

Objectius

L'objectiu de la pràctica és crear un servidor web funcional que treballi amb combinació amb Tomcat.

- Haureu d'aconseguir un servidor amb un nom de domini elvostrenom.com que permeti la creació de pàgines web dinàmiques en llenguatge php.
- Heu de donar accés a la gestió de bases de dades a través de phpMyAdmin.
- Creeu una aplicació bàsica de consulta a la base de dades.
- Sota la url elvostrenom.com/servlets/ heu de visualitzar una pàgina estàtica amb 2 links a servlets creats amb Java (veure següent punt).
- Incorporeu un parell de servlets de mostra senzills en les url
 - elvostrenom.com/servlets/servlet-one
 - elvostrenom.com/servlets/servlet-two

Haureu de fer un manual d'instal·lació, per a que qualsevol persona a partir d'un SO Ubuntu Desktop 18.04 pugui reproduir el que aquí es proposa. La documentació ha de ser ben acurada, heu de proporcionar les fonts d'informació, exposar els problemes que us heu trobat en el procés i exposar les solucions en aquests problemes.

Punto 1: Creación de nuestro dominio + integración con PHP

Paso 1: Primero necesitaremos instalar tomcat en nuestro ordenador, en nuestro caso optaremos por instalarlo de la forma fácil:

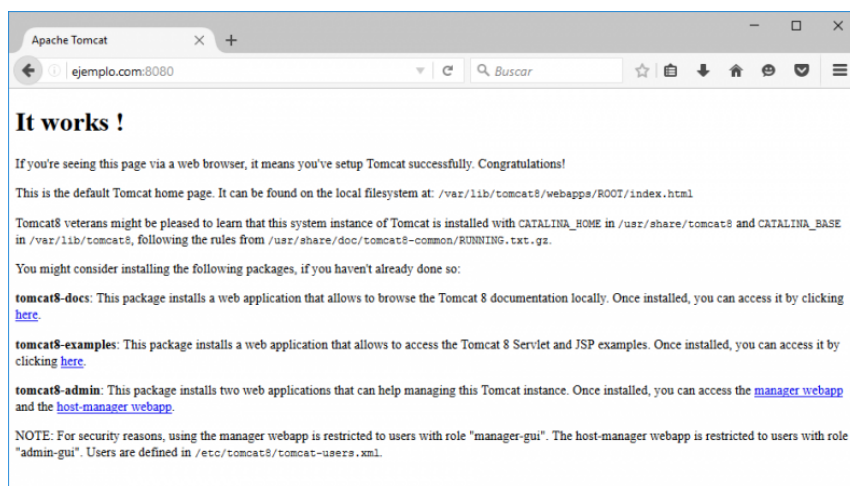
- // Metodo de instalación mas fácil.

```
apt-get install tomcat8-docs tomcat8-examples tomcat8-admin
```

- // Encender, parar y reiniciar el servidor.

```
systemctl start tomcat8  
systemctl stop tomcat8  
systemctl restart tomcat8
```

De esta forma solo obtenemos lo que es el servidor en si, esquivamos la opción de instalarlo con el gestor grafico para monitorear nuestro servidor tomcat.



Paso 2: Ahora nos descargaremos netbeans + JDK, la razón de la descarga

de netbeans es para realizar la creación de servlets mas fácil y el JDK porque es obligatorio descargarlo para la creación de nuestros servlets.

Paso 3: Conectaremos nuestro servidor tomcat con netbeans, para ello nos iremos a la siguiente ruta: **“Tools > Servers > Add server”**.

En esta pagina nos pedirá que digamos cual es el directorio de nuestro servidor Tomcat (`/var/lib/tomcat8`)

y nos dará la posibilidad de poder crear un usuario y contraseña para editar los archivos dentro de nuestro nuevo server (Nuevo dominio)

The screenshot shows the 'Add Server Instance' dialog box in NetBeans. The 'Steps' sidebar on the left indicates the current step is '2. Installation and Login Details'. The main panel contains the following fields and options:

- Server Location:** A text field with a 'Browse ...' button.
- ☐ Use Private Configuration Folder (Catalina Base)
- Catalina Base:** A text field with a 'Browse ...' button.
- Enter the credentials of an existing user in the manager or manager-script role**
- Username:** A text field.
- Password:** A text field.
- ☒ Create user if it does not exist

At the bottom of the dialog are buttons for 'Help', '< Back', 'Next >', 'Finish', and 'Cancel'. An information icon and the text 'Specify the Server Location (Catalina Home).' are located above the 'Next >' button.

Despues de crear nuestro nuevo servidor, crearemos un nuevo proyecto de tipo java web (web application).

(Pestaña de arriba, crear proyecto, java web > web application, y pondremos el nombre de nuestro nuevo dominio (nombreusuario.com)).

