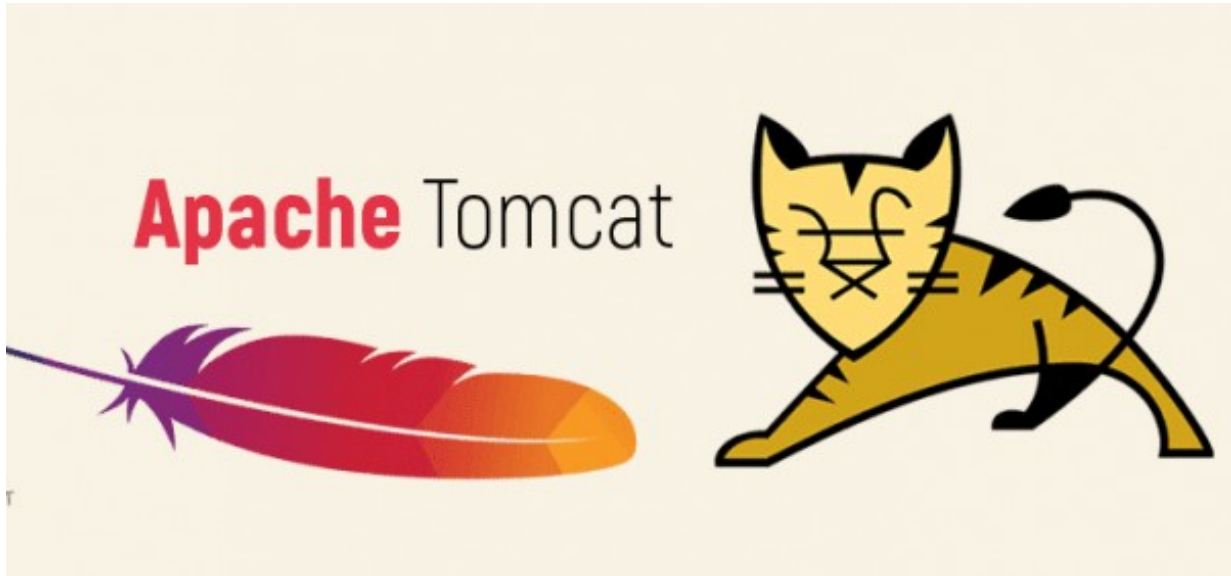


# Administración de Tomcat



Cristian Leandro Sánchez Mego  
2º DAW

## Sumario

1. Instalación de Tomcat en Linux:	3
1.1 Instalación del JDK:	3
1.2 Instalación del Tomcat 7:	4
1.3 Prueba de conexión al servidor:	6
2. Arquitectura y ficheros de configuración:	9
2.1 Arquitectura de Tomcat:	9
2.2 Estructura de directorios:	9
2.3 Ficheros de configuración:	10
3. Despliegue manual de aplicaciones web:	17
3.1 Código HTML:	17
3.2 Código Servlet:	18
3.3 Código JSP:	20
3.4 Código web.xml:	21
3.5 Prueba en servidor local:	22
3.6 Estructura del Proyecto:	29
3.7 Estructura de directorio:	29
3.8 Exportar a .WAR:	29
3.9 Despliegue de la aplicación a través de su carpeta:	30
3.10 Despliegue de aplicaciones en ROOT Context:	35
a) Eliminar el directorio <i>/opt/tomcat/webapps/ROOT</i> :	43
b) Configuración del proyecto para que se despliegue en ROOT context:	44
c) Acceso a la aplicación:	49
4. Uso del Tomcat Web Manager:	50
4.1 Editar archivo tomcat-user.xml:	50
4.2 Comprobación del Tomcat Manager en Windows:	50
4.3 Gestor de aplicaciones Web de Tomcat:	52
a) Despliegue de una aplicación .war:	53
5. Despliegue de CMS:	55
5.1 Descarga de CMS:	55
5.2 Despliegue de CMS:	56
6. Conclusiones:	57

## 1. Instalación de Tomcat en Linux:

### 1.1 Instalación del JDK:

El primer paso a seguir es instalar el JDK, para ello tendremos que incluir el repositorio del mismo para poder utilizarlo. En este caso usaremos el JDK en su versión 8.

```
crístian@crístian-VirtualBoxPC:~$ sudo add-apt-repository ppa:linuxuprising/java
You are about to add the following PPA:
  Oracle Java 11 (LTS) and 15 installer for Ubuntu (20.10, 20.04, 19.10, 18.04, 16.04
  and 14.04), Linux Mint and Debian.

Java binaries are not hosted in this PPA due to licensing. The packages in this PPA
download and install Oracle Java, so a working Internet connection is required.

The packages in this PPA are based on the WebUpd8 Oracle Java PPA packages: https://
launchpad.net/~webupd8team/+archive/ubuntu/java

Created for users of https://www.linuxuprising.com/

Installation instructions (with some tips), feedback, suggestions, bug reports etc.:
```

Paso seguido, instalaremos el JDK con el siguiente comando.

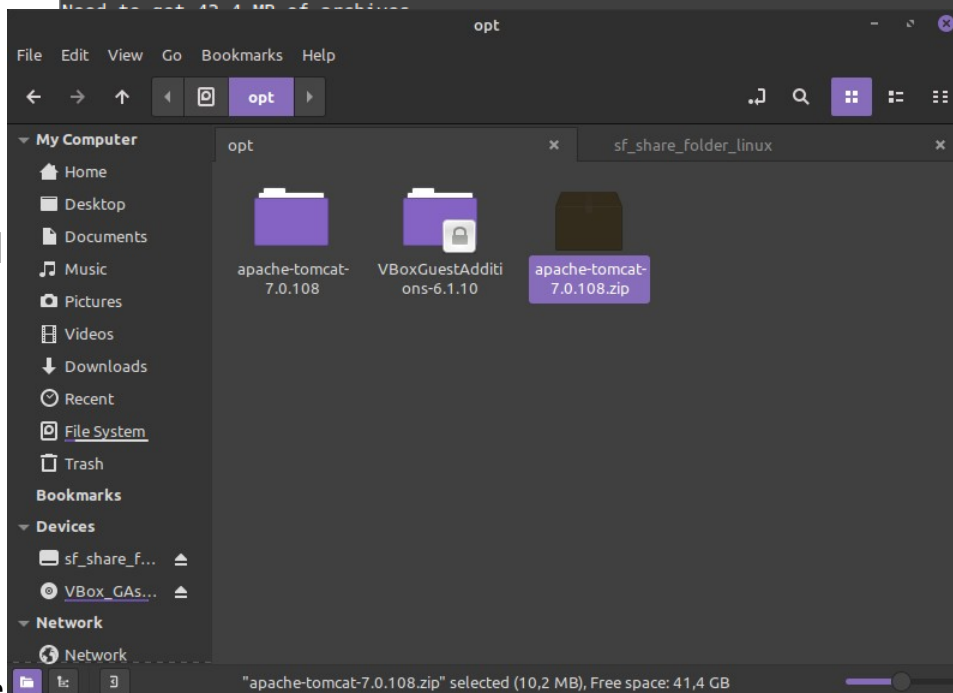
```
cristian@cristian-VirtualBoxPC: ~  
File Edit View Search Terminal Help  
cristian@cristian-VirtualBoxPC:~$ sudo apt-get install openjdk-8-jdk  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  ca-certificates-java fonts-dejavu-extra libatk-wrapper-java  
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev libx11-dev  
  libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-8-jdk-headless  
  openjdk-8-jre openjdk-8-jre-headless x11proto-core-dev x11proto-dev  
  xorg-sgml-doctools xtrans-dev  
Suggested packages:  
  libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-8-demo  
  openjdk-8-source visualvm icedtea-8-plugin fonts-ipafont-gothic  
  fonts-ipafont-mincho fonts-wqy-microhei fonts-wqy-zenhei  
The following NEW packages will be installed:  
  ca-certificates-java fonts-dejavu-extra libatk-wrapper-java  
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev libx11-dev  
  libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-8-jdk  
  openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless x11proto-core-dev  
  x11proto-dev xorg-sgml-doctools xtrans-dev  
0 upgraded, 20 newly installed, 0 to remove and 437 not upgraded.  
Need to get 42.4 MB of archives.  
After this operation, 190 MB of additional disk space will be used.
```

## 1.2 del

Para el que

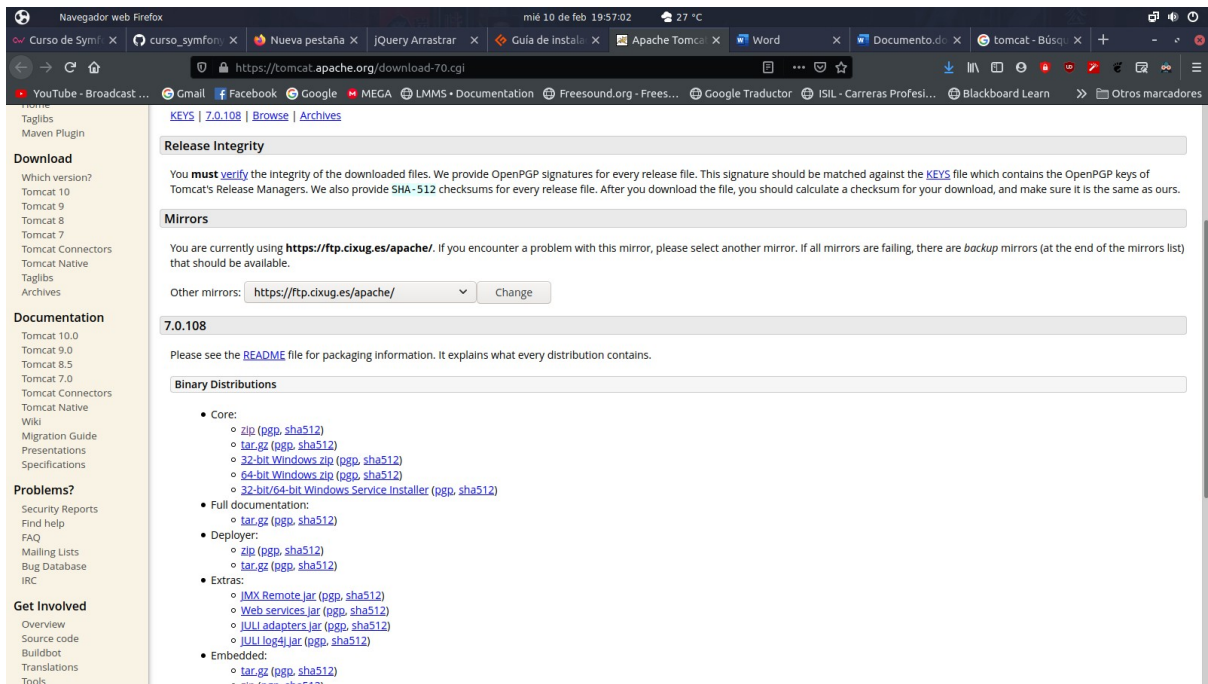
ello que Core

de su página principal.



## Instalación Tomcat 7:

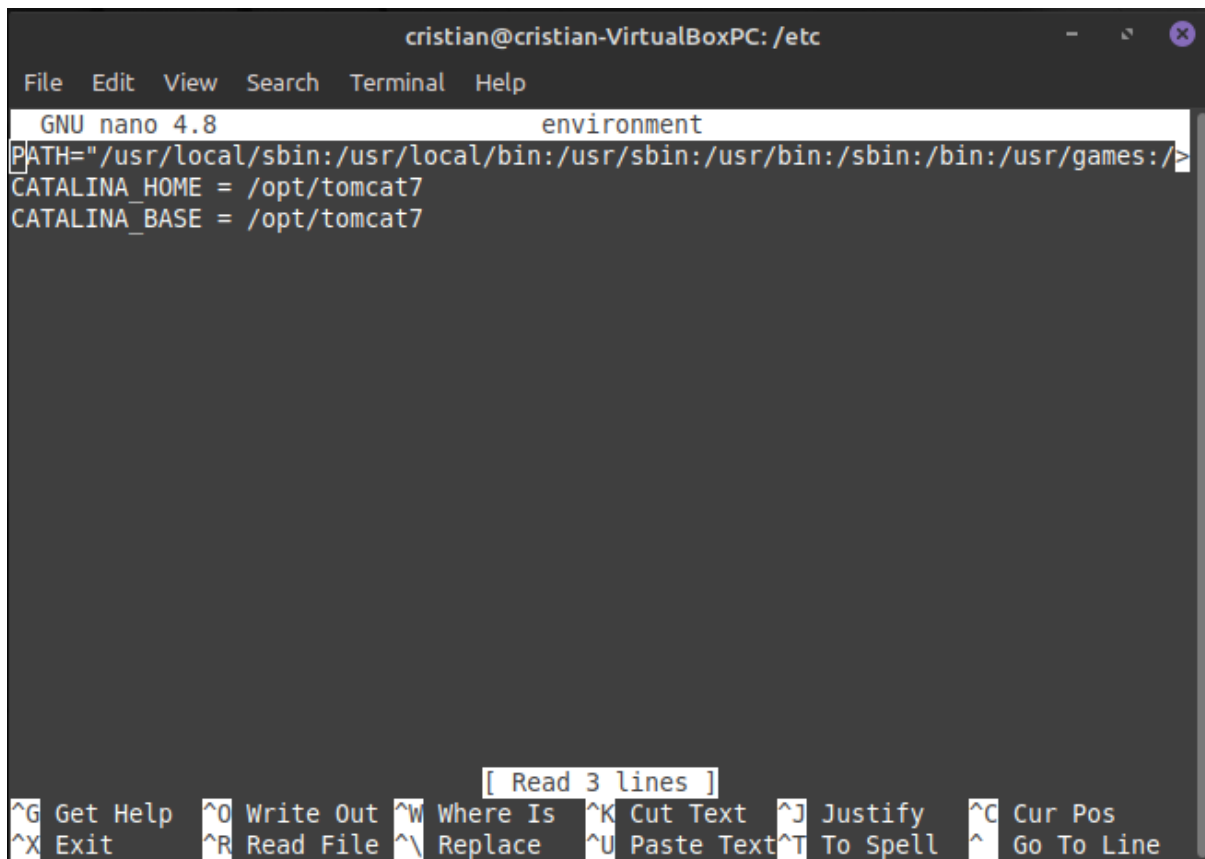
esta ocasión, no seguiremos lo que el libro nos indica, ya que ya no es posible instalar el Tomcat 7 desde sus repositorios. Para ello primero tendremos que descargar la versión en su formato ZIP



Luego de descargar, el archivo lo tendremos que ubicar en la carpeta raíz **/opt/** dónde tendremos que descomprimir la carpeta, dejándonos una carpeta con el Tomcat.

Ya que hemos descomprimido la carpeta en la carpeta raíz **opt**, las variables **CATALINA\_BASE** y **CATALINA\_HOME** apuntarán a **/opt/tomcat7** (habiendo cambiado previamente el nombre de la carpeta original que se obtiene de la descompresión) y se suele usar **CATALINA\_HOME**.

Ahora procederemos al hacer referencia a las variables del Tomcat, en el fichero **/etc/enviroment**. Ya que las variables apunta a la misma carpeta, las mencionaremos igual:



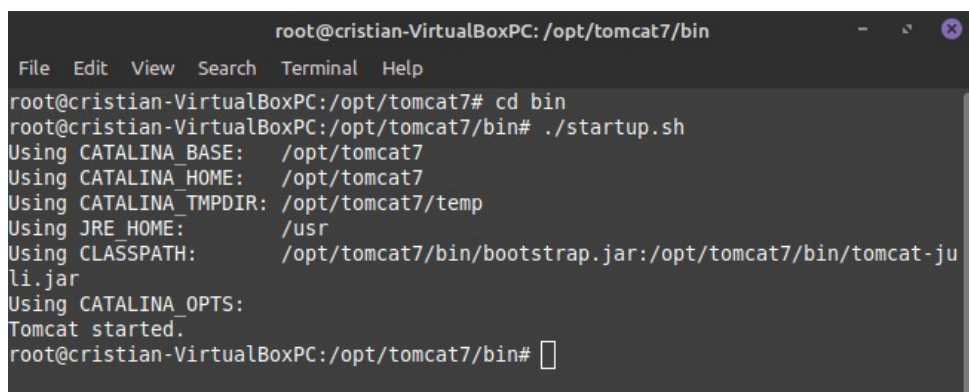
```
cristian@cristian-VirtualBoxPC: /etc
File Edit View Search Terminal Help
GNU nano 4.8 environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/>
CATALINA_HOME = /opt/tomcat7
CATALINA_BASE = /opt/tomcat7

[ Read 3 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line
```

Luego de esto procederemos a reiniciar la sesión para que las variables tomen valor.

