

---

**SMIX M7**

**UF3: Servidores WEB**

**MARTA MORENO**

# ÍNDICE

---

8.1 Internet y Servicios web

Servidor Web

Cliente Web

8.2 Instalación IIS

8.3 Instalación de Apache en Windows

8.4 Instalación de Apache en Linux

# EVOLUCIÓN HISTÓRICA

---

❖ La Web nace al 1989 a partir de un proyecto del CERN (Consejo Europeo para la Investigación Nuclear), Tim Berners-Lee construye el prototipo que dio lugar a lo que hoy en día se conoce como www (World Wide Web).

La intención original era hacer más fácil compartir textos de investigación entre científicos y permitir al lector revisar las referencias de un artículo mientras lo iba leyendo.

Un sistema de hipertexto enlazaría todos los documentos entre si para que el lector pudiera revisar las referencias de un artículo mientras fuera leyendo

---

# ÍNDICE

---

❖ Inicialmente el programa del CERN, sólo consideraba texto. Viola de Pei Wei (1992) añadió la capacidad de presentar gráficos.

Marc Andreessen presentó un navegador web llamado "Mosaic" en 1993 que disparó la popularidad de la web.

Andreessen fundó Mosaic Communications Corporation (después Netscape) añadiendo características como contenido dinámico, música y animaciones.

La capacidad de los navegadores y servidores fue más rápida que la evolución de los estándares → Al inicio las nuevas funciones no estaban soportadas en todos los navegadores: IE, Mozilla Firefox, Opera, Safari, Amaya, Epiphany, Galeon, Konqueror, Lynx, Netscape Navigator

---

# EVOLUCIÓN HISTÓRICA

---

❖ La web se basa en tres estándares:

**El Localizador Uniforme de Recursos (URL)**, que especifica como cada página de información se asocia a una dirección única donde se puede localizar;

**El protocolo de Transferencia de Hipertexto (HTTP)**, que especifica cómo el navegador y el servidor intercambian información en forma de peticiones y respuestas.

**El lenguaje de Marcación de Hipertexto (HTML)**, un método para codificar la información de los documentos y sus enlaces. Berners-Lee dirige en la actualidad el World Wide Web Consortium, que desarrolla y mantiene este y otros estándares.

---

# INTERNET Y SERVICIOS WEB

---

- ❖ El servicio web de Internet se basa en la transmisión de páginas web.
- ❖ Las páginas web se encuentran almacenadas en un ordenador (o en varios) capaz de funcionar como un servidor web.
- ❖ El servicio Web funciona siguiendo el denominado modelo cliente - servidor: habitual en las aplicaciones que funcionan en una red.
  - ❖ Servidor → Quien presta el servicio.
  - ❖ Cliente → Quien lo recibe.

# SERVIDOR WEB

---

- ❖ Un servidor web recibe peticiones de clientes y responde con el envío de ficheros solicitados, texto plano (html, php) o binarios (gif, jpeg).
  - ❖ Permanentemente escucha las peticiones de conexión de los clientes en determinados puertos: 80 para HTTP, 443 para el HTTPS
  - ❖ La atención a la petición del cliente consiste en buscar el archivo solicitado. Si lo encuentra, lo transmite; sino envía un mensaje de error.
  - ❖ El servidor web comprueba si el usuario tiene acceso a los documentos.
-

# PROCESO TRANSFERENCIA

---

- ❖ El usuario especifica en el cliente web la URL de la pagina que desea consultar.
  - ❖ El cliente establece la conexión con el servidor web y solicita la pagina deseada.
  - ❖ El servidor busca la pagina solicitada en su sistema de ficheros. Si la encuentra la transfiere, sino devuelve un código de error.
  - ❖ El cliente interpreta el código HTML y muestra la pagina al usuario.
  - ❖ Se cierra la conexión. La conexión siempre se libera al terminar la transmisión de la pagina.
-



# MODELO TRANSACCIONAL

---

- ❖ Las aplicaciones cliente-servidor siguen el denominado modelo de transacciones: el cliente realiza una petición, el servidor la atiende e inmediatamente se cierra la comunicación.
- ❖ Se establece una transacción independiente para cada documento u objeto que se transmite.
- ❖ No se mantiene memoria entre las sucesivas peticiones (acceso banco, sitios seguros, etc)
- ❖ El uso de cookies o acceso a base de datos resuelve este problema

# LOCALIZACIÓN SERVIDORES WEB

---

- ❖ Servidores locales. Localizados en la misma red de área local (LAN) donde se encuentra el cliente. Por ejemplo: Servidor web intranet
- ❖ Servidores remotos. No está en la red de área local en la que se encuentra la máquina cliente. Por ejemplo: [www.google.com](http://www.google.com).
- ❖ Es posible conocer información sobre el servidor remoto mediante la utilización del servicio whois en [www.whois.sc](http://www.whois.sc): Información del propietario, su localización, etc

# CLIENTE WEB

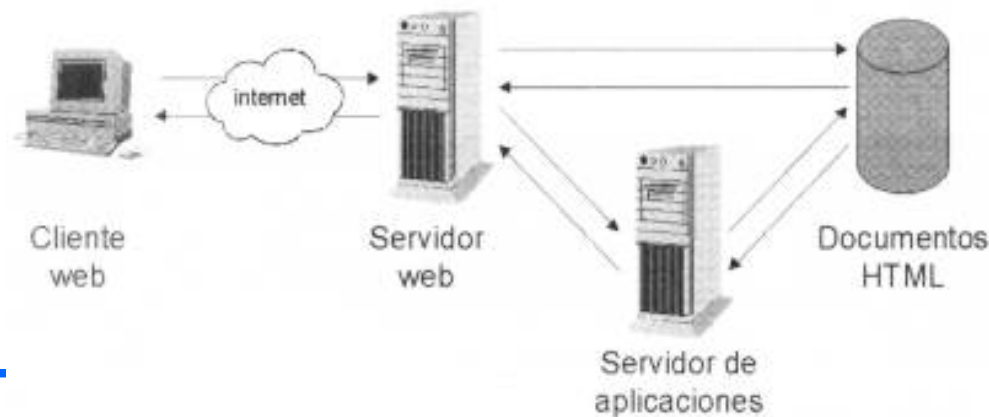
---

- ❖ Programa con el que el usuario interacciona para solicitar a un servidor el envío de páginas web.
  - ❖ Utiliza los protocolos HTTP o FTP
  - ❖ Las páginas web están codificadas en HTML. El cliente web interpreta estos documentos para mostrárselos al usuario en el formato adecuado.
  - ❖ Cuando un documento recibido es un objeto multimedia (vídeo o sonido), el cliente activa una aplicación externa capaz de gestionarlo.
  - ❖ Clientes web o navegadores mas usuales: Netscape, IE Explorer, Mozilla
-

