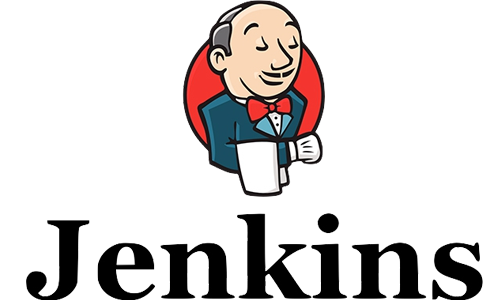
[Jenkins – Aprende a Montar un Entorno de Integración Continua](http://www.robertocrespo.net/kaizen/aprende-a-montar-un-entorno-de-integracion-continua-iv-jenkins/" \o "Jenkins – Aprende a Montar un Entorno de Integración Continua (IV))



**Jenkins**será el componente que ejerza como **servidor de integración continua**en el entorno que estamos montando.

**Jenkins** CI no es más que un sistema desplegado en un servidor que nos ayuda en la tarea de hacer integración continúa y programar tareas automáticas cuando ocurra una determinada acción. A este tipo de servicios se los conoce como CI/CD (Coninuous integration / Continuous deploy)

## Pero antes de empezar, ¿Sabes qué tipo de tareas puede realizar Jenkins?

**Jenkins**, como servidor de integración continua, permite planificar y realizar multitud de tareas, simplificando los procesos involucrados en el ciclo de vida de un proyecto. Algunas de sus características más importantes son:

* Comprobación cada cierto periodo de tiempo si se ha realizado algún commit en el repositorio de control de versiones (GIT), y en caso de ser así, compilar el código y ejecutar las pruebas para testearlo.
* Notificación de errores que se hayan detectado tras la ejecución de pruebas, por ejemplo vía mail, twitter, chat, etc.
* Generación y publicación de binarios.
* Ejecución de métricas de calidad y visualización los resultados.
* Generación de documentación asociada a un proyecto.

## Instalación

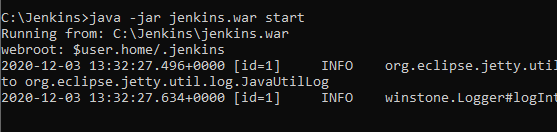
En seguida comprobaras que instalar Jenkins es muy sencillo. Simplemente sigue los siguientes pasos:

1. Descarga la herramienta de la página oficial (<https://jenkins.io/download/>). Observa que puedes descargar Jenkins como paquete nativo para tu sistema operativo en formato ***msi***, para Windows.

Como ves, hay dos versiones, la **LTS** y la **Weekly**, la versión LTS la sacan anualmente de tal forma que sea lo más estable posible, si por el contrario, quieres estar siempre a la última escoge la Weekly, aunque puede tener ciertos bugs.

