

# Instalación de un servidor local de App Inventor 2 en LINUX (rev.1)

## Requisitos previos (a Mayo 2015):

- **Versión 7 de Oracle Java JDK.** Otras versiones pueden funcionar pero no se han probado oficialmente. No utilices para este proyecto OpenJDK ya que no puede firmar paquetes "jar" (le falta la utilidad "jarsigner").
- **Apache ant 1.8.2** (o nueva). Como herramienta para auxiliar para compilar
- **git:** Para descargarnos el código fuente del proyecto a compilar.
- **Google AppEngine Java SDK.** Librerías para la creación de Apps
- **Python.** Suele venir por defecto con el sistema.

*NOTA: si el equipo ejecuta una versión de 64bits de Ubuntu será necesario añadir la compatibilidad para aplicaciones de 32 bits. Puedes averiguar la arquitectura del sistema instalado ejecutando la orden "uname -a" en un terminal:*

```
administrador@aula2profesor: ~
Archivo Editar Ver Buscar Terminal Ayuda
administrador@aula2profesor:~$ uname -a
Linux aula2profesor 3.19.0-031900-lowlatency #201502091451 SMP PREEMPT Mon Feb 9 14:58:45 UTC 2015
x86_64 x86_64 x86_64 GNU/Linux
administrador@aula2profesor:~$
```

Si fuese necesario puedes instalarlos con la orden:

```
sudo apt-get install lib32z1 lib32ncurses5 lib32bz2-1.0 lib32stdc++6
```

## Descarga e instalación de Oracle Java 7 JDK

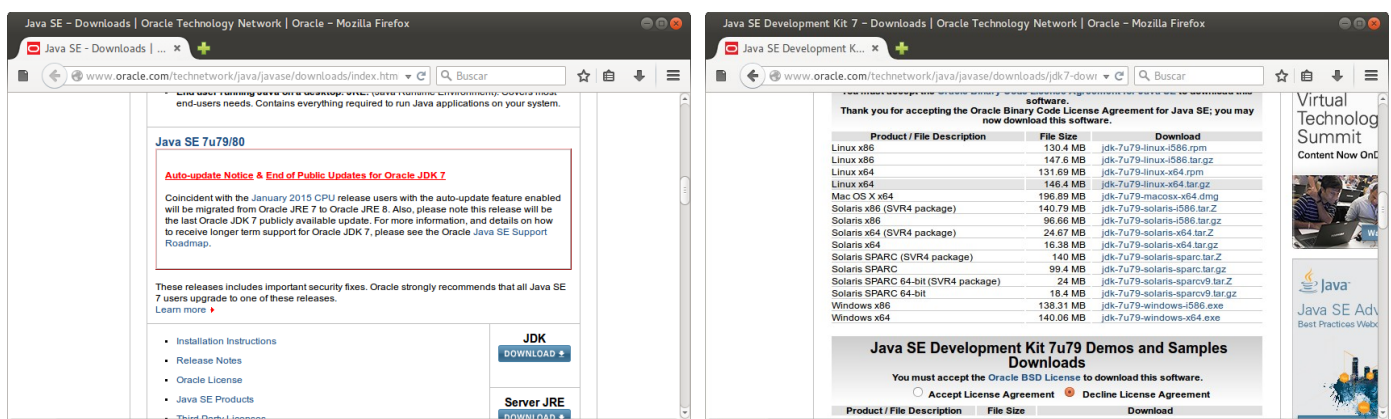
[1] Elimina cualquier instalación previa de OpenJDK ejecutando la orden

```
sudo apt-get purge openjdk-*
```

[2] Accede a la página de descargas de Java JDK y bájate la versión apropiada para tu arquitectura (32 o 64bits):

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

*NOTA: Las versión que necesitamos está por debajo de las descargas de la actual v8.*



**[3]** Descomprime el paquete descargado (en mi caso *"jdk-7u79-linux-x64.tar.gz"*) en la carpeta *"/opt"*. Luego puedes eliminar el archivo original.

```
sudo tar -zxf jdk-7u79-linux-x64.tar.gz -C /opt
rm jdk-7u79-linux-x64.tar.gz
```

