



# Iniciación a la programación móvil

Curso: Desarrollando aplicaciones con Android

## Algunos datos (I)

---

- ▶ Las ventas mundiales de teléfonos inteligentes en 2013 rebasaron por primera vez el listón de los 1.000 millones de unidades, según IDC.
  - ▶ Concretamente, se distribuyeron 1.004 millones, un 38,4% más respecto a 2012, año en que se vendieron 725 millones de unidades.
- ▶ La cifra contrasta con la venta de PC, que en 2013 alcanzaron los 315 millones de unidades.
- ▶ La venta de móviles básicos cae estrepitosamente
  - ▶ Las ventas de smartphones coparon el 55,1% del total del mercado de móviles, diez puntos más frente al año anterior, cuando la cifra era del 41,7%. Entre los teléfonos básicos y los inteligentes, las ventas totales de móviles en el mundo superaron los 1.821 millones de unidades, un 4,8% más:



## Algunos datos (II)

---



- ▶ Hoy en día, 6.000 millones de los 7.000 millones de habitantes del planeta tienen acceso a un teléfono móvil. Quienes pueden hacer sus necesidades en un aseo o letrina apenas llegan a los 4.500 millones.



## Algunos datos (y III)

---

- ▶ Según la consultora Analysys Mason, las personas utilizamos el móvil **195 minutos** al día pero solo empleamos el **25%** de ese tiempo para comunicarnos.
- ▶ El resto del tiempo lo empleamos en *apps*.



# Contenido

---

- ▶ Desarrollo de aplicaciones para dispositivos móviles
- ▶ Plataformas móviles
- ▶ ¿Qué es Android?
- ▶ Arquitectura de Android



# Desarrollo de aplicaciones móviles

---

- ▶ A medida que aumentan las capacidades, prestaciones y posibilidades de los dispositivos y terminales móviles, se está produciendo una demanda cada vez mayor de nuevos servicios y aplicaciones.
- ▶ Cuando queremos desarrollar aplicaciones móviles tenemos tres opciones
  - ▶ Aplicaciones web optimizadas para móviles
  - ▶ Aplicaciones móviles nativas
  - ▶ Aplicaciones híbridas



# Desarrollo de aplicaciones móviles

---

## ▶ Desarrollo Móvil web

- ▶ Desarrollo de aplicaciones web optimizadas para ser visualizadas desde un dispositivo móvil o tableta.
  - ▶ Diseño web adaptable (*Responsive web design*)
- ▶ Características de aplicaciones móviles web
  1. Serán accedidas desde un navegador instalado en el teléfono.
  2. Se desplegará casi de la misma forma en todos los dispositivos. Adaptará la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarla.
  3. Es prácticamente igual que desarrollar cualquier aplicación o sitio web: utilizas HTML, CSS y JavaScript.



# Desarrollo de aplicaciones móviles

---

## ► Desarrollo Móvil web

### ► Ventajas

- **Compatibilidad:** Se reutiliza casi en un 100% el mismo código fuente para todos los sistemas operativos orientados a móviles. Así sólo se mantiene una versión.
- **Rango de usuarios:** Se puede llegar a más tipos de dispositivos: no sólo a Android e iOS, también a los menos comunes, como Bada (Samsung).
- **Fácil uso:** No hace falta descargar nada, sólo con acceder a una URL los usuarios podrán utilizar tu aplicación.
- **Siempre actualizado:** Las actualizaciones llegan inmediatamente. No debes esperar una semana para que el AppStore apruebe tu nueva actualización.
- Normalmente, **más sencillo y rápido de desarrollar:** en muchas ocasiones, es mucho más fácil desarrollar y diseñar para un navegador utilizando HTML, CSS y JavaScript que hacer lo mismo en Java, Objective-C.





