

# **Tarea UD3**

## **Despregramento de Aplicacións Web**

24/25

**Índice.**

**Sumario**

1. Funcionamiento de Tomcat en Windows con XAMPP.....	3
2. Instalación, configuración y funcionamiento de Tomcat en Linux.....	4
3. Securización.....	5
4. Integración con un IDE.....	6
5. Cuestiones.....	7

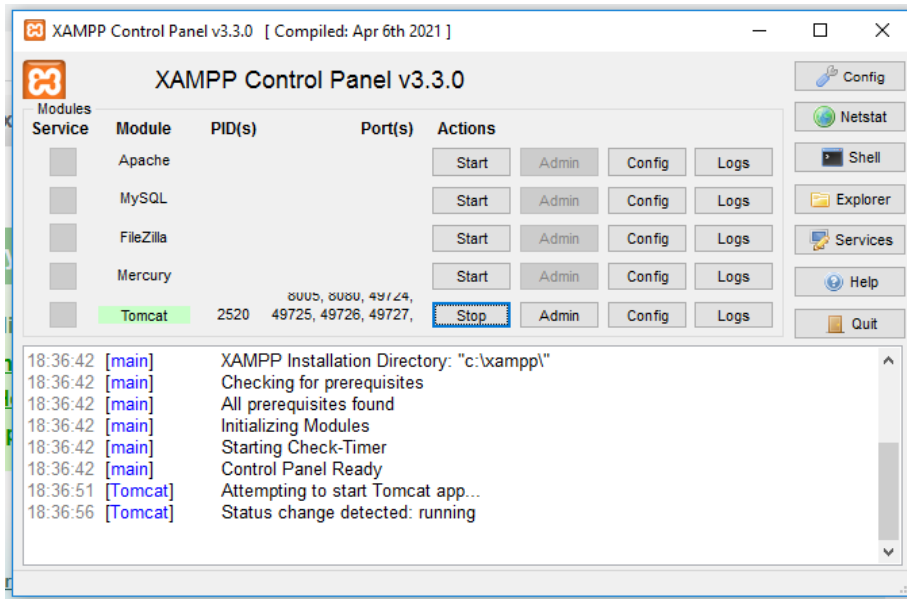
# 1. Funcionamiento de Tomcat en Windows con XAMPP

Se te facilita una máquina Windows, que ya viene con un Xampp operativo. Se te pide.

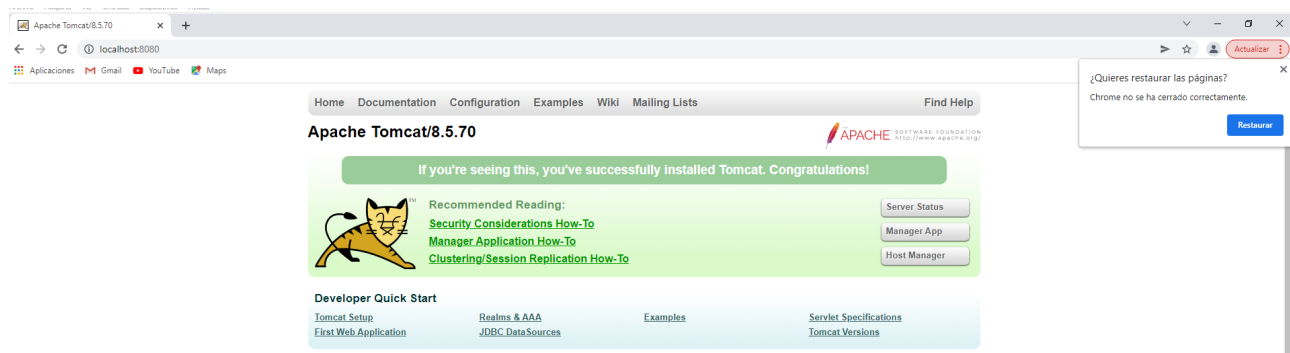
## a. Arranque y configuración inicial

Abrimos el panel de control de XAMP

Iniciamos el servicio de Tomcat



Accedemos en el navegador a localhost:8080 (el puerto por defecto para tomcat)



## SERVIDOR DE APLICACIONES – APACHE TOMCAT

### b. Acceso a opciones de gestión y administración

Configúralo para que pueda acceder a server status, Manager App y Host Manager

Pistas:

Vamos a la ruta C:\xampp\tomcat\conf y buscamos el archivo tomcat-users.xml y lo editamos:

Este equipo > Disco local (C:) > xampp > tomcat > conf					
	Nombre	Fecha de modifica...	Tipo	Tamaño	
	Catalina	28/10/2024 18:36	Carpeta de archivos		
	catalina.policy	09/08/2021 18:17	Archivo POLICY	14 KB	
	catalina.properties	09/08/2021 18:17	Archivo PROPERTI...	8 KB	
	context.xml	09/08/2021 18:17	Documento XML	2 KB	
	jaspic-providers.xml	09/08/2021 18:17	Documento XML	2 KB	
	jaspic-providers.xsd	09/08/2021 18:17	Archivo XSD	3 KB	
	logging.properties	09/08/2021 18:17	Archivo PROPERTI...	4 KB	
	server.xml	09/08/2021 18:17	Documento XML	8 KB	
	tomcat-users.xml	09/08/2021 18:17	Documento XML	3 KB	
	tomcat-users.xsd	09/08/2021 18:17	Archivo XSD	3 KB	
	web.xml	09/08/2021 18:17	Documento XML	173 KB	

editamos el archivo y descomentamos el usuario admin, le agregamos los 3 roles, de la siguiente manera:

-->

```
<user username="admin" password="abc123." roles="manager-gui,manager-script,manager-status"/>
<user username="hostmanager" password="abc123." roles="manager-jmx,admin-gui,admin-script"/>
```

Añadimos los siguientes roles:

admin:

- manager-gui
- manager-script
- manager-status

hostmanager:

- manager-jmx
- admin-gui
- admin-script

# SERVIDOR DE APLICACIONES – APACHE TOMCAT

Una vez modificado, reiniciamos el servicio de tomcat y vamos al dashboard nuevamente en localhost:8080 y entramos:


localhost:8080/manager/status

Aplicaciones

Gmail

YouTube

Maps



Estado de Servidor

Gestor

Listar Aplicaciones

Ayuda HTML de Gestor

Ayuda de Gestor

Información de Servidor

Versión de Tomcat	Versión JVM	Vendedor JVM	Nombre del SO	Versión de SO	Arquitectura de SO	NombreDeMáquina	Dirección
Apache Tomcat/8.5.70	1.8.0_261-b12	Oracle Corporation	Windows 10	10.0	amd64	base	10.0.2.15

JVM

Memoria disponible: 93.96 MB Total Memory: 123.00 MB Max Memory: 1820.50 MB

Pool de Memoria	Type	Initial	Total	Maximum	Used
PS Eden Space	Heap memory	32.50 MB	32.50 MB	672.50 MB	17.04 MB (2%)
PS Old Gen	Heap memory	85.50 MB	85.50 MB	1365.50 MB	6.99 MB (0%)
PS Survivor Space	Heap memory	4.00 MB	4.00 MB	4.00 MB	4.00 MB (100%)


localhost:8080/manager/html


Aplicaciones

Gmail

YouTube

Maps





Gestor de Aplicaciones Web de Tomcat

Mensaje:

OK

Gestor

Listar Aplicaciones

Ayuda HTML de Gestor

Ayuda de Gestor

Estado de Servidor

Aplicaciones

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado	Welcome to Tomcat	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar x 30 minutos
/docs	Ninguno especificado	Tomcat Documentation	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar x 30 minutos


localhost:8080/host-manager/html


Aplicaciones

Gmail

YouTube

Maps





Gestor de Máquina Virtual de Tomcat

Mensaje:

OK

Gestor de Máquina

Lista de Máquinas Virtuales

Ayuda de Gestor de Máquina.HTML.(¡En breve!)

Ayuda de Gestor de Máquina

Estado de Servidor

Nombre de Máquina

Nombre de Máquina	Alias de Máquina	Comandos
localhost		Instalado Gestor de Máquinas - comandos desactivados

Añadir Máquina Virtual

Máquina

Nombre:

Alias:

App base:

AutoDeploy

☒

DeployOnStartup

☒

DeployXML

☒

UnpackWARs

☒

App de Gestor

☒

CopyXML

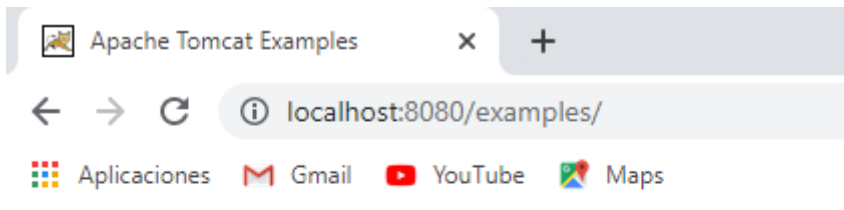
☐

Añadir

[Guide to Tomcat Manager Application | Baeldung](#)

### c. Ejemplos

En el dashboard de tomcat, vamos a la pestaña “ejemplos”:



### Apache Tomcat Examples

- [Servlets examples](#)
- [JSP Examples](#)
- [WebSocket Examples](#)

Seleccionamos el que mas rabia nos dé:

# SERVIDOR DE APLICACIONES – APACHE TOMCAT

JSP Examples

localhost:8080/examples/jsp/


Aplicaciones Gmail YouTube Maps


## JSP Samples


This is a collection of samples demonstrating the usage of different parts of the Java Server Pages (JSP) specification. Both JSP 2.0 and JSP 1.2 examples are presented below.

