

Manual de Instalación

Windows 10/8.1/7

Git: Accediendo al siguiente link: <https://git-scm.com/download/win>

Escoger la versión de git (x32 o x64) según la arquitectura del procesador.



Your download is starting...

You are downloading the latest (2.18.0) 64-bit version of Git for Windows. This is the most recent [maintained build](#). It was released 23 days ago, on 2018-06-22.

If your download hasn't started, [click here to download manually](#).

Other Git for Windows downloads

Git for Windows Setup

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

JDK 8: Atraves del siguiente link se selecciona la versión de java según la arquitectura del procesador:

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

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- [Java Developer Newsletter](#): From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- [Java Developer Day hands-on workshops \(free\) and other events](#)
- [Java Magazine](#)

JDK 8u171 [checksum](#)

JDK 8u172 [checksum](#)

Java SE Development Kit 8u171

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

☒ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.97 MB	jdk-8u171-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.89 MB	jdk-8u171-linux-arm64-vfp-hflt.tar.gz
Linux x86	170.05 MB	jdk-8u171-linux-i586.rpm
Linux x86	184.88 MB	jdk-8u171-linux-i586.tar.gz
Linux x64	167.14 MB	jdk-8u171-linux-x64.rpm
Linux x64	182.05 MB	jdk-8u171-linux-x64.tar.gz
Mac OS X x64	247.84 MB	jdk-8u171-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.83 MB	jdk-8u171-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.19 MB	jdk-8u171-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	140.6 MB	jdk-8u171-solaris-x64.tar.Z
Solaris x64	97.05 MB	jdk-8u171-solaris-x64.tar.gz
Windows x86	199.1 MB	jdk-8u171-windows-i586.exe
Windows x64	207.27 MB	jdk-8u171-windows-x64.exe

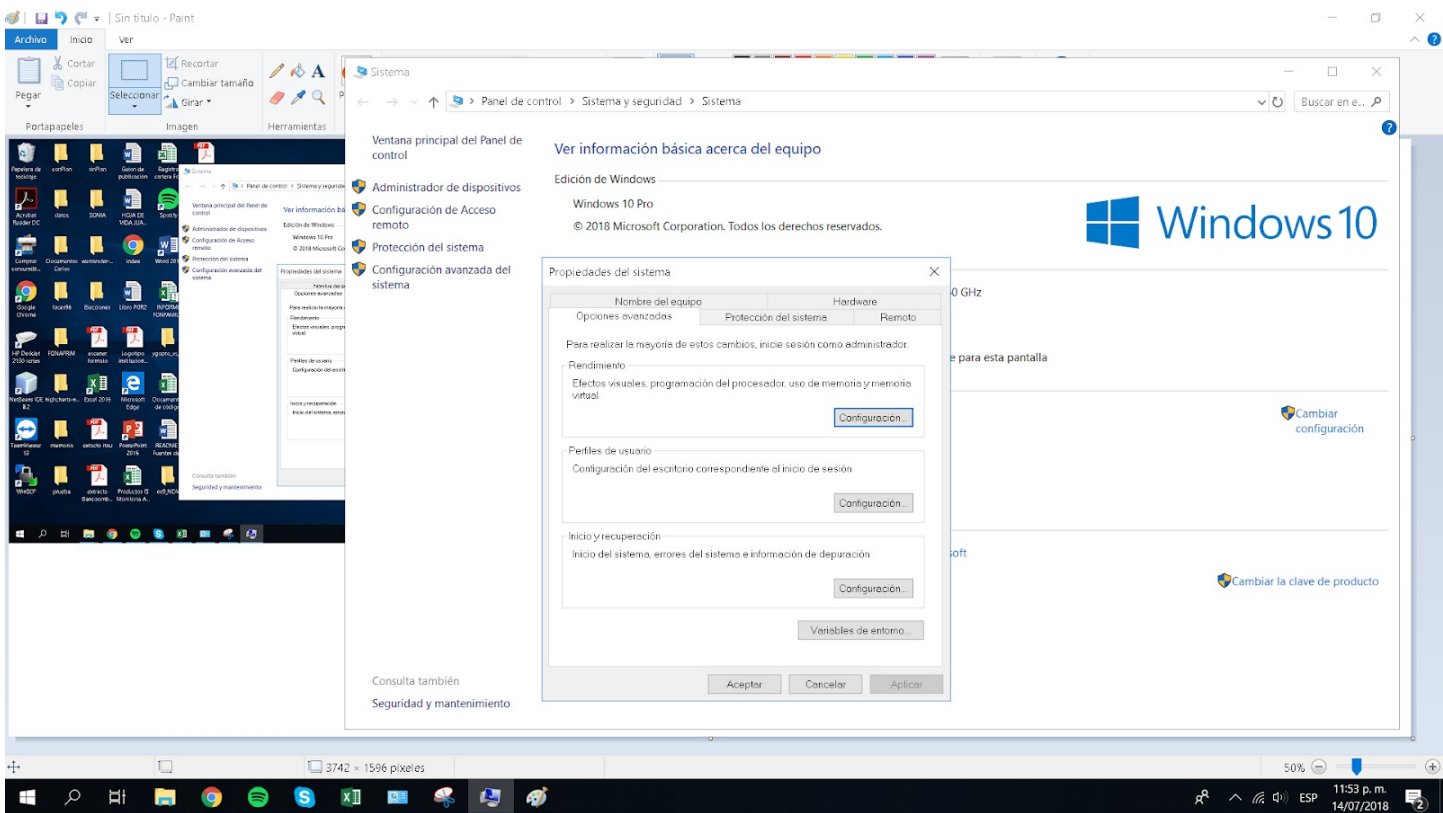
Una vez descargado e instalado se procede a configurar las variables de entorno.

Teniendo en cuenta la siguiente dirección donde Java hizo la instalación se realizará la configuración de las variables de entorno.

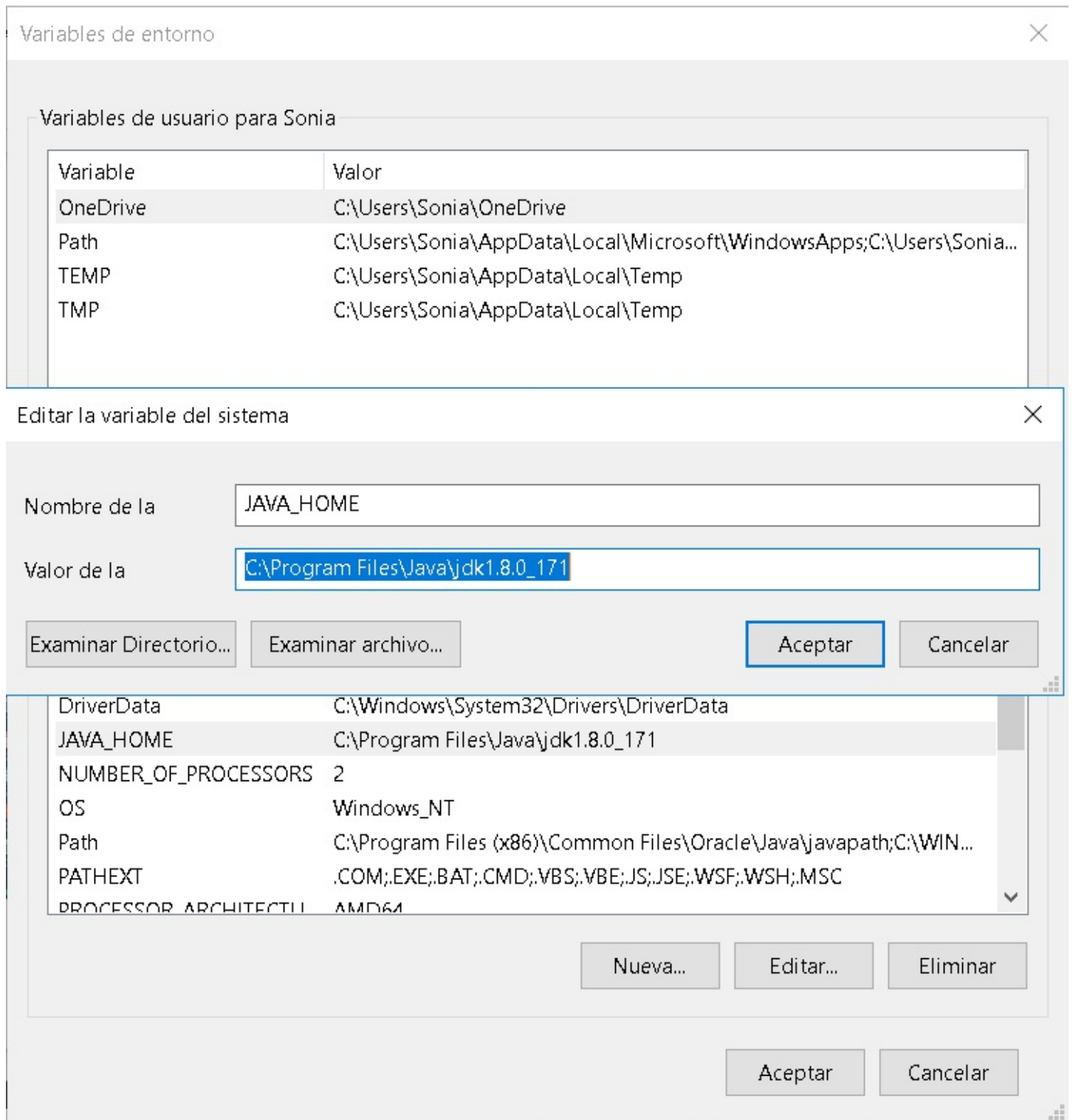
Este equipo > Disco local (C:) > Archivos de programa > Java

Nombre	Fecha de modifica...
apache-maven-3.5.4	9/07/2018 10:32 a.
jdk1.8.0_171	16/06/2018 5:33 p.
jre1.8.0_171	16/06/2018 5:33 p.

