



Manual Técnico de Punto de Venta Sierra

Usuario: Administrador

Fecha: Lunes 20 de enero de 2020

Versión: Ver. 1.0

Proyecto: SW-PDV-Sierra

Elaborado por:

Certuit Software S. de R.L de C.V.

Cerro de las Campanas No. 384, Insurgentes Oeste, CP. 21280

Mexicali, BC. México

Historial de Cambios

Fecha	Secciones	Razón de Cambios	Autor(es)	Revisor(es)
20 de enero de 2020	TODAS	Creación del documento	Gael Cabrera	Alexa Gutierrez
20 de enero de 2020	Imagenes	Logo y acomodo de imágenes	Gael Cabrera	Alexa Gutierrez
22 de enero de 2020	Correcciones de especificaciones	Cambiar descripciones y especificaciones de software	Gael Cabrera	Naomi Durán
28 de enero de 2020	Pie de Imagen	Descripción en las imágenes	Gael Cabrera	Naomi Durán
28 de enero de 2020	1.4 y 1.5	Falta de relevancia	Alejandro Figueroa	Alejandro Figueroa
29 de enero de 2020	Inclusión de tema GIT	Falto anexar	Gael Cabrera	Alejandro Figueroa

Tabla de Contenido

Tabla de Contenido	3
Manual Técnico	4
1. Especificación técnica del sistema	4
1.1 Especificaciones de hardware	4
1.2 Especificaciones de software	4
1.3 Instrucciones de instalación/despliegue	5
1.3.1 Java	5
1.3.2.1 Software previo a instalación:	6
1.3.2.1.1 Instalar jdk	6
1.3.2.1.2 Variables de entorno jdk	7
1.3.1.2 Instalación de Apache Tomcat	8
1.3.3 Jenkins	15
1.3.4 GIT	15

Manual Técnico

1. Especificación técnica del sistema

El Sistema de PDV de Sierra es una aplicación que se utiliza para los pedidos que se generan en las instalaciones.

La siguiente documentación técnica describe los requisitos técnicos para desplegar el Sistema de Punto de Venta Sierra.

1.1 Especificaciones de hardware

Los requerimientos mínimos recomendados para el despliegue del Sistema de PDV de Sierra se expresan en la siguiente tabla:

Equipo	Procesadores	Memoria RAM	Tarjeta de red
VM-PDV-SIERRA (Servidor Sierra)	2 CPU	2 GB	1 Gigabit

El hardware sobre el cual se despliegue el sistema idealmente deberá encontrarse en un ambiente controlado que cuente con las siguientes especificaciones:

- **Control ambiental:** Se deberá contar con los sistemas de control de temperatura y humedad para la óptima operación de hardware.
- **Acceso controlado:** Debido a la importancia del sistema es necesario implementar alguna medida de acceso al sitio donde se hospede el hardware.
- **Protección contra incendios.**

1.2 Especificaciones de software

El software requerido para previamente realizar los procedimientos de instalación y despliegue descritos en este manual, se listan en la siguiente tabla:

Software	Descripción	Estatus
Windows Server 2012R2	Sistema Operativo	Instalado
Apache Tomcat 8.5	Servidor de aplicaciones Java	Instalado
Java 8	Máquina Virtual de Java	Instalado
MySQL(5.7)	Motor de Base de Datos	Instalado
Jenkins	Servidor de integración continua	Instalado
Gitlab	Versionamiento de código	Instalado

Los archivos de software requeridos para realizar los procedimientos de instalación y despliegue descritos en este manual, se listan en la siguiente tabla:

Archivo	Descripción	Licenciamiento
ROOT.war	Empaquetado de despliegue de la aplicación web.	N/A

Para ejecutar el despliegue de Git y Jenkins, se requiere del navegador de internet Google Chrome Versión 70.0.3538.102 (64 bits) o superior.

1.3 Instrucciones de instalación/despliegue

1.3.1 Java

1.3.1.1 Instalación

- 1) Ir a la página de descarga del manual <https://www.java.com/es/download/>.
- 2) Haga clic en **Windows en línea**.
- 3) Aparecerá el cuadro de diálogo Descarga de archivos y le pedirá que ejecute o guarde el archivo descargado
 - a) Para ejecutar el instalador, haga clic en **Ejecutar**.
 - b) Para guardar el archivo y ejecutarlo más tarde, haga clic en **Guardar**.
Seleccione la ubicación de la carpeta y guarde el archivo en el sistema local.
Sugerencia: guarde el archivo en una ubicación conocida de su equipo; por ejemplo, en el escritorio.
Haga doble clic en el archivo guardado para iniciar el proceso de instalación.
- 4) Se iniciará el proceso de instalación. Haga clic en el botón **Instalar** para aceptar los términos de la licencia y continuar con la instalación.
- 5) Oracle colabora con empresas que ofrecen distintos productos. Es posible que el instalador le ofrezca la opción de instalar estos programas como parte de la instalación de Java. Una vez seleccionados los programas que desee, haga clic en el botón **Siguiente** para proseguir con el proceso de instalación.
- 6) Se abrirán varios cuadros de diálogo con información para completar las últimas etapas del proceso de instalación; haga clic en **Cerrar** en el último cuadro de diálogo. Con esta acción se completará el proceso de instalación de Java.

