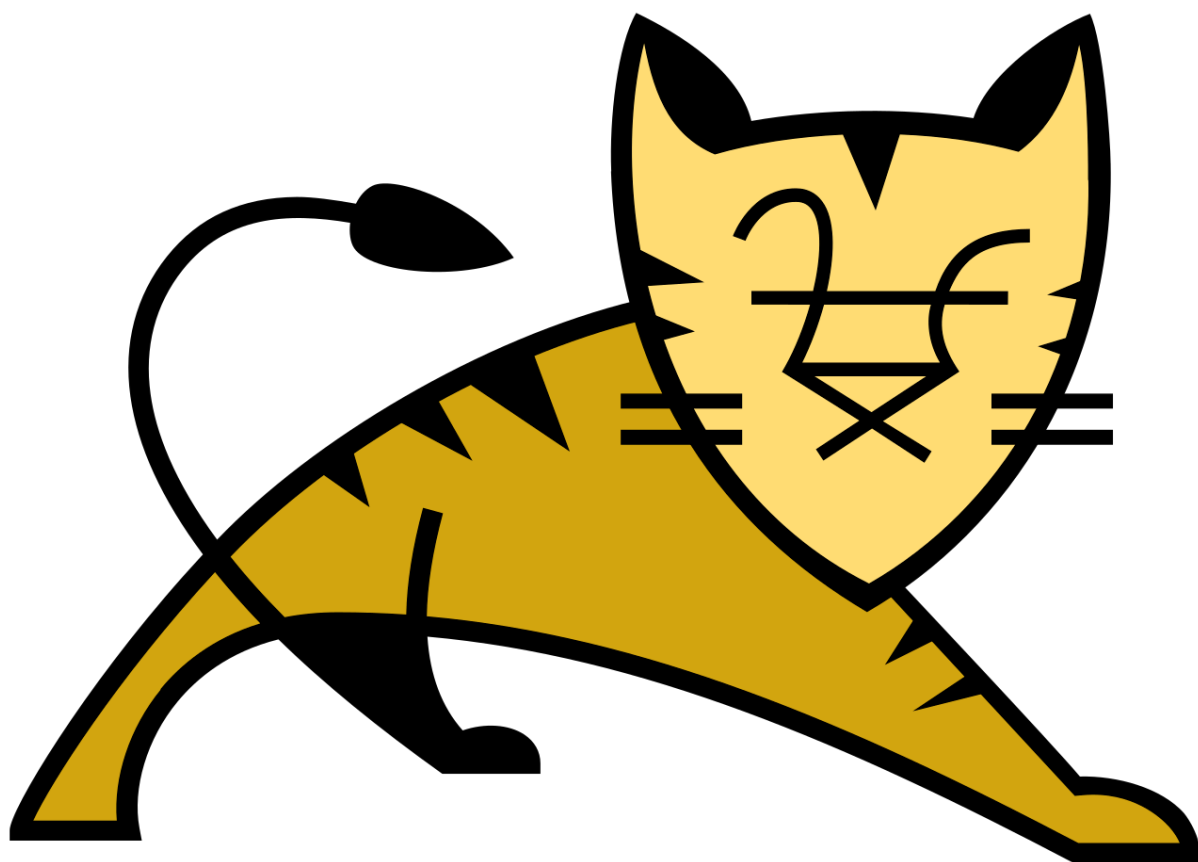


Servidor de Aplicaciones

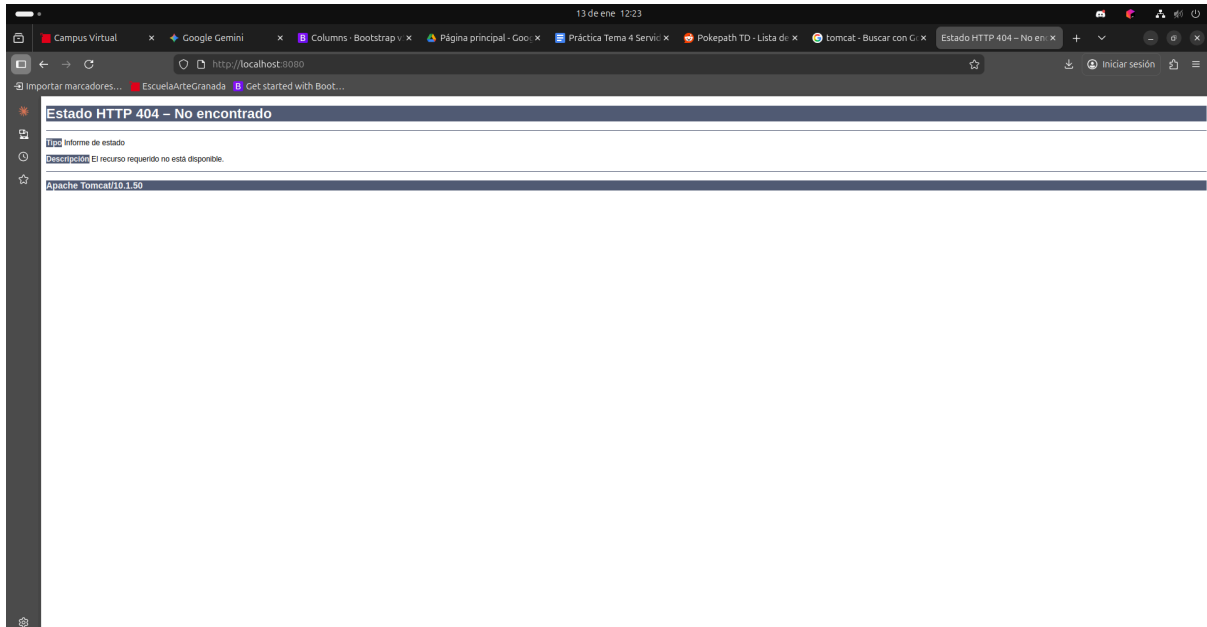


Javier Granizo Aguilar

Crear tu archivo docker-compose y comprobar que tu contenedor se levanta con normalidad.....	3
Generar los certificados autofirmados SSL/TLS y modificar docker-compose en consecuencia.....	4
Instala Maven y comprueba que lo has hecho correctamente.....	5
Crea el archivo pom.xml y asegúrate de que concuerde con la versión de java que está usando maven.....	6
Crea tu archivo server.xml, copiando el generado por Tomcat en tu proyecto.....	7
Modifica el server.xml para indicar donde encontrará el contenedor los certificados... sudo chmod 777 seguido de la ruta y arreglado;.....	7
Crea tu aplicación calculadora en dos pasos, un archivo java que contendrá la lógica y un index.jsp que será tu interfaz.....	8
Añade tu archivo web.xml e intenta probar a generar el archivo.war con maven.....	9
error:me decía que le faltaba el archivo login lo e copiapo del siguiente punto y solucionado.....	10
Añade roles a tu app con el archivo tomcat-users.xml y comprobar su correcto funcionamiento, tanto de los usuarios con sus contraseñas como con las restricciones impuestas a los roles.....	10

Paso 1:

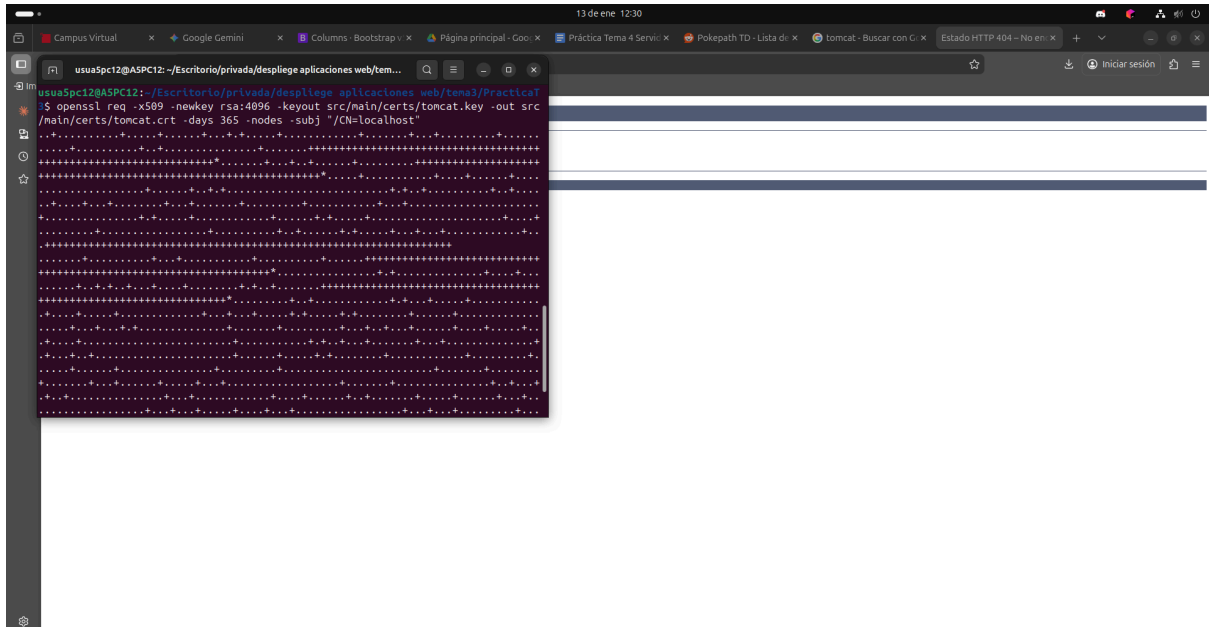
Crear tu archivo docker-compose y comprobar que tu contenedor se levanta con normalidad.



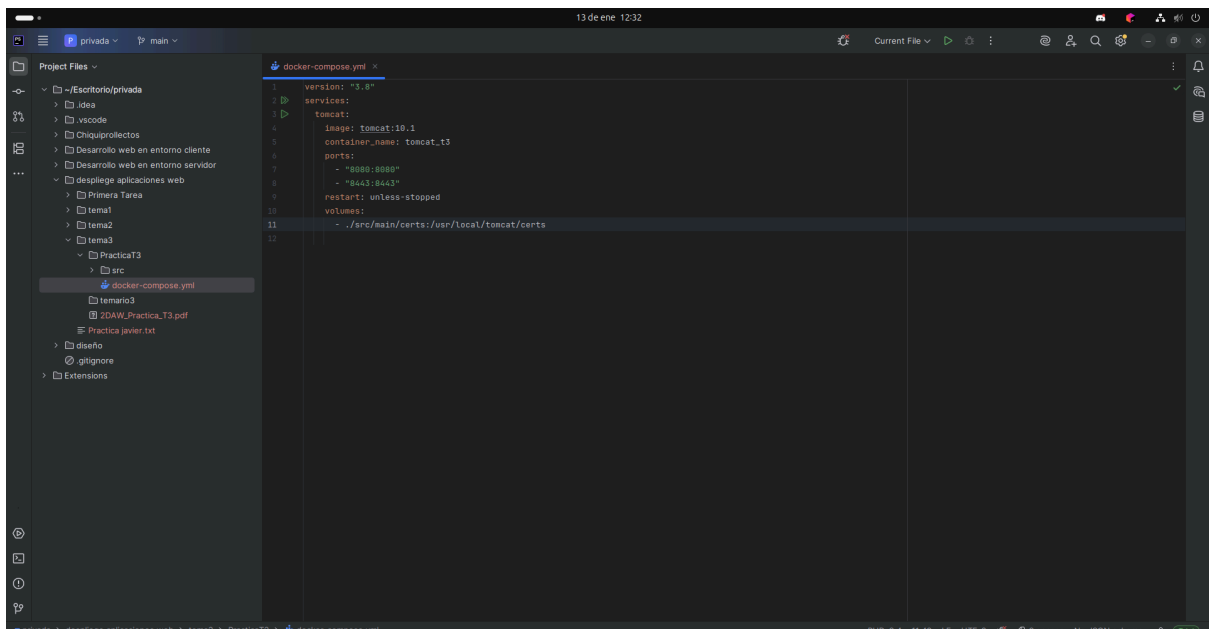
```
usua5pc12@A5PC12: ~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3
✓ 4f4fb700ef54 Pull complete 10.7s
✓ 901b8cfcfda7 Pull complete 11.0s
✓ 06902153f937 Pull complete 11.4s
✓ 939d8ec703a2 Pull complete 12.5s
[+] Running 2/2
✓ Network practicat3_default Created 0.3s
✓ Container tomcat_t3 Started 10.5s
usua5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$ sudo docker compose up -d
WARN[0000] /home/usua5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 1/1
✓ Container tomcat_t3 Running 0.0s
usua5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$ sudo docker compose down
WARN[0000] /home/usua5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 2/2
✓ Container tomcat_t3 Removed 2.1s
✓ Network practicat3_default Removed 0.4s
usua5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$
```

