

Introducción a las aplicaciones Web

Carlos Zuluaga

Agenda

1. El protocolo HTTP.
2. Análisis con Firebug.

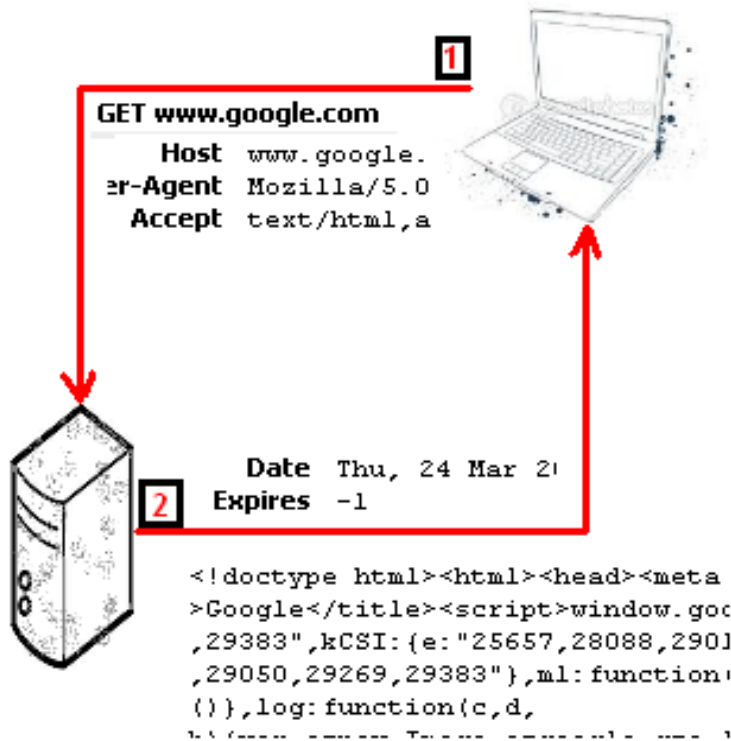
El Protocolo HTTP

1. Funciona sobre TCP/IP.
2. Protocolo sin estado (stateless).
3. Se puede usar para conectar un navegador con un servidor Web, dos aplicaciones distintas, un dispositivo móvil con una aplicación, etc.



http://

Dinámica del Protocolo



- El navegador o dispositivo abre un socket con el servidor y envía unos encabezados. Dependiendo de la operación envía o no contenido.
- El servidor responde con encabezados y contenido de acuerdo a la operación.
- El navegador hace peticiones para cada uno de los componentes de la página.

Operaciones

GET

Requests a representation of the specified resource. Requests using GET should only [retrieve data](#) and should have no other effect. (This is also true of some other HTTP methods.)^[1] The [W3C](#) has published guidance principles on this distinction, saying, "[Web application](#) design should be informed by the above principles, but also by the relevant limitations."^[11] See [safe methods](#) below.

POST

Submits data to be processed (e.g., from an [HTML form](#)) to the identified resource. The data is included in the body of the request. This may result in the creation of a new resource or the updates of existing resources or both.

PUT

Uploads a representation of the specified resource.

DELETE

Deletes the specified resource.

TRACE

Echoes back the received request, so that a client can see what (if any) changes or additions have been made by intermediate servers.

OPTIONS

Returns the HTTP methods that the server supports for specified [URL](#). This can be used to check the functionality of a web server by requesting `*/` instead of a specific resource.

