

Comercio Electrónico

Practica 3: Infraestructuras Web



Antonio Sanz
ansanz@unizar.es

Rafael del Hoyo
rdelhoyo@ita.es

Objetivo de la práctica

Definir los diversos elementos de una infraestructura de comercio electrónico, y familiarizarse con los aspectos básicos de configuración y manejo. Se tratarán aspectos como servidores web, bases de datos y servidores de FTP.

¿Qué hay preparar de forma previa a la práctica?

En un principio no hay que llevar nada previo a la práctica, pero sería recomendable el tener frescos algunos conceptos de protocolos de comunicaciones. Saber lo que es una dirección IP, un puerto, y cómo se establece una conexión TCP será importante para la comprensión total de la misma. Dado que gran parte de la práctica se va a realizar en un shell de Windows, sería interesante el saber cómo manejarse en este entorno (cambiar de directorio, copiar y renombrar archivos, etc.).

¿Cuál es el resultado de la práctica?

El resultado final es una infraestructura completa sobre la cual podemos empezar a desarrollar de forma real el proyecto de comercio electrónico. Dentro del trabajo total se deberán entregar de forma electrónica los ficheros que se requieren en cada apartado.

¿Qué se aprende con esta práctica?

Se aprende a instalar y configurar de forma funcional los diversos elementos que puede tener un proyecto de comercio electrónico a nivel de infraestructura básica, es decir: servidor web, bases de datos y transferencia de la información.

Introducción

Una vez definidos en la práctica anterior todos los aspectos de diseño de nuestra proyecto de comercio electrónico se plantea la necesidad de comenzar la implementación del proyecto.

Todas los proyectos de comercio electrónico están compuestas de al menos cuatro elementos básicos:

- Una forma de presentar la información.
- Una forma de almacenar la información.
- Una forma de tratar la información.
- Una forma de transferir información.

La forma más común de presentar la información es mediante un **Servidor Web**. Dicho servidor tiene unos contenidos (o páginas) en su interior que muestra mediante el protocolo HTTP. Dicho protocolo es interpretado por los navegadores para mostrar la información al usuario. Los servidores web más empleados actualmente son Apache (disponible en <http://httpd.apache.org/>) e IIS (Internet Information Server), disponible con cualquier versión de servidor de Windows.

El almacenamiento de la información es otro factor crucial de la infraestructura técnica. El método más usual de almacenar la información se realiza a través de una **Base de Datos**. Una base de datos almacena la información deseada y la entrega de forma rápida y coherente, evitando pérdidas de información y ofreciendo una serie de servicios adicionales al desarrollador). Algunas de las bases de datos más empleadas son MySQL (<http://www.mysql.com/>), Oracle (<http://www.oracle.com/>) y SQL Server, disponible para Windows en <http://www.microsoft.com/spain/sql/downloads/default.msp> (es posible descargar una versión reducida pero gratuita de SQL Server desde la siguiente URL: <http://www.microsoft.com/sql/downloads/default.msp>).

Teniendo ya un medio para almacenar la información y para mostrarla, lo único que hace falta es disponer de un medio de tratar la información de forma que podamos mostrarla a nuestro antojo (por ejemplo, mediante un motor de búsquedas). Dichos servicios se realizan mediante un **Lenguaje de Programación**. Actualmente existen muchos lenguajes de programación, pero si nos centramos en los “orientados a web” (con grandes facilidades de acceso a bases de datos y de generación de código HTML), los dos más importantes son PHP (<http://www.php.net>) y (para Windows) ASP (<http://msdn.microsoft.com/asp.net>). El lenguaje PHP se verá con más detenimiento en prácticas posteriores.

Estos tres puntos forman el soporte principal de una infraestructura de comercio electrónico. Sin embargo, existen otros muchos **Servicios de Apoyo** que ofrecen funcionalidades muy interesantes, como pueden ser:

- Servicios de transferencia de ficheros (para mover contenidos en nuestra página web).
- Clientes gráficos de bases de datos (para acceder de forma remota a nuestra base de datos).
- Servicios de correo electrónico (para enviar correos electrónicos a nuestros usuarios).

Durante la práctica se va a emplear un paquete integrado denominado XAMPP (Windows, Apache, MySQL, Perl y PHP) que puede ser descargado de forma gratuita desde: <http://www.apachefriends.org/en/xampp-windows.html>

Servidor Web: Apache

El servidor web Apache es a día de hoy el más empleado para servir páginas web en todo el mundo (pueden verse las estadísticas de cuota de mercado de los principales servidores en Netcraft) :

http://news.netcraft.com/archives/web_server_survey.html

Una de las ventajas principales de Apache (aparte de ser *open source*, por lo que su coste es nulo, y se dispone de su código fuente para realizar modificaciones al gusto) es su disponibilidad tanto para entornos Linux como Windows. La estructura de directorios es la misma, y los ficheros de configuración solo difieren en los *path* empleados (C:\xampp\apache\datos frente a /opt/apache/datos, por ejemplo).