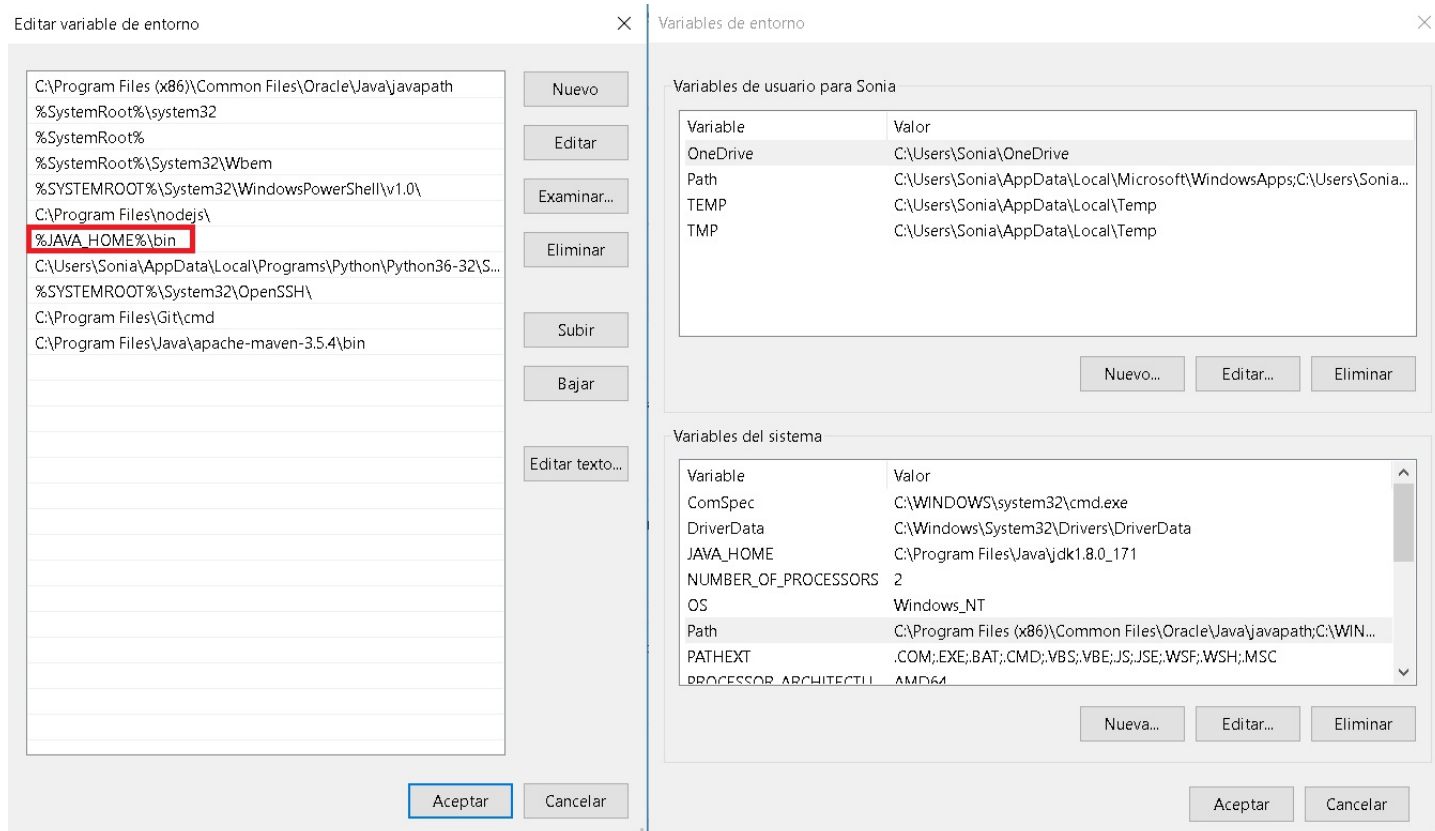
Accedemos a Sistema y damos click sobre la opción **“Configuración Avanzada del Sistema”** luego **“Variables de entorno”**.



Se crea una nueva variable llamada **“JAVA_HOME”** con la ruta donde está localizado el JDK.



Posteriormente se accede a la variable "Path" y se agrega la variable de entorno "JAVA_HOME" dando el valor como "%JAVA_HOME%\bin".



Se comprueba que java está instalado en el pc, accediendo al “Símbolo de sistema” y se escriben las siguientes líneas:

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

```
C:\Users\>java -version  
java version "1.8.0_171"  
Java(TM) SE Runtime Environment (build 1.8.0_171-b11)  
Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)  
  
C:\Users\>javac -version  
javac 1.8.0_171
```

Maven: Accediendo a la siguiente dirección

(<https://maven.apache.org/download.cgi>) se busca el bin de maven y se

descarga, después se deberá descomprimir en la dirección que desee, para la guía se dejará en el mismo lugar donde se instaló Java.

Files




Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the [installation instructions](#). Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release bundles against the public [KEYS](#) used by the Apache Maven developers.

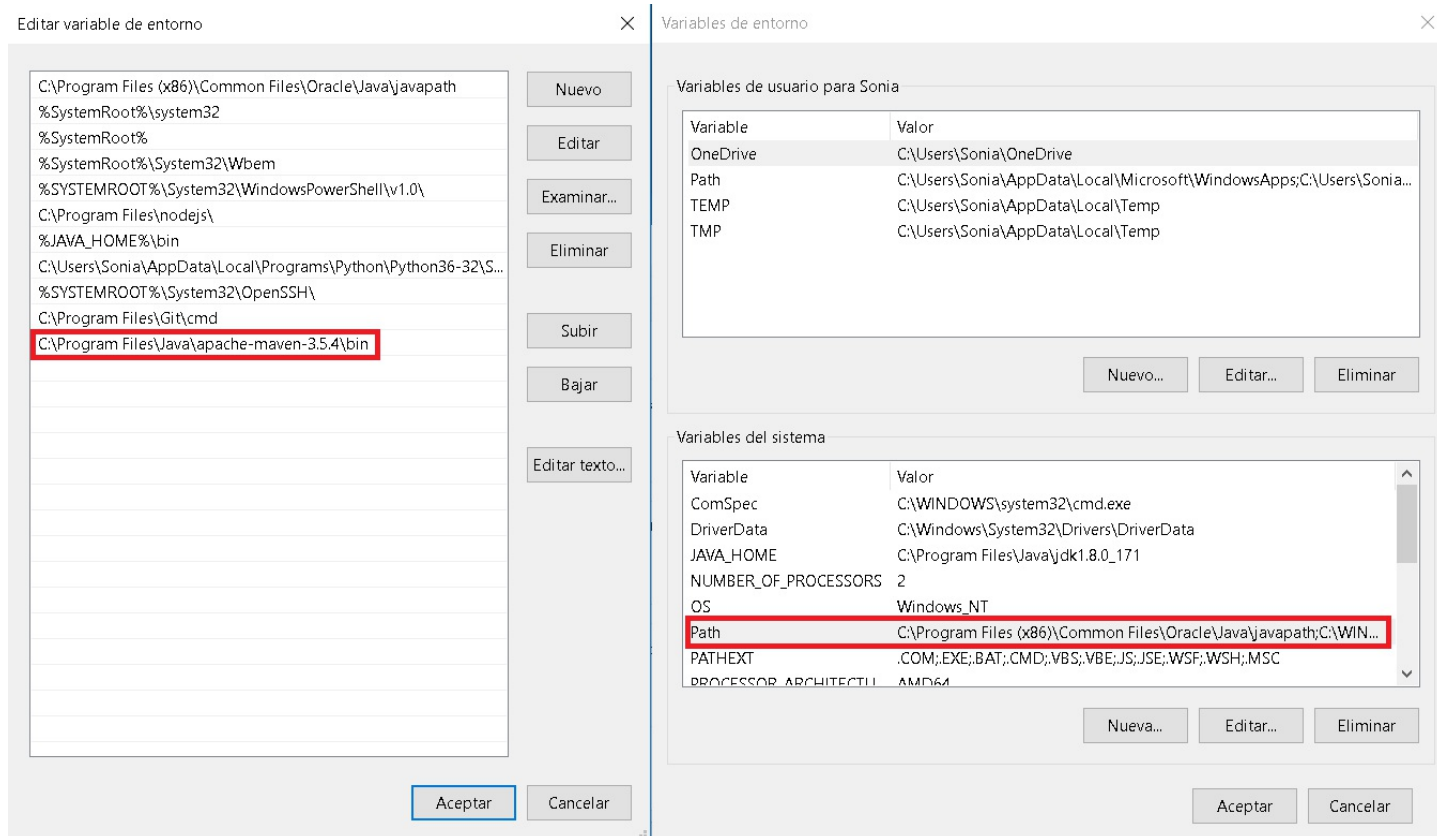
	Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.5.4-bin.tar.gz	apache-maven-3.5.4-bin.tar.gz.sha1	apache-maven-3.5.4-bin.tar.gz.asc
Binary zip archive	apache-maven-3.5.4-bin.zip	apache-maven-3.5.4-bin.zip.sha1	apache-maven-3.5.4-bin.zip.asc
Source tar.gz archive	apache-maven-3.5.4-src.tar.gz	apache-maven-3.5.4-src.tar.gz.sha1	apache-maven-3.5.4-src.tar.gz.asc
Source zip archive	apache-maven-3.5.4-src.zip	apache-maven-3.5.4-src.zip.sha1	apache-maven-3.5.4-src.zip.asc