### 1.3 Prueba de conexión al servidor:

El primer paso es arrancar el Tomcat dentro del directorio **tomcat/bin/**, el cuál se llama *startup.sh*:

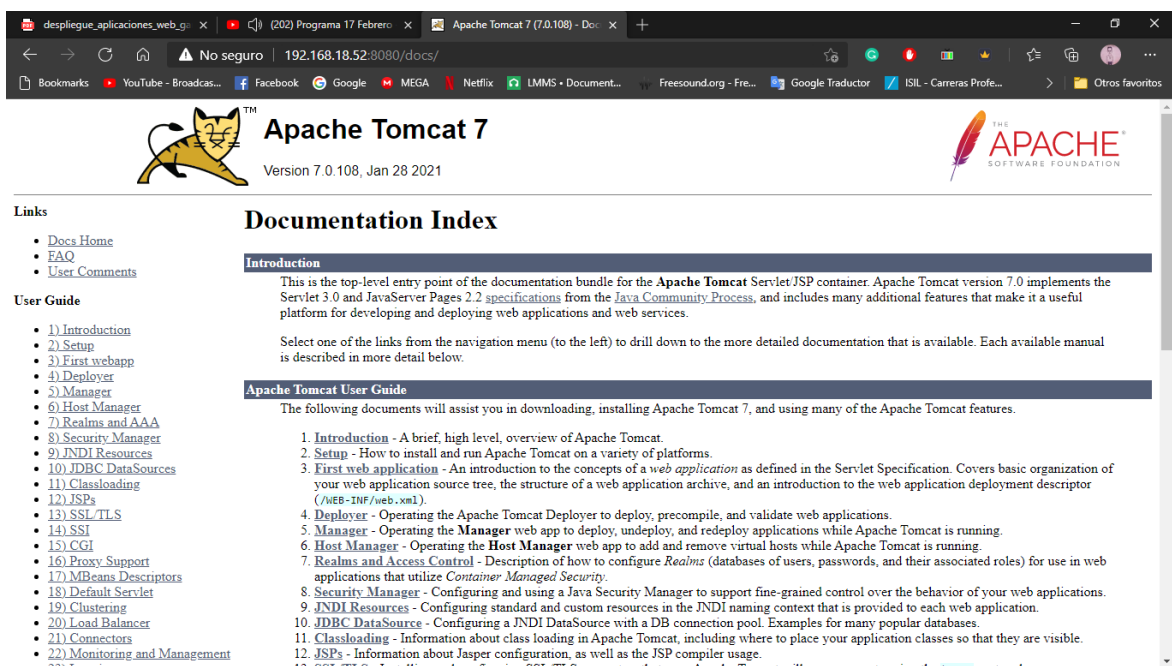
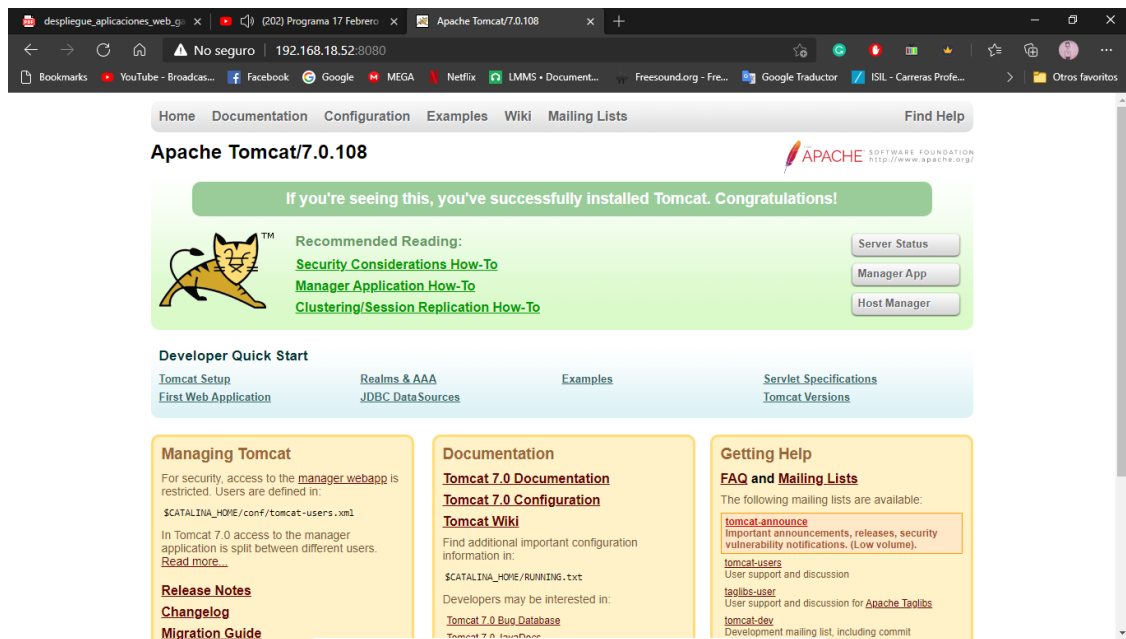


```
root@cristian-VirtualBoxPC: /opt/tomcat7/bin
File Edit View Search Terminal Help
root@cristian-VirtualBoxPC:/opt/tomcat7# cd bin
root@cristian-VirtualBoxPC:/opt/tomcat7/bin# ./startup.sh
Using CATALINA_BASE:   /opt/tomcat7
Using CATALINA_HOME:   /opt/tomcat7
Using CATALINA_TMPDIR: /opt/tomcat7/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/tomcat7/bin/bootstrap.jar:/opt/tomcat7/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
root@cristian-VirtualBoxPC:/opt/tomcat7/bin#
```

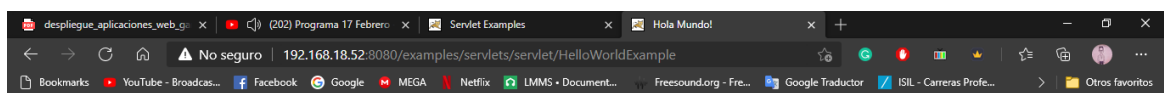
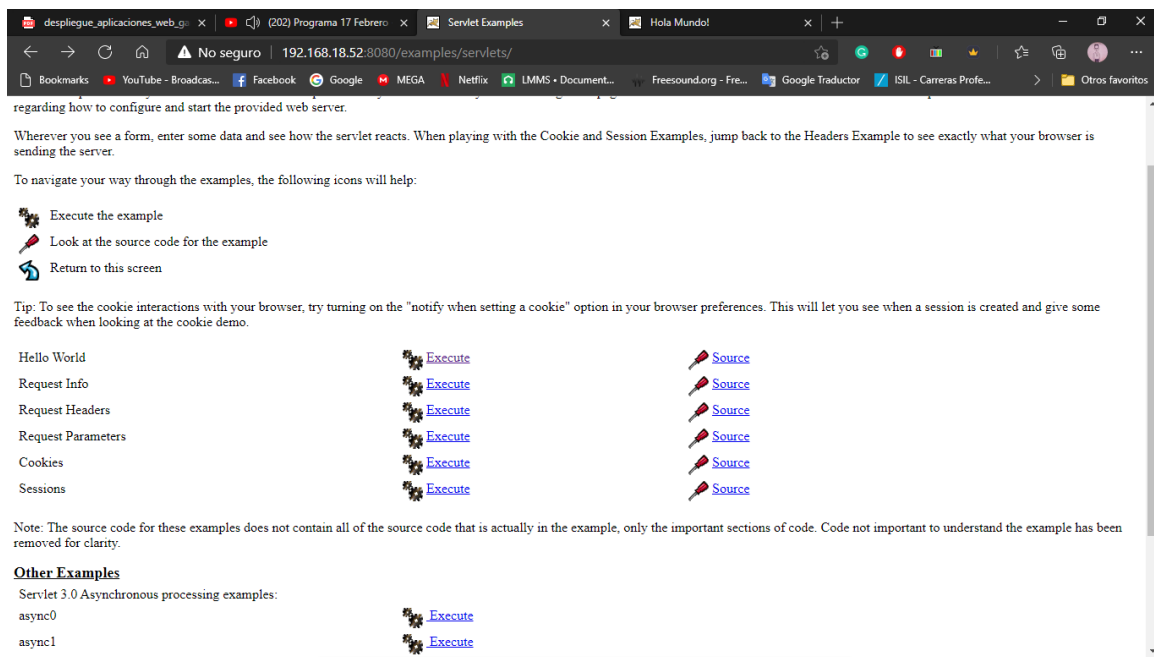
Luego, antes de intentar acceder a través de Windows, debemos desactivar el cortafuegos de Linux:

```
root@cristian-VirtualBoxPC: /opt/tomcat7/bin
File Edit View Search Terminal Help
root@cristian-VirtualBoxPC:/opt/tomcat7/bin# sudo ufw disable
Firewall stopped and disabled on system startup
root@cristian-VirtualBoxPC:/opt/tomcat7/bin#
```

Ya con esto, tendremos visible el Tomcat en Windows:



Desplegamos los ejemplos que ya están instalados en el servidor:

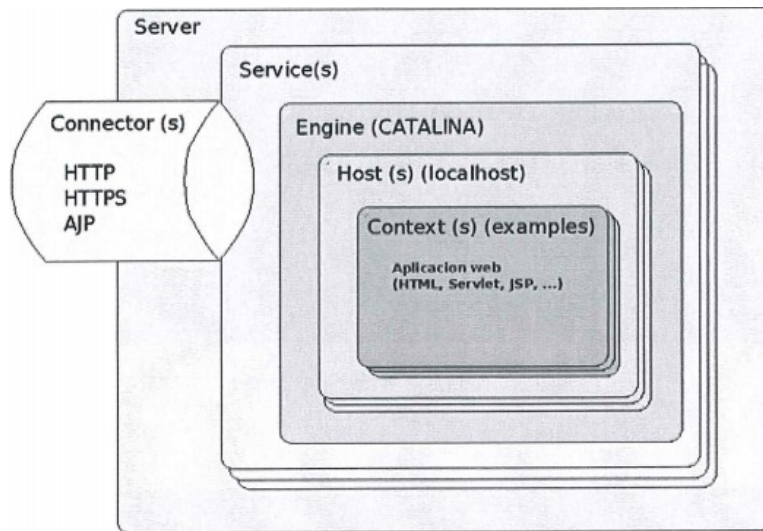




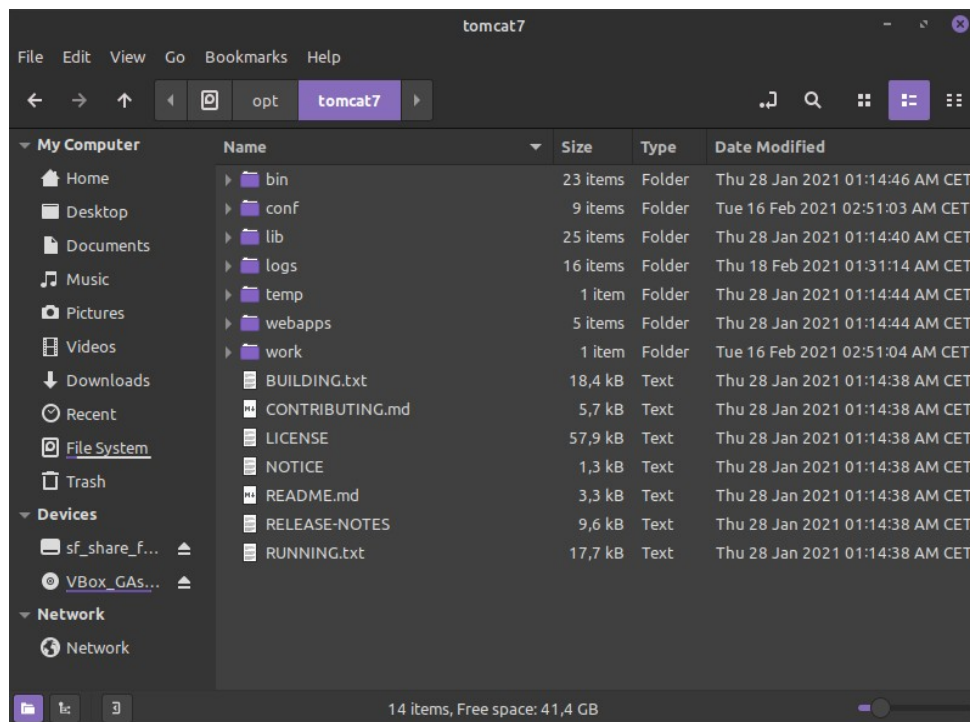
## 2. Arquitectura y ficheros de configuración

En este punto analizaremos la arquitectura y los ficheros de configuración del servidor Tomcat instalado en Linux.

### 2.1 Arquitectura de Tomcat



### 2.2 Estructura de directorios



- **bin:** Guarda los ficheros de encendido y apagado del Tomcat.
- **conf:** Contiene todos los archivos de configuración: los usuarios que tienen acceso al Tomcat, los puertos donde se aloja el servidor, lanzamiento de las aplicaciones, etc.
- **lib:** Contiene las librerías para que Tomcat se conecte con otros servicios, cómo puede ser el *MySQL* o servicios de Java.
- **logs:** Contiene las sesiones de cada aplicación y las veces que se interactúa con el servidor, la IP de quién lo hace y otros datos almacenados en los servidores.
- **temp:** Los temporales que se guardan de cada sesión y las configuraciones previas del despliegue.
- **webapps:** Guarda todas las aplicaciones que se despliegan en el servidor, también almacena los archivos **.WAR** para luego convertirlos en proyectos y poderlos desplegar en el servidor Tomcat.

Recordemos que todo el directorio de Tomcat se encuentra en la misma carpeta **opt**, ya que lo hemos descargado de manera directa y lo hemos descomprimido en la carpeta mencionada.

## 2.3 Ficheros de configuración

En el directorio **opt/tomcat7/conf/** seleccionamos el archivo **server.xml** y verificamos su contenido.

<Server>

```

-->
<Server port="8005" shutdown="SHUTDOWN">
  <Listener className="org.apache.catalina.startup.VersionLoggerListener" />
  <!-- Security listener. Documentation at /docs/config/listeners.html
  <Listener className="org.apache.catalina.security.SecurityListener" />
  -->
  <!--APR library loader. Documentation at /docs/apr.html -->
  <Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />
  <!--Initialize Jasper prior to webapps are loaded. Documentation at /docs/jasper-howto.html -->
  <Listener className="org.apache.catalina.core.JasperListener" />
  <!-- Prevent memory leaks due to use of particular java/javax APIs-->
  <Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener" />
  <Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener" />
  <Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener" />

  <!-- Global JNDI resources
       Documentation at /docs/jndi-resources-howto.html
  -->
  <GlobalNamingResources>
    <!-- Editable user database that can also be used by
         UserDatabaseRealm to authenticate users
    -->
    <Resource name="UserDatabase" auth="Container"
              type="org.apache.catalina.UserDatabase"
              description="User database that can be updated and saved"
              factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
              pathname="conf/tomcat-users.xml" />
  </GlobalNamingResources>

```

<Service> y <Connectors>









```

server.xml (/opt/tomcat7/conf)
File Edit View Search Tools Documents Help
server.xml x
so you may not define subcomponents such as "Valves" at this level.
Documentation at /docs/config/service.html
-->
<Service name="Catalina">

  <!--The connectors can use a shared executor, you can define one or more named thread pools-->
  <!--
  <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
    maxThreads="150" minSpareThreads="4"/>
  -->

  <!-- A "Connector" represents an endpoint by which requests are received
  and responses are returned. Documentation at :
  Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
  Java AJP Connector: /docs/config/ajp.html
  APR (HTTP/AJP) Connector: /docs/apr.html
  Define a non-SSL HTTP/1.1 Connector on port 8080
  -->
  <Connector port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />
  <!-- A "Connector" using the shared thread pool-->
  <!--
  <Connector executor="tomcatThreadPool"
    port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />
  -->
  <!-- Define an SSL HTTP/1.1 Connector on port 8443
  This connector uses the BIO implementation that requires the JSSE
  style configuration. When using the APR/native implementation, the
  -->
XML Spaces: 4 Ln 1, Col 1 INS

```

## <Engine>

```

<Engine name="Catalina" defaultHost="localhost">

  <!--For clustering, please take a look at documentation at:
  /docs/cluster-howto.html (simple how to)
  /docs/config/cluster.html (reference documentation) -->
  <!--
  <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>
  -->

  <!-- Use the LockOutRealm to prevent attempts to guess user passwords
  via a brute-force attack -->
  <Realm className="org.apache.catalina.realm.LockOutRealm">
    <!-- This Realm uses the UserDatabase configured in the global JNDI
    resources under the key "UserDatabase". Any edits
    that are performed against this UserDatabase are immediately
    available for use by the Realm. -->
    <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
      resourceName="UserDatabase"/>
  </Realm>

```

## <Host>

```

<Host name="localhost" appBase="webapps"
  unpackWARs="true" autoDeploy="true">

  <!-- SingleSignOn valve, share authentication between web applications
  Documentation at: /docs/config/valve.html -->
  <!--
  <Valve className="org.apache.catalina.authenticator.SingleSignOn" />
  -->

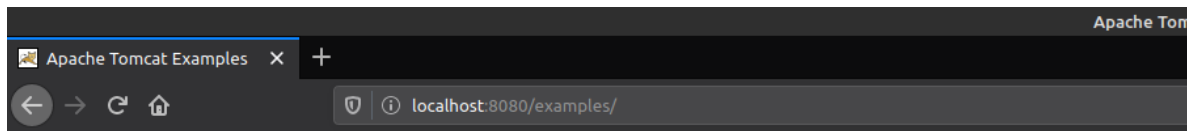
  <!-- Access log processes all example.
  Documentation at: /docs/config/valve.html
  Note: The pattern used is equivalent to using pattern="common" -->
  <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
    prefix="localhost_access_log." suffix=".txt"
    pattern="%h %l %u %t &quot;%r&quot; %s %b" />

</Host>

```

Y ahora procedemos a desplegar los ejemplos en Linux:





### Apache Tomcat Examples

- [Servlets examples](#)
- [JSP Examples](#)
- [WebSocket \(JSR356\) Examples](#)
- [WebSocket Examples using the deprecated Apache Tomcat proprietary API](#)

## 3. Despliegue manual de aplicaciones web

En este punto, procederemos a crear una aplicación en la máquina virtual de Windows, compuesta por un *Servlet* y un *JSP*. La desplegaremos manualmente en la máquina servidor de Linux.