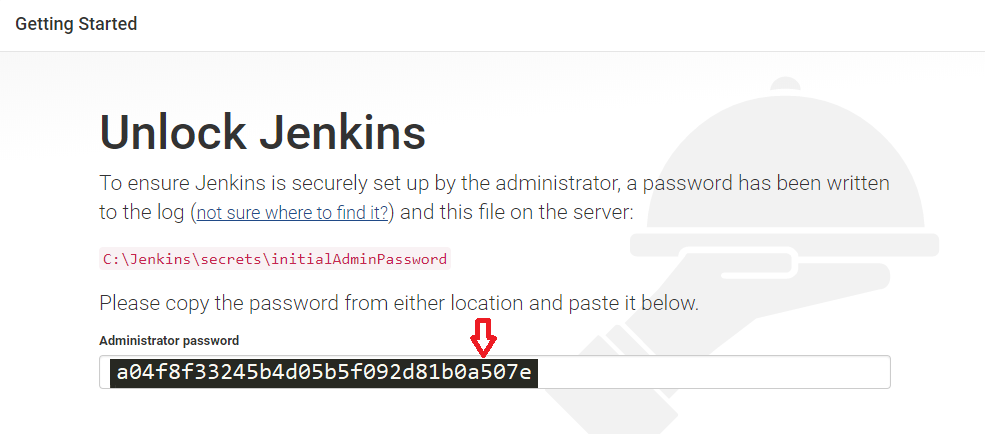


1. instalar Jenkins(yo lo he instalado en la carpeta **C:/Jenkins**, que **previamente me he creado**)
2. Por último, sólo queda **levantar el servidor**. Para ello sitúate en la carpeta donde guardaste el WAR y ejecuta desde línea de comandos la instrucción  **java -jar jenkins.war start**

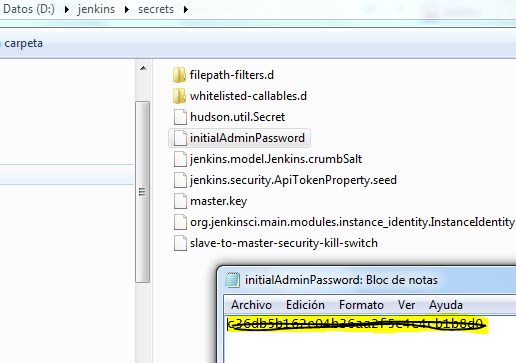


* + Otra opción es que utilices un servidor de aplicaciones que tengas instalado para ejecutar el WAR, por ejemplo Tomcat 8.

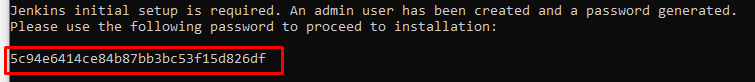
1. Podrás acceder a Jenkins a través de la URL [http://localhost:8080](http://localhost:8080/), y nos aparece el siguiente mensaje



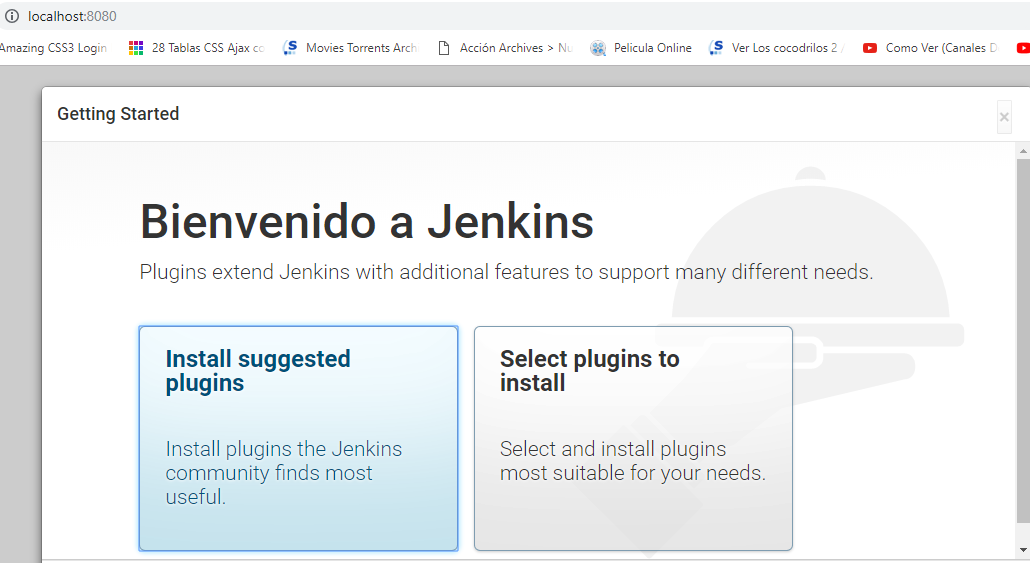
**Traducción**: Para asegurarse de que Jenkins esté bien configurado por el administrador, se ha escrito una contraseña en el fichero que se encuentra en esa dirección en mi caso es un string como este: **initialAdminPassword**



