**INTEGRACION CONTINUA**

**RESOLUCIÓN DE EJERCICIOS**

A continuación encontrará el desarrollo de los ejercicios que resolvió en la semana. Contraste las respuestas entregadas por el docente con las desarrolladas por usted. En caso que no coincidan, y persistan dudas, le invitamos a repasar los contenidos y/ o consultar con su profesor.

* 1. Montar un entorno local de Integración Continua con Jenkins

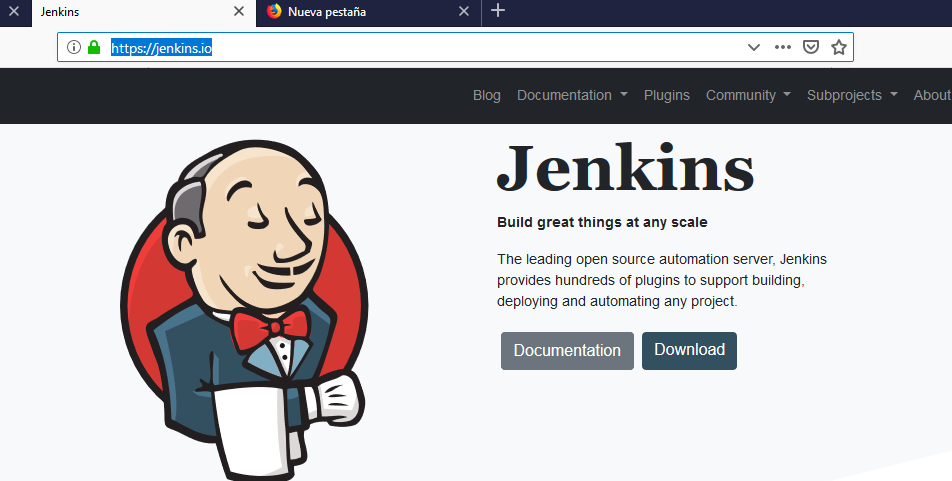
Instalación de Jenkins

Paso 1: Para la instalación de Jenkins el primer paso es ir a su página web en el siguiente enlace-> <https://jenkins.io/>

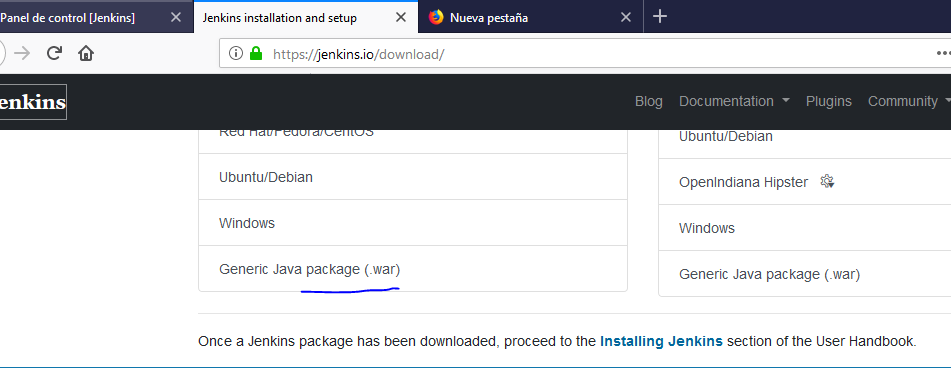
Nota 1: los enlaces pueden cambiar durante el tiempo en caso contrario se recomienda hacer una búsqueda en Google para descargar la herramienta.

Nota 2: para el correcto funcionamiento es necesario tener instalado el JDK de java versión 8 ya que esta herramienta este escrito en dicho lenguaje.

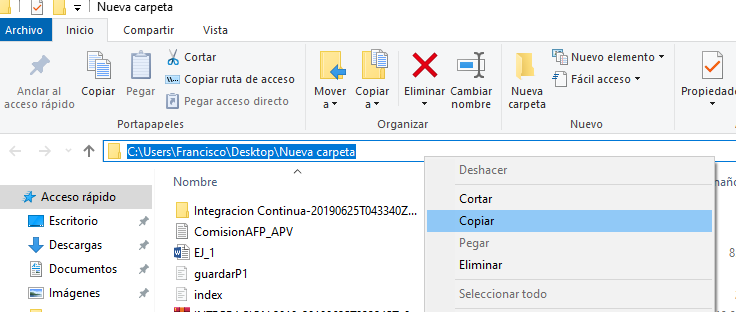
Una vez aclarado el punto anterior se procede a descargar Jenkin en la página principal presionando el botón download.



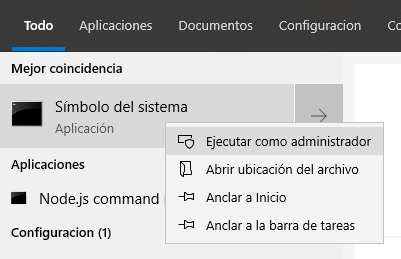
Paso 2: una vez dentro de la pagina de descarga dirigir el cursor hacia la versión **LTS** y presionar la opción de “Generic java Package(war)”.



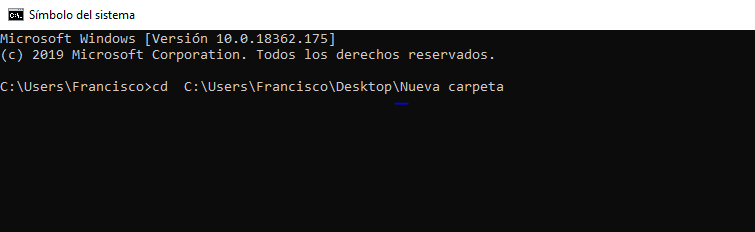
Paso 3: una vez descargado Jenkins nos dirigimos a la carpeta de descarga y copiamos la ruta de descarga (**esto es muy importante por que en los pasos posteriores la necesitaremos en la consola de comando de Windows CMD**)



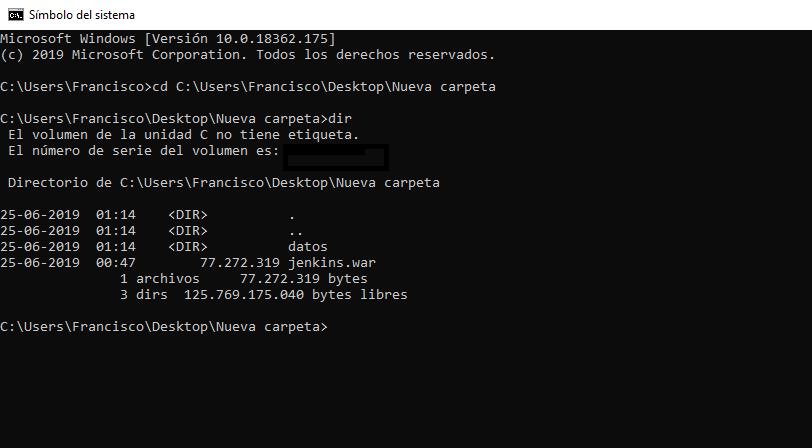
Paso 4: una vez copiado la ruta de descarga procedemos a abrir nuestra consola de comandos CMD de Windows, para esto presionamos el botón inicio del escritorio y en la barra de búsqueda presionamos CMD aparecerá una ventana negra y la ejecutamos como administrador.