Utilizaremos el último entorno de trabajo que usamos para el trabajo anterior. En este caso llamaré a esta nueva aplicación dinámica **NuevoEjemplo**. Crearemos los siguientes componentes:

- **index.html**
- **Servlet: HolaServlet.java**
- **JSP: buenas.jsp**
- **Descriptor de despliegue: web.xml**

### 3.1 Código HTML

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <meta charset="ISO-8859-1">
```

```
    <title>Ejemplo desde Windows</title>
```

```
</head>
```

```
<body>
```

```
    <h1>Curso de despliegue de aplicaciones web</h1>
```

```

        <ul>
            <li><h2><a href="hola"> Hola (Servlet)</a></h2></li>
            <li><h2><a href="buenas.jsp"> Buenas (Servlet)</a></h2></li>
        </ul>
    </body>
</html>

```

## 3.2 Código Servlet

```

import java.io.*;
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class HolaServlet
 */
@WebServlet("/HolaServlet")
public class HolaServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public HolaServlet() {
        super();
        // TODO Auto-generated constructor stub
    }
}

```

```

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
    // TODO Auto-generated method stub
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();

    try {
        out.println("<html>");
        out.println("<head><title>Hola, Alumno</title></head>");
        out.println("<body>");
        out.println("<h1>Hola, alumno!</h1>");
        out.println("<p>URI: " + request.getRequestURI() + "</p>");
        out.println("<p>Protocolo: " + request.getProtocol() + "</p>");
        out.println("<p>IP del cliente" + request.getRemoteAddr() + "</p>");
        out.println("</body></html>");
    } finally {
        out.close();
    }
}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}

```

```
}
```

```
}
```

### 3.3 Código JSP

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ page import="java.util.Calendar" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html" charset="ISO-8859-1">
<title>Buenas</title>
<style>
    .content{
        width: 50%;
        margin: 2% 25%;
        text-align: center;
        border: 2px black solid;
        border-radius: 10px;
        background-color: orange;
    }
</style>
</head>
<body>
    <div class="content">
        <b>Bienvenido al curso</b>

        <p>
            Hoy es
```

```

        <%=Calendar.getInstance().getTime()%>
    </p>
    <%
        String saludo;
        int hora = Calendar.getInstance().get(Calendar.HOUR_OF_DAY);
        if (hora < 12){
            saludo = "Buenos días";
        }
        else if (hora >= 12 && hora < 21){
            saludo = "Buenas tardes";
        }
        else {
            saludo = "Buenas noches";
        }
    %>

    <p><%=saludo%></p>
</div>
</body>
</html>