These examples will only work when these pages are being served by a servlet engine; of course, we recommend [Tomcat](#). They will not work if you are viewing these pages via a "file:///..." URL.

To navigate your way through the examples, the following icons will help:











 Execute the example

 Look at the source code for the example

 Return to this screen

Tip: For session scoped beans to work, the cookies must be enabled. This can be done using browser options.

### JSP 2.0 Examples

Expression Language		
Basic Arithmetic	 <a href="#">Execute</a>	 <a href="#">Source</a>
Basic Comparisons	 <a href="#">Execute</a>	 <a href="#">Source</a>
Implicit Objects	 <a href="#">Execute</a>	 <a href="#">Source</a>
Functions	 <a href="#">Execute</a>	 <a href="#">Source</a>
Composite Expressions	 <a href="#">Execute</a>	 <a href="#">Source</a>

Seleccionamos uno de los ejemplos:

JSP 2.0 Expression Language - Basic Arithmetic

localhost:8080/examples/jsp/jsp2/el/basic-arithmetic.jsp

Aplicaciones Gmail YouTube Maps

## JSP 2.0 Expression Language - Basic Arithmetic

This example illustrates basic Expression Language arithmetic. Addition (+), subtraction (-), multiplication (\*), division (/ or div), and modulus (% or mod) are all supported. Error conditions, like division by zero, are handled gracefully.

EL Expression	Result
<code>\$ {1}</code>	1
<code>\$ {1 + 2}</code>	3
<code>\$ {1.2 + 2.3}</code>	3.5
<code>\$ {1.2E4 + 1.4}</code>	12001.4
<code>\$ {-4 - 2}</code>	-6
<code>\$ {21 * 2}</code>	42
<code>\$ {3/4}</code>	0.75
<code>\$ {3 div 4}</code>	0.75
<code>\$ {3/0}</code>	Infinity
<code>\$ {10%4}</code>	2
<code>\$ {10 mod 4}</code>	2
<code>\$ {(1+2) ? 3 : 4}</code>	4

## d. Despliega tu propia aplicación web

Despliega un fichero Sample.war, y comprueba que puedes acceder a a la aplicación

Pistas:

[Apache Tomcat 11 \(11.0.0-M16\) - Tomcat Web Application Deployment](#)

[How to Deploy a WAR File to Tomcat | Baeldung](#)

[Tomcat Manager y su configuración - Arquitectura Java](#)

Vamos al gestor de aplicaciones web de tomcat:

Nombre y apellidos:David Rodriguez Leiras

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

En el apartado “desplegar” bucamos nuestro archivo .war y pulsamos sobre “desplegar”:

manager	Ninguno especificado	Tomcat Manager Application	true	1	Expirar sesiones sin trabajar ≥ 30 minutos
					Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
<b>Desplegar</b>					
Desplegar directorio o archivo WAR localizado en servidor					
Trayectoria de Contexto (opcional): <input type="text"/>					
URL de archivo de Configuración XML: <input type="text"/>					
URL de WAR o Directorio: <input type="text"/>					
<input type="button" value="Desplegar"/>					
<b>Archivo WAR a desplegar</b>					
Seleccione archivo WAR a cargar <input type="button" value="Seleccionar archivo"/> sample war					
<input type="button" value="Desplegar"/>					
<b>Configuration</b>					
Re-read TLS configuration files					

Ahora podemos ver la aplicación desplegada en la lista:

Aplicaciones					
Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
/docs	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
/sample	Ninguno especificado	Hello, World Application	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
<b>Desplegar</b>					
Desplegar directorio o archivo WAR localizado en servidor					
Trayectoria de Contexto (opcional): <input type="text"/>					
URL de archivo de Configuración XML: <input type="text"/>					
URL de WAR o Directorio: <input type="text"/>					
<input type="button" value="Desplegar"/>					
<b>Archivo WAR a desplegar</b>					
Seleccione archivo WAR a cargar <input type="button" value="Seleccionar archivo"/> Ningún archivo seleccionado					
<input type="button" value="Desplegar"/>					
<b>Configuration</b>					
Re-read TLS configuration files					
TLS host name (optional) <input type="text"/>					
<input type="button" value="Re-read"/>					



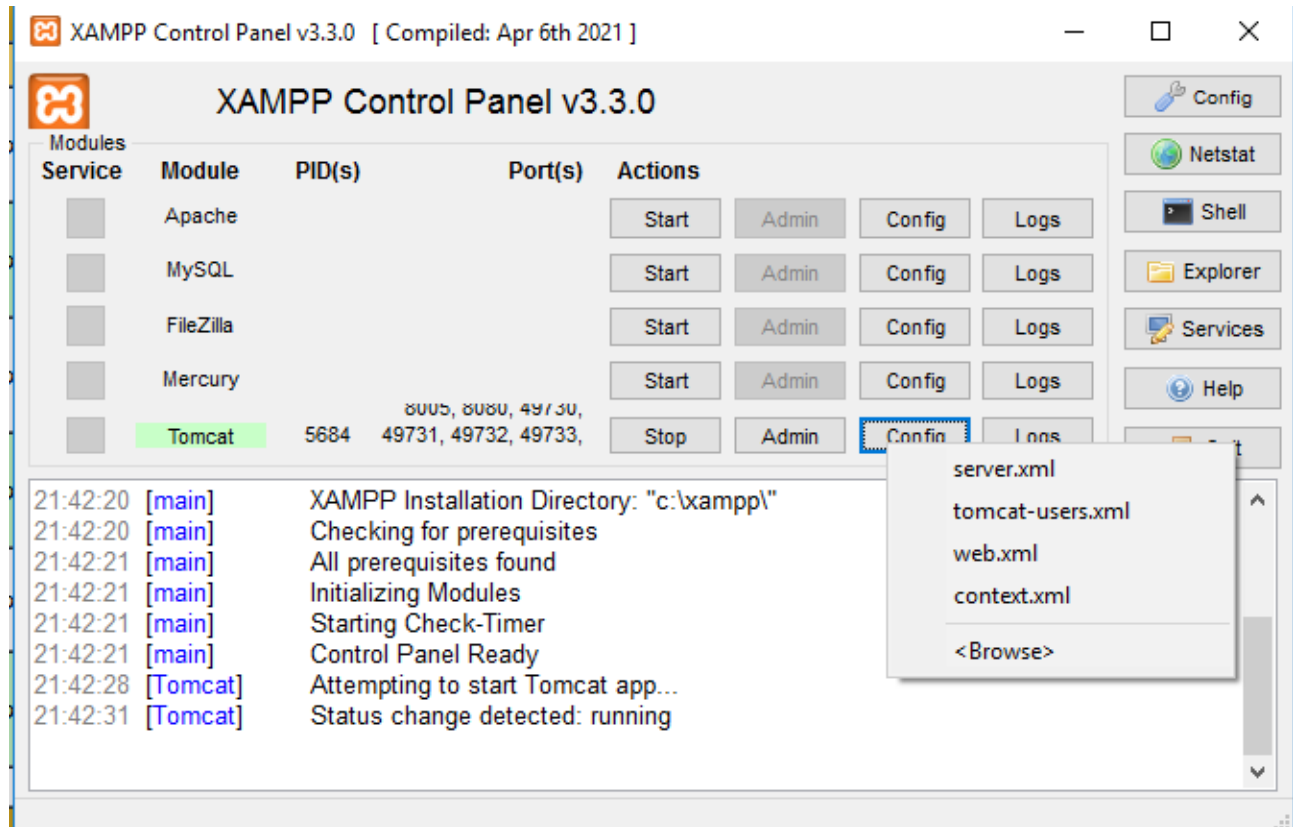
## SERVIDOR DE APLICACIONES – APACHE TOMCAT

### e. Cambia el puerto donde escucha Tomcat

Pista:

[Change the Default Port of the Tomcat Server \(microfocus.com\)](https://microfocus.com)

Vamos al panel de control de XAMP y sobre el servicio de tomcat pinchamos en el boton de “Config” y posteriormente en “server.xml”:



Se abra el archivo: y debemos modificar la siguiente entrada:

```
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443" />
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Modificamos el campo “redirectPort” y le indicamos un puerto que no este ocupado por ningun servicio:

```
<Connector port="8080" protocol="HTTP/1.1"  
          connectionTimeout="20000"  
          redirectPort="8475" />
```

Ahora guardamos y reiniciamos el servidor tomcat.

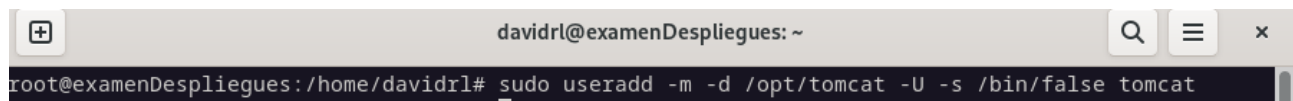
## 2. Instalación, configuración y funcionamiento de Tomcat en Linux

a) Instala y realiza configuración para que pueda acceder a server status, Manager App y Host Manager

Pista:

[How To Install Apache Tomcat 10 on Ubuntu 20.04 | DigitalOcean](#)

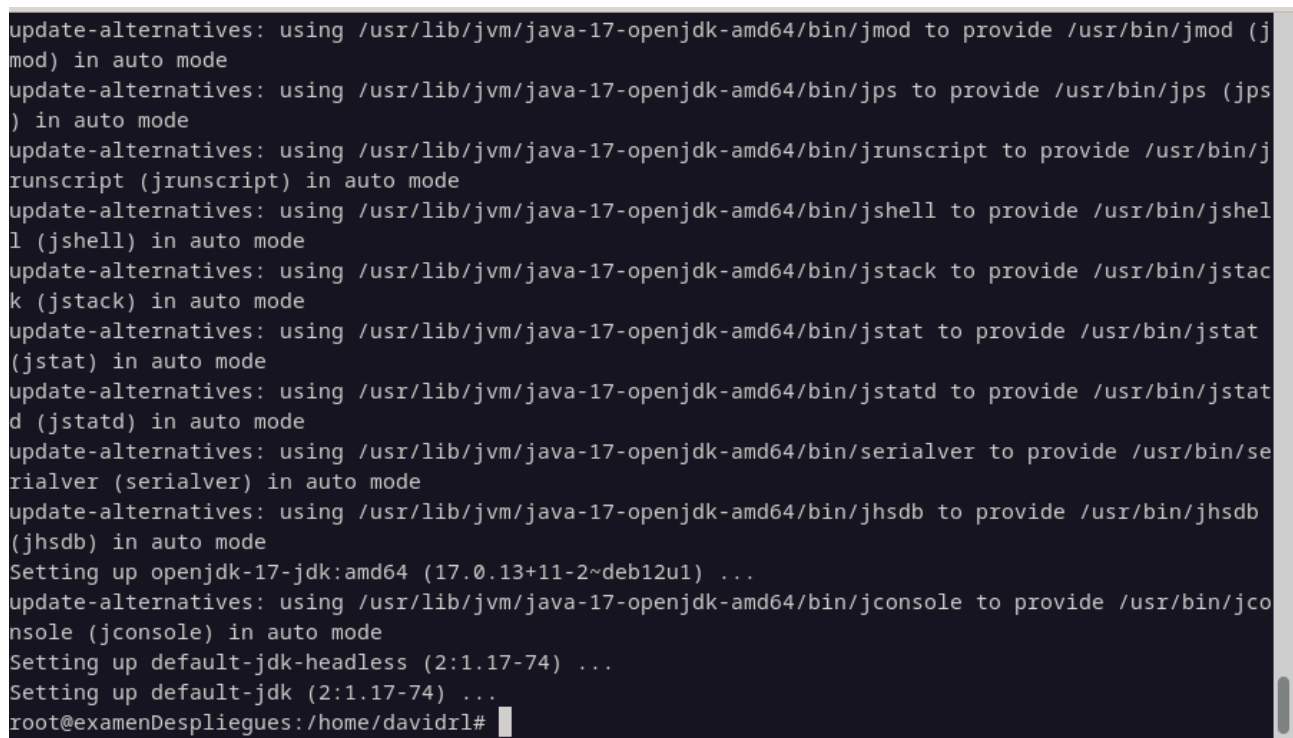
Creamos un usuario para tomcat:



```
davidrl@examenDespliegues: ~  
root@examenDespliegues:/home/davidrl# sudo useradd -m -d /opt/tomcat -U -s /bin/false tomcat
```

Actualizamos repositorio:

Instalamos el paquete jdk(Java development kit) de java:



```
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jmod to provide /usr/bin/jmod (jmod) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jps to provide /usr/bin/jps (jps) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jrunscript to provide /usr/bin/jrunscript (jrunscript) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jshell to provide /usr/bin/jshell (jshell) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jstack to provide /usr/bin/jstack (jstack) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jstat to provide /usr/bin/jstat (jstat) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jstatd to provide /usr/bin/jstatd (jstatd) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jhsdb to provide /usr/bin/jhsdb (jhsdb) in auto mode  
Setting up openjdk-17-jdk:amd64 (17.0.13+11-2~deb12u1) ...  
update-alternatives: using /usr/lib/jvm/java-17-openjdk-amd64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode  
Setting up default-jdk-headless (2:1.17-74) ...  
Setting up default-jdk (2:1.17-74) ...  
root@examenDespliegues:/home/davidrl#
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Comprobamos la version con :

java -version

```
root@examenDespliegues:/home/davidrl# java -version
openjdk version "17.0.13" 2024-10-15
OpenJDK Runtime Environment (build 17.0.13+11-Debian-2deb12u1)
OpenJDK 64-Bit Server VM (build 17.0.13+11-Debian-2deb12u1, mixed mode, sharing)
root@examenDespliegues:/home/davidrl#
```

Ahora navegamos a /tmp con cd /tmp

Nos bajamos la ultima version de tomcat:

wget <https://dlcdn.apache.org/tomcat/tomcat-11/v11.0.1/bin/apache-tomcat-11.0.1.tar.gz>

Podemos comprobar la ultima version en este [link](#):

```
root@examenDespliegues:/tmp# wget https://dlcdn.apache.org/tomcat/tomcat-11/v11.0.1/bin/apache-tomcat-11.0.1.tar.gz
--2024-11-11 18:35:04-- https://dlcdn.apache.org/tomcat/tomcat-11/v11.0.1/bin/apache-tomcat-11.0.1.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13643429 (13M) [application/x-gzip]
Saving to: 'apache-tomcat-11.0.1.tar.gz'

apache-tomcat-11.0.1.tar 100%[=====>] 13.01M 9.96MB/s in 1.3s

2024-11-11 18:35:05 (9.96 MB/s) - 'apache-tomcat-11.0.1.tar.gz' saved [13643429/13643429]

root@examenDespliegues:/tmp#
```

Extraemos el archivo usando el comando:

sudo tar xzvf apache-tomcat-11\*tar.gz -C /opt/tomcat --strip-components=1

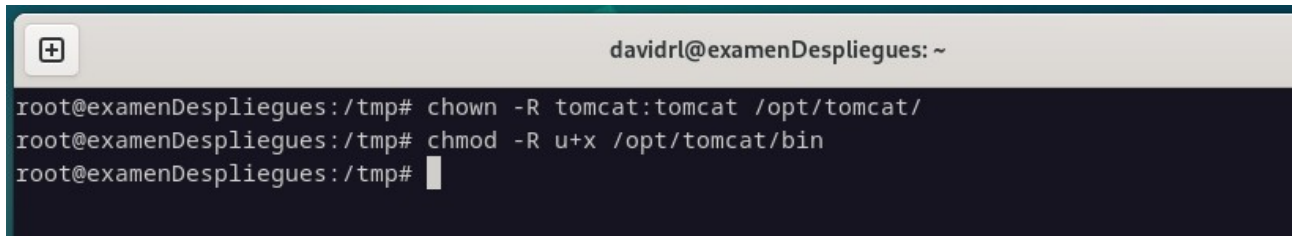
```
root@examenDespliegues:/tmp# sudo tar xzvf apache-tomcat-11*tar.gz -C /opt/tomcat --strip-components=1
apache-tomcat-11.0.1/conf/
apache-tomcat-11.0.1/conf/catalina.properties
apache-tomcat-11.0.1/conf/context.xml
apache-tomcat-11.0.1/conf/jaspic-providers.xml
apache-tomcat-11.0.1/conf/jaspic-providers.xsd
apache-tomcat-11.0.1/conf/logging.properties
apache-tomcat-11.0.1/conf/server.xml
apache-tomcat-11.0.1/conf/tomcat-users.xml
apache-tomcat-11.0.1/conf/tomcat-users.xsd
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Ya que hemos creado un usuario para tomcat, le vamos a dar permisos sobre los archivos que hemos extraído:

