UF 2.- Administración de servidores Web

Actividad 2.- Configuración de servidor

Contenido

[Enunciado 2](#_Toc63071121)

[Desarrollo actividad 3](#_Toc63071122)

[Requerimiento 1 3](#_Toc63071123)

[1.- Instalación Java 3](#_Toc63071124)

[2.- Instalación Apache 5](#_Toc63071125)

[3.- Instalación Tomcat 5](#_Toc63071126)

[4.- Instalación openSSH 8](#_Toc63071127)

[5.- Instalación MariaDB 9](#_Toc63071128)

[Requerimiento 2 10](#_Toc63071129)

[2.1.- Apache 10](#_Toc63071130)

[2.2.- Tomcat 10](#_Toc63071131)

[2.3.- MariaDb 11](#_Toc63071132)

# Enunciado

Requerimiento 1

Queremos preparar nuestro servidor Linux para poder desplegar una aplicación web. Para ello tenemos que verificar que están instalados:

* Java
* Apache
* Tomcat
* openSSH
* MariaDB

Valoración: 6 puntos sobre 10

Requerimiento 2

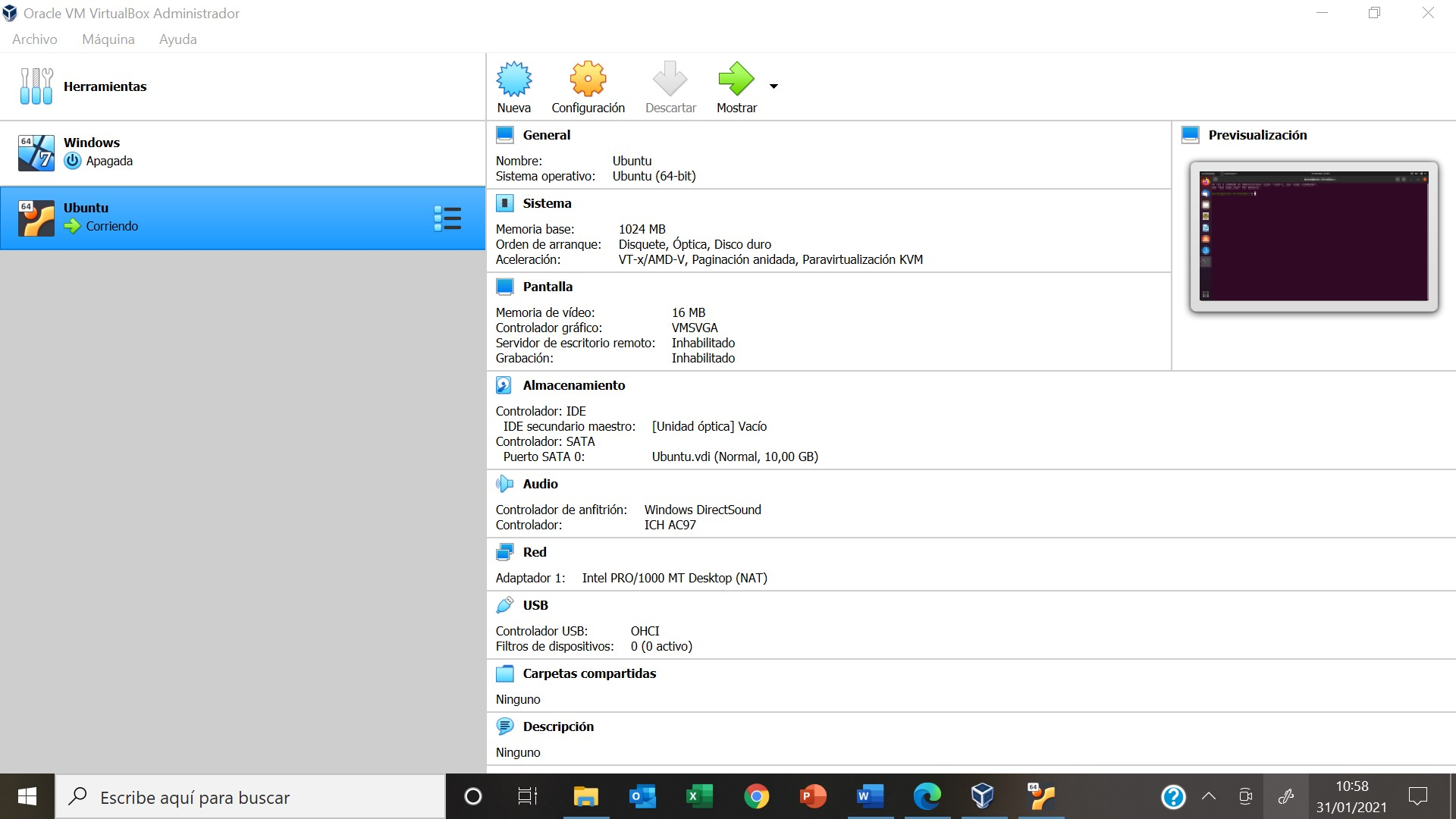
Así mismo, queremos asegurarnos de que los servidores están bien configurados y son accesibles antes de desplegar la aplicación. Por ello debemos configurar y comprobar que los puertos asociados a Apache, Tomcat y MariaDB están abiertos en el Firewall y son accesibles desde el exterior.