- [Release Notes](#)
- [Reference Documentation](#)
- [Apache Maven Website As Documentation Archive](#)
- [All current release sources \(plugins, shared libraries,...\)](#) available at <https://www.apache.org/dist/maven/>
- [latest source code from source repository](#)
- Distributed under the [Apache License](#), version 2.0

Este equipo > Disco local (C:) > Archivos de programa > Java ▼ ↺ Buscar en J 🔍

Nombre	Fecha de modifica...	Tipo	Tamaño
 apache-maven-3.5.4	9/07/2018 10:32 a.	Carpeta de archivos	
 jdk1.8.0_171	16/06/2018 5:33 p.	Carpeta de archivos	
 jre1.8.0_171	16/06/2018 5:33 p.	Carpeta de archivos	

Se dirige a la configuración de variables de entorno “paso similar de instalación de Java” y se agrega el recurso al **Path**.



Cassandra: Descargar la versión de cassandra según la arquitectura del procesador:

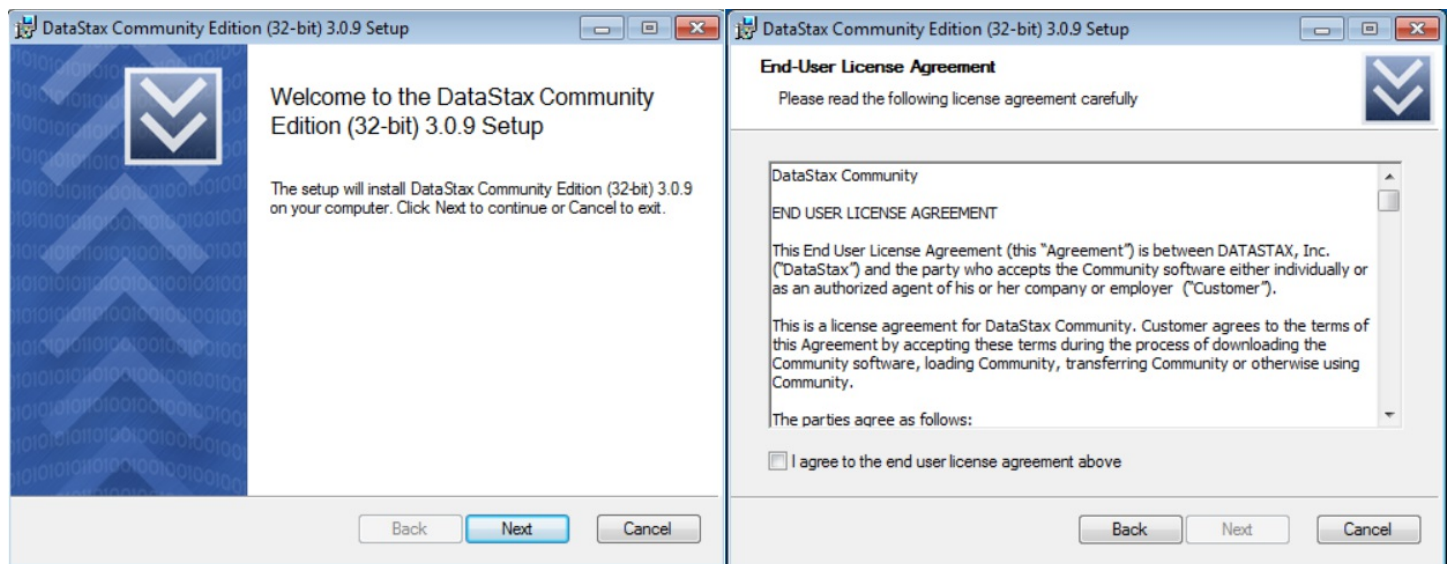
- MSI Installer (32 Bits):

http://downloads.datastax.com/community/datastax-community-32bit_3.0.9.msi

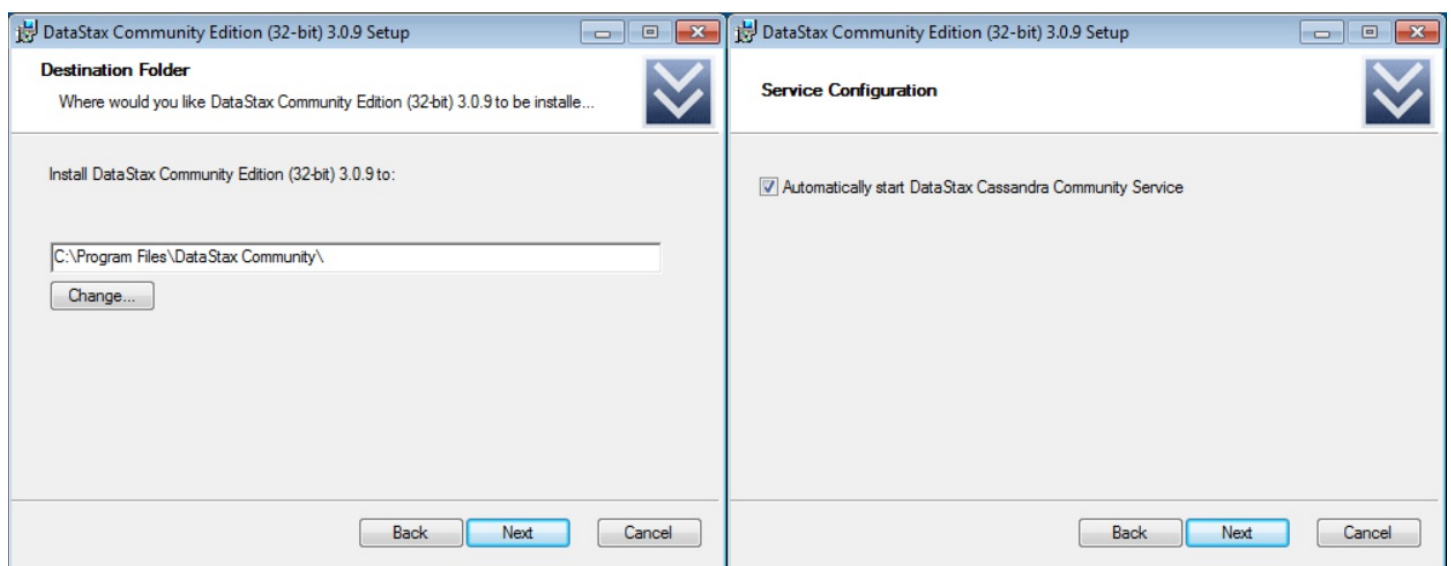
- MSI Installer (64 Bits):

http://downloads.datastax.com/community/datastax-community-64bit_3.0.9.msi

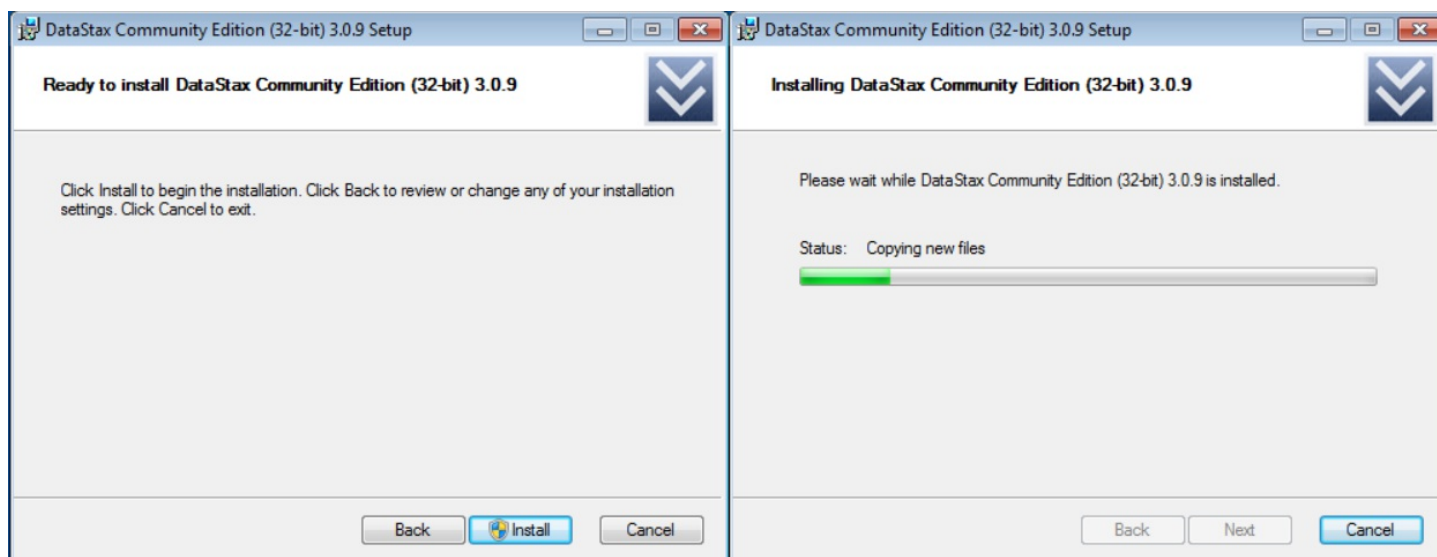
Una vez descargado, se corre el paquete de instalación, se aceptan los términos de licenciamiento.