**[4]** Modifica el fichero *"/etc/profile"* para agregar las variables del sistema que permitirán localizar la instalación de los binarios de **Java**. Ejecura la orden *"sudo gedit /etc/profile"* y añade al final del fichero estas líneas:

```
# Preparación entorno para Java 7
JAVA_HOME=/opt/jdk1.7.0_79
JRE_HOME=$JAVA_HOME/jre
PATH=$PATH:$JAVA_HOME/bin:$JRE_HOME/bin
export JAVA_HOME
export JRE_HOME
export PATH
```

**[5]** Informa a tu sistema Ubuntu Linux dónde está ubicado Oracle Java JDK/JRE.

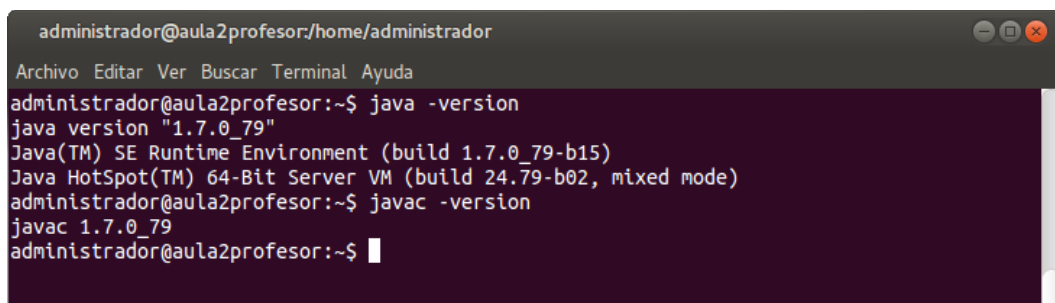
```
sudo update-alternatives --install "/usr/bin/java" "java" "/opt/jdk1.7.0_79/bin/java" 1
sudo update-alternatives --install "/usr/bin/javac" "javac" "/opt/jdk1.7.0_79/bin/javac" 1
sudo update-alternatives --install "/usr/bin/javaws" "javaws" "/opt/jdk1.7.0_79/bin/javaws" 1
sudo update-alternatives --set java /opt/jdk1.7.0_79/bin/java
sudo update-alternatives --set javac /opt/jdk1.7.0_79/bin/javac
sudo update-alternatives --set javaws /opt/jdk1.7.0_79/bin/javaws
```

**[6]** Vuelve a cargar las rutas del sistema escribiendo el comando *"source /etc/profile"*.

NOTA: Ten en mente que el *PATH* del sistema descrito en */etc/profile* se cargará después al reiniciar Ubuntu.

**[7]** Revisa si Oracle Java se instaló correctamente en el sistema. Ejecuta los siguientes comandos y anota la versión de Java:

```
java -version
javac -version
```



```
administrador@aula2profesor:/home/administrador
Archivo Editar Ver Buscar Terminal Ayuda
administrador@aula2profesor:~$ java -version
java version "1.7.0_79"
Java(TM) SE Runtime Environment (build 1.7.0_79-b15)
Java HotSpot(TM) 64-Bit Server VM (build 24.79-b02, mixed mode)
administrador@aula2profesor:~$ javac -version
javac 1.7.0_79
administrador@aula2profesor:~$
```

# Descarga e instalación de Apache Ant

[1] Accede a la página de descargas de los binarios de esta herramienta:

<http://ant.apache.org/bindownload.cgi>

En el ejemplo está disponible como última versión **"apache-ant-1.9.4-bin.tar.gz"**

[2] Descomprime el paquete descargado (en mi caso **"apache-ant-1.9.4-bin.tar.gz"**) en la carpeta **"/opt"**. Luego puedes eliminar el archivo original.

```
sudo tar -zxf apache-ant-1.9.4-bin.tar.gz -C /opt  
rm apache-ant-1.9.4-bin.tar.gz
```