La instalación de XAMPP no tiene misterio alguno (tan solo se tendrá que tener cuidado de no instalar los componentes como servicios, ya que no disponemos de privilegios de administrador en el equipo). XAMPP tiene un cómodo panel de control que nos permite arrancar de forma manual los diversos componentes disponibles. Una vez esté arrancado de forma exitosa, puede ser accedido mediante un navegador en la dirección: <http://localhost>

La estructura de directorios de Apache es la siguiente:

- Bin: Almacena los binarios del programa.
- Conf: Localiza los ficheros de configuración del Apache.
- Error: En este directorio se guardan las respuestas a los distintos mensajes de error (como por ejemplo, el típico "Error 404: Not Found" que aparece cuando una página web concreta no existe).
- Icons: Centraliza los iconos e imágenes del programa
- Logs: Almacena los logs de acceso y error del Apache.
- Modules: Guarda los distintos módulos que Apache carga al arrancar.
- Htdocs: En este directorio se guardan de forma inicial las páginas web que muestra nuestro servidor (situado en un nivel superior).
- Cgi-bin: Directorio utilizado para guardar los programas que tengan que tener permisos de ejecución (típicamente, los cgi escritos en C u otros programas). Situado en un nivel superior.

Configuración de Apache: httpd.conf

Todos los aspectos de configuración de Apache son controlados por un fichero de configuración denominado *httpd.conf*. Vamos a tratar las opciones más interesantes del mismo:

Server Root

Indica el directorio raíz del que colgarán el resto de directorios del fichero de configuración

ServerRoot "D:/xampp/apache"

Listen

Especifica los puertos en los que va a estar escuchando el Apache. El puerto estándar es el 80, pero es posible colocarlo en el que deseemos (8080 es una posibilidad que se usa por problemas de permisos frecuentemente).

Listen 80

Configuración del PHP dentro del Apache

Esta línea indica que todos los ficheros con dichas extensiones sean pasados al intérprete de PHP en lugar de al navegador (es decir, que sean ejecutados).

AddType application/x-httpd-php .php .php4 .php3 .phtml

Alias

Especifican atajos para acceder a un directorio

Alias /phpmyadmin "D:/xampp/phpmyadmin/"

ServerAdmin

Especifica la dirección de correo electrónico a usar en caso de querer contactar con el administrador de la página

ServerAdmin admin@localhost

ServerName

Especifica el nombre DNS de la página web

ServerName localhost

DocumentRoot

Especifica el directorio inicial de nuestras páginas web. Un consejo es localizarlo fuera del ámbito del servidor web (para tener una separación clara entre “páginas” y “programa”, y que una actualización del software no borre sin querer nuestra web...).

DocumentRoot "D:/web/htdocs"

Logs

La primera directiva indica el nivel de log que se toma de la actividad del servidor, mientras que las dos siguientes indican la localización del log de errores y del de acceso a la web (este último se usa para generar las estadísticas del servidor).

```
LogLevel error
ErrorLog logs/error.log
CustomLog logs/access.log common
```

Virtual Hosts

Es frecuente que se emplee un único equipo para dar servicio a más de un proyecto de comercio electrónico o página web. Para ofrecer esa multiplicidad de servicio se recurre a lo que se denominan “hosts virtuales”. En la configuración de Apache será necesario crear una sección diferente para cada nombre de dominio (también puede hacerse por direcciones IP, pero es más complicado).

```
NameVirtualHost *

<VirtualHost *>
  ServerName www.domain.tld
  DocumentRoot /www/domain
</VirtualHost>

<VirtualHost *>
  ServerName www.otherdomain.tld
  DocumentRoot /www/otherdomain
</VirtualHost>
```

La configuración de SSL (Secure Socket Layer) que ofrece seguridad en las transacciones, se verá en prácticas posteriores.

Bases de datos: MySQL + SQL + PHPMyAdmin

MySQL es una base de datos potente y de gran rapidez, muy empleada en entornos orientados a páginas web al integrarse con facilidad con Apache y PHP. Una de las ventajas inherentes es que está disponible tanto en entornos Windows como Linux/Unix, siendo la configuración altamente parecida.

La información en MySQL está almacenada en tres niveles:

- Base de datos: El conjunto global de la información. Es posible tener una base de datos para cada empresa o departamento.
- Tabla: Dentro de una base de datos propia, existen varias tablas, cada una con información relacionada (personal, ventas, discos, etc.).
- Campo: Cada tabla tiene una serie de campos que almacenan la información necesaria (nombre, DNI, teléfono, etc.).