```
sudo chown -R tomcat:tomcat /opt/tomcat/
```

```
sudo chmod -R u+x /opt/tomcat/bin
```



```
davidrl@examenDespliegues: ~  
root@examenDespliegues:/tmp# chown -R tomcat:tomcat /opt/tomcat/  
root@examenDespliegues:/tmp# chmod -R u+x /opt/tomcat/bin  
root@examenDespliegues:/tmp#
```

Ahora vamos a configurar los usuarios administradores, voy a configurar los usuarios para acceder a Manager y HostManager en nuestro tomcat, para ello debemos modificar los archivos de configuración correspondiente que se encuentra en:

/opt/tomcat/conf/tomcat-users.xml

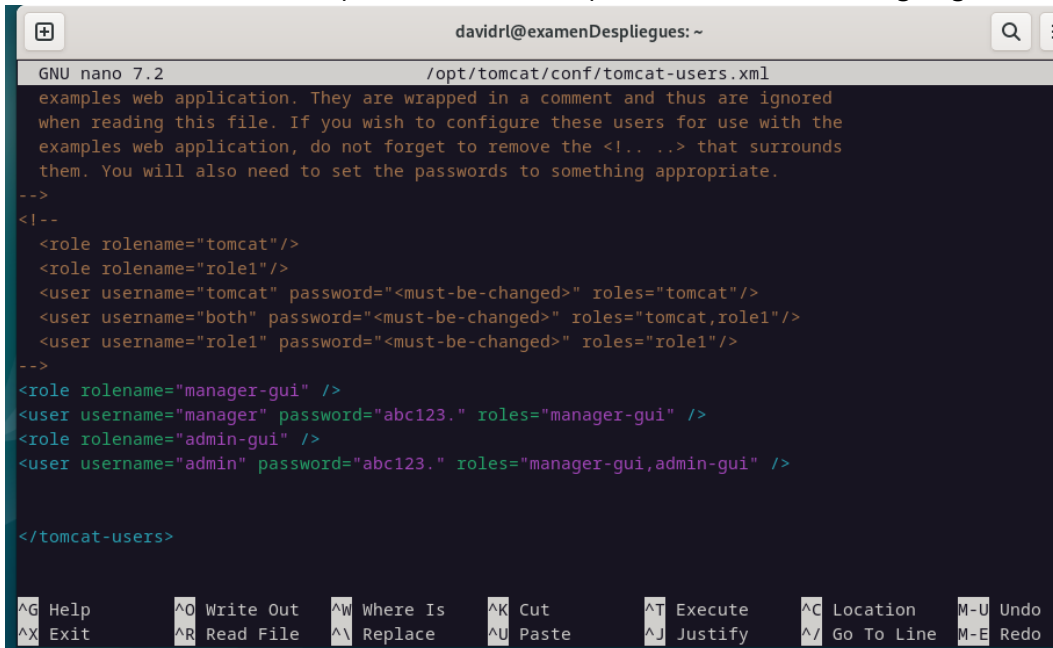
Añadimos las siguientes entradas:

```
<role rolename="manager-gui" />
```

```
<user username="manager" password="manager_password" roles="manager-gui" />
```

```
<role rolename="admin-gui" />
```

```
<user username="admin" password="admin_password" roles="manager-gui,admin-gui" />
```



```
davidrl@examenDespliegues: ~  
GNU nano 7.2 /opt/tomcat/conf/tomcat-users.xml  
examples web application. They are wrapped in a comment and thus are ignored  
when reading this file. If you wish to configure these users for use with the  
examples web application, do not forget to remove the <!-- ... --> that surrounds  
them. You will also need to set the passwords to something appropriate.  
-->  
<!--  
<role rolename="tomcat"/>  
<role rolename="role1"/>  
<user username="tomcat" password="<must-be-changed>" roles="tomcat"/>  
<user username="both" password="<must-be-changed>" roles="tomcat,role1"/>  
<user username="role1" password="<must-be-changed>" roles="role1"/>  
-->  
<role rolename="manager-gui" />  
<user username="manager" password="abc123." roles="manager-gui" />  
<role rolename="admin-gui" />  
<user username="admin" password="abc123." roles="manager-gui,admin-gui" />  
  
</tomcat-users>
```

Es importante cambiar las contraseñas por defecto

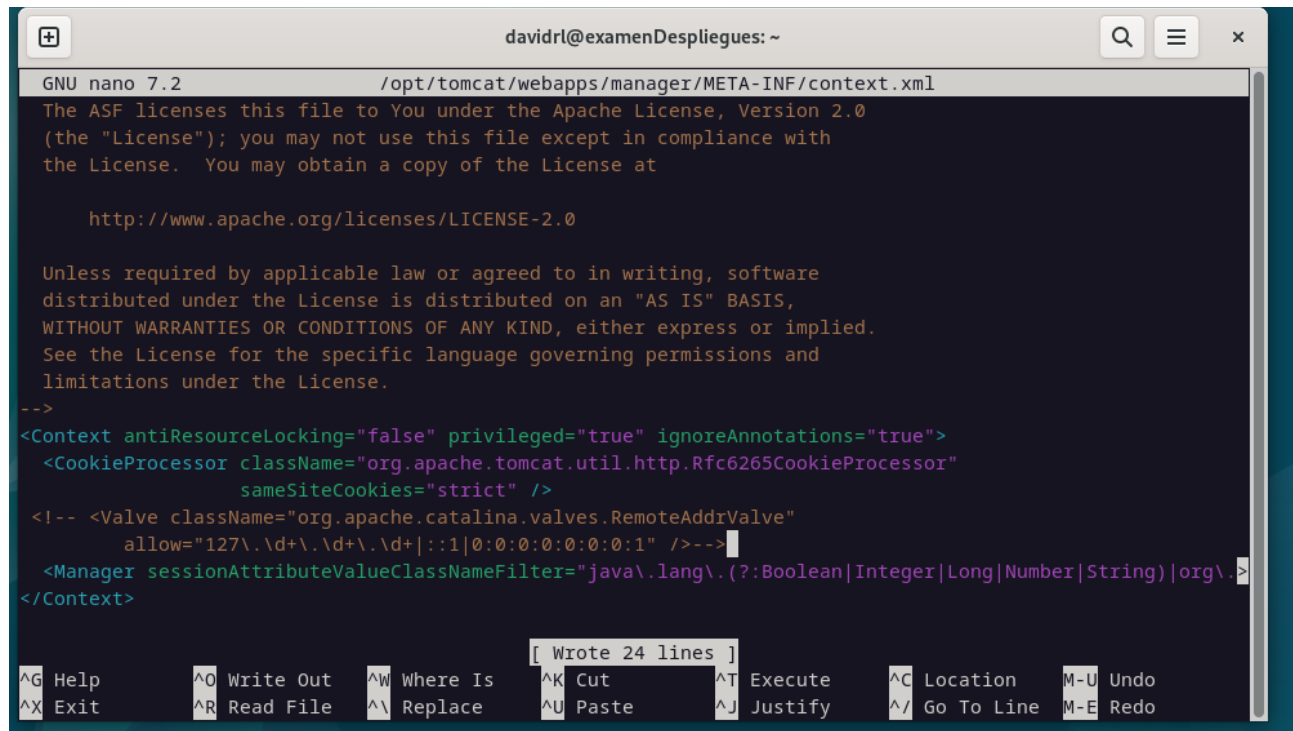
## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Por defecto tomcat restringe el acceso a las paginas de administracion, por lo tanto , si queremos acceder debemos eliminar dicha restrinccion en dichas páginas, para ello vamos a editarlas:

Modificamos primero este archivo:

```
sudo nano /opt/tomcat/webapps/manager/META-INF/context.xml
```

**Comentamos la etiqueta <valve>**



```
GNU nano 7.2 /opt/tomcat/webapps/manager/META-INF/context.xml
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License.  You may obtain a copy of the License at

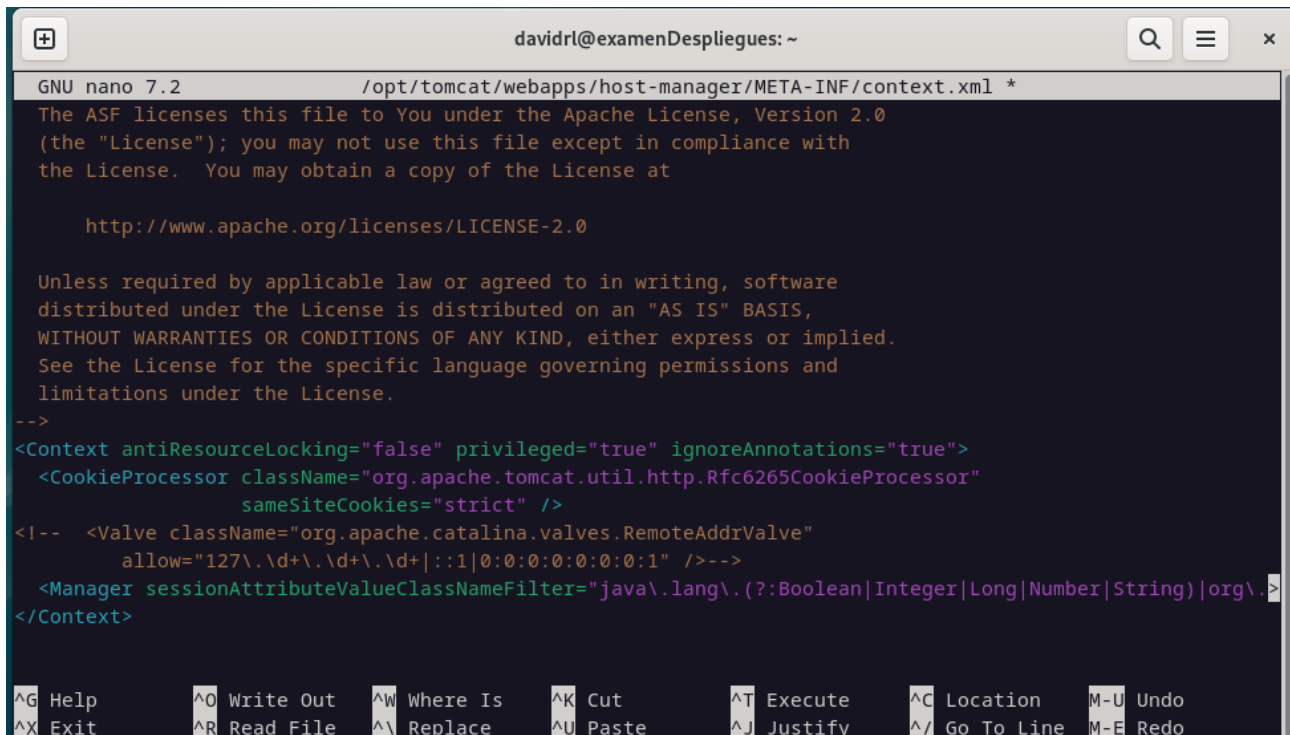
    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" ignoreAnnotations="true">
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
  <!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />-->
  <Manager sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|string)|org\.>
</Context>
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Guardamos y realizamos la misma operación en este otro archivo:

`sudo nano /opt/tomcat/webapps/host-manager/META-INF/context.xml`



```
GNU nano 7.2 /opt/tomcat/webapps/host-manager/META-INF/context.xml *
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" ignoreAnnotations="true">
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
  <!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.0\.0\.1|::1|0:0:0:0:0:0:0:1" />-->
  <Manager sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|string)|org\.>
</Context>
```

Con esto hemos definido dos usuarios con los roles de Manager y HostManager y hemos permitido el acceso a dichas páginas.