Valoración: 4 puntos sobre 10

# Desarrollo actividad

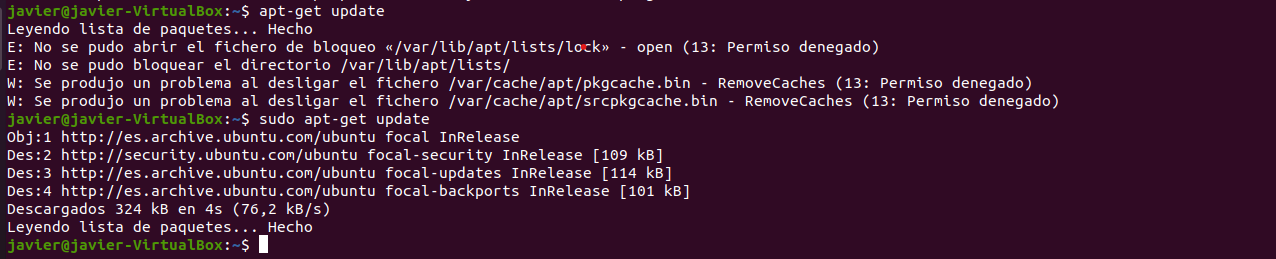
Para el desarrollo de esta actividad hemos instalado la Máquina Virtual VirtualBox y hemos instalado:

* Una máquina con Windows 10
* Otra máquina con la distribución Linux Ubuntu 20.04.1

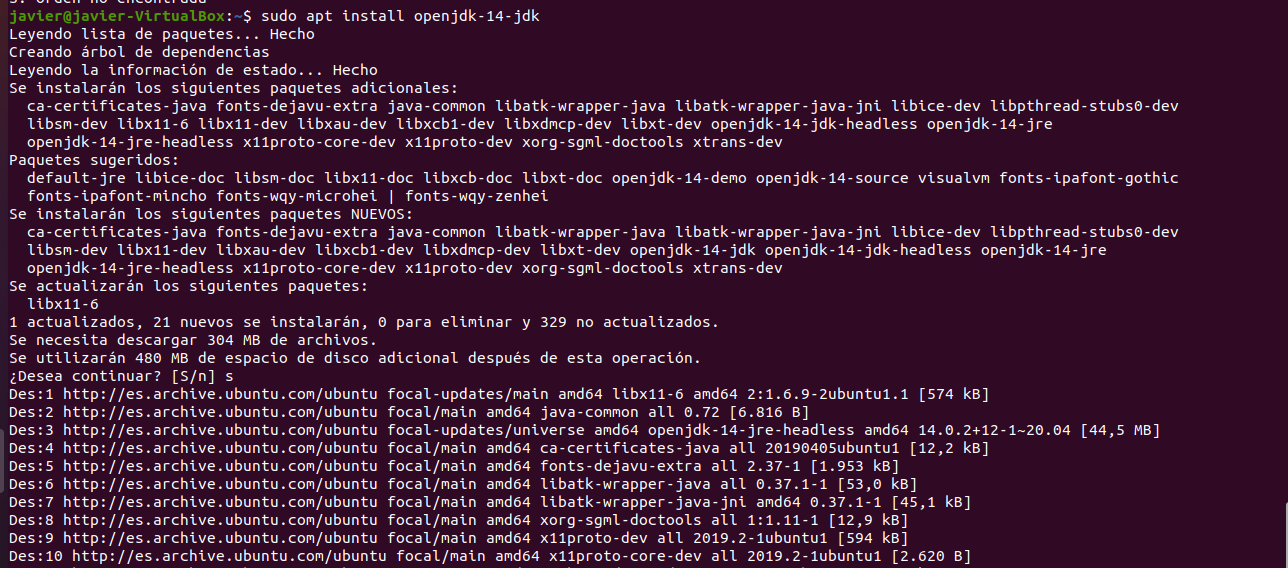


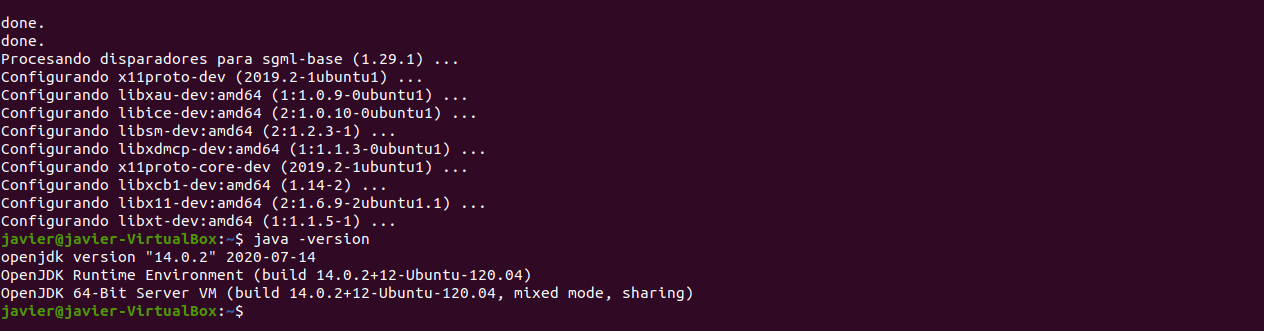
## Requerimiento 1

A continuación, vamos a proceder a instalar los diferentes programas, para ello hemos abierto la consola, para la instalación utilizaremos APT.

