

# Ejercicio 1: HTTP

Este es un ejemplo de una solicitud (request) HTTP.

```
GET /collaborators HTTP/1.1
Host: wareware.bunsan.io
User Agent: Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en)
AppleWebKit/124
(KHTML, like Gecko) Safari/125
Accept: ext/xml, application/xml, application/xhtml+xml,
text/html;q=0.9,
text/plain;q=0.8, image/png,*,*;q=0.5
Accept-Language: it
Keep-Alive: 300
Connection: keep-alive
```

Responde a las siguientes preguntas, indicando dónde (por ejemplo, en qué campo) en la solicitud HTTP puedes encontrar la respuesta:

1) ¿Cuál es la URL solicitada?

```
GET /collaborators HTTP/1.1
Host: wareware.bunsan.io
User Agent: Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en)
AppleWebKit/124
(KHTML, like Gecko) Safari/125
Accept: ext/xml, application/xml, application/xhtml+xml,
text/html;q=0.9,
text/plain;q=0.8, image/png,*,*;q=0.5
Accept-Language: it
Keep-Alive: 300
Connection: keep-alive
```

2) ¿Qué versión de HTTP se utiliza?

```
GET /collaborators HTTP/1.1
Host: wareware.bunsan.io
User Agent: Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en)
AppleWebKit/124
(KHTML, like Gecko) Safari/125
Accept: ext/xml, application/xml, application/xhtml+xml,
text/html;q=0.9,
text/plain;q=0.8, image/png,*,*;q=0.5
Accept-Language: it
Keep-Alive: 300
Connection: keep-alive
```

### 3) ¿El navegador solicita una conexión persistente o no persistente?

Por default se solicita una conexión persistente que se cierra hasta después de una transacción (dado que se usa HTTP/1.1). En este caso Connection: keep-alive se utiliza para reforzar la request.

```
GET /collaborators HTTP/1.1
Host: wareware.bunsan.io
User Agent: Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en)
AppleWebKit/124
(KHTML, like Gecko) Safari/125
Accept: ext/xml, application/xml, application/xhtml+xml,
text/html;q=0.9,
text/plain;q=0.8, image/png,*,*;q=0.5
Accept-Language: it
Keep-Alive: 300
Connection: keep-alive
```

### 4) ¿Cuál es, en tu opinión, la utilidad de indicar el tipo (y versión) de navegador utilizado por el cliente en la solicitud HTTP?

Nos permite conocer las capacidades del navegador (si está actualizado u obsoleto) con el fin de adaptar la información de la respuesta.

```
GET /collaborators HTTP/1.1
Host: wareware.bunsan.io
User Agent: Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en)
AppleWebKit/124
(KHTML, like Gecko) Safari/125
Accept: ext/xml, application/xml, application/xhtml+xml,
text/html;q=0.9,
text/plain;q=0.8, image/png,*,*;q=0.5
Accept-Language: it
Keep-Alive: 300
Connection: keep-alive
```

## Ejercicio 2: HTTP

Un cliente HTTP envía el siguiente mensaje:

```
GET /assignments HTTP/1.1
Host: wareware.bunsan.io
User-agent: Mozilla/4.0
Accept: text/html, image/gif, image/jpeg
If-modified-since: 27 Feb 2017 08:10:00
```

**a) Escribe dos respuestas posibles del servidor HTTP (solo la línea de estado)**

HTTP/1.1 200 OK

HTTP/1.1 404 Not Found

**b) Suponiendo que el mensaje se envía a través de un Proxy, especifica el comportamiento del propio Proxy.**

El proxy actúa como un intermediario por lo que primero se debe de hacer el request al proxy y después éste hará el request al servidor. Ahora bien, la respuesta será de igual manera solo que en sentido inverso: del servidor al proxy y del proxy al cliente.