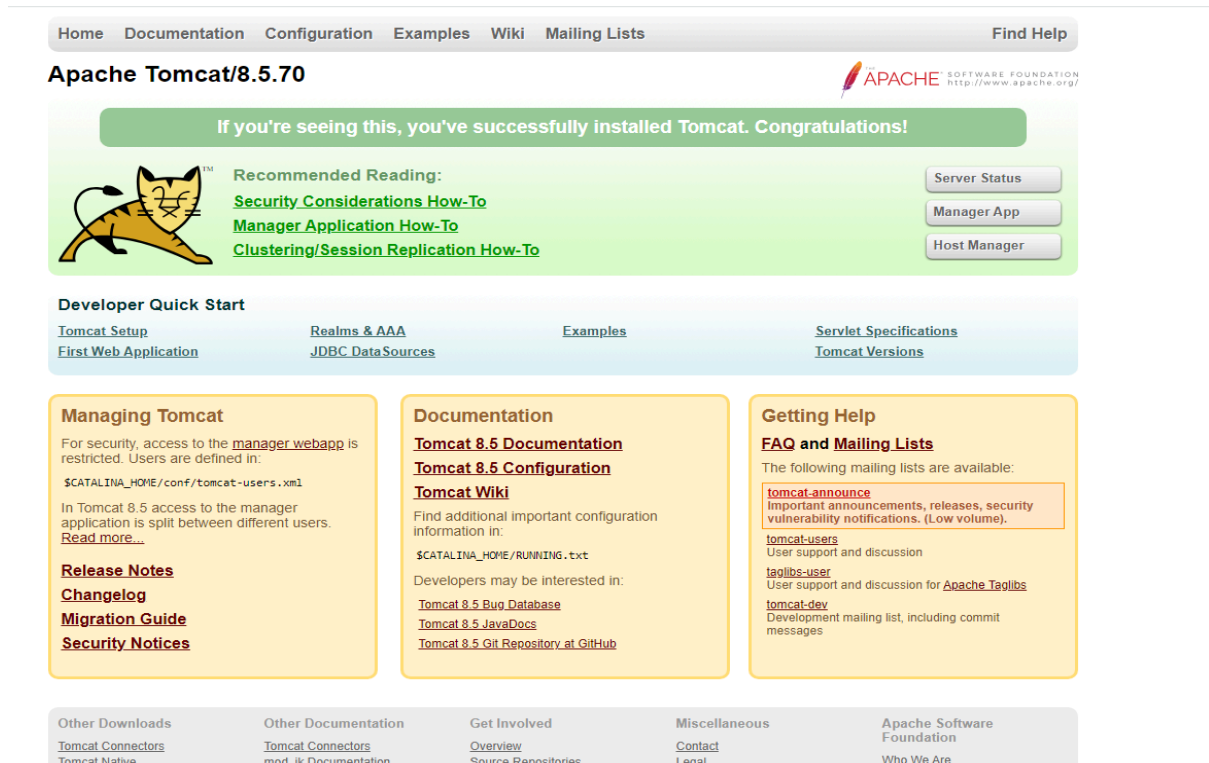


1. Funcionamiento de Tomcat en Windows con XAMPP

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

a. Arranque y configuración inicial



b. Acceso a opciones de gestión y administración

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

En el archivo tomcat-users.xml hago estos cambios:

pongo los roles

```
Will also need to set the password to something appropriate.
-->
<!--
<user username="admin" password="<must-be-changed>" roles="manager-gui"/>
<user username="robot" password="<must-be-changed>" roles="manager-script"/>
<role rolename="manager-gui"/>
<role rolename="admin-gui"/>
<user username="admin" password="abc123." roles="manague-gui, admin-gui"/>
-->
<role rolename="manager-gui"/>
<role rolename="manager-status"/>
<role rolename="admin-gui"/>
<user username="admin" password="1234.Abc" roles="admin,manager,admin-gui,manager-gui"/>
-->
<!--
The sample user and role entries below are intended for use with the
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
```



Estado de Servidor

Gestor

Listar Aplicaciones	Ayuda HTML de Gestor	Ayuda de Gestor	Estado Completo de Servidor
-------------------------------------	--------------------------------------	---------------------------------	---

Información de Servidor

Versión de Tomcat	Versión JVM	Vendedor JVM	Nombre del SO	Versión de SO	Arquitectura de SO	NombreDeMáquina	Dirección IP
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: 89.98 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	20.97 MB (3%)
PS Old Gen	Heap memory	85.50 MB	85.50 MB	1365.50 MB	7.06 MB (0%)
PS Survivor Space	Heap memory	5.00 MB	5.00 MB	5.00 MB	4.97 MB (99%)
Code Cache	Non-heap memory	2.43 MB	6.81 MB	240.00 MB	6.73 MB (2%)
Compressed Class Space	Non-heap memory	0.00 MB	2.25 MB	1024.00 MB	1.99 MB (0%)
Metaspace	Non-heap memory	0.00 MB	18.75 MB	-0.00 MB	18.04 MB