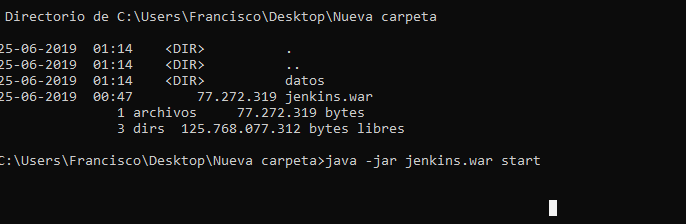
Paso 5: una vez dentro del la consola de comando CMD presionamos el comando CD (Change Directory) y pegamos la ruta copiada en el paso 3 o dentro de la consola presionar click derecho se pegara de forma automática y luego pres.



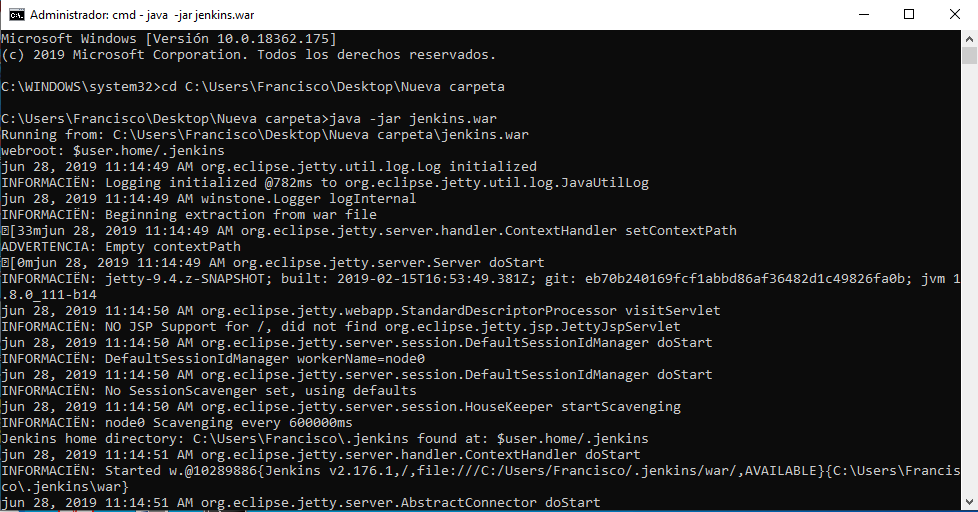
Paso 6: Dentro del directorio mencionado para realizar la verificación de los archivos ingresamos el comando DIR que nos mostrara los archivos en la carpeta de descarga.



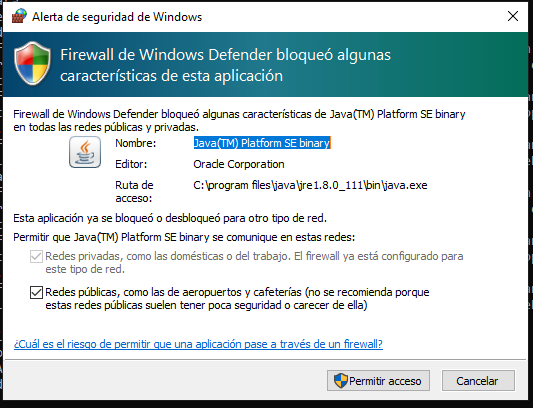
Paso 7: Una vez que hayamos encontrado el archivo “Jenkins.war” procede a la ejecución del servidor con el comando “java -jar Jenkins.war start”



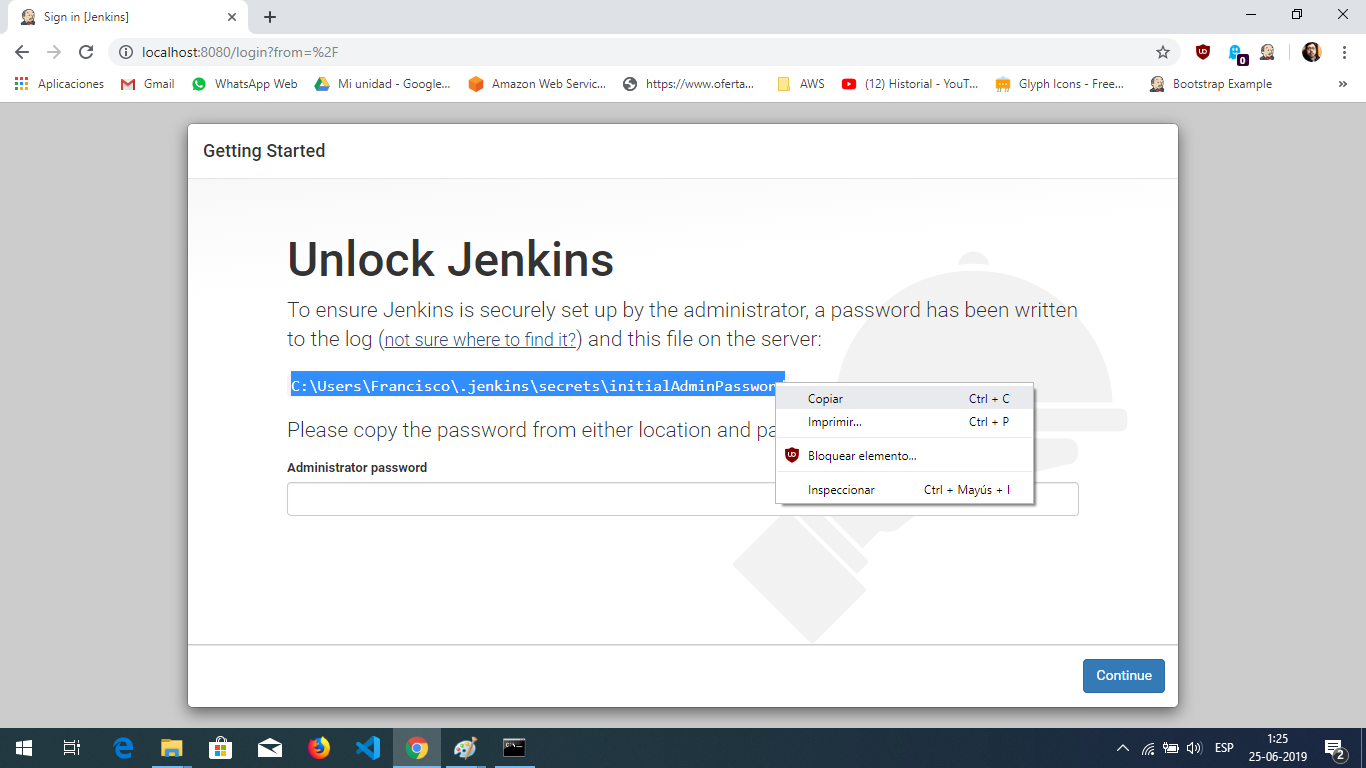
Se ejecutara el script montando el servidor como aparece en pantalla



