Índice

Creación y configuración de sitios virtuales.

Instalación de aplicaciones en entornos de prueba.

Instalación de aplicaciones Web en máquinas virtuales.

Configuración del servidor de aplicaciones.

Integración continua de código mediante Jenkins.

Despliegue de WAR's mediante consola.

Despliegue de aplicaciones en Tomcat/Weblogic.

Evaluación de la calidad del código mediante Sonar.

Empezaremos viendo cómo crear la máquina virtual donde vamos a instalar y configurar el sitio virtual.

Necesitaremos tener instalado Oracle VM VirtualBox.

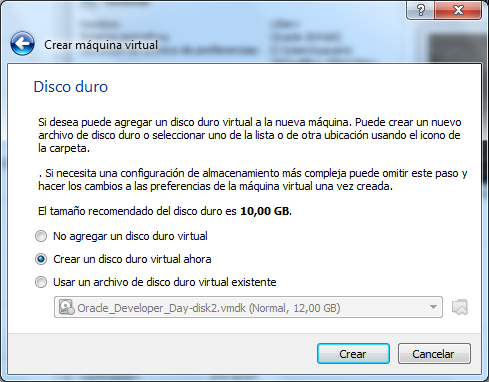
## Una vez lo tengamos, descargaremos la útlima versión de Ubuntu, en mi caso será Ubuntu Server 18.04.3 LTS.

## Le daremos a "Nueva" y crearemos nuestra maquina.

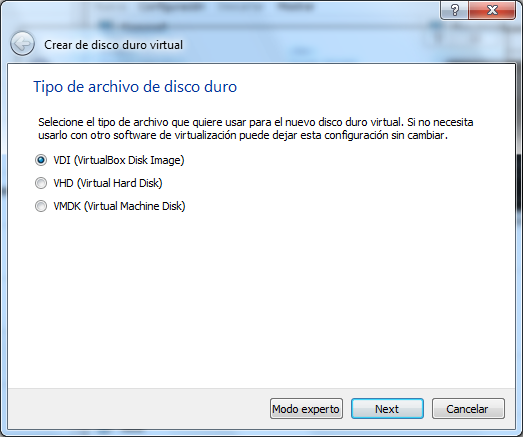
## El nombre de la máquina es a vuestro gusto.

## Nombre_maquina.PNG

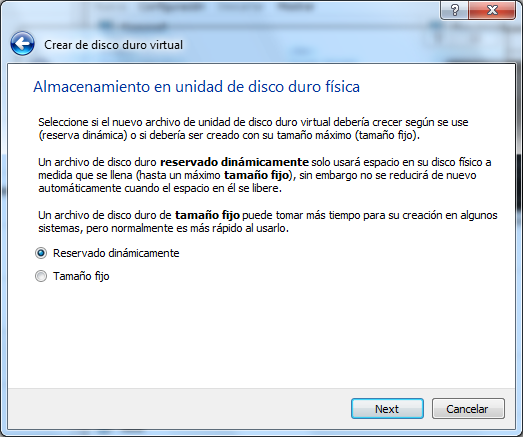
El disco que crearemos será virtual.



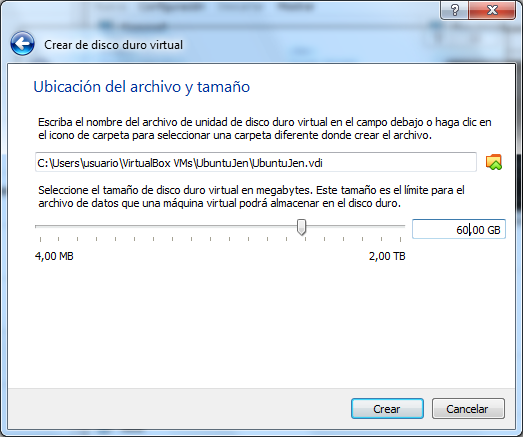
El tipo de archivo será VDI.



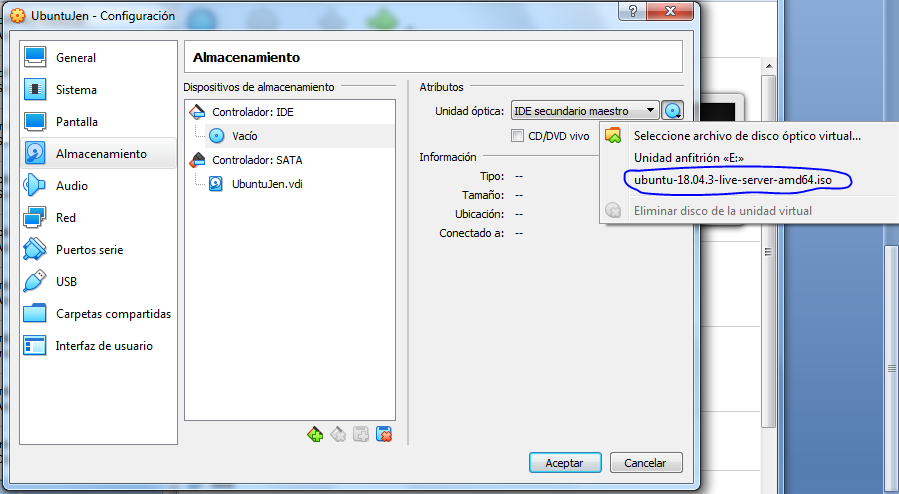
El almacenamiento será dinámico.



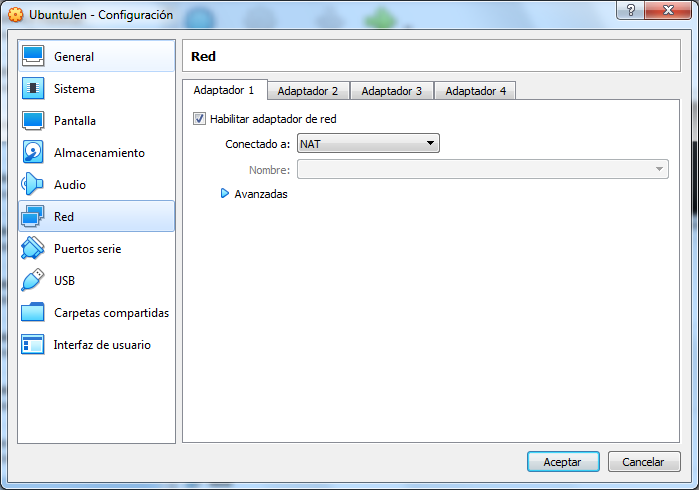
Para asegurarnos de que no nos quedamos sin espacio a la hora de instalar y configurar los distintos servicios, le daremos 60 GB de tamaño.