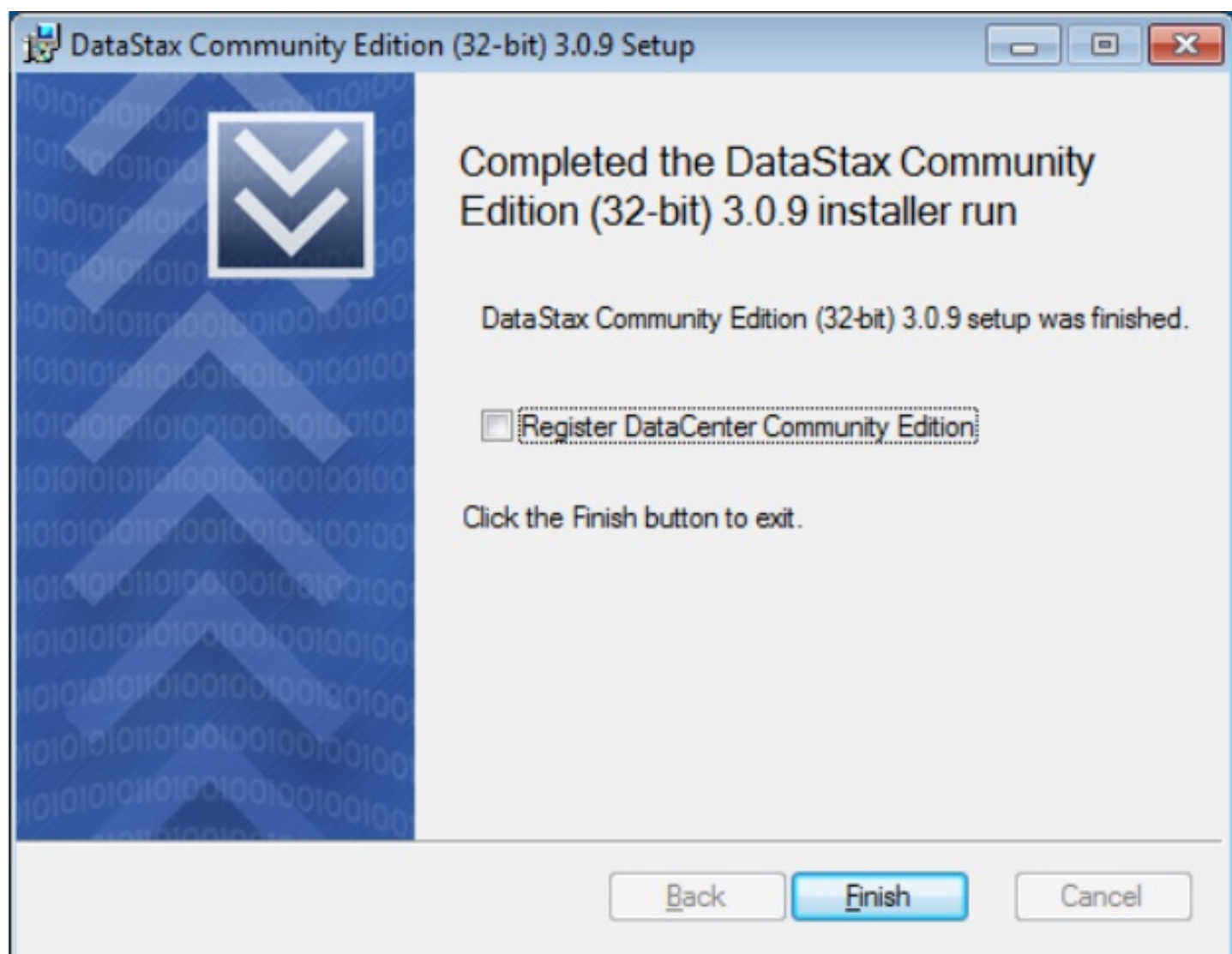
Se especifica en qué lugar se instalará Cassandra y se especifica si se desea que Cassandra se ejecute una vez encienda el equipo:



Se da inicio a la instalación:



Una vez termina aparecerá el aviso de que Cassandra se instaló correctamente:



Para comprobar si Cassandra está activo, se accede al **Símbolo de Sistema** y se ejecutan los comandos cqlsh.

```
> cqlsh
```

```
C:\Users\Dell>cqlsh  
  
WARNING: console codepage must be set to cp65001 to support utf-8 encoding on Windows platforms.  
If you experience encoding problems, change your console codepage with 'chcp 65001' before starting cqlsh.  
  
Connected to Test Cluster at 127.0.0.1:9042.  
[cqlsh 5.0.1 | Cassandra 3.0.9 | CQL spec 3.4.0 | Native protocol v4]  
Use HELP for help.  
WARNING: pyreadline dependency missing. Install to enable tab completion.  
cqlsh>
```

Linux: Ubuntu 16.04

Java: Se accede al siguiente link, y se ejecutan las siguientes líneas en la terminal de Linux.

<https://www.digitalocean.com/community/tutorials/how-to-install-java-with-apt-get-on-ubuntu-16-04#installing-the-oracle-jdk>

Instalar Oracle JDK

```
$ sudo add-apt-repository ppa:webupd8team/java  
$ sudo apt-get update
```

Oracle JDK 8

```
$ sudo apt-get install oracle-java8-installer
```

Installing the Oracle JDK

If you want to install the Oracle JDK, which is the official version distributed by Oracle, you will need to follow a few more steps.

First, add Oracle's PPA, then update your package repository.

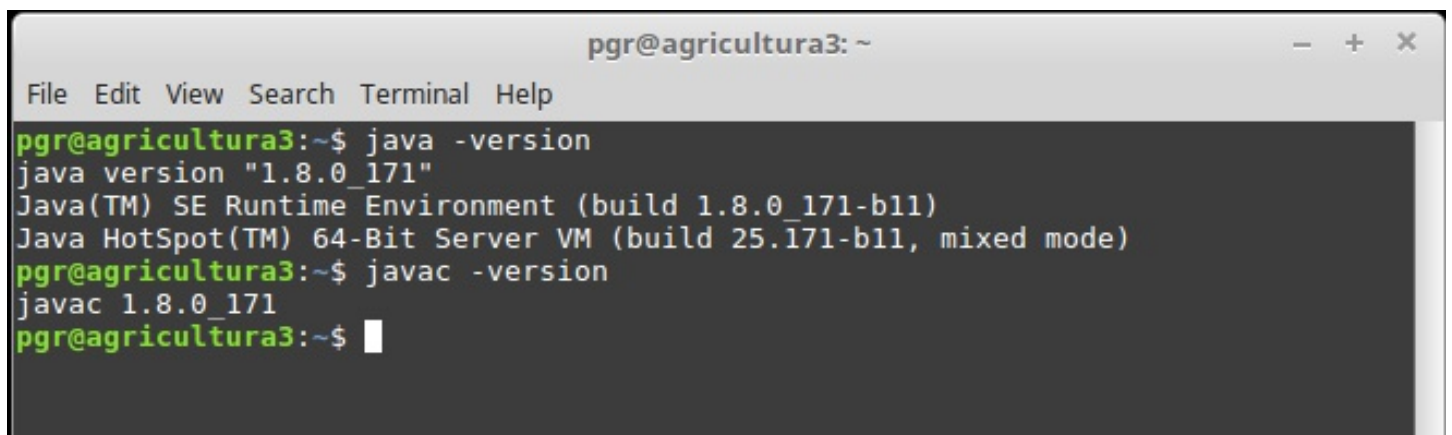
```
$ sudo add-apt-repository ppa:webupd8team/java
$ sudo apt-get update
```

Then, depending on the version you want to install, execute one of the following commands:

Oracle JDK 8

This is the latest stable version of Java at time of writing, and the recommended version to install. You can do so using the following command:

```
$ sudo apt-get install oracle-java8-installer
```

A screenshot of a terminal window titled 'pgr@agricultura3: ~'. The terminal shows the output of 'java -version' and 'javac -version' commands. The 'java -version' command outputs: 'java version "1.8.0_171"', 'Java(TM) SE Runtime Environment (build 1.8.0_171-b11)', and 'Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)'. The 'javac -version' command outputs: 'javac 1.8.0_171'. The prompt 'pgr@agricultura3:~\$' is visible at the end of each line.

```
pgr@agricultura3: ~
File Edit View Search Terminal Help
pgr@agricultura3:~$ java -version
java version "1.8.0_171"
Java(TM) SE Runtime Environment (build 1.8.0_171-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)
pgr@agricultura3:~$ javac -version
javac 1.8.0_171
pgr@agricultura3:~$
```