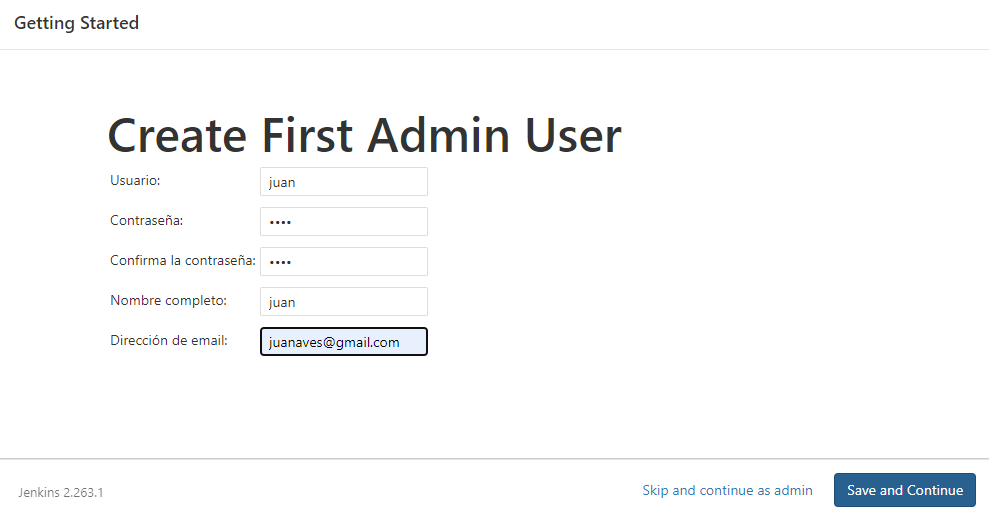
O bien cuando estoy instalando aparecerá



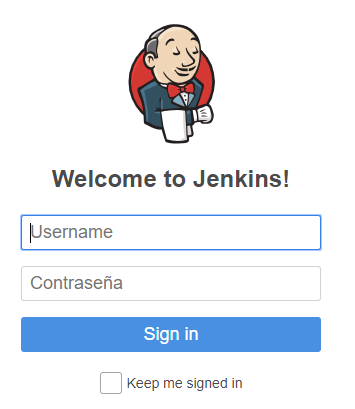
1. Si todo va bien no aparecerá la siguiente pantalla



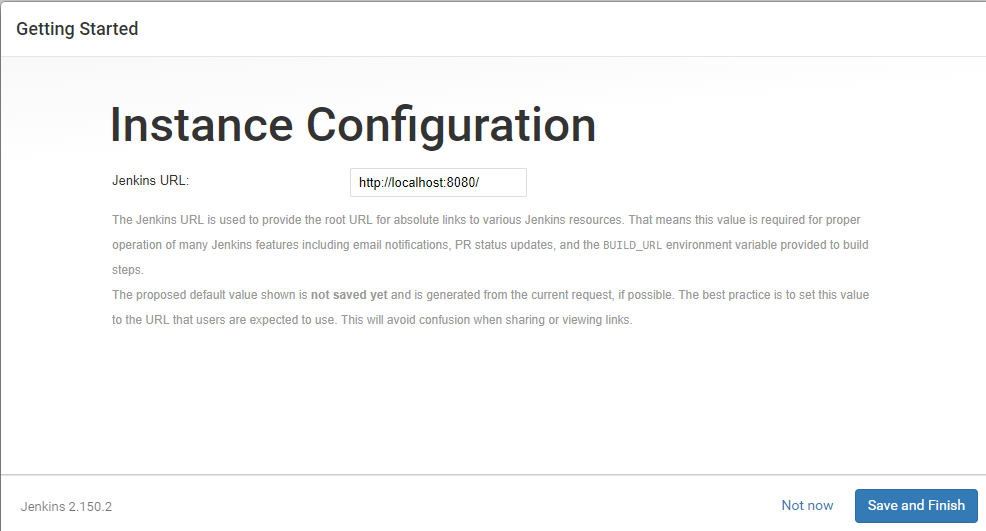
1. Pulsaremos la opción de **Install sugegest plugins**
2. Una vez instalados, nos aparece la **ventana de creación del usuario administrador**. Rellenamos los datos y pulsamos sobre **“Save and Finish”**