1.3.2 Tomcat

Antes de instalar **Tomcat** debemos tener instalado el **Java Development Kit**. Lo descargamos de la dirección:

["http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html"](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html).

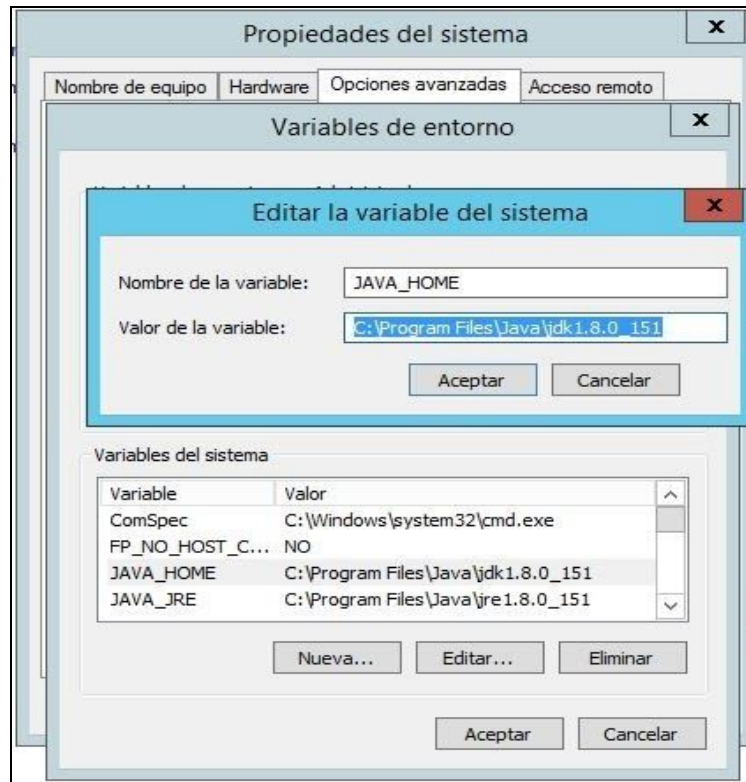
Java SE Development Kit 8u151		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.9 MB	jdk-8u151-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.85 MB	jdk-8u151-linux-arm64-vfp-hflt.tar.gz
Linux x86	168.95 MB	jdk-8u151-linux-i586.rpm
Linux x86	183.73 MB	jdk-8u151-linux-i586.tar.gz
Linux x64	166.1 MB	jdk-8u151-linux-x64.rpm
Linux x64	180.95 MB	jdk-8u151-linux-x64.tar.gz
macOS	247.06 MB	jdk-8u151-macosx-x64.dmg
Solaris SPARC 64-bit	140.06 MB	jdk-8u151-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.32 MB	jdk-8u151-solaris-sparcv9.tar.gz
Solaris x64	140.65 MB	jdk-8u151-solaris-x64.tar.Z
Solaris x64	97 MB	jdk-8u151-solaris-x64.tar.gz
Windows x86	198.04 MB	jdk-8u151-windows-i586.exe
Windows x64	205.95 MB	jdk-8u151-windows-x64.exe