# PÁGINAS DINÁMICAS

---

- ❖ Son páginas creadas en el momento de su petición (combinan una plantilla de documento con los resultados de la consulta a una BD).
- ❖ El servidor web cede el control al denominado **servidor de aplicaciones**, que es quien se encarga de construir la pagina.
- ❖ Una vez creada la pasa al servidor web, que a su vez la envía al cliente.



# PROTOCOLO HTTP (HYPERTEXT TRANSFER PROTOCOL)

---

- ❖ HTTP es el protocolo utilizado en cada transacción de la web (www), mediante el cual se envían las peticiones de acceso a una página web i la respuesta de esta web, cuyo contenido estará en Hipertexto (HTML). HTTP sirve para enviar información con mensajes mediante formularios.
- ❖ HTTP es un protocolo transaccional: Es un protocolo sin estado, es decir, que no almacena información sobre conexiones anteriores.
- ❖ Está basado en el modelo cliente - servidor: Un cliente HTTP abre una conexión y realiza una solicitud al servidor. Este responde al cliente y al finalizar la transacción, se cierra la conexión y se pierden los datos

# PROTOCOLO HTTP (HYPERTEXT TRANSFER PROTOCOL)

---

- ❖ Por este motivo se han popularizado las **cookies**, pequeños ficheros almacenados en el propio ordenador, que puede leer una página web al establecer conexión con el, y de esta forma reconocer un visitante antiguo.
- ❖ La versión actual de HTTP es la 1.1, y su especificación está en el documento RFC2616.
- ❖ HTTP dispone de una variante cifrada mediante SSL llamada HTTPS.
- ❖ La línea inicial es diferente en las solicitudes y en las respuestas.

# PROTOCOLO HTTP (HYPERTEXT TRANSFER PROTOCOL)

---

## Solicitudes HTTP

En las solicitudes vemos campos separados por un espacio en blanco:

Método            recurso                    versión\_del\_protocolo

P.e.: "GET /recursos/notas/notas.html HTTP/1.1"

## Respuesta HTTP

La línea inicial de respuesta tiene 3 campos separados por un espacio:

Versión\_del\_protocolo    codigo\_respuesta    mensaje

P.e.: " HTTP/1.0 200 Ok"    ó    "HTTP/1.1 404 Not found"

# CODIGOS DE RESPUESTA HTTP

---

<b>1xx Misatge d'informació</b> 100 Continúa 101 Canvi de protocol	<b>3xx Redirecció cap a una altre URL</b> 300 Múltiples possibilitats 301 Mudat permanentment 302 Trobat 303 Veure altres 304 No modificat 305 Utilitzi un proxy 307 Redirecció temporal
<b>2xx Operació satisfactòria</b> 200 OK 201 Creat 202 Aceptat 203 Informació no oficial 204 Sense Contingut 205 Contingut per recarregar 206 Contingut parcial	<b>5xx Error per part del servidor</b> 500 Error intern 501 No implementat 502 Pasarela incorrecta 503 Servei no disponible 504 Temps d'espera de la pasarela esgotat 505 Versió de HTTP no suportada

---



# CODIGOS DE RESPUESTA HTTP

---

<b>4xx Error per part del client</b>	408 Temps d'espera esgotat
400 Sol·licitud incorrecta	409 Conflicte
401 No autoritzat	410 Ja no es disponible
402 Pagament requerit	411 Requereix longitud
403 Prohibit	412 Errada precondició
404 No trobat	413 Entitat de sol·licitut massa llarga
405 Mètode no permès	414 URL de sol·licitut massa llarga
406 No acceptable	415 Tipus de mitjà no suportat
407 Proxy requerit	416 Rang sol·licitat no disponible
	417 Errada expectativa

# INTERNET INFORMATION SERVER (IIS)

---

- ❖ Son una serie de servicios para los ordenadores que utilizan el S.O. Microsoft Windows.
  - ❖ En los inicios de IIS formaban parte del pack de opciones de Windows NT. Después se integró dentro de los sistemas operativos de Microsoft, como Windows 2000 o Windows 2003 Server. Windows XP Profesional incluye una versión limitada de IIS.
  - ❖ Los servicios que ofrece son: FTP, SMTP, NNTP i HTTP/HTTPS.
  - ❖ Estos servicios permiten ofrecer páginas web, comportándose el ordenador como un servidor web de internet o intranet.
-

# INTERNET INFORMATION SERVER (IIS)

---

El servidor web se basa en módulos que le dan capacidad para procesar páginas dinámicas como Active Server Pages (ASP) i ASP.NET. También puede incluir otras tecnologías como PHP o Perl.

