



# Introducción Desarrollo de aplicaciones móviles (ANDROID).

## ▼ Dispositivos Móviles

La telefonía móvil ha tenido un crecimiento enorme y con el avance de las nuevas tecnologías los teléfonos pasaron de ser dispositivos de comunicación mediante solo llamadas, a convertirse en un mini computador de bolsillo lo que ha permitido que a la par nazcan un sin fin de aplicaciones y herramientas que suplen las necesidades de los usuarios.

Una aplicación móvil es un software construido de forma similar a como se construye un sistema web o un software de escritorio pero igual que ellos, estas aplicaciones tienen unas características propias, enfoques, tipos y particularidades que hacen que su uso sea cada vez más requerido.

Existen aplicaciones móviles para lo que se pueda imaginar, desde apps de ocio, productividad, sociales, informativas entre otras, algunas de forma independiente y otras como apoyo para facilitar procesos ya existentes.



## ▼ ¿Porqué Android?

Existen muchas plataformas para móviles (iOS, Symbian, Windows Phone, BlackBerry, Palm, Java Mobile Edition, Linux Mobile (LiMo), Firefox OS, etc.); sin embargo Android presenta una serie de características que lo hacen diferente.

## ▼ Características Android

- Uno de los SO más populares y usados en la actualidad
- Adaptable a cualquier tipo de hardware.
- Portabilidad.
- Arquitectura basada en componentes.
- Dispositivos con acceso a internet.
- Servicios incorporados.
- Seguridad.
- Optimización, memoria y potencia.
- Gráficos y Sonido optimizados.

- Plataforma abierta.
- Facilidad en la construcción de Apps bajo este sistema operativo

### ▼ Origen Android.

- Desarrollado por Android Inc desde 2003. Hasta que fue comprado por Google en 2005.
- Se liberó el código bajo licencia Apache al crearse la Open Handset Alliance el 5 de Noviembre de 2007. También considerado el cumpleaños de Android.
- En 2008 se crean los primero chips compatibles y se lanza el primer teléfono Android, el HTC Dream.
- Se empiezan a nombrar como dulces a las versiones de Android.