Tipo de instaladores para JAVA según el OS

1.3.2.1 Software previo a instalación:

1.3.2.1.1 Instalar jdk

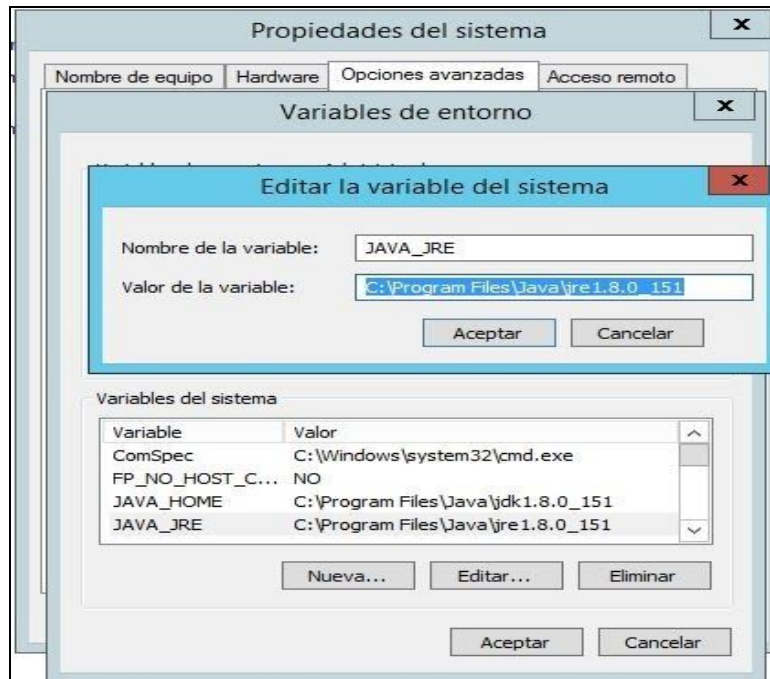
Establecer las dos **variables de entorno**. Hacemos clic en «**Equipo**» con el botón derecho y después en «**Propiedades**», «**Opciones avanzadas**», «**Variables de entorno**» y añadimos una nueva «**JAVA_HOME**» en «**Variables del sistema**» con el valor de la ruta donde está instalado el «**jdk de Java**»: «**C:\Program Files\Java\jdk1.8.0_151**».



Edición de Variables de Entorno (JAVA_HOME)

1.3.2.1.2 Variables de entorno jdk

Y lo mismo con la variable «**JAVA_JRE**», en la ruta del «**jre de Java**»: «**C:\Program Files\Java\jre1.8.0_151**».



Edición de Variables de Entorno (JAVA_JRE)

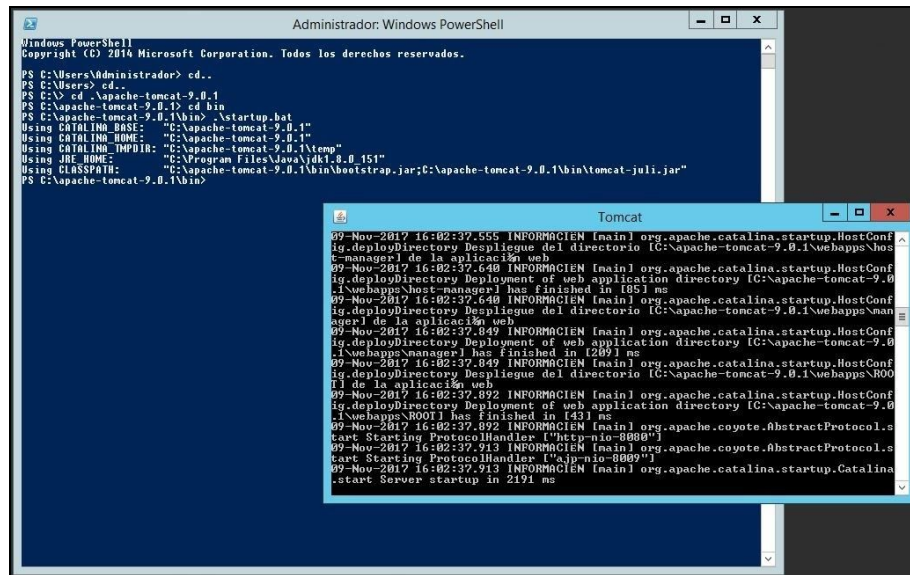
1.3.1.2 Instalación de Apache Tomcat

1. Después de reiniciar la máquina, **descargamos Apache Tomcat** (de preferencia Tomcat 8) de la dirección: "<https://tomcat.apache.org/download-70.cgi>", lo descomprimos y lo copiamos en c:\.
2. Instalamos el servicio **tomcat 8** desde consola. Debemos colocarnos en el directorio "**C:\apache-tomcat-8.0.1\bin**" y escribir: "**.\service.bat install**".

```
PS C:\apache-tomcat-9.0.1\bin> .\service.bat install
Installing the service 'Tomcat9' ...
Using CATALINA_HOME: "C:\apache-tomcat-9.0.1"
Using CATALINA_BASE: "C:\apache-tomcat-9.0.1"
Using JAVA_HOME: "C:\Program Files\Java\jdk1.8.0_151"
Using JRE_HOME: "C:\Program Files\Java\jdk1.8.0_151\jre"
Using JUM: "C:\Program Files\Java\jdk1.8.0_151\jre\bin\server\jum.dll"
The service 'Tomcat9' has been installed.
PS C:\apache-tomcat-9.0.1\bin>
```