## Versiones

IIS 1.0, Windows NT 3.51 Service Pack 3

IIS 2.0, Windows NT 4.0

IIS 3.0, Windows NT 4.0 Service Pack 3

IIS 4.0, Windows NT 4.0 Option Pack

IIS 5.0, Windows 2000

IIS 5.1, Windows XP Professional

IIS 6.0, Windows Server 2003 i Windows XP Profes. x64 Edition

IIS 7.0, Windows Vista i Windows Server "Longhorn"

---

# PRACTICA 1. INSTALACIÓN DE IIS EN WINDOWS

---

**Paso 1.** Ir a Inicio / Panel de control/ Agregar o quitar programas. Allí activar la opción Agregar o quitar componentes de Windows y marcar la entrada Servicios de Internet Information Server (US)

**Paso 2.** Realizar la instalación, por defecto, para la cual necesitaremos insertar el CD de Windows 10 Professional.

**Paso 3.** Finalizada la instalación, comprobar que el servidor está bien instalado. Desde un navegador web teclear: <http://localhost/> ó <http://127.0.0.1>. ¿Qué página de prueba aparece?

**Paso 4.** Para comprobar la configuración del servidor, ir a:

→ Inicio/Panel de control/Herramientas administrativas/Servicios de Internet Infor. Server/ sitios web/sitio web predeterminado  
ó

→ Mi PC/Administrar/servicios y aplicaciones/ servicios de IIS/sitios web/sitio web predeterminado

---

# PRACTICA 1. INSTALACIÓN DE IIS EN WINDOWS

---

**Paso 5.** ¿Como sabemos si el servicio está activo?

**Paso 6.** Desactivar el servidor IIS y llamar desde un navegador a `http://localhost`. ¿Qué ocurre?

**Paso 7.** Indica en que directorio local ofrece servicio el servidor IIS.

**Paso 8.** Indica la lista de prioridades de ficheros que servirá primero en caso de que no se le especifique ningún fichero en la URL de llamada.

**Paso 9.** Modificar el nombre del fichero `iisstart.asp` por `iisstart.html`. Activar el servidor y realizar la llamada `http://localhost` desde un navegador ¿qué ocurre?

**NOTA:** En las comprobaciones desde el navegador, en algunas situaciones, se debe refrescar el navegador hasta que cargue correctamente la página web o borrar la cache del mismo.

# PRACTICA 1. INSTALACIÓN DE IIS EN WINDOWS

---

**Paso 10.** Crear un fichero html, llamado default.htm con una mínima interficie de manera que indique "HOLA MUNDO". Realizar la llamada http://localhost desde un navegador ¿qué ocurre?

```
<html>  
<head></head>  
<body> <h1>HOLA MUNDO</h1>  
</body>  
</html>
```

**Paso 11.** Crear en C:\inetpub\wwwroot\ el directorio practica. Dentro de él, crear un fichero index.html muy simple. Desde el navegador ejecutar: http://localhost/practica. ¿Se visualiza el fichero?

**Paso 12.** ¿Qué se debería hacer para visualizar tal fichero?

# PRACTICA 1. INSTALACIÓN DE IIS EN WINDOWS

---

## Creación de un directorio virtual

Dentro de las Propiedades de Sitio Web Predeterminado, se encuentra así mismo la opción de Nuevo /Directorio Virtual. Con ella se puede añadir cualquier directorio del sistema al servidor web sin tener que copiarlo en un directorio concreto. Bastará con indicar cuál es el directorio original al que apunta el virtual que se ha creado. Así se logra que cualquier página web esté rápidamente accesible sin tener que estar en el directorio por defecto, C:\inetpub\wwwroot\

**Paso 13.** Crear un directorio virtual de alias "pepe", que apunte al directorio físico c:\pepedirectorio. Crear aquí un fichero index.html sencillo. Llamar a <http://localhost/pepe> y ver que ocurre

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 1.** Descargar el servidor apache desde la página web <http://www.apache.org/>, la versión Win32 Binary without crypto (no mod\_ssl) (MSI Installer)

**Paso 2.** Realizar la instalación del ejecutable.

**Paso 3.** Una vez finalizada la instalación, comprobar que el servidor está bien instalado. Desde un navegador web teclear:  
<http://localhost/> ó <http://127.0.0.1> ¿Qué se visualiza?

**Paso 4.** ¿Ha habido problemas con el arranque del servidor apache? Si hubiera otro servidor web activado, ocupando el mismo puerto 80, nuestro servidor no arrancaría correctamente. Abrir el monitor de configuración del servidor apache y comprobar que el servicio está activado.

**Paso 5.** Otra forma de comprobar la activación del servidor web apache, consiste en mirar los servicios activos de windows. Ejecutar services.msc y comprobar los servicios iniciados.

---



# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 6.** Indicar qué página se ha visualizado en el paso 3 y su ubicación en el sistema de archivos.

**Paso 7.** Modificar el fichero index.html ubicado en el directorio: c:\Archivos de programa\Apache Software Foundation\Apache2.2\htdocs, y visualizarlo, para comprobar que nos estamos conectando a nuestro servidor local.

**Paso 7.** A continuación renombrar el fichero index.html por otro nombre (p.e. indexxx.html). Volver a teclear desde un navegador: http://localhost/ ó http://127.0.0.1 ¿Qué ocurre ahora?

**Paso 8.** Crear en el directorio raíz "htdocs" otro directorio por ejemplo "dinosaurio". Insertar un fichero html p.e. "pepe.html". Realizar la llamada a dicho fichero, mediante **http://localhost/dinosaurio/pepe.html** ¿Funciona?

¿Qué pasa si ponemos sólo http://localhost/dinosaurio/? ¿Cómo se puede visualizar de esta forma el fichero que hemos puesto?

---

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 9.** Abrir el fichero de configuración de Apache → C:\Archivos de programa\Apache Software Foundation\Apache2.2\conf\httpd.conf. Para que toda modificación que se realice en el fichero http.conf tome efecto, se debe parar y reiniciar el servidor apache

**Paso 10.** Parámetros importantes del fichero httpd.conf:

- ServerRoot → Directorio donde se ubica el servidor
- Listen → Puerto donde se levanta el servidor
- DocumentRoot → Directorio sobre el cual el servidor ofrece servicio web
- DirectoryIndex → Indica los fichero que Apache servirá por defecto si un directorio es indicado.