"http-nio-8080"

Max threads: 200 Current thread count: 3 Current threads busy: 1 Keep alive sockets count: 1
Max processing time: 376 ms Tiempo de procesamiento: 0.444 s Request count: 2 Error count: 1 Bytes received: 0.00 MB Bytes sent: 0.01 MB

Stage	Time	Bytes Sent	Bytes Recv	Client (Forwarded)	Client (Actual)	VHost	Solicitud
S	92 ms	0 KB	0 KB	0.0.0.0.0.0.1	0.0.0.0.0.0.1	localhost	GET /manager/status HTTP/1.1

P: Parse and prepare request S: Service F: Finishing R: Ready K: Keepalive

Copyright © 1999-2021, Apache Software Foundation



Gestor de Máquina Virtual de Tomcat

Mensaje: OK

Gestor de Máquina

Lista de Máquinas Virtuales	Ayuda de Gestor de Máquina HTML (¡En breves!)	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 ☐

Configuración persistente

☒ Save current configuration (including virtual hosts) to server.xml and per web application context.xml files

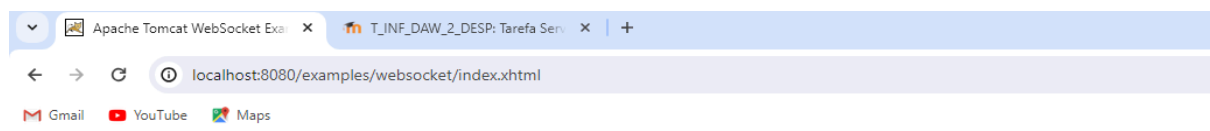


Gestor de Aplicaciones Web de Tomcat

Mensaje:	OK				
Gestor					
Lista de 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	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar a 30 minutos
/docs	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar a 30 minutos
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar a 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	2	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar a 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar a 30 minutos
Desplegar					
Desplegar directorio o archivo WAR localizado en servidor					

c. Ejemplos

Muestra su funcionamiento con alguno de los ejemplos disponibles



Apache Tomcat WebSocket Examples

- [Echo example](#)
- [Chat example](#)
- [Multiplayer snake example](#)
- [Multiplayer drawboard example](#)

d. Despliega tu propia aplicación web

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

En el Gestor de Aplicaciones Web de Tomcat: desplegamos el .war

Gestor de Aplicaciones Web de Tomcat

Mensaje:

FALLO - El fichero war [sample.war] ya existe en el servidor

Gestor

Listar Aplicaciones

Ayuda HTML de Gestor

Ayuda de Gestor

Estado de Se

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado	Welcome to Tomcat	true	0	<div>Arrancar</div> <div>Parar</div> <div>Recargar</div> <div>Replegar</div> <div>Expirar sesiones sin trabajar 30 minutos</div>
/docs	Ninguno especificado	Tomcat Documentation	true	0	<div>Arrancar</div> <div>Parar</div> <div>Recargar</div> <div>Replegar</div> <div>Expirar sesiones sin trabajar 30 minutos</div>
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	<div>Arrancar</div> <div>Parar</div> <div>Recargar</div> <div>Replegar</div> <div>Expirar sesiones sin trabajar 30 minutos</div>
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	1	<div>Arrancar</div> <div>Parar</div> <div>Recargar</div> <div>Replegar</div> <div>Expirar sesiones sin trabajar 30 minutos</div>
/manager	Ninguno especificado	Tomcat Manager Application	true	1	<div>Arrancar</div> <div>Parar</div> <div>Recargar</div> <div>Replegar</div> <div>Expirar sesiones sin trabajar 30 minutos</div>
sample	Ninguno especificado	Hello, World Application	true	0	<div>Arrancar</div> <div>Parar</div> <div>Recargar</div> <div>Replegar</div> <div>Expirar sesiones sin trabajar 30 minutos</div>

Desplegar

Desplegar directorio o archivos WAR localizados en servidor

después pulsamos en /sample

←

→


↺

localhost:8080/sample/

Gmail

YouTube

Maps



Sample "Hello, World" Application

This is the home page for a sample application used to illustrate the source directory organization of a web application utilizing the principles outlined in the Application Developer's Guide.