Al aceptar, ya tendremos nuestra máquina creada, solo falta meterle nuestra imagen de ubuntu y ver la red que va a tener.



La red que le pondremos será NAT.



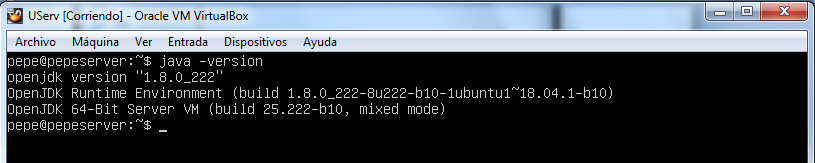
Una vez hecho esto, iniciamos la máquina y hacemos la instalación de Ubuntu.

Cuando la tengamos preparada, empezaremos con los servicios para poder instalar nuestro sitio web.

Para empezar necesitaremos Java 8.

sudo apt install openjdk-8-jdk

java -version



Ahora empezaremos a instalar nuestro sitio web y a configurarlo.

wget no-check-certificate -v -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add -

sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'

sudo apt-get update && sudo apt-get install jenkins

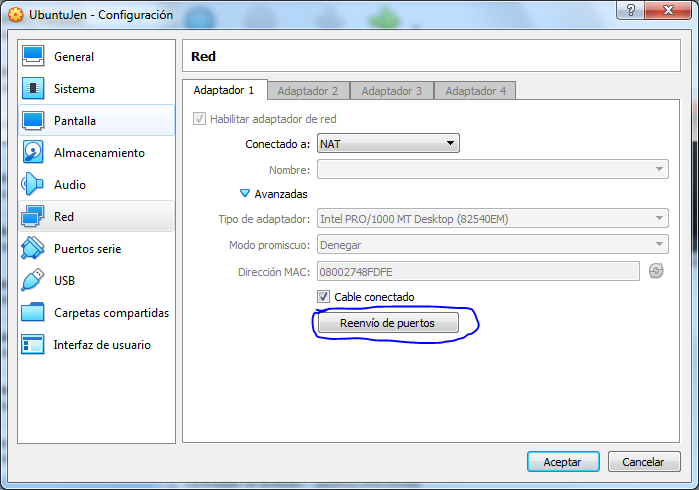
Una vez hecho eso, procedemos a configurar Jenkins.

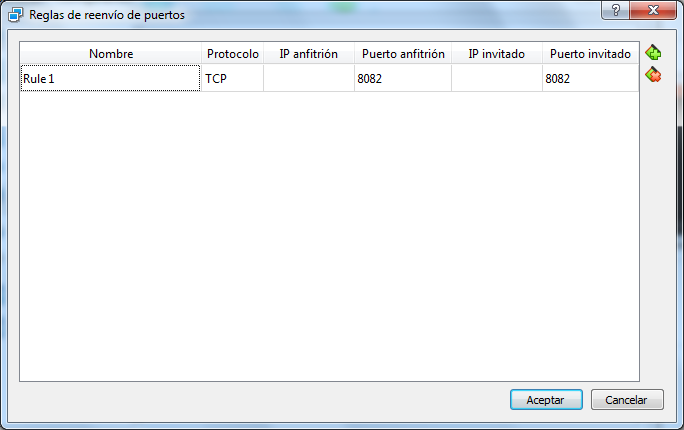
sudo nano /etc/default/jenkins

Dentro de este archivo modificamos la linea HTTP\_PORT indicándole el puerto 8082

10.PNG

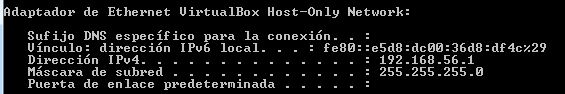
Antes de nada, apagaremos la máquina y le añadimos dicho puerto.