Es muy importante el realizar un diseño adecuado de la información para que las búsquedas sean lo más rápidas posibles (es decir, poner los datos que se tengan que recoger de una sola vez en una tabla, para no tener que realizar búsquedas múltiples).

La estructura de directorios de MySQL es la siguiente (dentro de %XAMPP%):

- /bin: Contiene los binarios de cliente y administrador de la base de datos, así como un script que lanza el servicio.
- /lib: Contiene las librerías dinámicas de MySQL (que son necesarias para el PHP).
- /data: Contiene tanto los mensajes de error como las propias bases de datos de MySQL.

MySQL puede arrancarse y detenerse mediante los scripts existentes en el directorio de %XAMPP%, aunque también puede instalarse como servicio de Windows mediante el script "mysql_install_service.bat" situado en el directorio "mysql". Es MUY importante detener MySQL mediante los scripts, o mediante el comando "mysqladmin shutdown" para evitar la pérdida de datos de nuestra base.

Es posible acceder a la base de datos en modo texto mediante el comando *mysql* (para ello previamente accederemos a Inicio→Ejecutar→cmd para lanzar una ventana de DOS, y nos colocaremos en %XAMPP%\mysql\bin).

Algunas de las órdenes más básicas del cliente son las siguientes:

- Show databases: Muestra las bases de datos disponibles.
- Use *database*: Selecciona una base de datos para su uso posterior.
- Drop *database*: Borra una base de datos, eliminando toda la información.
- Show tables: Muestra las tablas de una base de datos.
- Describe *tabla*: Describe los campos de una tabla.
- Quit: Sale del programa

SQL 101

El SQL (Simple Query Language) es un lenguaje de bases de datos adoptado por prácticamente todas las bases de datos actuales. Aunque cada base de datos tiene sus propias particularidades y funcionalidades propias, el conjunto de órdenes básicas se sigue manteniendo como un estándar. Algunos de los comandos básicos de SQL son:

- Query o consulta estándar: comando SELECT

```
SELECT * FROM table ;  
SELECT * FROM table WHERE nombre="Juan";  
SELECT nombre, telefono FROM table ;
```

- Crear tablas o bases de datos : comando CREATE

```
CREATE database nombre;  
CREATE TABLE clientes (name VARCHAR(64), empresa  
VARCHAR(64), mail VARCHAR(20) );
```

- Borrar tables o bases de datos: comando DROP

```
DROP database nombre;  
DROP table nombre ;
```

- Insertar datos en una base de datos: comando INSERT

```
INSERT INTO tabla (nombre, ciudad) VALUES('jose', 'zaragoza');
```

- Borrar datos de una base de datos: comando DELETE

```
DELETE FROM tabla where campo='dato';
```

- Modificar datos: comando UPDATE

```
UPDATE tabla SET campos= 'dato1' WHERE campob='dato2';
```

- Permisos sobre la base de datos: comando GRANT y REVOKE

```
GRANT ALL ON database.* TO username;  
REVOKE ALL ON database.* FROM username;  
FLUSH PRIVILEGES; (para aplicar los cambios)
```

Pueden encontrarse más comandos y más detalles en la referencia del lenguaje, en:

<http://dev.mysql.com/doc/>

Algunas de las tareas de administración más comunes pueden ser:

- Cambiar la contraseña de root:

```
UPDATE user SET Password= PASSWORD('new_password') WHERE  
user='root';  
FLUSH PRIVILEGES;
```


- Añadir un Nuevo usuario a la BBDD:

```
INSERT INTO user VALUES
('localhost','USUARIO',' ','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y');
```

- Cargar datos a partir de un fichero: Insertar los datos a razón de una entrada por línea, en el mismo orden el que se creó la tabla y separados por tabuladores. En el caso de que un campo sea nulo, se simboliza con \N (también tabulado):

```
LOAD DATA LOCAL INFILE "datos.txt" INTO TABLE clientes;
```

- Ejecutar conjuntos de comandos: Un comando por línea, terminado en “;”, grabado en un fichero de texto. Se ejecuta el comando mediante:

```
C:\mysql < fichero_comandos
```

- Realizar copias de seguridad: Cada base de datos tiene un directorio con todos sus ficheros, bajo el subdirectorio %XAMPP%\mysql\data. La copia de seguridad es tan sencilla como copiar ese directorio.

PHPMYAdmin

Dado que gran parte de las tareas de administración son costosas de realizar mediante línea de comandos, existe la posibilidad de emplear un interface vía web para realizar la gestión de una base de datos (solo si tenemos un servidor web en la misma máquina). Dicho interface se denomina PHPMYAdmin, y puede ser accedido mediante:

<http://localhost/phpmyadmin>

Mantenimiento de contenidos: Servicio FTP

Una vez establecida la página web, es necesario definir un mecanismo mediante el cual se puedan modificar los contenidos de la infraestructura de comercio electrónico (introducir productos en el catálogo, poner nuevas noticias, etc ...).