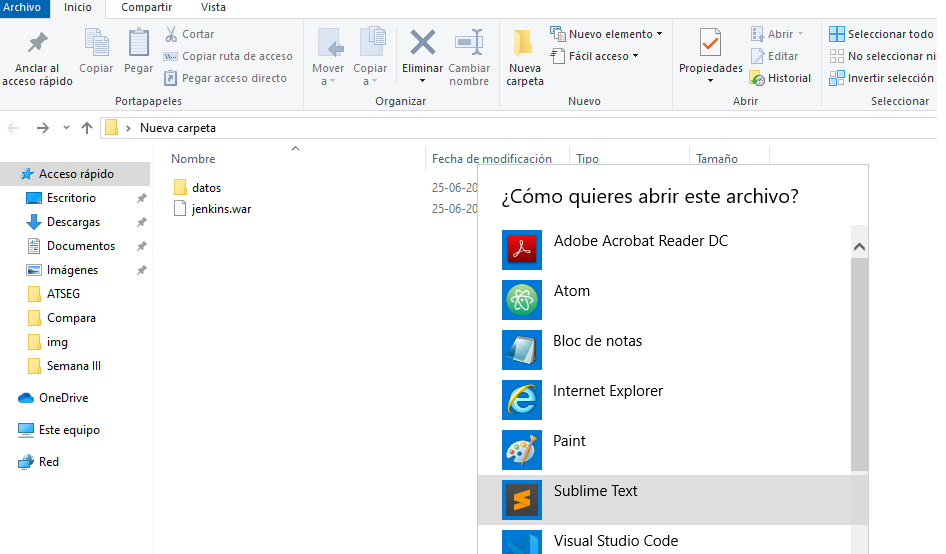
Durante el proceso aparecerá una alerta del firewall de Windows y presionamos permitir acceso.

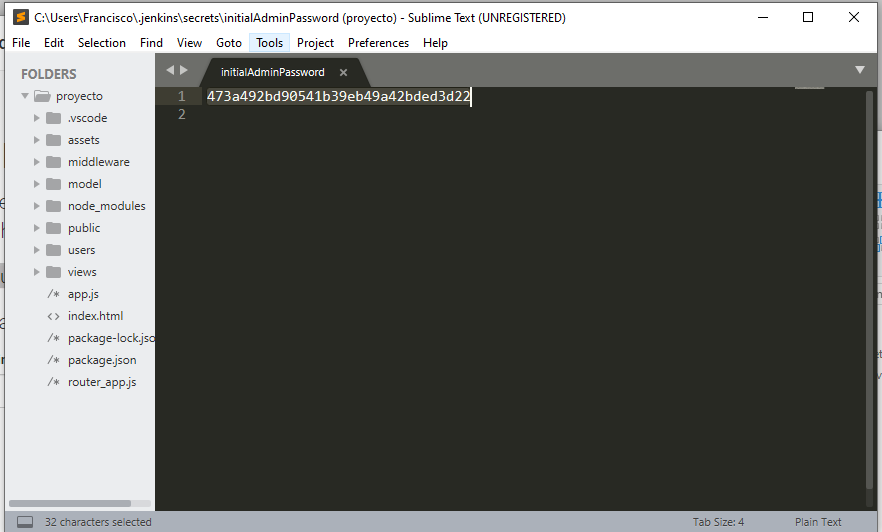


Paso 8: con el servidor ya montado nos dirigimos al navegador de vuestra preferencia ya sea Chrome o Firefox y en la barra de búsqueda ingresamos la dirección localhost:8080

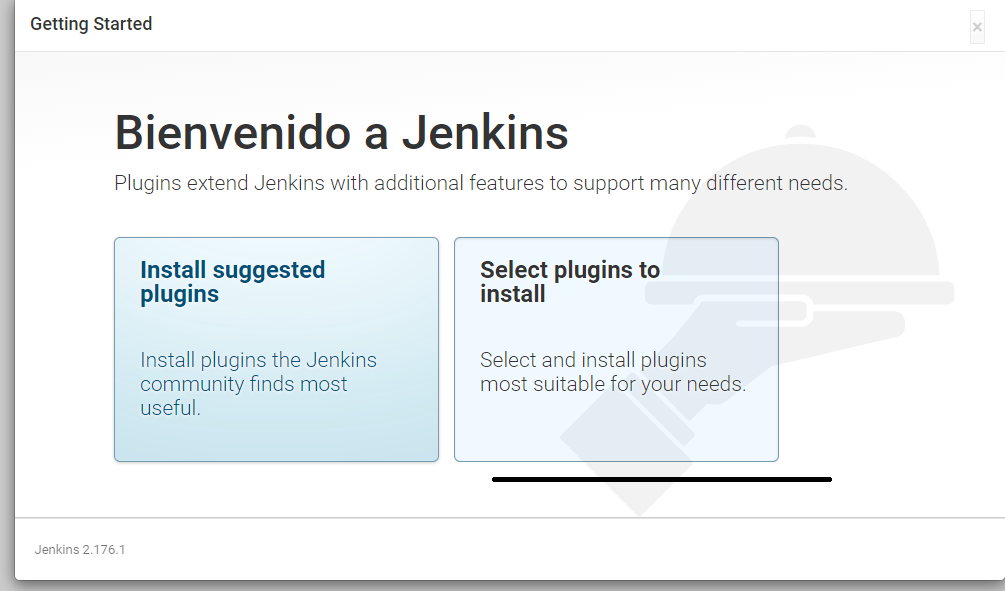


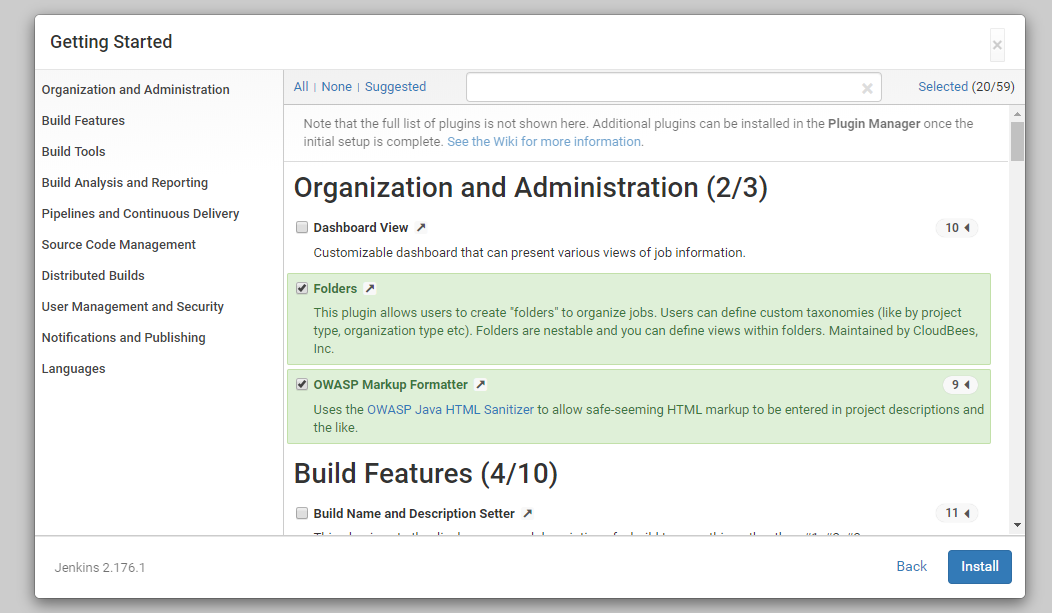
Nos pedirá ingresar una contraseña la cual copinado la dirección que nos aparece en pantalla y luego la pegamos en el navegador de Windows y abrimos nuestro editor de texto de preferencia, copiamos la contraseña y la pegamos en el navegador