Paso 2:

Generar los certificados autofirmados SSL/TLS y modificar docker-compose en consecuencia.



```
usua5pc12@ASPC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$ openssl req -x509 -newkey rsa:4096 -keyout src/main/certs/tomcat.key -out src/main/certs/tomcat.crt -days 365 -nodes -subj "/CN=localhost"
```



```
version: "3.8"
services:
  tomcat:
    image: tomcat:10.1
    container_name: tomcat_t3
    ports:
      - "8080:8080"
      - "8443:8443"
    restart: unless-stopped
    volumes:
      - ./src/main/certs:/usr/local/tomcat/certs
```

Paso 3:

Instala Maven y comprueba que lo has hecho correctamente.

```
usua5pc12@A5PC12: ~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$
Adding debian:TUBITAK_Kamu_SM_SSL_Kok_Sertifikasi_-_Surum_1.pem
Adding debian:TunTrust_Root_CA.pem
Adding debian:TWCA_Global_Root_CA.pem
Adding debian:TWCA_Root_Certification_Authority.pem
Adding debian:UCA_Extended_Validation_Root.pem
Adding debian:UCA_Global_G2_Root.pem
Adding debian:USERTrust_ECC_Certification_Authority.pem
Adding debian:USERTrust_RSA_Certification_Authority.pem
Adding debian:vTrus_ECC_Root_CA.pem
Adding debian:vTrus_Root_CA.pem
Adding debian:XRamp_Global_CA_Root.pem
done.
Configurando maven (3.8.7-2) ...
update-alternatives: utilizando /usr/share/maven/bin/mvn para proveer /usr/bin/mvn (mvn) en modo automático
Configurando default-jre-headless (2:1.21-75+exp1) ...
Apache Maven 3.8.7
Maven home: /usr/share/maven
Java version: 21.0.9, vendor: Ubuntu, runtime: /usr/lib/jvm/java-21-openjdk-amd64
Default locale: es_ES, platform encoding: UTF-8
OS name: "linux", version: "6.14.0-37-generic", arch: "amd64", family: "unix"
usua5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$
```

Paso 4:

Crea el archivo pom.xml y asegúrate de que concuerde con la versión de java que está usando maven.

```
1 <project xmlns="http://maven.apache.org/POM/4.0.0">
2   <modelVersion>4.0.0</modelVersion>
3
4   <groupId>calculadora</groupId> <artifactId>calculadora</artifactId> <version>1.0.0</version>
5   <packaging>war</packaging> <properties>
6     <maven.compiler.release>17</maven.compiler.release>
7     <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
8   </properties>
9
10  <dependencies>
11    <dependency>
12      <groupId>jakarta.servlet</groupId>
13      <artifactId>jakarta.servlet-api</artifactId>
14      <version>6.0.0</version>
15      <scope>provided</scope>
16    </dependency>
17  </dependencies>
18
19  <build>
20    <finalName>calculadora</finalName>
21    <plugins>
22      <plugin>
23        <groupId>org.apache.maven.plugins</groupId>
24        <artifactId>maven-war-plugin</artifactId>
25        <version>3.4.0</version>
26        <configuration>
27          <failOnMissingWebXml>>false</failOnMissingWebXml>
28        </configuration>
29      </plugin>
30    </plugins>
31  </build>
32</project>
```

