



Artículo

# Inicia a programar para Android: ¿Qué necesito?



Uriel Ramírez 23 PlatziRank ⌚ Sept. 14, 2016

A medida que Android madura como plataforma, las herramientas que nos apoyan durante el tiempo de desarrollo de nuestra aplicación mejoran. No miento, **esta es la mejor época para construir este tipo de proyectos.**

Si ya llevas tiempo en esto, quizá muchas cosas te resulten familiares; pero si estás comenzando, felicito tu entusiasmo y te comparto algunos elementos que te ayudarán a crear el **mejor entorno de trabajo** posible al desarrollar para esta plataforma.

## ¿Dijo Java, ¿puedo desarrollar en otro lenguaje?

Existen algunas opciones que te permiten portar otros lenguajes a una aplicación Android. Pero **\*\*Java sigue siendo la opción si quieres alcanzar todo el potencial de la plataforma.** **\*\***Lo sé, puede que no sea de tu agrado; pero no dejes que un lenguaje te limite. Seguro será difícil al inicio, pero aprenderás muchos conceptos que puedes llevar a otros lenguajes. Y, ¿Qué tal que descubres que sí te gusta?

## ¿Qué necesito instalar para empezar a Desarrollar en Android?

### 1) Java Development Kit (JDK) 7 (Minimo)

2) Si importas el sistema operativo que uses en tu computadora; es **primordial** [descargar el Java Development Kit desde su sitio oficial](#).

3) Esto porque Android Studio usa la JVM y Dalvik(ART) para compilar tu aplicación.

4) \*Posibles errores: **\*\***Puede que la variable de entorno no se configuré por defecto en tu sistema operativo (más frecuente en Windows), para ello, te recomiendo visitar el siguiente enlace:

6

7

SIGUE CON:

[Arquitectura de una aplicación en Android](#)



```


## 10) Android Studio



11 El IDE oficial para desarrollar en Android es Android Studio (Eclipse ya no es soportado). Es gratuito y multiplataforma. Además, si tienes alguna duda de qué requerimientos mínimos debes tener, puedes consultarlos en la misma página de descarga.



El instalador es muy sencillo, nada ajeno a hacer click en siguiente en cada paso.



## 12) SDK



Cada versión de Android cuenta con su propio SDK. El nombre comercial de cada una está inspirado en un dulce y para los desarrolladores sólo cambia el número de API. Al instalar Android Studio, tendrás algunas versiones instaladas; pero el verdadero valor es manejarlas por nuestra cuenta.



Para lograrlo, en la página de inicio de Android Studio, localiza la opción le settings:

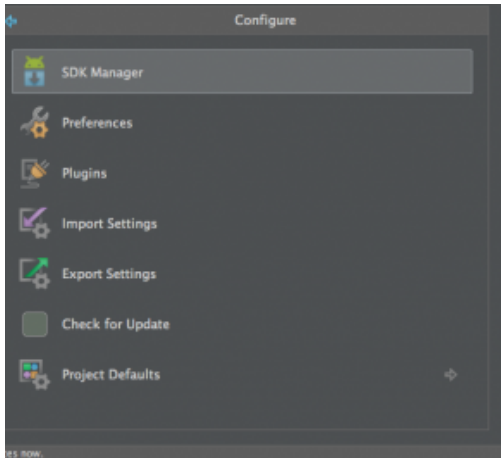


SIGUE CON:  

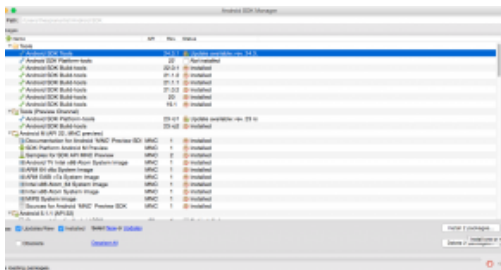
Arquitectura de una aplicación en Android


```

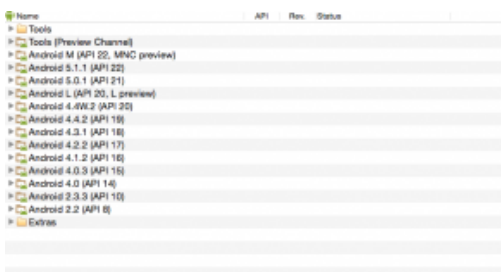
> ve a SDK Manager:



Las actualizaciones del SDK son independientes de las actualizaciones Android Studio. Y es importante tenerlas al día. Esto se puede configurar de forma automática, tal como con Android Studio.