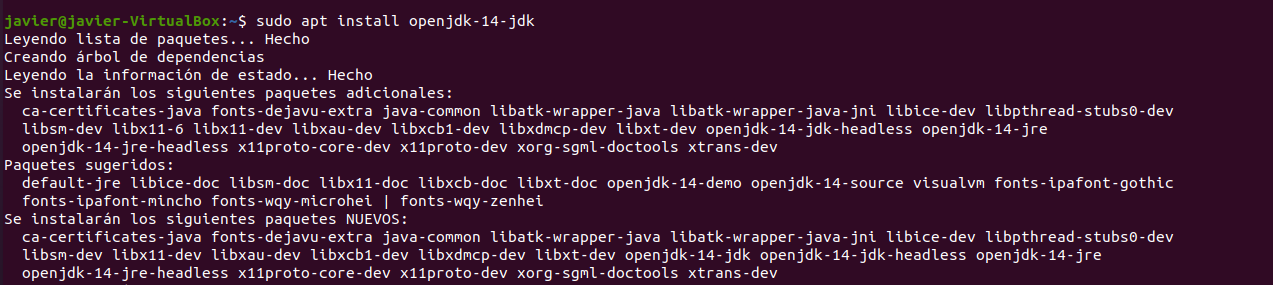
Lo primero es actualizar APT con la instrucción sudo apt.get update

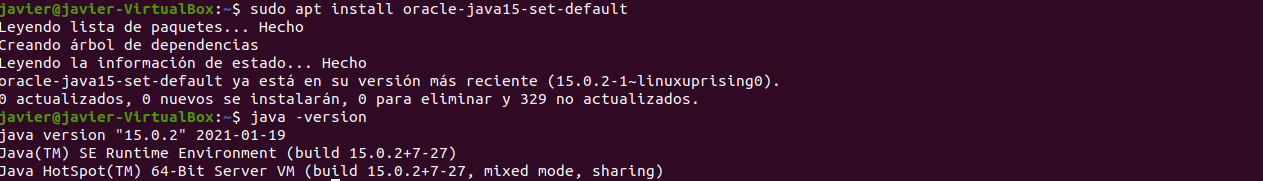
### 1.- Instalación Java

* Lo primero que vamos a hacer es buscar los paquetes disponibles de JDK, para ello introducimos en la consola lo siguiente:
  + sudo apt search openjdk
  + Nos solicita la contraseña de superusuario, la introducimos.
* Segundo paso, instalamos la versión 14, para ello introducimos
  + sudo apt install openjdk-14-jdk
  + Comienza la instalación.
  + Nos pregunta si deseamos continuar y le decimos que si.
* Paso 3: Comprobamos la versión instalada, para ello introducimos
  + Java -version

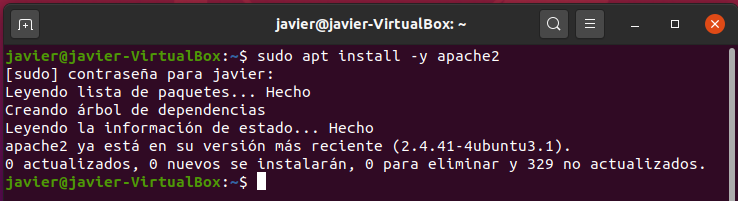
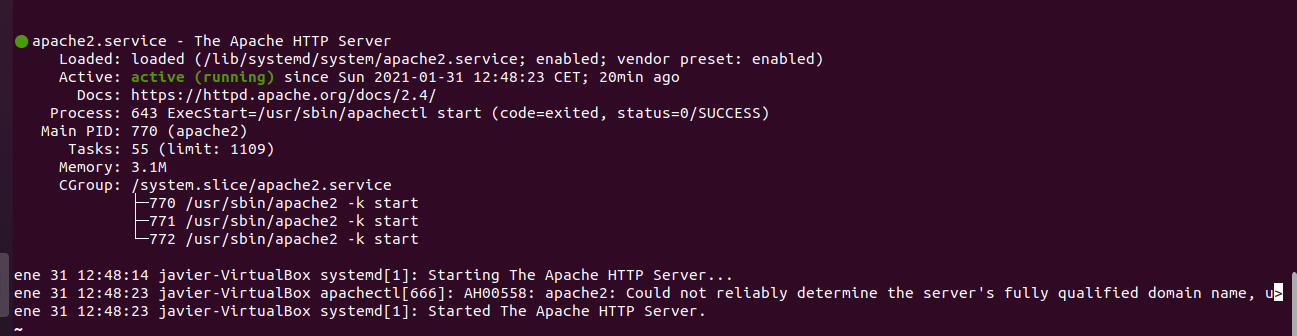


* Paso 4: Instalamos el repositorio. Para ello introducimos:
  + sudo add-apt-repository ppa:linuxuprising/java



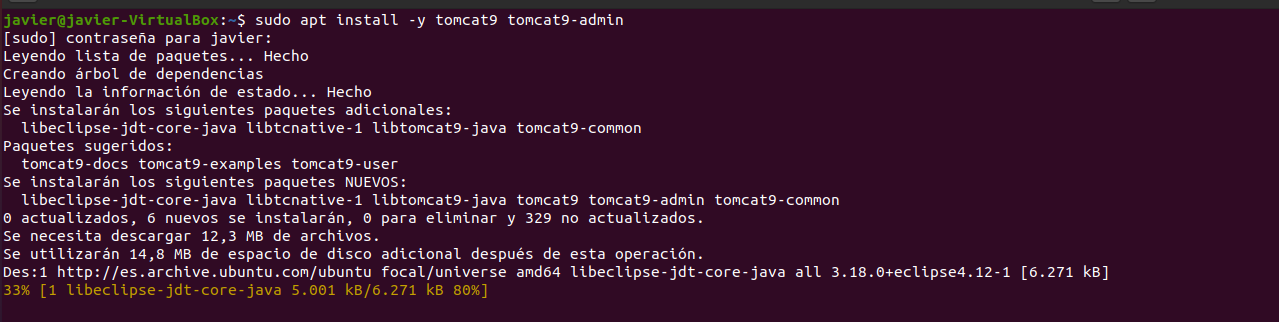
* Ejecutamos: sudo apt install oracle-java15-set-default
* Aceptamos los términos de la licencia
* Finaliza la instalación, comprobamos versión instalada con: java -version