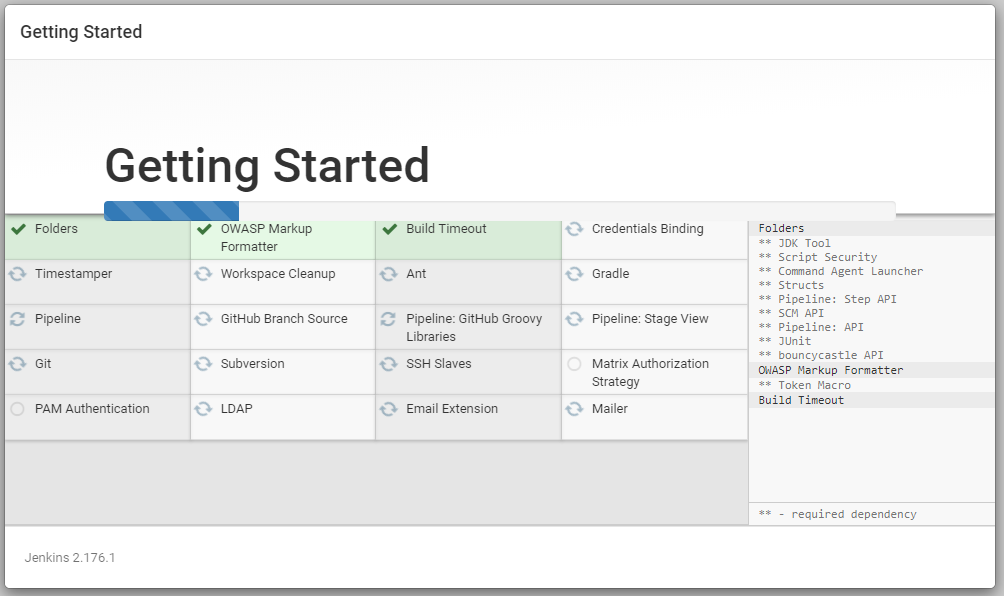




Nota: Esto también se puede realizar abriendo una nueva pestaña copiando la ruta y pegándola en la barra de direcciones de nuestro navegador web.

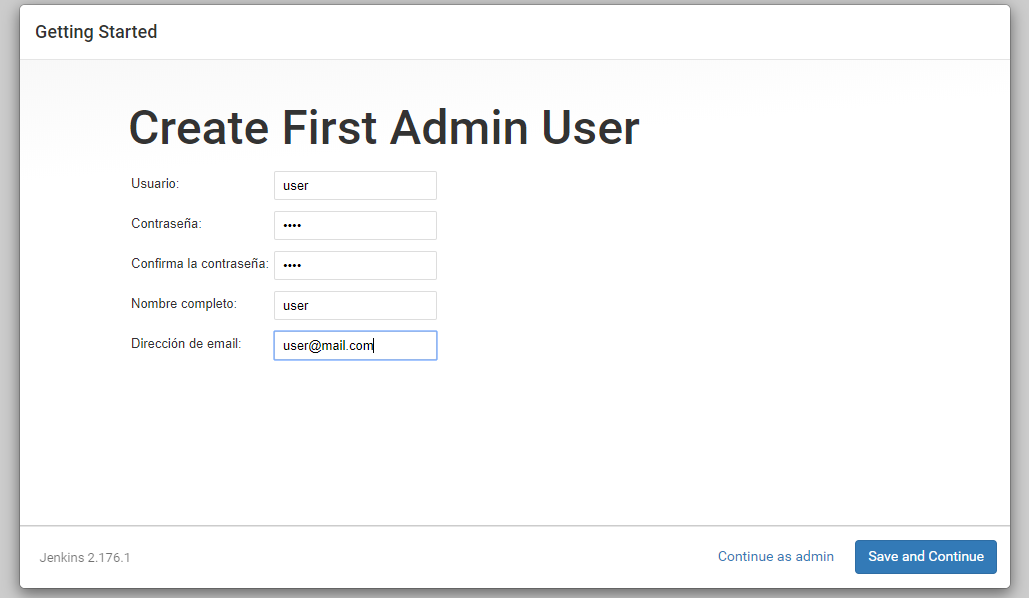
Paso 9: nos aparecerán dos opciones de instalación, seleccionamos la opción de “select plugin to install” como objetivo de verificar los plugins instalados en la plataforma y finalmente install.

