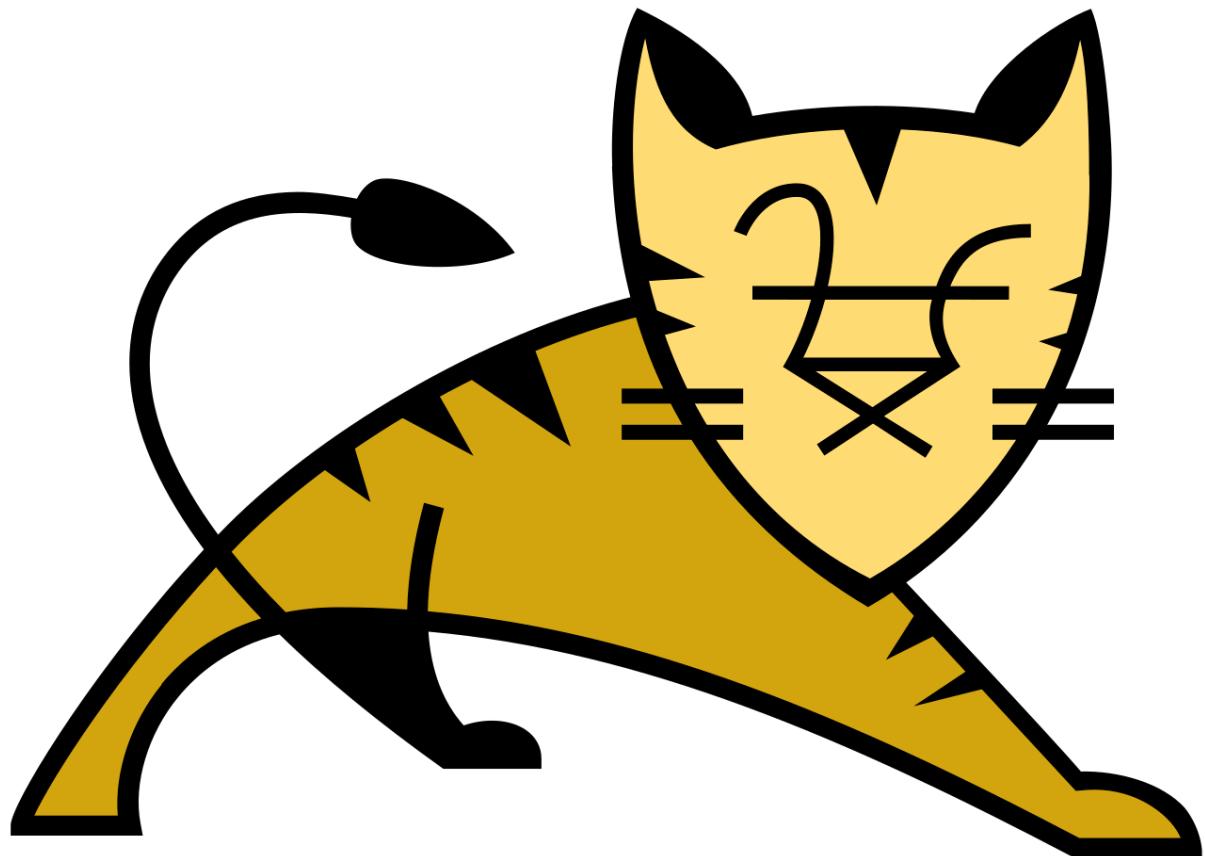


# Servidor de Aplicaciones

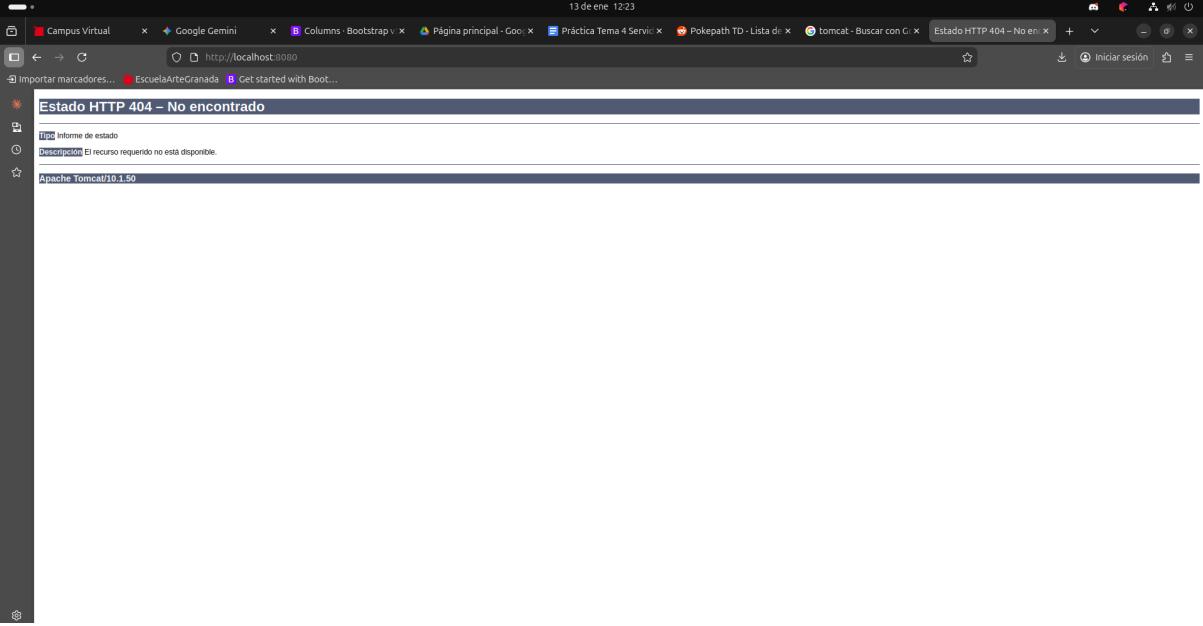


Javier Granizo Aguilar

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

# Paso 1:

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

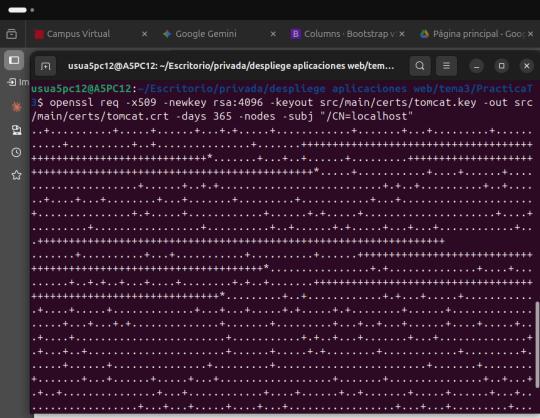


The screenshot shows a browser window with multiple tabs open. The active tab is titled "Estado HTTP 404 – No encontrado" and displays the message "El recurso requerido no está disponible." Below the message, it says "Apache Tomcat/10.1.50".