# Desarrollo de aplicaciones móviles

---

## ► Desarrollo Móvil web

### ► Desventajas

- **Olvídate de utilizar la cámara y otros componentes nativos** (al menos por ahora): En aplicaciones web, al ser accedidas desde el navegador, no tienes disponibilidad sobre ciertos componentes y funcionalidades nativas del teléfono, entre ellos la cámara.
- **Cross-Browser Support versión 2.0:** Antes te preocupaba Internet Explorer 6, ahora te preocupan otros, como el pésimo soporte a los estándares en BlackBerry 4 y 5, lo que trae mayores costos de desarrollo en muchos casos.
- **Posibilidad de usarla offline** (en algunos casos): Tienes una aplicación en mente que necesitas que se pueda utilizar en cualquier momento. Olvídate del “offline mode” cuando estés desarrollando para la web. Esto no se aplica para iOS y otros, ya que Safari implementa Web Storage de HTML5 y Offline Application Cache.



# Desarrollo de aplicaciones móviles

---

## ► Desarrollo Móvil Nativo

- Desarrollo de aplicaciones que serán instaladas en el sistema de archivos de cada dispositivo y serán distribuidas por los mercados de aplicaciones móviles, como el AppStore (iOS) o el Play Market (Android).
- Características de las aplicaciones móviles nativas
  1. Serán descargadas desde el mercado de aplicaciones para el sistema operativo: Google Play o el AppStore.
  2. Es una aplicación distinta para cada sistema operativo: una para Android y otra para iOS (y en muchos casos una para iPhone y otra para iPad).
  3. Se desarrollan con lenguajes como Java (Android) u Objective-C (iOS), pero hay alternativas como Appcelerator que permiten desarrollarlas utilizando JavaScript puro, para ambas plataformas.



# Desarrollo de aplicaciones móviles

---

## ► Desarrollo Móvil Nativo

### ► Ventajas

- **Posibilidad de utilizar funcionalidades nativas:** siempre puedes utilizar funcionalidades nativas, como la cámara, el acelerómetro, etc.
- **Nuevas alternativas para desarrollar multiplataforma:** Aunque lo normal es desarrollar estas aplicaciones utilizando las herramientas oficiales (como el Android SDK y Java en Android), han surgido una serie de nuevas tecnologías para llegar a varios dispositivos con casi el mismo código fuente, como Appcelerator.
- **Rendimiento:** Normalmente son más eficientes que las aplicaciones web.
- **Hay un “market” para estas aplicaciones:** Las personas pueden encontrar tu aplicación utilizando el AppStore o el Google Play. En cualquiera de los casos, estas aplicaciones pueden ser puestas a la venta.
- **Modo offline:** Tu aplicación puede utilizarse sin necesidad de una conexión a internet.



# Desarrollo de aplicaciones móviles

---

## ► Desarrollo Móvil Nativo

### ► Desventajas

- **Normalmente son más costosas de desarrollar:** Por lo general, son más costosas de desarrollar, ya que se necesita invertir mucho más tiempo.
- **Mayores costes de mantenimiento:** Mantener varias versiones de la aplicación es costoso.
- **Actualizaciones y aprobaciones:** Poner una aplicación en el AppStore es tedioso, puede llevar hasta más de una semana. De igual forma, si tienes una nueva funcionalidad o descubriste un bug que quieres arreglar, debes esperar otra semana para que los chicos de Apple aprueben tus cambios.
- **Aún cuando hay opciones como Appcelerator, no llegas a todos:** Appcelerator, por ejemplo, permite desarrollar para Android, iPhone y iPad. ¿Qué pasa con tus usuarios de Blackberry, Symbian o Bada (si es que hay alguno)? Seguramente tendrás que hacer una versión alternativa web para ellos.



# Desarrollo de aplicaciones móviles

---

## ► Aplicaciones Híbridas

- La idea es fácil: utilizar el desarrollo nativo cuando es mejor o es necesario (para utilizar la cámara, por ejemplo), pero utilizar tecnologías web y el desarrollo web cuando es más práctico (por ejemplo en interfaces más complejas).
- Algunas de las herramientas más utilizadas hoy en día para crear aplicaciones híbridas son:
  - [Phonegap](#): Quien permite realizar aplicaciones con HTML5, CSS3 y JavaScript, pero que serán empaquetadas como aplicaciones nativas. Seguramente si vas a utilizar Phonegap necesitarás oír [Sencha Touch](#) o [jQuery Mobile](#) para la interfaz.
  - [Titanium Appcelerator](#): Utilizando [Web Views](#) para embeber un navegador web dentro de alguna ventana nativa.