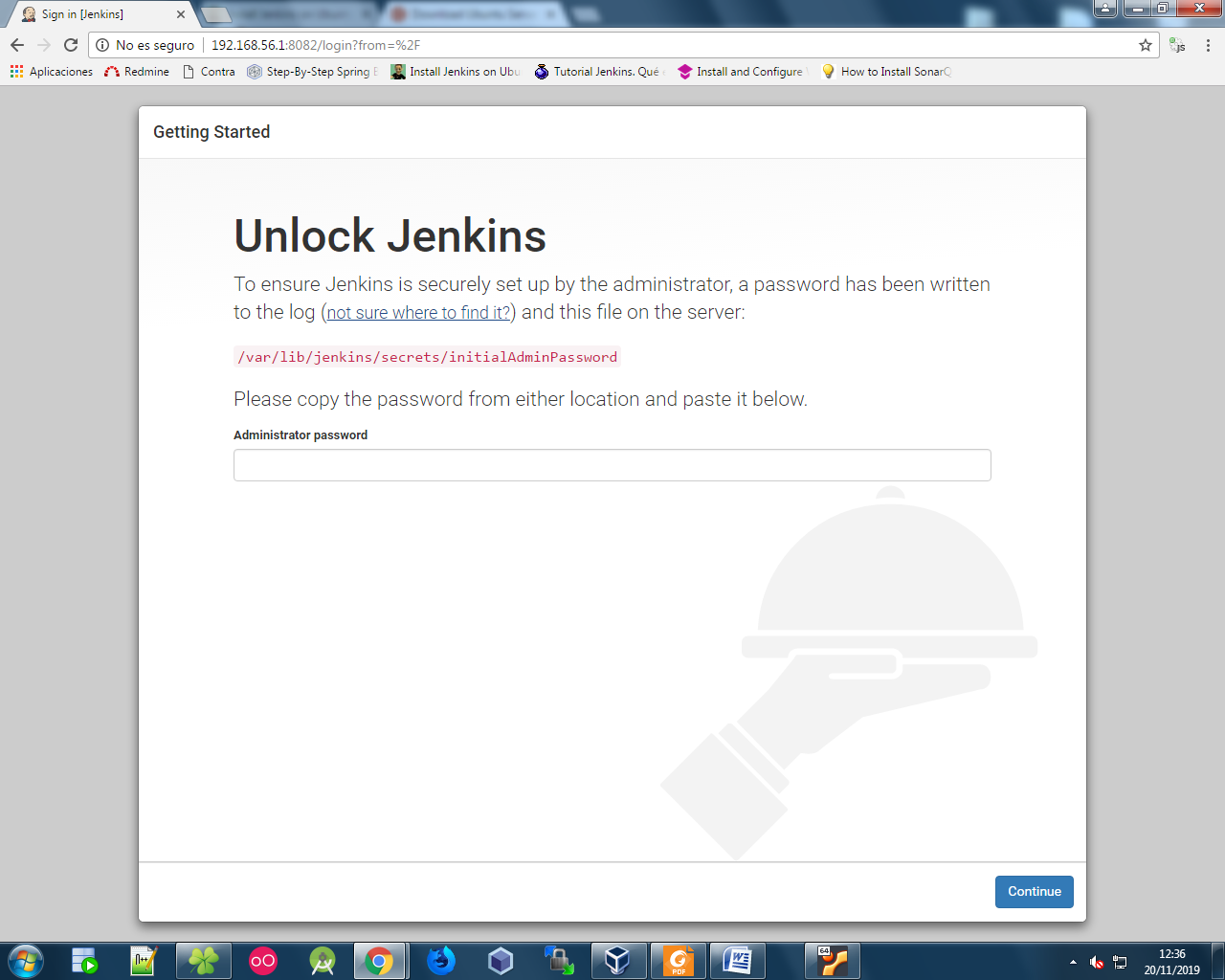


Ahora que tenmos instalado y configurado Jenkins, es hora de probar que funcione, en nuestro navegador de la máquina física tendremos que poner la dirección IP que nos ofrece la tarjeta de red de VirtualBox en nuestro pc junto con el puerto al que queremos acceder.

En mi caso tengo la 192.168.56.1



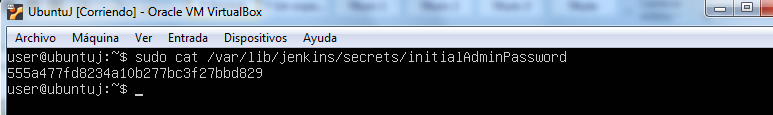
Ahora en el navegador pondremos 192.168.56.1:8082 y nos deberá mostrar esta pantalla.



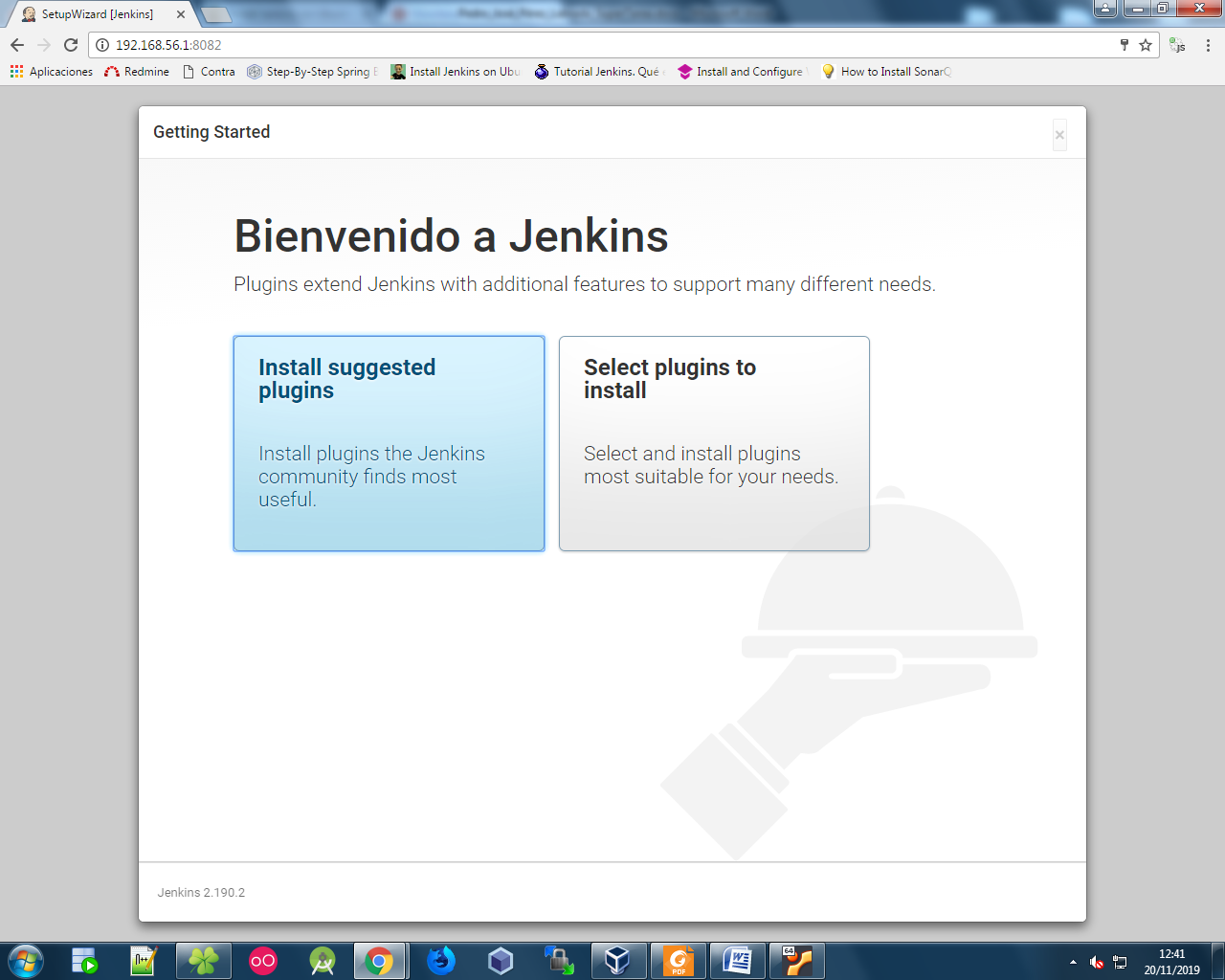
A continuación tendremos que abrir el fichero que está en la ruta que nos indica la página.