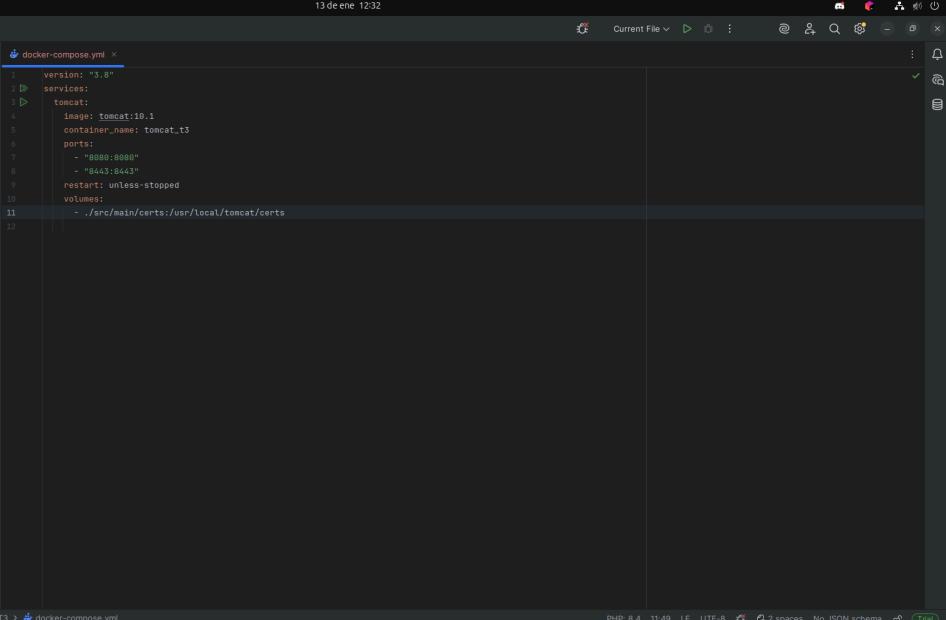
```
usu5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3$ sudo docker compose up -d
WARN[0000] /home/usua5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 2/2
  ✓ Network practicat3_default Created
  ✓ Container tomcat_t3 Started
usu5pc12@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
  ✓ Network practicat3_default Removed
usu5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$
```