```

### 3.4 Código web.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-
app_4_0.xsd" id="WebApp_ID" version="4.0">

    <display-name>NuevoEjemplo</display-name>

    <welcome-file-list>

        <welcome-file>index.html</welcome-file>

    </welcome-file-list>

```

```
<servlet>

    <servlet-name>HolaServlet</servlet-name>

    <servlet-class>HolaServlet</servlet-class>

</servlet>

<servlet-mapping>

    <servlet-name>HolaServlet</servlet-name>

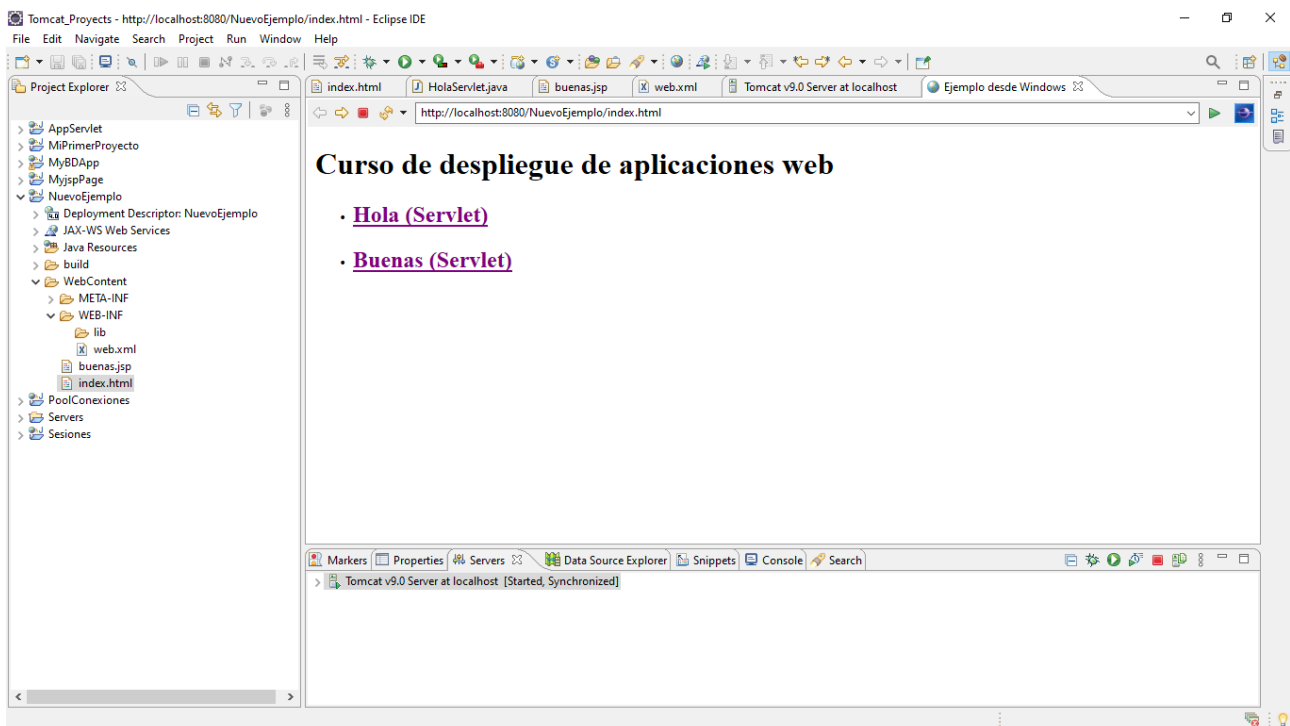
    <url-pattern>/hola</url-pattern>

</servlet-mapping>

</web-app>
```

## 3.5 Prueba en servidor local

Paso a seguir, procederemos a probar en el Tomcat local la pequeña aplicación que hemos creado en la máquina virtual de Windows:



- Servlet:



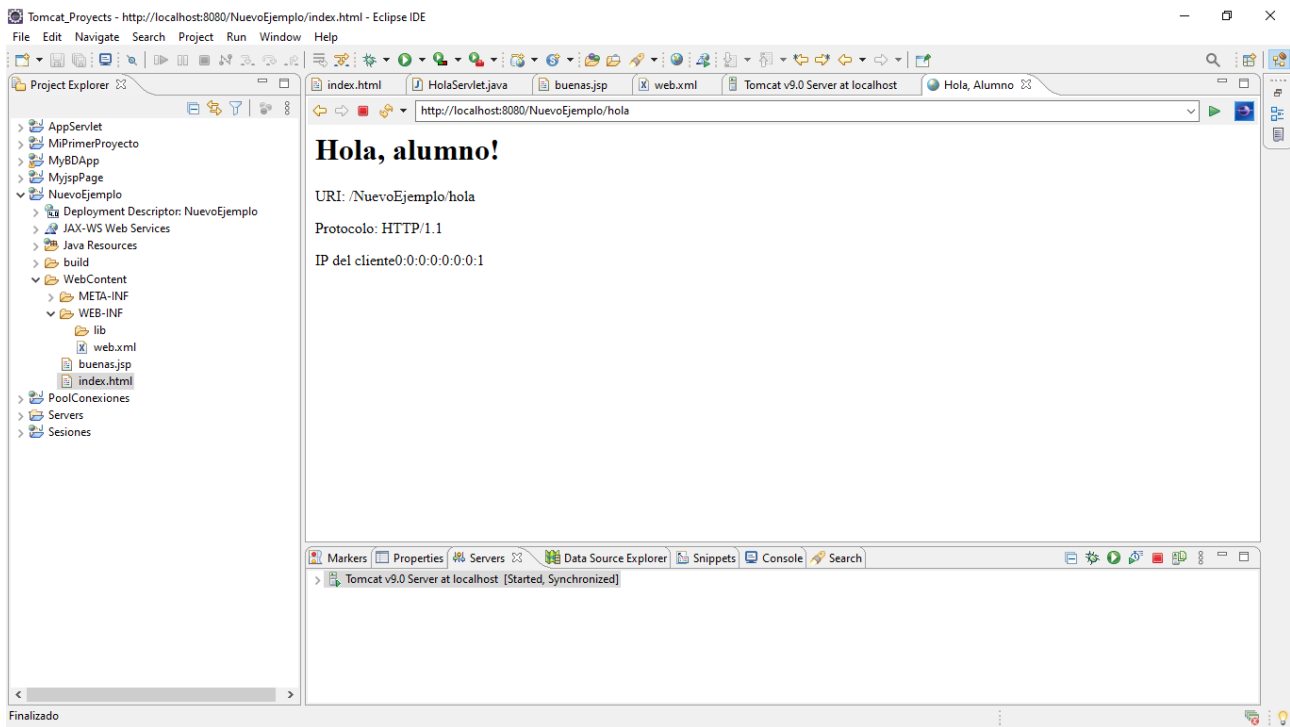




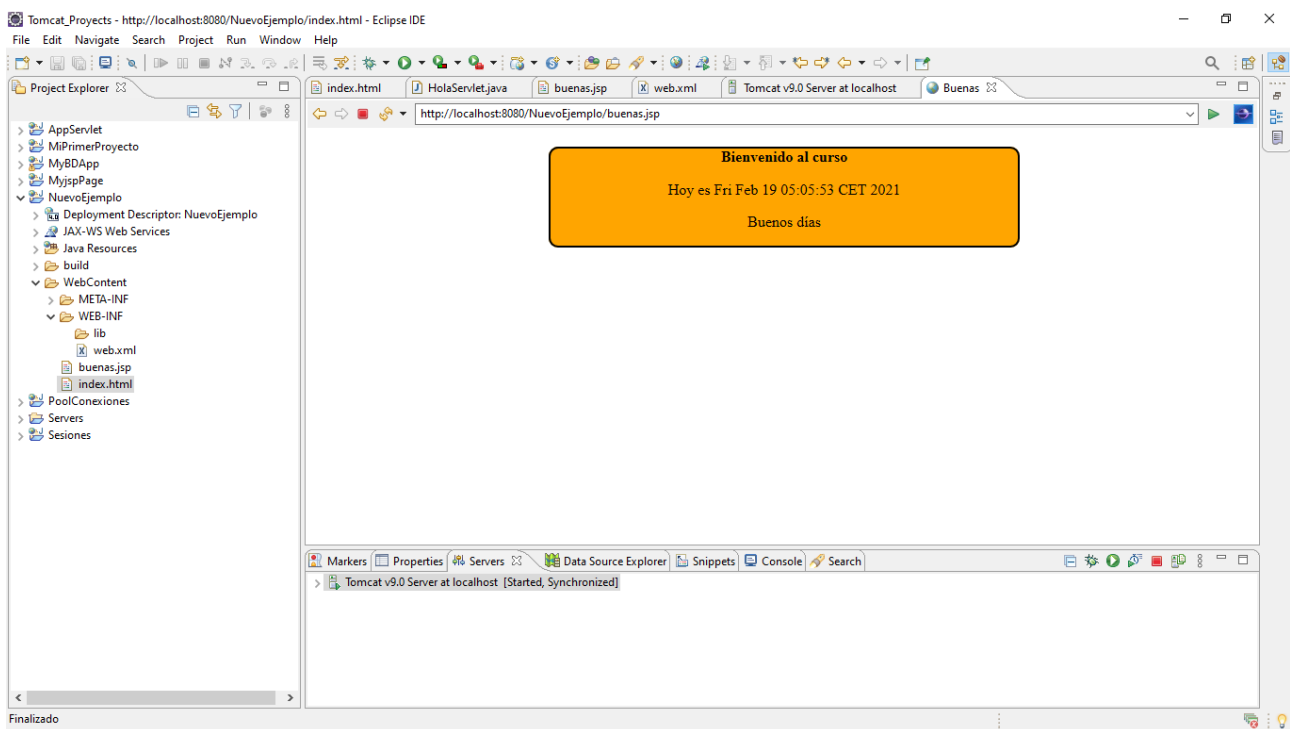




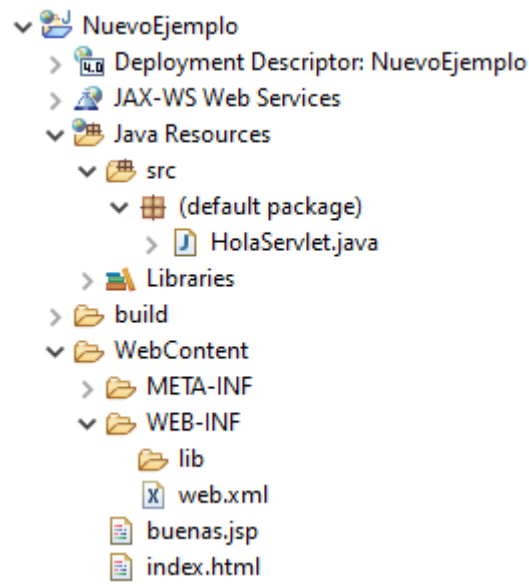




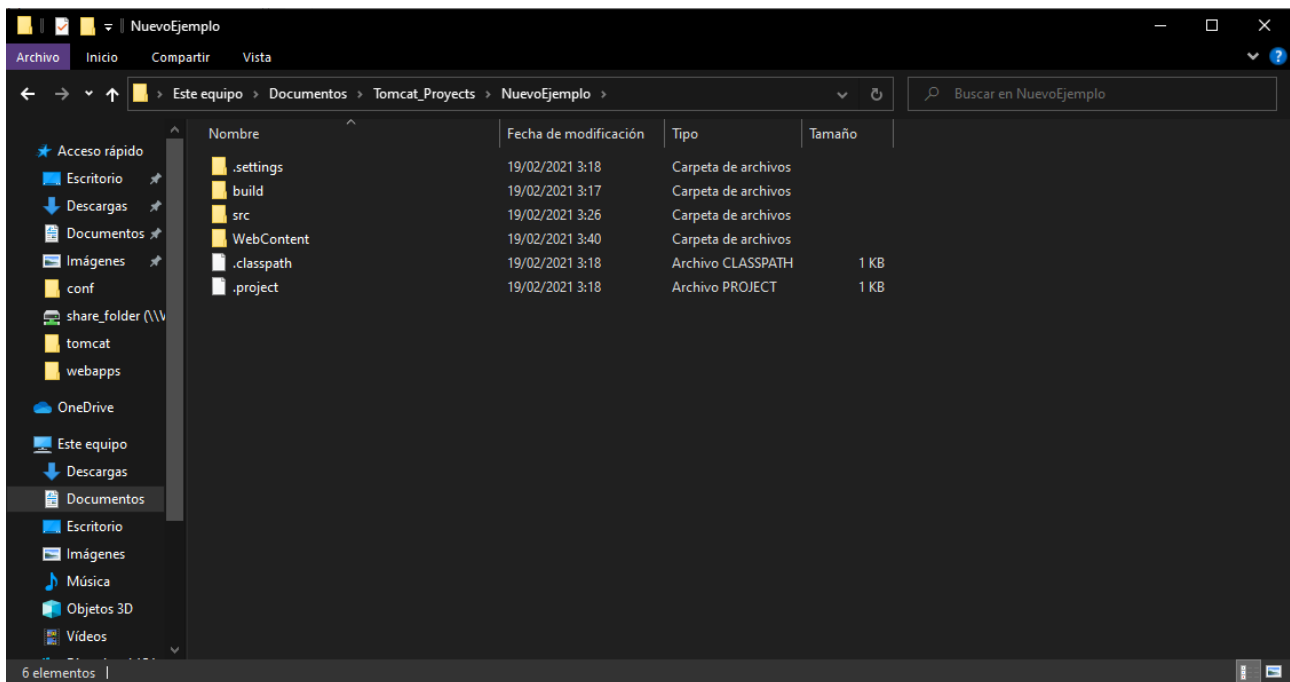
- JSP:



## 3.6 Estructura del Proyecto

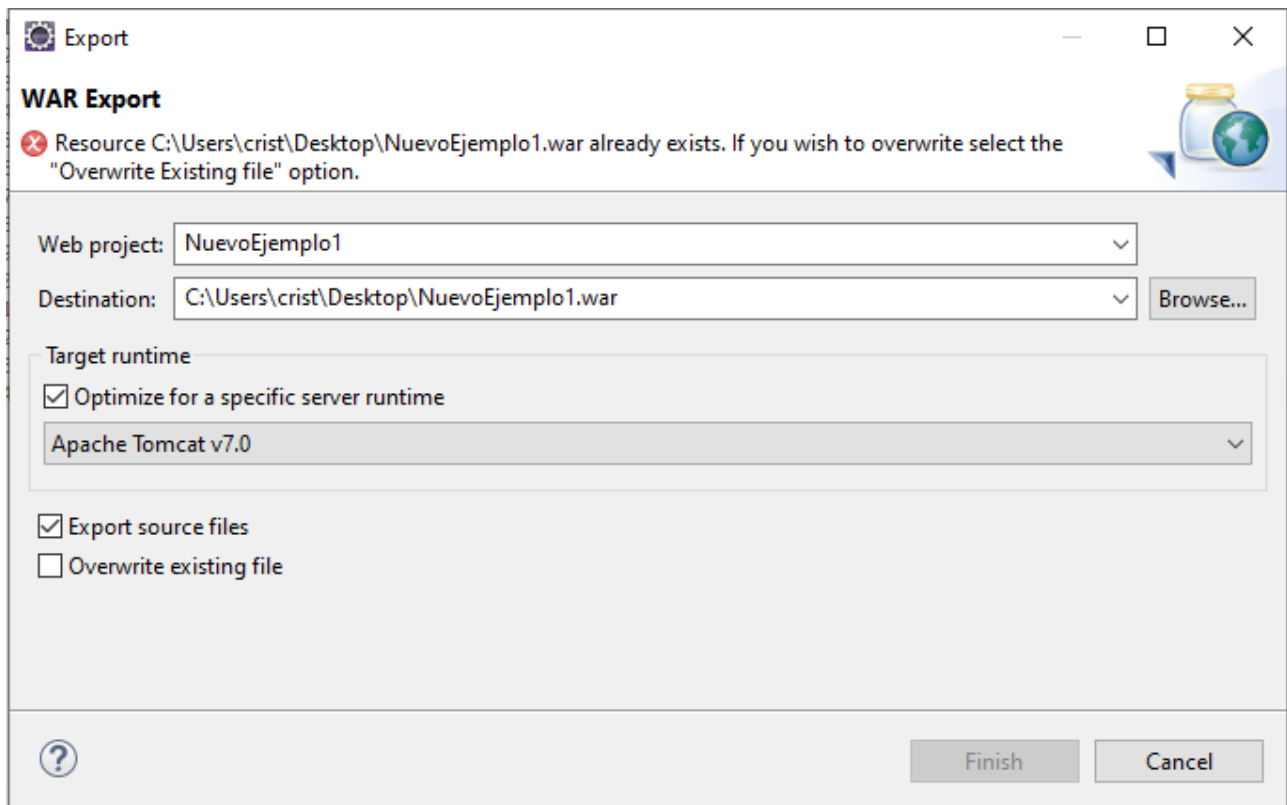


## 3.7 Estructura de directorio



## 3.8 Exportar a .WAR

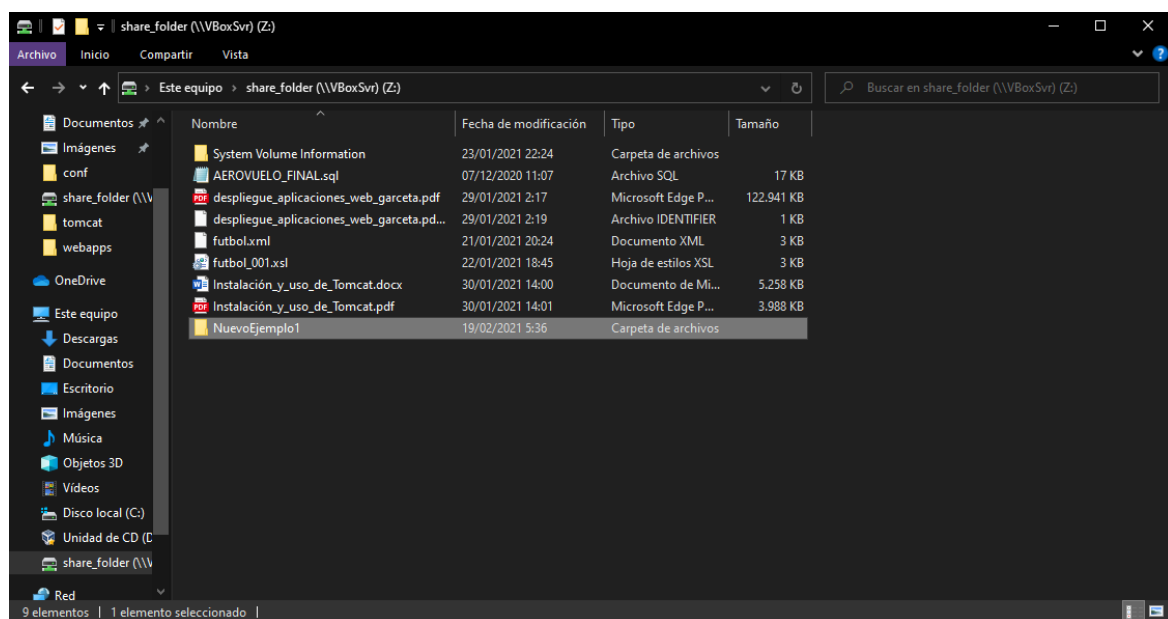
Ahora procederemos a exportar el nuevo proyecto a un fichero **.WAR**, tendremos que adaptarlo para Tomcat 7.0.



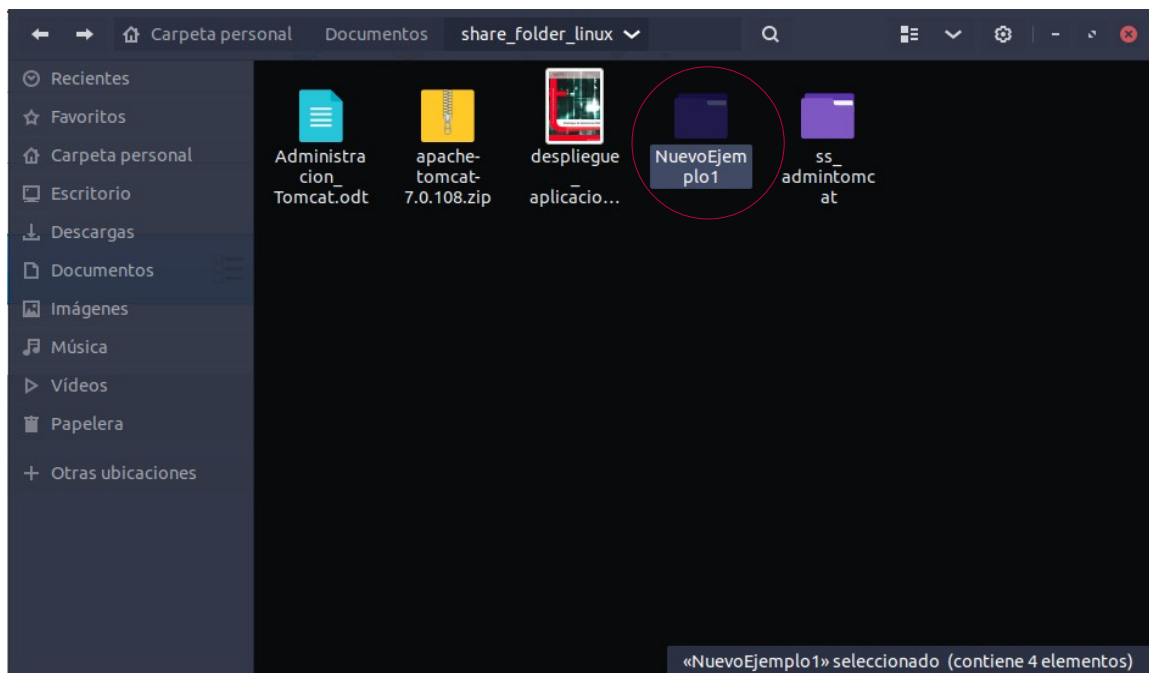
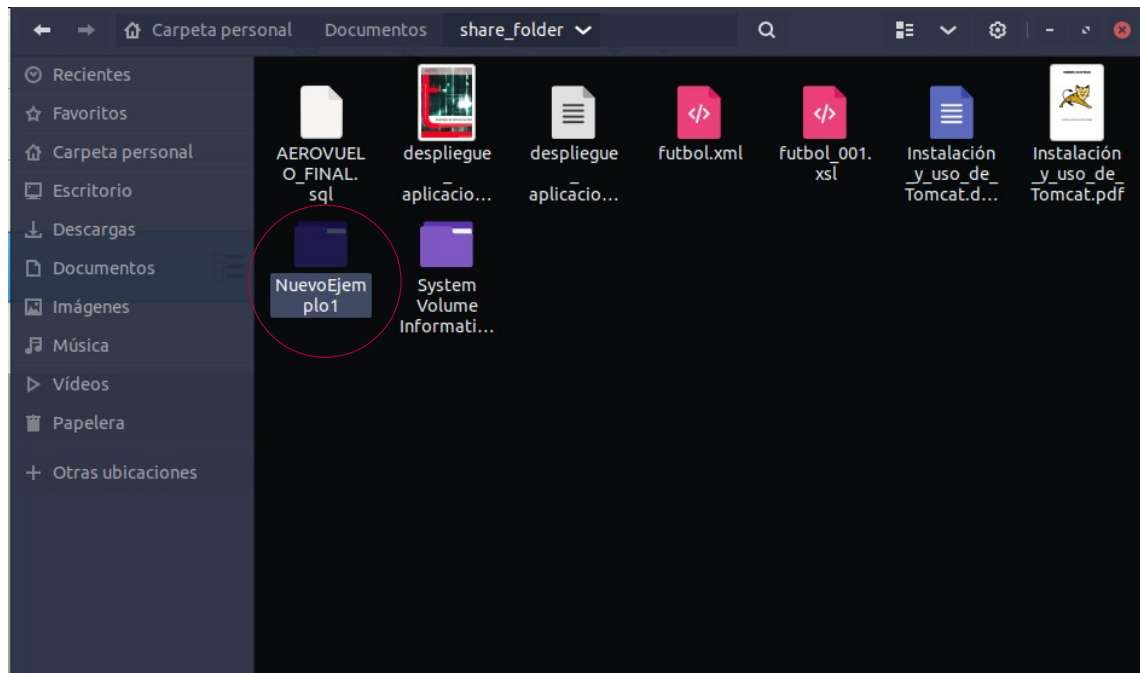
### 3.9 Despliegue de la aplicación a través de su carpeta

En este caso, vamos a usar las carpetas compartidas entre Windows y Ubuntu. Primero copiaremos la carpeta raíz del ejemplo en Windows a la carpeta compartida con la máquina real, luego esta misma carpeta la copiaremos a la carpeta compartida entre Ubuntu y Linux Mint.

- Máquina virtual Windows:



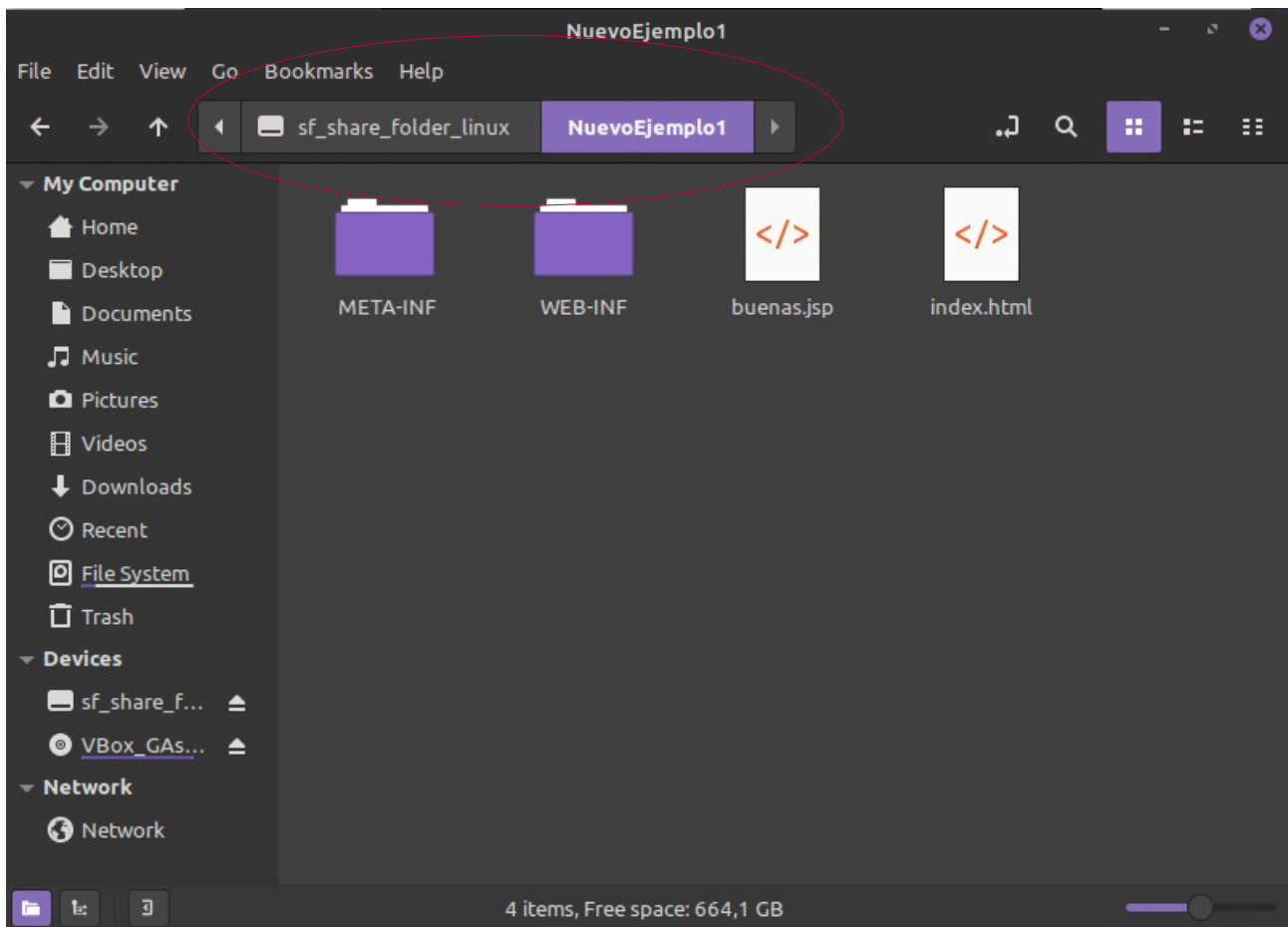
- Máquina real Ubuntu



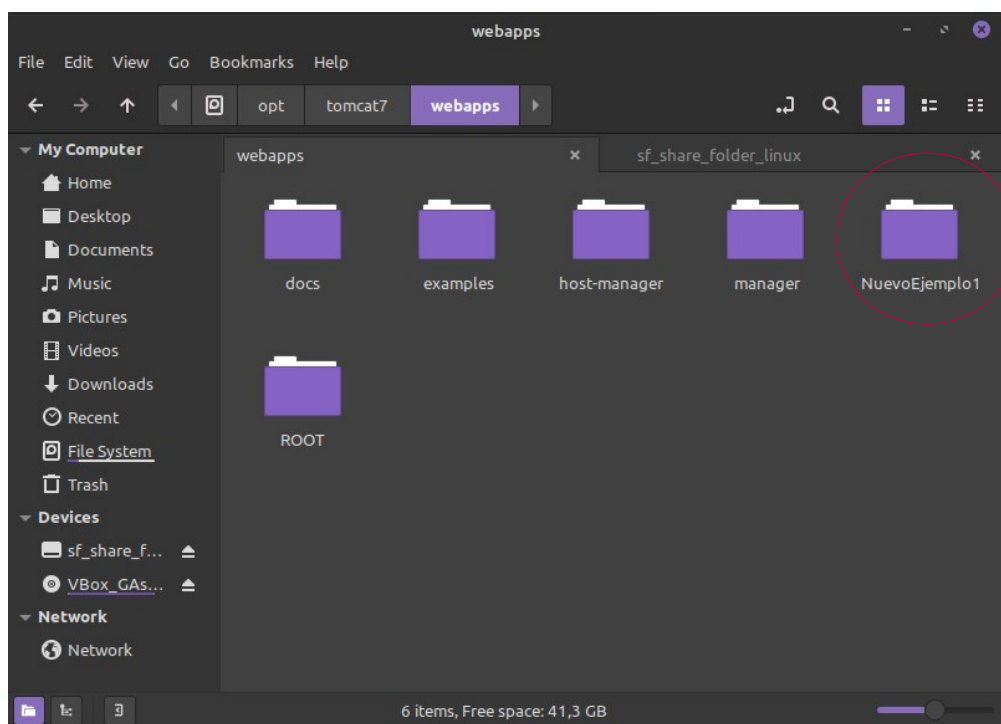
- Máquina virtual Linux Mint



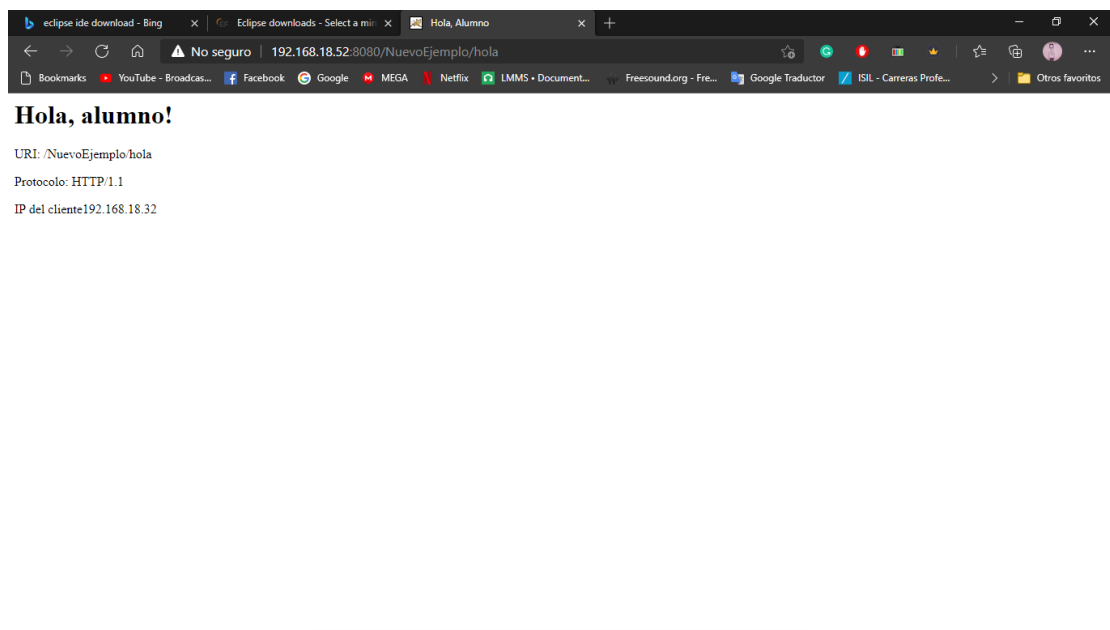
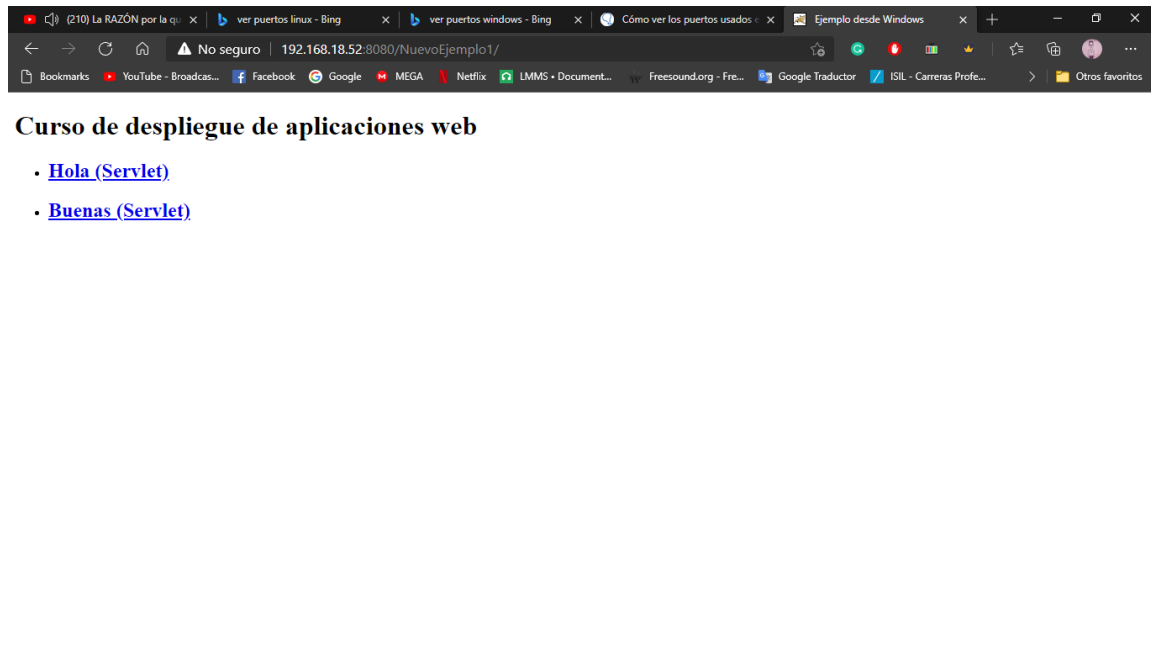


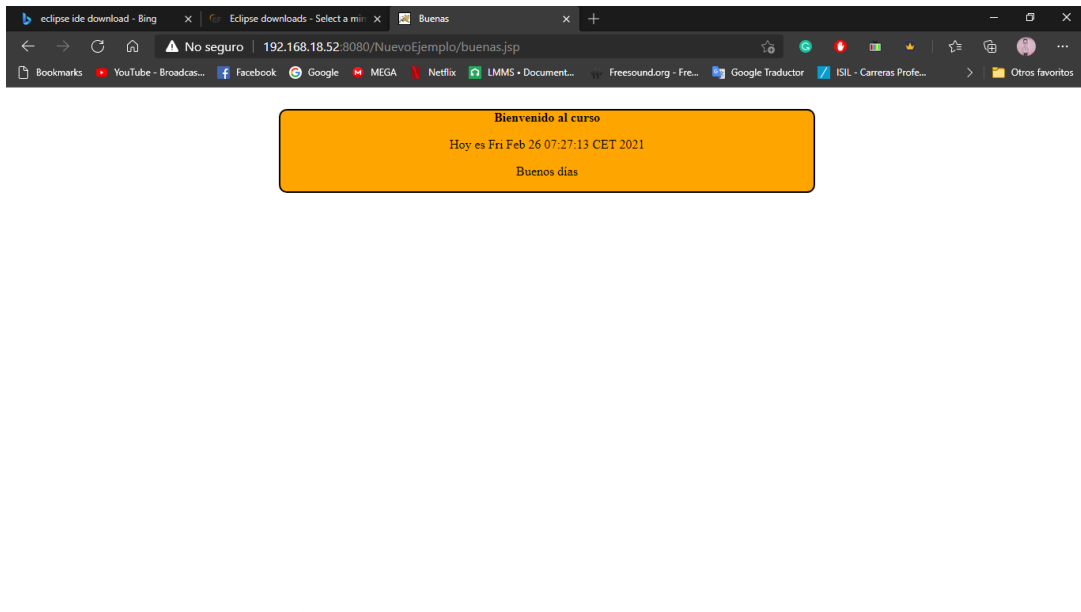


Ahora debemos copiar esta carpeta al directorio **opt/tomcat7/webapps/** y luego encender el Tomcat para desplegar la aplicación desde el servidor:



Despliegue en Windows:

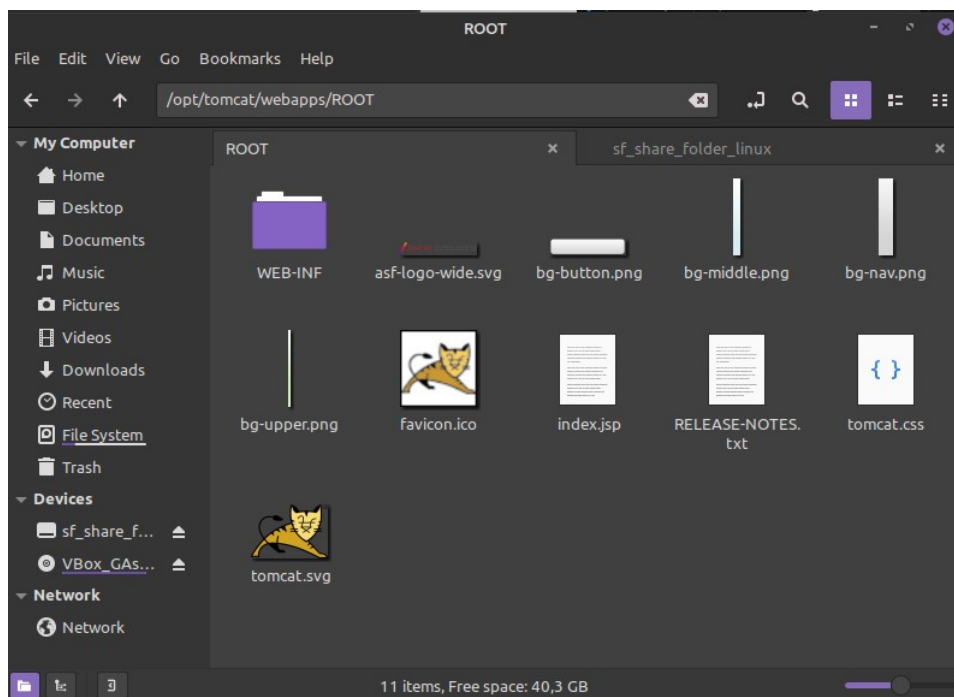




### 3.10 Despliegue de aplicaciones en ROOT Context

En la carpeta **ROOT** dentro de **webapps** se encuentra la página principal del Tomcat, desde dónde podremos acceder al menú de despliegue y demás información del Tomcat. Reemplazaremos el contenido de esta carpeta con el proyecto creado previamente.

Contenido de la carpeta **ROOT**:



Contenido de **index.jsp**

<%--

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. 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.

--%>

<%@ page session="false" pageEncoding="UTF-8" contentType="text/html; charset=UTF-8" %>

<%

java.text.SimpleDateFormat sdf = new java.text.SimpleDateFormat("yyyy");

request.setAttribute("year", sdf.format(new java.util.Date()));

request.setAttribute("tomcatUrl", "https://tomcat.apache.org/");

request.setAttribute("tomcatDocUrl", "/docs/");

request.setAttribute("tomcatExamplesUrl", "/examples/");

%>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title><%=request.getServletContext().getServerInfo() %></title>

<link href="favicon.ico" rel="icon" type="image/x-icon" />