To prove that they work, you can execute either of the following links:

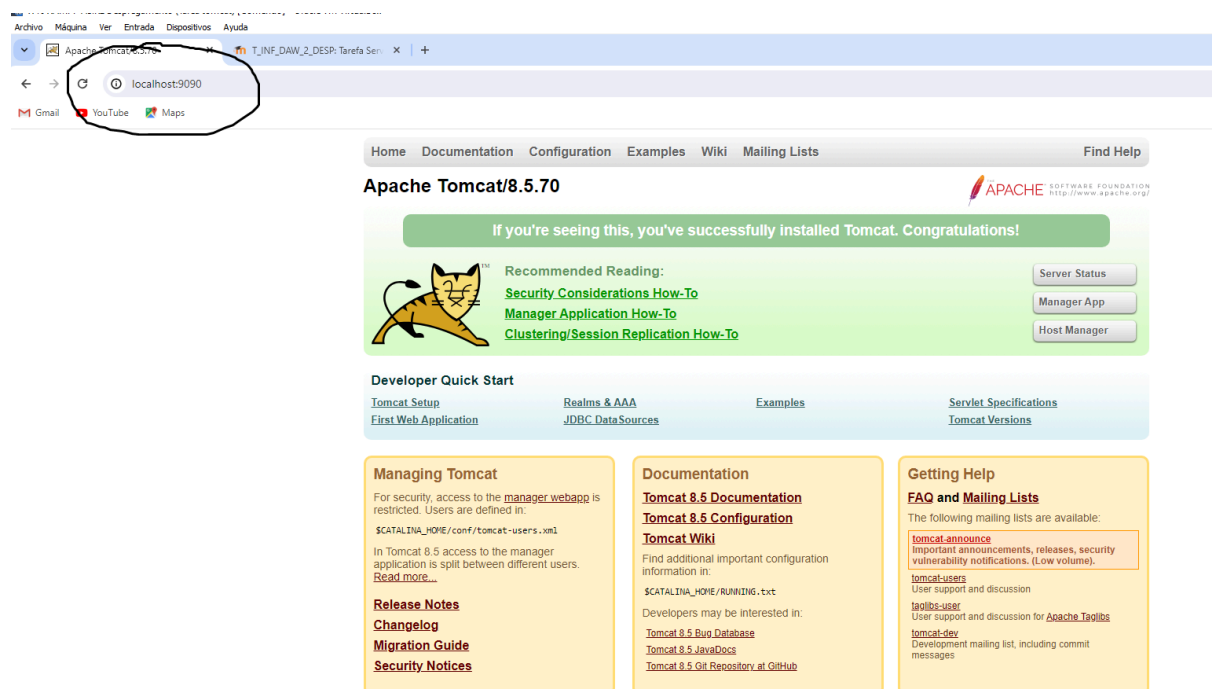
- To a [JSP page](#).
- To a [servlet](#).

e. Cambia el puerto donde escucha Tomcat

abrimos el archivo server.xml y hacemos el cambio : donde pone puerto 8080 a 9090

```
server.xml
58      maxThreads="150" minSpareThreads="4"/>
59      -->
60
61      <!-- A "Connector" represents an endpoint by which requests are received
62      and responses are returned. Documentation at :
63      Java HTTP Connector: /docs/config/http.html
64      Java AJP Connector: /docs/config/ajp.html
65      APR (HTTP/AJP) Connector: /docs/apr.html
66      Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
67      -->
68      <Connector port="9090" protocol="HTTP/1.1"
69      connectionTimeout="20000"
70      redirectPort="8443" />
71      <!-- A "Connector" using the shared thread pool-->
72      <!--
73      <Connector executor="tomcatThreadPool"
74      port="8080" protocol="HTTP/1.1"
75      connectionTimeout="20000"
76      redirectPort="8443" />
77      -->
78      <!-- Define an SSL/TLS HTTP/1.1 Connector on port 8443
79      This connector uses the NIO implementation. The default
80
```