Ahora vamos a configurar el servicio a través de systemd, para ello, antes de nada, al ser Tomcat un programa hecho en Java, debemos iniciar el servicio de Java, para ello primero localizaremos donde se encuentra nuestra instalación de Java con el siguiente comando:

`sudo update-java-alternatives -l`



```
root@examenDespliegues:/tmp# sudo update-java-alternatives -l
java-1.17.0-openjdk-amd64 1711 /usr/lib/jvm/java-1.17.0-openjdk-amd64
```

Usaremos la ruta resultante de este comando en nuestro servicio de Tomcat, ahora, para crear el servicio de tomcat crearemos un nuevo archivo debajo de la siguiente ruta:

`/etc/systemd/system/`

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Creamos un archivo llamado tomcat.service y añadimos a este archivo las siguientes entradas:

[Unit]

Description=Tomcat

After=network.target

[Service]

Type=forking

User=tomcat

Group=tomcat

Environment="JAVA\_HOME=/usr/lib/jvm/java-1.17.0-openjdk-amd64"

Environment="JAVA\_OPTS=-Djava.security.egd=file:///dev/urandom"

Environment="CATALINA\_BASE=/opt/tomcat"

Environment="CATALINA\_HOME=/opt/tomcat"

Environment="CATALINA\_PID=/opt/tomcat/temp/tomcat.pid"

Environment="CATALINA\_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"

ExecStart=/opt/tomcat/bin/startup.sh

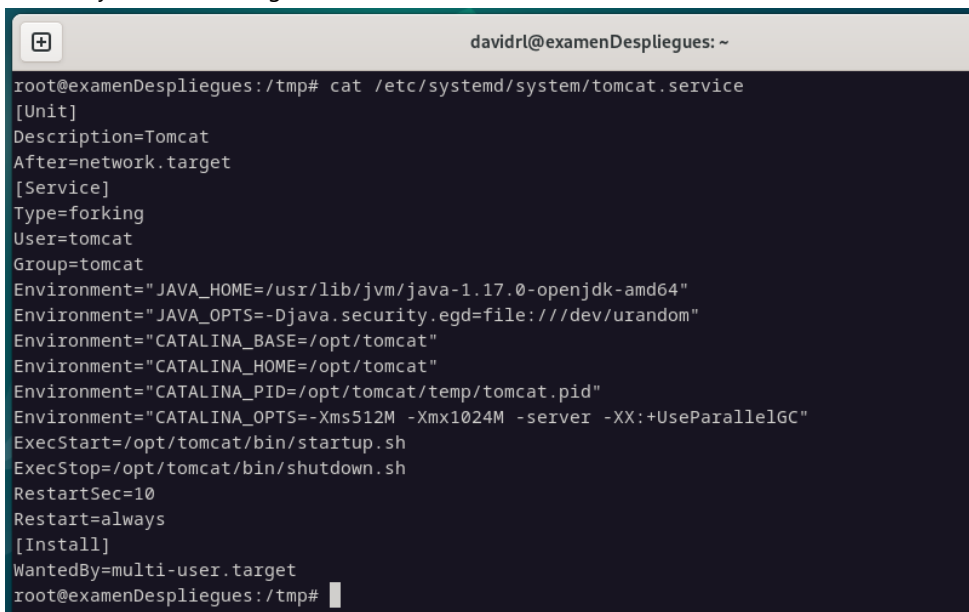
ExecStop=/opt/tomcat/bin/shutdown.sh

RestartSec=10

Restart=always

[Install]

WantedBy=multi-user.target

A terminal window with a dark background and light text. The title bar shows 'davidrl@examenDespliegues: ~'. The prompt is 'root@examenDespliegues:/tmp#'. The user has entered 'cat /etc/systemd/system/tomcat.service'. The output shows the contents of the service file, which matches the text provided in the previous blocks. The terminal ends with the prompt 'root@examenDespliegues:/tmp#' and a cursor.

```
davidrl@examenDespliegues: ~
root@examenDespliegues:/tmp# cat /etc/systemd/system/tomcat.service
[Unit]
Description=Tomcat
After=network.target
[Service]
Type=forking
User=tomcat
Group=tomcat
Environment="JAVA_HOME=/usr/lib/jvm/java-1.17.0-openjdk-amd64"
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom"
Environment="CATALINA_BASE=/opt/tomcat"
Environment="CATALINA_HOME=/opt/tomcat"
Environment="CATALINA_PID=/opt/tomcat/temp/tomcat.pid"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
RestartSec=10
Restart=always
[Install]
WantedBy=multi-user.target
root@examenDespliegues:/tmp#
```



## SERVIDOR DE APLICACIONES – APACHE TOMCAT

En la entrada en negrita copiamos el path que hemos obtenido antes.

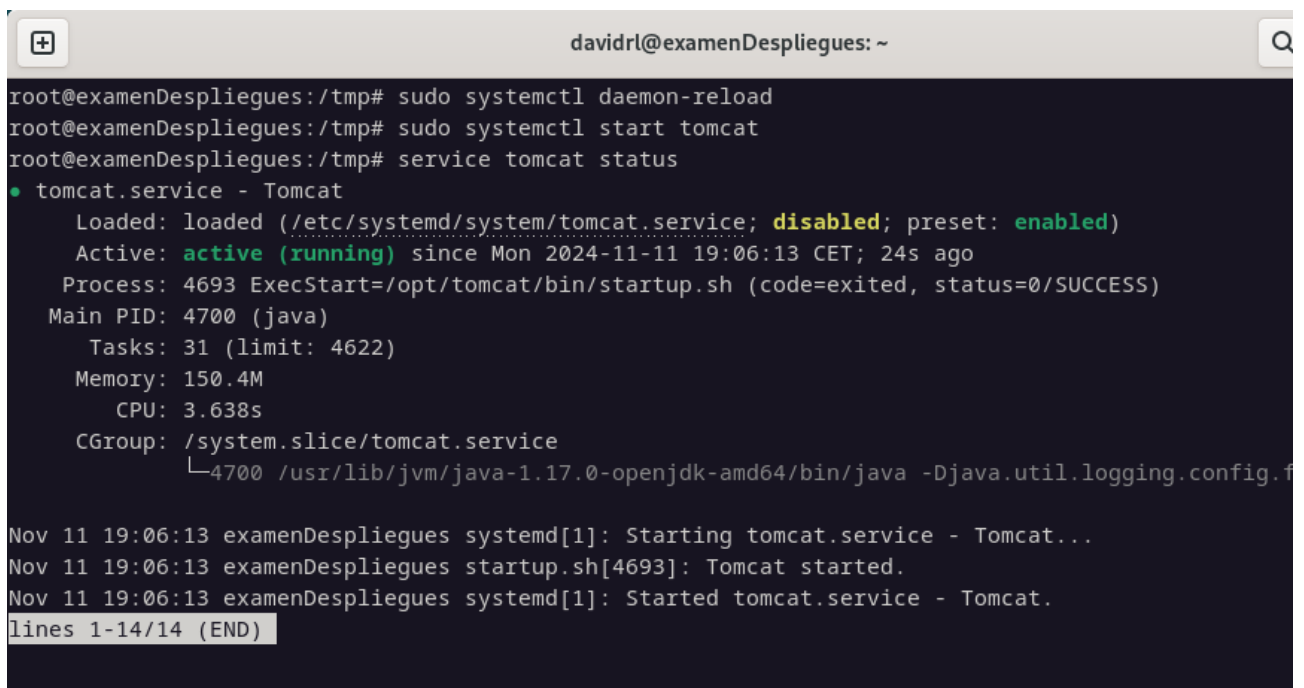
Reiniciamos el daemon de systemd para que pille el nuevo servicio creado:

```
sudo systemctl daemon-reload
```

Ahora iniciamos el servicio que hemos creado con el comando:

```
sudo systemctl start tomcat
```