Pantalla de instalación Tomcat 8

3. Para iniciar el **servicio Tomcat** debemos colocarnos en el directorio "**C:\apache-tomcat-8.0.1\bin**" y escribir: "**.\startup.bat**".



```

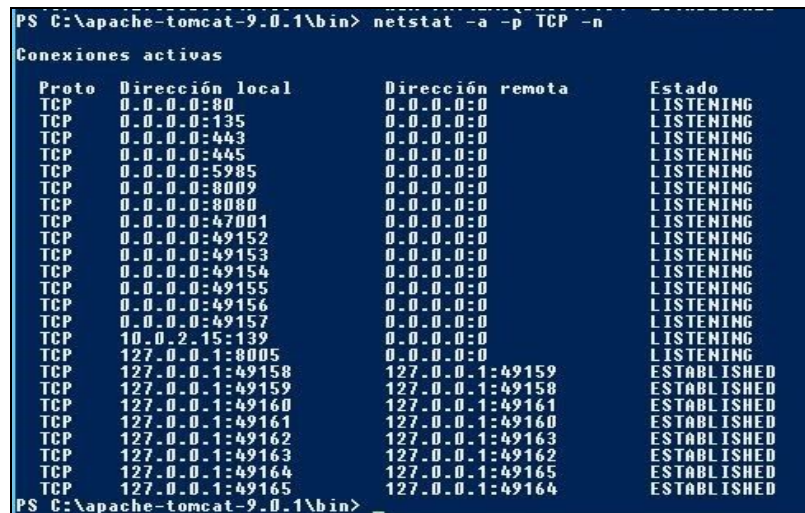
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\Administrador> cd..
PS C:\Users> cd..
PS C:\> cd \apache-tomcat-9.0.1
PS C:\apache-tomcat-9.0.1> cd bin
PS C:\apache-tomcat-9.0.1\bin> .\startup.bat
Using CATALINA_HOME: "C:\apache-tomcat-9.0.1"
Using CATALINA_BASE: "C:\apache-tomcat-9.0.1"
Using CATALINA_TMPDIR: "C:\apache-tomcat-9.0.1\temp"
Using JRE_HOME: "C:\Program Files\Java\jdk1.8.0_151"
Using CLASSPATH: "C:\apache-tomcat-9.0.1\bin\bootstrap.jar;C:\apache-tomcat-9.0.1\bin\tomcat-juli.jar"
PS C:\apache-tomcat-9.0.1\bin>

Tomcat
09-Nov-2017 16:02:37.555 INFORMACION [main] org.apache.catalina.startup.HostConfig
ig.deployDirectory Despliegue del directorio [C:\apache-tomcat-9.0.1\webapps\hos
t-manager] de la aplicación web
09-Nov-2017 16:02:37.648 INFORMACION [main] org.apache.catalina.startup.HostConf
ig.deployDirectory Deployment of web application directory [C:\apache-tomcat-9.0
1\webapps\host-manager] has finished in 185 ms
09-Nov-2017 16:02:37.648 INFORMACION [main] org.apache.catalina.startup.HostConf
ig.deployDirectory Despliegue del directorio [C:\apache-tomcat-9.0.1\webapps\man
ager] de la aplicación web
09-Nov-2017 16:02:37.649 INFORMACION [main] org.apache.catalina.startup.HostConf
ig.deployDirectory Deployment of web application directory [C:\apache-tomcat-9.0
1\webapps\manager] has finished in 120 ms
09-Nov-2017 16:02:37.649 INFORMACION [main] org.apache.catalina.startup.HostConf
ig.deployDirectory Despliegue del directorio [C:\apache-tomcat-9.0.1\webapps\ROO
T] de la aplicación web
09-Nov-2017 16:02:37.692 INFORMACION [main] org.apache.catalina.startup.HostConf
ig.deployDirectory Deployment of web application directory [C:\apache-tomcat-9.0
1\webapps\ROOT] has finished in 43 ms
09-Nov-2017 16:02:37.692 INFORMACION [main] org.apache.coyote.AbstractProtocol.s
tart Starting ProtocolHandler ["http-nio-8080"]
09-Nov-2017 16:02:37.713 INFORMACION [main] org.apache.coyote.AbstractProtocol.s
tart Starting ProtocolHandler ["ajp-nio-8009"]
09-Nov-2017 16:02:37.913 INFORMACION [main] org.apache.catalina.startup.Catalina
.start Server startup in 2191 ms
  
