

Alumno/a: José Manuel González Peña

Práctica 4: Más sobre la configuración de un servicio web.

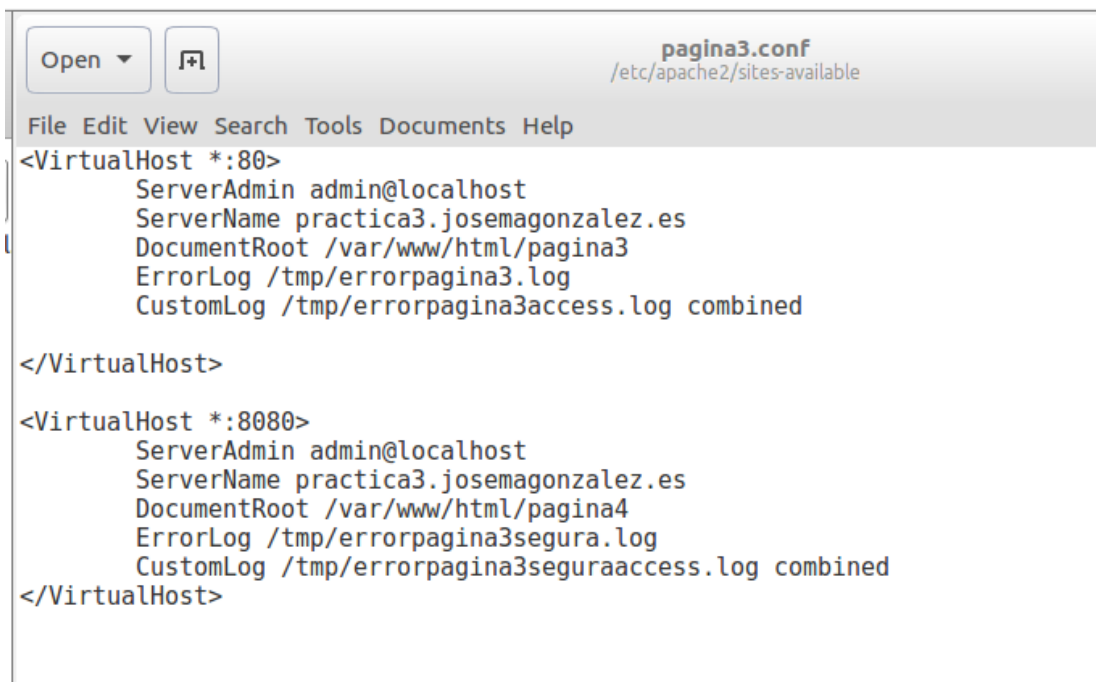
Objetivo:

En esta práctica vamos a probar otras opciones de configuración para nuestros servidores web virtuales creados en las prácticas anteriores.

Fuentes: http://es.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Versiones

Procedimiento:

1. Para facilitar detectar los errores en los distintos servidores virtuales, asigna un fichero de log diferente y personalizado para cada uno. Ayúdate de la "Guía de supervivencia de Apache" página 18.



```
Open [icon] pagina3.conf
/etc/apache2/sites-available

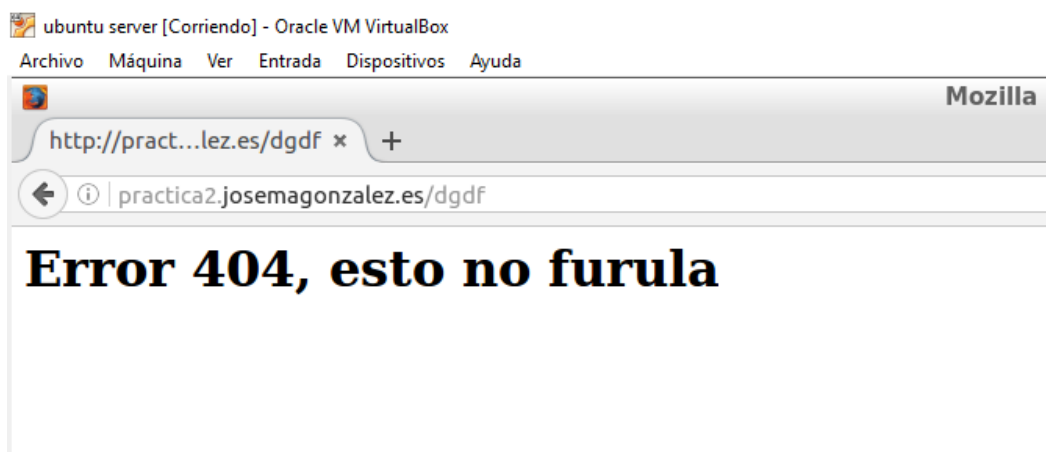
File Edit View Search Tools Documents Help

<VirtualHost *:80>
    ServerAdmin admin@localhost
    ServerName practica3.josemagonzalez.es
    DocumentRoot /var/www/html/pagina3
    ErrorLog /tmp/errorpagina3.log
    CustomLog /tmp/errorpagina3access.log combined
</VirtualHost>

<VirtualHost *:8080>
    ServerAdmin admin@localhost
    ServerName practica3.josemagonzalez.es
    DocumentRoot /var/www/html/pagina4
    ErrorLog /tmp/errorpagina3segura.log
    CustomLog /tmp/errorpagina3seguraaccess.log combined
</VirtualHost>
```