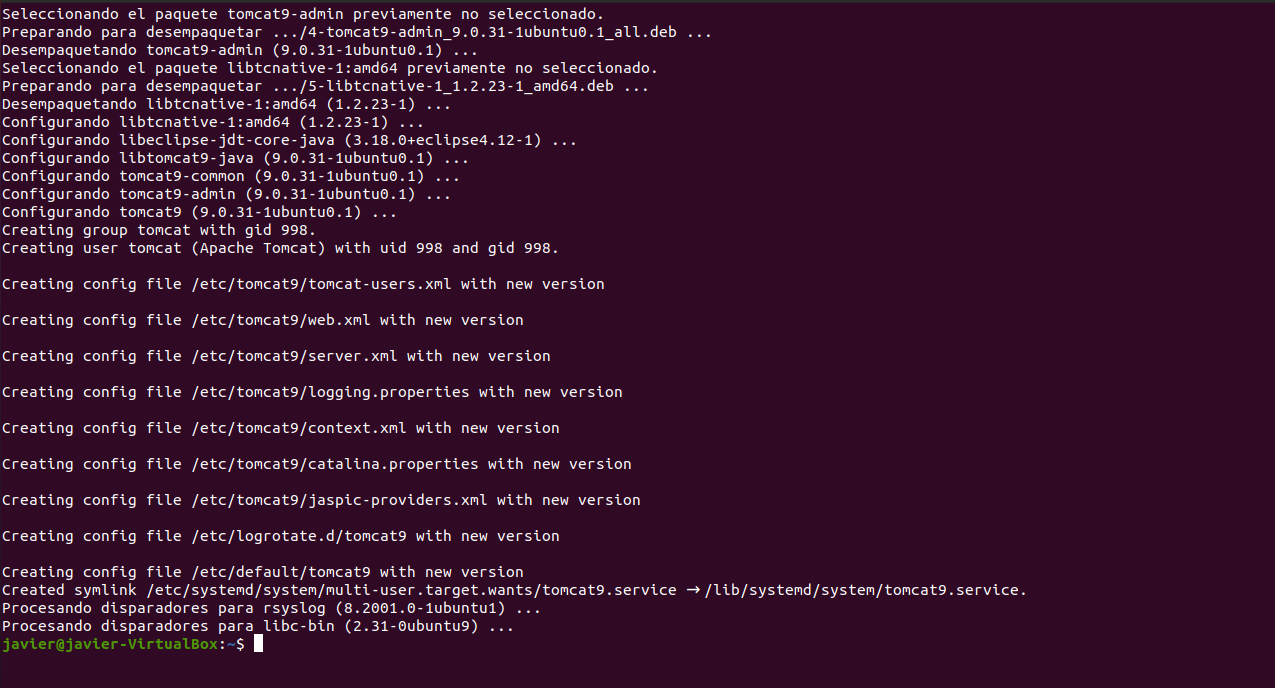
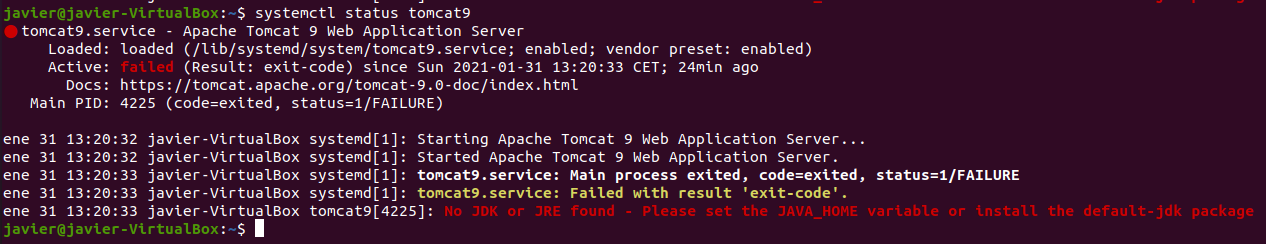
### 2.- Instalación Apache

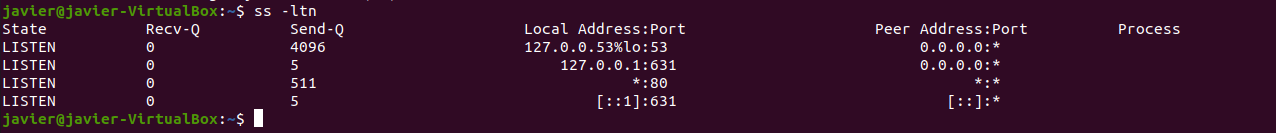
* Lo primero es actualizar los repositorios de la distribución. Nosotros ya lo hemos hecho anteriormente, para hacerlo consignamos: sudo apt update
* Para proceder a la instalación de apache consignamos:
  + Sudo apt install -y apache2
* Se ha quedado la máquina virtual colgada, adjunto captura de comprobante que ya ha finalizado el proceso de instalación
* Para comprobar el estado del Apache introducimos: systemctl status apache2