Paso 4: Llegados a este paso deberíamos tener listo los siguientes recursos:

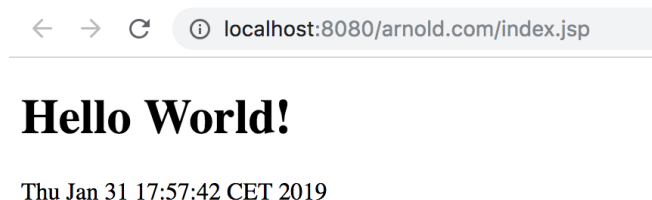
- Apache tomcat instalado
- Netbeans
- JDK

Antes de continuar con la compatibilidad con PHP probaremos el funcionamiento de nuestro servidor tomcat creando un fichero JSP, para ello será tan sencillo como:

1. Click derecho en nuestro dominio, pulsar new , y seleccionamos JSP.
2. Lo llamaremos como quedamos, en mi caso lo llame index.jsp para que sea el primer archivo en mostrar al llamar al servidor.
3. Para comprobar que el servidor tomcat y java funcionan bien añadiremos la siguiente línea dentro del body de nuestro nuevo archivo JSP.

```
<%=new java.util.Date() %>
```

4. El resultado debería ser el siguiente:

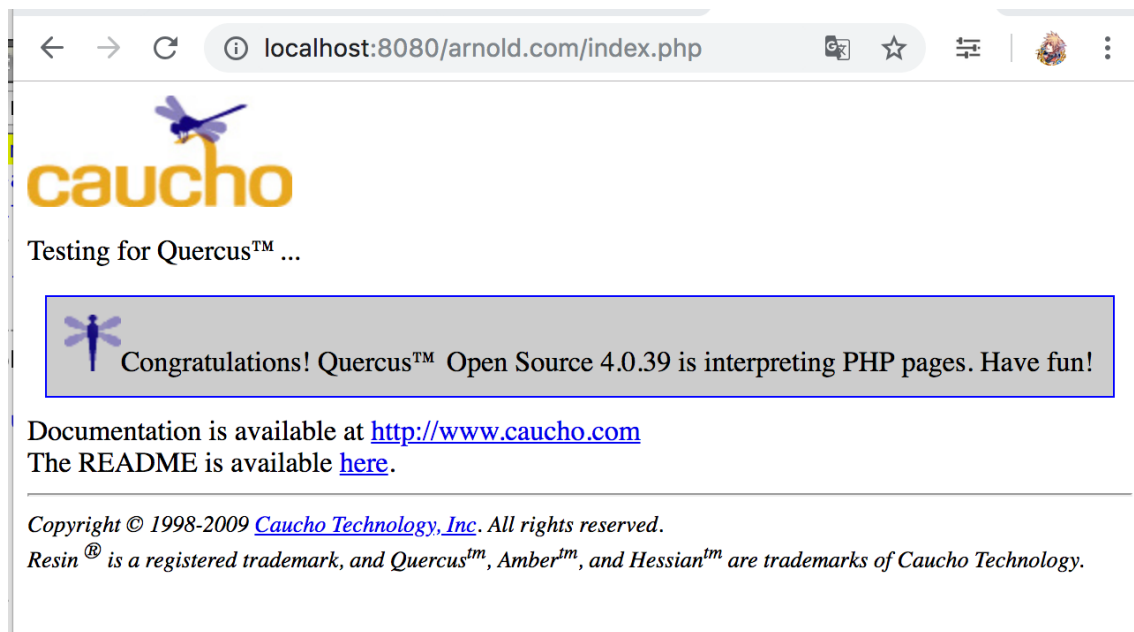


Paso 5: Ahora empezaremos con la implementación de PHP en el servidor tomcat, para ello nos descargaremos el fichero .war de esta pagina:

<http://quercus.caucho.com>

Desde el terminal, iremos al directorio de nuestro proyecto (El servidor que creamos anteriormente, no confundir con el directorio tomcat) y lo descomprimiremos.

Ahora deberíamos tener compatibilidad con PHP, para comprobarlo abriremos el fichero “index.php” que es paquete nos habrá entregado para comprobar si la compatibilidad con php existe en el servidor tomcat:



Punto 2 y 3: Incorporación de bases de datos (phpmyadmin)

Paso 1: Descargamos e instalamos phpmyadmin (yo directamente me instale XAMPP)

1. Descargar XAMPP de : <https://www.apachefriends.org/download.html>
2. Dar permisos al paquete descargado (chmod 755 (Usuario: permiso total , grupo y demás leer y ejecución)
3. Ejecutar el instalador (Para ejecutar scripts en Ubuntu es con “.”)

```
sudo ./xampp-linux-*-installer.run
```