    <IfModule dir\_module>

        DirectoryIndex index.html

    </IfModule>

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 11.** Crear un directorio virtual en apache. Crearemos un directorio en el disco duro c:\pepe, e insertaremos el fichero jose.html. Dotaremos de servicio web a este directorio mediante la etiqueta elefante.

```
<IfModule alias_module>
  Alias /elefante "c:/pepe/"
</IfModule>
<Directory "C:/pepe/">
  Options Indexes FollowSymLinks
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>
```

Copiar este código en http.conf encima de la clausula #DefaultType (parte final de las clausulas para los alias)

Es necesario proporcionar una sección <Directory> para permitir el acceso al sistema de ficheros del directorio c:\pepe

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 12.** Comprobar la visualización de la página.

`http://localhost/elefante/jose.html`

**Paso 13.** Crear una contraseña con login y password para acceder al directorio htdocs. Se realiza con el fichero .htaccess

Se trata de un archivo oculto donde se especifican instrucciones llamadas directivas de Apache, que definen su funcionamiento de Apache. Normalmente estas directivas se especifican en `httpd.conf`, pero los usuarios de un sitio no suelen tener permitido modificar este archivo. El administrador del servidor Apache puede habilitar el uso de `.htaccess` para que los usuarios puedan hacer uso de ciertas directivas que se aplican únicamente a su sitio y no a todo el servidor.

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 14.** Habilitar el uso de .htaccess en el directorio que nos interesa:

```
<Directory "D:/Archivos de programa/Apache Software  
Foundation/Apache2.2/htdocs">
```

```
Options Indexes FollowSymLinks
```

```
AllowOverride All ←
```

```
Order allow,deny
```

```
Allow from all
```

```
</Directory>
```

**Paso 15.** Ejecutar desde consola MSDOS:

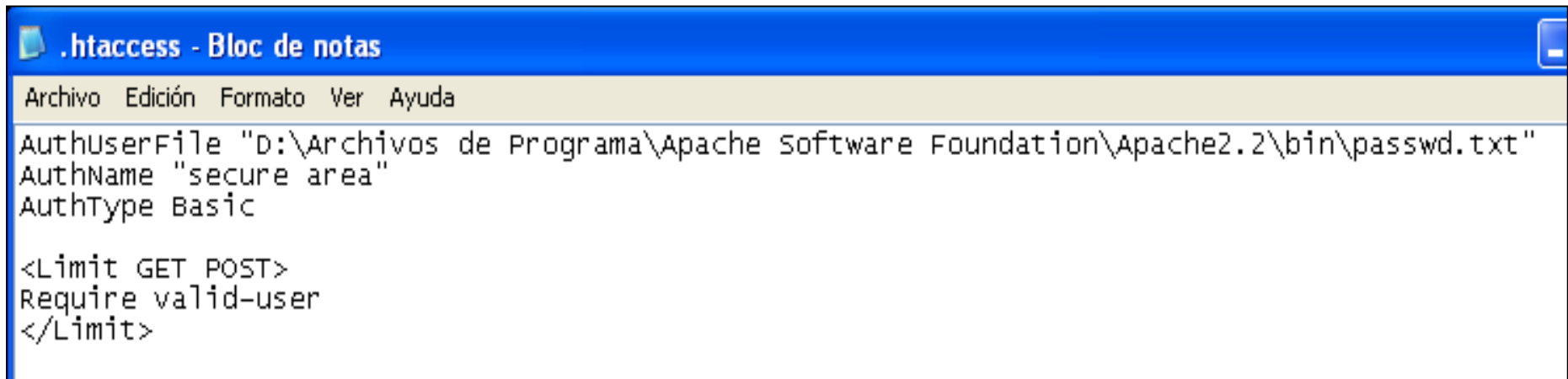
```
C:\Archivos de Programa\Apache Software Foundation\Apache2.2\  
bin> htpasswd.exe -c -b passwd.txt miusuario mipassword
```

Crear un usuario miusuario con password upc, para que pueda acceder a nuestra zona, el cual queda guardado en el fichero passwd.txt

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 16.** Crear el fichero .htaccess en el directorio htdocs con el siguiente contenido:



```
.htaccess - Bloc de notas
Archivo Edición Formato Ver Ayuda
AuthUserFile "D:\Archivos de Programa\Apache Software Foundation\Apache2.2\bin\passwd.txt"
AuthName "secure area"
AuthType Basic

<Limit GET POST>
Require valid-user
</Limit>
```

Desde notepad guardar el fichero poniendo comillas, es decir ".htaccess"

# PRACTICA 2. INSTALACIÓN DE APACHE EN WINDOWS

---

**Paso 17.** Ejecutar desde un navegador `http://127.0.0.1` ¿Qué ocurre?

**Paso 18.** Borrar o mover el fichero `.htaccess` del directorio `htdocs` a otro (quitamos el acceso restringido) y comprobar que ocurre.

**Paso 19.** Existe alguna forma de activar dos servidores web diferentes (p.e. Apache e IIS) en una misma máquina. Indica como

**Paso 20.** Desinstalar el paquete `APACHE` e instalar el paquete `XAMP`, el cual permite la instalación de todo un paquete de servicio web mucho más completo:

Apache

Mysql

Phpmyadmin

Filezilla

PHP

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

Paso 1. Arranca una maquina de Linux dentro del VirtualBox o VMware.

Paso 2. Comprueba de la máquina no tiene ninguna IP asignada de forma fija, de manera que su interficie consiga la dirección IP dinámicamente (por dhcp):

**sudo nano 01-network-manager-all.yaml**

```
GNU nano 4.8 /etc/netplan/01-network-manager-all.yaml
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
```

Paso 3. Reinicia la tarjeta de red mediante el comando:

**sudo systemctl restart NetworkManager**

**sudo systemctl status NetworkManager**

```
marta@escolatreball:~$ sudo systemctl status NetworkManager
● NetworkManager.service - Network Manager
   Loaded: loaded (/lib/systemd/system/NetworkManager.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-01-23 13:36:07 CET; 9s ago
     Docs: man:NetworkManager(8)
   Main PID: 4150 (NetworkManager)
    Tasks: 4 (limit: 2183)
   Memory: 4.6M
   CGroup: /system.slice/NetworkManager.service
           └─4150 /usr/sbin/NetworkManager --no-daemon

ene 23 13:36:07 escolatreball.org NetworkManager[4150]: <info> [1642941367.765]
ene 23 13:36:07 escolatreball.org NetworkManager[4150]: [1642941367.765]
ene 23 13:36:07 escolatreball.org NetworkManager[4150]: [1642941367.765]
```



# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 4.** Para instalar un servidor apache web tenemos dos opciones:

- Instalando directamente el paquete LAMP (contiene Apache, Php y Mysql todo conjunto y preconfigurado). Se puede hacer con tasksel
- Instalando el servidor web apache directamente.

