

Programación Web

Tema 1.1: Introducción

Ramón Alcarria
Miguel Ángel Manso
Emerson Castañeda
ETSI en Topografía, Geodesia y Cartografía - UPM

Contenido

- ¿Qué es una aplicación web?
- Recursos pasivos y activos
- Aplicaciones Web y el *Web Application Server*

¿Qué es una aplicación web? (I)

Aplicación web: aquellas que pueden utilizar los usuarios, mediante un navegador, accediendo a un servidor web a través de Internet o de una intranet

Esta formada por componentes Web que realizan tareas específicas y ofrecen servicios para sean utilizados desde la web

De ésta manera se incrementa el alcance de las aplicaciones y se disminuyen los problemas de versiones y actualizaciones

¿Qué es una aplicación web? (II)

En la modalidad cliente-servidor de escritorio, **cada aplicación** tiene su **propio programa cliente** (que hace de interfaz de usuario) y tiene que ser **instalado** por separado **en cada ordenador**. El cliente realiza peticiones al programa servidor que le da respuesta

Una mejora en el servidor, requiere habitualmente **una actualización de los clientes instalados**, añadiendo un coste de soporte técnico, disminuyendo la productividad

¿Qué es una aplicación web? (II)

Las aplicaciones web generan dinámicamente una serie de páginas con formatos estándar (**HTML** o **XHTML**), soportados por los visores (browsers)

Se utilizan lenguajes interpretados en el lado del cliente, directamente o a través de *plugins*, tales como *JavaScript*, *Java*, *Flash*, etc., para añadir elementos dinámicos a la interfaz de usuario

Generalmente cada página web se envía al cliente como un documento estático, la comunicación entre páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador web interpreta y muestra en pantalla las páginas.

¿Qué es una aplicación web? (III)

Un ejemplo de aplicación web en lenguaje Java es el *servlet*

El **servlet** *HelloServlet* es una demostración completa y funcional, devuelve una página HTML con un mensaje "Hello".

¿Qué es una aplicación web? (IV)

```
1 import java.io.*;
2 import javax.servlet.*;
3 import javax.servlet.http.*;
4
5 /** Simple servlet used to test server.
6  * <P>
7  * Taken from Core Servlets and JavaServer Pages 2nd Edition
8  * from Prentice Hall and Sun Microsystems Press,
9  * http://www.coreservlets.com/.
10  * &copy; 2003 Marty Hall; may be freely used or adapted.
11  */
12
13 public class HelloServlet extends HttpServlet {
14
15     public void doGet(HttpServletRequest request,
16     HttpServletResponse response) throws ServletException, IOException {
17         response.setContentType("text/html");
18         PrintWriter out = response.getWriter();
19         String docType = "<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 " +
20             "Transitional//EN">\n";
21         out.println(docType +
22             "<html>\n" +
23             "<head><TITLE>Hello</TITLE></head>\n" +
24             "<body>\n" +
25             "<h1Hello</h1>\n" +
26             "</body></html>");
27     }
28 }
29
```

¿Qué es una aplicación Web? (V)

En la practica, una aplicación, consiste en múltiples *Servlets*, paginas JSP, ficheros HTML, imágenes, etc.

Todos estos componentes se coordinan unos con otros para proporcionar un conjunto completo de servicios para los usuarios

Estructura de una Apl. web. (I)

Una aplicación web está normalmente estructurada como una aplicación de tres-capas:

- El navegador web ofrece la primera capa
- Como segunda capa, un motor capaz de usar alguna tecnología web dinámica (ejemplo: PHP, Java Servlets, ASP, ASP.NET, CGI, ColdFusion, Perl, Python, o Ruby on Rails) que constituye la capa intermedia
- Finalmente, una base de datos constituye la tercera y última capa

El navegador web envía peticiones a la capa intermedia que ofrece servicios y realiza consultas y/o actualizaciones a una base de datos.

Recursos pasivos y activos (I)

Un recurso es pasivo cuando no implica ningún procesamiento en sí mismo; es activo cuando cuenta con capacidades propias para el procesamiento

Por ejemplo, cuando un navegador envía una petición `www.myserver.com/myfile.html`, el servidor web `myserver.com` buscará el fichero `myfile.html`, un recurso pasivo, y lo retornará al navegador

Los recursos pasivos también se llaman recursos estáticos, ya que su contenido no cambia con las solicitudes

Recursos pasivos y activos (II)

De forma similar, cuando un navegador envía una petición `www.myserver.com/reportServlet`, el servidor web `myserver.com` enviará una solicitud al *servlet reportServlet*, un recurso activo. El *Servlet* generara el texto HTML y lo entregará al servidor Web. El servidor Web, a su vez, se lo envía al navegador

Una aplicación Web, usualmente es una mezcla de recursos activos y pasivos, si bien los recursos activos son los que la hacen realmente interactiva

Los recursos activos usualmente proporcionan contenido dinámico a los usuarios y habilitan el acceso a la lógica de negocio mediante un navegador

Aplicaciones Web y el Web Application Server (I)

Una aplicación web reside en un *Web Application Server* (Servidor de Aplicaciones)

El servidor de aplicaciones *sirve* las aplicaciones web y administra el acceso a los recursos del sistema

También proporciona servicios de bajo nivel, como la implementación del protocolo HTTP y administración de las conexiones con las bases de datos

http://en.wikipedia.org/wiki/Comparison_of_application_servers

Aplicaciones Web y el Web Application Server (II)

Un contenedor de *servlets* es una parte del *application server*.

Un *servlet container*, puede proveer otros componentes J2EE, como un contenedor de EJB, un servidor JNDI, y un servidor JMS

Ejemplos de servlet containers:

- Sun GlassFish
- Jboss
- Apache Tomcat
- Jetty

Temas Relacionados

Software as a Service (SaaS)

Web 2.0

Web Services

Web Widgets