# Plataformas móviles

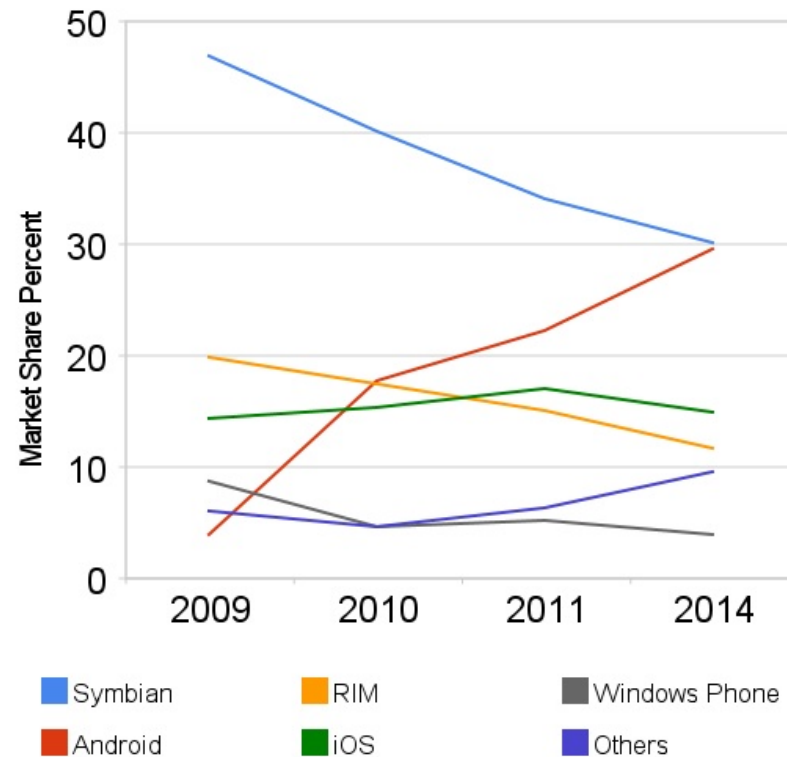


	Apple iOS 7	Android 4.3	Windows Phone 8	BlackBerry OS 7	Symbian 9.5
Compañía	Apple	Open Handset Alliance	Microsoft	RIM	Symbian Foundation
Núcleo del SO	Mac OS X	Linux	Windows NT	Mobile OS	Mobile OS
Licencia de software	Propietaria	Software libre y abierto	Propietaria	Propietaria	Software libre
Año de lanzamiento	2007	2008	2010	2003	1997
Fabricante único	Sí	No	No	Sí	No
Variedad de dispositivos	modelo único	muy alta	media	baja	muy alta
Soporte memoria externa	No	Sí	Sí	Sí	Sí
Motor del navegador web	WebKit	WebKit	Pocket Internet Explorer	WebKit	WebKit
Soporte Flash	No	Sí	No	Si	Sí
HTML5	Sí	Sí	Sí	Sí	No
Tienda de aplicaciones	App Store	Google Play	Windows Marketplace	BlackBerry App World	Ovi Store
Número de aplicaciones	825.000	850.000	160.000	100.000	70.000
Coste publicar	\$99 / año	\$25 una vez	\$99 / año	sin coste	\$1 una vez
Actualizaciones automáticas del S.O.	Sí	depende del fabricante	depende del fabricante	Sí	Sí
Familia CPU soportada	ARM	ARM, MIPS, Power, x86	ARM	ARM	ARM
Máquina virtual	No	Dalvik	.net	Java	No
Aplicaciones nativas	Siempre	Sí	Sí	No	Siempre
Lenguaje de programación	Objective-C, C++	Java, C++	C#, muchos	Java	C++
Plataforma de desarrollo	Mac	Windows, Mac, Linux	Windows	Windows, Mac	Windows, Mac, Linux

# Plataformas móviles

---

Mobile OS Market Share (Source: Gartner)



Porcentaje de teléfonos inteligentes vendidos según su sistema operativos (Fuente: Gartner Group).