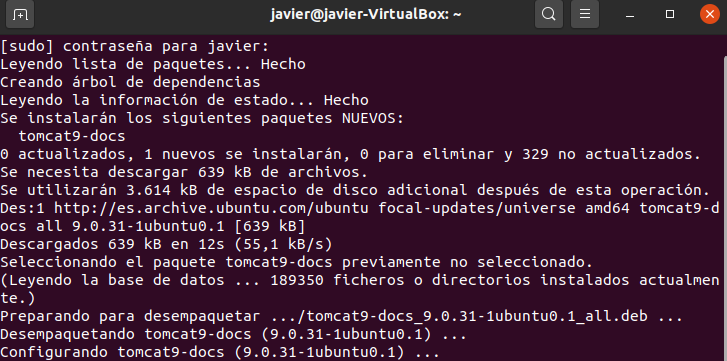
### 3.- Instalación Tomcat

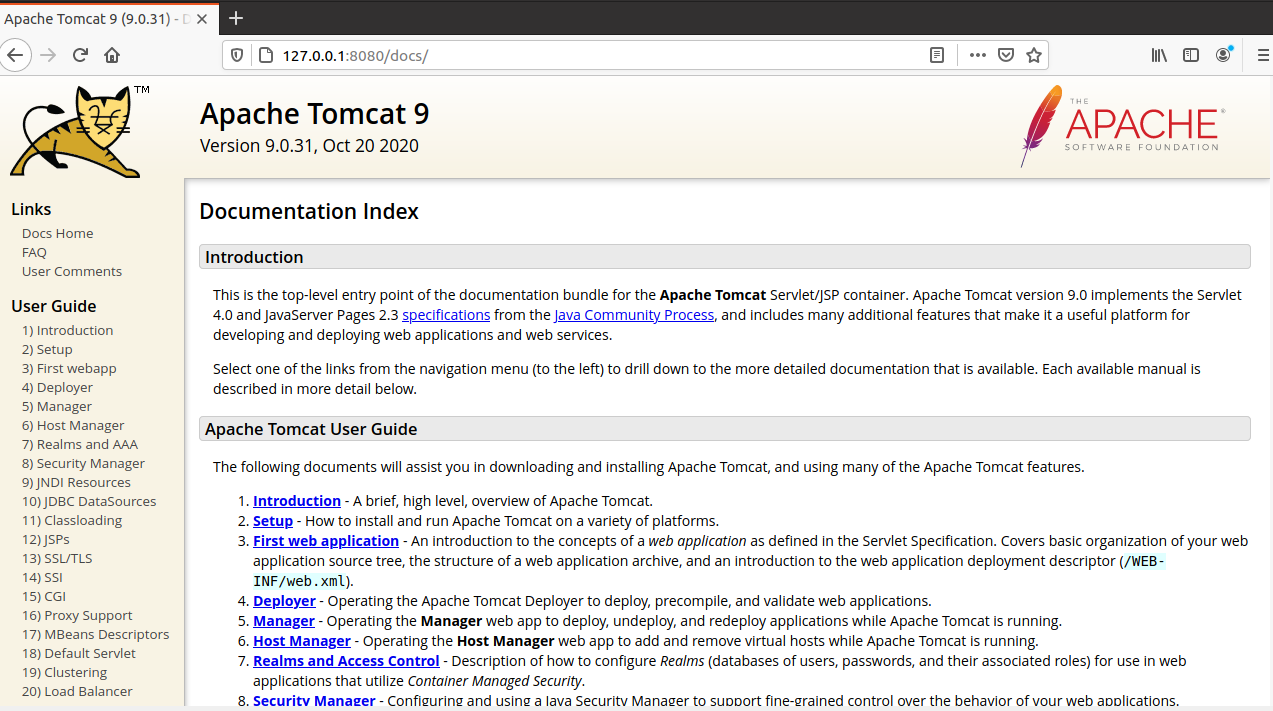
* Vamos a instalar el tomcat, para ello consignamos:
  + Sudo apt install -y tomcat9 tomcat9-admin



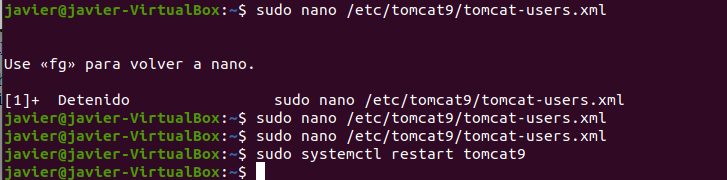
* Se completa
* Comprobamos el estado del tomcat
  + Systemctl status tomcat9
* Comprobamos que el puerto está escuchando



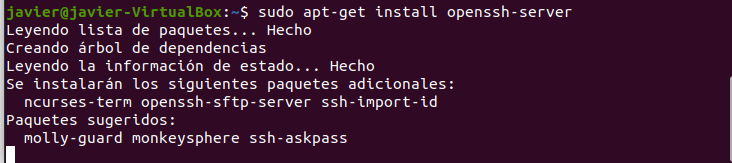
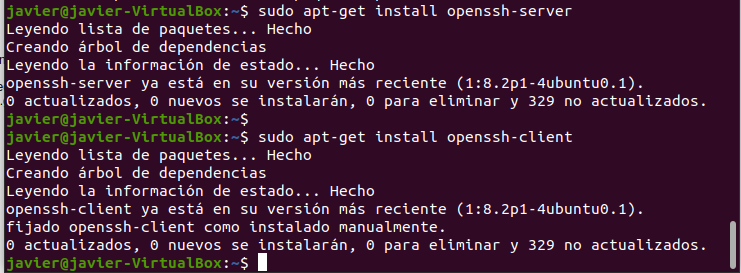
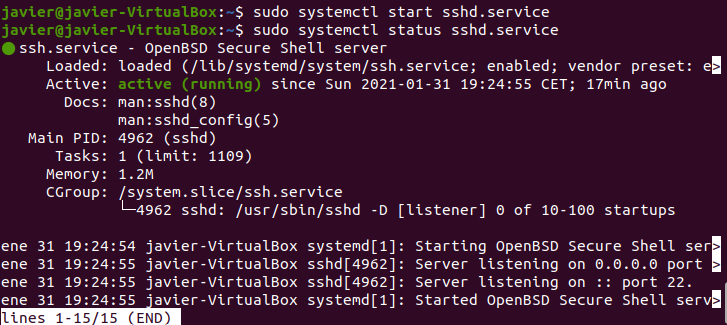
Ahora vamos a instalar tomcat9 – docs, para ello introducimos: sudo apt-get install tomcat9-docs