**sudo apt update** (actualiza herramienta descarga aplicaciones)

**sudo apt install apache2** (instalación del servidor web)

```
marta@escolatreball:~$ sudo apt install apache2
[sudo] contraseña para marta:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
apache2 ya está en su versión más reciente (2.4.41-4ubuntu3.8).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 176 no actualizados.
```

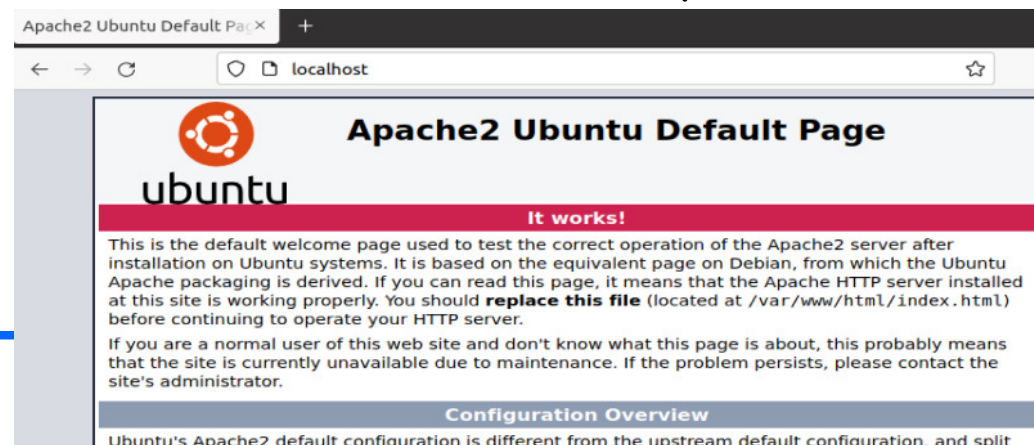
# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

**Paso 5.** Una vez finalizada la instalación, comprobar que el servidor está bien instalado. Reinícialo y comprueba su estado:

**sudo systemctl restart apache2    sudo systemctl status apache2**

```
marta@escolatreball:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese
   Active: active (running) since Sun 2022-01-23 16:12:59 CET; 7s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 5580 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SU
   Main PID: 5584 (apache2)
      Tasks: 6 (limit: 2183)
     Memory: 9.6M
    CGroup: /system.slice/apache2.service
           └─5584 /usr/sbin/apache2 -k start
             └─5585 /usr/sbin/apache2 -k start
               └─5586 /usr/sbin/apache2 -k start
                 └─5587 /usr/sbin/apache2 -k start
                   └─5588 /usr/sbin/apache2 -k start
                     └─5589 /usr/sbin/apache2 -k start
```

**Paso 6.** Desde un navegador web teclea: `http://localhost/` ó `http://127.0.0.1` ¿Qué se visualiza? ¿Por qué?



# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

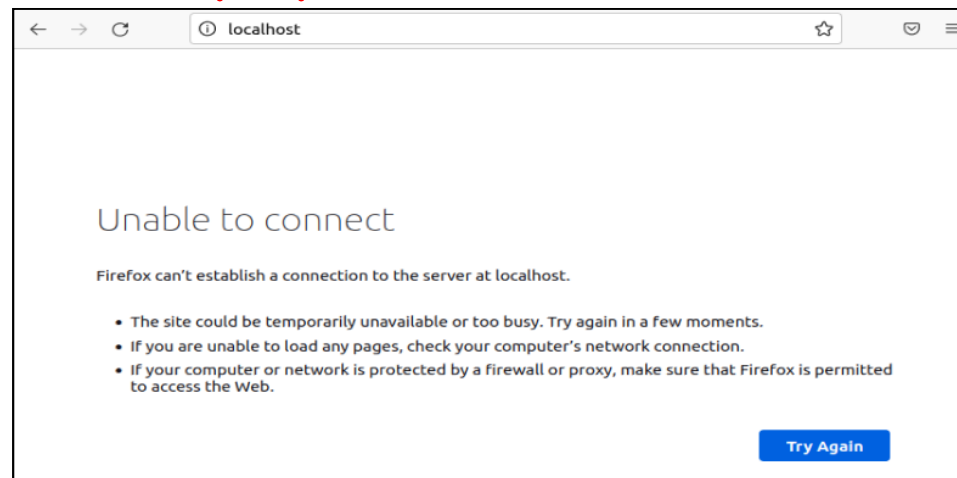
---

Paso 7. Donde se ubica este fichero? Visualízalo con un cat

**cat /var/www/html/index.html**

Paso 8. Para el servidor apache y vuelve a hacer la llamada http://localhost desde el navegador. ¿Que ocurre? Que mensaje da?