## Paso 2:

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



```
usu5pc12@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
```

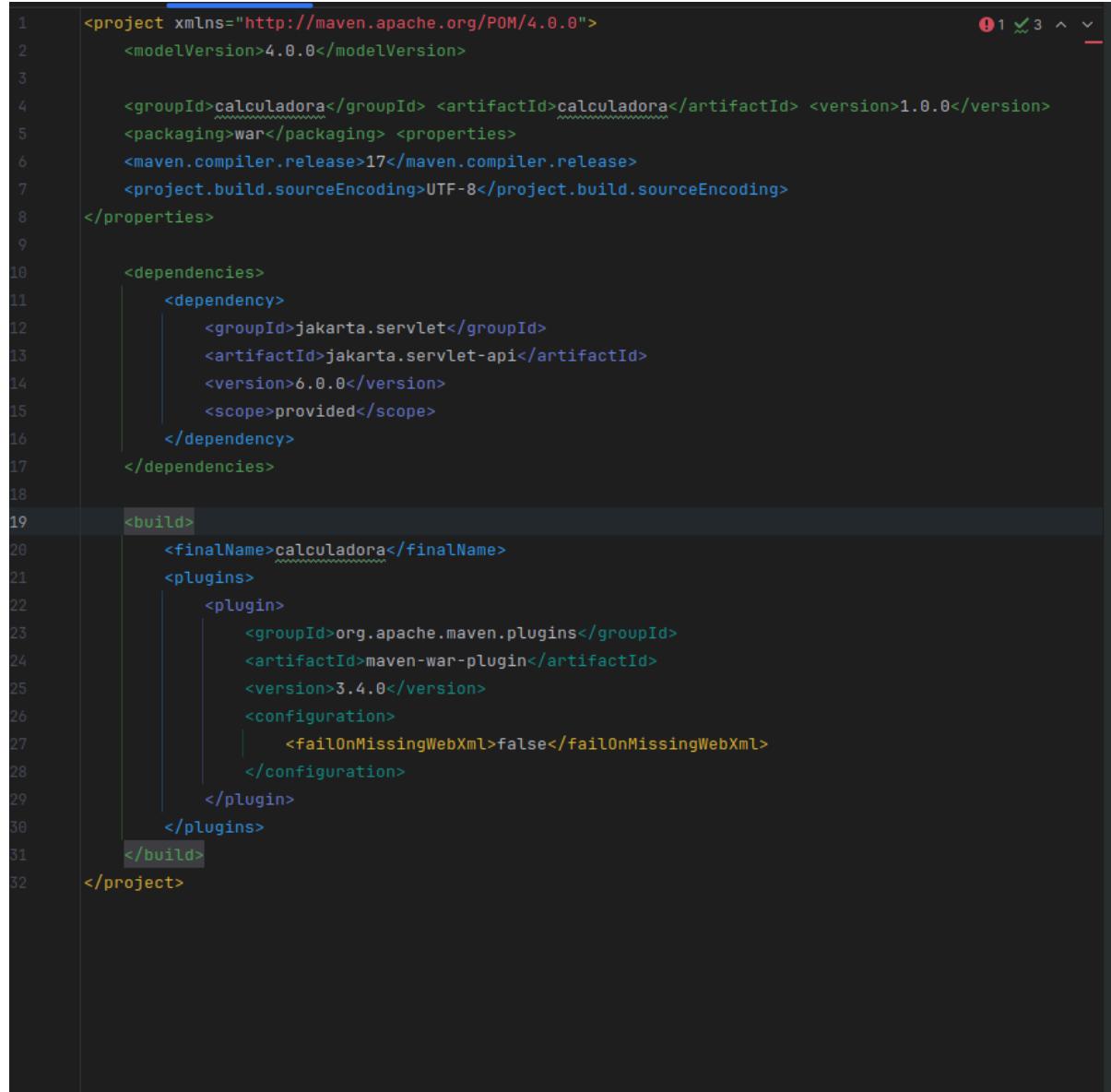
## Paso 3:

Instala Maven y comprueba que lo has hecho correctamente.

```
usu5pc12@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"
usu5pc12@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.



The screenshot shows a code editor with a dark theme displaying a Maven Project Object Model (POM) XML file. The file is structured as follows:

```
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>
```

The code editor interface includes a status bar at the top right with icons for file operations like save, undo, redo, and zoom.

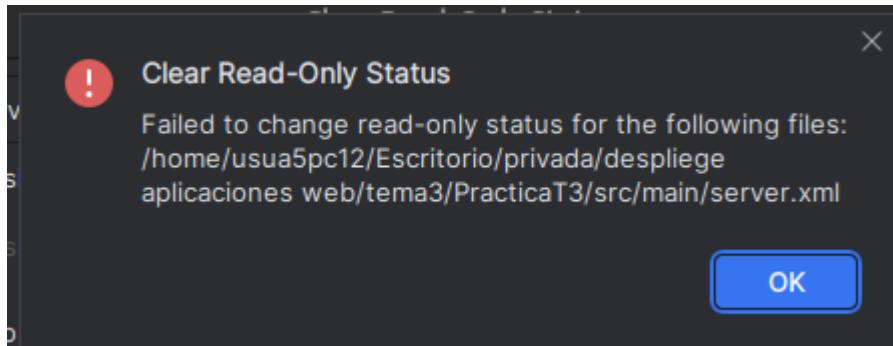
## Paso 5:

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

```
usu5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3$ sudo docker cp tomcat_t3:/usr/local/tomcat/conf/server.xml /home/usu5pc12/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3/src/main/server.xml
Successfully copied 8.7kB to /home/usu5pc12/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.

The screenshot shows the Eclipse IDE interface. On the left, the Project Explorer displays a project named 'PracticaT3' with several subfolders and files. The file 'calculadoraServlet.java' is selected and shown in the code editor on the right. The code implements a HttpServlet to handle POST requests and perform arithmetic operations based on user roles ('basic' or 'full').

```
PracticaT3
  src/main/java/calculadora/CalculadoraServlet.java
  webapp/index.jsp
  docker-compose.yml
  pom.xml
  temario3
  2DAW Practica T3.pdf

calculadoraServlet.java
  public class CalculadoraServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException {
      if (n1 == null || n2 == null || op == null) {
        return;
      }
      // Convertimos los valores de texto a números (double).
      double a, b;
      try {
        a = Double.parseDouble(n1);
        b = Double.parseDouble(n2);
      } catch (Exception e) {
        out.print("\\"ok\\":false,\"message\":\"Números no válidos.\")");
        return;
      }
      //Comprobamos los roles del usuario:
      boolean basica = request.isUserInRole("calc_basic");
      boolean full   = request.isUserInRole("calc_full");
      /*
       * Reglas de autorización:
       * - Los usuarios "full" pueden hacer cualquier operación
       * - Los usuarios "basic" SOLO pueden hacer "suma"
       */
      if (!(full || (basica && "suma".equals(op)))) {
        out.print("\\"ok\\":false,\"message\":\"Acceso denegado.\")");
        return;
      }
      //Realizamos la operación solicitada.
      double r;
      switch (op) {
        case "suma": r = a + b; break;
        case "resta": r = a - b; break;
        default:
          out.print("\\"ok\\":false,\"message\":\"Operación no válida.\")");
          return;
      }
      /*Devolvemos el resultado en formato JSON.
      El frontend (Javascript) leerá este mensaje y lo mostrará en pantalla.*/
      out.print("\\"ok\\":true,\"message\":\"Resultado: " + r + "\\")";
    }
  }
```

The screenshot shows the Eclipse IDE interface with the code editor open to 'calculadoraServlet.java'. A dialog box from the 'Extension Pack for Java' extension is visible, asking if the user wants to install it. The code in the editor is identical to the one shown in the previous screenshot.