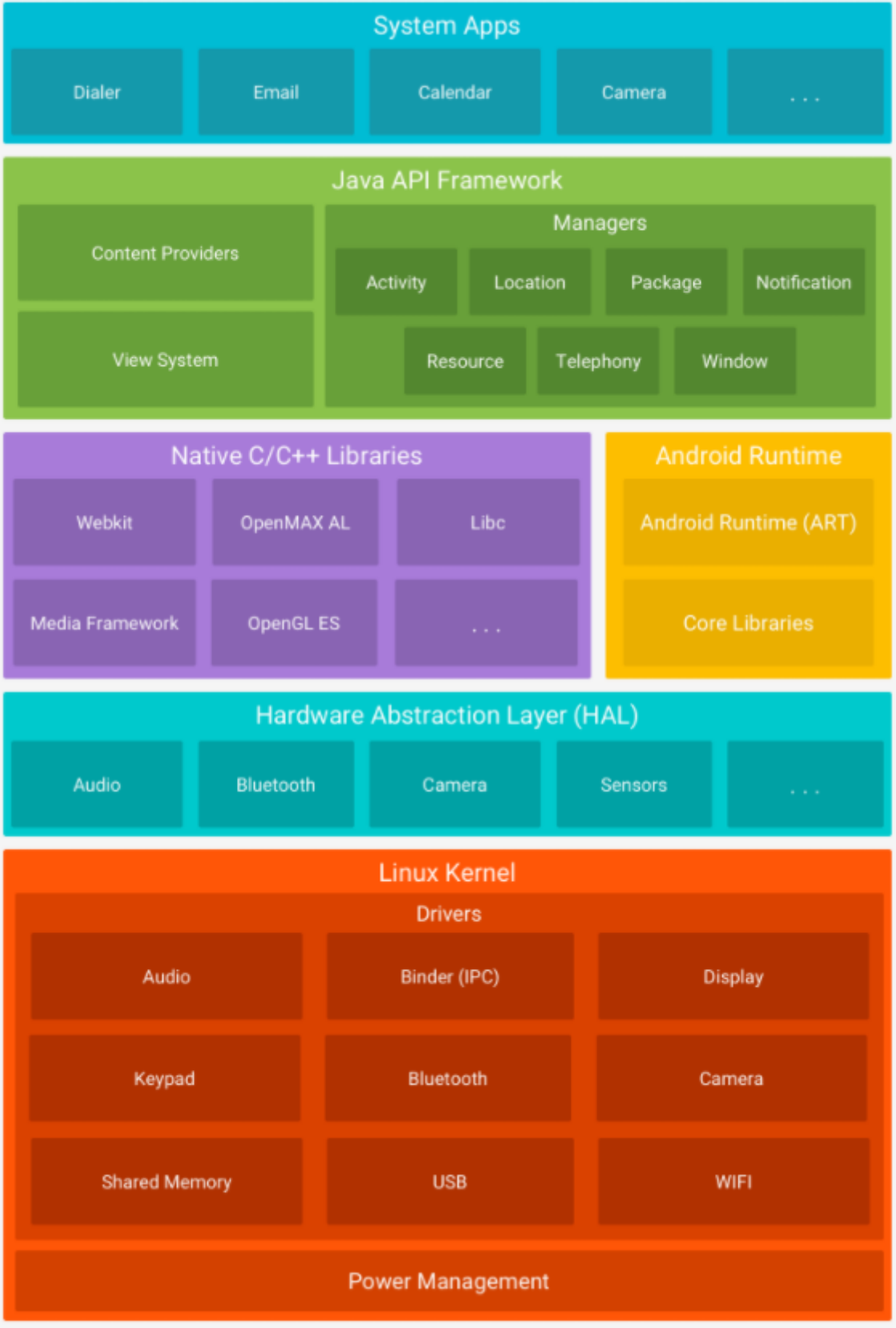
### ▼ Versiones Android

En android se manejan 3 formas de denominar las versiones:

- Versión Comercial
- Versión Fabricantes
- Versión Desarrollo

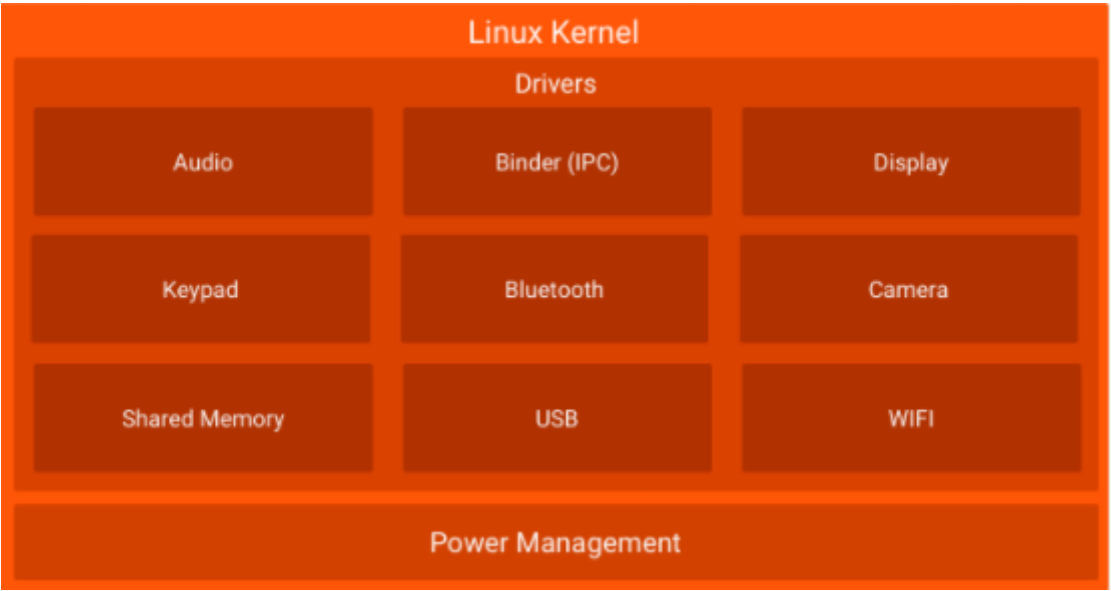
V. Comercial	V. Fabricantes	V. Desarrollo (API)	Fecha Lanzamiento
Apple Pie	1.0	1	Sep 23 de 2008
Banana Bread	1.1	2	Feb 9 de 2009
Cupcake	1.5	3	Abr 25 de 2009
Donut	1.6	4	Sept 15 de 2009
Eclair	2.0 / 2.1	5 / 7	Oct 26 de 2009
Froyo	2.2 / 2.2.3	8	Mayo 20 de 2010
Gingerbread	2.3 / 2.4.7	9 / 10	Dic 6 de 2010
Honeycomb	3.0 / 3.2.6	11 / 12 / 13	Feb 22 de 2011
Ice Cream Sandwich	4.0 / 4.0.5	14 / 15	Oct 18 de 2011
Jelly Bean	4.1 / 4.3.1	16 / 17 / 18	Jul 9 de 2012
KitKat	4.4 / 4.4.4	19 / 20	Oct 31 de 2013
Lollipop	5.0 / 5.1.1	21 / 22	Nov 12 de 2014
Marshmallow	6.0 / 6.0.1	23	Oct 5 de 2015
Nougat	7.0 / 7.1.2	24 7 25	Jun 15 de 2016
Oreo	8.0 / 8.1	26 / 27	Agos 21 de 2017
Pie	9.0 / 9.1	28	Agos 6 de 2018
Android 10	10.0	29	Sept 3 de 2019
Android 11	11.0	30	Sep 8 de 2020
Android 12	12.0	31	Agos - Sept de 2021

### ▼ Arquitectura Android



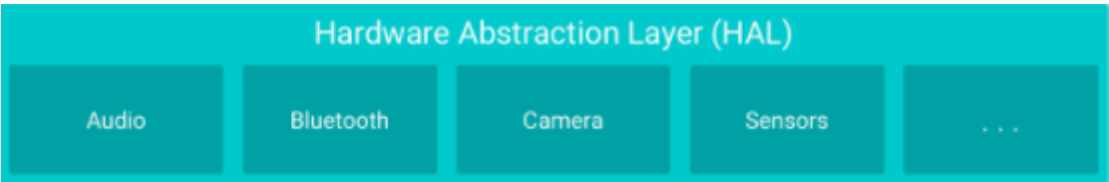
▼ **Linux Kernel**

Esta capa es la base de la plataforma, en ella se definen procesos para la administración de memoria de bajo nivel, proporciona servicios de seguridad, multiprocesos, soporte de drivers para dispositivos, esta capa es la única dependiente del hardware.