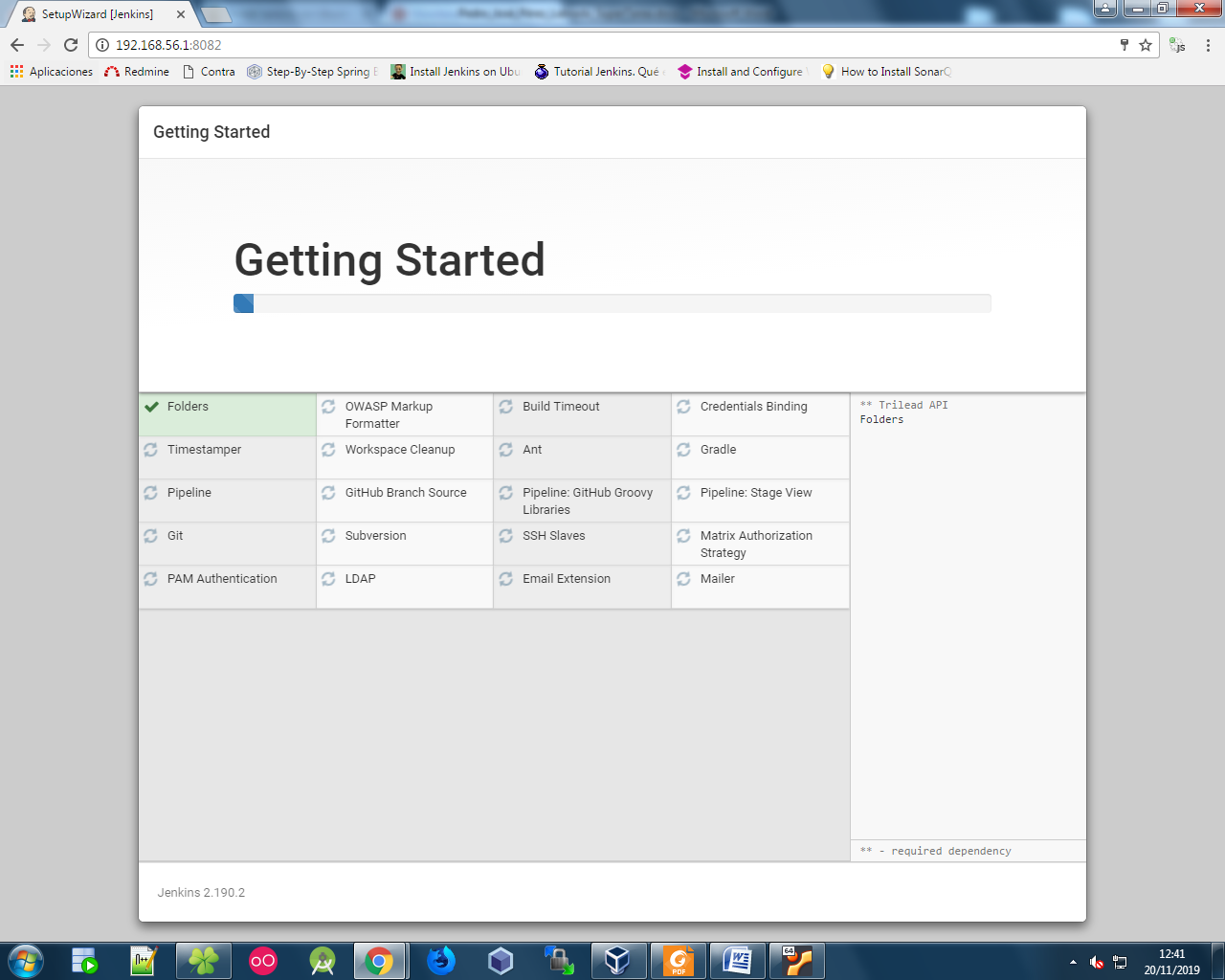
sudo cat /var/lib/jenkins/secrets/initialAdminPassword

y escribimos en ese rectángulo la contraseña que nos ofrece dicho fichero.