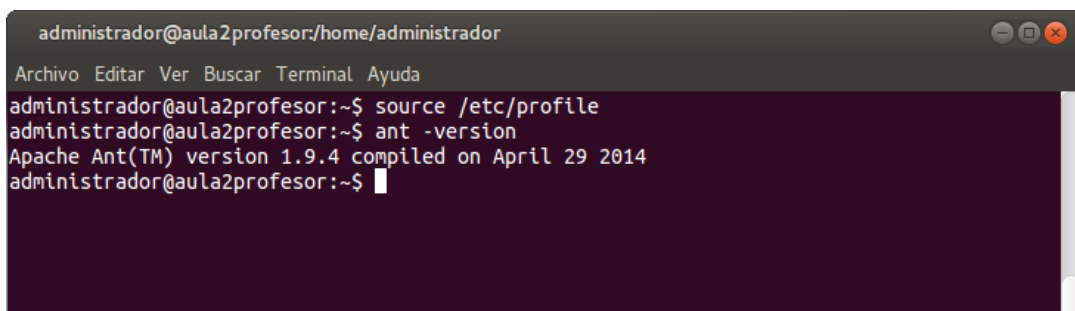
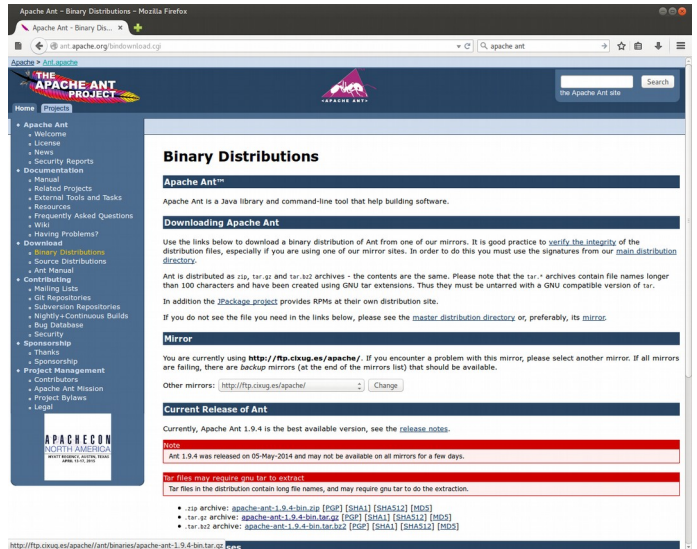
[3] Modifica el fichero **"/etc/profile"** para agregar las variables del sistema que permitirán localizar la instalación de los binarios de **Ant**. Ejecuta el orden **"sudo gedit /etc/profile"** y añade/actualiza al final del fichero la líneas resaltadas tal como aparece a continuación:

```
# Preparación entorno para Java 7  
JAVA_HOME=/opt/jdk1.7.0_79  
JRE_HOME=$JAVA_HOME/jre  
ANT_HOME=/opt/apache-ant-1.9.4  
PATH=$PATH:$JAVA_HOME/bin:$JRE_HOME/bin:$ANT_HOME/bin  
export JAVA_HOME  
export JRE_HOME  
export ANT_HOME  
export PATH
```

[4] Vuelve a cargar las rutas del sistema escribiendo el comando **"source /etc/profile"**.  
NOTA: Ten en mente que el **PATH** del sistema descrito en **/etc/profile** se cargará después al reiniciar Ubuntu.

[5] Revisa si Apache Ant se instaló correctamente en el sistema. Ejecuta el siguiente comando y anota la versión:

```
ant -version
```



# Descarga e instalación de Google App Engine SDK for Java

[1] Accede a la página de descargas del App Engine de Google:

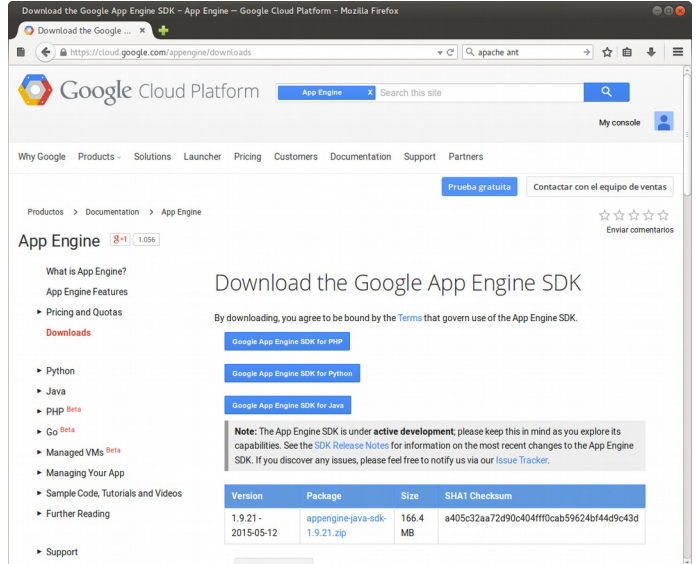
<https://cloud.google.com/appengine/downloads>

Pulsa sobre el botón "Google App Engine SDK for Java" para desplegar el enlace de descarga y las instrucciones de instalación

En el ejemplo está disponible como última versión "**appengine-java-sdk-1.9.21.zip**"

[2] Descomprime el paquete descargado en la carpeta "/opt". Luego puedes eliminar el archivo original.

```
sudo unzip appengine-java-sdk-1.9.21.zip -d /opt
rm appengine-java-sdk-1.9.21.zip
```



[3] Modifica el fichero "*/etc/profile*" para agregar las variables del sistema que permitirán localizar la instalación de **AppEngine**. Ejecuta la orden "`sudo gedit /etc/profile`" y añade/actualiza al final del fichero la líneas resaltadas tal como aparece a continuación:

```
# Preparación entorno para Java 7
JAVA_HOME=/opt/jdk1.7.0_79
JRE_HOME=$JAVA_HOME/jre
ANT_HOME=/opt/apache-ant-1.9.4
GAP_HOME=/opt/appengine-java-sdk-1.9.21/
PATH=$PATH:$JAVA_HOME/bin:$JRE_HOME/bin:$ANT_HOME/bin:$GAP_HOME
export JAVA_HOME
export JRE_HOME
export ANT_HOME
export PATH
```

[4] Vuelve a cargar las rutas del sistema escribiendo el comando

```
source /etc/profile
```

NOTA: Ten en mente que el *PATH* del sistema descrito en */etc/profile* se cargará después al reiniciar Ubuntu. **En este punto es recomendable hacerlo antes de continuar.**

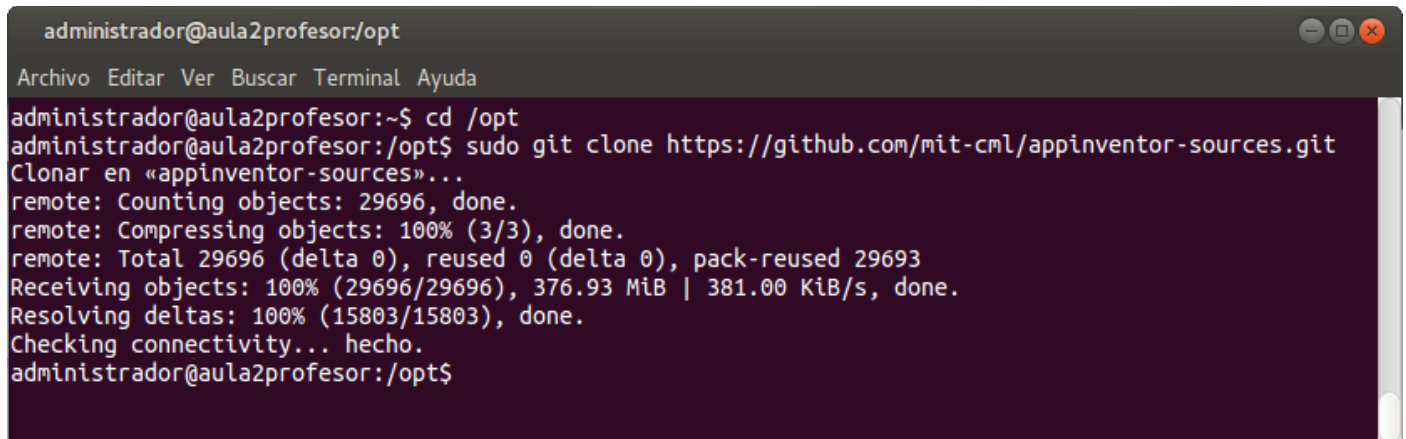
# Clonado y compilación del código de Mit App Inventor

