

## GRUPO: 8

### COMPONENTES:

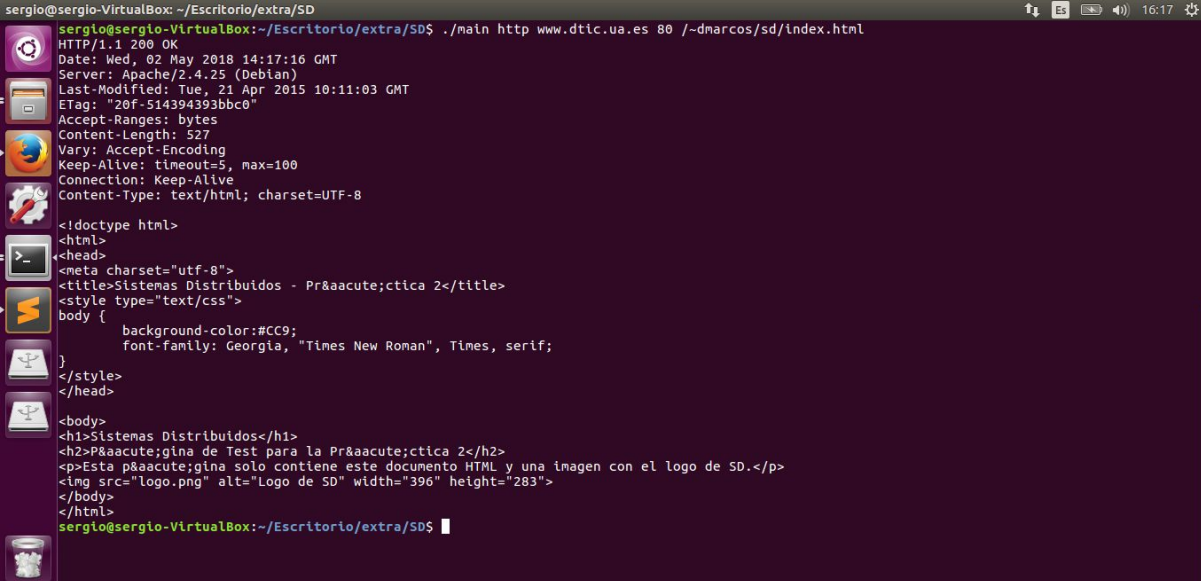
- Sergio Bañuls
- Sergio Campos
- Hawer Bonilla

### CLIENTE:

Nuestro cliente (mi\_wget), realiza peticiones GET, para ello debemos ejecutarlo en el terminal pasándole como parámetros la dirección IP o el nombre del servidor, el puerto de escucha y el recurso al cual queremos acceder.

El nombre del servidor lo capturamos gracias a la función gethostbyname(), a la cual hay que pasarle el nombre que introducimos al hacer la llamada. Así solo necesita saber su DNS para acceder a cualquier servidor.

Tras esto el programa realiza la petición GET y muestra por pantalla los resultados.



```
sergio@sergio-VirtualBox: ~/Escritorio/extra/SD
sergio@sergio-VirtualBox:~/Escritorio/extra/SD$ ./main http www.dtic.ua.es 80 /~dnarcos/sd/index.html
HTTP/1.1 200 OK
Date: Wed, 02 May 2018 14:17:16 GMT
Server: Apache/2.4.25 (Debian)
Last-Modified: Tue, 21 Apr 2015 10:11:03 GMT
Etag: "20f-514394393bbc0"
Accept-Ranges: bytes
Content-Length: 527
Vary: Accept-Encoding
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Sistemas Distribuidos - Práctica 2</title>
<style type="text/css">
body {
    background-color:#CC9;
    font-family: Georgia, "Times New Roman", Times, serif;
}
</style>
</head>
<body>
<h1>Sistemas Distribuidos</h1>
<h2>Página de Test para la Práctica 2</h2>
<p>Esta página solo contiene este documento HTML y una imagen con el logo de SD.</p>

</body>
</html>
sergio@sergio-VirtualBox:~/Escritorio/extra/SD$
```