```

<link href="tomcat.css" rel="stylesheet" type="text/css" />
</head>

<body>
  <div id="wrapper">
    <div id="navigation" class="curved container">
      <span id="nav-home"><a href="{tomcatUrl}">Home</a></span>
      <span id="nav-hosts"><a href="{tomcatDocUrl}">Documentation</a></span>
      <span id="nav-config"><a href="{tomcatDocUrl}config/">Configuration</a></span>
      <span id="nav-examples"><a href="{tomcatExamplesUrl}">Examples</a></span>
      <span id="nav-wiki"><a
href="https://wiki.apache.org/tomcat/FrontPage">Wiki</a></span>
      <span id="nav-lists"><a href="{tomcatUrl}lists.html">Mailing Lists</a></span>
      <span id="nav-help"><a href="{tomcatUrl}findhelp.html">Find Help</a></span>
      <br class="separator" />
    </div>
    <div id="asf-box">
      <h1>{pageContext.servletContext.serverInfo}</h1>
    </div>
    <div id="upper" class="curved container">
      <div id="congrats" class="curved container">
        <h2>If you're seeing this, you've successfully installed Tomcat. Congratulations!</h2>
      </div>
      <div id="notice">
        
        <div id="tasks">
          <h3>Recommended Reading:</h3>
          <h4><a href="{tomcatDocUrl}security-howto.html">Security Considerations
How-To</a></h4>
          <h4><a href="{tomcatDocUrl}manager-howto.html">Manager Application How-
To</a></h4>

```

```

        <h4><a href="{tomcatDocUrl}cluster-howto.html">Clustering/Session Replication
How-To</a></h4>
    </div>
</div>
<div id="actions">
    <div class="button">
        <a class="container shadow" href="/manager/status"><span>Server
Status</span></a>
    </div>
    <div class="button">
        <a class="container shadow" href="/manager/html"><span>Manager
App</span></a>
    </div>
    <div class="button">
        <a class="container shadow" href="/host-manager/html"><span>Host
Manager</span></a>
    </div>
</div>
<br class="separator" />
</div>
<div id="middle" class="curved container">
    <h3>Developer Quick Start</h3>
    <div class="col25">
        <div class="container">
            <p><a href="{tomcatDocUrl}setup.html">Tomcat Setup</a></p>
            <p><a href="{tomcatDocUrl}appdev/">First Web Application</a></p>
        </div>
    </div>
    <div class="col25">
        <div class="container">
            <p><a href="{tomcatDocUrl}realm-howto.html">Realms & AAA</a></p>

```

```

        <p><a href="{tomcatDocUrl}jndi-datasource-examples-howto.html">JDBC
DataSources</a></p>
    </div>
</div>
<div class="col25">
    <div class="container">
        <p><a href="{tomcatExamplesUrl}">Examples</a></p>
    </div>
</div>
<div class="col25">
    <div class="container">
        <p><a href="https://wiki.apache.org/tomcat/Specifications">Servlet
Specifications</a></p>
        <p><a href="https://wiki.apache.org/tomcat/TomcatVersions">Tomcat
Versions</a></p>
    </div>
</div>
<br class="separator" />
</div>
<div id="lower">
    <div id="low-manage" class="">
        <div class="curved container">
            <h3>Managing Tomcat</h3>
            <p>For security, access to the <a href="/manager/html">manager webapp</a> is
restricted.
            Users are defined in:</p>
            <pre>$CATALINA_HOME/conf/tomcat-users.xml</pre>
            <p>In Tomcat 9.0 access to the manager application is split between
            different users. &nbsp;<a href="{tomcatDocUrl}manager-howto.html">Read
more...</a></p>
            <br />
            <h4><a href="{tomcatDocUrl}RELEASE-NOTES.txt">Release Notes</a></h4>

```