[1] Asegúrate de tener instalado el cliente de Git con la orden:

```
sudo apt-get install git
```

[2] Desplázate a la carpeta “/opt” y ejecuta la orden para clonar el código del proyecto de AppInventor desde los servidores de “github”

```
cd /opt
sudo git clone https://github.com/mit-cml/appinventor-sources.git
```



```
administrador@aula2profesor:/opt
Archivo Editar Ver Buscar Terminal Ayuda
administrador@aula2profesor:~$ cd /opt
administrador@aula2profesor:/opt$ sudo git clone https://github.com/mit-cml/appinventor-sources.git
Clonar en «appinventor-sources»...
remote: Counting objects: 29696, done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 29696 (delta 0), reused 0 (delta 0), pack-reused 29693
Receiving objects: 100% (29696/29696), 376.93 MiB | 381.00 KiB/s, done.
Resolving deltas: 100% (15803/15803), done.
Checking connectivity... hecho.
administrador@aula2profesor:/opt$
```

[3] Una vez descargado el código fuente asígnate como propietario de todo el contenido descargado o no funcionará la compilación.

```
sudo chown -R $USERNAME appinventor-sources
```

[4] Colócate en la subcarpeta “appinventor”, copia el fichero “sample-.gitignore” a “.gitignore” y ejecuta el programa “ant”...

```
cd /opt/appinventor-sources/appinventor/
cp sample-.gitignore .gitignore
ant
```

...Espera unos minutos hasta que termine el proceso y aparezca la frase “**BUILD SUCCESSFUL**”

[5] Ejecuta los servidores de la “aplicación principal” y el “constructor”, cada uno en un terminal diferente de la siguiente manera:

**Terminal 1 (es todo una misma línea):**

```
/opt/appengine-java-sdk-1.9.21/bin/dev_appserver.sh --port=8888 --address=0.0.0.0
/opt/appinventor-sources/appinventor/appengine/build/war
```

Acabará mostrando el mensaje “*INFORMACIÓN: Dev App Server is now running*”

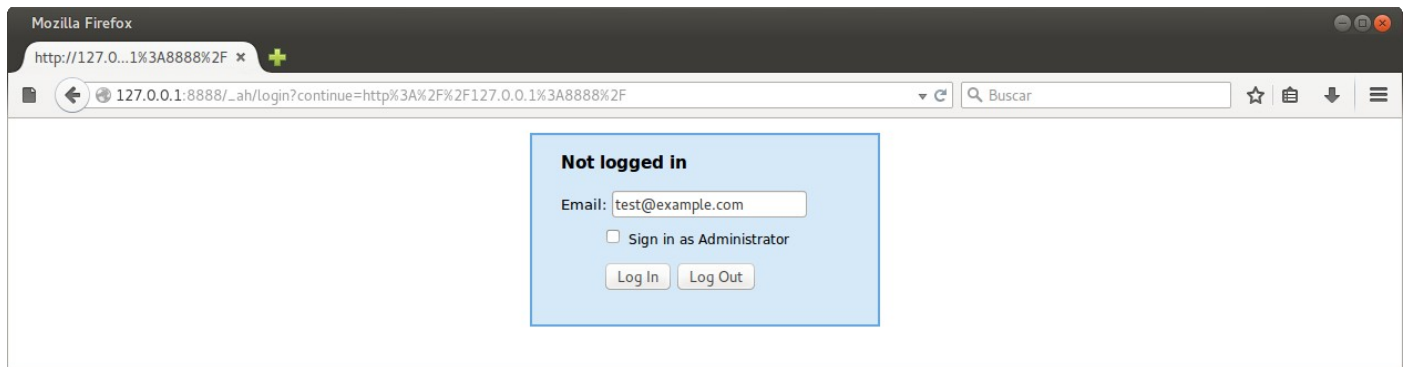
**Terminal 2:**

```
cd /opt/appinventor-sources/appinventor/buildserver/
ant RunLocalBuildServer
```