## SERVIDOR:

El servidor, en principio, es capaz de interpretar 4 tipos de llamadas diferentes:

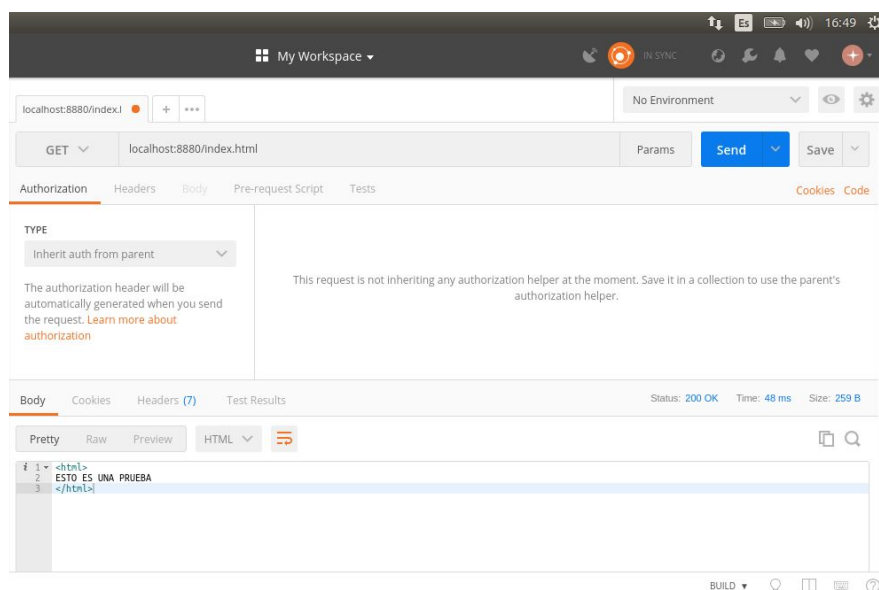
- Una sin introducir ningún parámetro en la cual podrá el puerto por defecto 8880 que está declarado en el código || `./mi_httpd`
- Otra será pasándole un puerto en concreto || `./mi_httpd puerto`
- La tercera forma será pasándole un archivo de configuración con la ruta (**DocumentRoot**), el nombre del recurso (**DirectoryIndex**), el número máximo de clientes(**MaxClients**) y el puerto de escucha (**Listen**). Para nuestra práctica nos hemos creado un recurso llamado "index.html" que contiene un código básico de HTML, y un archivo de configuración llamado "conf.conf" con los parámetros anteriormente mencionados. || `./mi_httpd -c archivo.conf`
- La última forma será pasándole un puerto en concreto más el archivo de configuración. || `./mi_httpd puerto -c archivo.conf`

Una vez se hayan establecido todos los parámetros del servidor, se realizará la apertura del socket de escucha y se pondrá a la espera de recibir peticiones.

Cuando se realiza una petición se crea un nuevo hijo mediante la función **fork()** y se cierra el socket principal.

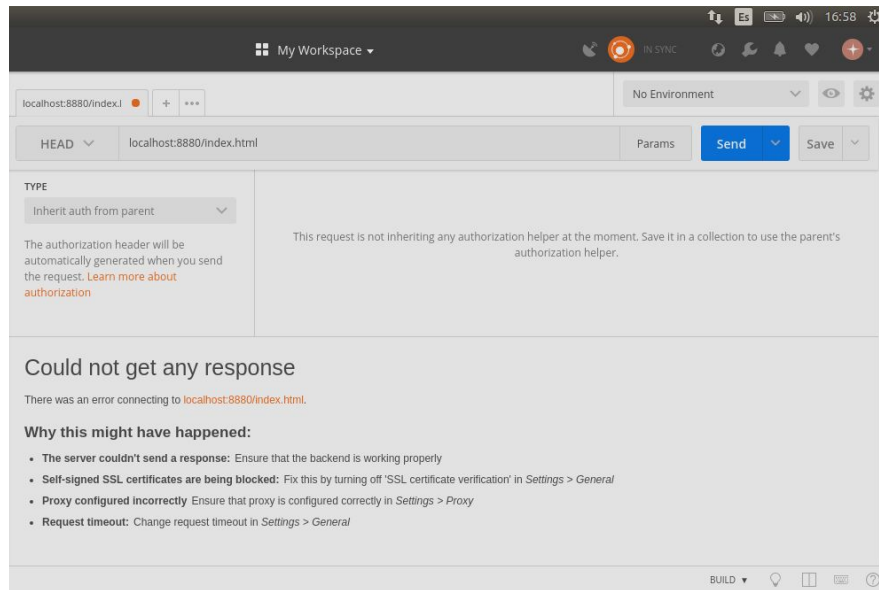
Ahora creamos el código necesario para cada petición (**GET**, **HEAD**, **PUT**, **DELETE**):