```

    <h4><a href="{tomcatDocUrl}changelog.html">Changelog</a></h4>
    <h4><a href="{tomcatUrl}migration.html">Migration Guide</a></h4>
    <h4><a href="{tomcatUrl}security.html">Security Notices</a></h4>
</div>
</div>
<div id="low-docs" class="">
    <div class="curved container">
        <h3>Documentation</h3>
        <h4><a href="{tomcatDocUrl}">Tomcat 9.0 Documentation</a></h4>
        <h4><a href="{tomcatDocUrl}config/">Tomcat 9.0 Configuration</a></h4>
        <h4><a href="https://wiki.apache.org/tomcat/FrontPage">Tomcat Wiki</a></h4>
        <p>Find additional important configuration information in:</p>
        <pre>$CATALINA_HOME/RUNNING.txt</pre>
        <p>Developers may be interested in:</p>
        <ul>
            <li><a href="https://tomcat.apache.org/bugreport.html">Tomcat 9.0 Bug
Database</a></li>
            <li><a href="{tomcatDocUrl}api/index.html">Tomcat 9.0 JavaDocs</a></li>
            <li><a href="https://github.com/apache/tomcat/tree/master">Tomcat 9.0 Git
Repository at GitHub</a></li>
        </ul>
    </div>
</div>
<div id="low-help" class="">
    <div class="curved container">
        <h3>Getting Help</h3>
        <h4><a href="{tomcatUrl}faq/">FAQ</a> and <a href="{tomcatUrl}lists.html">Mailing Lists</a></h4>
        <p>The following mailing lists are available:</p>
        <ul>
            <li id="list-announce"><strong><a href="{tomcatUrl}lists.html#tomcat-announce">tomcat-announce</a><br />

```



Important announcements, releases, security vulnerability notifications. (Low volume).

[tomcat-users](#)

User support and discussion

[taglibs-user](#)

User support and discussion for [Apache](#)

[Taglibs](#)

[tomcat-dev](#)

Development mailing list, including commit messages

Other Downloads

[Tomcat](#)

[Connectors](#)

[Tomcat Native](#)

[Taglibs](#)

[Deployer](#)

```
<div class="col20">
```

```
  <div class="container">
```

```
    <h4>Other Documentation</h4>
```

```
    <ul>
```

```
      <li><a href="{tomcatUrl}connectors-doc/">Tomcat Connectors</a></li>
```

```
      <li><a href="{tomcatUrl}connectors-doc/">mod_jk Documentation</a></li>
```

```
      <li><a href="{tomcatUrl}native-doc/">Tomcat Native</a></li>
```

```
      <li><a href="{tomcatDocUrl}deployer-howto.html">Deployer</a></li>
```

```
    </ul>
```

```
  </div>
```

```
</div>
```

```
<div class="col20">
```

```
  <div class="container">
```

```
    <h4>Get Involved</h4>
```

```
    <ul>
```

```
      <li><a href="{tomcatUrl}getinvolved.html">Overview</a></li>
```

```
      <li><a href="{tomcatUrl}source.html">Source Repositories</a></li>
```

```
      <li><a href="{tomcatUrl}lists.html">Mailing Lists</a></li>
```

```
      <li><a href="https://wiki.apache.org/tomcat/FrontPage">Wiki</a></li>
```

```
    </ul>
```

```
  </div>
```

```
</div>
```

```
<div class="col20">
```

```
  <div class="container">
```

```
    <h4>Miscellaneous</h4>
```

```
    <ul>
```

```
      <li><a href="{tomcatUrl}contact.html">Contact</a></li>
```

```
      <li><a href="{tomcatUrl}legal.html">Legal</a></li>
```

```
      <li><a
```

```
href="https://www.apache.org/foundation/sponsorship.html">Sponsorship</a></li>
```

```
      <li><a href="https://www.apache.org/foundation/thanks.html">Thanks</a></li>
```

```

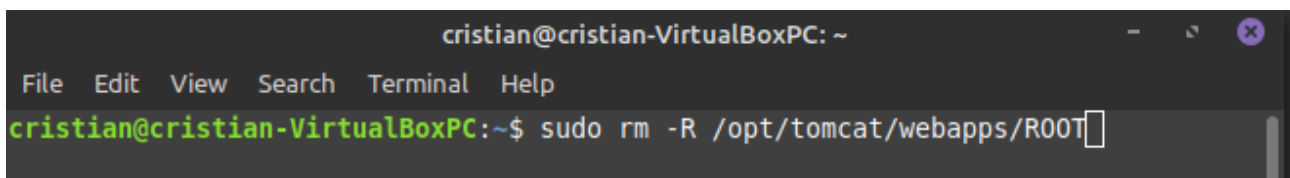
        </ul>
    </div>
</div>
<div class="col20">
    <div class="container">
        <h4>Apache Software Foundation</h4>
        <ul>
            <li><a href="{tomcatUrl}whoweare.html">Who We Are</a></li>
            <li><a href="{tomcatUrl}heritage.html">Heritage</a></li>
            <li><a href="https://www.apache.org">Apache Home</a></li>
            <li><a href="{tomcatUrl}resources.html">Resources</a></li>
        </ul>
    </div>
</div>
<br class="separator" />
</div>
<p class="copyright">Copyright &copy;1999-{$year} Apache Software Foundation. All
Rights Reserved</p>
</div>
</body>

</html>

```

### a) Eliminar el directorio */opt/tomcat/webapps/ROOT*

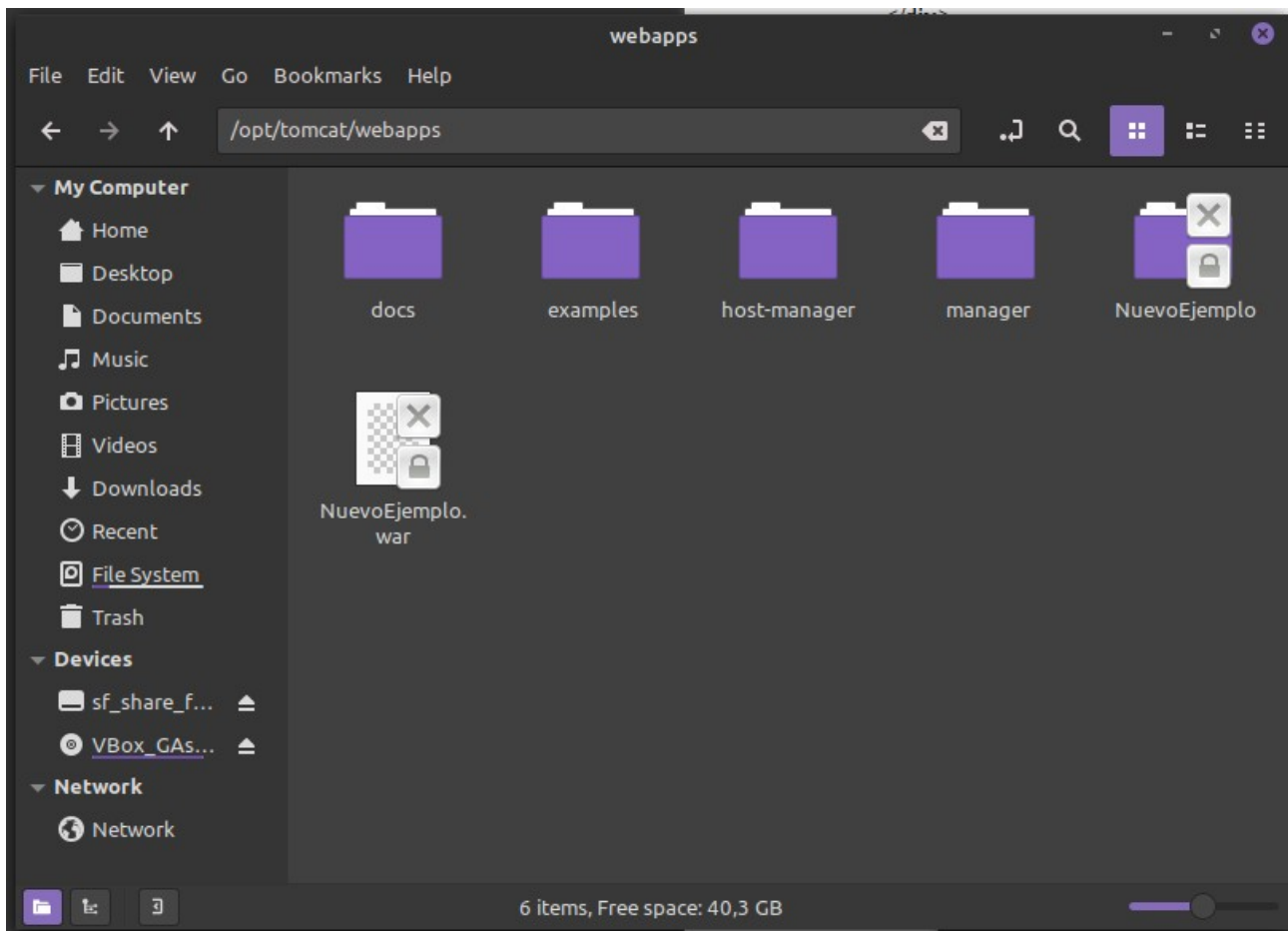
Eliminamos el contenido de **ROOT CONTEX**:



```

cristian@cristian-VirtualBoxPC: ~
File Edit View Search Terminal Help
cristian@cristian-VirtualBoxPC:~$ sudo rm -R /opt/tomcat/webapps/ROOT

```



Reiniciamos el tomcat:

```
cristian@cristian-VirtualBoxPC:~$ sudo systemctl restart tomcat
```

## b) Configuración del proyecto para que se despliegue en ROOT context

Movemos el directorio NuevoEjemplo a /home/cristian/Documents

```
cristian@cristian-VirtualBoxPC:~$ sudo cp -R /opt/tomcat/webapps/NuevoEjemplo /home/cristian/Documents
```

Eliminamos el fichero /opt/tomcat/webapps/NuevoEjemplo.war

```
cristian@cristian-VirtualBoxPC:~$ sudo rm -R /opt/tomcat/webapps/NuevoEjemplo.war
```

Creamos el fichero **ROOT.xml** en /opt/tomcat/conf/Catalina/localhost



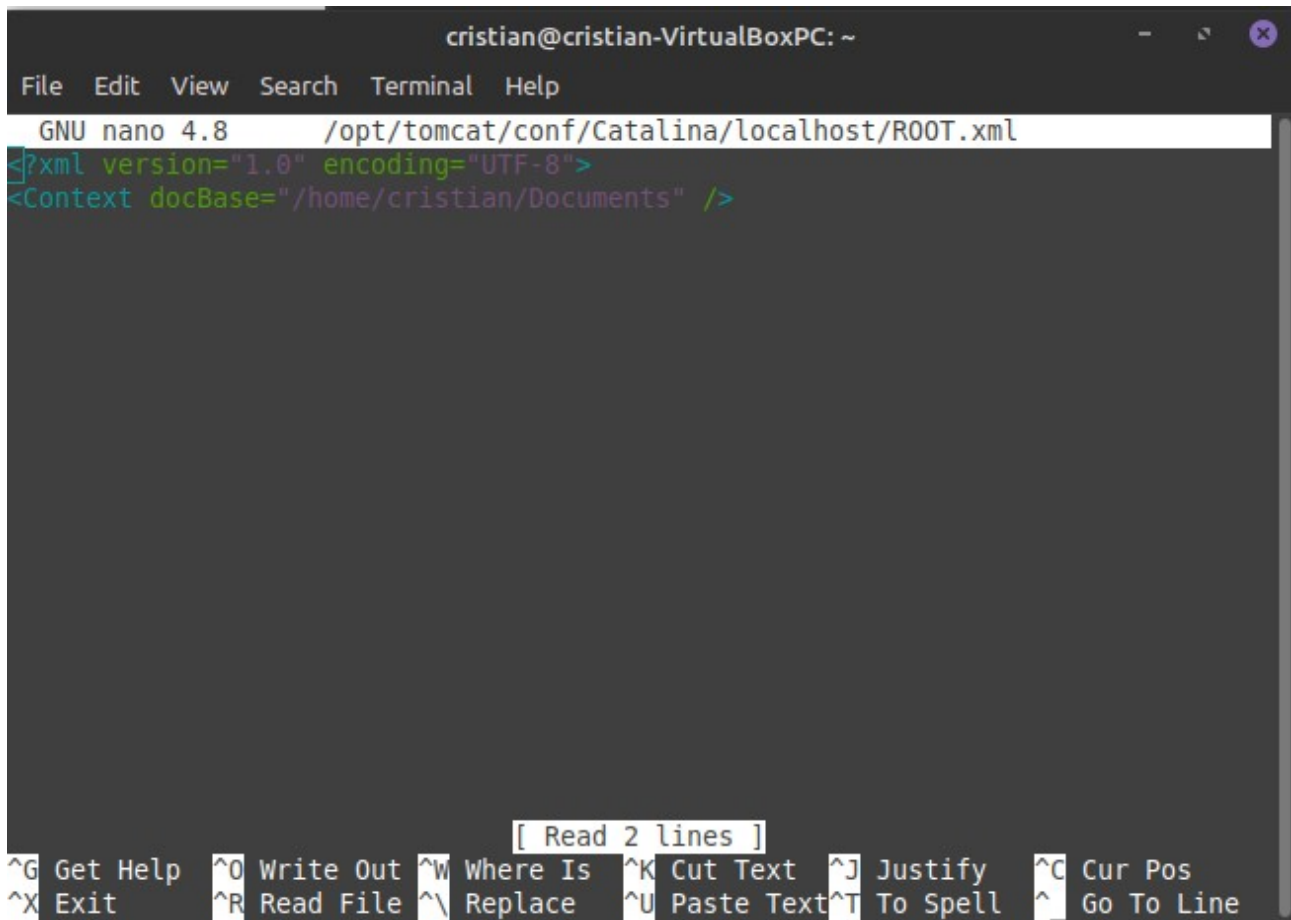








```
cristian@cristian-VirtualBoxPC:~$ sudo nano /opt/tomcat/conf/Catalina/localhost/ROOT.xml
```



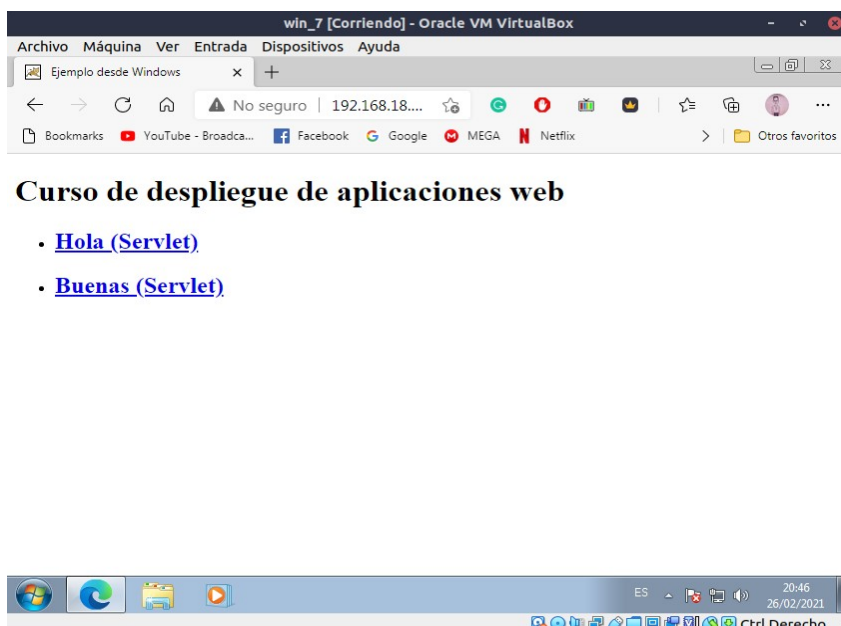
```
cristian@cristian-VirtualBoxPC: ~
File Edit View Search Terminal Help
GNU nano 4.8 /opt/tomcat/conf/Catalina/localhost/ROOT.xml
<?xml version="1.0" encoding="UTF-8">
<Context docBase="/home/cristian/Documents" />