---



# Plataformas móviles

---

## ► Cuota de plataformas móviles en España:

Spain	3 m/e July 2013	3 m/e July 2014	% pt. Change
Android	90.4	87.5	-2.9
iOS	5.8	6.2	0.4
Windows	1.6	4.8	3.2
Other	2.2	1.5	-0.7

Fuente: Kantar World Panel





# ¿Qué es Android? - Historia

---

## ▶ Julio 2005

- ▶ Google adquiere Android, Inc.
  - ▶ Pequeña empresa que desarrolla software para móviles (hasta entonces una gran desconocida)

## ▶ Noviembre 2007

- ▶ Nace la Open Handset Alliance
  - ▶ Consorcio de empresas unidas con el objetivo de desarrollar estándares abiertos para móviles
    - Texas Instruments, Broadcom co., Google, HTC, Intel, LG, Marvel Tech., Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, T-Mobile
  - ▶ Se anuncia su primer producto, Android, plataforma para móviles construida sobre el kernel de Linux 2.6



# ¿Qué es Android? - Historia

---

## ▶ Octubre 2008

- ▶ Publicado el proyecto open source Android
  - ▶ Distribuido principalmente con licencia Apache 2.0
    - Partes en otras licencias, p.e. GPL v2 para el núcleo
- ▶ Se abre el Android Market
- ▶ HTC Dream (G1), primer teléfono con Android

## ▶ Diciembre 2008

- ▶ Nuevas incorporaciones
  - ▶ ARM Holdings, Athreos Communications, Asustek Computer Inc., Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp y Vodafone Group Plc.



# ¿Qué es Android? - Historia

---

## ▶ Noviembre 2009

### ▶ Motorola Droid

- ▶ Consigue vender 1.05 millones de unidades en 74 días, superando el record establecido por el iPhone de Apple



## ▶ Diciembre 2009

- ▶ 6.000 aplicaciones en el Market
  - ▶ 60% gratuitas, 30% de pago aprox.

## ▶ Enero 2010

### ▶ Google Nexus One (HTC)

- ▶ Malas cifras de ventas, apenas 135 mil unidades en 74 días

## ▶ Febrero 2010

- ▶ Google anuncia la buena marcha de Android
  - ▶ 60.000 teléfonos con Android vendidos al día



- ▶ ... <http://www.droidlife.com/>



# ¿Qué es Android?

---

- ▶ Android es un *entorno software* construido para dispositivos móviles.
- ▶ **No** es una plataforma hardware
- ▶ Android incluye:
  - ▶ Sistema operativo basado en el kernel Linux
  - ▶ Interfaz de usuario enriquecido,
  - ▶ Funcionalidad de telefonía
  - ▶ Aplicaciones de usuario final
  - ▶ Librerías de código
  - ▶ Frameworks/marcos de aplicación
  - ▶ Soporte multimedia
- ▶ Las aplicaciones de usuario para Android se desarrollan en Java



# ¿Qué es Android? - Características

---

- ▶ Framework/Marco de aplicaciones que permite la reutilización y sustitución de componentes.
- ▶ Máquina virtual **Dalvik** optimizada para dispositivos móviles.
- ▶ Navegador web integrado basado en **WebKit**.
- ▶ Mensajería: SMS y MMS
- ▶ Gráficos optimizados basados en **OpenGL ES**.
- ▶ **SQLite**: ligera y potente base de datos relacional.
- ▶ Soporte **multimedia** para MPEG4, H264, MP3, MIDI, Ogg Vorbis, WAV, AAC, AMR, JPG, PNG, GIF, BMP.
- ▶ Conectividad: GSM/EDGE, IDEM, CDMA, EV-DO, Bluetooth, LTE, WiMAX y WiFi



# ¿Qué es Android? - Características

---

- ▶ Cámara, GPS, brújula, sensor de proximidad y acelerómetro.
- ▶ **Multi-touch**: soporte de pantallas multi-touch
- ▶ **Multi-tasking**: soporte de app multitarea
- ▶ **Tethering**: admite el uso compartido de las conexiones a Internet como un punto de acceso.
- ▶ Entorno de desarrollo incluyendo un emulador de dispositivos, herramientas para la depuración, la memoria y perfiles de rendimiento, y un plugin para el IDE de **Eclipse**.