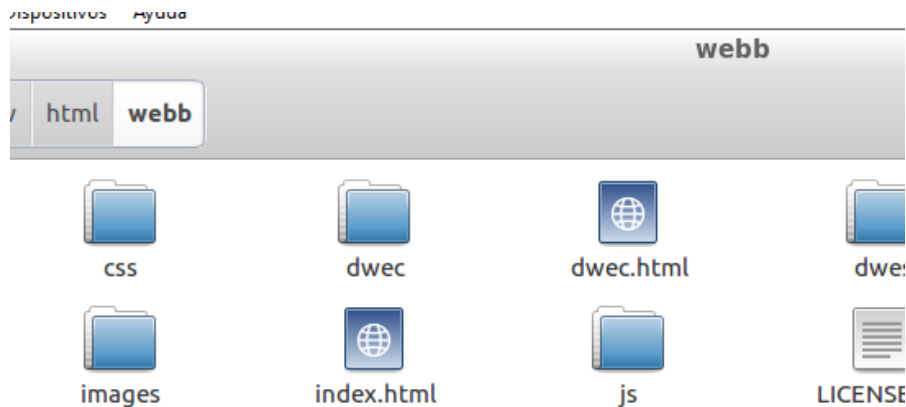
2. Configura la página de error 404 y muestra un mensaje personalizado.

<http://www.desarrolloweb.com/articulos/configurar-pagina-error-404-apache.html>



Lee el capítulo sobre Alias, Redirect y Rewrite (Guía Supervivencia Apache, pág 39).

a. - Asegurate de tener en tu servidor web virtual (el usado y corregido en la practica 2) un site completo con varias páginas, carpeta de imagenes ,etc.

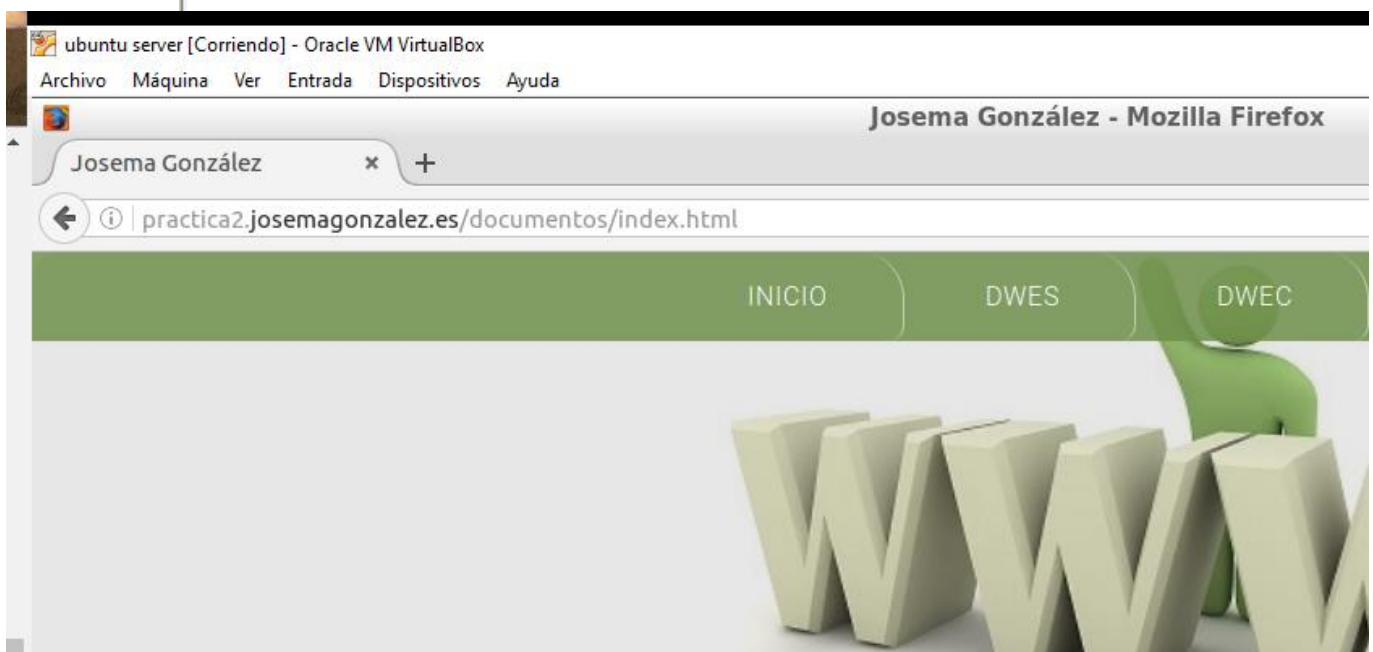


b.- Haz uso en tu site de archivos que estén fuera del DocumentRoot (p. ej. imagenes) Usa un alias para que esto sea posible.

