

Capítulo 1



Protocolo HTTP

Estadísticas



- Browser, resoluciones, sistemas operativos
 - <https://gs.statcounter.com/>
 - <https://www.w3counter.com/globalstats.php>
- Host, páginas web, servidores web
 - <http://news.netcraft.com/>



- HTTP
 - Protocolo de transferencia de hipertexto
 - Define que mensajes pueden enviar los clientes a los servidores y que respuestas obtienen
 - RFC 822: Solicitud ASCII seguida de respuesta MIME
 - Todos los clientes y servidores deben obedecer este protocolo
 - Conexiones
 - TCP en el puerto 80 de la máquina servidor
 - Ni los navegadores ni servidores se preocupan por los mensajes largos, perdidos o duplicados, ni confirmaciones
 - HTTP1.0: se enviaba la solicitud, se recibe respuesta
 - HTTP1.1: conexiones persistentes



- HTTP
 - Métodos
 - Soporta métodos diferentes a solicitar una página web
 - El método GET solicita al servidor que envíe la página
 - Método HEAD solicita el encabezado del mensaje
 - Método PUT inverso de GET
 - Cada solicitud obtiene una respuesta con un código de estado
 - Códigos 2xx indica que se manejó con éxito la solicitud
 - Códigos 3xx indican al cliente que busque en otro lado
 - Códigos 4xx indica que la solicitud falló por error del cliente
 - Códigos 5xx indican que el servidor falló

HTTP



Method	Description
GET	Request to read a Web page
HEAD	Request to read a Web page's header
PUT	Request to store a Web page
POST	Append to a named resource (e.g., a Web page)
DELETE	Remove the Web page
TRACE	Echo the incoming request
CONNECT	Reserved for future use
OPTIONS	Query certain options

Los métodos de solicitudes HTTP

HTTP



Code	Meaning	Examples
1xx	Information	100 = server agrees to handle client's request
2xx	Success	200 = request succeeded; 204 = no content present
3xx	Redirection	301 = page moved; 304 = cached page still valid
4xx	Client error	403 = forbidden page; 404 = page not found
5xx	Server error	500 = internal server error; 503 = try again later

Grupos de respuesta del código de estado

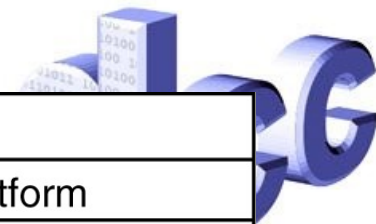
<https://httpstatusdogs.com/>

<https://http.cat/>

HTTP



- HTTP:
 - Encabezados del mensaje
 - A la línea de solicitud la pueden seguir líneas con más información (encabezados de solicitud)
 - Las respuestas pueden tener encabezados de respuesta
 - User-agent: informa al servidor el navegador que está usando el cliente
 - Con “Accept” indica lo que el cliente está dispuesto a aceptar
 - El servidor utiliza “Server” para indicar quien es y entregar algunas de sus propiedades
 - Con los “Content-*” el servidor describe propiedades de la página que está enviando
 - Con “Location” el servidor indica al cliente que debe tratar con una URL diferente



Header	Type	Contents
User-Agent	Request	Information about the browser and its platform
Accept	Request	The type of pages the client can handle
Accept-Charset	Request	The character sets that are acceptable to the client
Accept-Encoding	Request	The page encodings the client can handle
Accept-Language	Request	The natural languages the client can handle
Host	Request	The server's DNS name
Authorization	Request	A list of the client's credentials
Cookie	Request	Sends a previously set cookie back to the server
Date	Both	Date and time the message was sent
Upgrade	Both	The protocol the sender wants to switch to
Server	Response	Information about the server
Content-Encoding	Response	How the content is encoded (e.g., gzip)
Content-Language	Response	The natural language used in the page
Content-Length	Response	The page's length in bytes
Content-Type	Response	The page's MIME type
Last-Modified	Response	Time and date the page was last changed
Location	Response	A command to the client to send its request elsewhere
Accept-Ranges	Response	The server will accept byte range requests
Set-Cookie	Response	The server wants the client to save a cookie

URL



- URL
 - Indica la ubicación de un recurso
- URI
 - Mas nuevo que URL, identifica un recurso
 - No necesariamente, relacionado con su ubicación
- Formato:
 - protocolo://nombre-dns:puerto/identificador
- Ejemplo:
`http://www.dcc.uchile.cl/1877/channel.html`

URL



Name	Used for	Example
http	Hypertext (HTML)	<code>http://www.cs.vu.nl/~ast/</code>
ftp	FTP	<code>ftp://ftp.cs.vu.nl/pub/minix/README</code>
file	Local file	<code>file:///usr/suzanne/prog.c</code>
news	Newsgroup	<code>news:comp.os.minix</code>
news	News article	<code>news:AA0134223112@cs.utah.edu</code>
gopher	Gopher	<code>gopher://gopher.tc.umn.edu/11/Libraries</code>
mailto	Sending e-mail	<code>mailto:JohnUser@acm.org</code>
telnet	Remote login	<code>telnet://www.w3.org:80</code>

Algunos URLs comunes

Direccionamiento



- Distintas direcciones:
 - MAC Address
 - Dirección IP
 - Nombre DNS
- Contactar una aplicación se usa un puerto
 - Hay puertos estándares que las aplicaciones escuchan y conocen
 - Telnet (23), SMTP (25)
 - Por ejemplo en producción: HTTP (80), SSL (443), MYSQL (3306)
 - Por ejemplo en desarrollo: HTTP (8080), SSL (8443)