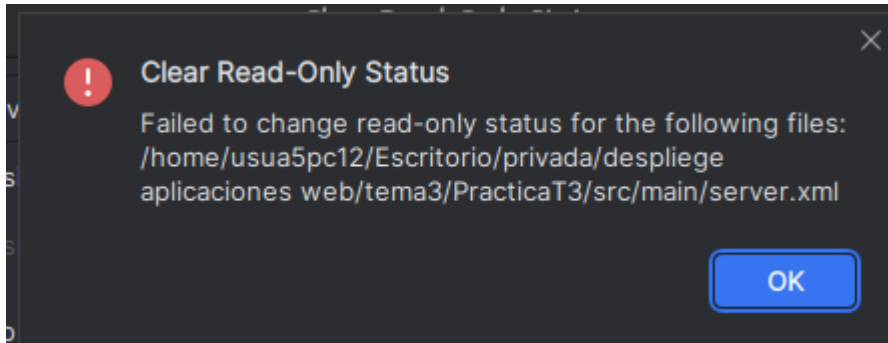
Paso 5:

Crea tu archivo server.xml, copiando el generado por Tomcat en tu proyecto.

```
usua5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$ sudo docker cp tomcat_t3:/usr/local/tomcat/conf/server.xml /home/usua5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/src/main/server.xml
Successfully copied 8.7kB to /home/usua5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/src/main/server.xml
```

Paso 6:

Modifica el server.xml para indicar donde encontrará el contenedor los certificados.