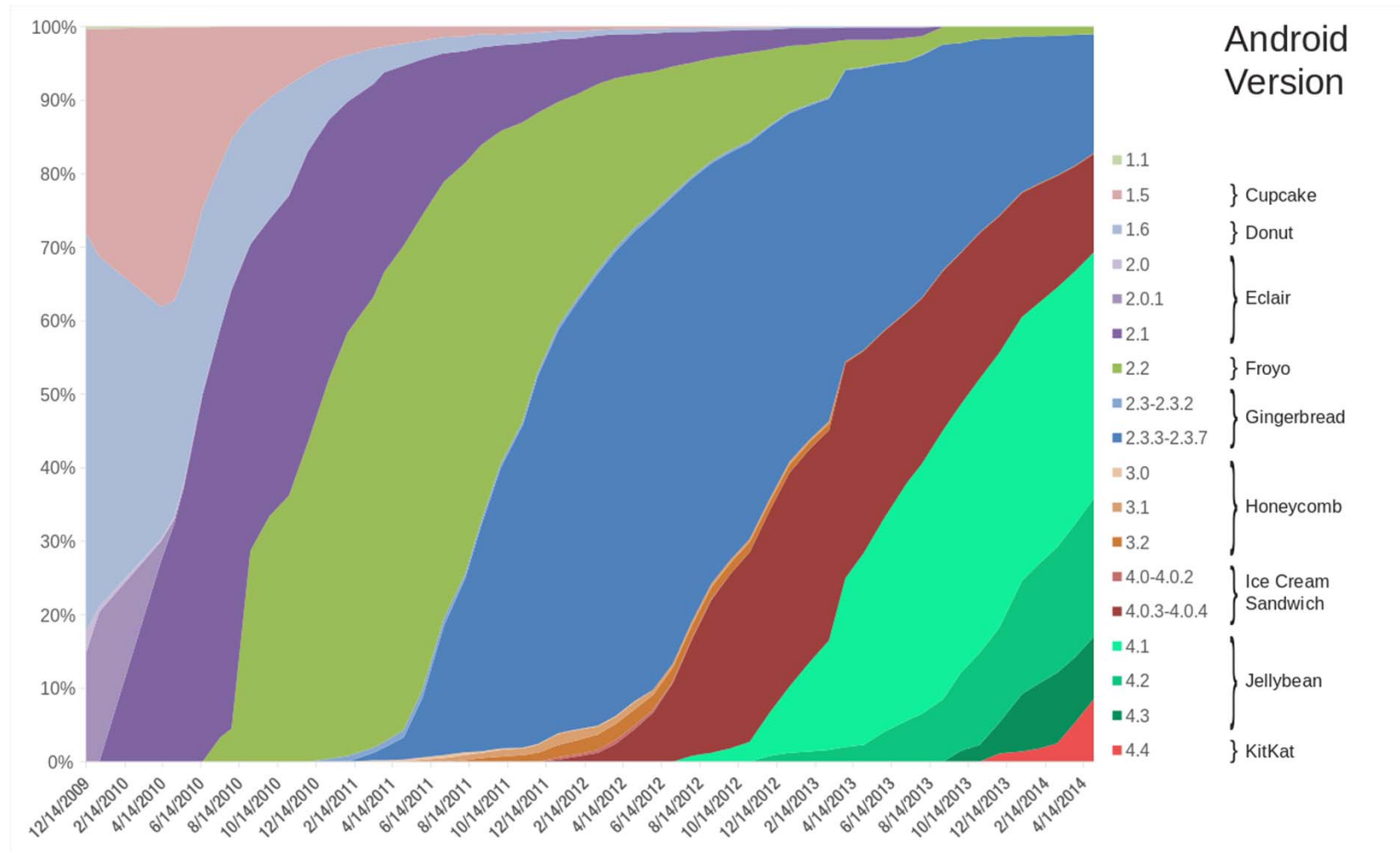


# ¿Qué es Android? - Versiones

---



# ¿Qué es Android? - Versiones



Un gráfico que muestra la distribución mundial de la versión Android de Diciembre de 2009 a Abril de 2014