http://httpd.apache.org/docs/2.2/mod/mod_alias.html

Nota: asegúrate que está activo el módulo mod_alias

```
<VirtualHost *:80>
    ServerAdmin admin@localhost
    DocumentRoot /var/www/html/sitio2
    ServerName practica2.josemagonzalez.es
    Alias /documentos /var/www/html/webb
    <Directory "/var/www/html/sitio2/carpeta">
        Options +Indexes -SymLinksIfOwnerMatch +Multiview
```



c.- Redireccionamiento de página: tipo temporal , permanente, etc. Explicalo.

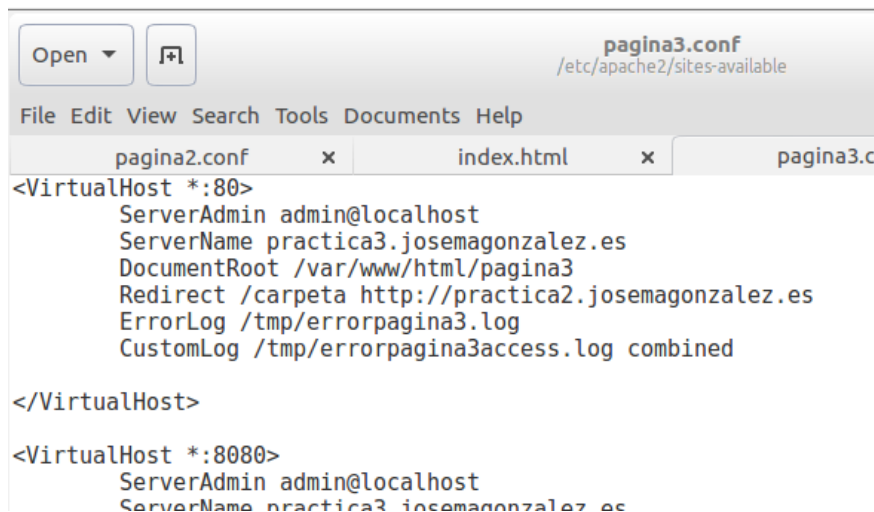
La directiva Redirect permite redirigir una URL a otra dirección nueva, permitiendo además añadir un argumento "status", que indica el tipo de redirección que se hará. El formato de la directiva es el siguiente:

Redirect [status] URLAntigua NuevaURL

El argumento status puede ser opcional, mientras que URLAntigua indica qué ruta, comenzando por '/' (ruta relativa al DocumentRoot) , será redireccionada a la dirección NuevaURL.

Status indica al navegador el tipo de redirección, y puede ser uno de los siguientes:

- *Permanent: devuelve un estado de redirección permanente (código 301), indicando que el recurso se ha movido permanentemente.*
- *Temp: devuelve un estado de redirección temporal (código 302). Este es el valor por defecto, por lo que si no se introduce ningún argumento en la directiva, se tomará este.*



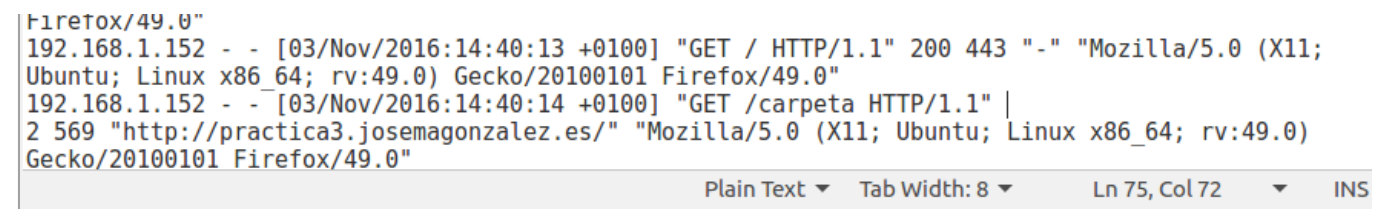
```

Open [icon] pagina3.conf /etc/apache2/sites-available
File Edit View Search Tools Documents Help
pagina2.conf x index.html x pagina3.c
<VirtualHost *:80>
    ServerAdmin admin@localhost
    ServerName practica3.josemagonzalez.es
    DocumentRoot /var/www/html/pagina3
    Redirect /carpeta http://practica2.josemagonzalez.es
    ErrorLog /tmp/errorpagina3.log
    CustomLog /tmp/errorpagina3access.log combined
</VirtualHost>

<VirtualHost *:8080>
    ServerAdmin admin@localhost
    ServerName practica3.josemagonzalez.es
  