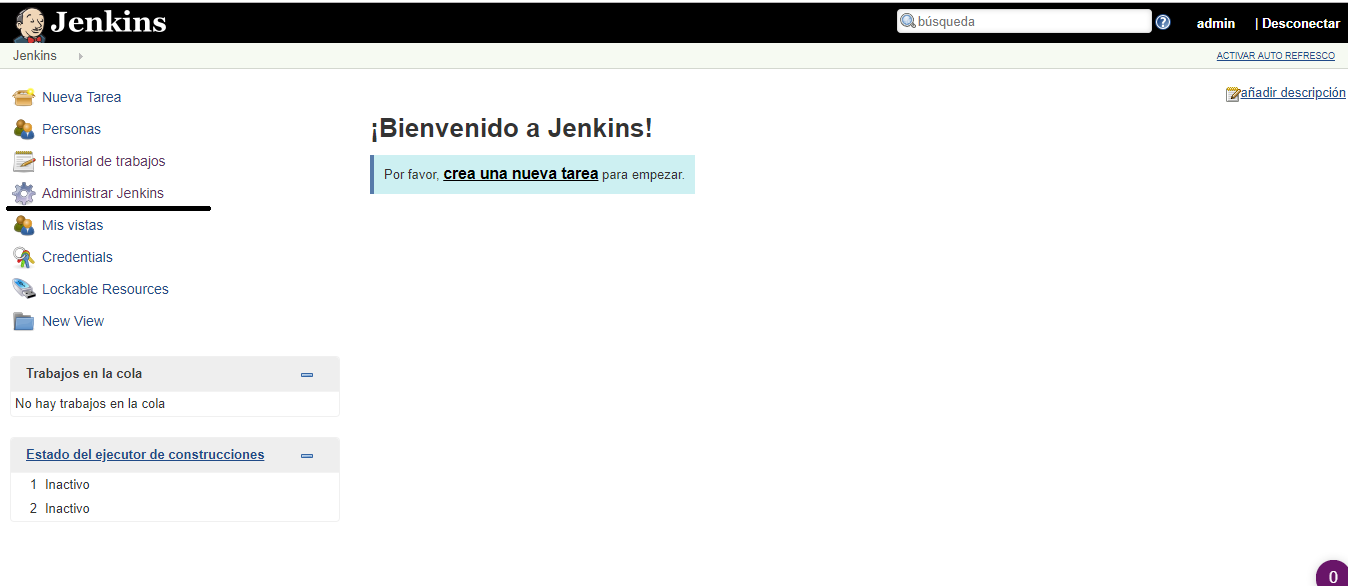




Paso 10: una vez instalado Jenkins creamos la cuenta de usuario administrador y presionamos “Save and Continue” con esto ya tendremos instalado nuestra plataforma de integración continua.

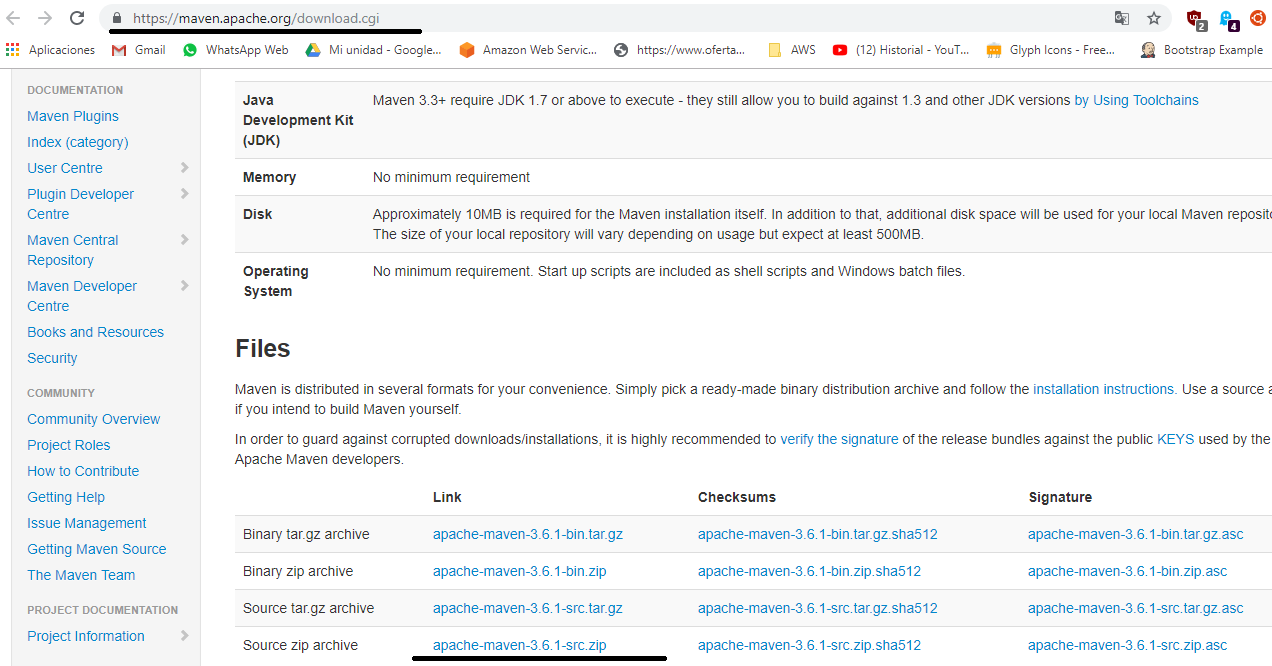
.

****



Instalación de Maven:

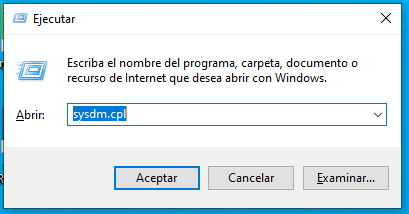
Paso 1: Para la instalación de Maven debes dirigirte a la siguiente enlace-> <https://maven.apache.org/download.cgi> luego presiona la opción de files y se descarga el archivo zip y descomprimir el archivo.



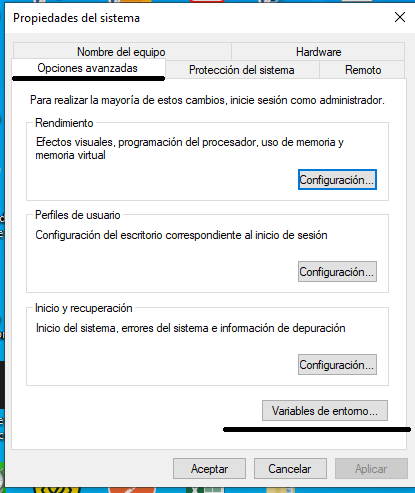
Nota: los enlaces pueden cambiar durante el tiempo en caso contrario se recomienda hacer una búsqueda en Google para descargar la herramienta.

Paso 2: Crear una carpeta en el disco C: llamada dev(el nombre puede variar la idea es que se identifique la carpeta), copiar el contenido dentro del archivo comprimido hacia la nueva carpeta.

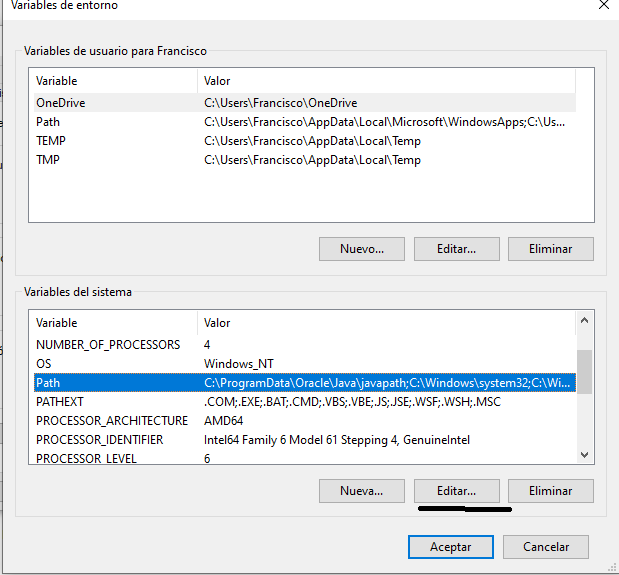
Paso 3: Configurar Variable de entorno en esta oportunidad se realiza la configuración de variable de entorno en el sistema operativo Windows 10. Lo primero que debemos hacer es presionar la tecla Windows + r en nuestro teclado y escribir el siguiente comando “sysdm.cpl”.