# ¿Qué es Android? - Versiones

---

## ► Lo nuevo...Android L

- Nuevo Diseño: Material Design  
(<http://www.google.com/design/>)  
(<https://www.youtube.com/watch?v=Q8TXgCzxEnw>)
- Cambio de *runtime*: de Dalvik a ART.
- Compatibilidad con Unreal Technology 4
- Project Volta



# ¿Qué es Android? - Dispositivos

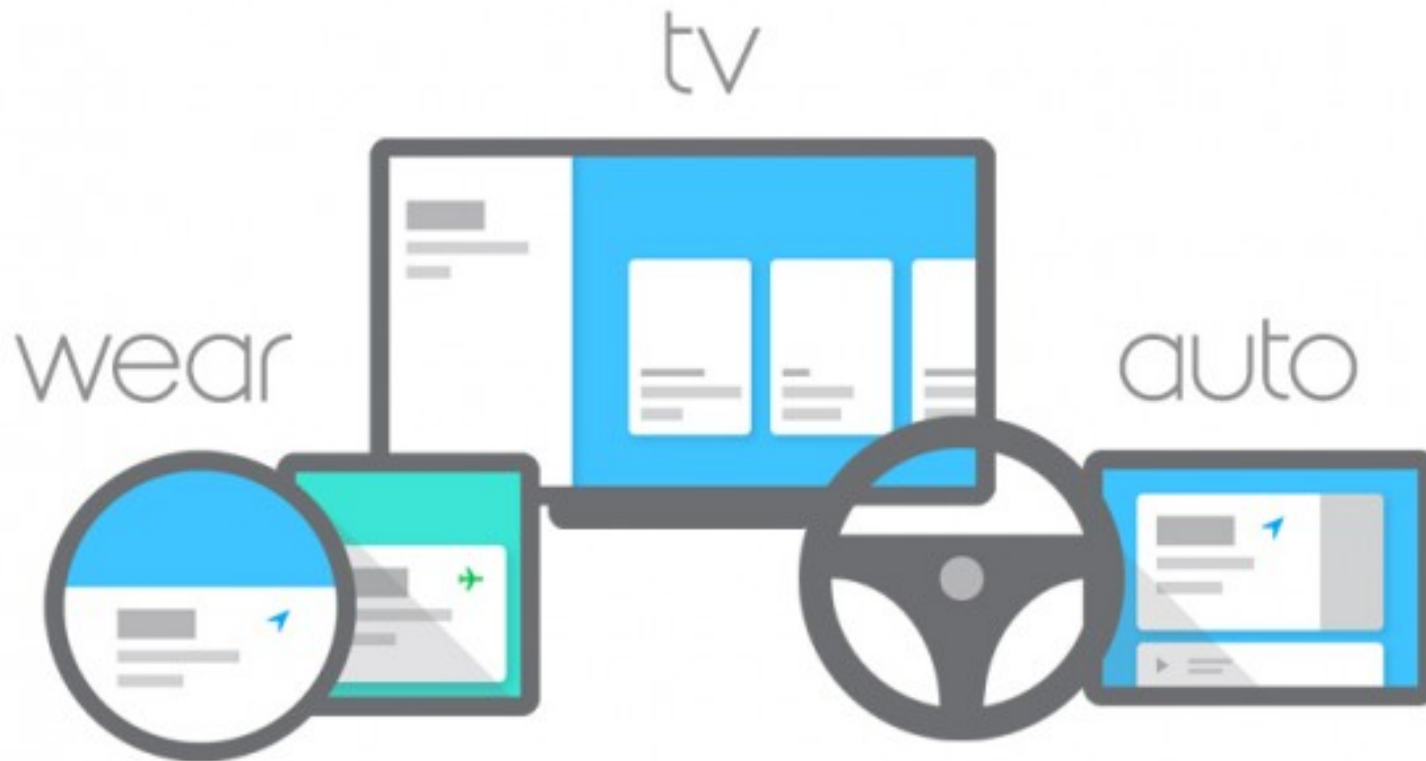
---

- ▶ Existen diversos tipos de dispositivos Android:
  - ▶ Smartphones
  - ▶ Tabletas
  - ▶ Lectores e-books
  - ▶ Netbooks
  - ▶ Reproductores MP4
  - ▶ TVs con internet