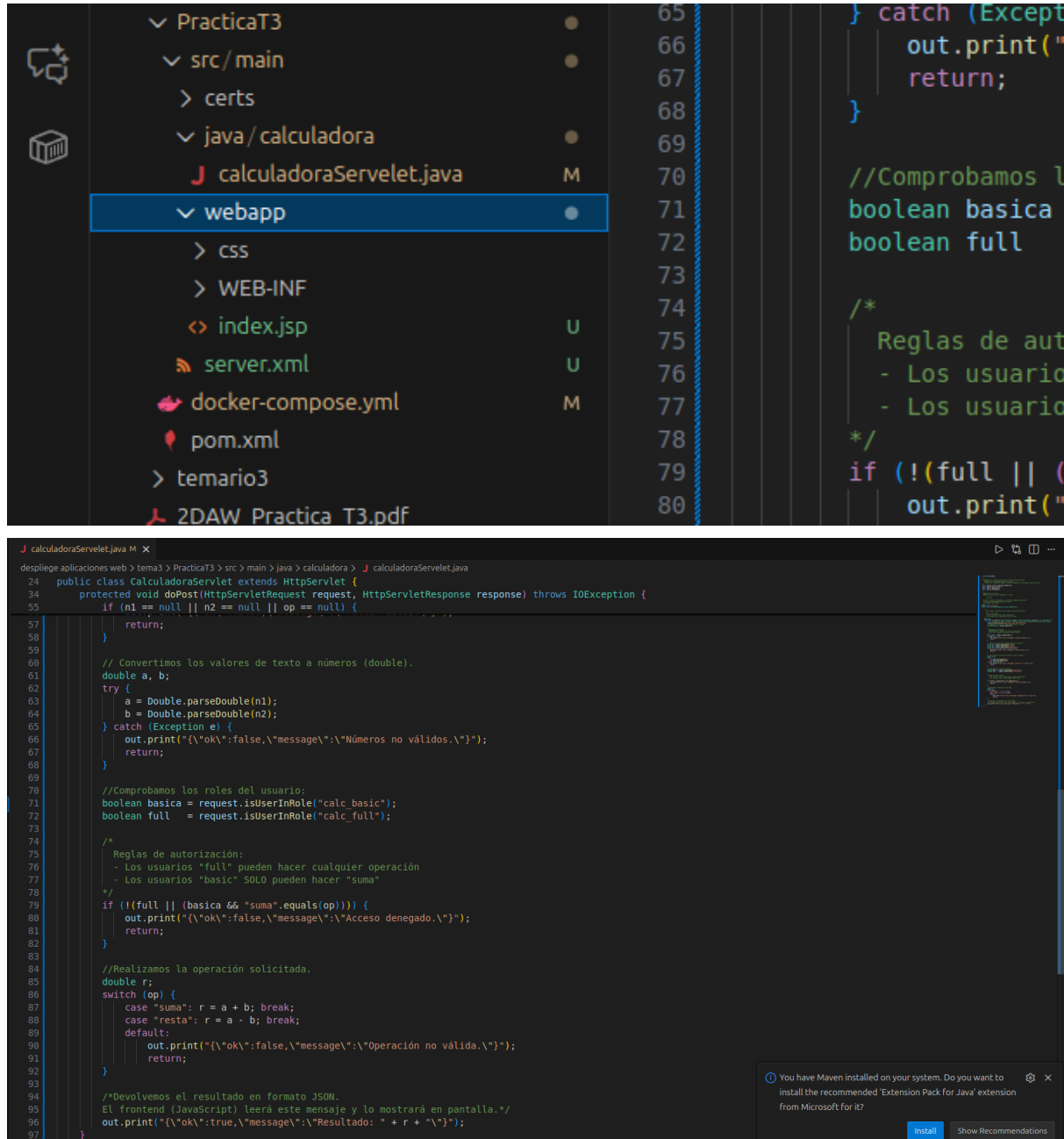
Error:no me deja editar el server.xml

sudo chmod 777 seguido de la ruta y arreglado;

```
maxThreads="150" SSLEnabled="true">
<SSLHostConfig>
  <Certificate certificateFile="/usr/local/tomcat/certs/tomcat.crt"
    certificateKeyFile="/usr/local/tomcat/certs/tomcat.key"
    type="RSA" />
</SSLHostConfig>
</Connector>
```

Paso 7:

Crea tu aplicación calculadora en dos pasos, un archivo java que contendrá la lógica y un index.jsp que será tu interfaz.



Paso 8:

Añade tu archivo web.xml e intenta probar a generar el archivo.war con maven.

```
[INFO] -----[ war ]-----
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ calculadora ---
[INFO] Deleting /home/usuario5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/target
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ calculadora ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/usuario5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.13.0:compile (default-compile) @ calculadora ---
[INFO] Recompiling the module because of changed source code.
[INFO] Compiling 1 source file with javac [debug target 21] to target/classes
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.869 s
[INFO] Finished at: 2026-01-15T14:10:55+01:00
[INFO]
usuario5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$
```

```
target
└─ classes/calculadora
    └─ CalculadoraServlet.class
└─ generated-sources/annotations
```