comprobamos que funciona



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

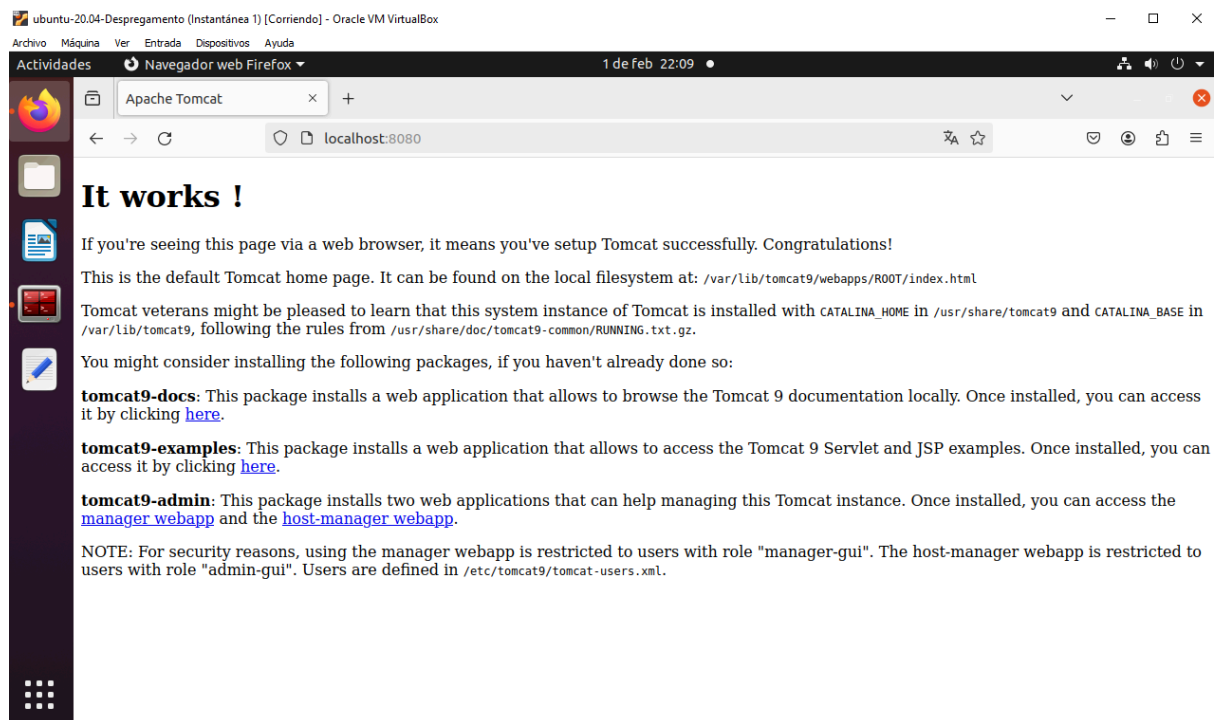
Ahora instalaré el JDK

`sudo apt install default-jdk`

Una vez instalado comprobamos la versión de java con : `java -version`

Instalamos el tomcat: `sudo apt install tomcat9`

Comprobamos que funciona



b. Despliega tu propia aplicación web

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

Instalamos “tomcat-admin” con: `sudo apt-get install tomcat9-admin`

accedemos al tomcat-manager y desplegamos el archivo .war

				Expirar sesiones sin trabajar ≥ 30
Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30

Archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):
 Version (for parallel deployment):
 URL de archivo de Configuración XML:
 URL de WAR o Directorio:
 Desplegar

Seleccione archivo WAR a cargar No se ha seleccionado ningún archivo.

Files

TLS host name (optional)

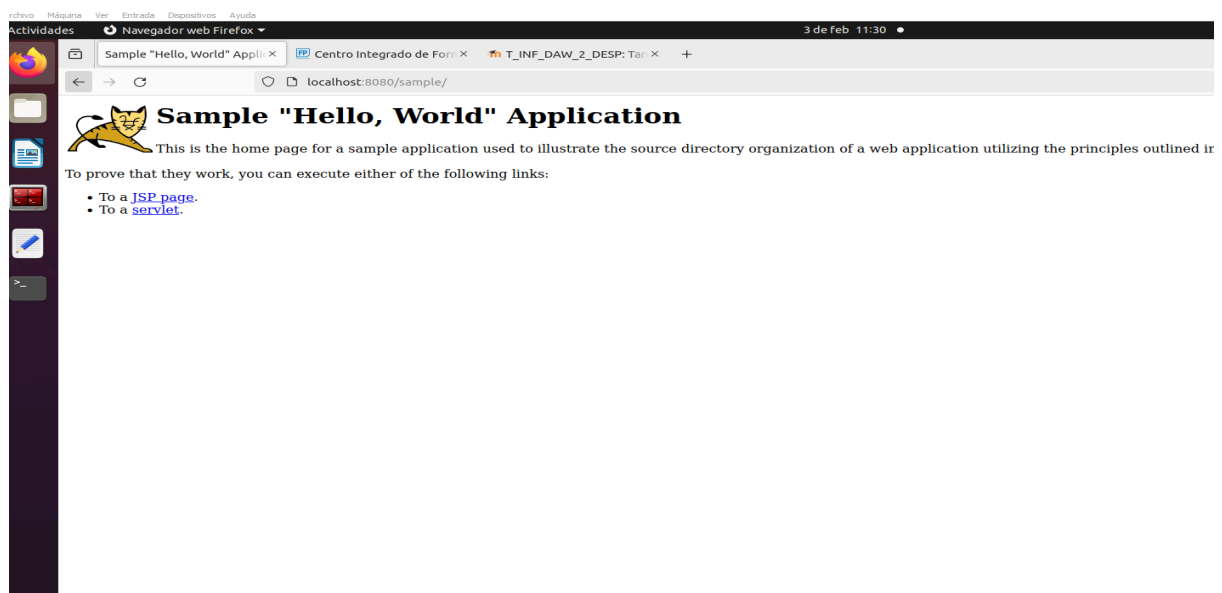
Web application has caused memory errors when stopping, reloading or redeploying.