4. Termina la instalación con el asistente de instalación.

Paso 2: Ahora ya tenemos instalado phpmyadmin (Gestor de base de datos) junto MySQL (sistema de gestión de base de datos)

Ahora lo vincularemos con nuestro servidor tomcat descargando un paquete que relacionara a ambos (Connector/J):

```
https://dev.mysql.com/downloads/connector/j/8.0.html
```

Una vez descargado, lo descomprimos y buscaremos el siguiente archivo:

```
mysql-connector-java-5.1.47-bin.jar
```

Una vez encontrado, abriremos netbeans, seleccionamos nuestro proyecto web y en la sección de “librería” seleccionaremos “añadir nuevo paquete JAR” y seleccionaremos el archivo anteriormente mencionado. **(Punto 2 terminado)**

Paso 3: Con la conexión terminada y el gestor de base de datos instalado probaremos la conexión con un pequeño CRUD en php, en mi caso, descargue uno simple de github.

<https://github.com/chapagain/crud-php-simple>

Descargamos el CRUD y lo insertamos en un nuevo directorio dentro de nuestro proyecto web (Servidor tomcat).

Antes de probar el proyecto debemos ejecutar los comandos dentro del fichero “database.sql” en el terminal de phpmyadmin y en “database.php” dar la información sobre nuestra base de datos (Usuario:root ; contraseña:“”;database:”test”;))

Ahora simplemente probaremos si funciona correctamente abriendo el fichero index.php de este nuevo proyecto.



Ilustración 1: Si no se obtuvo ningún mensaje de error como en la foto es que lo hicimos todo perfectamente.

Punto 4 y 5: Manejando servlets en java.

Paso 1: Realizar click derecho en el proyecto web > Nuevo > servlet y nombre de los nuevos servlets (2 y 3).

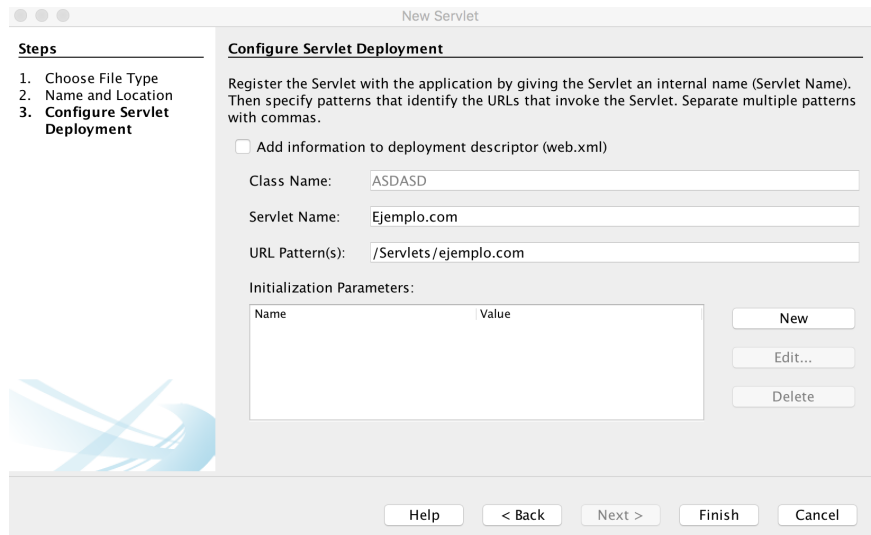
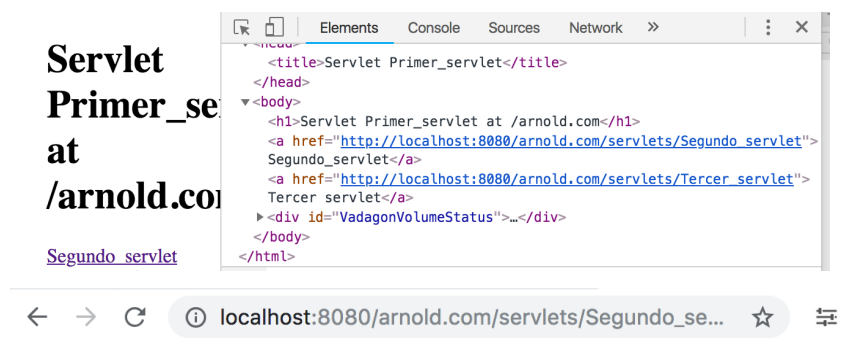


Ilustración 2: Existen 3 tipos de nombres para un servlet, el nombre de la clase (Ya que un servlet equivale a una clase, el nombre del servlet y su URL.

Paso 2: Crear una pagina estatica (HTML) y con la etiqueta “a” hacer un enlace a esos servlets.

Paso 3: Probarlos desde la pagina estatica:



Servlet Segundo_servlet at /arnold.com