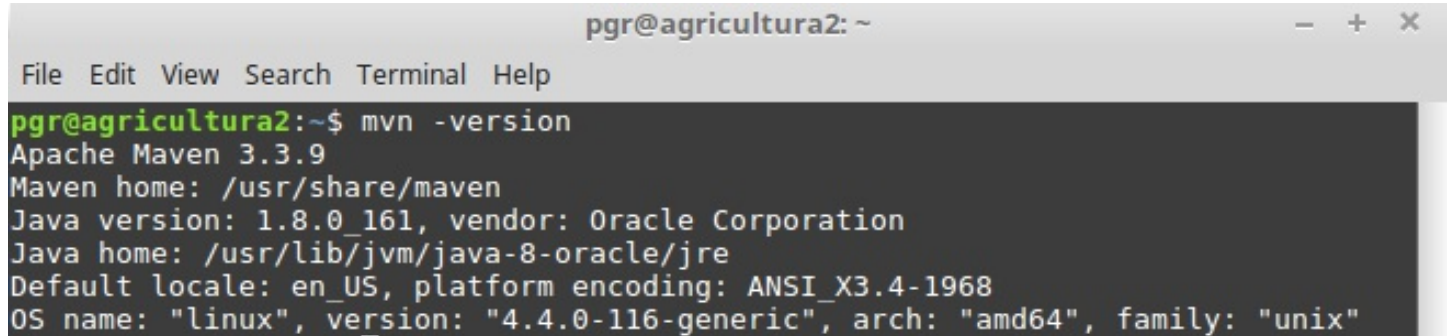
Maven: Para instalar maven usando la Terminal de linux, se escribe el siguiente comando.

```
$ sudo apt-get install maven
```

Una vez instalado, se verifica consultado la versión usando la siguiente

línea.

```
$ mvn -version
```

A terminal window titled 'pgr@agricultura2: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The command 'mvn -version' has been executed, resulting in the following output:

```
pgr@agricultura2:~$ mvn -version
Apache Maven 3.3.9
Maven home: /usr/share/maven
Java version: 1.8.0_161, vendor: Oracle Corporation
Java home: /usr/lib/jvm/java-8-oracle/jre
Default locale: en_US, platform encoding: ANSI_X3.4-1968
OS name: "linux", version: "4.4.0-116-generic", arch: "amd64", family: "unix"
```

Cassandra:

Lo primero que debemos hacer es verificar que en Ubuntu 16.04 esté instalado **“wget”** el cual es es una herramienta libre que permite la descarga de contenidos desde servidores web de una forma simple. El propósito de instalar **“wget”** será obtener el script de instalación de Cassandra que ofrece Thingsboard el cual agrega el repositorio de Cassandra al sistema para posteriormente actualizarlo con sus herramientas de trabajo.

Pasos:

1. Escribir el siguiente comando en la terminal para instalar curl en caso de no tenerlo sobre Ubuntu.

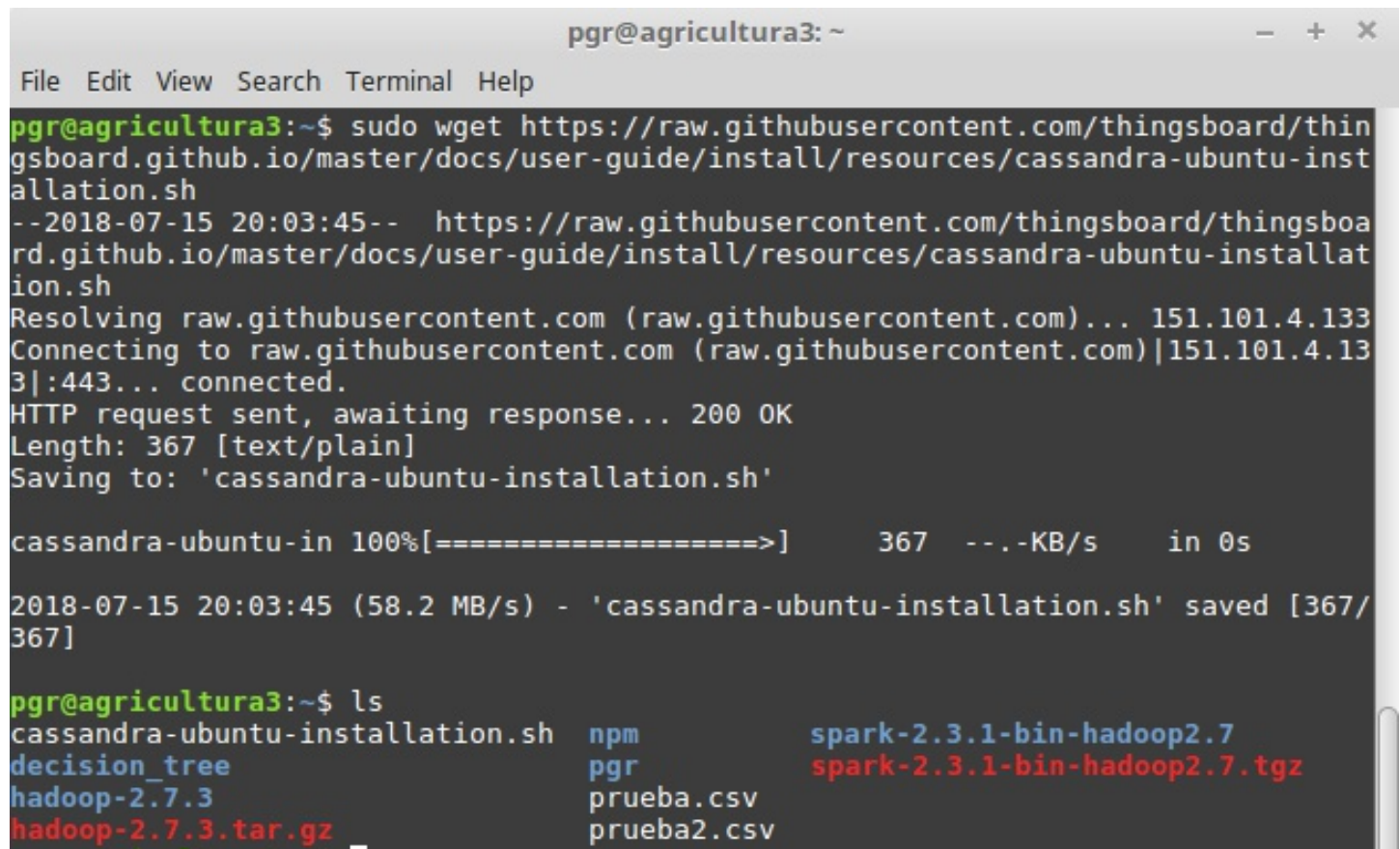
```
$ sudo apt-get install wget
```

2. Descargar el script de instalación de cassandra usando curl.

```
$ sudo wget https://raw.githubusercontent.com/thingsboard/thi
```

```
ngsboard.github.io/master/docs/user-guide/install/resources/cassandra-ubuntu-installation.sh
```

Verificar que el archivo esté en la carpeta donde está actualmente abierta la terminal.



The screenshot shows a terminal window titled 'pgr@agricultura3: ~'. The user runs the command `sudo wget https://raw.githubusercontent.com/thingsboard/thingsboard.github.io/master/docs/user-guide/install/resources/cassandra-ubuntu-installation.sh`. The terminal output shows the file being downloaded from the specified URL, resolving the domain, connecting to the IP 151.101.4.133, and saving the file as 'cassandra-ubuntu-installation.sh'. The download progress is shown as 100% complete. After the download, the user runs `ls`, and the terminal displays a list of files in the current directory, including the newly downloaded script and other files like 'decision_tree', 'hadoop-2.7.3', 'hadoop-2.7.3.tar.gz', 'npm', 'pgr', 'prueba.csv', 'prueba2.csv', 'spark-2.3.1-bin-hadoop2.7', and 'spark-2.3.1-bin-hadoop2.7.tgz'.

```
pgr@agricultura3:~$ sudo wget https://raw.githubusercontent.com/thingsboard/thingsboard.github.io/master/docs/user-guide/install/resources/cassandra-ubuntu-installation.sh
--2018-07-15 20:03:45-- https://raw.githubusercontent.com/thingsboard/thingsboard.github.io/master/docs/user-guide/install/resources/cassandra-ubuntu-installation.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.4.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.4.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 367 [text/plain]
Saving to: 'cassandra-ubuntu-installation.sh'

cassandra-ubuntu-in 100%[=====>] 367 --.-KB/s in 0s

2018-07-15 20:03:45 (58.2 MB/s) - 'cassandra-ubuntu-installation.sh' saved [367/367]

pgr@agricultura3:~$ ls
cassandra-ubuntu-installation.sh  npm          spark-2.3.1-bin-hadoop2.7
decision_tree                    pgr          spark-2.3.1-bin-hadoop2.7.tgz
hadoop-2.7.3                    prueba.csv
hadoop-2.7.3.tar.gz             prueba2.csv
```

Dar permisos de ejecución del script usando el comando

```
$ sudo chmod 777 cassandra-ubuntu-installation.sh
```

Ejecutar script de instalación de Cassandra.

```
$ sudo ./cassandra-ubuntu-installation.sh
```


Verificar Cassandra en Linux usando el comando cqlsh.

```
$ cqlsh
```



```
pgr@agricultura2: ~  
File Edit View Search Terminal Help  
pgr@agricultura2:~$ cqlsh  
Connected to Test Cluster at 127.0.0.1:9042.  
[cqlsh 5.0.1 | Cassandra 3.11.2 | CQL spec 3.4.4 | Native protocol v4]  
Use HELP for help.  
cqlsh>
```

MongoDB: Para realizar la instalación de MongoDB usaremos como guía base el siguiente link que realiza la instalación de MongoDB en Ubuntu 16.04.