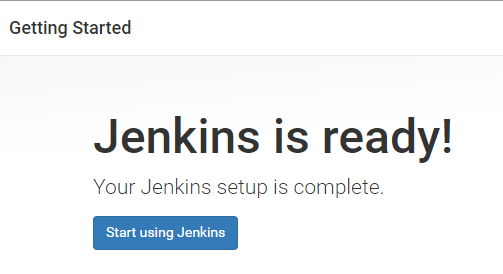
1. Lo siguiente es crear un usuario administrador, para a continuación una vez creado nos aparecerá la siguiente pantalla para loguearnos:



1. Una vez nos logueemos correctamente nos aparecerá la siguiente pantalla, en donde pulsamos el botón Save and Finish



1. Donde finalmente nos aparecerá



1. Y tras pulsar Star using Jenkins



1. ¿Cómo iniciar jenkins en un puerto diferente en lugar de 8080 usando el símbolo del sistema en Windows?

Se inició bien y se navegó fácilmente como http://localhost:8080

Quiero comenzar en el puerto 9090. ¿Cómo puedo hacer eso?

Usa el siguiente comando en el símbolo del sistema:

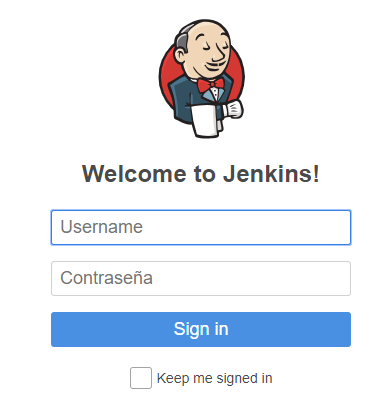
java -jar jenkins.war --httpPort=9090

Si quieres usar https usa el siguiente comando:

java -jar jenkins.war --httpsPort=9090

Los detalles están [aquí](https://wiki.jenkins-ci.org/display/JENKINS/Starting+and+Accessing+Jenkins)

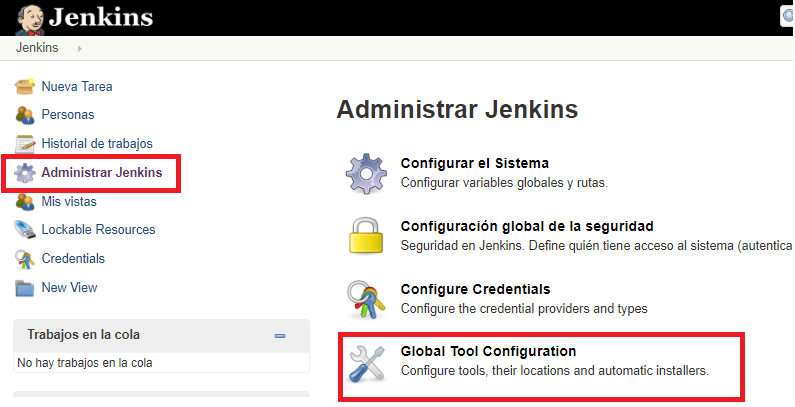
1. Si, ahora, cambiamos la URL de acceso por **“http://localhost:9090”**, accederíamos a la pantalla de acceso, en la que teclearíamos los datos del usuario administrador que configuramos anteriormente.