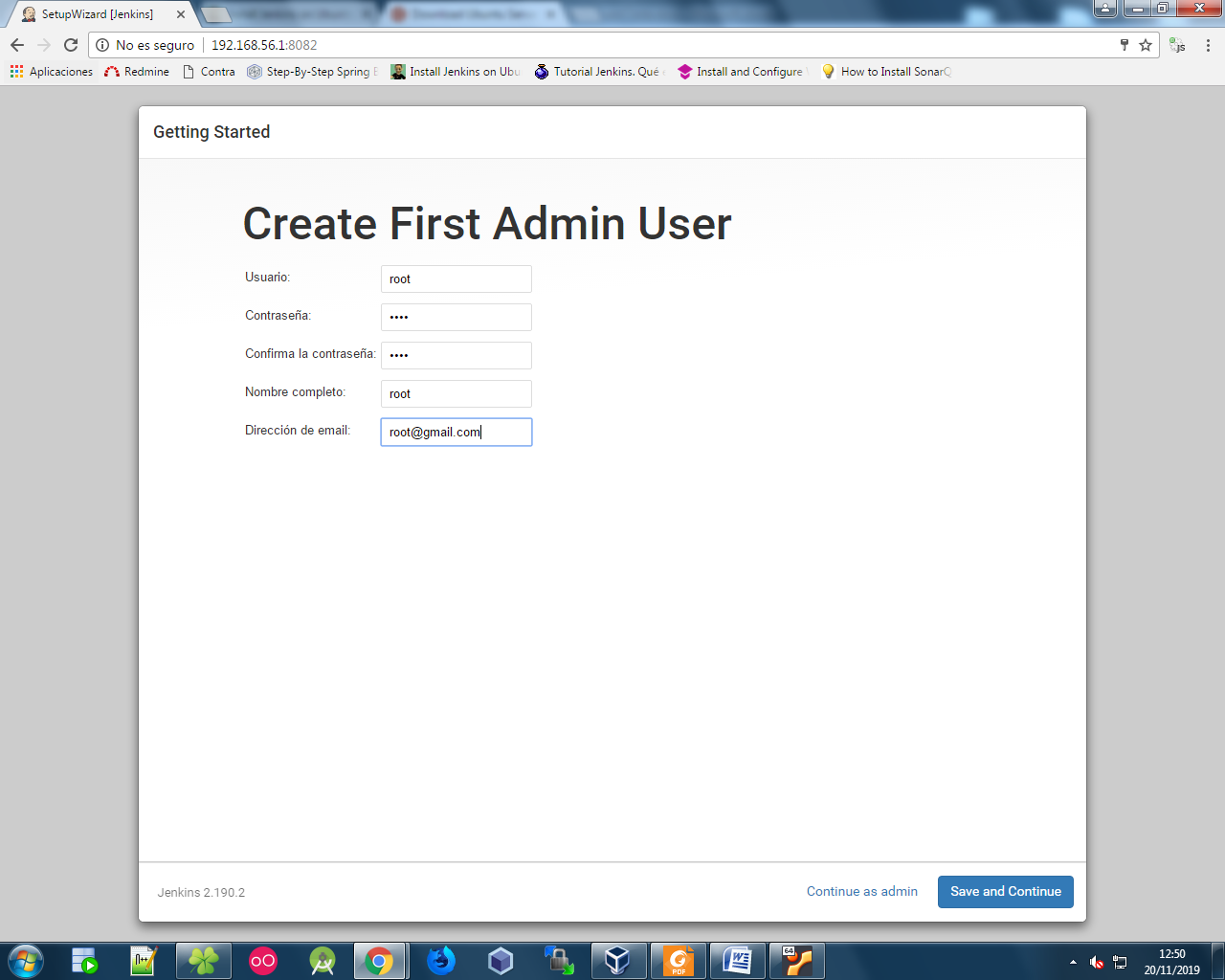


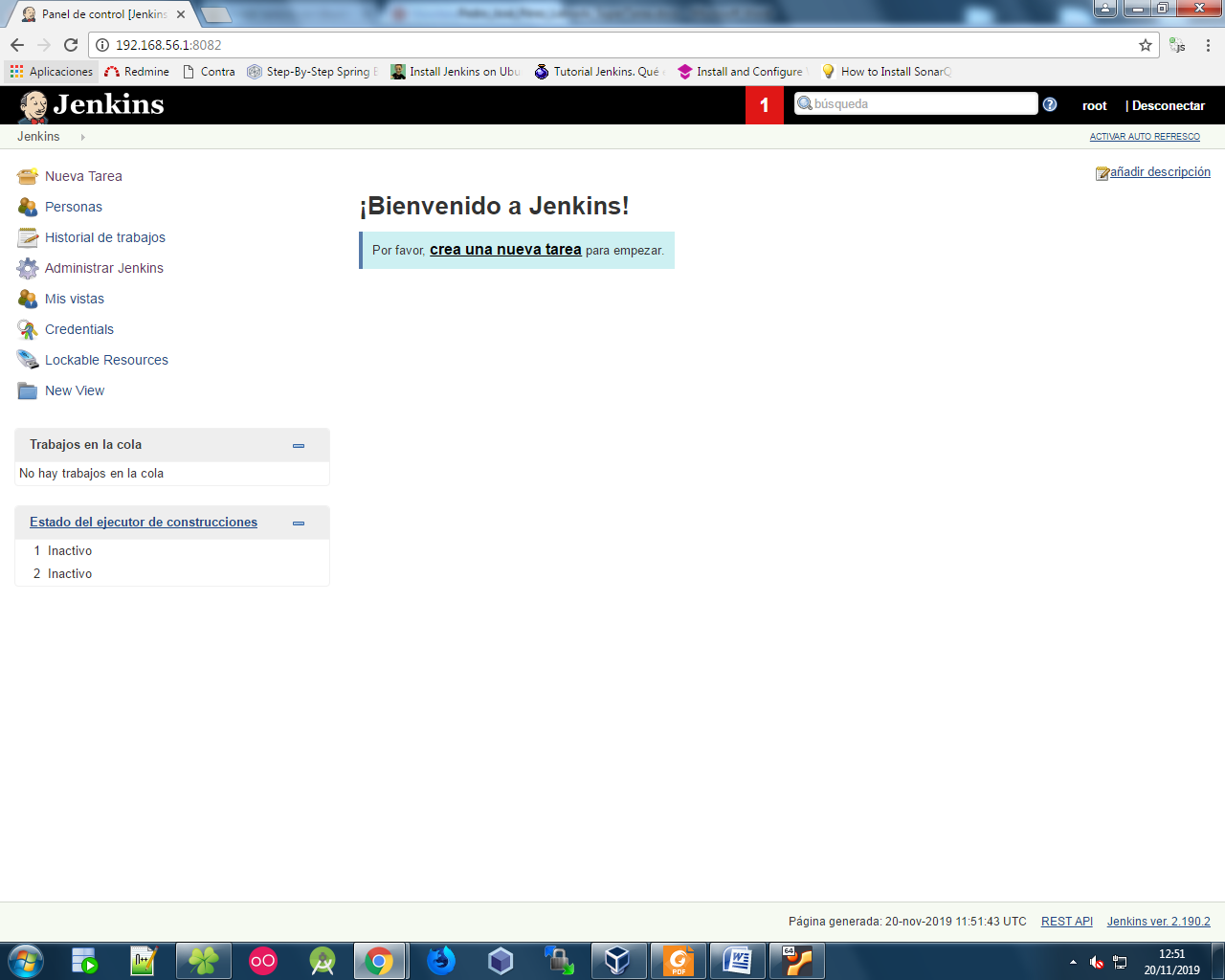
Una vez hecho eso, procedemos a instalar. Elegimos la opción de los plugins propuestos por Jenkins.





Una vez haya acabado de instalar, nos pedirá que creemos un usuario con su respectiva contraseña.



Finalizado esto, ya tendremos acceso a Jenkins.