```

Servicio de Tomcat

- Validar que el puerto 8080 se encuentre abierto. Tecleamos el comando “netstat”, que permite monitorizar las conexiones existentes, con las opciones “-a” (para que las muestre todas), “-p TCP” (para que filtre las que son de protocolo TCP) y “-n” (para que muestre las direcciones de forma numérica). “netstat -a -p TCP -n”.



```

PS C:\apache-tomcat-9.0.1\bin> netstat -a -p TCP -n
Conexiones activas

Proto  Dirección local      Dirección remota      Estado
TCP    0.0.0.0:80            0.0.0.0:0             LISTENING
TCP    0.0.0.0:135           0.0.0.0:0             LISTENING
TCP    0.0.0.0:443           0.0.0.0:0             LISTENING
TCP    0.0.0.0:445           0.0.0.0:0             LISTENING
TCP    0.0.0.0:5985          0.0.0.0:0             LISTENING
TCP    0.0.0.0:8009          0.0.0.0:0             LISTENING
TCP    0.0.0.0:8080          0.0.0.0:0             LISTENING
TCP    0.0.0.0:47001         0.0.0.0:0             LISTENING
TCP    0.0.0.0:49152         0.0.0.0:0             LISTENING
TCP    0.0.0.0:49153         0.0.0.0:0             LISTENING
TCP    0.0.0.0:49154         0.0.0.0:0             LISTENING
TCP    0.0.0.0:49155         0.0.0.0:0             LISTENING
TCP    0.0.0.0:49156         0.0.0.0:0             LISTENING
TCP    0.0.0.0:49157         0.0.0.0:0             LISTENING
TCP    10.0.2.15:139         0.0.0.0:0             LISTENING
TCP    127.0.0.1:8005        0.0.0.0:0             LISTENING
TCP    127.0.0.1:49158       127.0.0.1:49159       ESTABLISHED
TCP    127.0.0.1:49159       127.0.0.1:49158       ESTABLISHED
TCP    127.0.0.1:49160       127.0.0.1:49161       ESTABLISHED
TCP    127.0.0.1:49161       127.0.0.1:49160       ESTABLISHED
TCP    127.0.0.1:49162       127.0.0.1:49163       ESTABLISHED
TCP    127.0.0.1:49163       127.0.0.1:49162       ESTABLISHED
TCP    127.0.0.1:49164       127.0.0.1:49165       ESTABLISHED
TCP    127.0.0.1:49165       127.0.0.1:49164       ESTABLISHED
PS C:\apache-tomcat-9.0.1\bin>
  
```

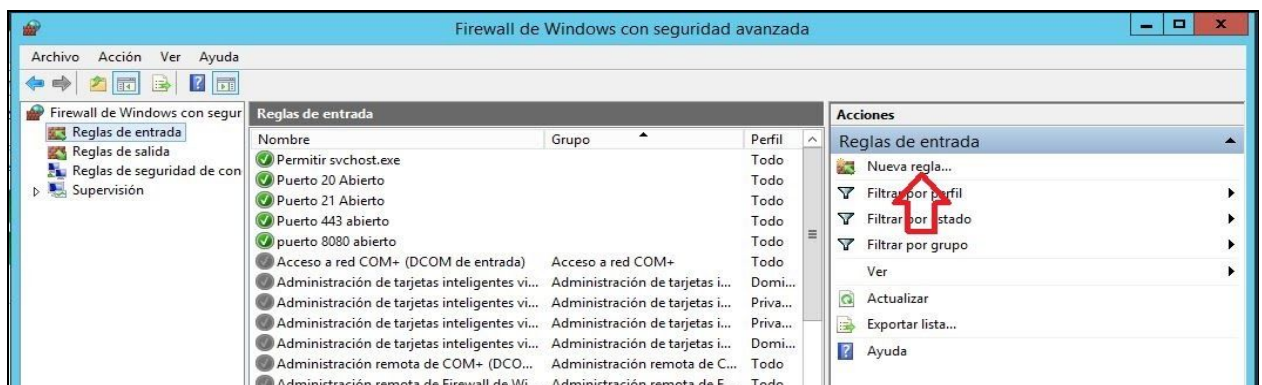
Verificación de Servicio Tomcat

5. Debemos permitir las conexiones entrantes en el puerto **8080** en el **firewall**. Para ello debemos acceder a la configuración del firewall de Windows
6. Para hacerlo vamos al **panel de control** y hacemos clic en «**Sistema y seguridad**»
7. Luego en «**Firewall de Windows**»
8. Y por último en «**Configuración avanzada**»