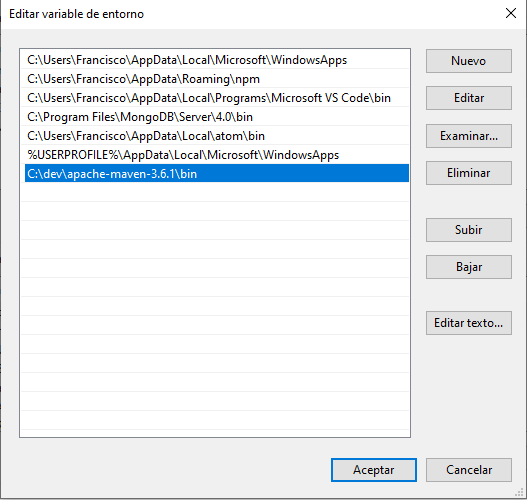
Se abrirá la ventana de propiedades sistema y debemos seleccionar la pestaña de opciones avanzadas y posteriormente presionar click en la opción de variables de entorno



Mostrará la ventana de configuración en donde nosotros podremos agregar o eliminar nuestras variables de entorno para efectos en esta ocasión agregaremos las variables de entorno de los archivos que descargamos de maven

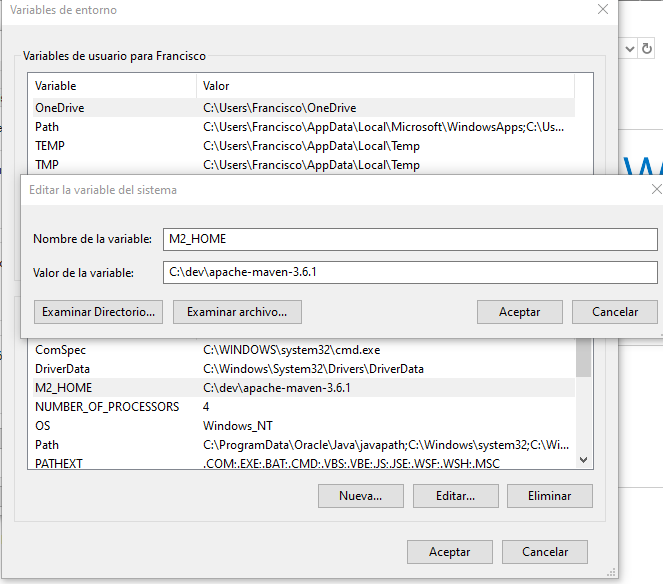


Como recomendación agregaremos las variables tanto variables de sistema y usuario, seleccionamos la opción path en editar

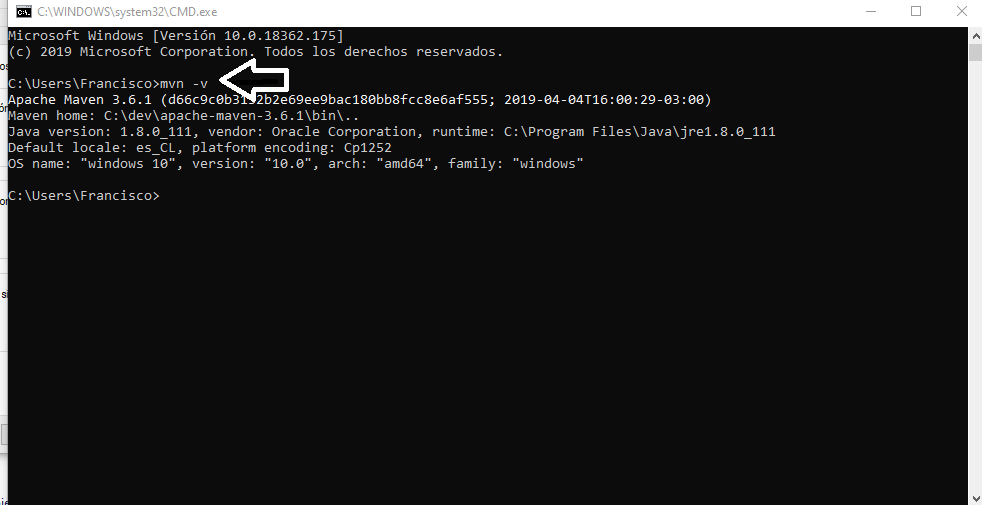


Visualizaremos una ventana como la siguiente presionaremos en examinar y seleccionamos la siguiente ruta c: \carpeta en donde esta maven\apache-maven-3.6.1\bin presionamos aceptar y visualizaremos como resultado lo que vemos en la imagen repetir el mismo procedimiento en variables del sistema.

Paso 4: como penúltimo paso crearemos una variable de entorno en el apartado de variables de sistema de nombre M2\_HOME seleccionamos examinar directorio y seleccionamos la ruta que aparece en la imagen.



Paso 5: como último paso validamos si todo está correctamente instalado, abrimos la consola de comandos de Windows CMD e ingresamos el comando mvn –v, si la prueba salió exitosa veremos el mismo resultado en donde muestra la versión de malven y el sistema operativo en el cual se está ejecutando.