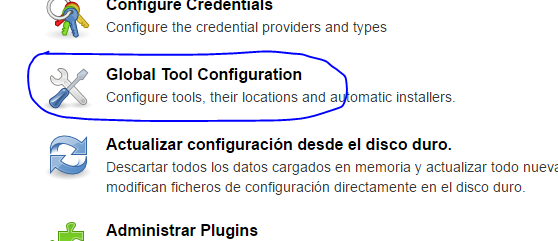
Ahora veremos cómo integrar código en Jenkins, en mi caso he utilizado GitHub, para que sea más cómodo a la hora de meter el código.

Mi proyecto es algo simple, es tan solo una prueba que muestra por pantalla "Prueba!" con spring boot.

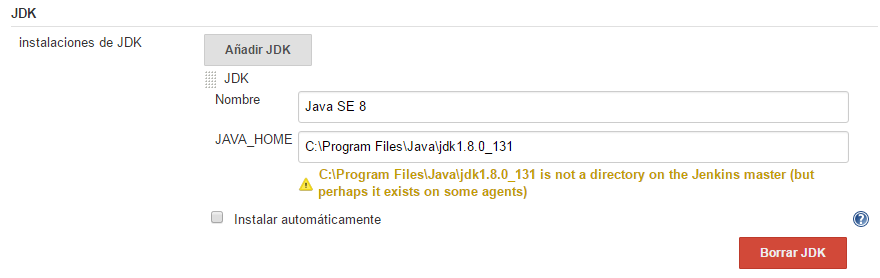
Necesitaremos Maven para poder realizar algunas de las operaciones que nos ofrece Jenkins con respecto a nuestro proyecto.

Para ello iremos a:





Una vez dentro veremos distintos apartados como por ejemplo el JDK de Java que también lo pondremos.

JDK

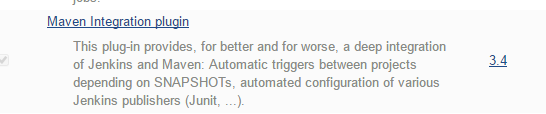
Maven



Ahora instalaremos el plugin de Maven para Jenkins.

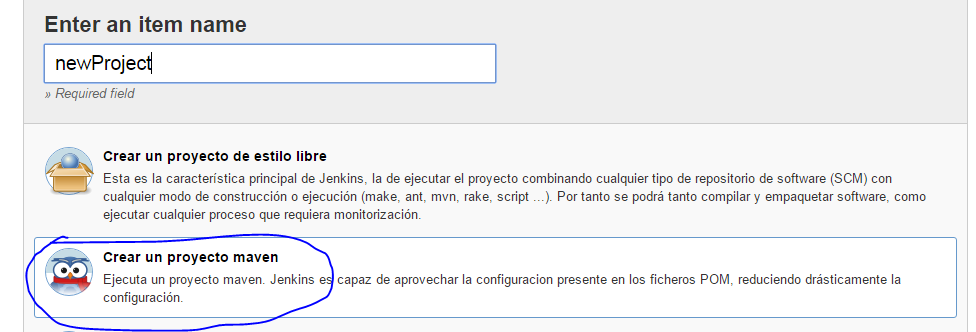
Administrar Jenkins --> Administrar plugins

y buscamos en todos los plugins uno que se llama "Maven Integration plugin".



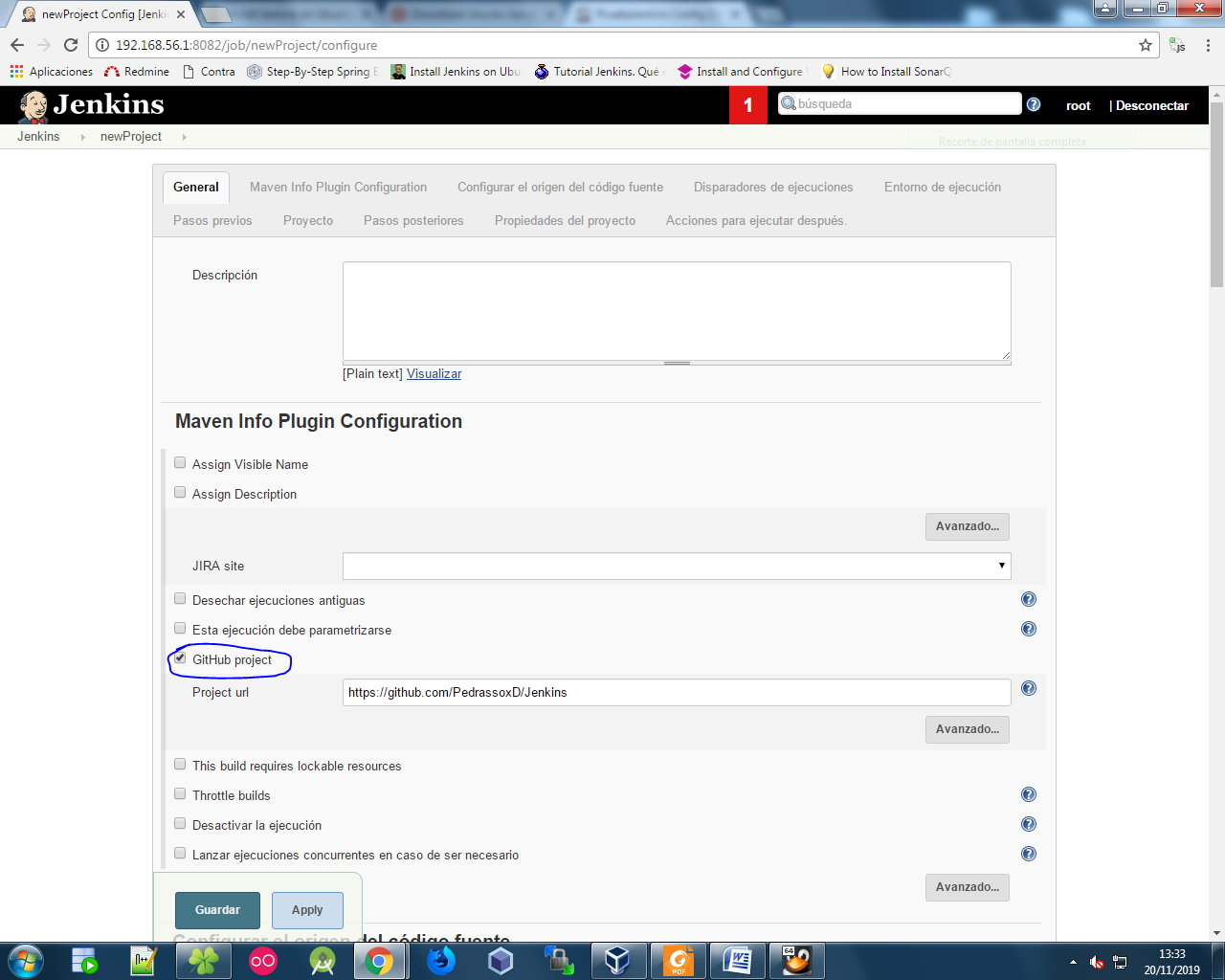
Ahora veremos cómo crear el proyecto en Jenkins.