El mecanismo más empleado para dicha tarea es el servicio de transferencia de ficheros o FTP (File Transfer Protocol), aunque hay otras formas de transferir información (como puede ser directamente mediante HTTP o mediante la función de “copia segura” de SSH empleando SCP).

FTP es un protocolo cliente/servidor, por lo que para poder emplearlo necesitaremos en el equipo cuyos contenidos queramos gestionar un servidor de FTP. El que vamos a emplear bajo Windows en este caso es el **FileZilla FTP Server**, un servidor de FTP *open source* que aunque está incluido en XAMPP puede ser descargado de forma independiente desde :

<http://sourceforge.net/projects/filezilla/>

Bajo entornos Linux puede ser usado el ProFTPD (<http://www.proftpd.org/>) con características muy similares.

Para instalar de forma exitosa el FileZilla, tan solo será necesario hacer que se ejecute como servicio de Windows. Bastará con seleccionar la carpeta de FileZillaFTP, y ejecutar **FileZillaServer.exe**, seleccionando las opciones de “Instalar Servicio” y “Iniciar al arranque” (serán necesarios privilegios de administrador, como hemos indicado con anterioridad).

Para arrancarlo pulsaremos sobre la tarea “FileZillaFTP_start.bat”, y para apagarlo sobre “FileZillaFTP_stop.bat”. Una vez arrancado el servicio, podemos hacer pruebas con cualquier cliente de FTP para verificar su correcto funcionamiento. Se arrancará un cliente de FTP de Windows en modo texto mediante “Inicio → Ejecutar → cmd”, y tecleando:

```
ftp localhost
```

Será necesario introducir un nombre de usuario y una contraseña. FTP permite un modo de acceso denominado “anonymous FTP”, por el cual es posible acceder a una parte muy restringida del servidor sin introducir una contraseña (muy útil para establecer una zona de descargas pública). Se introducirá pues “anonymous” como nombre de usuario, y contraseña en blanco.

Algunos comandos básicos del FTP son los siguientes:

- ls: Lista los ficheros de un directorio
- cd *directorio*: Permite acceder al directorio
- cd..: Subimos un directorio hacia arriba
- pwd: Especifica la ruta de directorios
- !ls: Lista los ficheros del directorio de la máquina cliente
- BIN: Activa el modo de transferencia binario
- HASH: Se muestra el proceso de descarga mediante una barra de “#”
- GET *fichero*: Descarga el fichero indicado del servidor
- PUT *fichero*: Coloca el fichero indicado en el servidor
- QUIT: Cierra la sesión

Puede verse que es posible descargar ficheros del servidor, pero no se puede escribir más que en el directorio “incoming”, debido a la estructura de permisos del servidor. De la misma forma, por mucho que subamos en la estructura de directorios, no vamos a poder ver el resto de ficheros del sistema. Esto es debido a que el servidor de FTP “enjaula” al usuario dentro de una estructura acotada para que no pueda salir de ella (mediante lo que se llama “chrooting”, técnica muy útil y empleada con frecuencia en los servidores de FTP).

Para cambiar la configuración del servidor de FTP accederemos al programa “**FileZilla Server Interface**”, a la sección de “Edit → Settings/Users/Groups”, que nos permitirá modificar varios parámetros de la configuración del mismo (así como ver la actividad de nuestro servidor de FTP).

Trabajo a realizar durante la práctica

- Entregar un fichero de configuración *httpd.conf* en el que se resalten los cambios en la configuración para nuestro proyecto de comercio electrónico.
- Diseñar la estructura de datos de al menos **dos** tablas de datos del proyecto, teniendo en cuenta el nombre y el tipo de dato de cada variable.

Se deberá entregar el resultado en forma de fichero que pueda ser cargado de forma automática en la base de datos (es decir, una concatenación de sentencias SQL).

- Modificar la configuración del servidor de FTP para que:
 - o Presente un mensaje de bienvenida adecuado.
 - o Permita el acceso únicamente desde una subred de clase C.
 - o Guarde registros o logs de la actividad.
 - o Existan dos usuarios: imágenes y ftp. El primero sólo tiene derechos sobre las carpetas de imágenes, y el resto sobre todos los ficheros. Los directorios de entrada y los permisos deberán estar ajustados al máximo posible.

Se deberán entregar los ficheros **FileZilla Server.xml** y **\Logs\ FileZilla Server.log** junto con la explicación de la configuración del servidor.