El SDK Manager nos muestra de manera gráfica el nombre del paquete, la versión y si lo tenemos instalado. Además, Contamos con 3 categorías: Tools, SDK API (Level) y extras:



En esta sección nos encontramos todos los paquetes que necesita Android Studio para interpretar tu código, compilar y ejecutar de manera correcta.

## SDK API (level)

Aquí es donde podemos instalar las diferentes versiones de Android con las que queremos desarrollar. A su vez, cada versión cuenta con paquetes como se muestra a continuación:

Android M (API 22, MNC preview)			
Documentation for Android 'MNC' Preview SDI	MNC	1	Installed
SDK Platform Android M Preview	MNC	1	Installed
Samples for SDK API MNC Preview	MNC	2	Installed
Android TV Intel x86 Atom System Image	MNC	1	Installed
ARM 64 v8a System Image	MNC	1	Installed
ARM EABI v7a System Image	MNC	1	Installed
Intel x86 Atom_64 System Image	MNC	1	Installed
Intel x86 Atom System Image	MNC	1	Installed
MIPS System Image	MNC	1	Installed
Sources for Android 'MNC' Preview SDK	MNC	1	Installed
Android 5.1.1 (API 22)			
Documentation for Android SDK	22	1	Not installed
SDK Platform	22	2	Installed
Samples for SDK	22	6	Installed
Android TV ARM EABI v7a System Image	22	1	Not installed
Android TV Intel x86 Atom System Image	22	1	Installed
Android Wear ARM EABI v7a System Image	22	2	Not installed
Android Wear Intel x86 Atom System Image	22	2	Installed
ARM EABI v7a System Image	22	1	Not installed
Intel x86 Atom_64 System Image	22	1	Not installed
Intel x86 Atom System Image	22	1	Not installed

**SDK Platform:** es obligatorio para poder crear y ejecutar tus aplicaciones de acuerdo a las versiones a las que quieras dar soporte. Por ejemplo, si quieres desarrollar a partir de Ice Cream Sandwich, instala Android 4.0 (API 14). Si solo quieres soporte para Lollipop instala 5.0.1 (API 20).

Como mínimo, se recomienda instalar la versión menor y la más reciente a la que quieras soportar. No te preocupes por versiones anteriores a Ice Cream Sandwich, ya no vale la pena sacrificar funcionalidad por la cantidad de dispositivos corriendo esta versión.

**Samples:** instalará todos los ejemplos de ese SDK (opcional).

**System Images:** aquí es donde encontramos las diversas imágenes para usar el emulador nativo de Android Studio (AVD). Estas tienen gran peso y especifican si son para teléfonos, Wear, Glass, TV o Auto.

**Sources:** recordemos que Android es Open Source y tenemos acceso al código fuente e imágenes del mismo. No es necesario instalarlo.

\*Extras: \*\*incluye todas las librerías oficiales de Google; además de herramientas que mejorarán tu experiencia al ejecutar las aplicaciones. En este caso, te recomiendo instalar todo.

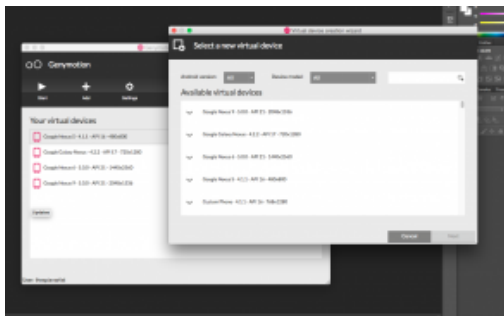


➤ no ocupan tanto espacio. Y ahora que ya tienes todo instalado, ya puedes empezar a crear tus aplicaciones.

## ¿No tengo un teléfono Android: ¿El emulador nativo es suficiente?

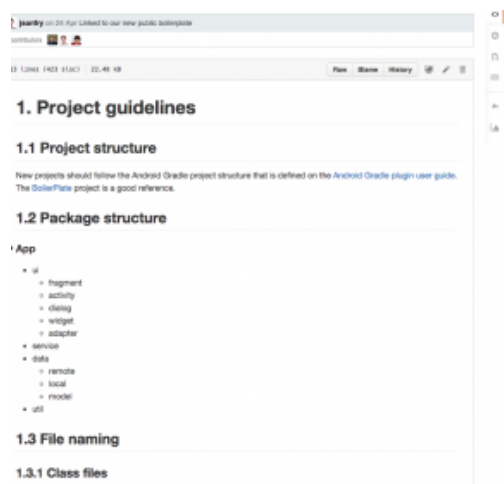
---

Es bueno, pero consume muchos recursos y no es tan rápido. Yo te recomiendo usar [Genymotion](#). Tiene una versión gratuita y sólo requiere de un plugin para funcionar en Android Studio (en la página de descarga te muestran cómo instalarlo). Además, crear una máquina virtual es muy sencillo; ya que cuenta con presets de equipos Android que actualmente existen en el mercado.