Este chequeo de diagnóstico disparará una colección completa de basura. Utilízalo con extremo cuidado en sistemas en producción.

Diagnostic

List the configured TLS virtual hosts and the ciphers for each.

Lista los virtual hosts configurados con TLS y la cadena de certificado para cada uno de ellos.

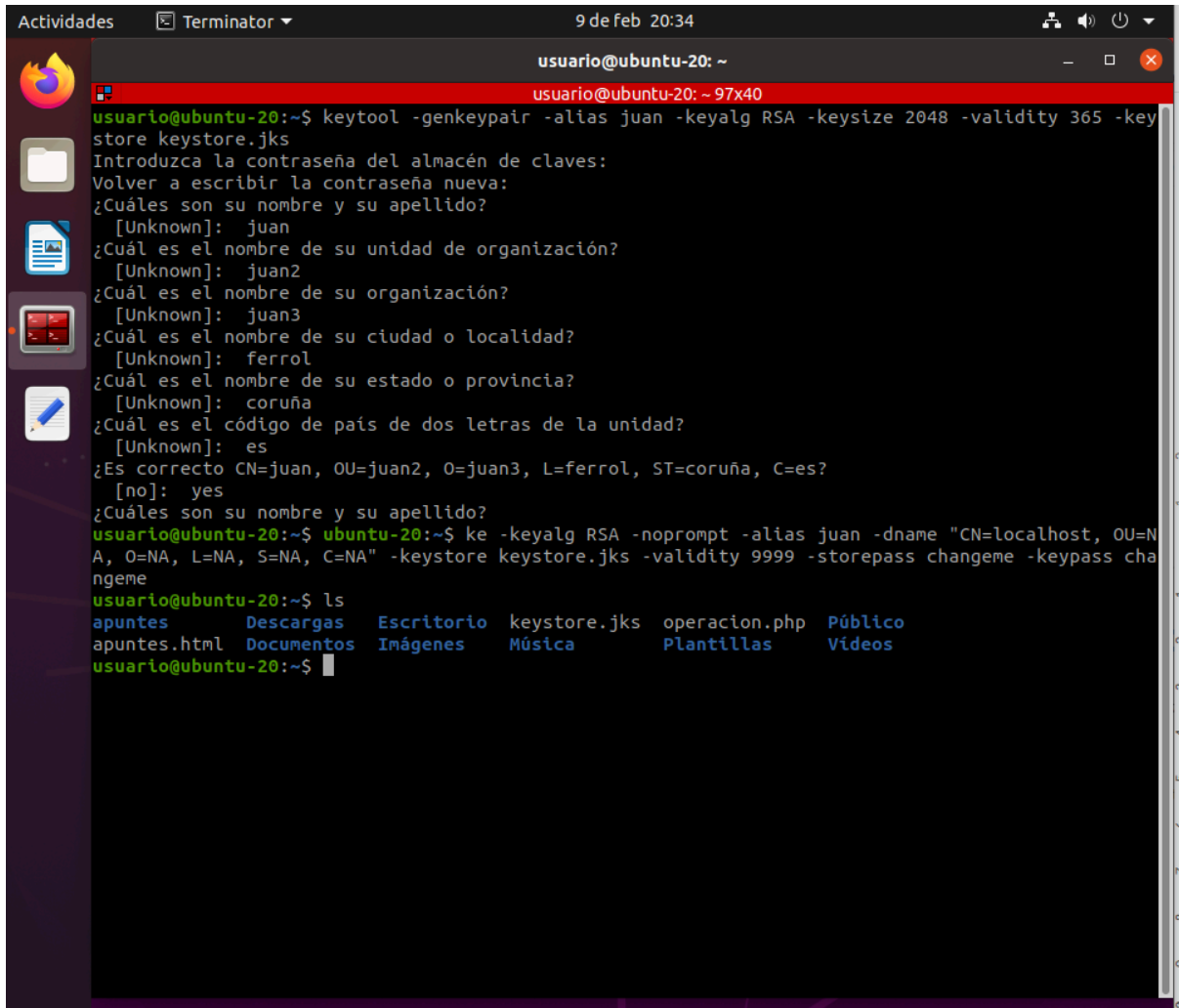


3. Securización

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

Para crear el almacén de claves usamos :

```
keytool -genkey -keyalg RSA -noprompt -alias juan -dname "CN=localhost, OU=NA, O=NA, L=NA, S=NA, C=NA" -keystore keystore.jks -validity 9999 -storepass changeme -keypass changeme
```



```
Actividades Terminator 9 de feb 20:34
usuario@ubuntu-20: ~
usuario@ubuntu-20: ~ 97x40
usuario@ubuntu-20:~$ keytool -genkeypair -alias juan -keyalg RSA -keysize 2048 -validity 365 -keystore keystore.jks
Introduzca la contraseña del almacén de claves:
Volver a escribir la contraseña nueva:
¿Cuáles son su nombre y su apellido?
[Unknown]: juan
¿Cuál es el nombre de su unidad de organización?
[Unknown]: juan2
¿Cuál es el nombre de su organización?
[Unknown]: juan3
¿Cuál es el nombre de su ciudad o localidad?
[Unknown]: ferrol
¿Cuál es el nombre de su estado o provincia?
[Unknown]: coruña
¿Cuál es el código de país de dos letras de la unidad?
[Unknown]: es
¿Es correcto CN=juan, OU=juan2, O=juan3, L=ferrol, ST=coruña, C=es?
[no]: yes
¿Cuáles son su nombre y su apellido?
usuario@ubuntu-20:~$ ke -keyalg RSA -noprompt -alias juan -dname "CN=localhost, OU=NA, O=NA, L=NA, S=NA, C=NA" -keystore keystore.jks -validity 9999 -storepass changeme -keypass changeme
usuario@ubuntu-20:~$ ls
apuntes Descargas Escritorio keystore.jks operacion.php Público
apuntes.html Documentos Imágenes Música Plantillas Vídeos
usuario@ubuntu-20:~$
```

Al poner en el buscador <https://localhost:8443/> la seguridad está funcionando porque aparece el icono del escudo