* Lo siguiente es instalar los examples, para ello consignamos sudo apt-get install tomcat9-examples.
* Por último, vamos a crear el usuario, para ello abrimos el archivo de usuarios xml del tomcat, a través de la instrucción: sudo nano /etc/tomcat9/tomcat-users.xml y escribimos el usuario y la contraseña. En el archivo XML incluimos los roles manager-gui y admin-gui con su nombre y contraseña
* Cuando terminamos, para que los cambios se apliquen tenemos que reiniciar el tomcat:
  + sudo systemctl restart tomcat9

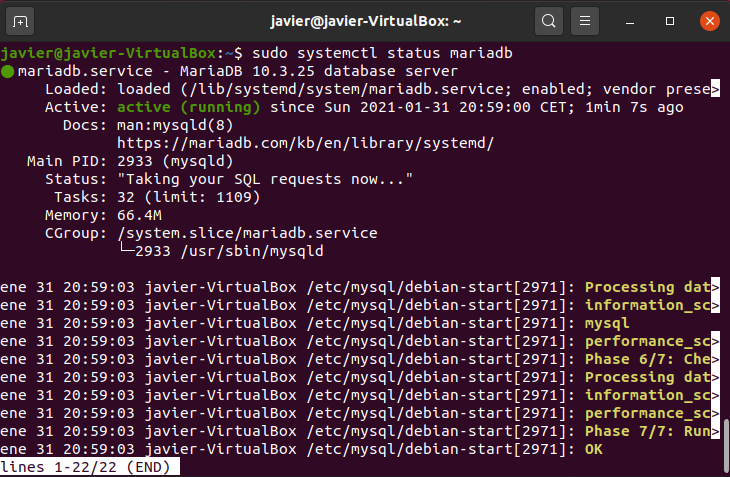


### 4.- Instalación openSSH

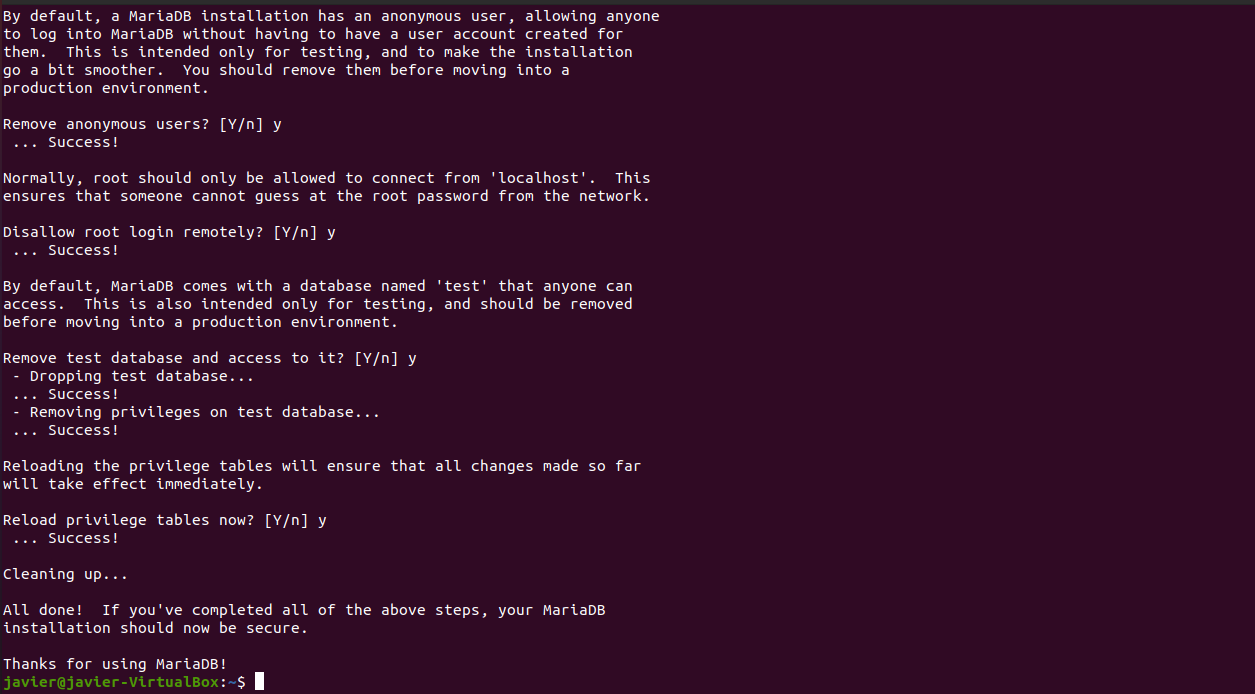
* Para instalar openSSH, introducimos: sudo apt-get install openssh-server
* Instalamos la aplicación de cliente: sudo apt-get install openssh-client
* Iniciamos el servicio: sudo systemctl start sshd.service
* Comprobamos el estado del servicio: sudo systemctl status sshd.service
* Y comprobamos que está activo.
* El puerto utilizado es el 22.

