

Práctica 1.3

Realización de un servidor web

CURSO 2021-2022

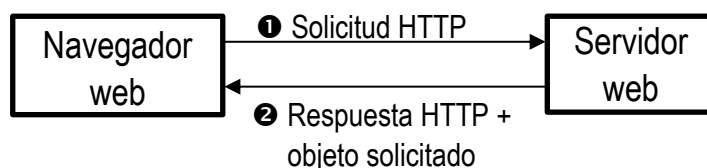
Objetivos

El objetivo de la práctica es el desarrollo de una aplicación que funcione como un servidor de páginas web que sea capaz de, a partir de una solicitud HTTP, dar una respuesta HTTP seguida de la página web solicitada. Para la correcta realización de la práctica se requieren, además de los conocimientos adquiridos hasta el momento acerca de la programación de *sockets*, los conocimientos adquiridos acerca del protocolo HTTP adquiridos en la asignatura Arquitectura de Redes I.

Desarrollo de la práctica

HTTP es el protocolo principal de la capa de aplicación del servicio web, basado en el modelo cliente-servidor, y concebido para la transferencia de recursos (objetos). Dichos recursos pueden ser archivos contenidos en repositorios de archivos y/o el resultado de la ejecución de un programa, el resultado de la consulta a una base de datos, etc.

La práctica consiste en la realización de un programa escrito en Python que hace de servidor HTTP y que es capaz de atender múltiples conexiones TCP simultáneamente por medio de la función *select* y el uso de *sockets*. El funcionamiento básico de un servidor web en el caso de que no se produzcan errores consiste en escuchar las solicitudes HTTP que hacen los navegadores web en un determinado puerto y contestarlas por medio de una respuesta HTTP seguida del objeto que ha sido solicitado. Los objetos que devuelve el servidor HTML son archivos que estarán almacenados en el propio servidor y que pueden ser archivos HTML, imágenes,...



Un ejemplo de solicitud HTTP tras haber pedido en el navegador la página de www.google.es puede ser el siguiente:

```
GET http://www.google.es/ HTTP/1.1
Host: www.google.es
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; es-ES; rv:1.8.1.2)
Gecko/20070219 Firefox/2.0.0.2
Accept:text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
Accept-Language: es-es,es;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive
```

Tras recibir estas líneas de texto del navegador (el final de la línea está fijado por los caracteres `\r\n`), el servidor web debe contestar a dicha solicitud mediante (en este orden):

- Línea de estado que contiene la versión del protocolo HTTP (por ejemplo, versión 1.0), el código de respuesta (por ejemplo 200) y una palabra/frase explicativa.
- Una serie de cabeceras HTTP.
- Una línea en blanco.

- Objeto solicitado, en caso de que exista.

Un ejemplo de respuesta, donde se ha omitido el contenido del objeto HTML que contiene es:

```
HTTP/1.0 200 OK
Date: Friday, 6-May-11 15:40:00 GMT
Server: Apache/1.1.1
Content-type: text/html
Content-length: 450
Connection: close

<HTML><HEAD><TITLE> ..... </HTML>
```

Con el fin de optimizar el funcionamiento del servidor web, se pide que se pueda procesar más de una petición a la vez, para ello se deberá utilizar la función *select*. **Además, programe el servidor de manera que, cuando esté 10 segundos inactivo, muestre por pantalla el mensaje “Servidor web inactivo”.**

Plazo para la realización de la práctica y evaluación

Para la realización de esta práctica dispone, además del tiempo de trabajo personal que requiera cada alumno, de **tres** sesiones de dos horas cada una las semanas 8, 9 y 11 de la asignatura.

La evaluación de la práctica se hará teniendo en cuenta:

- Presentación de memoria de la práctica donde debe constar el código fuente **comentado** de los archivos desarrollados. Dicha memoria debe enviarse con fecha límite el día anterior a la sesión 12 mediante el Aula Virtual de la asignatura. Se valorarán positivamente las funcionalidades adicionales o avanzadas que haya programado el estudiante. **El código fuente debe incorporarse a la memoria como texto, nunca como imagen.**
- Presentación oral donde se muestre y defienda la solución planteada en la semana 12 de la asignatura. En dicha presentación el profesor o el resto de estudiantes podrán hacer las preguntas que estimen pertinentes.

Para poder superar la práctica con un 50% de su calificación, el servidor web deberá ser capaz de servir al navegador, y que este la visualice correctamente, una página web sencilla que conste de un archivo HTML y una imagen. Además, debe poder configurarse el servidor para trabajar tanto en modo persistente como no persistente. Para obtener una calificación superior al 50% debe implementar funcionalidades más avanzadas como, por ejemplo, el tratamiento de métodos diferentes a GET, el envío de páginas web más complejas, manejo de diferentes códigos HTTP, uso de HTTPS, manejo de cabeceras de idiomas...

NOTA IMPORTANTE: el servidor web debe realizarse por el alumno mediante la programación de sockets y el uso de la función *select*, no siendo válido el uso de módulos en Python como *http.server* o *SimpleHTTPServer*. El uso de dichos módulos no permitirá superar la práctica.

La honestidad en el trabajo personal o de grupo debe ser una característica fundamental de un estudiante universitario. Por este motivo, se confía en que las memorias, programas,... que se presenten en la asignatura solo serán realizados por sus personas responsables y firmantes. La ruptura de esta confianza implicará que los alumnos y alumnas afectados no puedan superar la asignatura.

Referencias

- Documentación oficial de Python: <http://www.python.org/doc/>
- RFC 2616: R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach y T. Berners-Lee, “Hypertext Transfer Protocol – HTTP/1.1,” junio 1999. (<http://tools.ietf.org/html/rfc2616>)