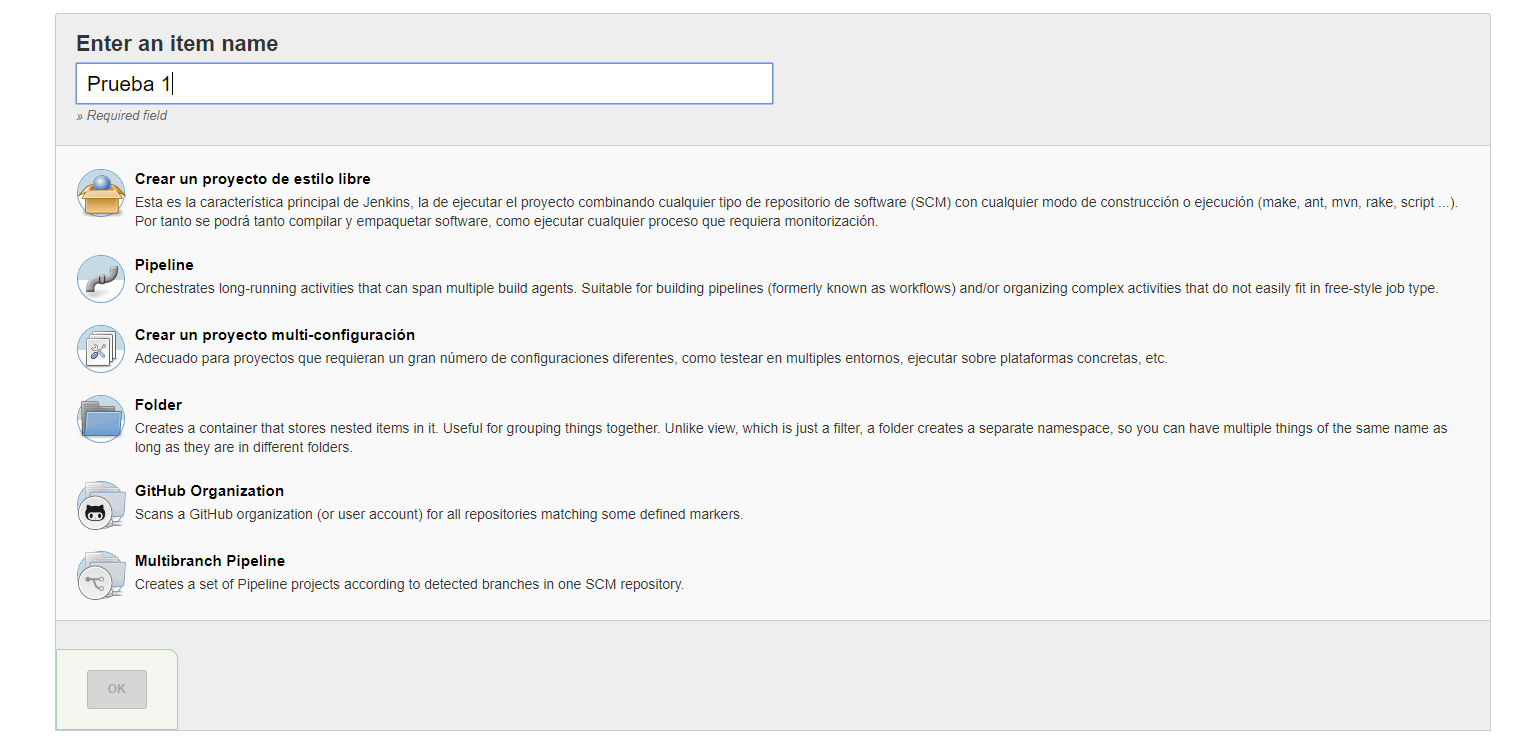


* 1. Realizar una prueba básica

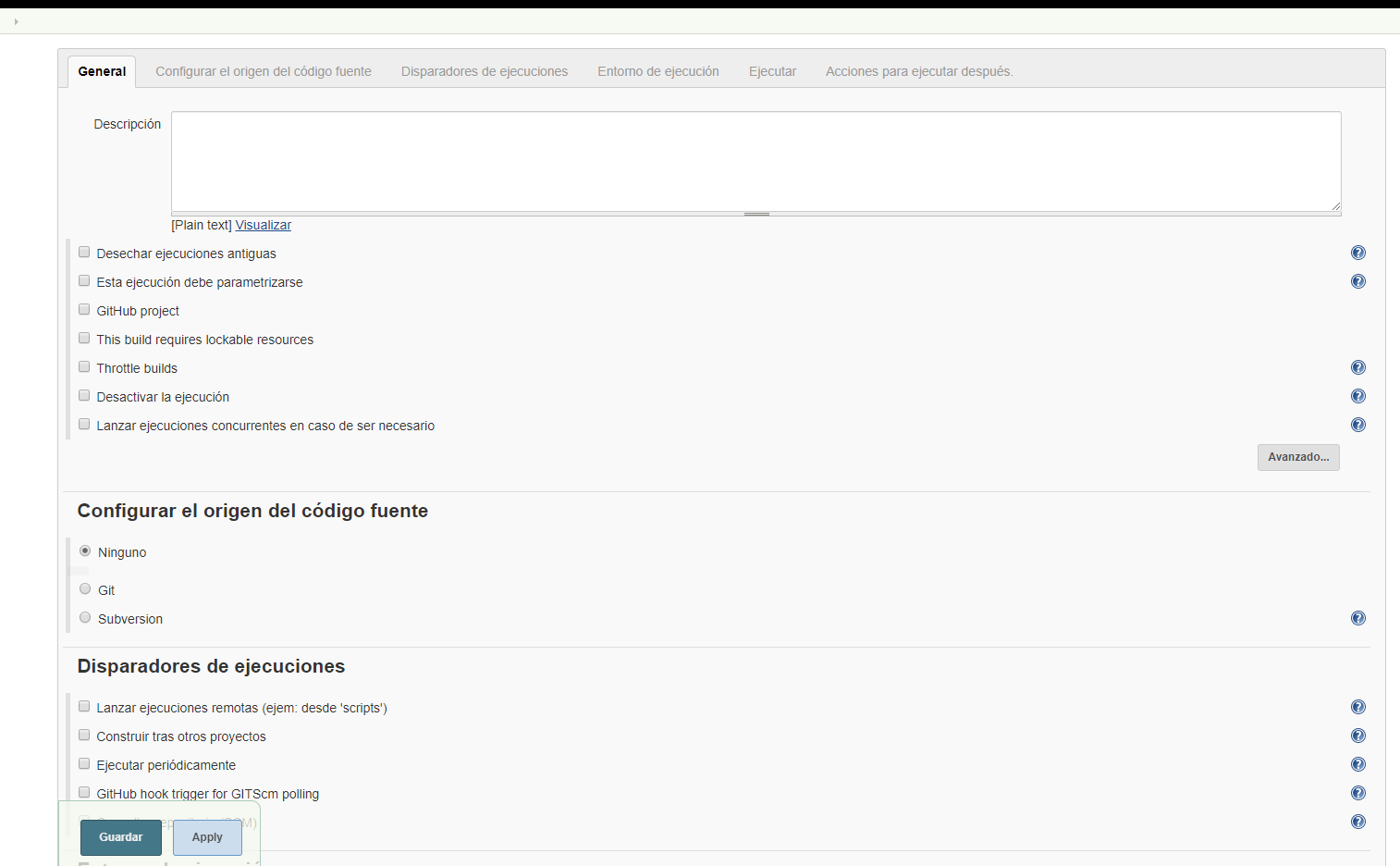
Paso 1: ingresamos a nuestro servidor de Jenkins localhost:8080, luego presionamos en crear nueva tarea.



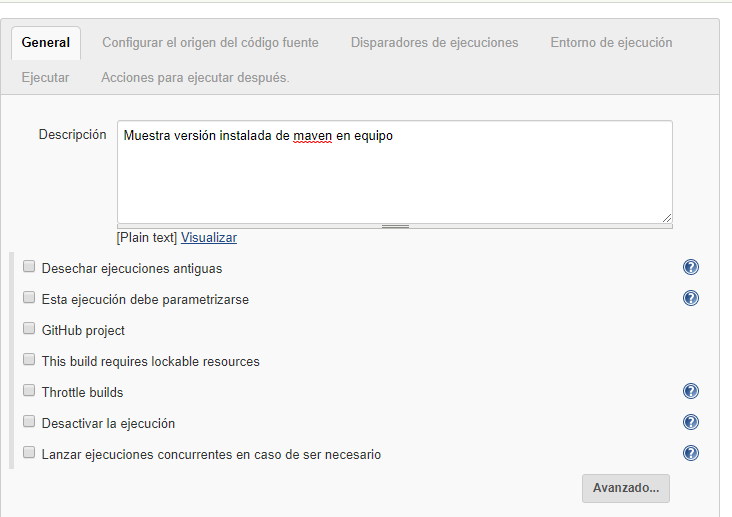
Paso 2: Colocamos el nombre a nuestra tarea y posteriormente seleccionamos proyecto de estilo libre y luego en ok



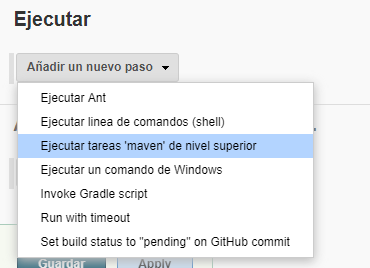
Una vez presionado ok veremos una pantalla como la que muestra en pantalla en donde podremos configurar nuestra tarea en este caso nos centraremos en las opciones de general y ejecutar.