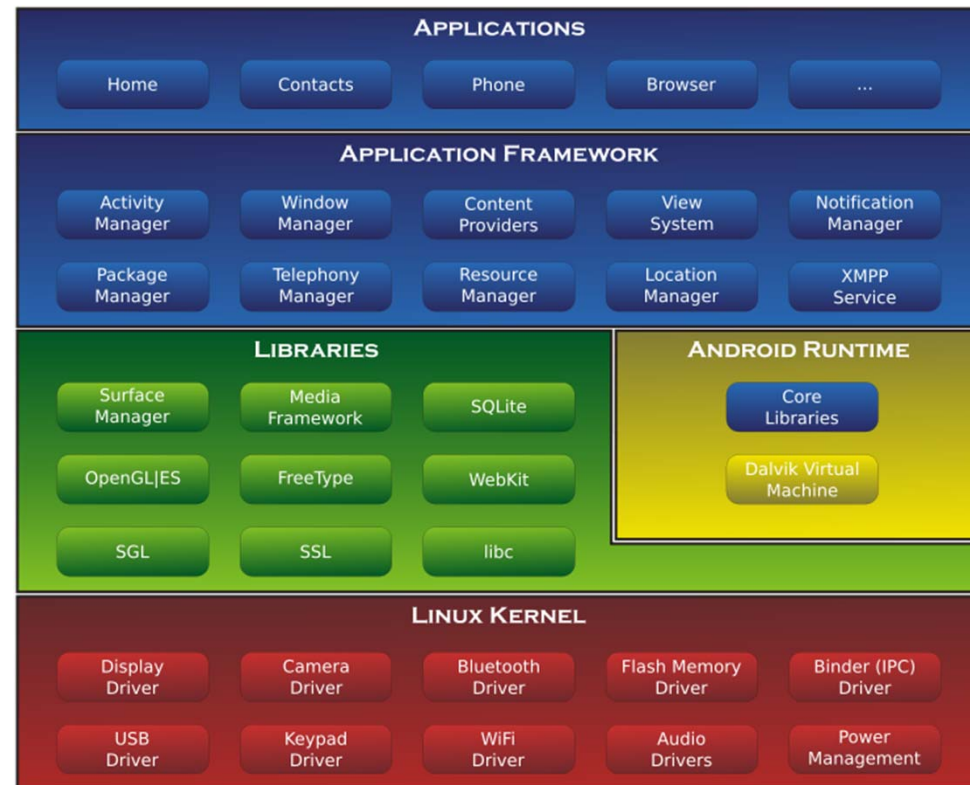
# ¿Qué es Android? - Dispositivos

---



# Arquitectura de Android

- ▶ **Kernel de Linux 2.6** que contiene los controladores de bajo nivel de varios componentes hardware.
- ▶ *Runtime* basado en VM Dalvik
- ▶ **Conjunto de librerías C/C++**
- ▶ **Framework** común a todas las aplicaciones basada en servicios
- ▶ Aplicaciones base implementadas en Java (todas reemplazables)



# Arquitectura de Android - Linux Kernel

---

- ▶ Android se basa en la versión **2.6 de Linux** para servicios básicos tales como:
  - ▶ Seguridad.
  - ▶ Gestión de memoria.
  - ▶ Gestión de procesos.
  - ▶ Pila de red.
  - ▶ Modelo del controlador.
- ▶ El núcleo también actúa como capa de **abstracción** entre el hardware y el resto del software. Por lo tanto, es la única dependiente del hardware.



## Arquitectura de Android - *Android Runtime*

---

- ▶ Utiliza el Kernel de Linux para la ejecución de aplicaciones, procesos y gestión de memoria
- ▶ Formado por un conjunto de librerías base, para ofrecer funcionalidad elemental y de bajo nivel, escritas en C y C++
- ▶ Máquina virtual Dalvik, con código preparado teniendo en cuenta la duración de las baterías y la limitación de memoria.