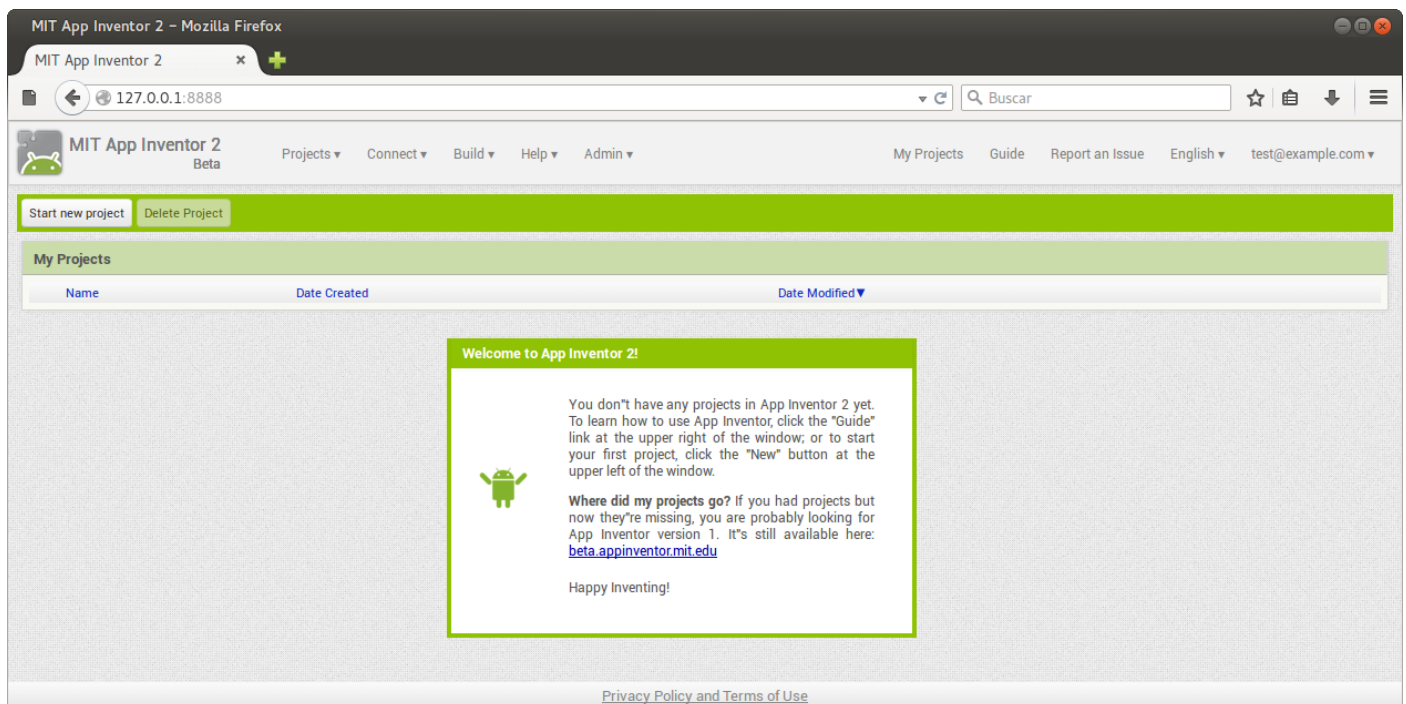
y acabará mostrando el mensaje “[java] *INFORMACIÓN: Server running*”



[6] Inicia un navegador y abre la dirección “<http://127.0.0.1:8888>”.



Pulsa sobre “Log In” para acceder al entorno de desarrollo. Más adelante comentaremos las ventajas e inconvenientes de este método de identificación de usuarios.



EUREKA! FUNCIONA!