

2º DAW/Despliegue de aplicaciones web

UNIDAD 2

Arquitectura Web y Protocolos



Autor: Luz María Álvarez

Fecha: 16/10/2025

Sesión 3 – Arquitectura Cliente-Servidor y Protocolo

HTTP

La arquitectura cliente-servidor es el modelo básico sobre el que se sustenta Internet.

El **cliente** (por ejemplo, un navegador o aplicación móvil) **realiza peticiones**, y el **servidor responde** con los recursos solicitados.

HTTP (HyperText Transfer Protocol) regula este intercambio de información mediante solicitudes (requests) y respuestas (responses).

1. Arquitectura Cliente-Servidor

El **modelo cliente-servidor** divide las tareas entre dos entidades:

- **Cliente:** realiza solicitudes de servicios o recursos.
- **Servidor:** procesa las solicitudes y envía las respuestas.

Ejemplo: Cuando escribes www.spotify.com en el navegador, el cliente solicita una página y el servidor la devuelve.

2. Evolución del Protocolo HTTP

- **HTTP/1.1 (1997):** Utiliza múltiples conexiones TCP, lo que ralentiza la carga.
- **HTTP/2 (2015):** Introduce multiplexing (varios recursos por una conexión), compresión y priorización.
- **HTTP/3 (2020→):** Basado en QUIC/UDP, optimizado para redes móviles con menor latencia.

Comparativa de Versiones HTTP

Versión	Año	Características	Mejoras
HTTP/1.1	1997	Múltiples conexiones TCP	Estandarización inicial
HTTP/2	2015	Multiplexing, compresión	Menor latencia, mayor eficiencia
HTTP/3	2020→	Basado en QUIC/UDP	Velocidad y fiabilidad en móviles

4. Caso de Estudio: YouTube y la transición a HTTP/2 y HTTP/3

YouTube enfrentaba problemas de latencia en la carga de vídeos.

Estrategia: Google implementó HTTP/2 para reducir la sobrecarga mediante multiplexing y priorización de recursos críticos.

Posteriormente, adoptó HTTP/3 sobre QUIC para mejorar el rendimiento en redes móviles.

Resultado: Reducción del tiempo de inicio en un 30%, mejora de la experiencia de usuario y menor buffering.

Mitos y Realidades sobre HTTP

- Mito: 'HTTP es inseguro por sí mismo.'
 - FALSO: con HTTPS y TLS, el tráfico está cifrado.
- Mito: 'HTTP/3 solo sirve para móviles.'
 - FALSO: también mejora entornos empresariales con alto volumen de tráfico.
 -

Resumen Final

- **Cliente-servidor** = cliente pide, servidor responde.
- **HTTP** regula la comunicación web.
- **HTTP/1.1**: múltiples conexiones.
- **HTTP/2**: multiplexing y compresión.
- **HTTP/3**: QUIC/UDP, ideal para móviles.

Tabla de códigos de estado HTTP más comunes

Código	Significado	Descripción
100	Continue	El servidor espera que el cliente continúe con la petición.
101	Switching Protocols	Cambio de protocolo solicitado por el cliente.
200	OK	Petición exitosa.
201	Created	Recurso creado correctamente.
204	No Content	Petición exitosa sin contenido.
301	Moved Permanently	Recurso movido permanentemente.
302	Found	Redirección temporal.
304	Not Modified	El recurso no ha cambiado.
400	Bad Request	Petición mal formada.

401	Unauthorized	Autenticación requerida.
403	Forbidden	Acceso denegado.
404	Not Found	Recurso no encontrado.
500	Internal Server Error	Error interno del servidor.
502	Bad Gateway	El servidor intermedio recibió una respuesta inválida.
503	Service Unavailable	Servidor no disponible.
504	Gateway Timeout	Tiempo de espera agotado.

Preguntas para reflexionar

1. ¿Qué diferencia hay entre un código 301 y un 302?
2. ¿Por qué los navegadores siguen automáticamente las redirecciones (opción -L)?
3. ¿Qué información útil nos ofrecen las cabeceras Server, Content-Type y Date?
4. ¿Por qué es importante interpretar correctamente los códigos 4xx y 5xx en desarrollo web?