1. Y ya estaríamos de nuevo dentro de **Jenkins**, con el puerto que deseábamos.

## Configuración del entorno de trabajo

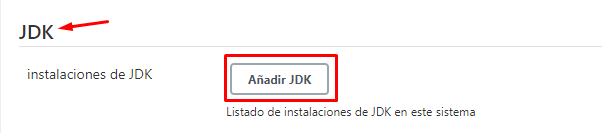
Desde la **pantalla de bienvenida**, pulsamos de nuevo sobre **“Administrar Jenkins”**, para posteriormente pulsar sobre **Global Tool Configuration**



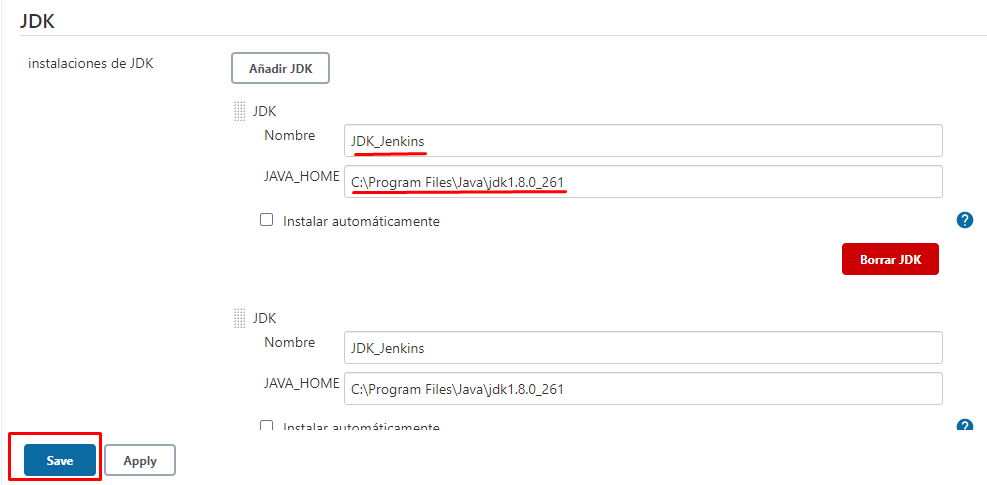
Desde esta pantalla configuraremos lo que podemos llamar el “**núcleo**” de nuestro Jenkins: **JDK** (para la creación de programas en Java), **Git** (para control de versiones) y **Maven** (para la gestión y construcción de proyectos), **Sonar** y **Sonar Scanner(**Analizadores de código**)**. Para cada uno de estos aspectos, existe la correspondiente sección en esta pantalla, además de otras para **Gradle**, **Ant** y **Docker**.

Java

En la sección **“JDK”**, **pulsamos** sobre **“Añadir JDK”**. Tecleamos en primer lugar un nombre para nuestro JDK, por ejemplo **“JDK\_Jenkins”**.

