arnau@tulkass ~]$ keytool -genkey -keystore claves -alias 'clave firma android' -keyalg RSA -validity 36500
Introduzca la contraseña del almacén de claves:
Volver a escribir la contraseña nueva:
¿Cuáles son su nombre y su apellido?
[Unknown]: Arnau
¿Cuál es el nombre de su unidad de organización?
[Unknown]:
¿Cuál es el nombre de su organización?
[Unknown]:
¿Cuál es el nombre de su ciudad o localidad?
[Unknown]:
¿Cuál es el nombre de su estado o provincia?
[Unknown]:
¿Cuál es el código de país de dos letras de la unidad?
[Unknown]: ES
¿Es correcto CN=Arnau, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=ES?
[no]: sí

Introduzca la contraseña de clave para <clave firma android>
(INTRO si es la misma contraseña que la del almacén de claves):
Volver a escribir la contraseña nueva:
[arnau@tulkass ~]$
```

 En Windows, el ejecutable keytool se encuentra en la carpeta bin de la carpeta de instalación de Java.³⁰

También puede firmar una aplicación a posteriori mediante el ejecutable **jarsigner**.

```
jarsigner -verbose -verify -certs miAplicacion.apk
```

Ver los siguientes documentos:

- Cómo exportar un APK en Android Studio.doc
- Formato APK.doc