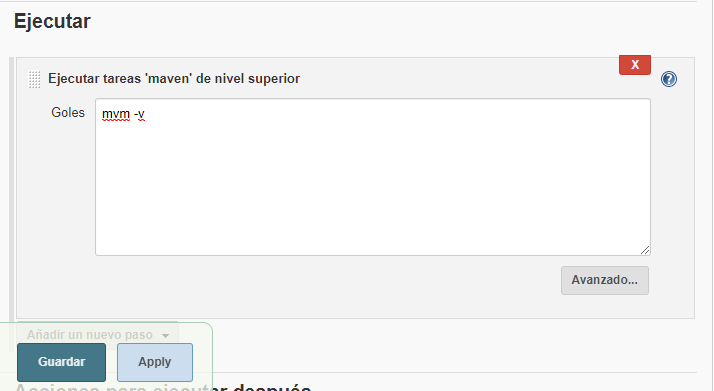
Paso 3: ingresamos una descripción de nuestra tarea de lo que realizara nuestra tarea.



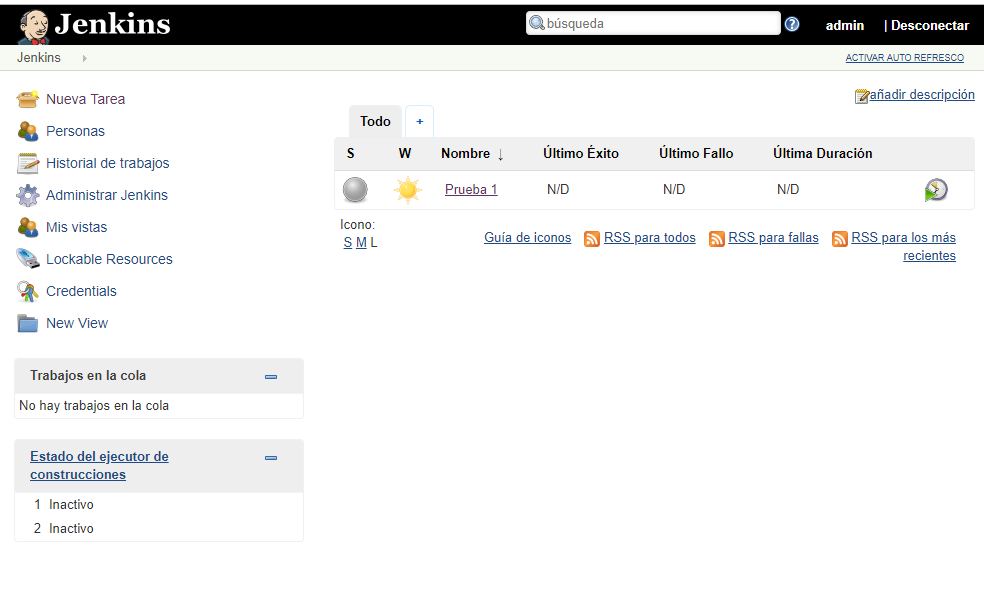
Paso 4: Una vez ingresada la descripción de nuestra tarea presionamos la opción de ejecutar y una vez dentro seleccionamos la opción ejecutar línea de comando

****

Nos aparecerá una ventana en donde ingresaremos el comando echo mvn -v y presionamos en guardar.



Luego nos dirigimos a la página principal en donde nos mostrara la lista de tareas

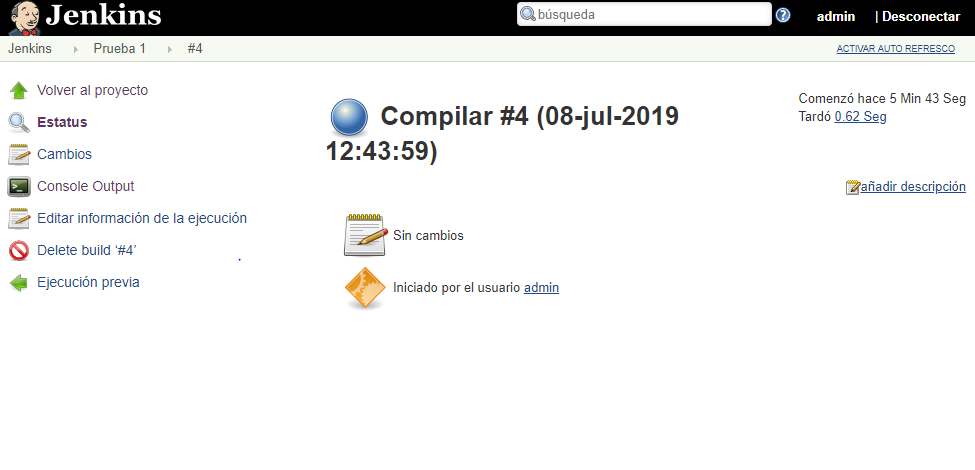


Presionamos el botón del reloj lo que ejecutara nuestra tarea y nos muestra los siguientes campos último éxito, ultimo fallo ultima duración 

Paso 5: presionamos, nuestra prueba y veremos las ultimas ejecuciones y presionamos la opción de enlaces permanentes “ultima ejecución”



Nos aparecerá un recuadro y posteriormente presionamos en console output y nos mostrara la información correspondiente de la versión de Maven





**RESPUESTAS ESPERADAS**

1. ¿Cuál es la importancia del desarrollo de software basado en la integración continua?

Una de las razones por la cual la integración continua es importante es la centralización el trabajo elaborado en nuestro código de esta manera tendremos la última versión escrita por nuestro equipo de trabajo. Por otro lado, nos permite la automatización de las pruebas mitigando los errores de forma temprana y preparar el paso a producción.

1. ¿Qué herramientas existen orientado a la integración continua?

* Jenkins
* Travis CI
* Bamboa
* GitLab CI
* CruiseControl
* Amazon Pipeline

1. ¿Cuáles son las desventajas de implementar proyectos de desarrollo sin integración continua?

R: desventaja de no implementar integración continua en el desarrollo de software se produce en el momento de que el proyecto empieza a crecer y a la vez el equipo de desarrollo provocando inconsistencias de código, en la realización de pruebas no se realizan de forma íntegra, el software se transforma poco escalable a futuro.