

Cómo Funciona un Servidor Web

Francisco Javier Otero Herrero

Grupo ATU

26-3-2025

Cómo Funciona un Servidor Web

Cómo Funciona un Servidor Web

Contenido

<i>SERVIDOR WEB, PROTOCOLO HTTP</i>	<i>3</i>
<i>Actividades</i>	<i>3</i>

Cómo Funciona un Servidor Web

SERVIDOR WEB, PROTOCOLO HTTP

Supongamos que escribimos en el navegador de internet la siguiente URL:

<http://www.cursolnet.es/MF1271/es/index.html>

Actividades

1) ¿En qué partes se descompone esta dirección URL?

Una **URL (Uniform Resource Locator)** es una dirección que permite localizar un recurso en la web. La URL proporcionada Se puede descomponer en las siguientes partes:

Esquema o protocolo (http):

- Indica el protocolo que se utilizará para acceder al recurso.
- En este caso, **http** significa que se usará el protocolo **HTTP (HyperText Transfer Protocol)**.

Nombre de dominio (www.cursolnet.es):

- Es el nombre del servidor donde se encuentra el recurso.
- Este nombre se traduce a una dirección IP mediante un **DNS (Domain Name System)**.

Ruta del recurso (/MF1271/es/index.html):

- Indica la ubicación específica del recurso dentro del servidor.
- Se divide en:
 - /MF1271: Directorio principal.
 - /es: Subdirectorio dentro de /MF1271.
 - /index.html: Archivo HTML que se solicita.

Por lo tanto, la URL completa se interpreta como:

"Utiliza el protocolo HTTP para acceder al archivo index.html, ubicado en el directorio /MF1271/es/ del servidor www.cursolnet.es."

Cómo Funciona un Servidor Web

Adjuntamos la tabla de la descomposición de la URL que nos facilitaron para esta actividad:

<i>Componente</i>	<i>Ejemplo</i>	<i>Descripción</i>
<i>Protocolo</i>	<i>http://</i>	<i>Indica el protocolo de comunicación</i>
<i>Subdominio</i>	<i>www.</i>	<i>Subdominio del servidor(opcional)</i>
<i>Dominio</i>	<i>Cursolnet.es</i>	<i>Nombre del servidor</i>
<i>Ruta del recurso</i>	<i>/MF1271/es/index.html</i>	<i>Ubicación del archivo en el servidor</i>

2) Supongamos que, en la anterior ruta, la página contuviera referencias a 5 imágenes. ¿Cuál sería la secuencia de comunicación entre nuestro cliente y el servidor?

Cuando accedes a una página web que contiene múltiples recursos (como imágenes), el navegador realiza varias solicitudes HTTP para obtener todos los elementos necesarios. La comunicación sigue estos pasos:

✓ **Paso 1: Resolución DNS**

El navegador consulta el DNS para convertir www.cursolnet.es en una dirección IP.

✓ **Paso 2: Solicitud HTTP inicial (HTML)**

El navegador envía una petición HTTP GET al servidor para:

`GET /MF1271/es/index.html HTTP/1.1`

Host: www.cursolnet.es

Si todo ha ido bien, entonces el servidor responde con el **código 200 OK** y el contenido de **index.html**.

✓ **Paso 3: Análisis del HTML y descarga de imágenes**

El navegador analiza el HTML y detecta las 5 imágenes, por ejemplo:

``

Para cada imagen, el navegador realiza una nueva petición HTTP GET (en paralelo o secuencial, según la configuración):

`GET /MF1271/es/imagen1.jpg HTTP/1.1`

Host: www.cursolnet.es

(Se repite para imagen2.jpg, imagen3.jpg, etc.).

Cómo Funciona un Servidor Web

✓ **Paso 4: Respuestas del servidor (imágenes)**

Si todo ha ido bien, el servidor responde con **200 OK** y los datos binarios de cada imagen (si existen).

✓ **Paso 5: Renderizado final**

El navegador ensambla el HTML con las imágenes y muestra la página completa.

- **En resumen, la secuencia de comunicación sería:**

- i. *El navegador solicita el archivo HTML principal (index.html) al servidor.*
- ii. *El servidor responde con el archivo HTML.*
- iii. *El navegador analiza el HTML y detecta referencias a 5 imágenes.*
- iv. *El navegador envía 5 solicitudes HTTP GET adicionales, una por cada imagen.*
- v. *El servidor responde con el contenido de cada imagen.*
- vi. *El navegador renderiza la página completa con el HTML y las imágenes.*

- **Cuestiones a tener en cuenta:**

- **Paralelismo:** Los navegadores modernos suelen enviar varias solicitudes simultáneamente (en paralelo) para mejorar el rendimiento.
- **Caché:** Si algunas imágenes ya están almacenadas en caché en el navegador, no se enviarán nuevas solicitudes para esos recursos.
- **Errores:** Si alguna imagen no está disponible, el servidor responderá con un código de estado de error (por ejemplo, 404 Not Found).
- **Protocolo:** Si fuera **HTTPS**, se añadiría un **handshake TLS** antes de las peticiones.