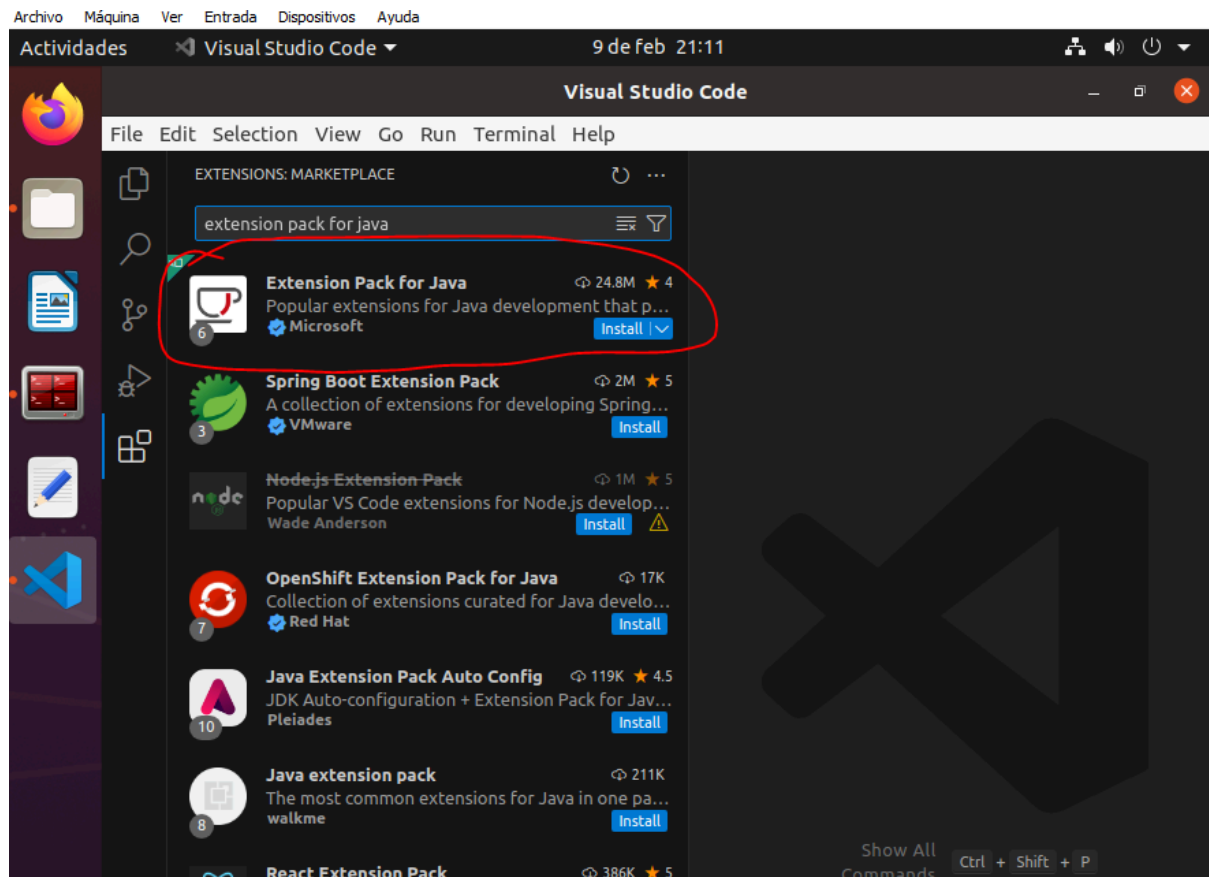
4. Integración con un IDE

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

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

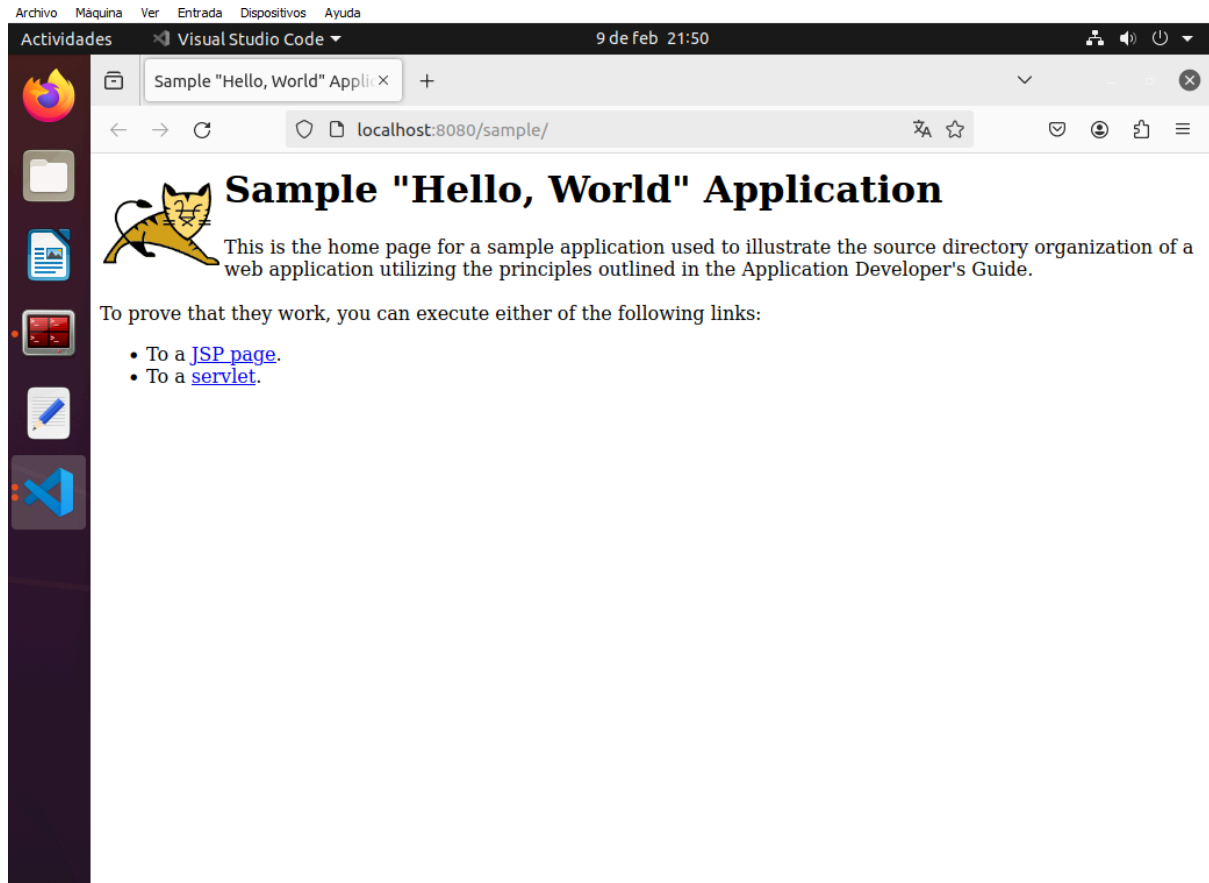
Instalamos el Visual Studio Code con : `sudo snap install --classic code`

Un vez instalado descargamos esta extensión de Java



Para el siguiente paso hice uso del tutorial facilitado

<https://code.visualstudio.com/docs/java/java-tomcat-jetty> . En el cual se detalla perfectamente. El fallo que tuve es que no había parado el tomcat que tenía instalado.