**sudo systemctl stop apache2**



Refresca con F5  
para actualizar  
la cache

Paso 9. Reinicia el servidor web:

**sudo systemctl restart apache2**

---

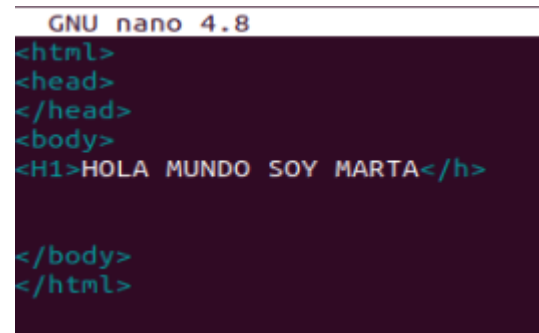
# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

Paso 10. Crea el siguiente fichero:

**sudo nano /var/www/html/index2.html**

```
<html><head></head>
<body>HOLA MUNDO,
nombre alumno
</body></html>
```



```
GNU nano 4.8
<html>
<head>
</head>
<body>
<H1>HOLA MUNDO SOY MARTA</h>

</body>
</html>
```

Paso 11. Prueba de hacer de nuevo <http://localhost> desde el navegador ¿funciona? ¿Qué se debe de poner?




# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 12.** Crea en "/var/www/html" un directorio llamado, por ejemplo, "dinosaurio". Copia el fichero index2.html realizado anteriormente con el nombre "pepe.html". ¿Cómo se haría la llamada desde un navegador?

```
marta@escolatreball:/var/www/html$ sudo mkdir dinosaurio
[sudo] contraseña para marta:
marta@escolatreball:/var/www/html$ ls
dinosaurio  index2.html  index.html  squirrelmail
marta@escolatreball:/var/www/html$ sudo cp index2.html dinosaurio/pepe.html
marta@escolatreball:/var/www/html$
```



A screenshot of a web browser window. The address bar shows the URL "localhost/dinosaurio/pepe.html". The main content area displays the text "HOLA MUNDO SOY MARTA" in a large, bold, black serif font.

**Paso 13.** ¿Qué cambios tendríamos que hacer en el paso anterior si queremos obtener lo mismo con la url http://localhost/ejemplo/?

- Renombrar la carpeta dinosaurio a ejemplo
- Renombrar pepe.html a index.html



# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 14.** Parámetros importantes del servidor web apache repartidos en diferentes ficheros:

	Valor	Fichero
ServerRoot	/etc/apache2	apache2.conf
Alias	/personal/	/sites-enabled/000-default.conf
DocumentRoot	/var/www/html	/sites-enabled/000-default.conf
DirectoryIndex	index.html, index.php	/mods-enabled/dir.conf

Las variables del servidor apache2 en Linux ya no se encuentran todas agrupadas en el fichero httpd.conf, sino que están dispersadas entre numerosos ficheros. Comenta que significan las variables anteriores.

Mirar: <https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-ubuntu-20-04-es>

---

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 15.** Examina el fichero principal de configuración de apache /etc/apache2/apache2.conf. Muestra la jerarquía de configuración de ficheros:

```
GNU nano 4.8          apache2.conf
# virtual hosts, and extra configuration directives as flexible as possible, in
# order to make automating the changes and administering the server as easy as
# possible.

# It is split into several files forming the configuration hierarchy outlined
# below, all located in the /etc/apache2/ directory:
#
#       /etc/apache2/
#       |-- apache2.conf
#       |-- ports.conf
#       |-- mods-enabled
#           |-- *.load
#           |-- *.conf
#       |-- conf-enabled
#           |-- *.conf
#       |-- sites-enabled
#           |-- *.conf
#
```

**Paso 16.** En el fichero /etc/apache2/sites-enabled/000-default.conf, vemos que ya existe un directorio virtual creado por defecto, con la directiva DocumentRoot, que es donde ofrece servicio web el servidor

```
GNU nano 4.8          000-default.conf
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
```



# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

## Definición de un Alias

**Paso 17.** Dentro de este directorio virtual crearemos un alias `"/proyecto"` que apunte a vuestro directorio personal `"/home/usuario/www"`. Sabemos que el servidor apache ofrece servicio web en `/var/www/html`, pero también se puede habilitar otros directorios a nuestro gusto para dotarlos de servicio web. No es necesario que todo se deba de poner en `/var/www/html`.

```
GNU nano 4.8 /etc/apache2/sites-enabled/000-default.conf
# For most configuration files from conf-available/, w
# enabled or disabled at a global level, it is possibl
# include a line for only one particular virtual host.
# following line enables the CGI configuration for thi
# after it has been globally disabled with "a2disconf"
#Include conf-available/serve-cgi-bin.conf
Alias "/proyecto" "/home/marta/www"
<Directory "/home/marta/www">
    Options +Indexes +FollowSymLinks +MultiViews
    AllowOverride none
    Require local
    Require all granted
    #Order allow,deny
    #Allow from all
</Directory>
</VirtualHost>
```

Para la  
versión 2.4  
de apache



# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

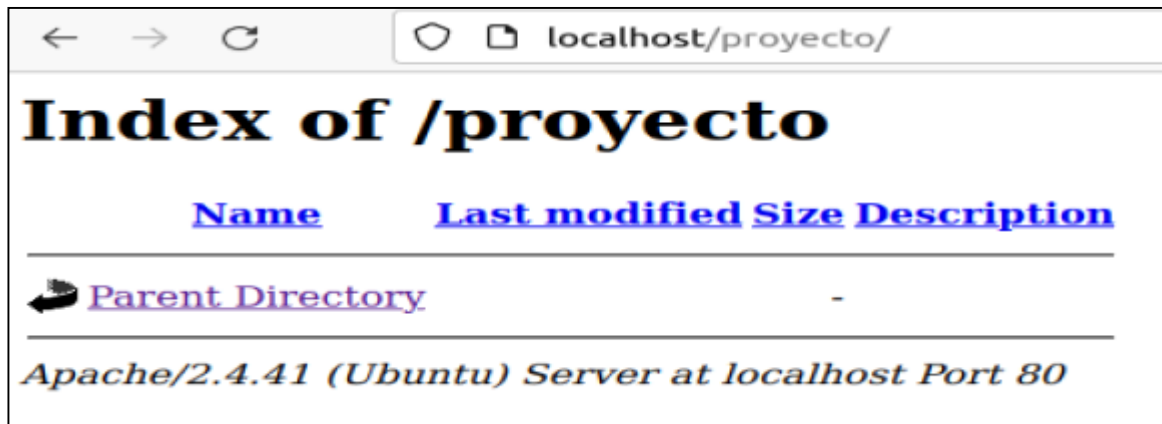
---

**Paso 18.** Comprueba la versión del servidor web apache2:

```
marta@escolatreball:/etc/apache2$ sudo apache2 -v
Server version: Apache/2.4.41 (Ubuntu)
Server built:   2021-10-14T16:24:43
marta@escolatreball:/etc/apache2$
```

**NOTA:** Es importante saber la versión de apache2 puesto que las directivas cambian en una u otra versión: Apache 2.4 utiliza Require para el control de acceso, mientras que Apache 2.2 utiliza las directivas Order, Deny, Allow, Satisfy.