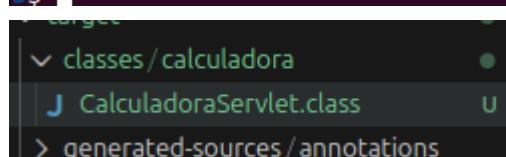
```
calculadoraServlet.java
  public class CalculadoraServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException {
      if (n1 == null || n2 == null || op == null) {
        return;
      }
      // Convertimos los valores de texto a números (double).
      double a, b;
      try {
        a = Double.parseDouble(n1);
        b = Double.parseDouble(n2);
      } catch (Exception e) {
        out.print("\\"ok\\":false,\"message\":\"Números no válidos.\")");
        return;
      }
      //Comprobamos los roles del usuario:
      boolean basica = request.isUserInRole("calc_basic");
      boolean full   = request.isUserInRole("calc_full");
      /*
       * Reglas de autorización:
       * - Los usuarios "full" pueden hacer cualquier operación
       * - Los usuarios "basic" SOLO pueden hacer "suma"
       */
      if (!(full || (basica && "suma".equals(op)))) {
        out.print("\\"ok\\":false,\"message\":\"Acceso denegado.\")");
        return;
      }
      //Realizamos la operación solicitada.
      double r;
      switch (op) {
        case "suma": r = a + b; break;
        case "resta": r = a - b; break;
        default:
          out.print("\\"ok\\":false,\"message\":\"Operación no válida.\")");
          return;
      }
      /*Devolvemos el resultado en formato JSON.
      El frontend (Javascript) leerá este mensaje y lo mostrará en pantalla.*/
      out.print("\\"ok\\":true,\"message\":\"Resultado: " + r + "\\")";
    }
  }
```

## 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/usua5pc12/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/usua5pc12/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] -----
```

usu5pc12@A5PC12:~/Escritorio/privada/despliegue aplicaciones web/tema3/PracticaT3\$



añadir

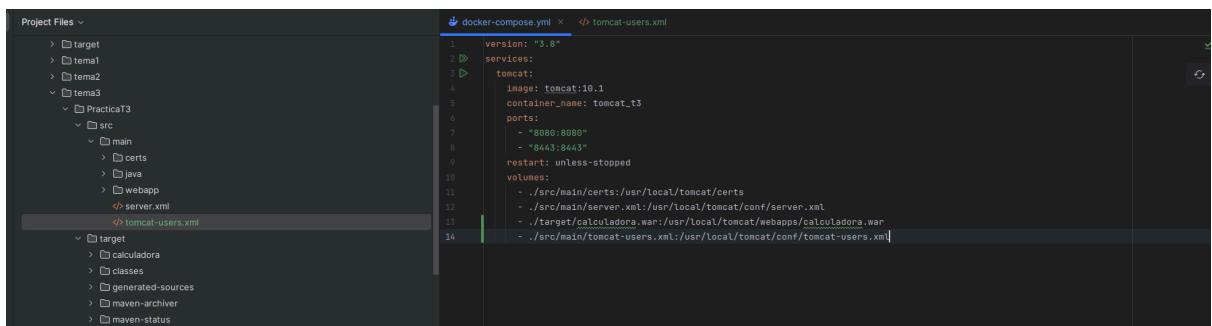
```
container_name: tomcat_t3
ports:
  - "8080:8080"
  - "8443:8443"
restart: unless-stopped
volumes:
  - ./src/main/certs:/usr/local/tomcat/certs
  - ./src/main/server.xml:/usr/local/tomcat/conf/server.xml
  - ./target/calculadora.war:/usr/local/tomcat/webapps/calculadora.war
```

## 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.



The screenshot shows a code editor with two files open. On the left is the project tree, which includes 'target', 'tema1', 'tema2', 'tema3' (which contains 'PracticaT3' and 'src' with subfolders 'main', 'certs', 'java', 'webapp', and 'server.xml'), and 'tomcat-users.xml'. On the right is the 'docker-compose.yml' file, which defines a service named 'tomcat' using the 'tomcat:10.1' image. It maps ports 8880 and 8443, sets 'restart: unless-stopped', and mounts volumes from the local 'src/main/tomcat-users.xml' to the container's '/usr/local/tomcat/conf/tomcat-users.xml' and from 'target/calculadora.war' to '/usr/local/tomcat/webapps/calculadora.war'. The 'tomcat-users.xml' file itself is shown in the code editor.

```
version: "3.8"
services:
  tomcat:
    image: tomcat:10.1
    container_name: tomcat_t3
    ports:
      - "8880:8880"
      - "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
      - ./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.