# Arquitectura de Android - Dalvik Virtual Machine

---

- ▶ Ejecuta aplicaciones en formato .dex (Dalvik Executable)
- ▶ Herramienta dx convierte .class en .dex
- ▶ Usada en entorno con restricciones de memoria y procesador
- ▶ Permite la ejecución de varias instancias a la vez
- ▶ No es una JVM, no ejecuta bytecode
- ▶ Clases de JavaME, AWT o Swing no están soportadas



# Arquitectura de Android – Librerías nativas

---

- ▶ `android.util`: paquete de utilidades del core, clases de bajo nivel
- ▶ `android.os`: acceso a servicios básicos del sistema operativo (mensajes, reloj,...)
- ▶ `android.graphics`: colores, dibujos de primitivas,...
- ▶ `android.text`: mostrar y parsear (transformar) cadenas de texto
- ▶ `android.database`: gestión de las BBDD, cursores, etc.
- ▶ `android.content`: servicios para el acceso a distintos tipos de recursos
- ▶ `android.view`: clases para la gestión de las vistas
- ▶ `android.widget`: artefactos nativos de Android: listas, botones,...
- ▶ `com.google.android.maps`: acceso a controles nativos para mapas (MapView, Overlay,...)





# Arquitectura de Android - Librerías

---

- ▶ `android.app`: acceso al contexto de la aplicación, actividades y API de servicios
- ▶ `android.provider`: acceso a ciertos proveedores de contenido
- ▶ `android.telephony`: acceso a los dispositivos del teléfono para comunicación (call, send SMS,...)
- ▶ `android.location`: servicios basados en localización
- ▶ `android.media`: grabación y reproducción de audio, video, imágenes,...
- ▶ `android.opengl`: motor de renderizado 3D
- ▶ `android.hardware`: acceso a la cámara, acelerómetro, brújula...
- ▶ Otras librerías: FreeType, SGL, libc, SQLite, SSL



# Arquitectura de Android – Entorno de aplicación

---

- ▶ Al ser una plataforma de desarrollo abierto, Android permite crear **aplicaciones muy ricas e innovadoras**.
- ▶ Los desarrolladores tienen acceso a las mismas **APIs** empleadas por las aplicaciones base.
- ▶ La arquitectura de las aplicaciones está diseñada para facilitar la **reutilización de los componentes**
  - ▶ Una aplicación puede publicar sus capacidades y otra puede usarlas.
- ▶ Esta **reutilización** permite reducir en gran medida la **carga** de los **programadores** y también permite **personalizar** una gran cantidad de aspectos del comportamiento de Android



# Arquitectura de Android – Entorno de aplicación

---

- ▶ Tras toda aplicación hay un conjunto de servicios y sistemas:
  - ▶ Un amplio conjunto de vistas (**Views**) que pueden usarse para construir una aplicación.
  - ▶ **Content Provider** que permite a las aplicaciones acceder a los datos de otras aplicaciones.
  - ▶ Un **Resource Manager** que localiza elementos como cadenas, gráficos, ...
  - ▶ Un **Notification Manager** que posibilita a todas las aplicaciones mostrar alertas en la barra de estado.
  - ▶ Un **Activity Manager** que gestiona el ciclo de vida de las aplicaciones.



# Arquitectura de Android - Aplicaciones

---

- ▶ En esta capa se ubican tanto las aplicaciones base como las creadas por el desarrollador
- ▶ Corren en la máquina virtual Dalvik
- ▶ Escritas en lenguaje Java (Android SDK)
- ▶ Aplicaciones base incluidas por defecto:
  - ▶ Un cliente de **e-mail**.
  - ▶ Un gestor de **SMS**.
  - ▶ Un **gestor de información personal** (calendario, una lista de contactos)
  - ▶ Un navegador web basado en **WebKit**.
  - ▶ Un reproductor de **música** y una galería de **imágenes**.
  - ▶ Una cámara y grabación de **vídeo**.
  - ▶ Una **calculadora**.
  - ▶ Una pantalla de **inicio**.
  - ▶ Reloj y alarma.



# Arquitectura de Android - Aplicaciones

---

- ▶ Existe otra opción consistente en desarrollar las aplicaciones utilizando C/C++.
- ▶ Para esta opción podemos utilizar el Android NDK (Native Development Kit).
- ▶ <http://developer.android.com/tools/sdk/ndk/index.html>