**Paso 19.** Comprueba el resultado desde un navegador <http://localhost/proyecto>. ¿Qué aparece? ¿Por que?



Reinicia el servidor cada vez que Deba actualizar cambios  
`sudo systemctl restart apache2`

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 20.** Crea un fichero index.html en el directorio alias anterior. Visualiza <http://localhost/proyecto>. ¿Qué aparece? ¿Por que?



**NOTA: Ficheros .htaces.** Son archivos ocultos donde se especifican directivas de Apache, para extender su funcionamiento y comportamiento en diferentes directorios. Normalmente estas directivas se especifican en la jerarquía de ficheros de apache, pero los usuarios de un sitio no suelen tener permitido modificar este archivo. El administrador del servidor Apache puede habilitar el uso de .htaccess para que los usuarios puedan hacer uso de ciertas directivas que se aplican únicamente a su sitio y no a todo el servidor.

---

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

## Acceso con login y contraseña mediante .htaccess

**Paso 21.** Crearemos una contraseña con login y password para acceder al directorio personal "/proyecto". Se realiza con el fichero .htaccess. Primero habilitaremos el uso de .htaccess en el directorio /proyecto, modificando la directiva AllowOverride en el archivo 000-default.conf

```
GNU nano 4.8 /etc/apache2/sites-enabled/000-default.conf

# For most configuration files from conf-available/, w
# enabled or disabled at a global level, it is possibl
# include a line for only one particular virtual host.
# following line enables the CGI configuration for thi
# after it has been globally disabled with "a2disconf"
#Include conf-available/serve-cgi-bin.conf
Alias "/proyecto" "/home/marta/www"
<Directory "/home/marta/www">
    Options +Indexes +FollowSymLinks +MultiViews
    AllowOverride All
    Require local
    Require all granted
    #Order allow,deny
    #Allow from all
</Directory>
</VirtualHost>
```

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

Acceso con login y contraseña mediante .htaccess

Paso 22. Localiza el fichero htpasswd, mediante el comando locate.

```
marta@escolatreball:~/www$ sudo apt install mlocate
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Paquetes sugeridos:
  nocache
```

```
marta@escolatreball:~/www$ locate htpassw
/snap/core18/2253/usr/share/bash-completion/completions/htpasswd
/snap/core18/2284/usr/share/bash-completion/completions/htpasswd
/snap/core20/1270/usr/share/bash-completion/completions/htpasswd
/usr/bin/htpasswd
/usr/share/bash-completion/completions/htpasswd
/usr/share/man/man1/htpasswd.1.gz
```

Paso 23. Ejecuta desde terminal:

**/usr/bin/htpasswd -c -b ficheropw miusuario mipassword**

Hemos creado el usuario **miusuario** con password **mipassword**, el cual queda guardado en /etc/password.

```
marta@escolatreball:~/www$ /usr/bin/htpasswd -c -b ficheropw miusuario mipassword
Adding password for user miusuario
marta@escolatreball:~/www$
```

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 24.** Mueve o copia el ficheropw al directorio proyecto alias del usuario

```
marta@escolatreball:~/www$ pwd
/home/marta/www
marta@escolatreball:~/www$ ls
ficheropw  index.html
marta@escolatreball:~/www$
```

**Paso 25.** Crea el fichero .htaccess en el directorio alias con el siguiente contenido: **sudo nano .htaccess**

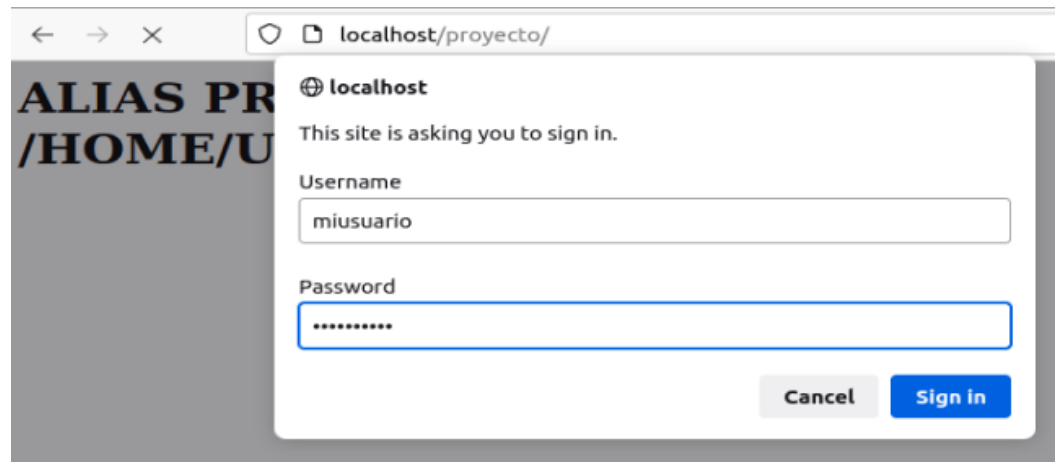
```
GNU nano 4.8 .htaccess
AuthUserFile "/home/marta/www/ficheropw"
AuthName "Secure Area"
AuthType Basic

<LIMIT GET POST>
Require valid-user
</LIMIT>
```