[ Read 2 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line
```

Reiniciamos Tomcat.

### c) Acceso a la aplicación

Abrimos el navegador en la máquina de Windows y comprobamos los resultados:



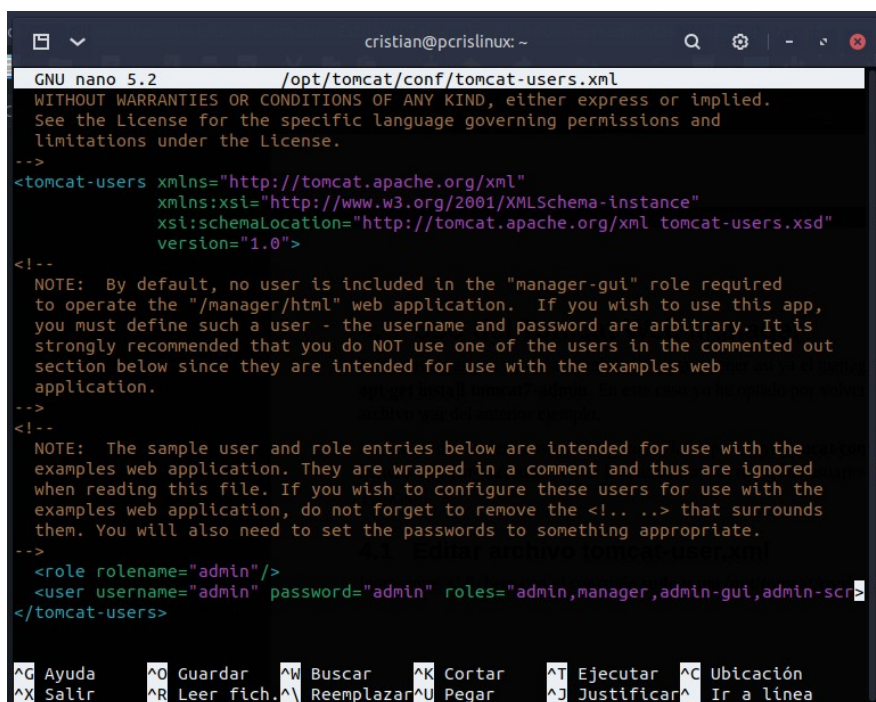
## 4. Uso del Tomcat Web Manager

Podemos optar por volver a instalar el Tomcat y tener así ya el manager o con los comandos **sudo apt-get install tomcat7-admin**. En este caso yo he optado por volver a instalarlo, conservando el archivo war del anterior ejemplo.

Paso seguido, tenderemos que dirigirnos a la carpeta **/opt/tomcat/conf** y dentro encontraremos el fichero **tomcat-user.xml**. A este le añadiremos los role y los usuarios necesarios para administrar Tomcat.

### 4.1 Editar archivo tomcat-user.xml

Ingresamos al fichero con el comando **sudo nano /opt/tomcat/conf/tomcat-users.xml**.

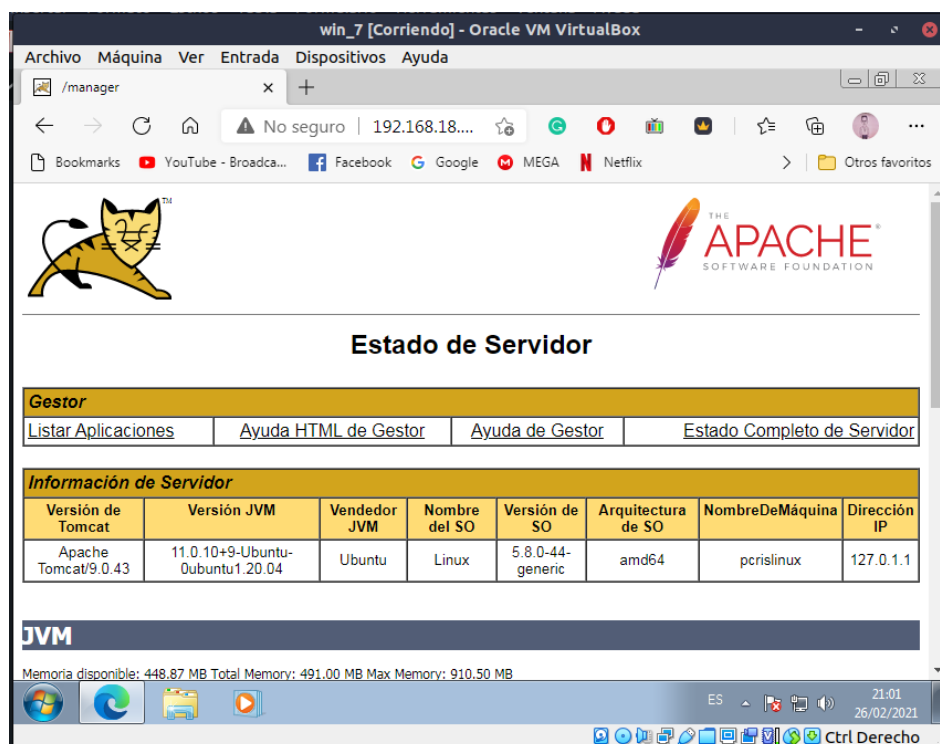
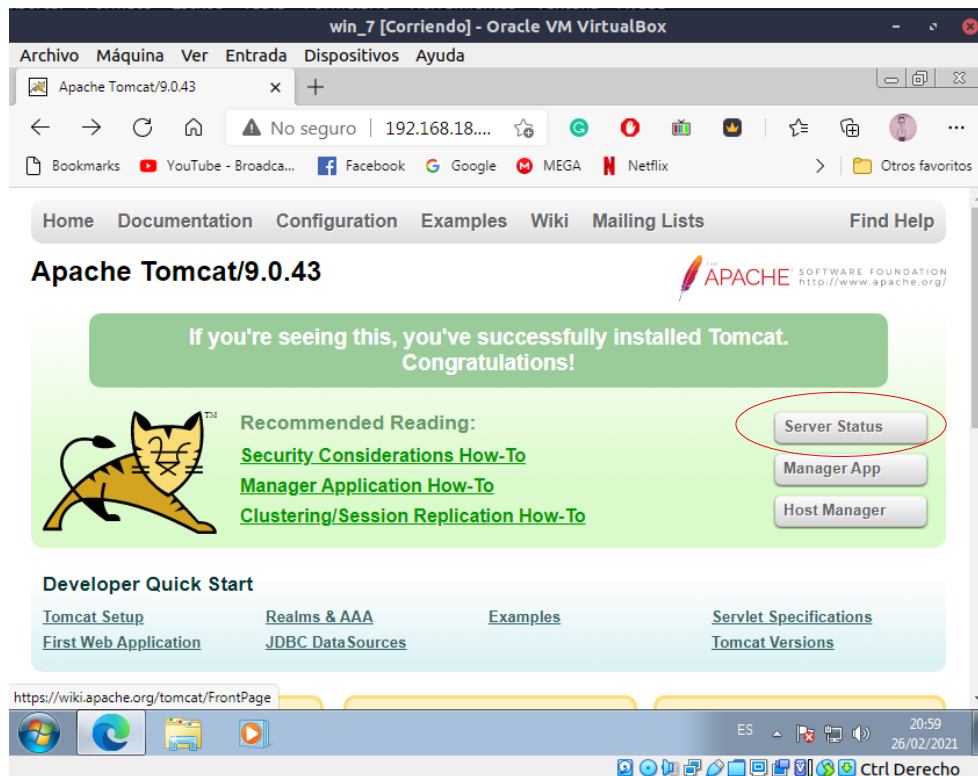


```
GNU nano 5.2 /opt/tomcat/conf/tomcat-users.xml
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
  version="1.0">
<!--
  NOTE: By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application.  If you wish to use this app,
  you must define such a user - the username and password are arbitrary.  It is
  strongly recommended that you do NOT use one of the users in the commented out
  section below since they are intended for use with the examples web
  application.
-->
<!--
  NOTE: 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
  them.  You will also need to set the passwords to something appropriate.
-->
  <role rolename="admin"/>
  <user username="admin" password="admin" roles="admin,manager,admin-gui,admin-script"/>
</tomcat-users>
```

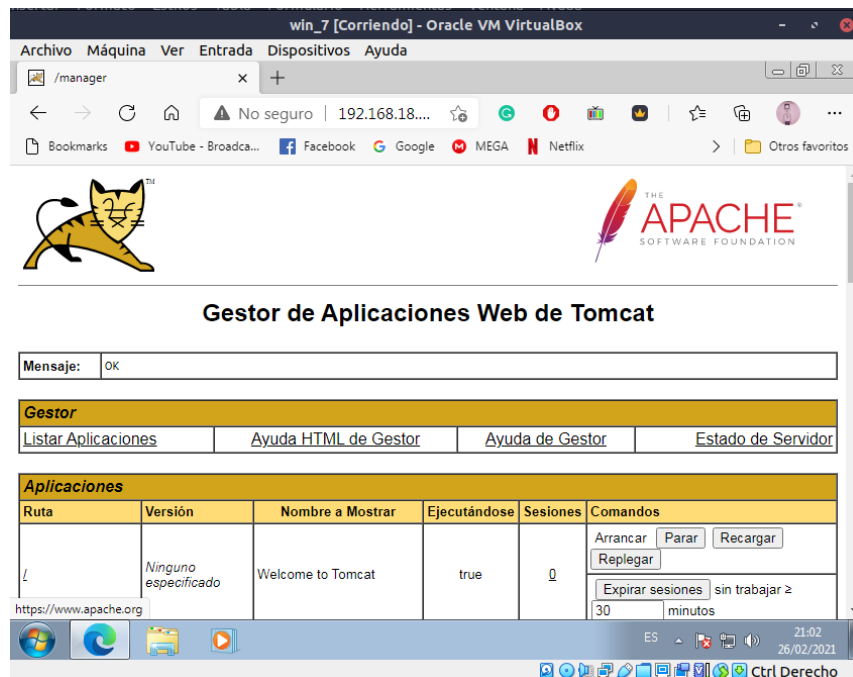
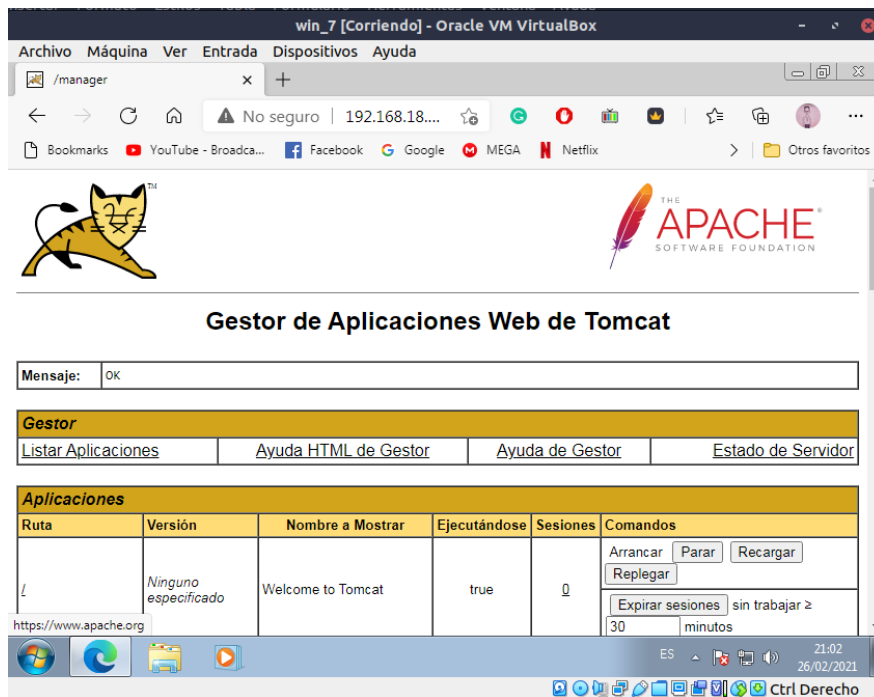
Aquí se asigna el rol del usuario, que en este caso es el administrador y sus roles dentro del entorno, que serán administrador de interfaz, administrador de script y administrador en general.

### 4.2 Comprobación del Tomcat Manager en Windows

Intentaremos loguear al usuario admin desde la máquina virtual de Win7.



Y así también para las opciones de Manager App y Host Manager:



### 4.3 Gestor de aplicaciones Web de Tomcat

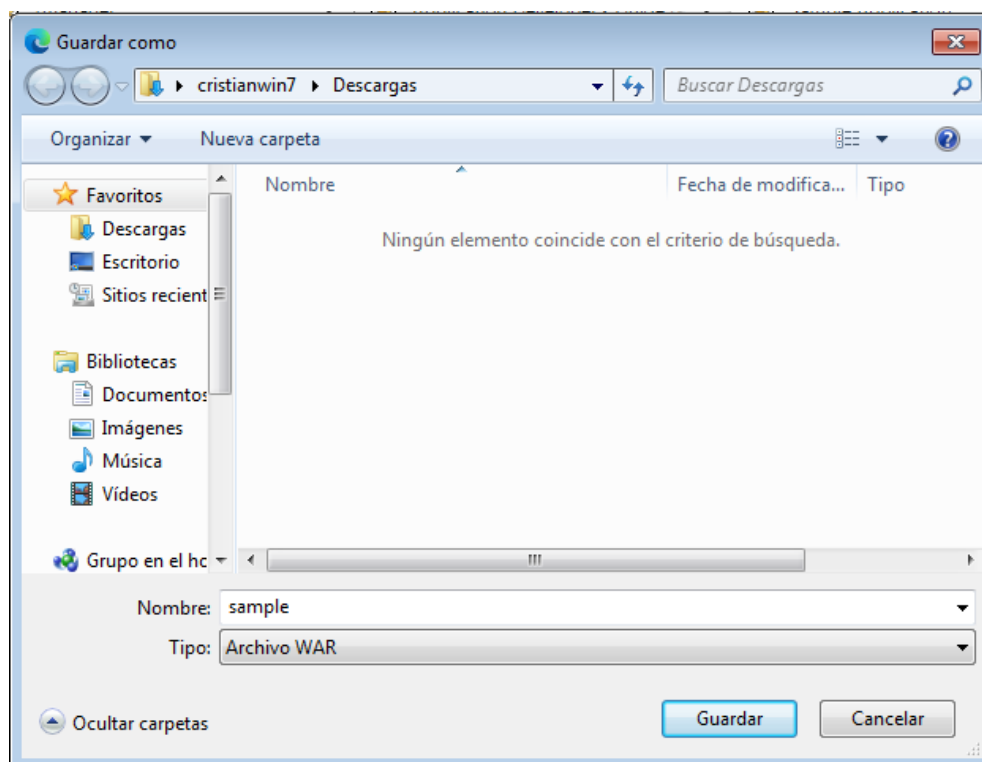
Aquí podremos ver la lista de aplicaciones disponibles en el servidor. También es posible denter, recargar y replegar las aplicaciones disponibles, así también cómo desde ficheros war desplegar nuevas aplicaciones.

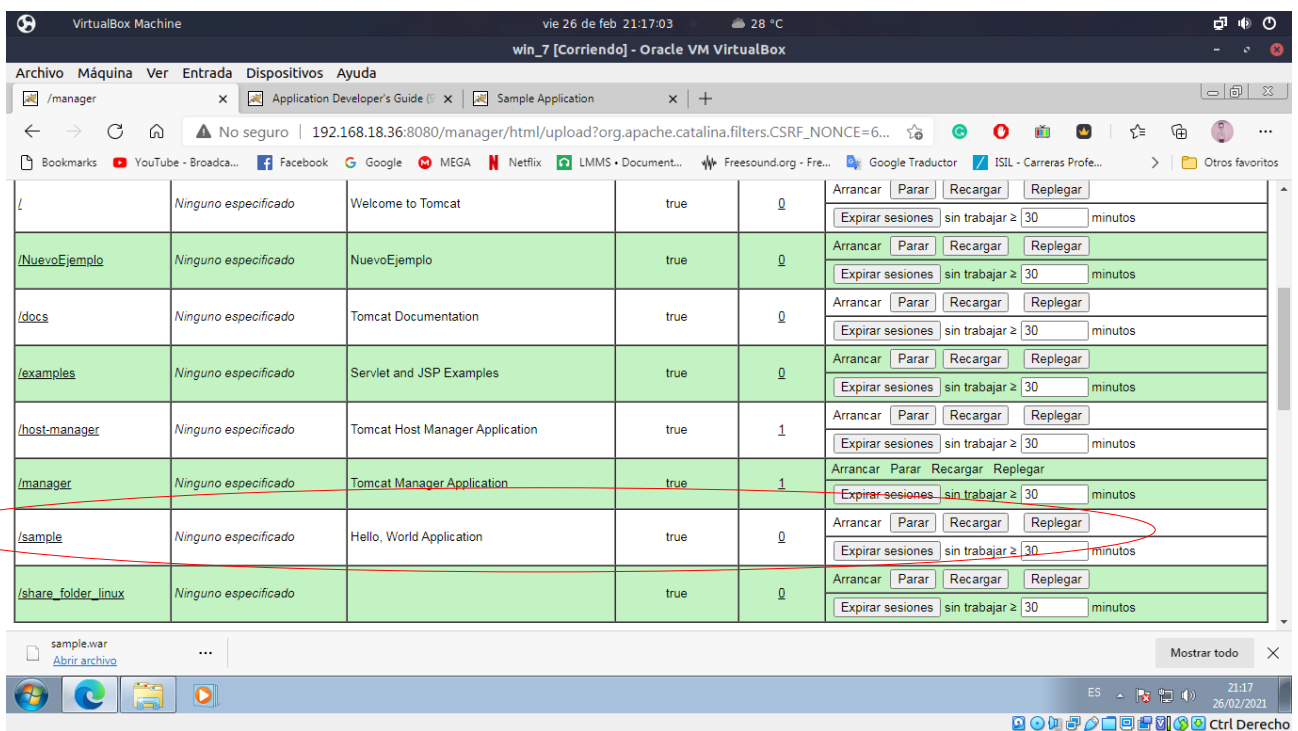
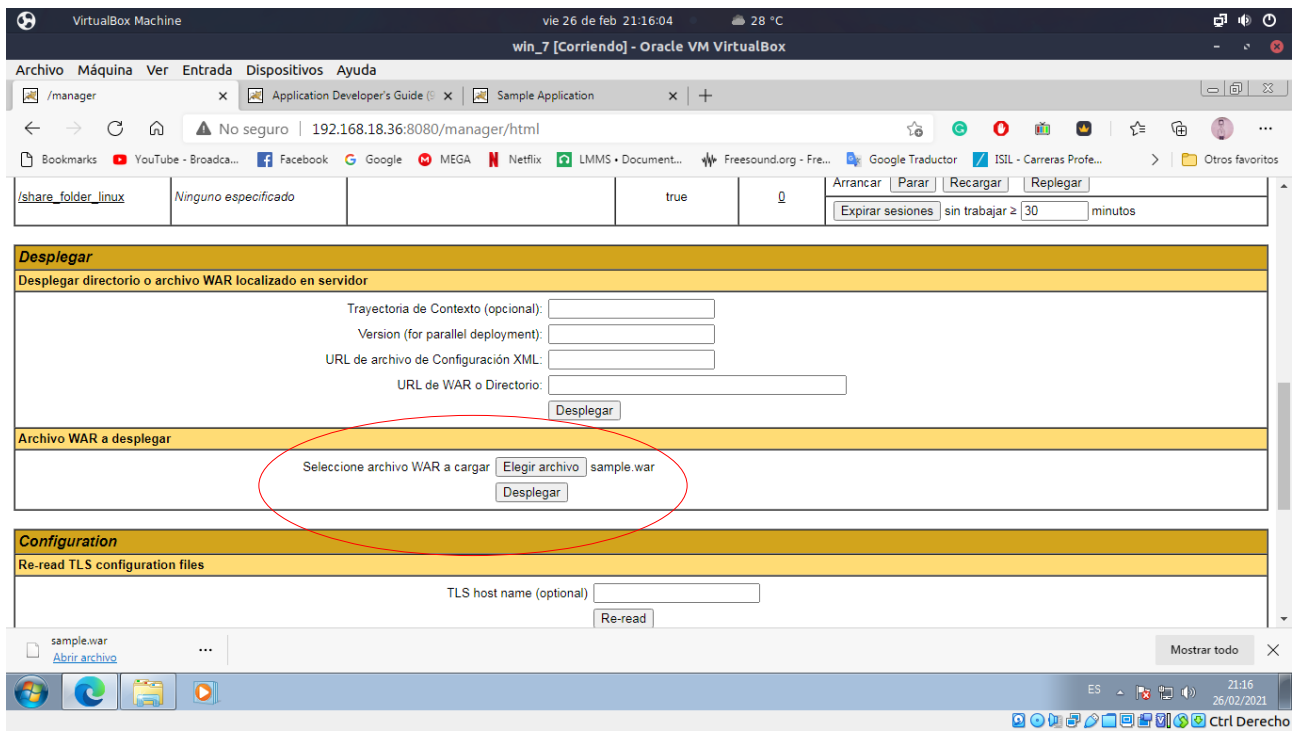
Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguna especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar 30 minutos
/NuevoEjemplo	Ninguna especificado	NuevoEjemplo	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar 30 minutos
/docs	Ninguna especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar 30 minutos
/examples	Ninguna especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar 30 minutos
/host-manager	Ninguna especificado	Tomcat Host Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar 30 minutos
/manager	Ninguna especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar

Además de los controles, hay que tener en cuenta el estado de ejecución de los proyectos, para así saber si podremos visualizarlos o no.

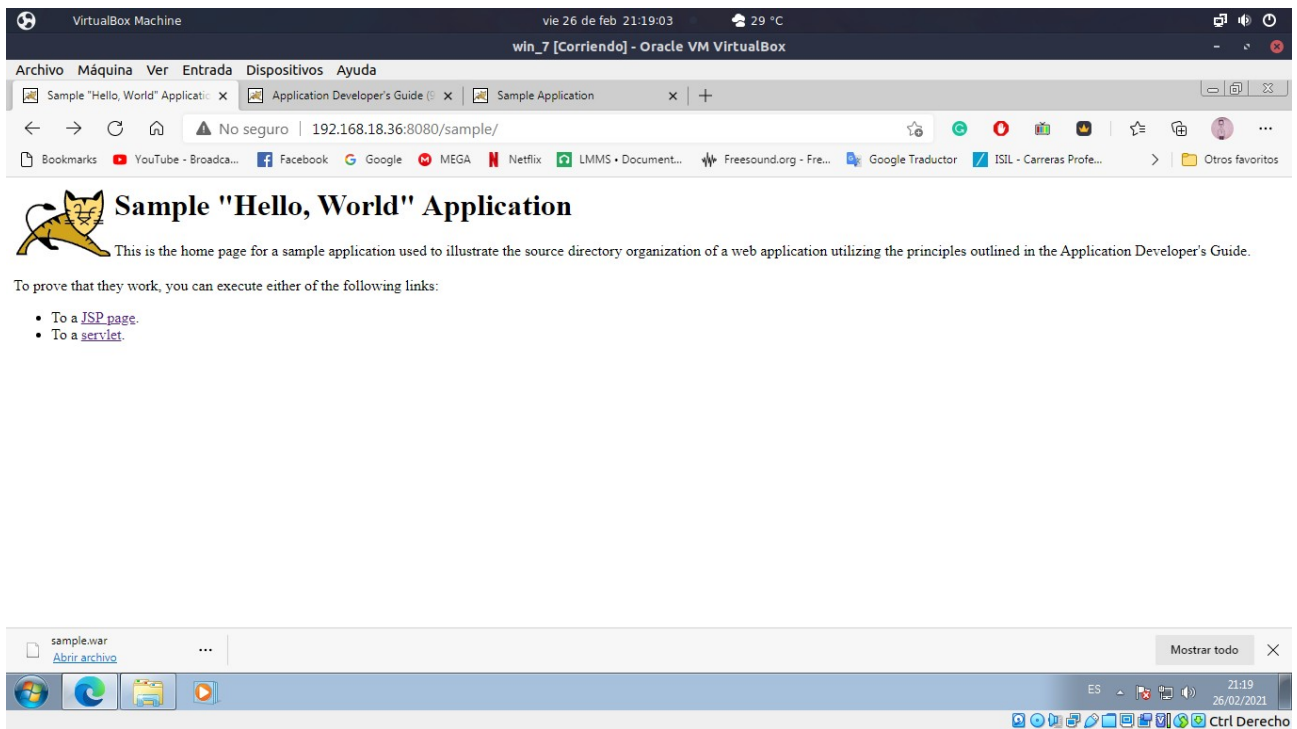
### a) Despliegue de una aplicación .war

Descargaremos un ejemplo de Tomcat y lo desplegaremos por su interfaz para archivos .war.





Y estos son los resultados:



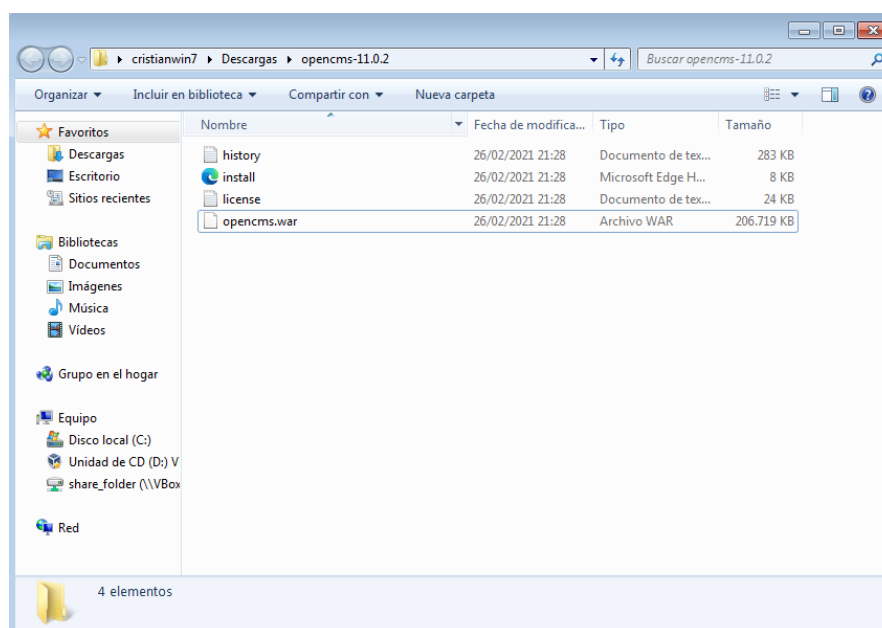
## 5. Despliegue de CMS

Usaremos el OpenCMS, que es un sistema de gestión de archivos basado en JAVA y en XML.

### 5.1 Descarga de CMS

Lo descargaremos e instalaremos en la máquina virtual de Windows.

Obtendremos un archivo **.zip** el cuál tendremos que descomprimir y obtendremos un **.war**.





Antes de proseguir, es necesario dirigirnos a la carpeta **/opt/tomcat/webapps/manager/WEB-INF** y editar el fichero **web.xml** para que tenga una mayor capacidad de almacenamiento y pueda desplegar aplicación más pesadas. Por defecto tiene un máximo de 50 MB, pero lo cambiaremos a 300 MB para que haya ningún inconveniente.