5. Cuestiones

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

a.1) Y para la versión 8? [Se recomienda Apache Tomcat8x, ya que son versiones compatibles con Java 8](#)

a.2) Y para la versión 21? [A partir de Java 9, se introdujeron cambios significativos en el modelo de módulos, y algunas APIs de Tomcat 8 pueden no ser compatibles. Por lo tanto, se recomienda usar Apache Tomcat 9.x o superior para estas versiones de Java](#)

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

[De software libre, tenemos:](#)

WildFly: Desarrollado por Red Hat, es un servidor de aplicaciones Java EE de código abierto y es compatible con una amplia gama de tecnologías Java.

Apache TomEE: Es una distribución de Apache Tomcat que agrega características de Java Enterprise Edition (Java EE) utilizando componentes de Apache OpenEJB y otros proyectos de Apache.

GlassFish: Un servidor de aplicaciones Java EE de código abierto desarrollado por Oracle. Ofrece una implementación de referencia para las especificaciones de Java EE.

Payara Server: Una bifurcación del GlassFish Server Open Source Edition. Payara Server es una solución de servidor de aplicaciones Java EE de código abierto y compatible con Eclipse MicroProfile

Comerciales:

WebLogic Server: Desarrollado por Oracle, es un servidor de aplicaciones Java EE para implementaciones empresariales de Java.

WebSphere Application Server: Desarrollado por IBM, es otro servidor de aplicaciones Java EE que ofrece una amplia gama de características y herramientas para aplicaciones empresariales.

JBoss EAP (Enterprise Application Platform): Versión comercial de WildFly, desarrollado por Red Hat, que ofrece soporte y características adicionales para entornos empresariales.

TmaxSoft JEUS: Un servidor de aplicaciones Java EE desarrollado por TmaxSoft, que se enfoca en alta disponibilidad y rendimiento.

Microsoft Azure App Service: Aunque no es un servidor de aplicaciones tradicional, Azure App Service de Microsoft proporciona un entorno administrado para ejecutar aplicaciones web y API en la nube de Microsoft Azure.

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

Entre varias que hay tenemos :

Apache JMeter: Una herramienta de prueba de carga de código abierto desarrollada por Apache. Es altamente configurable y puede utilizarse para simular diferentes escenarios de carga, así como para medir el rendimiento y la escalabilidad de las aplicaciones web.

Gatling: Otra herramienta de prueba de carga de código abierto que se basa en Scala. Gatling permite escribir escenarios de carga como código y ofrece informes detallados sobre el rendimiento de la aplicación.

LoadRunner: Desarrollado por Micro Focus, es una herramienta comercial que ofrece una amplia gama de funcionalidades para pruebas de carga, incluyendo la capacidad de simular

miles de usuarios virtuales y medir el rendimiento de la aplicación en diferentes condiciones de carga.

BlazeMeter: Una plataforma de prueba de carga en la nube que permite a los equipos ejecutar pruebas de carga escalables utilizando JMeter, Gatling y otros motores de prueba de carga. Proporciona informes detallados y análisis en tiempo real.

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?

Apache Tomcat Manager App: Tomcat incluye una aplicación de gestión que proporciona información básica sobre el estado del servidor y las aplicaciones desplegadas. Es accesible a través de un navegador web

JavaMelody: Una biblioteca de monitorización de código abierto para aplicaciones Java que se puede integrar fácilmente en aplicaciones web desplegadas en Tomcat.

Prometheus y Grafana: Prometheus es una herramienta de código abierto para la recopilación y el almacenamiento de métricas, mientras que Grafana es una herramienta para la visualización de datos. Se pueden usar juntas para monitorear y visualizar métricas de servidores de aplicaciones como Tomcat.

Depende de las necesidades de cada trabajo, pero podemos monitorizar:

Tráfico de red: Monitorizar el tráfico de red entrante y saliente puede ayudar a identificar problemas de red y ataques de denegación de servicio (DoS)

Número de solicitudes HTTP: Seguir el número de solicitudes HTTP procesadas por el servidor puede ayudar a entender la carga de trabajo y detectar patrones de tráfico inusuales.

Tiempo de respuesta de la aplicación: Supervisar el tiempo de respuesta de las solicitudes HTTP puede ayudar a identificar problemas de rendimiento en la aplicación

Errores y excepciones: Registrar y monitorizar errores y excepciones en el servidor de aplicaciones puede ayudar a identificar problemas de funcionalidad y seguridad.