

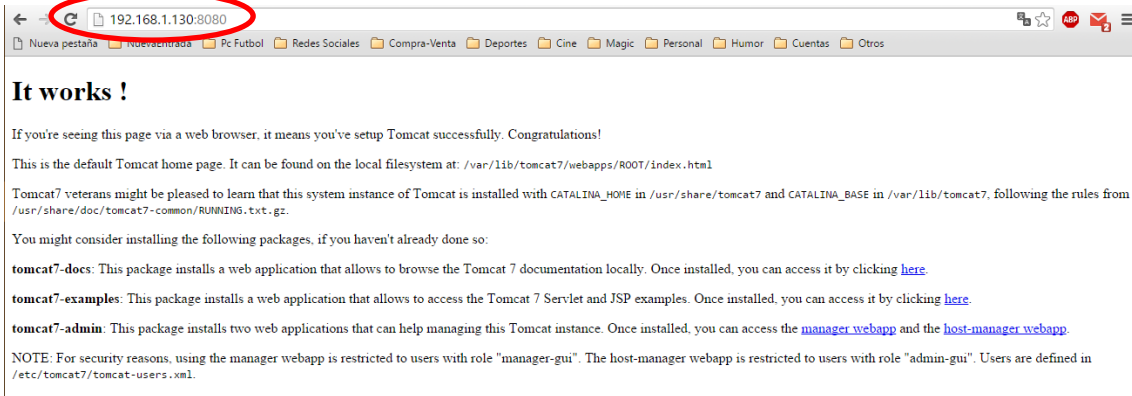


## **PRACTICA 1 DE TOMCAT**

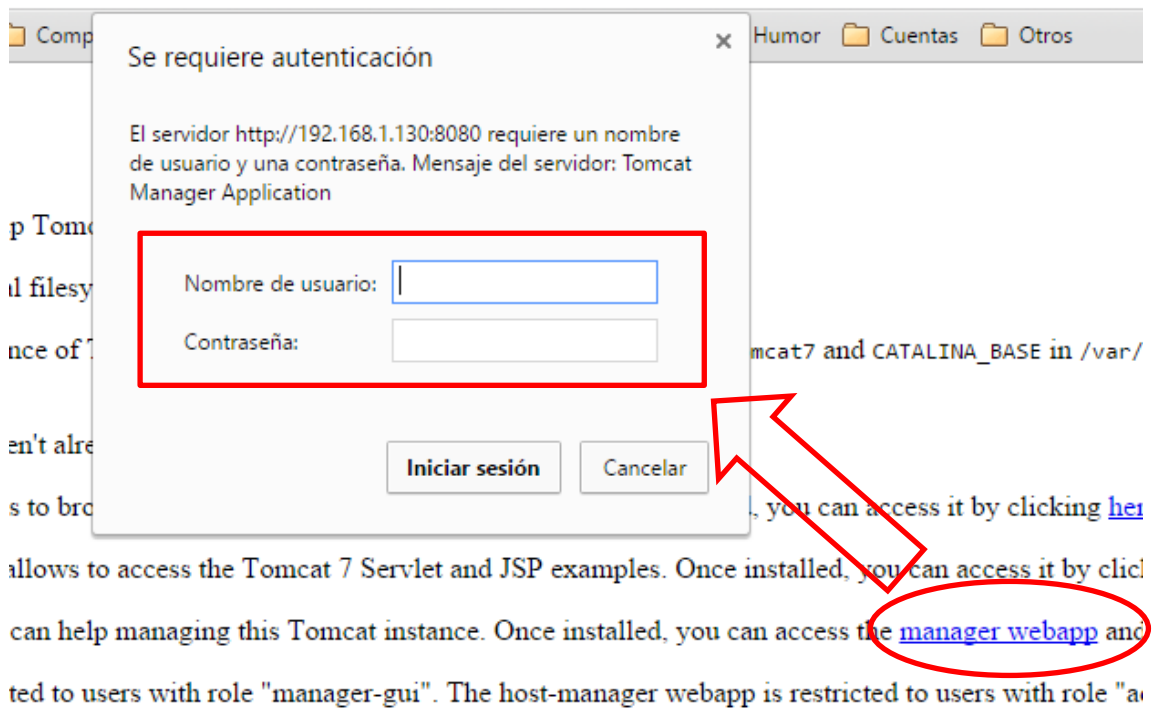
Fecha entrega: 22/01/2016

Autor: Pedro J. Ramos

1. Tenemos un servidor Tomcat instalando y funcionando en la dirección 192.168.1.130 trabajando por el puerto 8080



2. Entramos en el manager webapp e introducimos el usuario administrador que teníamos creado



### 3. Una vez autenticados con el usuario podemos ver el gestor de aplicaciones de Tomcat

The screenshot shows the Apache Tomcat Manager web interface. At the top, there's a navigation menu with various categories like 'Nueva pestaña', 'NuevaEntrada', 'Pc Futbol', etc. Below the menu is the Apache Software Foundation logo and the title 'Gestor de Aplicaciones Web de Tomcat'. A message box shows 'Mensaje: OK'. The main content area has a 'Gestor' section with links for 'Ayuda HTML de Gestor' and 'Ayuda de Gestor'. Below this is a table of applications:

| Trayectoria   | Versión              | Nombre a Mostrar                | Ejecutándose | Sesiones | Comandos   |
|---------------|----------------------|---------------------------------|--------------|----------|--|
| /             | Ninguno especificado |                                 | true         | 0        | Arrancar Parar Recargar Replegar<br>Expirar sesiones sin trabajar 30 minutos |
| /docs         | Ninguno especificado | Tomcat Documentation            | true         | 0        | Arrancar Parar Recargar Replegar<br>Expirar sesiones sin trabajar 30 minutos |
| /examples     | Ninguno especificado | Servlet and JSP Examples        | true         | 0        | Arrancar Parar Recargar Replegar<br>Expirar sesiones sin trabajar 30 minutos |
| /host-manager | Ninguno especificado | Tomcat Host Manager Application | true         | 0        | Arrancar Parar Recargar Replegar<br>Expirar sesiones sin trabajar 30 minutos |
| /manager      | Ninguno especificado | Tomcat Manager Application      | true         | 1        | Arrancar Parar Recargar Replegar<br>Expirar sesiones sin trabajar 30 minutos |

Below the table, there are sections for 'Desplegar' (Deploy) and 'Archivo WAR a desplegar' (WAR file to deploy). The 'Desplegar' section has input fields for 'Trayectoria de Contexto (opcional)', 'URL de archivo de Configuración XML', and 'URL de WAR o Directorio', with a 'Desplegar' button. The 'Archivo WAR a desplegar' section has a 'Seleccionar archivo' button and a 'Desplegar' button.

### 4. Cargamos la aplicación de ejemplo que viene comprimida en un formato war

This is a close-up of the 'Desplegar' section from the previous screenshot. It shows the 'Archivo WAR a desplegar' section where the 'Seleccionar archivo' button is highlighted with a red circle. The text 'sample.war' is visible next to the button, indicating that the example application has been selected for deployment.

### 5. Ahora veremos como en la parte de aplicaciones está instalada la aplicación de ejemplo

This screenshot shows the 'Aplicaciones' table from the Tomcat Manager interface. The table lists various applications, and the 'sample' application row is highlighted with a red circle. The 'sample' application is listed with the name 'Hello, World Application' and is shown as running.

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

- Si entramos en la dirección 192.168.1.130:8080/sample podemos ver el módulo que acabamos de cargar



- Ahora copio el contenido del módulo de ejemplo y modifico algunos de sus archivos empezando por el index.html

```
index.html
1 <html>
2   <head>
3     <title>Practica 1 - Pedro J. Ramos</title>
4   </head>
5   <body bgcolor=white>
6     <table border="0">
7       <tr>
8         <td>
9           
10        </td>
11      </tr>
12    </table>
13    <ul>
14      <li>Ir a <a href="hello.jsp">Pagina en JSP</a>.
15      <li>Ir a <a href="hello">servlet de la tabla del 8</a>.
16    </ul>
17  </body>
18 </html>
```

8. Modifico también el archivo .jsp para que muestre una imagen propia y mi nombre y apellidos

```
index.html x hello.jsp x
1 <html>
2   <head>
3     <title>Pedro J. Ramos</title>
4   </head>
5   <body bgcolor=white>
6     <table border="0">
7       <tr>
8         <td align=center>
9           
10          </td>
11         <td>
12           <h1>Nombre y Apellidos</h1>
13         </td>
14       </tr>
15     </table>
16     <%= new String("Pedro J. Ramos") %>
17   </body>
18 </html>
19
```

9. Y creamos un archivo .java para contener las instrucciones escritas en lenguaje java para después construir el servlet

```
index.html x hello.jsp x hello.java x
1 package mypackage;
2
3 import java.io.IOException;
4 import java.io.PrintWriter;
5 import javax.servlet.ServletException;
6 import javax.servlet.http.HttpServlet;
7 import javax.servlet.http.HttpServletRequest;
8 import javax.servlet.http.HttpServletResponse;
9
10 @Override
11 public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {
12     response.setContentType("text/html");
13     response.setCharacterEncoding("UTF-8");
14     try (PrintWriter write = response.getWriter()){
15         writer.println("<html>");
16         writer.println("<head>");
17         writer.println("<meta charset='UTF-8'/>");
18         writer.println("</head>");
19         writer.println("<body>");
20         writer.println("<h1>Tabla del 8</h1>");
21         for (int i = 0; i < 10; i++){
22             writer.println("8 x " + i + " = " + 8*i);
23         }
24         writer.println("</body>");
25         writer.println("</html>");
26     }
27 }
```