▼ **Hardware Abstraction Layer (HAL)**

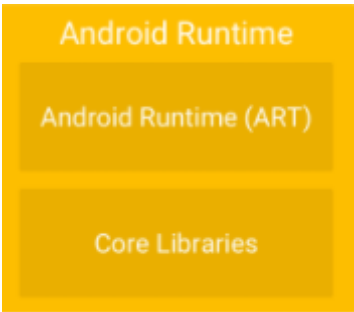
Esta capa brinda interfaces que exponen las capacidades de hardware de los dispositivos para la interacción con la API de java, brinda diferentes módulos que implementan interfaces para el uso de componentes de hardware, por ejemplo para el uso de cámara, GPS, bluetooth etc.



▼ **Android Runtime**

Está basado en el concepto de máquina virtual usado en java, y brinda todo el entorno de ejecución necesario para probar las aplicaciones, inicialmente antes de **android 5 (API 21)** se usaba el entorno de ejecución **Dalvik** pero para versiones iguales o superiores se reemplaza por **ART** que optimiza este proceso permitiendo ejecutar varias máquinas virtuales en dispositivos de baja memoria, mejorando también la compatibilidad con la depuración y optimización de recursos.

Adicional vincula nuevas mejoras en cuanto a compatibilidad y acceso a librerías java.



▼ **Librerías Nativas C/C++**

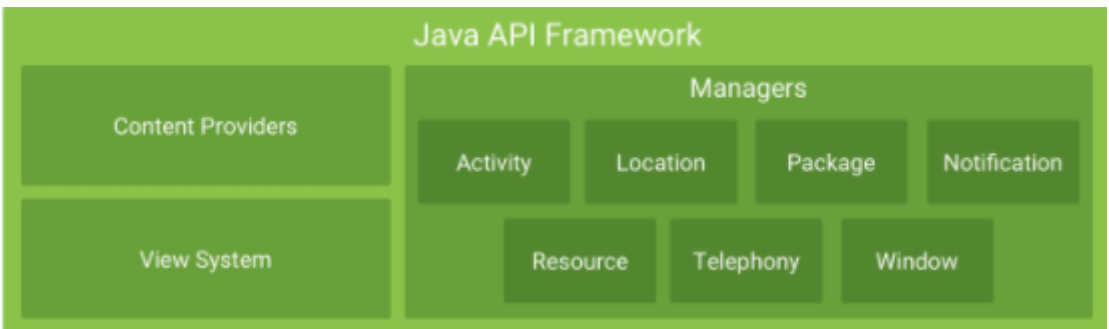
Muchos de los componentes centrales de Android se basan código nativo que requiere librerías escritas en C y C++, esta capa nos permite precisamente acceder a componentes nativos dependiendo de nuestras necesidades, por ejemplo para el trabajo con librerías 2D, 3D, WebKit, SSL entre otros.

Para desarrollo nativo con C o C++ se debe usar el NKD de Android (Native Development Kit).



▼ **Framework API Java**

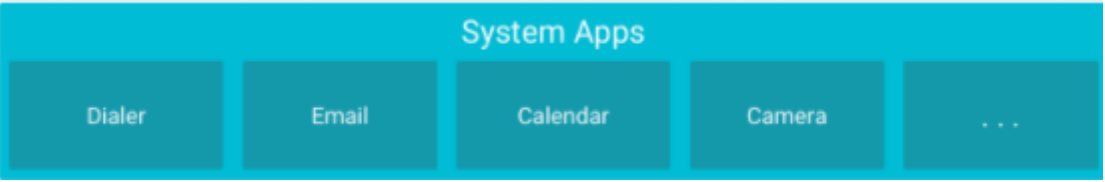
El conjunto de funciones del Sistema operativo está disponible mediante API's desarrolladas en JAVA, esta capa brinda el soporte para la creación de actividades, content providers, manejo del ciclo de vida, componentes gráficos, elementos de localización entre otras funcionalidades, esto es independiente del lenguaje con el que se decida desarrollar, sea java o kotlin, los cuales requieren del SDK.



▼ **System Apps**

Esta capa está formada por el conjunto de aplicaciones instaladas por defecto en una máquina Android, brinda apps para la gestión de calendario, acceso a internet, mensajería, correo electrónico, contactos entre otros.

Estas apps brindan una alternativa para los usuarios, permitiendo que no sea necesaria la instalación de otras aplicaciones externas, sin embargo no lo impide.



***Instructor: Cristian David Henao Hoyos***