

# **Práctica 1: Instalación de un servidor de aplicaciones web y diseño de la vista de una aplicación**

Sistemas de Información

Grado de Informática

Dpto. de Informática e Ingeniería de  
Sistemas, Universidad de Zaragoza  
Escuela de Ingeniería y Arquitectura

2 de octubre de 2019

## **1. Objetivos**

Las aplicaciones Web son aplicaciones que se ejecutan en, al menos, un servidor Web y a las que los usuarios acceden desde un cliente de propósito general (por ejemplo: un navegador Web en un PC, teléfono móvil, etc.). Se caracterizan por poseer un contenido estático (html, pdf, etc.) y contenido dinámico, que se genera cuando los usuarios realizan diferentes peticiones. El contenido dinámico lo genera el contenedor/servidor de aplicaciones Web.

En esta práctica se usará un servidor de aplicaciones (preferentemente un servidor conforme a Java Enterprise Edition - JavaEE, aunque puede usarse cualquier otra tecnología Web que respete las mismas funcionalidades) para trabajar conceptos básicos de programación Web en posteriores sesiones. Además, se diseñará la vista correspondiente a una aplicación web que se implementará en las siguientes sesiones. En mayor detalle, los objetivos de esta práctica son:

- Determinar las funcionalidades que va a ofrecer el sistema de información Web

que vais a desarrollar.

- Analizar y reflexionar sobre el modelo de negocio ligado al sistema de información que comenzáis a desarrollar en esta práctica.
- Diseñar la capa vista de dicho sistema de información Web.
- Instalar el servidor de aplicaciones Web Apache Tomcat u otro servidor Web con el que deséis trabajar en prácticas posteriores.
- Configurar una aplicación Web sencilla en dicho servidor.

## 2. Contenidos

La práctica consta de dos bloques, de los cuales el primero requiere el prototipado y construcción de la primera versión de la capa vista del sistema de información Web cuyos requisitos habéis recogido en la práctica 0 y el segundo consiste en la instalación de dicha vista en un servidor de aplicaciones Web (por ejemplo, Tomcat si se opta por tecnología JEE). Para la creación de la capa vista se debe emplear HTML y CSS. Además, opcionalmente, se pueden usar también Javascript y las librerías que se considere oportunas (por ejemplo, Bootstrap: <https://www.w3schools.com/bootstrap>). Por último, se requiere que se encapsule el prototipo desarrollado y se instale en un servidor Web (por ejemplo, Apache Tomcat).

### 2.1 Diseño de la capa vista de una aplicación Web

Se pretende desarrollar un sistema de información Web que permita a los diferentes tipos de usuarios acceder los servicios ofrecidos en función de sus perfiles y necesidades. Los usuarios del sistema de información podrán estar registrados en el sistema o no (si no están registrados se consideran usuarios anónimos), y según su perfil, podrán autorregistrarse o no. Los usuarios registrados también podrán consultar y modificar su perfil y datos sobre su actividad.

En el caso de un usuario registrado se almacenarán al menos los siguientes datos cuando se dé de alta en el sistema: *login*, apellidos, nombre, año de nacimiento, email, clave de acceso y tipo de usuario. Cada usuario puede solicitar la creación de uno o más registros siempre y cuando en cada uno de ellos emplee una dirección de correo electrónico (e-mail) diferente.

Se debe tener en cuenta que para almacenar la información propia del sistema (usuarios y otros datos) es necesario diseñar e implementar una base de datos. Por ello, en la siguiente práctica, se diseñará y creará la base de datos que dará soporte a vuestro

sistema.

## Requisitos

Se debe diseñar la vista de una aplicación web (sistema de información web) usando al menos los lenguajes: HTML y CSS. Esta interfaz debe permitir al menos realizar las siguientes operaciones:

- Formulario de contacto
  - Formulario que permita ponerse en contacto con los responsables del sitio Web para solicitarle información o requerirles algún tipo de servicio.
- Dar de baja usuarios, en caso de que se permitan usuarios registrados.

Además de estas funcionalidades, habrá que desarrollar aquellas que resulten necesarias, en base al diseño funcional que hayáis realizado del proyecto.

En cada caso, se diseñarán los formularios de entrada de datos que se consideren oportunos y un esqueleto de cómo deberían ser las páginas a mostrar al realizarse las operaciones que se consideren, tanto correcta como incorrectamente. En prácticas posteriores se le añadirá el comportamiento dinámico a la interfaz creada.

Para realizar el diseño de la vista podéis hacer uso de las herramientas de prototipado que creáis convenientes (como por ejemplo Balsamic). Existen una gran cantidad de ellas que os pueden ayudar con el diseño a la hora de estructurar el contenido de vuestra página, así como a obtener una usabilidad idónea.