El **GET** nos devolverá el body del recurso solicitado más la cabecera, el **HEAD** nos devolverá la cabecera, el **PUT** subirá el recurso al servidor (documento en blanco), y el **DELETE** borrará el recurso.



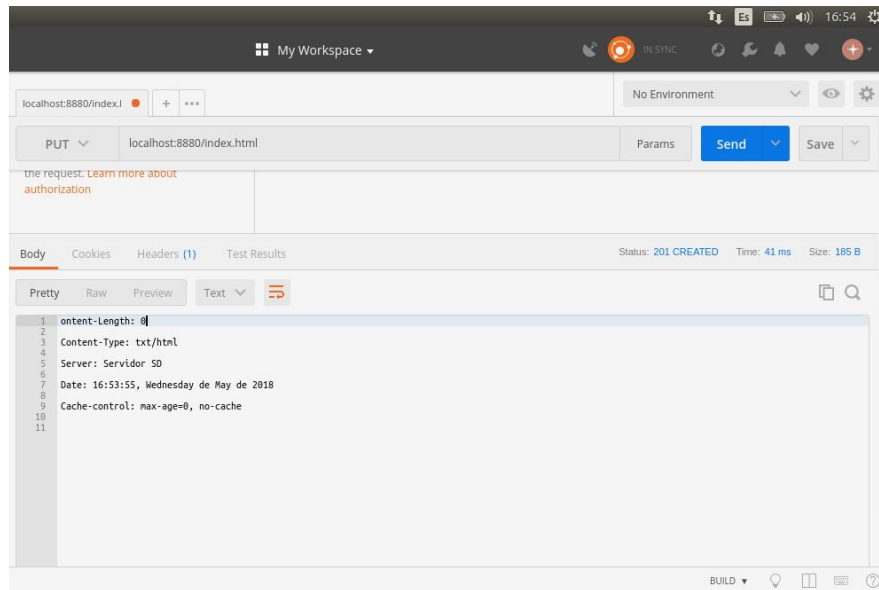
(GET)

El **GET** funciona correctamente.



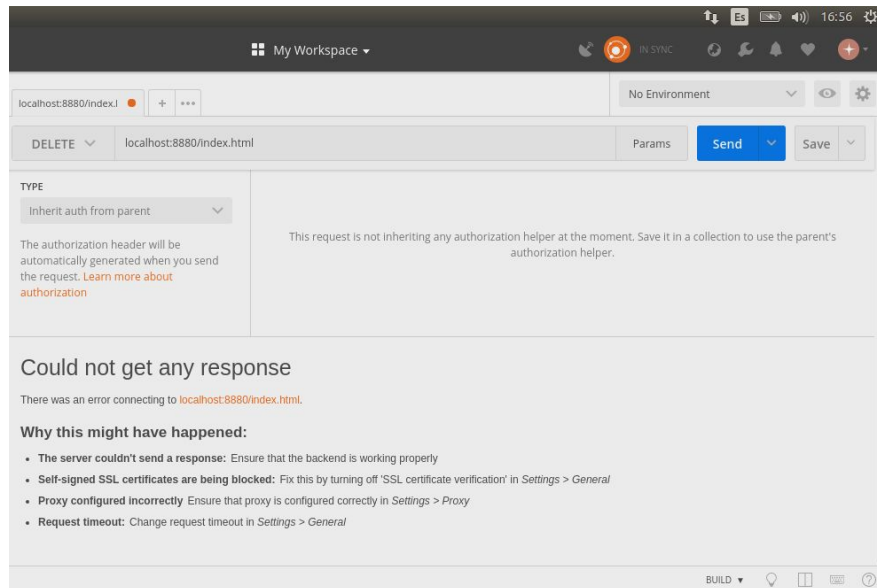
(HEAD)