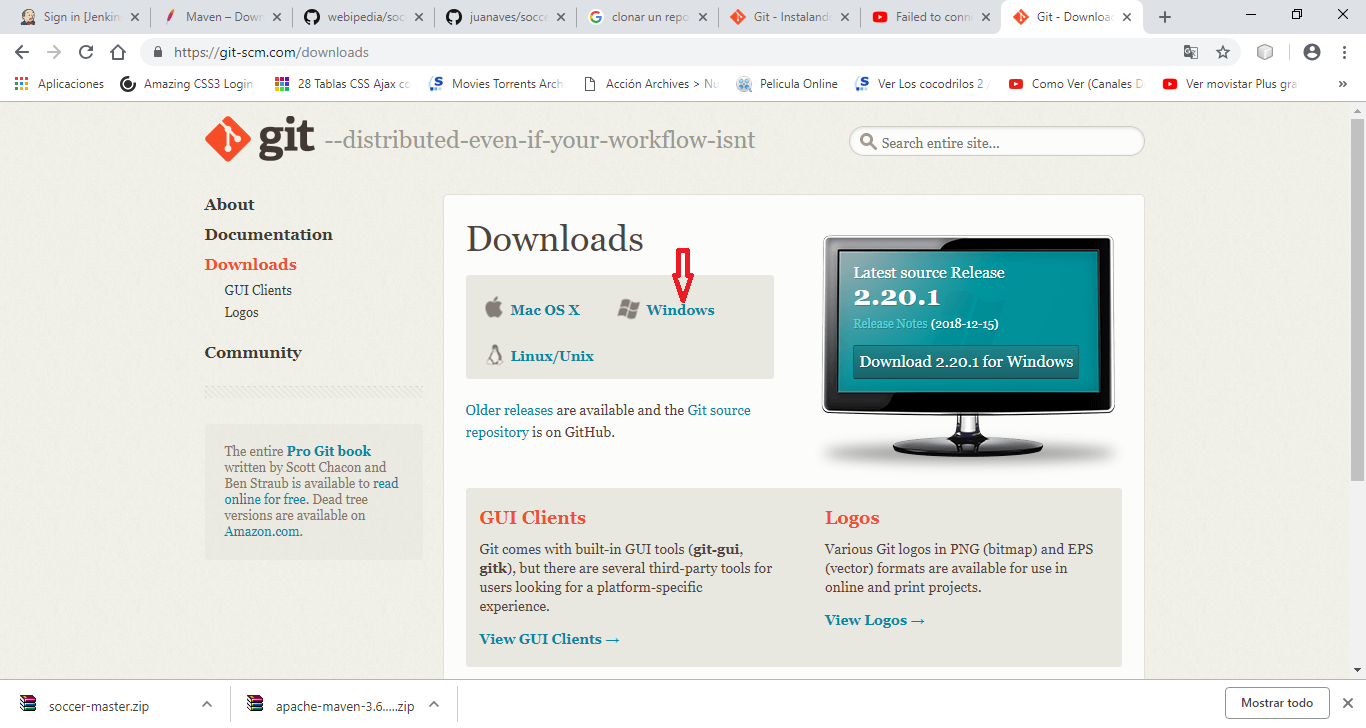


Vemos que aparece marcado el check de **“Instalar automáticamente”**, indicándonos la opción de descarga de **“java.sun.com”** de la última versión de Java disponible (actualmente, la **“9.0.1”**). En mi caso, ya tengo instalada una versión 8 con la que estoy trabajando en otros de mis proyectos, así que voy a desmarcar esta opción y voy a configurar la versión de Java la que tengo instalada. Al desmarcar la opción de descarga, se habilita una cajita para introducir el **JAVA\_HOME**. Como tengo la **versión 8 de Java update 151**, tecleo la ruta correspondiente (**“C:\Program Files\Java\jdk1.8.0\_151”**).

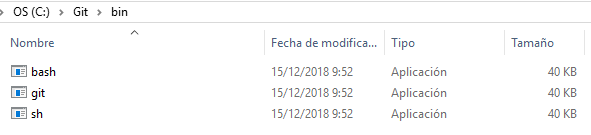


**Git**

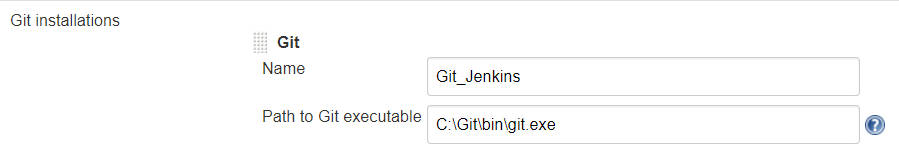
(en nuestro caso recordar que cuando instalamos docker también se instaló Git, que esa versión de git a **mí** ( no tiene por qué dárselos a todo el mundo) me ha dado problemas por lo que la voy a desinstalar, con el revoUnistaler, para volver a instalarla desde esta dirección <https://git-scm.com/downloads>



Yo la voy a instalar en la carpeta **C:\Git (la instalacion con las opciones predeterminadas por defecto que aparecen en dicha instalacion)**



Pasamos a la siguiente sección, la de **Git**.. Aquí simplemente tecleamos un nombre (por ejemplo **“Git\_Jenkins”**) e indicamos la ruta del ejecutable **“Git.exe”** en nuestra máquina (en mi caso, **“C:\Git\bin\git.exe”**).