Y comprobamos su status:

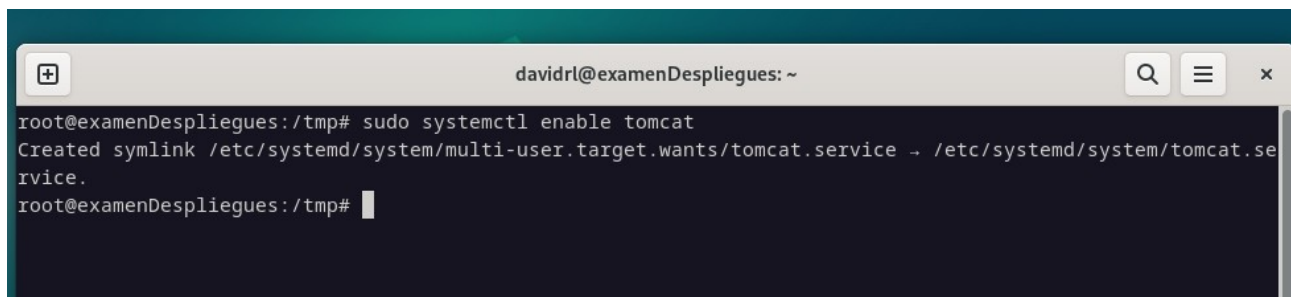


```
davidrl@examenDespliegues: ~
root@examenDespliegues:/tmp# sudo systemctl daemon-reload
root@examenDespliegues:/tmp# sudo systemctl start tomcat
root@examenDespliegues:/tmp# service tomcat status
• tomcat.service - Tomcat
   Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; preset: enabled)
   Active: active (running) since Mon 2024-11-11 19:06:13 CET; 24s ago
     Process: 4693 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
    Main PID: 4700 (java)
       Tasks: 31 (limit: 4622)
      Memory: 150.4M
         CPU: 3.638s
    CGroup: /system.slice/tomcat.service
            └─4700 /usr/lib/jvm/java-1.17.0-openjdk-amd64/bin/java -Djava.util.logging.config.f

Nov 11 19:06:13 examenDespliegues systemd[1]: Starting tomcat.service - Tomcat...
Nov 11 19:06:13 examenDespliegues startup.sh[4693]: Tomcat started.
Nov 11 19:06:13 examenDespliegues systemd[1]: Started tomcat.service - Tomcat.
lines 1-14/14 (END)
```

Ahora usamos el siguiente comando para que este servicio se inicia al arranque de la máquina.

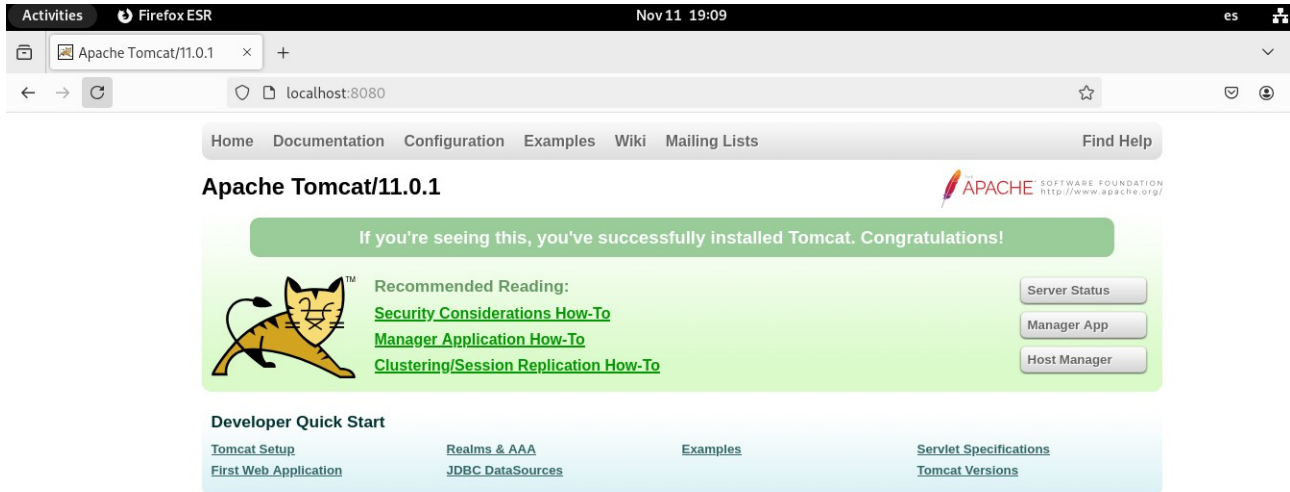
```
sudo systemctl enable tomcat
```



```
davidrl@examenDespliegues: ~
root@examenDespliegues:/tmp# sudo systemctl enable tomcat
Created symlink /etc/systemd/system/multi-user.target.wants/tomcat.service → /etc/systemd/system/tomcat.service.
root@examenDespliegues:/tmp#
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Ahora accedemos a tomcat usando el navegador, tomcat por defecto se despliega en el puerto 8080 así que iremos a localhost:8080



### b. Despliega tu propia aplicación web

Despliega un fichero Sample.war, y comprueba que puedes acceder a la aplicación

Pistas:

[Apache Tomcat 11 \(11.0.0-M16\) - Tomcat Web Application Deployment](#)

[How to Deploy a WAR File to Tomcat | Baeldung](#)

Vamos a la carpeta /opt/tomcat/webapps y copiamos el archivo .war a dicha carpeta:

```
root@examenDespliegues:/opt/tomcat/webapps# cp /home/davidrl/Downloads/sample.war sample.war
root@examenDespliegues:/opt/tomcat/webapps# ls
docs  examples  host-manager  manager  ROOT  sample.war
root@examenDespliegues:/opt/tomcat/webapps#
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Comprobamos que la aplicación aparece ahora en la pagina de Manager:

Activities Firefox ESR Nov 11 19:22

/manager T\_INF\_DAW\_2\_DESP: Ser x +

localhost:8080/manager/html

### Tomcat Web Application Manager

Message: OK

#### Manager

<a href="#">List Applications</a>	<a href="#">HTML Manager Help</a>	<a href="#">Manager Help</a>
-----------------------------------	-----------------------------------	------------------------------

#### Applications

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/sample	None specified	Hello, World Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

#### Deploy

Deploy directory or WAR file located on server

Context Path:

### 3. Securización

Configura para que el acceso sea seguro mediante certificados SSL, accediendo por HTTPS

Pistas:

[A Step-By-Step Guide to Apache Tomcat with SSL Configuration | by Kayathiri Mahendrakumaran | Analytics Vidhya | Medium](#)

[Instalación del certificado SSL en el servidor Apache Tomcat \(sslmarket.es\)](#)

[Apache Tomcat 9 \(9.0.85\) - SSL/TLS Configuration How-To](#)

Para securizar la conexión y usar HTTPS primero debemos crear , a traves de nuestro entorno JDK un “Keystore”, un Keystore es un archivo protegido que guardara las llaves necesarias para realizar una transaccion SSL, para su creacion JDK incluye en su paquete un programa llamado keytool.

Primero crearemos el Keystore, que contendra una llave privada y un certificado autofirmado usando el siguiente comando:

Windows:

```
"%JAVA_HOME%\bin\keytool" -genkey -alias tomcat -keyalg RSA
```

UNIX:

```
$JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA
```

Yo lo haré en un Debian12, primero sacamos el path de \$JAVA\_HOME:

En mi caso esta aquí:

```
/usr/lib/jvm/java-1.17.0-openjdk-amd64
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Vamos a ese directorio y ejecutamos el siguiente comando:

`keytool -genkey -alias tomcat -keyalg RSA nombreakivo.jks`

```
root@examenDespliegues:/usr/lib/jvm/java-1.17.0-openjdk-amd64# keytool -genkey -alias tomcat -keyalg RSA -keystore davidrl.jks

Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]:  davidrl
What is the name of your organizational unit?
  [Unknown]:  davidrl
What is the name of your organization?
  [Unknown]:  davidrl
What is the name of your City or Locality?
  [Unknown]:  ferrol
What is the name of your State or Province?
  [Unknown]:  coruna
What is the two-letter country code for this unit?
  [Unknown]:  es
Is CN=davidrl, OU=davidrl, O=davidrl, L=ferrol, ST=coruna, C=es correct?
  [no]:  yes

Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 90 days
```

El keystore archivo se creará en la misma carpeta donde tiramos el comando