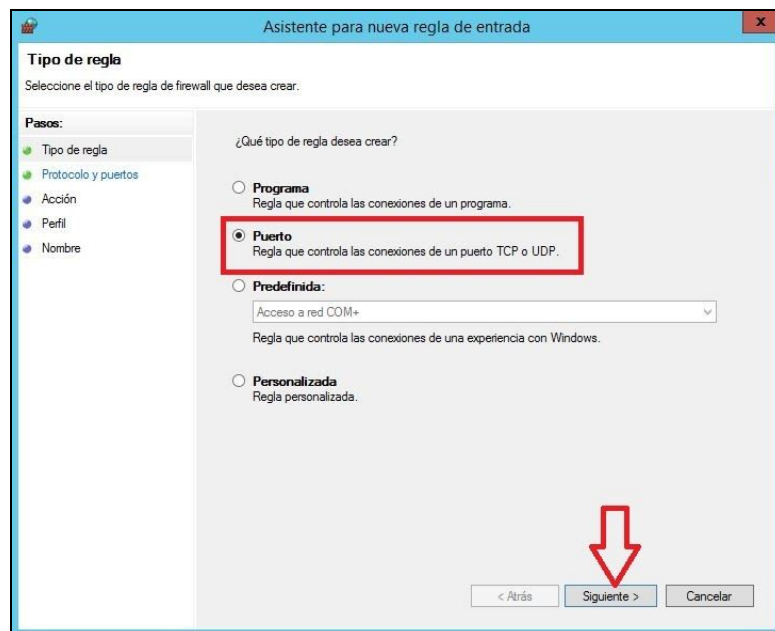
Menu de Firewall Windows

9. Crear una nueva regla de entrada. Hacemos clic en «**Reglas de entrada**»
10. Luego en «**Nueva Regla**»



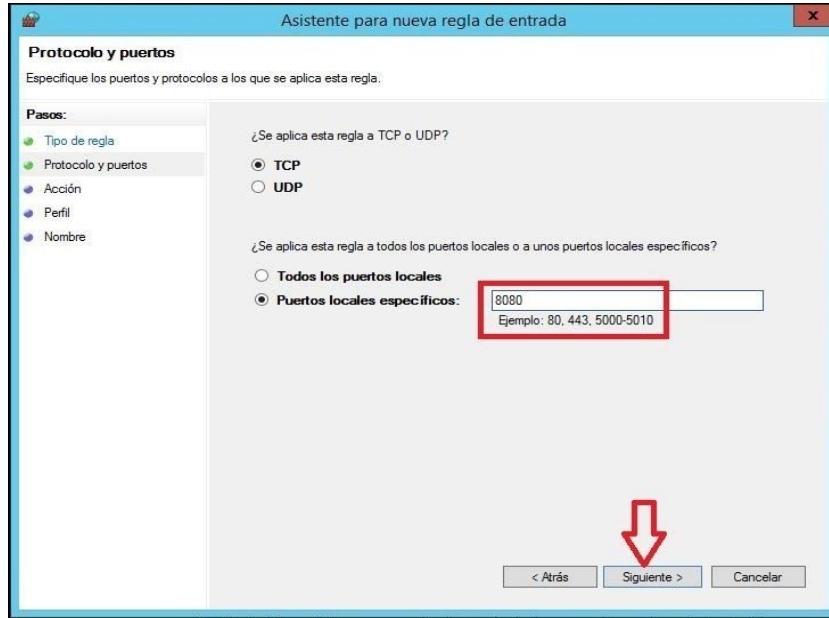
Creación de nueva regla

11. Activamos la casilla de «Puerto» y hacemos clic en «Siguiente»