```

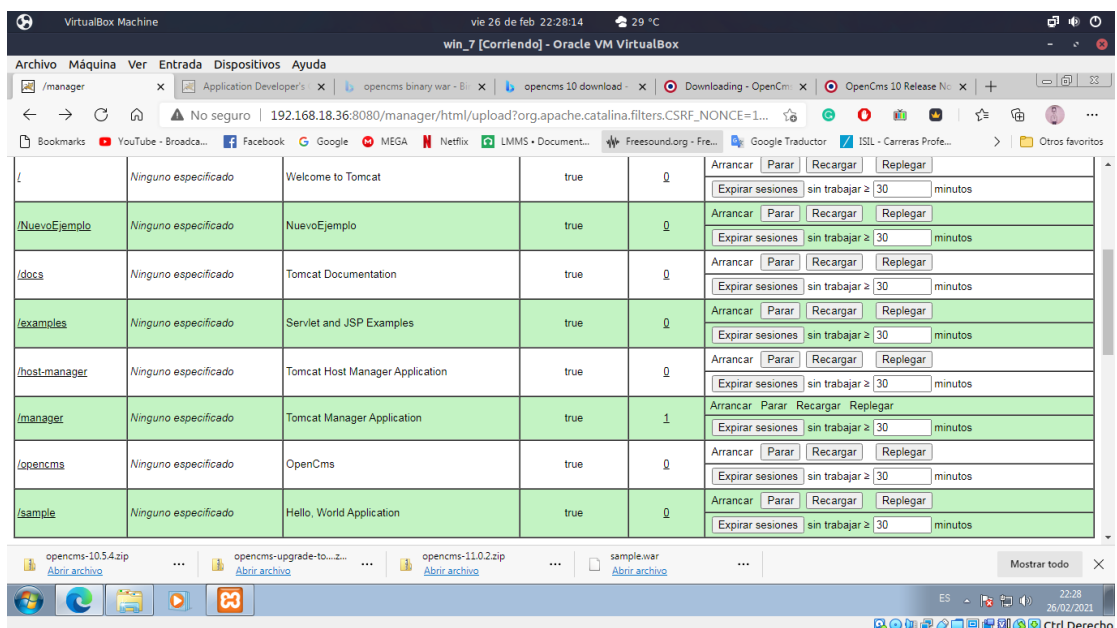
50 <init-param>
51 <param-name>showProxySessions</param-name>
52 <param-value>true</param-value>
53 </init-param>
54 -->
55 <multipart-config>
56 <!-- 300MB max -->
57 <max-file-size>314572800</max-file-size>
58 <max-request-size>52428800</max-request-size>
59 <file-size-threshold>0</file-size-threshold>
60 </multipart-config>
61 </servlet>
62 <servlet>
63 <servlet-name>Status</servlet-name>
64 <servlet-class>org.apache.catalina.manager.StatusManagerServlet</servlet-class>
65 <init-param>
66 <param-name>debug</param-name>
67 <param-value>0</param-value>
68 </init-param>
69 </servlet>

```

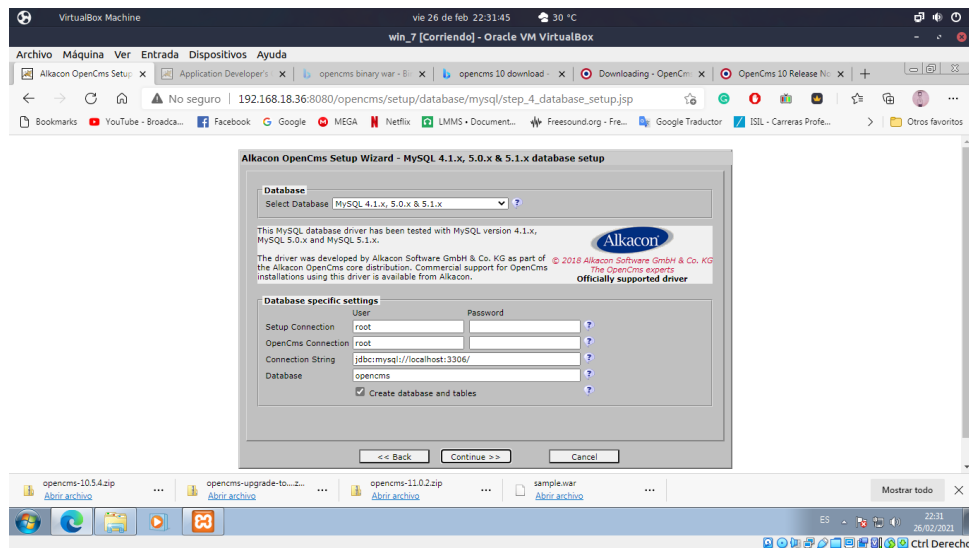
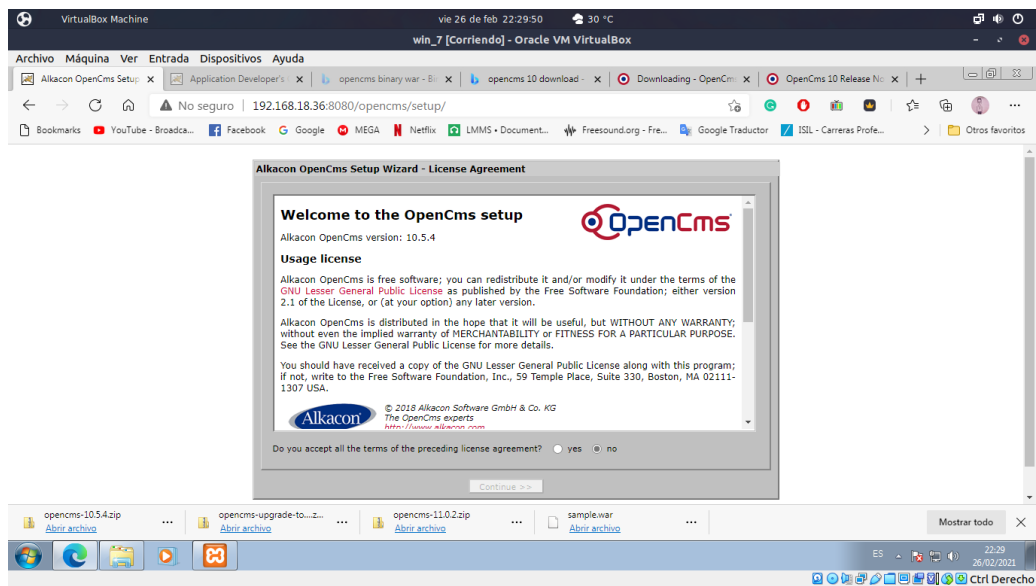
Reiniciamos el Tomcat.

## 5.2 Despliegue de CMS

Procedemos a mostrar el CMS a través de la herramienta de despliegue del Tomcat manager:







## 6. Conclusiones

- Tomcat, aunque por mucha potencia que tenga, es un opersource difícil de entender y manejar. Al igual que en el otro trabajo con Windows, la falta de drivers concretos y la antigüedad del soporte hace muy tedioso encontrar soluciones a problemas de conexión.
- Es útil para quienes estén familiarizados con Java y su entorno, pero tendrán una curva de aprendizaje mucho más empinada en relación a otros servidores y frameworks que se valen de apache para funcionar.
- Lo rescatable es que los archivos war son muy potentes. Son capaces de almacenar gran cantidad de información y eso es positivo para ahorrar espacio.