El **HEAD** nos salta error de conexión, por motivos que desconocemos.



(PUT)

El **PUT**, como hemos comentado anteriormente, deja el recurso index.html en blanco.

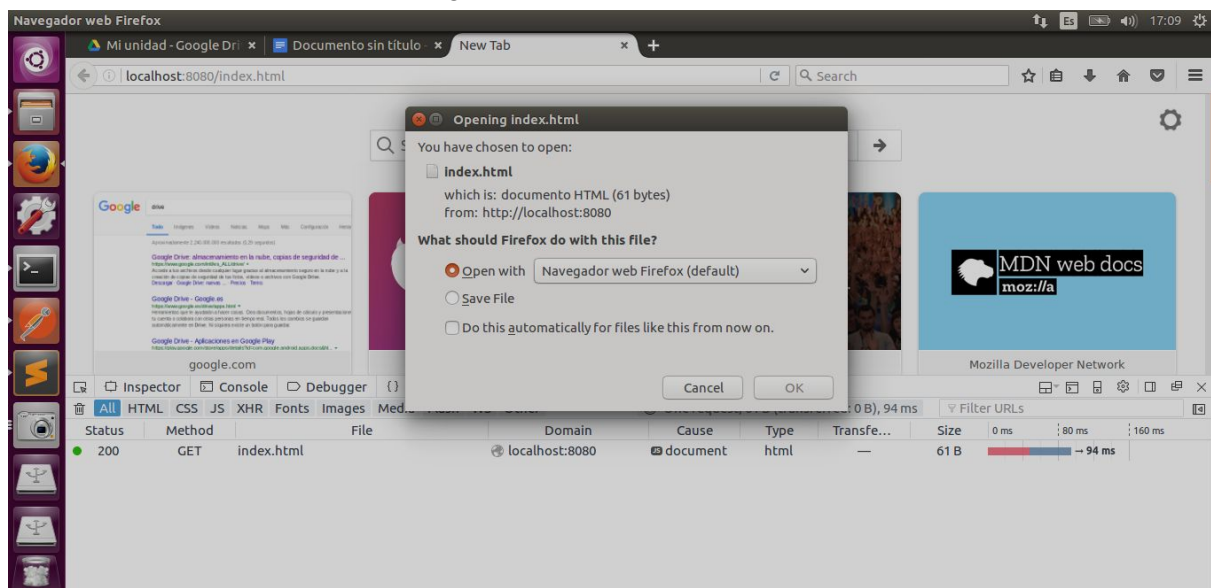


(DELETE)

En el **DELETE**, el Postman da un error de conexión, pero realmente si que borra el recurso index.html.

Hemos tenido un problema, que seguimos sin encontrarle solución, y es que a la hora de ejecutar el servidor, si no lo hacemos mediante un archivo de configuración, ninguna de las peticiones funciona. Siempre es necesario ejecutarlo como `./mi_httpd -c archivo.conf` o `./mi_httpd PUERTO -c archivo.conf`

La petición **GET** mediante el navegador funcionan correctamente



Por último hemos implementado una serie de mensajes de error que nos detallaran donde o con que, ha ocurrido el error.

- HTTP/1.1 404 not found
- HTTP/1.1 505 HTTP version not supported
- HTTP/1.1 403 Forbidden
- HTTP/1.1 405 Method not allowed