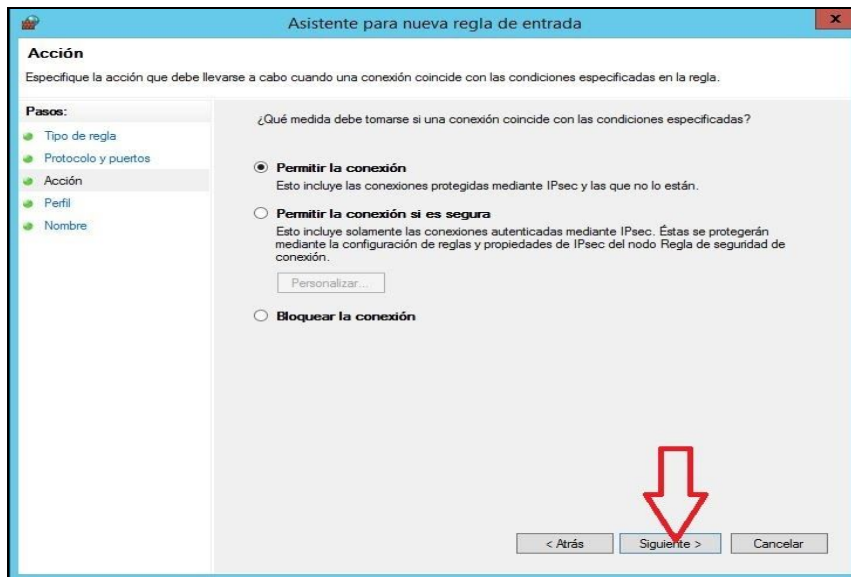
Wizard de regla de entrada

12. Escribimos el puerto **8080** y hacemos clic en «**Siguiente**»



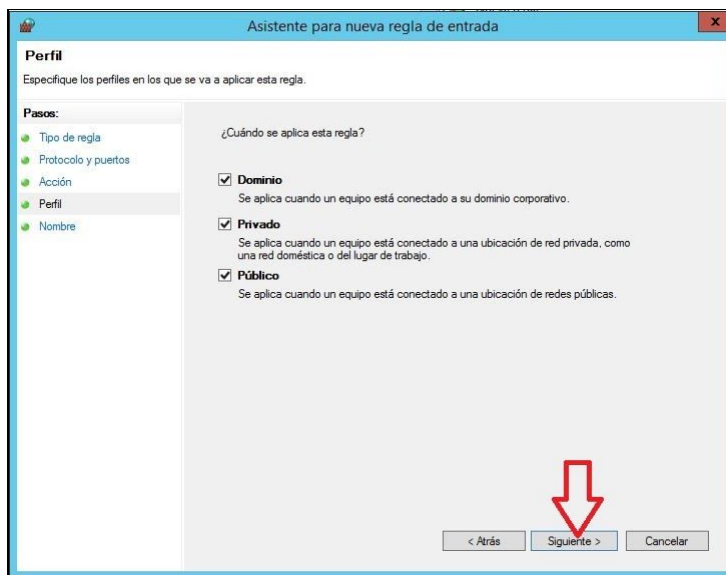
Wizard de regla de entrada puerto 8080

13. Hacemos clic en «**Siguiente**»



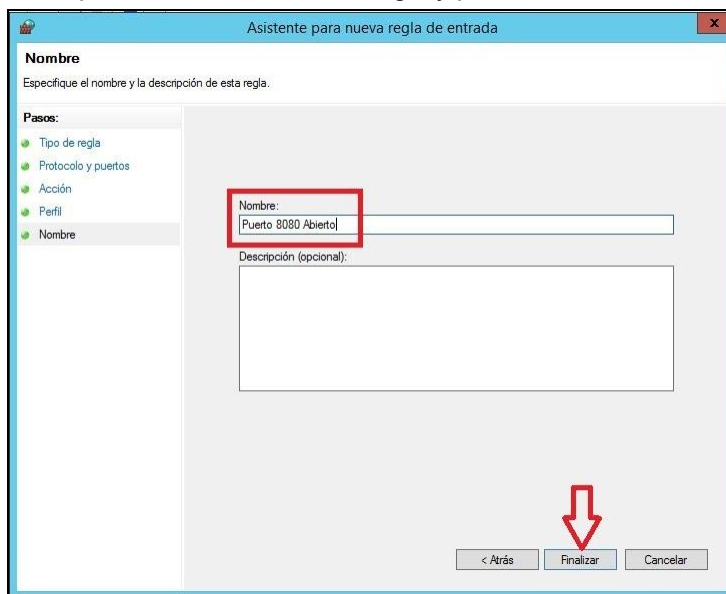
Wizard de regla de entrada tipo de conexión