Nueva Tarea.

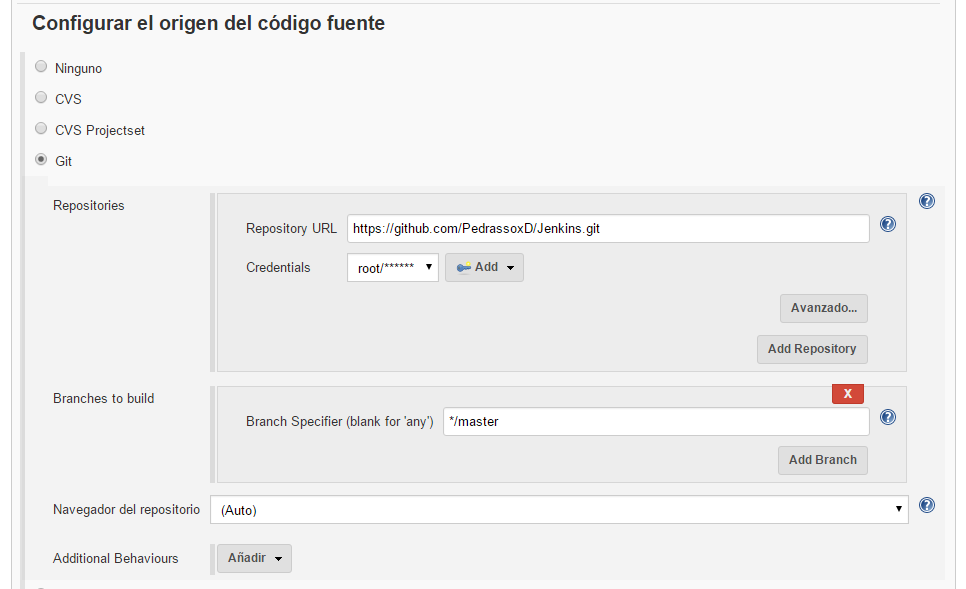


Y ahora viene donde configuramos nuestro proyecto para hacer que se compile y se despliegue. Jenkins automáticamente a la hora de desplegar el proyecto te lo empaqueta en un WAR.

Lo primero que haremos será poner la URL de nuestro repositorio en GitHub.



Acto seguido configuraremos el origen del código fuente, es decir, poner de nuevo la URL de nuestro repositorio .git



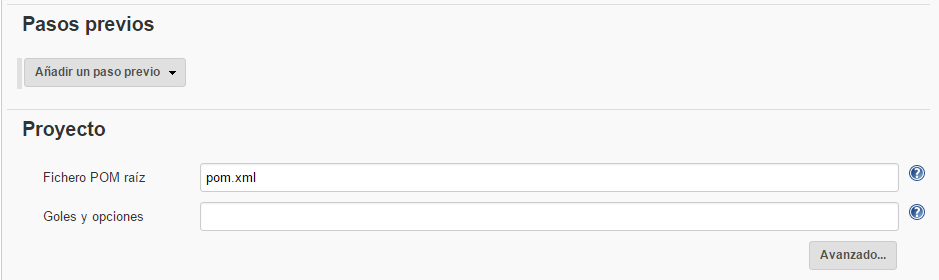
Ahora le daremos unos tiempos de ejecución y otro para que consulte el repositorio, los dos los pondremos para 15 minutos y además que ejecute siempre que cualquier SNAPSHOT de los que dependa sea creado.



Lo siguiente será el entorno de ejecución, que borrara el workspace antes de que se construya el proyecto y que aborte la ejecución del programa a los 4 minutos de haberlo desplegado.

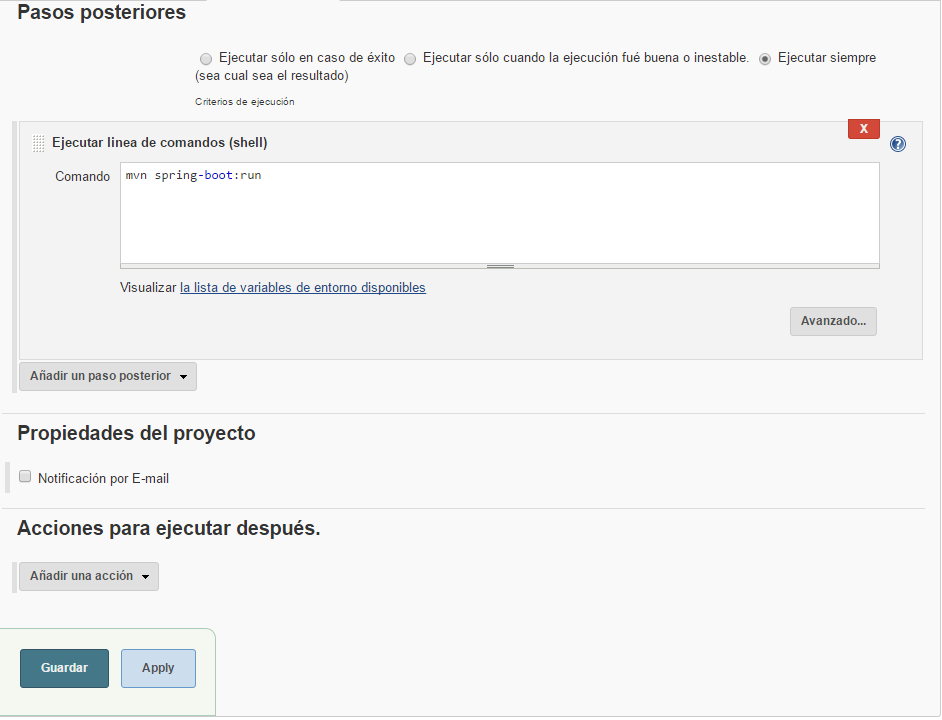


En los pasos previos por ahora no ponemos nada y en proyecto, por defecto nos aparecerán estos apartados así.