```
root@examenDespliegues:/usr/lib/jvm/java-1.17.0-openjdk-amd64# ls -la
total 44
drwxr-xr-x  9 root root 4096 Nov 11 20:09 .
drwxr-xr-x  4 root root 4096 Nov 11 18:29 ..
drwxr-xr-x  2 root root 4096 Nov 11 18:29 bin
drwxr-xr-x  4 root root 4096 Nov 11 18:29 conf
-rw-r--r--  1 root root 2728 Nov 11 20:09 davidrl.jks
lrwxrwxrwx  1 root root   42 Oct 17 22:50 docs -> ../../../../share/doc/openjdk-17-jre-headless
drwxr-xr-x  3 root root 4096 Nov 11 18:29 include
drwxr-xr-x  2 root root 4096 Nov 11 18:29 jmods
drwxr-xr-x 72 root root 4096 Nov 11 18:29 legal
drwxr-xr-x  5 root root 4096 Nov 11 18:29 lib
drwxr-xr-x  3 root root 4096 Nov 11 18:29 man
-rw-r--r--  1 root root 1230 Oct 17 22:50 release
```

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

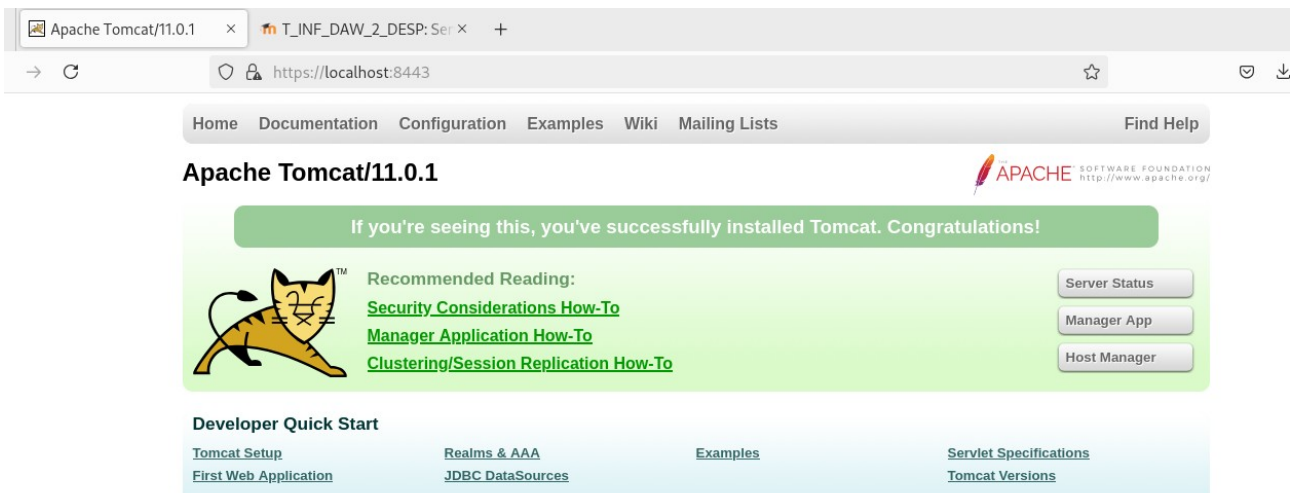
Ahora vamos al archivo server.xml ubicado en /opt/tomcat/conf  
y descomentamos la entrada donde el puerto es el 8443:

```
GNU nano 7.2 /opt/tomcat/conf/server.xml
<!-- Define an SSL/TLS HTTP/1.1 Connector on port 8443 with HTTP/2
This connector uses the NIO implementation. The default
SSLImplementation will depend on the presence of the APR/native
library and the useOpenSSL attribute of the AprLifecycleListener.
Either JSSE or OpenSSL style configuration may be used regardless of
the SSLImplementation selected. JSSE style configuration is used below.
-->
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
    maxThreads="150" SSLEnabled="true">
    <UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />
    <SSLHostConfig>
        <Certificate certificateKeystoreFile="/usr/lib/jvm/java-1.7.0-openjdk-amd64/davidrl.jks"
            certificateKeystorePassword="abc123." type="RSA" />
    </SSLHostConfig>
</Connector>