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 26.** Ejecuta desde un navegador `http://127.0.0.1/proyecto`. Si no sale el prompt pueda que sea un problema de la cache del navegador. Se recomienda hacer F5 para forzar a que se actualice:



**Paso 27.** Introduce el login y password anteriores para entrar.

**Paso 28.** Borrar o mover el fichero `.htaccess` del directorio `htdocs` y comprobar que ocurre.

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

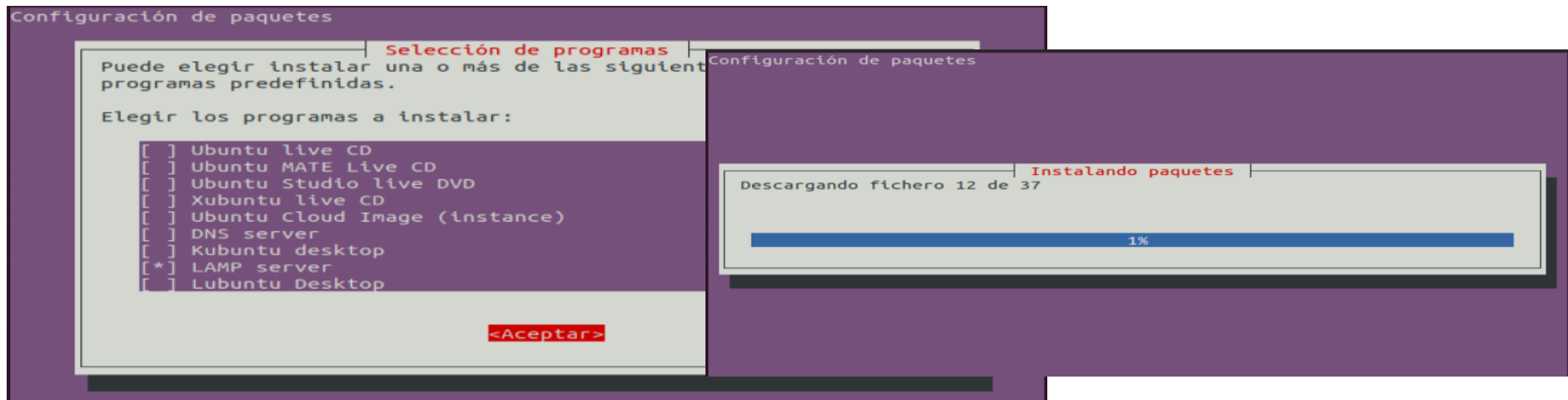
---

## Ejecucion fichero PHP

Paso 29. Instalaremos el paquete LAMP en Ubuntu:

**sudo apt install tasksel**

**sudo tasksel**



Paso 30. Indica los nuevos paquetes que se habrán introducido en esta instalación.

Apache ya estaba instalado. Habrá instalado MSQL y PHP

---

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

**Paso 31.** Por ejemplo podemos renombrar el fichero creado en el paso 10, index2.html a index.php.

```
marta@escolatreball:/var/www/html$ cat index2.html
<html>
<head>
</head>
<body>
<H1>HOLA MUNDO SOY MARTA</h>

</body>
</html>
marta@escolatreball:/var/www/html$
```

**Paso 32.** Podemos poner las siguientes instrucciones php:

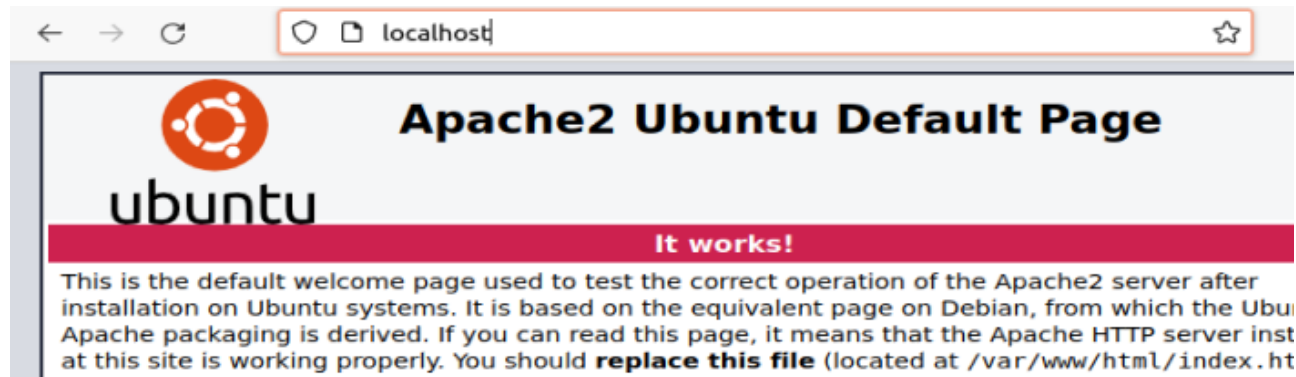
```
GNU nano 4.8 index.php
<html>
<head>
</head>
<body>
<H1>HOLA MUNDO SOY MARTA</h1>
<?php
    echo "Imprimo mi primera linea en PHP<br>";
    for ($i=0;$i<10;$i++){
        echo "Esta practica la voy a aprobar si o si<br>";
    }
?>

</body>
</html>
```



# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

Paso 33. Ejecutamos `http://localhost`. ¿Qué obtenemos? ¿Por que?



Paso 34. Tenemos dos ficheros que se podrían ejecutar por defecto (`index.php` y `index.html`), pero en la directiva **DirectoryIndex** de `/mods-enabled/dir.conf`, aparece `index.html` antes que `index.php`. Debemos cambiar ese orden y poner `index.php` como la primera opción:

```
GNU nano 4.8 dir.conf
<IfModule mod_dir.c>
  DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

GNU nano 4.8 dir.conf
<IfModule mod_dir.c>
  DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

# PRACTICA 3. INSTALACIÓN DE APACHE EN LINUX

---

Paso 35. Ejecutamos de nuevo (hacemos F5). ¿Funciona ahora?