```

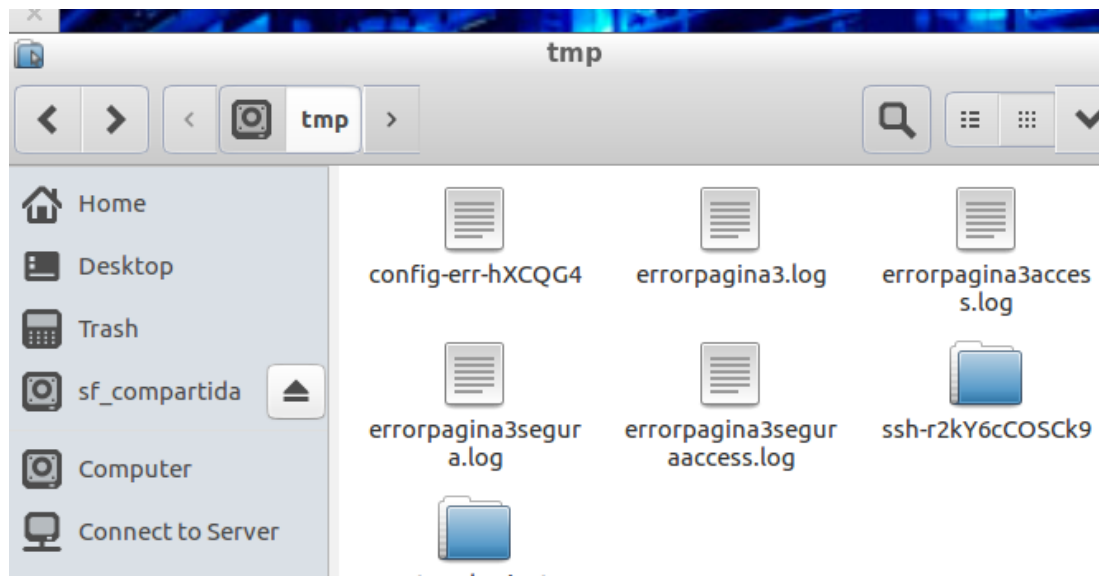
- *Seeother: indica que el recurso al que se pensaba acceder ha sido reemplazado (código 303).*
- *Gone: indica que el recurso ha sido eliminado permanentemente (código 410). Cuando se usa este valor, se debe omitir NuevaURL en la directiva.*

d.- Supon que temporalmente tu página está en obras y necesitas redireccionarla a otra que avise al usuario de la situación. Una vez realizado esto, ve al fichero de log y comprueba el código de error que envía. Captura pantalla indicando este código.



```

Firefox/49.0"
192.168.1.152 - - [03/Nov/2016:14:40:13 +0100] "GET / HTTP/1.1" 200 443 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:49.0) Gecko/20100101 Firefox/49.0"
192.168.1.152 - - [03/Nov/2016:14:40:14 +0100] "GET /carpeta HTTP/1.1" |
2 569 "http://practica3.josemagonzalez.es/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:49.0) Gecko/20100101 Firefox/49.0"
Plain Text Tab Width: 8 Ln 75, Col 72 INS
  
```



e.- **(PRACTICA OPCIONAL)** Rewrite: : usando RewriteRule para acortar una URL.

Nota: RewriteLog ya está obsoleto, no se permite generar un fichero de log para el Rewrite.

Ejemplo: Supongamos que dentro de nuestro DocumentRoot tenemos una carpeta búsqueda con un fichero buscar.php ([descargar](#)) cuyo contenido es:

<?

```
$pag=fopen("http://www.google.com/search?q=".$_GET["id"],"r");
```

```
if($pag)
```

```
{
```

```
$text=fpassthru($pag);
```

```
fclose($pag);
```

```
echo $text;
```

```
}
```

```
?>
```

1. Este fichero me permite obtener la página de búsqueda de google con el parámetro dado, de esta forma:

`http://localhost/busqueda/buscar.php?id=hola`

Nos gustaría poder crear una URL más corta que haga lo mismo, escribiríamos en nuestro .htaccess un RewriteRule de la siguiente forma:

```
RewriteRule ^buscar busqueda/buscar.php
```

De esta forma accederíamos por medio de la URL:

`http://localhost/buscar?id=hola`