<https://www.digitalocean.com/community/tutorials/como-instalar-mongodb-en-ubuntu-16-04-es>

Como primera instancia, esta información es útil conocerla antes de instalar MongoDB.

MongoDB está actualmente incluido en el repositorio de paquetes de Ubuntu, pero el repositorio oficial de MongoDB proporciona la versión más actualizada y es el camino recomendado para instalar este software.

Pasos:

Ubuntu se asegura de autenticar los paquetes de software verificando que han sido firmados con llaves GPG, así que primero importamos la llave para el repositorio oficial de MongoDB.


```
$ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80  
--recv EA312927
```

A continuación, debemos agregar los detalles del repositorio de Mongo de tal manera que apt pueda saber de donde descargar los paquetes. Corriendo el siguiente comando crearemos la lista para MongoDB.

```
$ echo "deb http://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/3.2 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.2.list
```

Después de agregar los detalles del repositorio, debemos actualizar la lista de paquetes.

```
$ sudo apt-get update
```

Ahora podemos instalar el propio paquete de MongoDB.

```
$ sudo apt-get install -y mongodb-org
```

Vamos a crear un archivo de unidad para administrar el servicio de MongoDB. Crearemos un archivo de configuración llamado `mongodb.service` en el directorio `/etc/systemd/system` utilizando `nano` o su editor de texto favorito.

```
$ sudo nano /etc/systemd/system/mongodb.service
```

Pegamos el siguiente contenido en el archivo.

/etc/systemd/system/mongodb.service

```
[Unit]
```

```
Description=High-performance, schema-free document-oriented database
```

```
After=network.target
```

```
[Service]
```

```
User=mongodb
```

```
ExecStart=/usr/bin/mongod --quiet --config /etc/mongod.conf
```

```
[Install]
```

```
WantedBy=multi-user.target
```

pgr@agricultura3: ~

File Edit View Search Terminal Help

```
pgr@agricultura3:~$ ls /etc/systemd/system
```

```
dbus-org.freedesktop.resolve1.service  network-pre.target.wants
default.target.wants                    open-vm-tools.service.requires
final.target.wants                      paths.target.wants
getty.target.wants                      sockets.target.wants
graphical.target.wants                  sshd.service
iscsi.service                           sysinit.target.wants
mongodb.service                         syslog.service
multi-user.target.wants                  timers.target.wants
```

```
pgr@agricultura3:~$
```

```
pgr@agricultura3: ~  
File Edit View Search Terminal Help  
GNU nano 2.8.6 File: /etc/systemd/system/mongodb.service  
[Unit]  
Description=High-performance, schema-free document-oriented database  
After=network.target  
[Service]  
User=mongodb  
ExecStart=/usr/bin/mongod --quiet --config /etc/mongod.conf  
[Install]  
WantedBy=multi-user.target
```

Lo siguiente, será iniciar el servicio recién creado con systemctl.

```
$ sudo systemctl start mongodb
```

Aun cuando este comando no responde con un mensaje, puede utilizar systemctl para revisar que el servicio ha arrancado apropiadamente.

```
$ sudo systemctl status mongodb
```

El último paso es habilitar automáticamente el arranque de MongoDB cuando el sistema inicie.

```
$ sudo systemctl enable mongodb
```

Permitir acceso a puertos de otros servidores

En la mayoría de los casos, MongoDB solo debería ser accesible desde ubicaciones seguras, como por ejemplo el otro servidor que aloja la

aplicación. Para cumplir con esta tarea, puede permitir a una IP de un servidor específico acceder y conectarse al puerto por defecto de MongoDB.

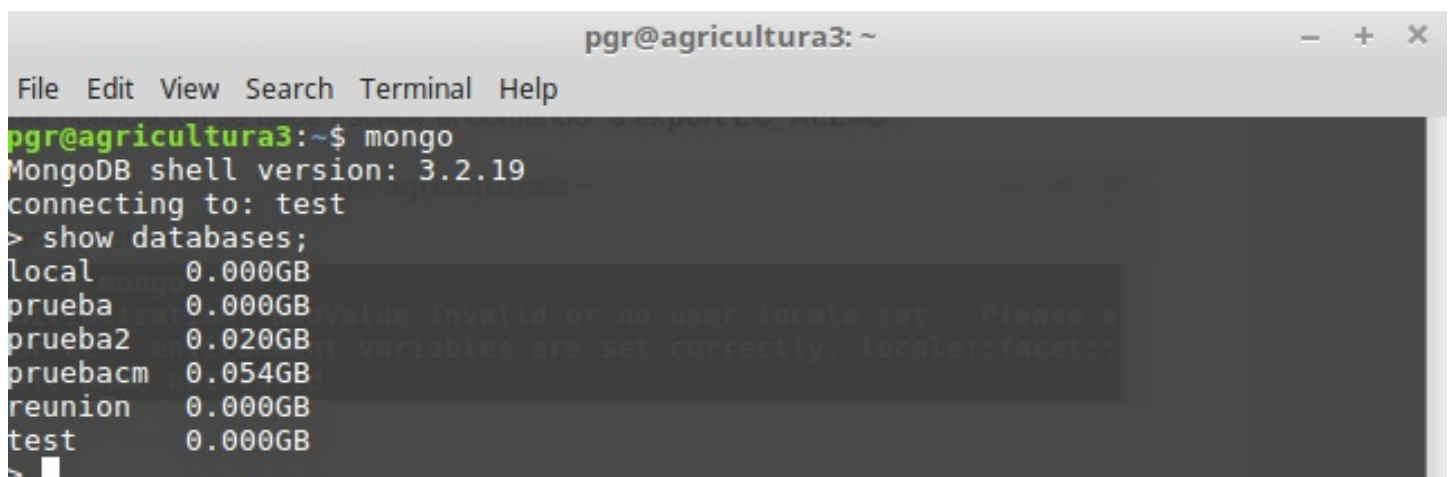
```
$ sudo ufw allow from la_IP_del_otro_servidor/32 to any port 27017
```

Puede verificar el cambio en la configuración del firewall con ufw.

```
$ sudo ufw status
```

Para comprobar que mongo está corriendo en la máquina, se verifica usando el siguiente comando.

```
$ mongo
```



The screenshot shows a terminal window titled 'pgr@agricultura3: ~'. The terminal output is as follows:

```
pgr@agricultura3:~$ mongo
MongoDB shell version: 3.2.19
connecting to: test
> show databases;
local          0.000GB
prueba         0.000GB
prueba2        0.020GB
pruebacm       0.054GB
reunion        0.000GB
test           0.000GB
>
```

Problemas comunes:

Si aparece la siguiente notificación se debe escribir el comando **\$ export LC_ALL=C**.

File Edit View Search Terminal Help

pgr@agricultura3:~\$ mongo

Failed global initialization: BadValue Invalid or no user locale set. Please ensure LANG and/or LC_* environment variables are set correctly. locale::facet::_S_create_c_locale name not valid

Agregar archivo para despliegue de nodos Apache Spark

En cada equipo que conformará el cluster de Spark se debe tener el siguiente archivo:

Utilizando la siguiente dirección <http://apache.uniminuto.edu/spark/spark-2.3.1/spark-2.3.1-bin-hadoop2.7.tgz> se descarga el archivo que se obtiene usando el comando wget.

```
$ wget http://apache.uniminuto.edu/spark/spark-2.3.1/spark-2.3.1-bin-hadoop2.7.tgz
```

The image shows three terminal windows from different hosts in a cluster:

- Top Left (pgr@agricultura1: ~/pgr):** Shows the command `ls` and a directory listing including `certificados`, `gatling-charts-highcharts-bundle-2.3.1`, `gatling-mqtt`, `performance-tests`, `spark-2.3.1-bin-hadoop2.7`, `spark-2.3.1-bin-hadoop2.7.tgz`, and `thingsboard`.
- Top Right (pgr@agricultura3: ~):** Shows the command `wget http://apache.uniminuto.edu/spark/spark-2.3.1/spark-2.3.1-bin-hadoop2.7.tgz` being executed. It displays the progress of the download, showing it is 100% complete (215.42M, 9.00MB/s, in 22s).
- Bottom (pgr@agricultura2: ~):** Shows the command `ls` and a directory listing including `MqttSimulation.scala`, `cassandra-ubuntu-installation.sh`, `decision_tree`, `pgr`, `python_scripts`, `redis-stable`, `redis-stable.tar.gz`, `scriptCassandra`, `spark-2.3.1-bin-hadoop2.7`, `spark-2.3.1-bin-hadoop2.7.tgz`, `spark-prueba`, `spark-warehouse`, `thingsboard`, and `thingsboard-spark-backend`.

Posteriormente se debe descomprimir. (Leer manual técnico para ver la configuración de Spark).