### 5.- Instalación MariaDB

* En primer lugar instalamos el paquete: sudo apt install mariadb-server
* Una vez instalado vamos a comprobar si esta activo: sudo systemctl status mariadb y comprobamos que efectivamente esta en ejecución.



* Ahora vamos a configurar MariaDB
  + Sudo mysql\_secure\_installation. Le decimos que no queremos establecer contraseñas y al resto de preguntas le decimos que si



* Ya estaría instalado y configurado mariadb

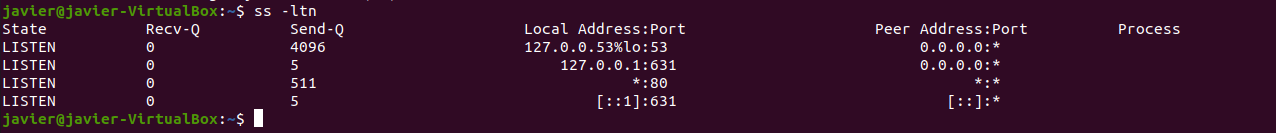
## Requerimiento 2

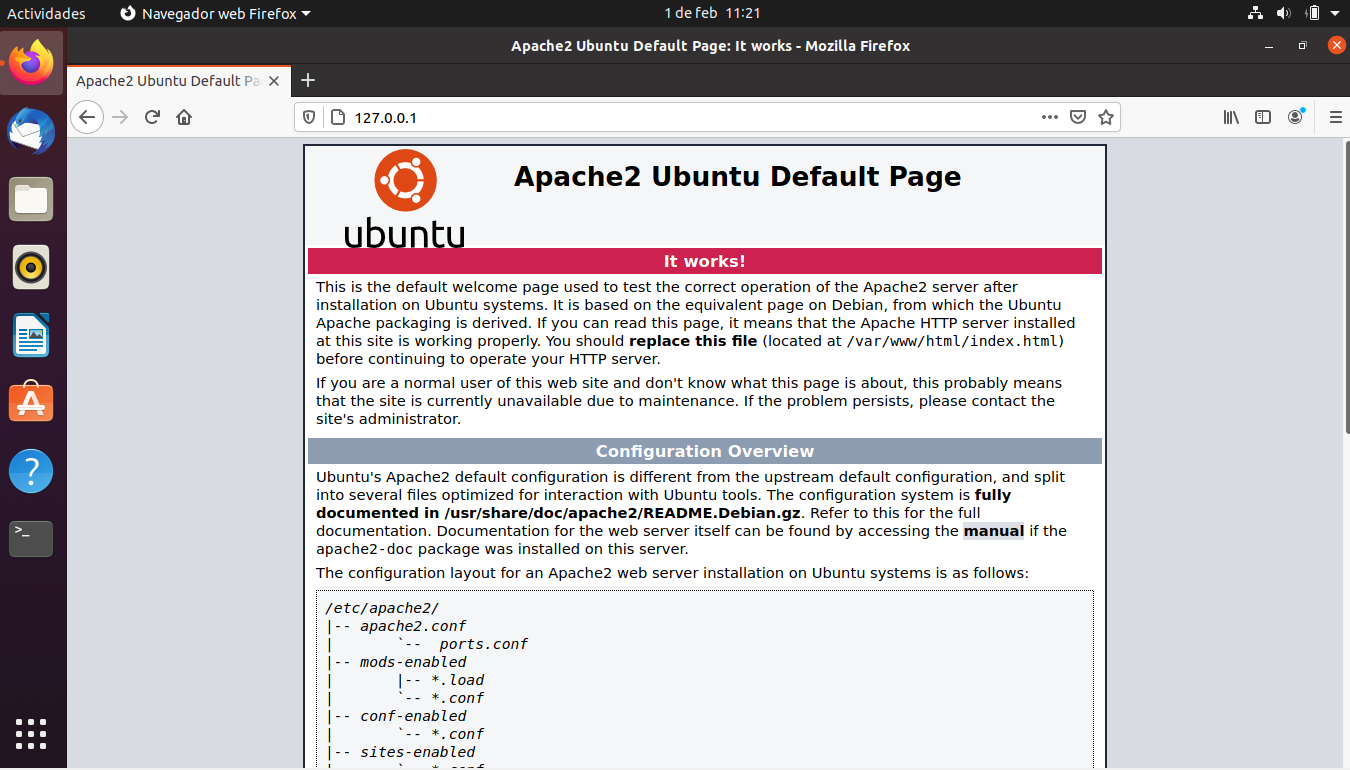
Configurar y comprobar que los puertos asociados a Apache, Tomcat y MariaDB están abiertos en el Firewall y son accesibles desde el exterior.