CONNECT

Converts the request connection to a transparent [TCP/IP tunnel](#), usually to facilitate [SSL](#)-encrypted communication ([HTTPS](#)) through an unencrypted [HTTP proxy](#).^[12]

PATCH

Is used to apply partial modifications to a resource.^[13]

HTTP servers are required to implement at least the GET and HEAD methods^[14] and, whenever possible, also the OPTIONS method.^[citation needed]

Códigos de Respuesta

Table 1-3 Common Status-Codes and Reasons

Status-Code	Reason
100	Continue
200	OK
201	Created
300	Multiple Choices
301	Moved Permanently
302	Found
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
407	Proxy Authentication Required
408	Request Time-out
413	Request Entity Too Large
500	Internal Server Error
501	Not Implemented



HTTP por consola

Términos y Conceptos

- Sesión
- Request
- Cookie
- Recurso estático
- Recurso dinámico
- Javascript
- Servidor
- Navegador

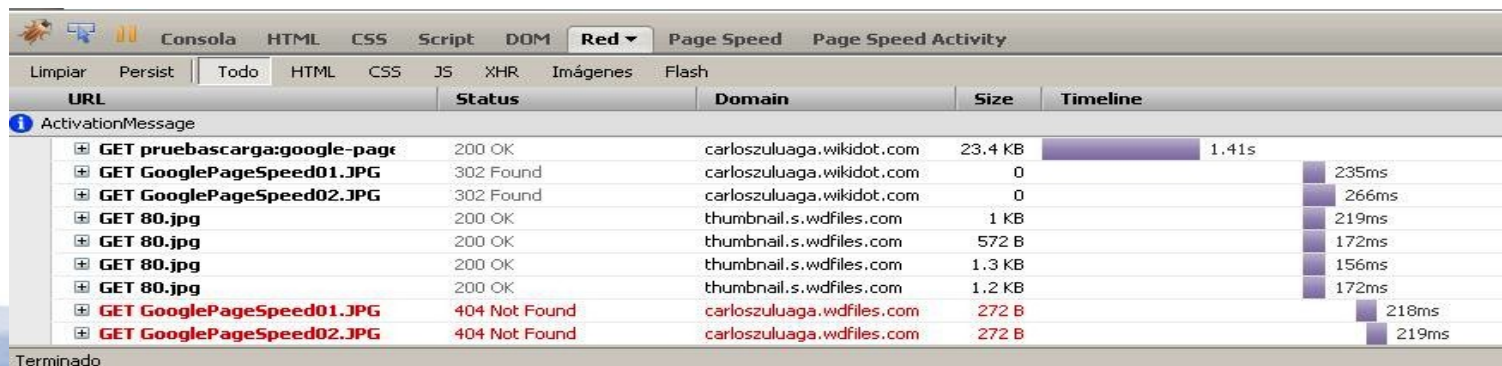


Análisis con Firebug

Firebug

Es un sofisticado plugin para Firefox o Chrome que permite analizar todos los componentes de una página Web:

- Llamados a diferentes recursos
- Encabezados
- Detalles de las peticiones y las respuestas



The screenshot shows the Firebug network tab with a table of requests. The table has columns for URL, Status, Domain, Size, and Timeline. The requests include a page load, several image requests, and two 404 Not Found errors.

URL	Status	Domain	Size	Timeline
ActivationMessage				
GET pruebascarga:google-page	200 OK	carloszuluaga.wikidot.com	23.4 KB	1.41s
GET GooglePageSpeed01.JPG	302 Found	carloszuluaga.wikidot.com	0	235ms
GET GooglePageSpeed02.JPG	302 Found	carloszuluaga.wikidot.com	0	266ms
GET 80.jpg	200 OK	thumbnail.s.wdfiles.com	1 KB	219ms
GET 80.jpg	200 OK	thumbnail.s.wdfiles.com	572 B	172ms
GET 80.jpg	200 OK	thumbnail.s.wdfiles.com	1.3 KB	156ms
GET 80.jpg	200 OK	thumbnail.s.wdfiles.com	1.2 KB	172ms
GET GooglePageSpeed01.JPG	404 Not Found	carloszuluaga.wdfiles.com	272 B	218ms
GET GooglePageSpeed02.JPG	404 Not Found	carloszuluaga.wdfiles.com	272 B	219ms

Terminado



Firebug