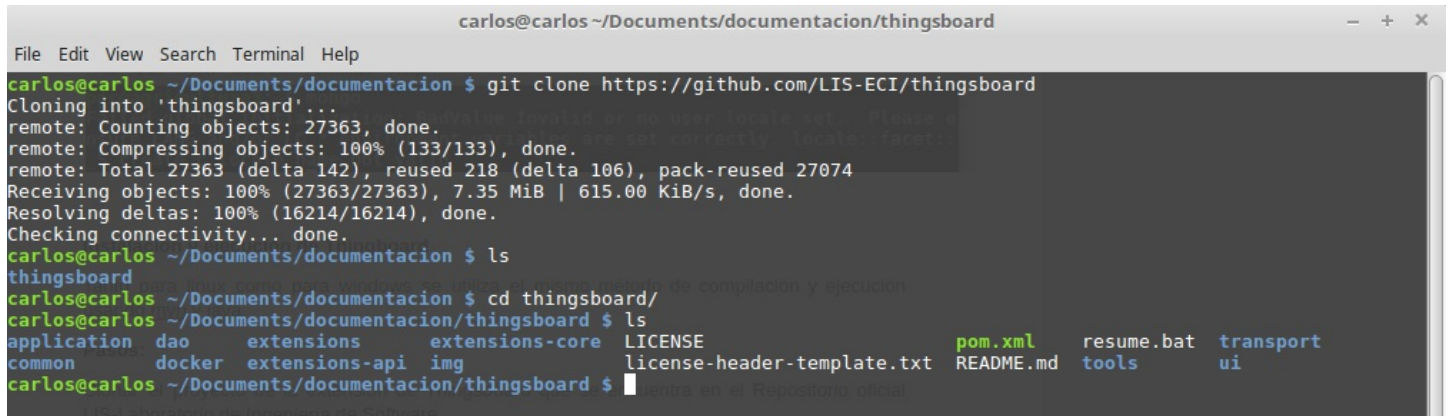
Instalación y ejecución de Thingboard

Tanto para linux como para windows se utiliza el mismo método de compilación y ejecución usando mvn y java.

Pasos:

Clonar el proyecto de la extensión de Thingsboard que se encuentra en el Repositorio oficial LIS-Laboratorio de Ingeniería de Software.

```
$ git clone https://github.com/LIS-ECI/thingsboard
```



```
carlos@carlos ~/Documents/documentacion/thingsboard
File Edit View Search Terminal Help
carlos@carlos ~/Documents/documentacion $ git clone https://github.com/LIS-ECI/thingsboard
Cloning into 'thingsboard'...
remote: Counting objects: 27363, done.
remote: Compressing objects: 100% (133/133), done.
remote: Total 27363 (delta 142), reused 218 (delta 106), pack-reused 27074
Receiving objects: 100% (27363/27363), 7.35 MiB | 615.00 KiB/s, done.
Resolving deltas: 100% (16214/16214), done.
Checking connectivity... done.
carlos@carlos ~/Documents/documentacion $ ls
thingsboard
carlos@carlos ~/Documents/documentacion $ cd thingsboard/
carlos@carlos ~/Documents/documentacion/thingsboard $ ls
application  dao  extensions  extensions-core  LICENSE  pom.xml  resume.bat  transport
common       docker  extensions-api  img  license-header-template.txt  README.md  tools  ui
carlos@carlos ~/Documents/documentacion/thingsboard $
```

Lo primero será compilar el proyecto, por lo que debemos localizar la raíz del proyecto y ejecutarlo completamente con la siguiente línea:

```
$ mvn clean install -DskipTests -Dlicense.skip=true
```

Omitiremos la compilación de las pruebas y la revisión de licencias de

maven

```
carlos@carlos ~/Documents/documentacion/thingsboard
File Edit View Search Terminal Help
carlos@carlos ~/Documents/documentacion/thingsboard $ mvn clean install -DskipTests -Dlicense.skip=true
[INFO] Scanning for projects...
[WARNING]
[WARNING] Some problems were encountered while building the effective model for org.thingsboard:application:jar:1.4.1-SNAPSHOT
[WARNING] 'dependencies.dependency.(groupId:artifactId:type:classifier)' must be unique: org.thingsboard.extensions:extension-kafka:jar -> duplicate declaration of version ${project.version} @ org.thingsboard:application:[unknown-version], /home/carlos/Documents/documentacion/thingsboard/application/pom.xml, line 274, column 15
[WARNING]
[WARNING] It is highly recommended to fix these problems because they threaten the stability of your build.
[WARNING]
[WARNING] For this reason, future Maven versions might no longer support building such malformed projects.
[WARNING]
[INFO] -----
[INFO] Detecting the operating system and CPU architecture
[INFO] -----
[INFO] os.detected.name: linux
[INFO] os.detected.arch: x86_64
[INFO] os.detected.version: 4.8
[INFO] os.detected.version.major: 4
[INFO] os.detected.version.minor: 8
[INFO] os.detected.release: linuxmint
[INFO] os.detected.release.version: 18.2
[INFO] os.detected.release.like.linuxmint: true
```

Para saber si la compilación terminó correctamente, deberá aparecer cada uno de los paquetes mostrando una compilada satisfactoria.

```
carlos@carlos ~/Documents/documentacion/thingsboard
File Edit View Search Terminal Help
[INFO] Thingsboard Server Extensions API ..... SUCCESS [ 1.725 s]
[INFO] Thingsboard Server Core Extensions ..... SUCCESS [ 11.388 s]
[INFO] Thingsboard Extensions ..... SUCCESS [ 0.042 s]
[INFO] Thingsboard Server RabbitMQ Extension ..... SUCCESS [ 6.133 s]
[INFO] Thingsboard Server REST API Call Extension ..... SUCCESS [ 13.902 s]
[INFO] Thingsboard Server Kafka Extension ..... SUCCESS [ 11.721 s]
[INFO] Thingsboard Server MQTT Extension ..... SUCCESS [ 9.327 s]
[INFO] Thingsboard Server SQS Extension ..... SUCCESS [ 10.653 s]
[INFO] Thingsboard Server SNS Extension ..... SUCCESS [ 8.801 s]
[INFO] Thingsboard Server Transport Modules ..... SUCCESS [ 0.533 s]
[INFO] Thingsboard HTTP Transport ..... SUCCESS [ 8.701 s]
[INFO] Thingsboard COAP Transport ..... SUCCESS [ 3.896 s]
[INFO] Thingsboard MQTT Transport ..... SUCCESS [ 2.246 s]
[INFO] Thingsboard Server UI ..... SUCCESS [22:14 min]
[INFO] Thingsboard Server Tools ..... SUCCESS [01:10 min]
[INFO] Thingsboard Server Application ..... SUCCESS [03:59 min]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 29:45 min
[INFO] Finished at: 2018-07-15T21:04:46-05:00
[INFO] Final Memory: 76M/474M
[INFO] -----
carlos@carlos ~/Documents/documentacion/thingsboard $
```

Posteriormente debemos instalar el esquema de base de datos de Cassandra por lo que cambiará si está en sistema operativos Windows o Ubuntu.

Windows:

Sobre la raíz del proyecto debemos ingresar con el “símbolo del sistema”, nos dirigimos a la siguiente ruta “application\target\windows\” con el siguiente comando:

```
> cd application/target/windows
```

Luego, se ejecuta el script

```
> install_dev_db.bat
```

```
C:\Users\Dell\Desktop\NovenoSemestre\PGR1\thingsboard\application\target\windows>install_dev_db.bat
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/C:/Users/Dell/Desktop/NovenoSemestre/PGR1/thingsboard/application/target/extensions/extension-sns-1.4.1-SNAPSHOT-extension.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/C:/Users/Dell/Desktop/NovenoSemestre/PGR1/thingsboard/application/target/extensions/extension-sqs-1.4.1-SNAPSHOT-extension.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/C:/Users/Dell/Desktop/NovenoSemestre/PGR1/thingsboard/application/target/thingsboard-1.4.1-SNAPSHOT-bootstrap.jar!/BOOT-INF/lib/logback-classic-1.2.3.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [ch.qos.logback.classic.util.ContextSelectorStaticBinder]
=====
:: ThingsBoard ::      (v1.4.1-SNAPSHOT)
=====

Starting ThingsBoard Installation...
Installing DataBase schema...
Installing Cassandra DataBase schema...
Loading system data...
Loading demo data...
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Installation finished successfully!
ThingsBoard DB installed successfully!
```

Linux:

Dentro del proyecto Thingsboard, nos dirigimos a la carpeta

“application/target/bin/install”, allí, encontraremos el archivo

“install_dev_db.sh”, necesitamos darle los permisos correspondientes por lo que usaremos la siguiente línea:

```
carlos@carlos ~/Documents/documentacion/thingsboard/application/target/bin/install - + x
File Edit View Search Terminal Help
carlos@carlos ~/Documents/documentacion/thingsboard/application/target/bin/install $ ls
install_dev_db.sh install.sh logback.xml upgrade.sh
carlos@carlos ~/Documents/documentacion/thingsboard/application/target/bin/install $ chmod
777 install_dev_db.sh
carlos@carlos ~/Documents/documentacion/thingsboard/application/target/bin/install $
```