### 2.1.- Apache

El puerto que utiliza el Apache es el 80, comprobamos si está escuchando

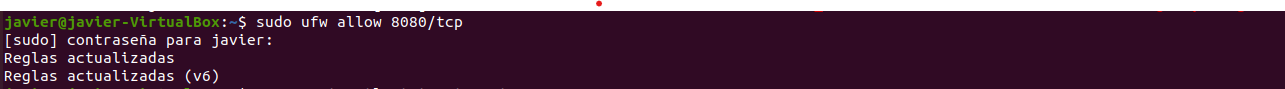
* Comprobamos que el puerto está escuchando con ss -ltn



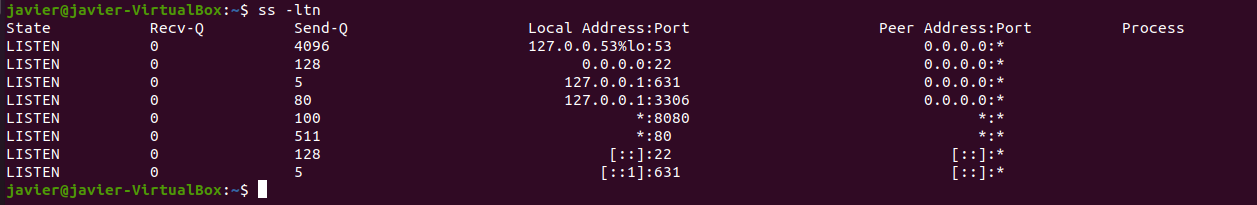
* Comprobamos que es accesible desde fuera, para ello introducimos en el navegador: <http://127.0.0.1> y vemos que Apache responde

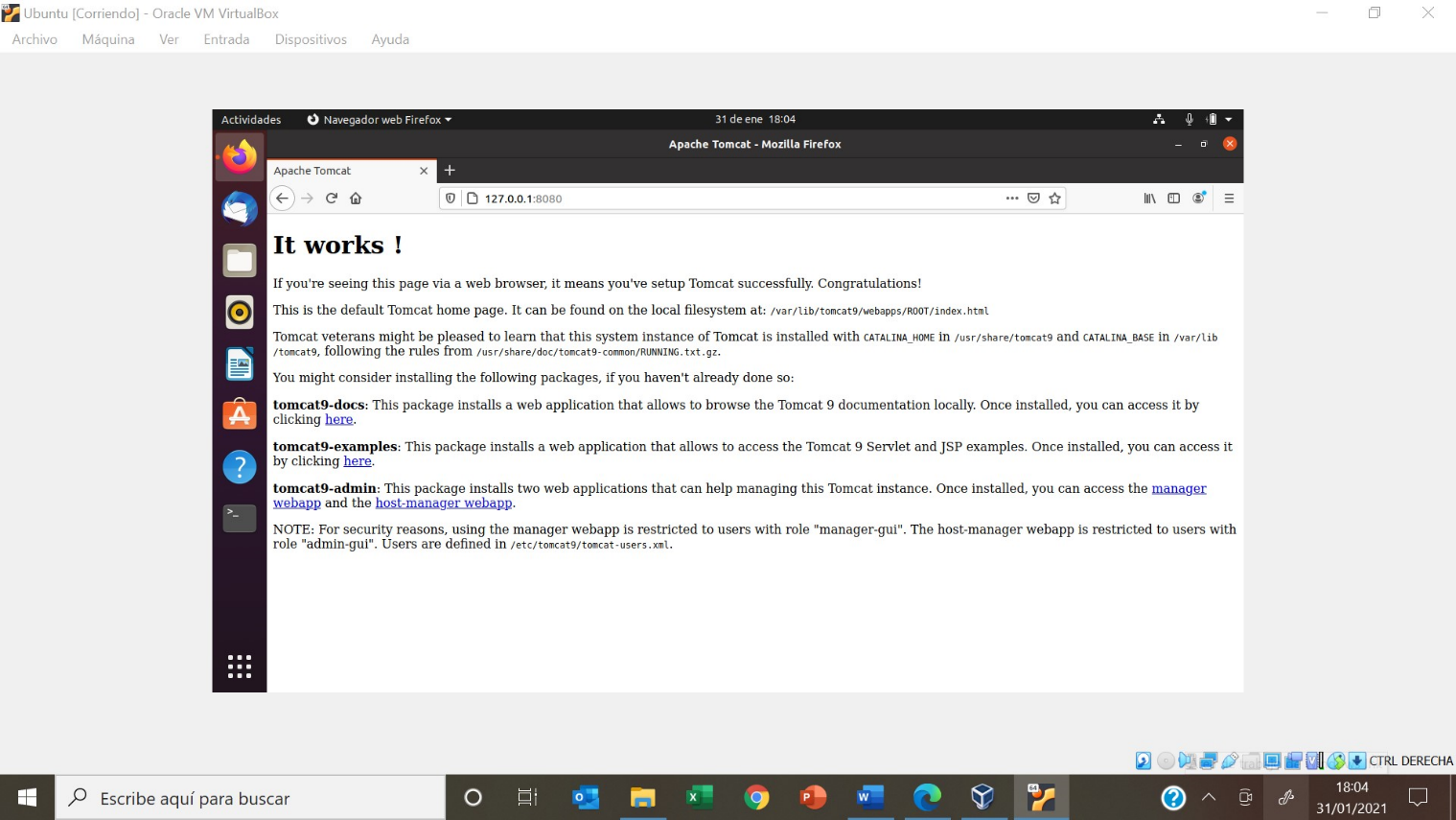
### 2.2.- Tomcat

* Para habilitar el puerto 8080 en el Firewall introduciremos lo siguiente:
  + Sudo ufw allow 8080/tcp



* Comprobamos que el puerto está escuchando



* Nos vamos al navegador e introducimos la dirección: <http://127.0.0.1:8080> para comprobar que nuestro Tomcat está escuchando y funciona

### 2.3.- MariaDb

* El puerto que utiliza mariadb es el 3306
* Comprobamos que el puerto está escuchando con ss -ltn y abierto en el firewall