14. De nuevo en «**Siguiente**»



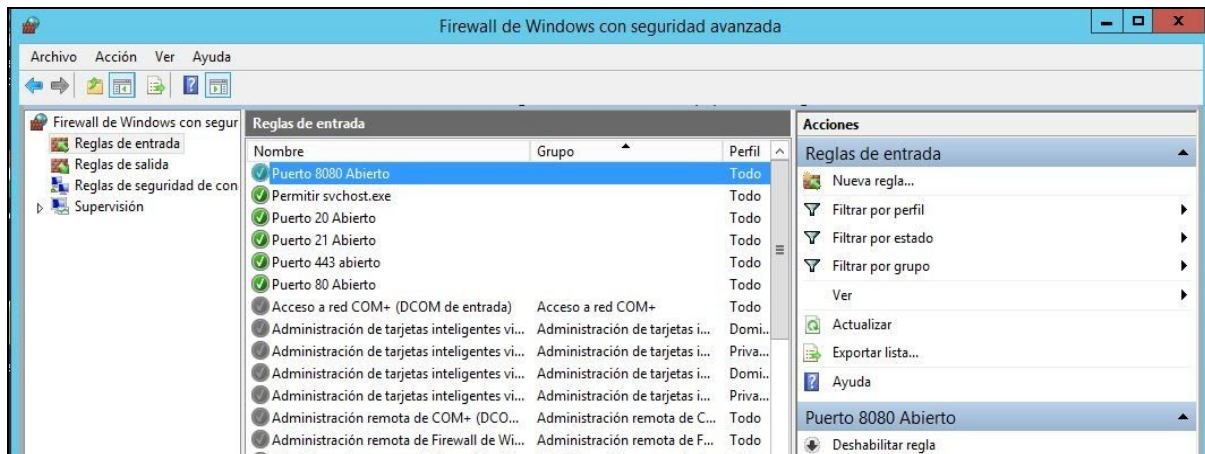
Wizard de regla de entrada perfiles

15. Escribimos un **nombre** para identificar la nueva regla y por último hacemos clic en «**Finalizar**»



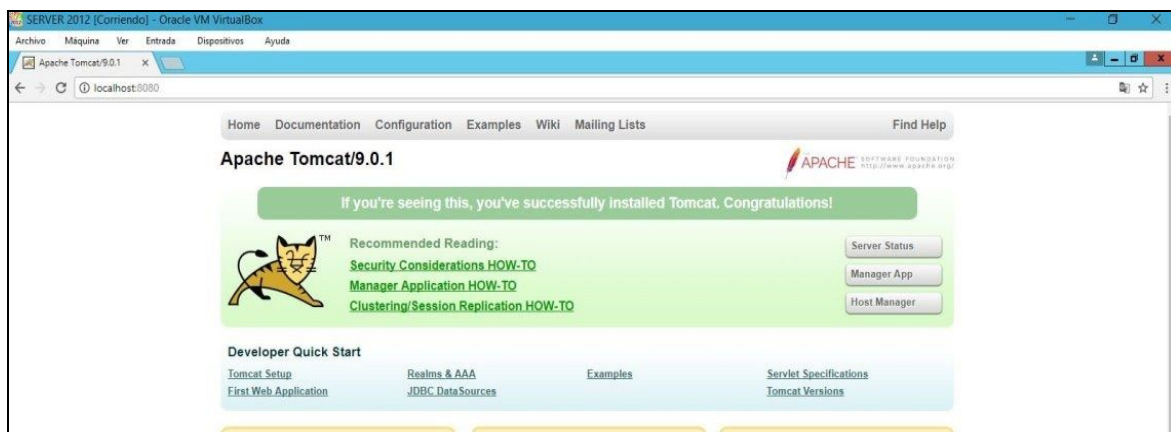
Wizard de regla de entrada nombre

16. Al finalizar podemos ver que nuestra nueva **regla de entrada** ya se encuentra en la lista



Ver reglas de entrada

17. Comprobaremos el funcionamiento en local escribiendo en el navegador: **"http://localhost:8080"**.



Prueba de funcionamiento Tomcat

1.3.3 Jenkins

Jenkins es un servidor de integración continua, open-source y actualmente uno de los más empleados para esta función. Además es muy fácil de utilizar.

La base de Jenkins son las tareas, donde indicamos qué es lo que hay que hacer en un build. Por ejemplo, podríamos programar una tarea en la que se compruebe el repositorio de control de versiones cada cierto tiempo, y cuando un desarrollador quiera subir su código al control de versiones, este se compile y se ejecuten las pruebas.

Si el resultado no es el esperado o hay algún error, Jenkins notificará al desarrollador, al equipo de QA, por email, para que lo solucione. Si el build es correcto, podremos indicar a Jenkins que intente integrar el código y subirlo al repositorio de control de versiones.

Desde Jenkins podrás indicar que se lancen métricas de calidad y visualizar los resultados dentro de la misma herramienta. También podrás ver el resultado de los tests, generar y visualizar la documentación del proyecto o incluso pasar una versión estable del software al entorno de QA para ser probado, a pre-producción o producción.

1.3.4 GIT

SISTEMA DE CONTROL DE VERSIONES

Git, es un software de control de versiones diseñado por Linus Torvalds. Se define como control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo es decir a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración, resumido, control de versiones es lo que se hace al momento de estar desarrollando un software o una página web.

Características de GIT

1. Rapidez en la gestión de ramas, debido a que Git nos dice que un cambio será fusionado mucho más frecuentemente de lo que se escribe originalmente.
2. Gestión distribuida; Los cambios se importan como ramas adicionales y pueden ser fusionados de la misma manera como se hace en la rama local.
3. Gestión eficiente de proyectos grandes.
4. Realmacenamiento periódico en paquetes.