**Maven**

Por último, configuraremos nuestro **Maven**. En la sección correspondiente, pulsamos sobre **“Añadir Maven”**.

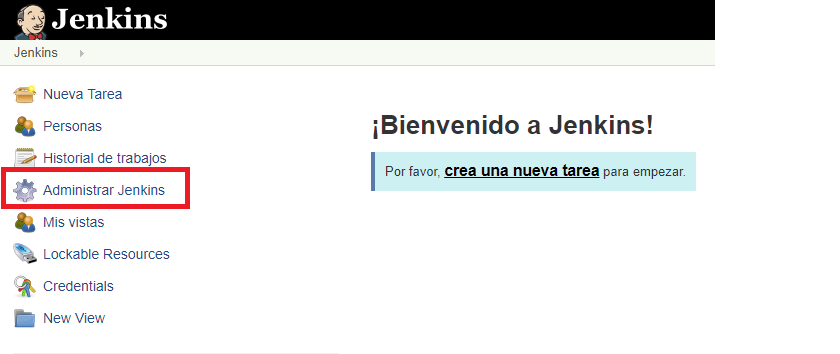
  
Tecleamos en primer lugar un nombre, como por ejemplo **“Maven\_Jenkins”**. Como vemos, al igual que con el JDK, aparece marcado un check **“Instalar automáticamente”**, que nos da la opción de instalar automáticamente desde Apache la **versión 3.5.2**(la última disponible en ese momento).

 Si la tengo instalada, lo que haré será **desmarcar esta casilla** e introducir la ruta en mi sistema del home de Maven (en este caso, **“C:\apache-maven-3.6.0”**).

Y ya estaría todo. Pulsamos **“Save”** en la parte inferior de nuestra pantalla de configuración, lo que nos llevaría de nuevo a la **pantalla de administración de Jenkins**.

**Administrar Plugins**

Lo primero de todo será configurar la herramienta para que localice las rutas de instalación de los distintos componentes que forman parte de nuestro entorno de integración continua. Para ello accede primero a la opción de menú: Administrar Jenkins -> Configurar el sistema.

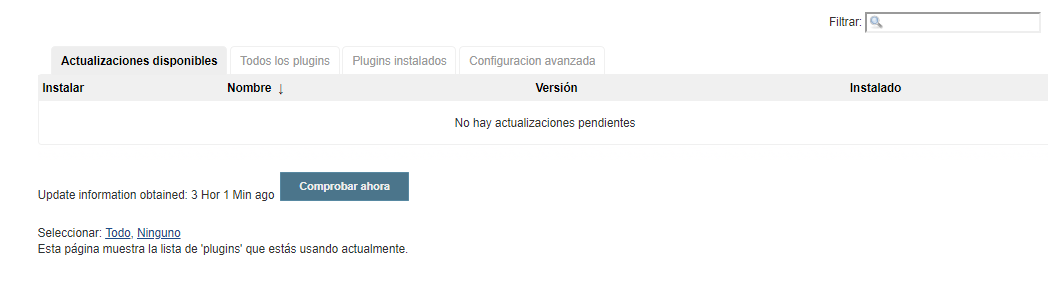


La siguiente pantalla nos mostrará un mensaje de configuración proxy inverso incorrecta

(ver anexo final que es un proxy inverso). Pulsamos **Ocultar** y luego **“Administrar Plugins”**