## 2.2 Instalación del servidor de aplicaciones web Apache Tomcat

Para realizar la instalación del servidor de aplicaciones Tomcat en nuestra cuenta basta con descargar (de <https://tomcat.apache.org>) el fichero tar.gz o zip correspondiente a la versión que deseamos instalar y descomprimirlo. Para arrancar y parar el servidor se emplean los scripts start-up.sh y shut-down.sh disponibles en el directorio /bin/. Por defecto el servidor se encuentra disponible en el puerto 8080 pero si dicho puerto se encuentra ocupado se puede emplear otro indicándolo en el fichero /conf/server.xml. Se recomienda el uso del entorno CentOS. No obstante, Tomcat está desarrollado en Java, por tanto, es multiplataforma.

## Instalación de una aplicación Web

Para instalar una aplicación Web en un servidor de aplicaciones Web conforme a JEE es necesario empaquetarla (crear un fichero WAR). Los ficheros WAR se crean de forma análoga a los ficheros tar usando la siguiente orden: `jar -cvf aplicacionWeb.war directorioAplicacionAEmpaquetar`, y deben contener la siguiente estructura:

- `directorioAplicacionAEmpaquetar/WEB-INF/classes`: agrupa los ficheros `.class` que conforman la aplicación organizados en directorios según su estructura en paquetes. En esta primera práctica no tendrá contenido.
- `directorioAplicacionAEmpaquetar/WEB-INF/lib`: agrupa los ficheros `.jar` de librerías que usa la aplicación. En esta primera práctica tampoco tendrá contenido.
- `directorioAplicacionAEmpaquetar/WEB-INF/web.xml`: es el fichero de configuración estándar de la aplicación web.
- `directorioAplicacionAEmpaquetar/`: vista de la aplicación (por ejemplo: ficheros HTML, imágenes, ficheros CSS, ficheros JSP, ...) ordenados y estructurados en directorios según el criterio de sus creadores.

El contenido del directorio WEB-INF sólo será visible al servidor de aplicaciones Web, mientras que la vista será visible a los navegadores que accedan al servidor. El fichero `web.xml` tiene una estructura determinada y permite expresar un gran número de opciones de configuración (la mayoría opcionales). A continuación, se muestra el contenido de un fichero de este tipo:

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

<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

  <description> Hola Mundo </description>
  <display-name> Prueba </display-name>
</web-app>
```

La etiqueta ***display-name*** define el nombre de la aplicación web. La etiqueta ***description*** especifica una pequeña descripción del objetivo de la aplicación. Una vez empaquetada la aplicación en el fichero WAR, este se debe incluir en el directorio ***webapps*** del servidor de aplicaciones. Cuando se arranca el servidor, este desempaquetará el fichero WAR (instalando de este modo la aplicación creada para que los usuarios puedan acceder a ella). Cada usuario que desee acceder a la aplicación debe teclear en el navegador la dirección correspondiente (por ejemplo, `http://localhost:8080/nombreAplicacion`).

Una vez instalado y arrancado Tomcat, se puede acceder a una interfaz Web para administrarlo en la dirección <http://localhost:8080>. En la sección *Developer Quick Start* que se muestra al acceder a esa interfaz, se puede obtener documentación de cómo crear una aplicación Web en el enlace *First Web Application*. Ahí podemos encontrar información sobre la estructura de un fichero de aplicación .war en el enlace *Deployment Organization* de la sección *Table of Content*. Además, también se encuentra disponible un ejemplo de aplicación Web empaquetada en un fichero War en el enlace *Example Application*. Tomcat también proporciona una aplicación Web de administración de las aplicaciones instaladas en el servidor donde estas se pueden parar, arrancar, desplegar (o instalar), desplegar (o desinstalar), etc. Para acceder a dicha aplicación Web basta teclear <http://localhost:8080/Admin> en el navegador.

### 3. Entrega de la práctica

La práctica se realizará en grupos de tres personas. Cuando se finalice la práctica se debe entregar en un fichero denominado *práctica1\_NIP1\_NIP2\_NIP3.tar* o *práctica1\_NIP1\_NIP2\_NIP3.zip* (donde *NIP1*, *NIP2* y *NIP3* son los NIPs de los autores de la práctica) con el siguiente contenido:

1. Un fichero de texto denominado *autores.txt* que contendrá el NIP, los apellidos y el nombre de los autores de la práctica en las primeras líneas del fichero. Por ejemplo:

NIA	Apellidos	Nombre
-----		
545689	Rodríguez Quintela	Sabela
745689	Caamaño Cives	Antón
677893	López Marín	Adrián

2. Un fichero War (o equivalente) a instalar en un servidor de aplicaciones web correspondiente al esqueleto de aplicación que se ha creado y conforme a JEE (o equivalente).

3. Un fichero pdf denominado *MemoriaPractica1.pdf* cuya extensión no excederá de 10 páginas donde se indique la metodología de trabajo empleada para el desarrollo de la práctica (recursos, herramientas utilizadas, distribución del trabajo, las decisiones tomadas, planificación, distribución de tareas, etc.) y las principales dificultades encontradas durante la realización de esta.

Al descomprimir el fichero .tar o .zip, se deben extraer los ficheros y directorios indicados en el directorio *practica 1 Nip1 Nip2 y Nip3*. Es importante seguir las

convenciones de nombrado y la estructura de ficheros y directorios descrita.

Para la entrega del fichero .tar o .zip, se utilizará Moodle 2 del Anillo Digital Docente de la Universidad de Zaragoza. La fecha límite de entrega es el día anterior al de la siguiente sesión de prácticas en el laboratorio a las 23:59.

#### **4. Criterios de corrección**

Una vez realizadas las prácticas y entregadas estas, cada grupo debe presentárselas al profesorado de prácticas en la siguiente sesión de prácticas. Al realizar la presentación el profesorado le formulará cuestiones sobre las decisiones de diseño e implementación que ha realizado.

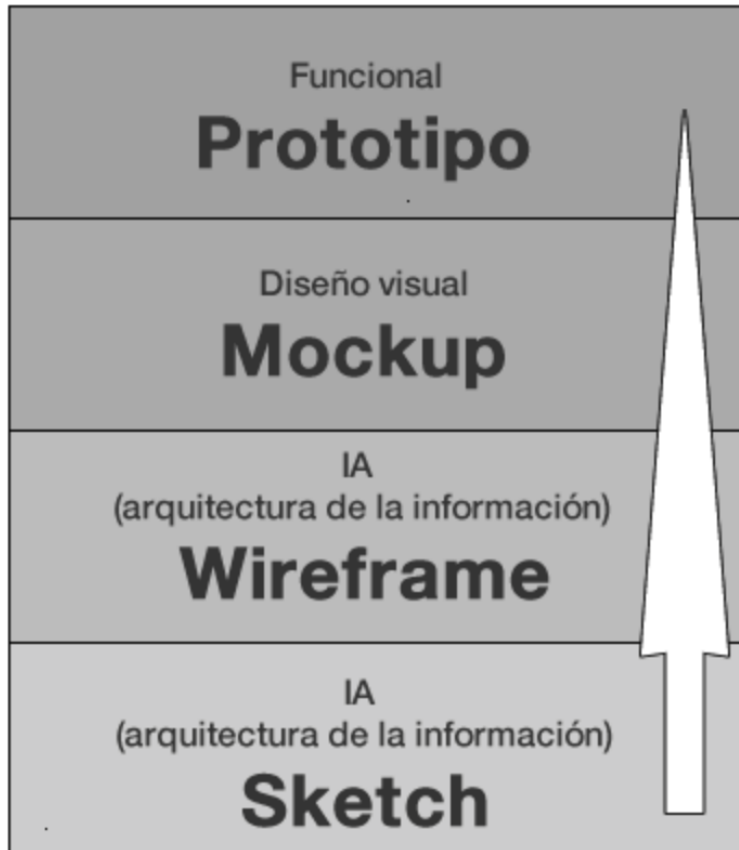
La práctica debe entregarse en los términos indicados anteriormente, funcionar correctamente en cualquier ordenador con Java 1.7 (en caso de que se haya optado por emplear un servidor conforme a JEE como, por ejemplo, Apache Tomcat) y no haber sido copiada. También es importante someter código limpio (donde se ha evitado introducir mensajes de depuración y comentarios que no proporcionan información al usuario). Los criterios de corrección se dividen en dos bloques:

- *Requisitos mínimos a cumplir (5 puntos)*
  - Consideración de los requisitos (funcionalidades) especificados en el enunciado y en la práctica 0 (2,5 puntos).
  - Organización adecuada de contenidos y carpetas, y creación de un fichero War o equivalente (en caso de que se emplee un servidor Web que no sea conforme a JEE) correcto en estructura y contenido (1 punto).
  - Aspectos relacionados con la tecnología de construcción de la capa de vista del sistema de información web diseñado, por ejemplo, uso de html 5 en lugar de html 4.1 o versión anterior, etiquetas anidadas correctamente, etc. (1,5 puntos).
- *Otros requisitos (5 puntos)*
  - Aspectos relacionados con navegabilidad, usabilidad, accesibilidad y seguridad del sitio web (1,5 puntos).
  - Aspectos relacionados con la presentación del informe (estructura, redacción, puntuación, índice, referencias y bibliografía...), justificación de las afirmaciones recogidas en él e inclusión de un apartado metodología y costes (por ejemplo, en horas de trabajo de los miembros del equipo) para la elaboración de la práctica (1,5 puntos).

- o Originalidad, aspectos novedosos que se han considerado (2 puntos).

## ANEXOS:

<https://blog.prototypr.io/4-best-web-ui-mockup-tools-for-free-89a1513c3fcd>



<https://www.iebschool.com/blog/herramientas-prototipado-analitica-usabilidad/>

<https://mosaic.uoc.edu/2015/09/15/proceso-de-desarrollo-de-un-proyecto-digital/>

<https://www.woorank.com/en/blog/building-responsive-websites-grids-and-wireframes>

<http://mpiua.invid.udl.cat/herramientas-de-prototipado-cual-escoger/>

<https://www.creativebloq.com/wireframes/top-wireframing-tools-11121302>

<https://www.lancetalent.com/blog/top-3-herramientas-de-prototipado-web-aplicaciones/>

<https://www.1and1.es/digitalguide/paginas-web/creacion-de-paginas-web/las-mejores->

[mockup-y-wireframe-tools/](#)