<!-- Define an AJP 1.3 Connector on port 8009 -->
```

Añadimos en la etiqueta <Certificate> el path al keystore que hemos generado y cambiamos también la contraseña por la del certificado.

Ahora vamos a <https://localhost:8443> y comprobamos el acceso:



## 4. Integración con un IDE

Raliza la integración de Tomcat con un IDE de tu elección (IntelliJ IDEA, Eclipse, Netbeans, Visual Studio Code,...)

Pistas:

[How to Deploy a WAR File to Tomcat | Baeldung](#)

[Visual Studio Code Tomcat y su configuración - Arquitectura Java](#)

[Instalación de Tomcat y configuración con Visual Studio Code - YouTube](#)

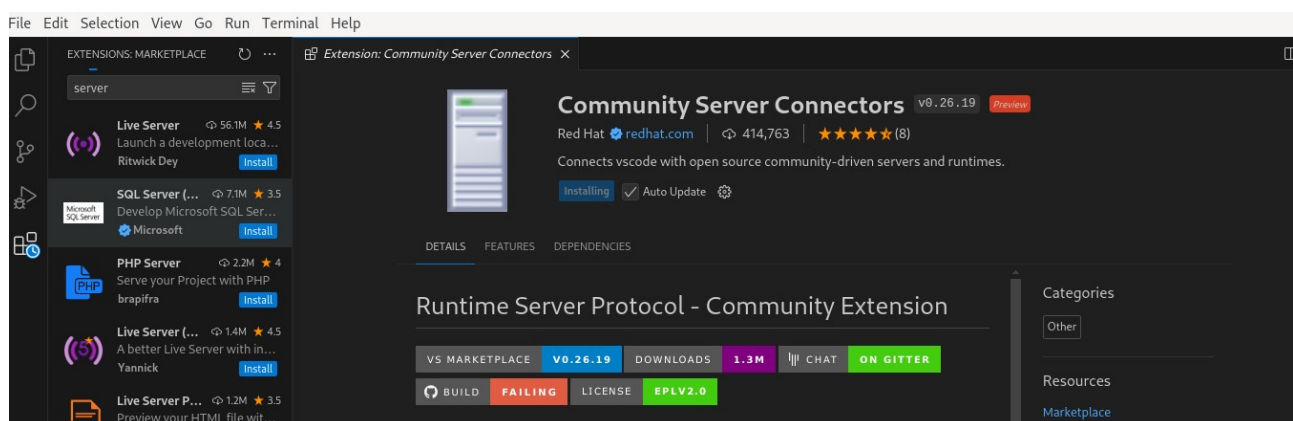
[\(Microsoft Word - Cap\355tulo 6. Eclipse con el plug-in de TOMCAT.doc\) \(us.es\)](#)

[Cómo configurar Eclipse y Apache Tomcat para desarrollar aplicaciones Java Web - YouTube](#)

[AGREGAR EL SERVIDOR TOMCAT 9 A NETBEANS – VIDELCLOUD \(wordpress.com\)](#)

[Instalar y configurar Apache Tomcat 9 en NetBeans - YouTube](#)

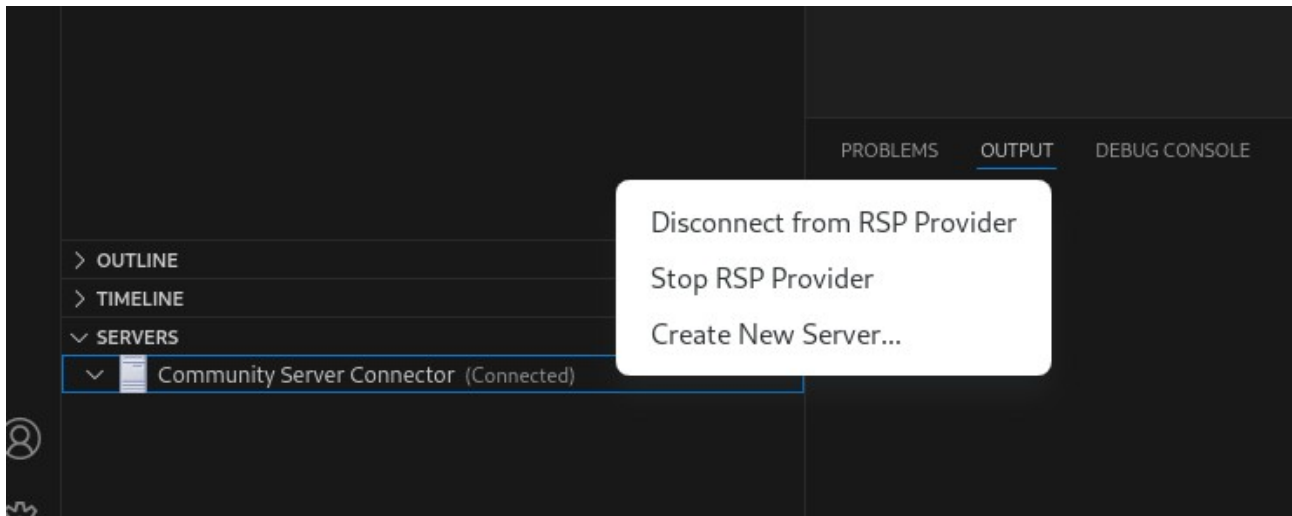
Lo voy a hacer en VSCode, vamos a la pestaña de extensiones y buscamos el Community Server Connector y lo instalamos:



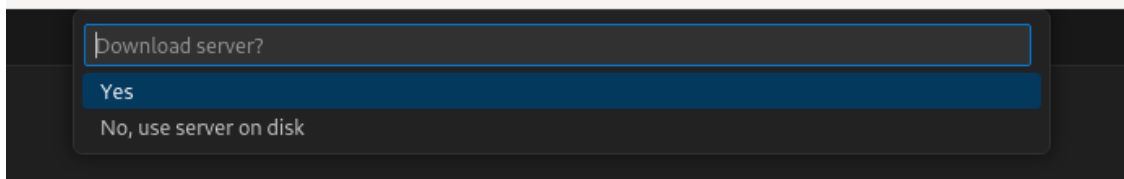
Reiniciamos el vscode una vez este instalado.

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

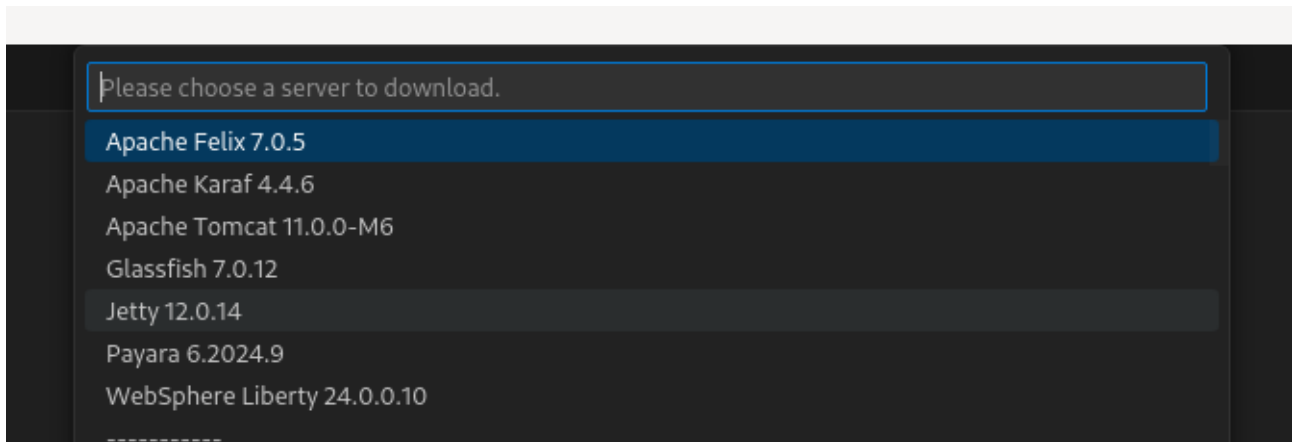
Ahora con el reiniciado, abajo a la derecha, y tendremos una nueva pestaña llamada “servers” desplegamos esa nueva pestaña y damos click derecho sobre “Community Server Conector” y pinchamos sobre “Create New Server...”



El instalador no realizara varias preguntas:



Le decimos que si.

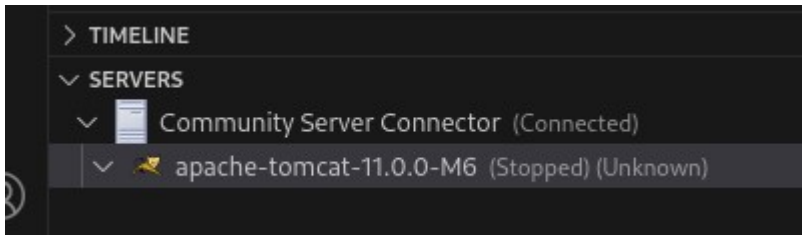


Nos da a escoger servidor, en este caso sera Apache Tomcat 11.0.0.0-M6

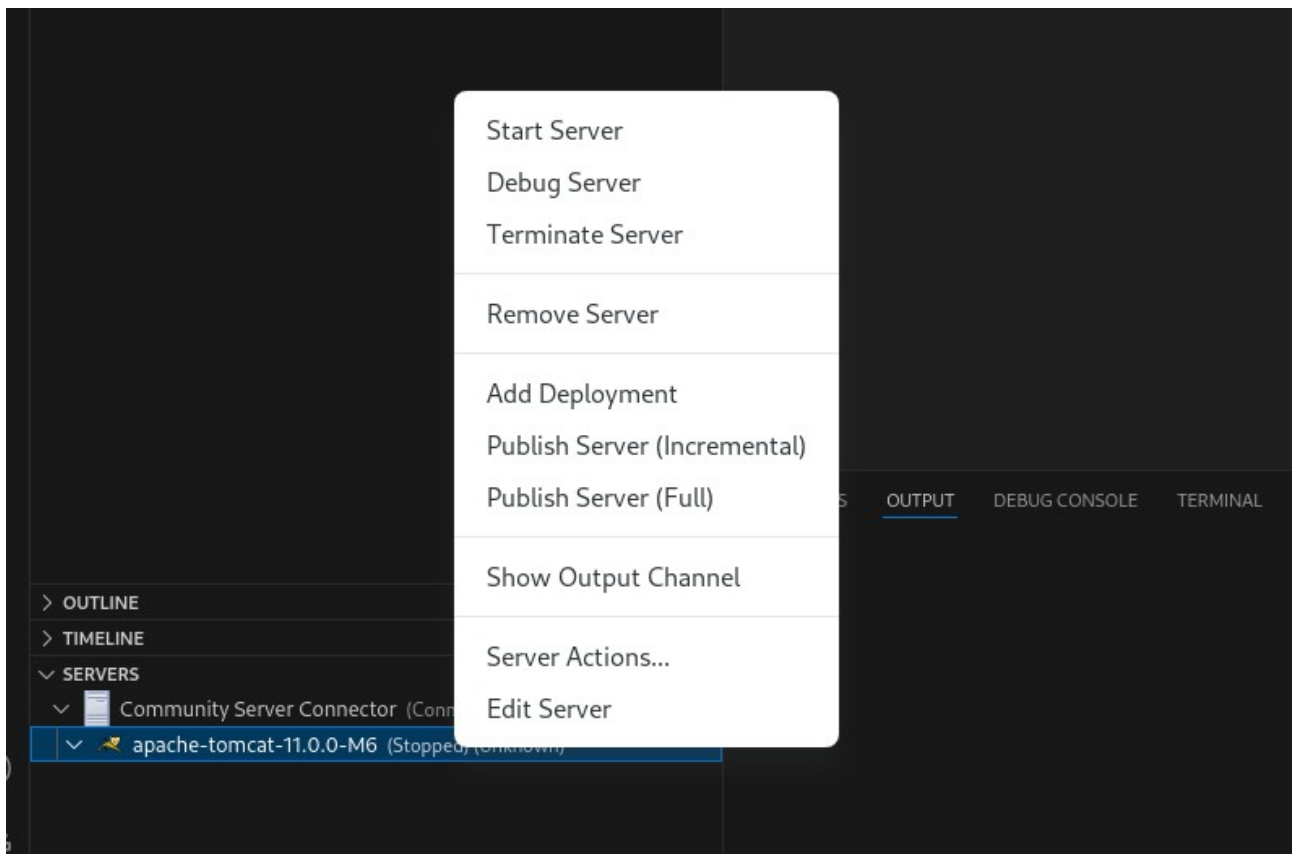


## SERVIDOR DE APLICACIONES – APACHE TOMCAT

Le damos a continuar y aceptar el resto de preguntas y desplegará el servidor tomcat:



Ahora podemos arrancar este servidor dando click derecho y Start Server:



Para hacer un despliegue de una aplicación le daremos a “Add deployment” y seleccionaremos el archivo a desplegar.

## 5. Cuestiones

**a) ¿Qué versión de Apache Tomcat instalarás dependiendo de tu versión de Java?**

<https://tomcat.apache.org/whichversion.html>

Debemos consultar la version de Java dependiendo de la version del Apache, lo podemos consultar en el link de arriba

**a.1) Y para la versión 8?**

Para la version 8 de Java instalaremos la version de Tomcat 9.0.x

**a.2) Y para la versión 21?**

A partir de la version 17 de Java se instalaran las versiones de Tomcat 11.0.x

**b) ¿Qué otros servidores de aplicaciones hay en el mercado? ¿Cuáles son software libre y cuales productos comerciales?**

- Nodejs.

-Tomcat es de software libre y por ejemplo Azure App Service es un producto comercial.

**c) Una de las cuestiones a tener en cuenta es el rendimiento de las aplicaciones. ¿Sabrías indicar alguna herramienta para pruebas de carga?**

Jmeter,LoadView,Gatling.

**d) Otra de las cuestiones a tener en cuenta es la monitorización del servidor de aplicaciones. ¿Sabrías indicarme alguna herramienta para monitorizar tomcat u otro servidor de aplicaciones? ¿Qué indicadores puede interesar monitorizar?**

Un servidor Zabbix puede ser una herramienta muy util para monitorizar multiples servicios.

Me interesaria monitorizar la carga de la maquina que realiza el hosting de la aplicación, el numero de peticiones, y el estado del servicio.

Pistas:

[Apache Tomcat monitoring and integration with Zabbix](#)

[Apache Tomcat Monitoring Guide & 5 Best Tools for 2024 \(comparitech.com\)](#)

[Apache Tomcat - Nagios Exchange](#)

[JMeter vs Other Performance Testing Tools | automateNow](#)

[List of application servers - Wikipedia](#)

## SERVIDOR DE APLICACIONES – APACHE TOMCAT

[List of application servers - Wikipedia \(aimultiple.com\)](#)

[WebLogic Server | Oracle España](#)

[WebSphere Application Server | IBM](#)

[WildFly](#)