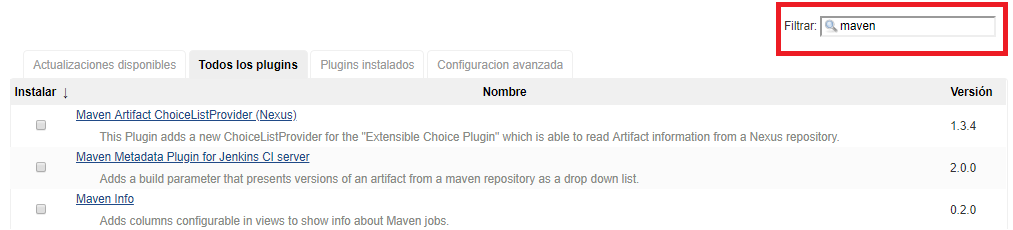
Si recordamos, al acceder por primera vez a Jenkins, seleccionamos la opción de instalar los **plugins más populares**. Para **instalar el resto**, accederíamos a esta sección. La primera pestaña nos indica el listado de actualizaciones disponibles para los plugins que tengamos instalador actualmente (en nuestro caso, como acabamos de instalarlos, no aparece ninguno).



Pulsamos sobre la pestaña **“Todos los plugins”**, desde la cual buscaremos el plugin que queremos instalar.

**Maven**

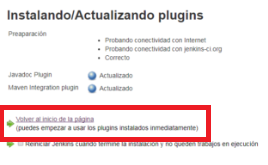
Para ello, en la parte superior derecha tenemos una cajita para realizar un filtrado por nombre: tecleamos **“maven”**.

****

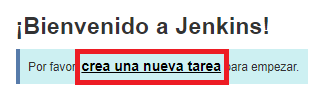
Buscamos el plugin **“Maven Integration”**, que normalmente aparecerá al ***final del listado***. Lo seleccionamos y pulsamos sobre **“Instalar sin reiniciar”**

****

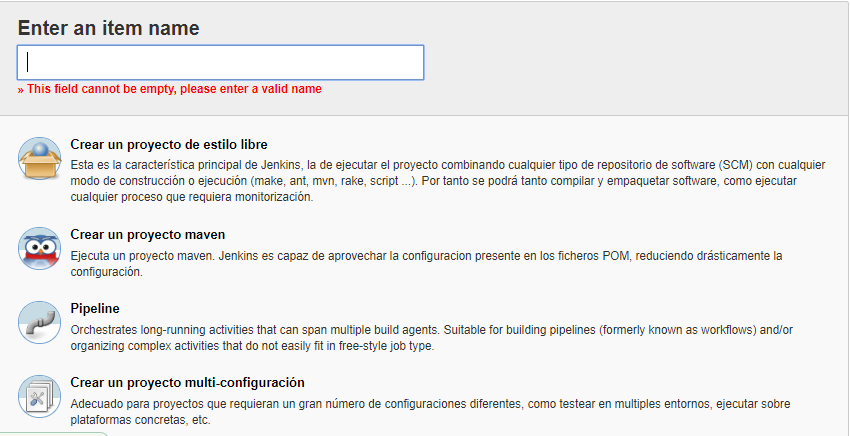
Se lanzará entonces el **proceso de instalación** del plugin. Esperamos que finalice y pulsamos sobre **“Volver al inicio de la página”**.

****

De nuevo en la **pantalla de bienvenida**, pulsamos ahora sobre **“Nueva Tarea”**.

****

Jenkins nos lleva entonces a la **pantalla de configuración inicial de la nueva tarea**, en la que se nos pide que le asignemos un nombre, y un listado de ítems en la parte inferior en la que debemos indicar el tipo de tarea que será. En la imagen vemos que, como segundo ítem, aparece **“Crear un proyecto maven”**. ***Si no hubiéramos instalado el plugin anterior, dicho ítem no aparecería.***



https://dev.to/vanessa\_corredor/instalar-manualmente-maven-en-windows-10-50pb