añadir

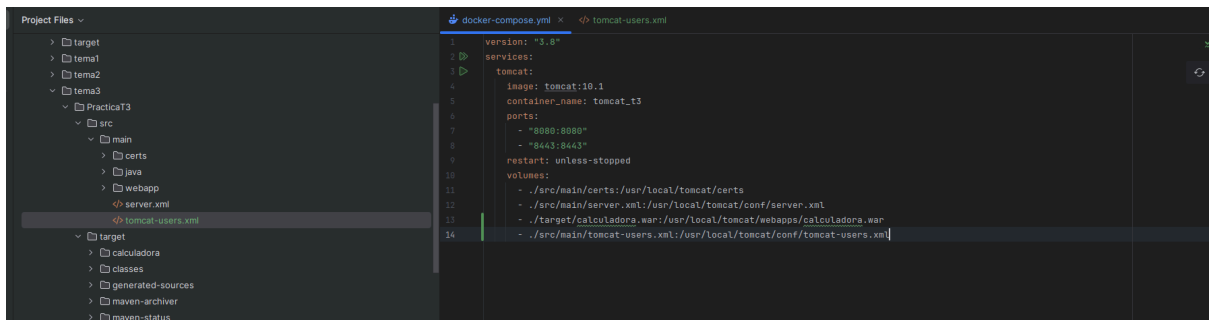
```
container_name: tomcat_t3
ports:
  - "8080:8080"
  - "8443:8443"
restart: unless-stopped
volumes:
  - ./src/main/certs:/usr/local/tomcat/certs
  - ./src/main/server.xml:/usr/local/tomcat/conf/server.xml
  - ./target/calculadora.war:/usr/local/tomcat/webapps/calculadora.war
```

Error: me decía que le faltaba el archivo login

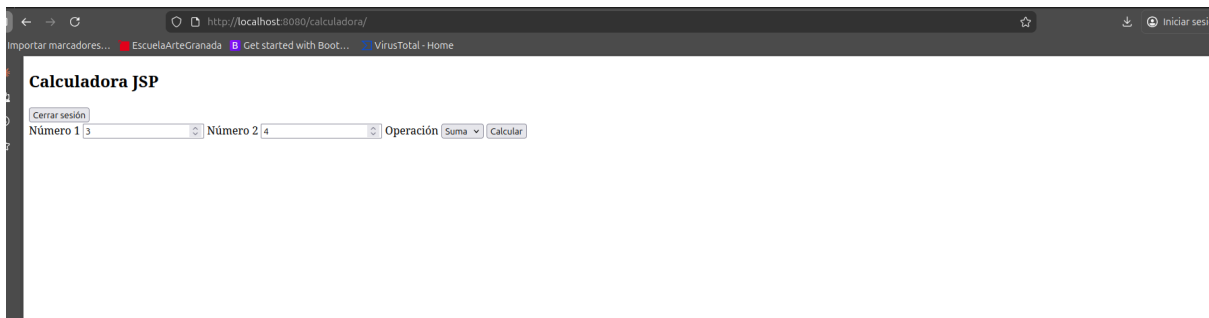
lo e copiapo del siguiente punto y solucionado.

Paso 9:

Añade roles a tu app con el archivo tomcat-users.xml y comprobar su correcto funcionamiento, tanto de los usuarios con sus contraseñas como con las restricciones impuestas a los roles.



```
1 version: "3.8"
2 services:
3   tomcat:
4     image: tomcat:10.1
5     container_name: tomcat_t3
6     ports:
7       - "8080:8080"
8       - "8443:8443"
9     restart: unless-stopped
10    volumes:
11      - ./src/main/certs:/usr/local/tomcat/certs
12      - ./src/main/server.xml:/usr/local/tomcat/conf/server.xml
13      - ./target/calculadora.war:/usr/local/tomcat/webapps/calculadora.war
14      - ./src/main/tomcat-users.xml:/usr/local/tomcat/conf/tomcat-users.xml
```



Paso Extra:

Buenas practicas

Normalización de Versiones: Mantener la recomendación docente de usar una versión fija de Java (Java 17) garantiza que el artefacto `.war` se comporte igual en cualquier entorno (Principio de Reproducibilidad).

Propiedades Centralizadas: Utilizar la sección `<properties>` del `pom.xml` para gestionar las versiones de todas las dependencias en un solo lugar, facilitando futuras actualizaciones.

Separación de Responsabilidades: Mantener la lógica de negocio en el archivo Java (Servlet) y la presentación en los JSPs sigue el patrón de diseño MVC (Modelo-Vista-Controlador).

+4

Nombres Descriptivos: Evitar nombres genéricos como `ejemploJose.java` y usar nombres que describan la función, como `CalculadoraServlet.java`.