10. Para poder compilar este archivo .java necesitamos el paquete tomcat-api.jar que se encuentra en el servidor en la ruta /usr/share/tomcat7/lib

| Nombre                  | Tamaño | Modificado          | Permisos   | Propiet... |
|-------------------------|--------|---------------------|------------|------------|
| ..                      |        | 17/01/2016 14:59:50 | rw-r-xr-x  | root       |
| annotations-api.jar     | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| catalina.jar            | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| catalina-ant.jar        | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| catalina-ha.jar         | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| catalina-jmx-remote.jar | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| catalina-tribes.jar     | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| commons-dbc.jar         | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| commons-pool.jar        | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| el-api.jar              | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| jasper.jar              | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| jasper-el.jar           | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| jsp-api.jar             | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| servlet-api.jar         | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| <b>tomcat-api.jar</b>   | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| tomcat-coyote.jar       | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| tomcat-i18n-es.jar      | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| tomcat-i18n-fr.jar      | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| tomcat-i18n-ja.jar      | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| tomcat-jdbc.jar         | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |
| tomcat-util.jar         | 1 KB   | 28/03/2015 19:36:12 | rw-rw-rw-r | root       |

11. Para compilar el archivo .java necesitamos utilizar:

```
C:\Users\PJX\Desktop\Nueva carpeta>javac -classpath servlet-api.jar hello.java
```

12. Esta compilación nos deja un archivo .class que es el que debemos incluir en nuestra aplicación

| Nombre      | Fecha de modifica... | Tipo          | Tamaño |
|-------------|----------------------|---------------|--------|
| do          |                      |               |        |
| Hello.class | 30/07/2007 16:26     | Archivo CLASS | 2 KB   |

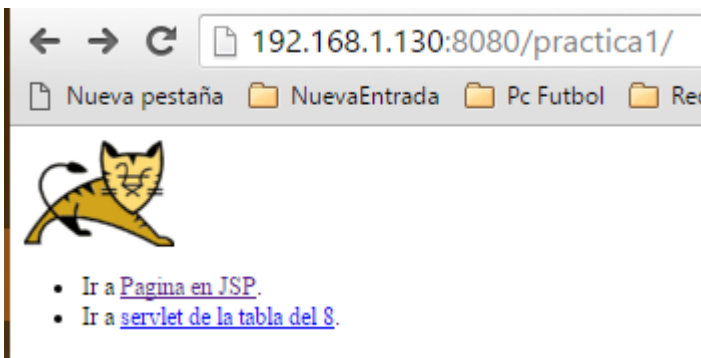
### 13. Hay que modificar el archivo web.xml para incluir la información del servlet

```
index.html x hello.jsp x web.xml x
1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <web-app xmlns="http://java.sun.com/xml/ns/j2ee"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd"
5   version="2.4">
6
7   <display-name>Practica1</display-name>
8   <description>
9     Practica 1 - Tomcat - Pedro J. Ramos
10  </description>
11
12  <servlet>
13    <servlet-name>HelloServlet</servlet-name>
14    <servlet-class>mypackage.Hello</servlet-class>
15  </servlet>
16
17  <servlet-mapping>
18    <servlet-name>HelloServlet</servlet-name>
19    <url-pattern>/hello</url-pattern>
20  </servlet-mapping>
21
22 </web-app>
```

### 14. Ahora comprimimos en un archivo partica1.war la aplicación para ser desplegado en el servidor tomcat

| Desplegar   |  |
|---|--|
| 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"/>                  |  |
| Archivo WAR a desplegar                                   |  |
| Seleccione archivo WAR a cargar:                          | <input type="button" value="Seleccionar archivo"/> practica1.war |
| <input type="button" value="Desplegar"/>                  |  |

### 15. Una vez desplegada podemos acceder a ella mediante la dirección 192.168.1.130:8080/practica1





## Estructura de los archivos

practica1

| Nombre      | Fecha de modifica... | Tipo                | Tamaño |
|-------------|----------------------|---------------------|--------|
| META-INF    | 30/07/2007 17:18     | Carpeta de archivos |        |
| WEB-INF     | 30/07/2007 16:26     | Carpeta de archivos |        |
| hello.java  | 01/02/2016 19:02     | Archivo JAVA        | 1 KB   |
| hello.jsp   | 01/02/2016 18:24     | Archivo JSP         | 1 KB   |
| index       | 01/02/2016 17:52     | Chrome HTML Do...   | 1 KB   |
| servlet-api | 28/03/2015 19:36     | Archivo WinRAR      | 193 KB |
| images      | 01/02/2016 17:49     | Carpeta de archivos |        |

practica1 > WEB-INF > classes > mypackage

| Nombre      | Fecha de modifica... | Tipo          | Tamaño |
|-------------|----------------------|---------------|--------|
| Hello.class | 30/07/2007 16:26     | Archivo CLASS | 2 KB   |