Posteriormente, se debe editar el archivo cambiando el “**run_user**” con el nombre del equipo, para este ejemplo será carlos.

```
carlos@carlos ~/Documents/documentacion/thingsboard/application/target/bin/install - + x
File Edit View Search Terminal Help
GNU nano 2.5.3 File: install_dev_db.sh Modified
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
BASE=/home/carlos/Documents/documentacion/thingsboard/application/target
CONF_FOLDER=${BASE}/conf
jarfile="${BASE}/thingsboard-1.4.1-SNAPSHOT-boot.jar"
installDir=${BASE}/data
loadDemo=true
# el archivo cambiando el "run_user" con el nombre del equipo.

export JAVA_OPTS="$JAVA_OPTS -Dplatform=@pkg.platform@"
export LOADER_PATH=${BASE}/conf,${BASE}/extensions
export SQL_DATA_FOLDER=${SQL_DATA_FOLDER:-/tmp}

run_user=carlos

su -s /bin/sh -c "java -cp ${jarfile} $JAVA_OPTS -Dloader.main=org.thingsboard.server.Thin$
-Dinstall.data_dir=${installDir} \
-Dinstall.load_demo=${loadDemo} \
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Linter ^ Go To Line
```

Se guardarán los cambios y se procede a ejecutar el script usando la siguiente línea:


```

carlos@carlos ~/Documents/documentacion/thingsboard/application/target/bin/install $ ./install_dev_db.sh
Password:
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/carlos/Documents/documentacion/thingsboard/application/target/extensions/extension-sns-1.4.1-SNAPSHOT-extension.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/carlos/Documents/documentacion/thingsboard/application/target/extensions/extension-sqs-1.4.1-SNAPSHOT-extension.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/carlos/Documents/documentacion/thingsboard/application/target/thingsboard-1.4.1-SNAPSHOT-boot.jar!/BOOT-INF/lib/logback-classic-1.2.3.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [ch.qos.logback.classic.util.ContextSelectorStaticBinder]
=====
:: ThingsBoard :: (v1.4.1-SNAPSHOT)
=====

Starting ThingsBoard Installation...
Installing DataBase schema...
Installing Cassandra DataBase schema...
Loading system data...
Loading demo data...
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Dispositivo: null
Installation finished successfully!
ThingsBoard DB installed successfully!

```

Ejecución:

Sobre el proyecto ThingsBoard, acceder a la carpeta

“application/target”, listamos los archivos y se debe encontrar un ejecutable llamado **“thingsboard-1.4.1-SNAPSHOT-boot.jar”**:

```

carlos@carlos ~/Documents/documentacion/thingsboard/application/target
File Edit View Search Terminal Help
carlos@carlos ~/Documents/documentacion/thingsboard/application/target $ ls
archive-tmp  generated-sources  thingsboard-1.4.1-SNAPSHOT.jar
bin          generated-test-sources  thingsboard.changes
classes      maven-archiver      thingsboard.deb
conf         protoc-3.0.2-linux-x86_64.exe  thingsboard.rpm
control      protoc-dependencies  thingsboard-windows.zip
data         protoc-plugins      tmp
debian       test-classes        windows
extensions   thingsboard-1.4.1-SNAPSHOT-boot.jar
carlos@carlos ~/Documents/documentacion/thingsboard/application/target $

```

Para ejecutar el archivo jar se debe escribir la siguiente línea:

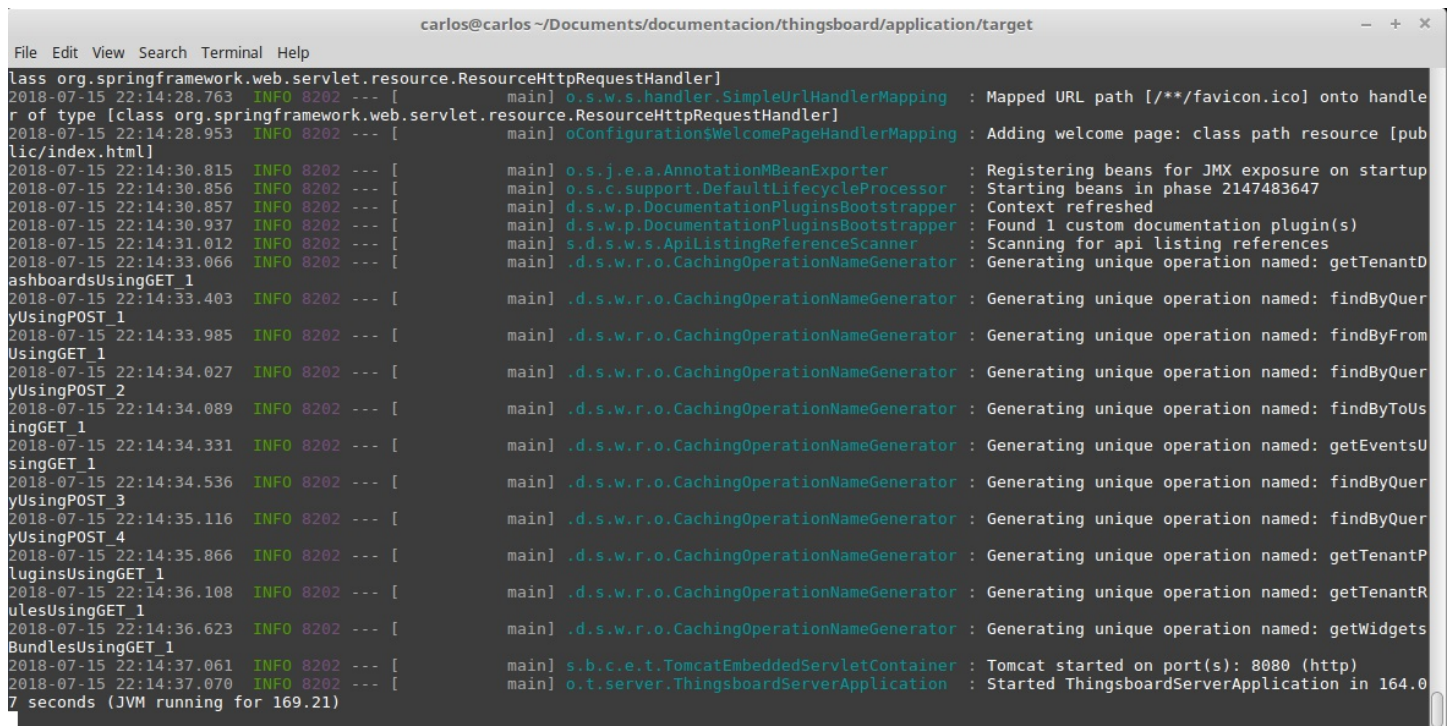
Linux

```
$ java -jar thingsboard-1.4.1-SNAPSHOT-boot.jar
```

Windows

```
thingsboard\application\target> java -jar thingsboard-1.4.1-SNAPSHOT-boot.jar
```

La terminal deberá quedar en el siguiente estado para acceder desde cualquier navegador **(RECOMENDACIÓN GOOGLE CHROME)**:



```
carlos@carlos ~/Documents/documentacion/thingsboard/application/target
File Edit View Search Terminal Help
[...]  
2018-07-15 22:14:28.763 INFO 8202 --- [main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/**/favicon.ico] onto handler of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]  
2018-07-15 22:14:28.953 INFO 8202 --- [main] oConfiguration$WelcomePageHandlerMapping : Adding welcome page: class path resource [public/index.html]  
2018-07-15 22:14:30.815 INFO 8202 --- [main] o.s.j.e.a.AnnotationMBeanExporter : Registering beans for JMX exposure on startup  
2018-07-15 22:14:30.856 INFO 8202 --- [main] o.s.c.support.DefaultLifecycleProcessor : Starting beans in phase 2147483647  
2018-07-15 22:14:30.857 INFO 8202 --- [main] d.s.w.p.DocumentationPluginsBootstrapper : Context refreshed  
2018-07-15 22:14:30.937 INFO 8202 --- [main] d.s.w.p.DocumentationPluginsBootstrapper : Found 1 custom documentation plugin(s)  
2018-07-15 22:14:31.012 INFO 8202 --- [main] s.d.s.w.s.ApiListingReferenceScanner : Scanning for api listing references  
2018-07-15 22:14:33.066 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: getTenantDetails  
2018-07-15 22:14:33.403 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: findByQueryUsingPOST_1  
2018-07-15 22:14:33.985 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: findByFromUsingGET_1  
2018-07-15 22:14:34.027 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: findByQueryUsingPOST_2  
2018-07-15 22:14:34.089 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: findByToUsingGET_1  
2018-07-15 22:14:34.331 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: getEventsUsingGET_1  
2018-07-15 22:14:34.536 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: findByQueryUsingPOST_3  
2018-07-15 22:14:35.116 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: findByQueryUsingPOST_4  
2018-07-15 22:14:35.866 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: getTenantPluginsUsingGET_1  
2018-07-15 22:14:36.108 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: getTenantRulesUsingGET_1  
2018-07-15 22:14:36.623 INFO 8202 --- [main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: getWidgetsBundlesUsingGET_1  
2018-07-15 22:14:37.061 INFO 8202 --- [main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 8080 (http)  
2018-07-15 22:14:37.070 INFO 8202 --- [main] o.t.server.ThingsboardServerApplication : Started ThingsboardServerApplication in 164.07 seconds (JVM running for 169.21)
```

Una vez ingresamos al navegador accediendo desde la url “localhost:8080” se deberá ingresar el siguiente usuario:

Username: tenant@thingsboard.org

Password: tenant