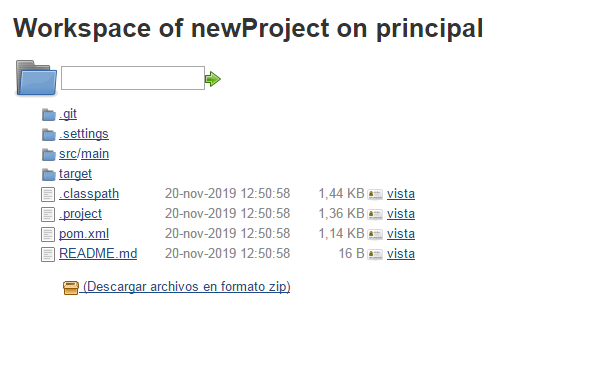


Por último para poder compilar dicho proyecto debemos poner en pasos posteriores lo siguiente.

Con esto haremos que se compile y que se despliegue, guardamos y comprobamos.



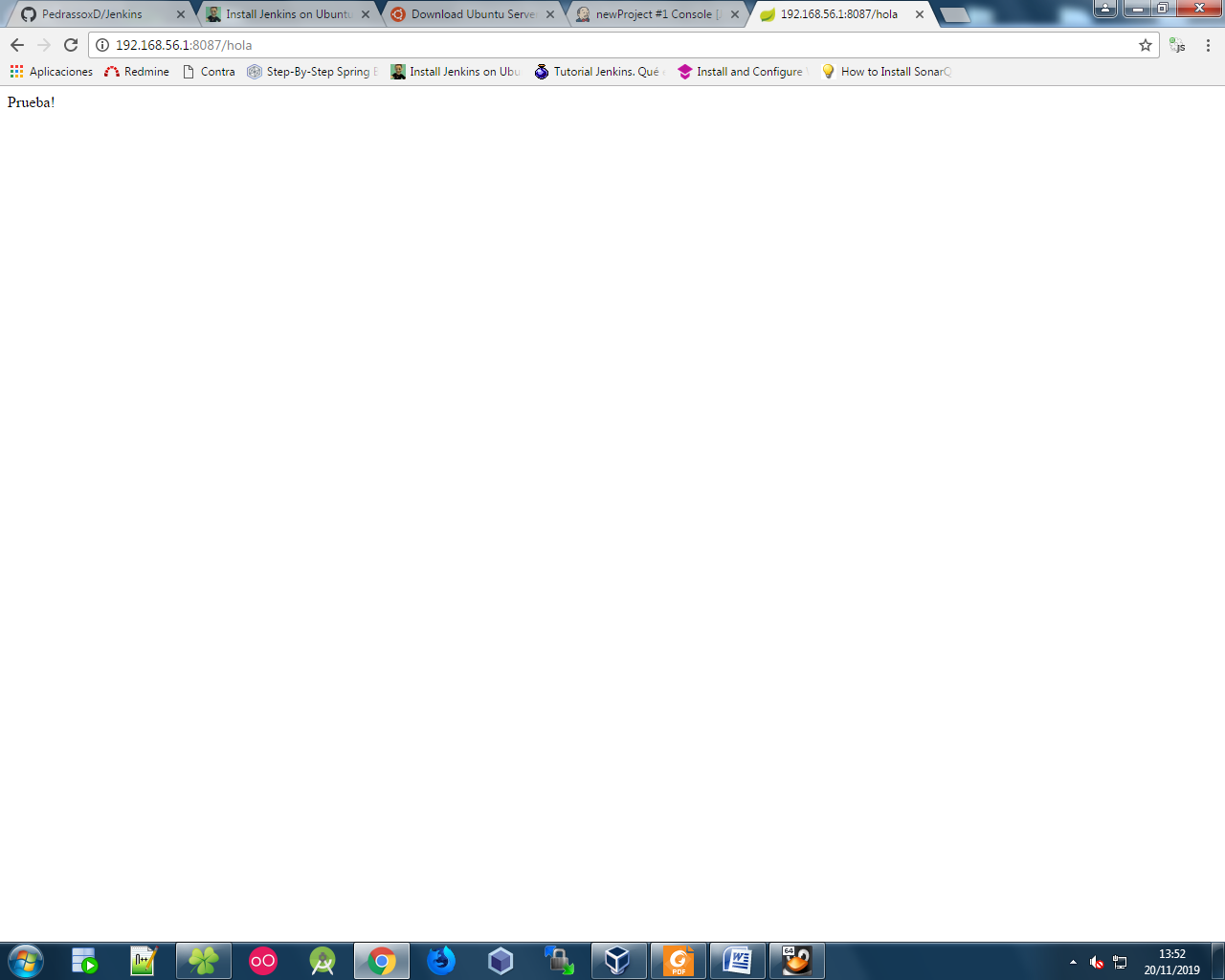
Veremos que se nos ha creado el proyecto bien.



Tenemos que hacer clic en "Construir ahora" y comenzará a cargar.



Como podemos observar en la siguiente imagen, vemos que se ha desplegado correctamente, entramos con la misma dirección IP, pero el puerto lo cambiamos, en mi caso es el 8087 porque es el que le tengo indicado dentro de mi proyecto.



Evaluación de la calidad del código mediante Sonar.

Para poder realizar dicha evaluación, antes tendremos que instalar y configurar Sonar.

En nuestra máquina virtual donde tenemos nuestro Jenkins, podemos hacerlo.

Como ya tenemos Java instalado porque lo hicimos al principio no nos hará falta volver a instalarlo, así que empezamos directos con lo que quermos