## Existe algún estándar o guía a seguir en mi proyecto?

---



En proyectos grandes y/o colaborativos, es importante mantener nuestro código y archivos lo más ordenados y legibles posible. Para ello, es recomendado **seguir la guía propuesta por la**

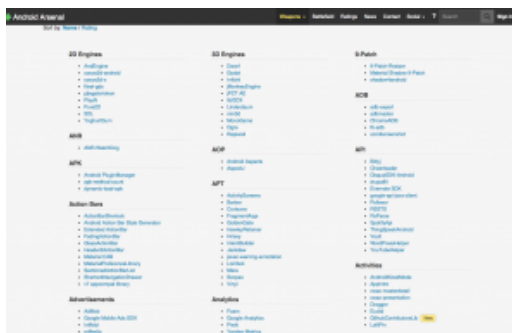


## ➤ ¿Dónde puedo encontrar librerías y recursos para mis proyectos?



Google constantemente actualiza su documentación y escribe sobre algún cambio importante en su [blog](#). Aunque también es una buena idea seguir los canales de [Android](#) y de [Diseño](#) en Google+.

Además, en [Android Arsenal](#) encontrarás un amplio catálogo de librerías recomendadas por la comunidad para usar en tu proyecto. [Gradle](#), [please](#) también hace buen trabajo si ya sabes lo que buscas



## ➤ Por último

Si se te ocurre alguna otra herramienta que quieras recomendar, la sección de comentarios es el lugar ideal para hacerlo. Continuemos con el curso.



## Suma tu comentario

Top Nuevas



**carlosmillans3** 24 Puntos

🕒 9 meses

Este enlace está excelente sobre librerías:

<http://blog.etongwl.com/archives/931.html>



**ThespianArtist**

Cierto, es muy bueno

🕒 9 meses

Responder



**iks\_kill** 0 Puntos

🕒 8 meses

Hasta ahorita muy buen contenido, solo quisiera agregar que en mi poca experiencia desarrollando en android he probado varios emuladores y yo como usuario de windows recomiendo ampliamente usar genymotion.



**Rgoyzueta**

hace diferencia instalar con Virtual Box o no??

🕒 7 meses



**asahicantu**

🕒 23 días

Para usuarios Windows recomiendo probar Visual Studio Emulator for Android, puede encontrarse en el siguiente [enlace](#). Fue sorprendente el rendimiento y la velocidad con la que funciona. Uno debe tener activado Hyper-V dentro de Windows.

Responder



**alafourcadedespaigne** 0 Puntos

🕒 6 meses

Para buenas ideas Material design aquí esta este sitio



**alafourcadedespaigne**

🕒 6 meses

<https://material.uplabs.com/>

Responder





**judisaro10** 0 Puntos

🕒 5 meses


SIGUE CON:

[Arquitectura de una aplicación en Android](#)



>  0  **alx\_rodriguez** 0 Puntos 🕒 8 meses  
Genymotion puede ser una gran alternativa para los que no tenemos procesador Intel y tendríamos dificultades para correr el SDK :)

Responder

 0  **reiterweg** 3 Puntos 🕒 10 meses  
El enlace del canal Diseño lleva a una página con error 404.

 2  **ThespianArtist** 🕒 10 meses  
Gracias por avisar, mira aquí el enlace actualizado: <https://plus.google.com/communities/107975318755450452035>

(Igual la corrección en el post)

 0  **reiterweg** 🕒 10 meses  
Perfecto, ese enlace si funciona!



 0  **nelsonangaritaparra** 🕒 9 meses  
Me solicita JDK 8 para Android 7.0

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

---

🔍 Ver todas las respuestas

Responder

 0  **mgoscar** 108 Puntos 🕒 4 meses  
Usare Genymotion, a ver si la emulación se ejecuta mejor ya que uno de los puntos que tengo encontra de Android Studio es este.

Responder

 0  **jusbarylazaro** 0 Puntos 🕒 9 meses  
Muy buen material para tener disponible

Responder

  **edgarmoises** 232 Puntos 🕒 3 meses








Gracias por avisarnos, lo arreglaremos pronto



 **GOLLUM23**

Ya quedaron arregladas las imagenes, muchas gracias por el reporte

 3 meses

Responder




0



**MYCHIP** 254 Puntos

Estoy muy emocionado con este curso, gracias Platzi

 2 meses

Responder

SIGUE CON:

[Arquitectura de una aplicación en Android](#)

