

# **Tema 2:**

## **Instalación, configuración y documentación del entorno de desarrollo y del entorno de explotación**

Pablo Cidón

Lucía Rodríguez

## Sumario

1. Instalación y configuración del entorno de desarrollo para la asignatura de Desarrollo web del lado servidor sobre dos máquinas virtuales xxx-USED y xxx-W7ED.....	3
XXX-USED.....	3
Configuración de la máquina.....	3
Instalación del software.....	3
Configuración del servidor SSH.....	3
Configuración de Apache.....	4
XXX-W7.....	7
Configuración inicial.....	7
Administración del sitio.....	8
Instalación y uso de aplicaciones.....	9
Google Chrome:.....	9
Putty:.....	10
Netbeans.....	11
Notepad ++.....	14
WebStorm/PhpStorm.....	16
2. Instalación y configuración del entorno de explotación personal para la asignatura de Desarrollo web del lado servidor sobre una máquina virtual xxx-USEE.....	18
XXX-USEE.....	18
Configuración de la máquina.....	18
Instalación del software.....	18
Configuración del servidor SSH.....	19
Configuración de Apache.....	19
Administración del sitio.....	21

# 1. Instalación y configuración del entorno de desarrollo para la asignatura de Desarrollo web del lado servidor sobre dos máquinas virtuales xxx-USED y xxx-W7ED.

## XXX-USED

### Configuración de la máquina

Máquina XXX-USED		
Sistema Operativo	Ubuntu Server 16.04 64bits	
Memoria	512 MB	
Usuarios y contraseñas	miadmin	paso
	operadorweb	paso
Configuración de red	IP	192.168.3.10X
	Máscara	255.255.255.0
	Servidor DNS	192.168.20.20
Software	Servidor Web	Apache (con módulo PHP)
	Servidor SSH	OpenSSH-server
	Servicio SFTP	SFTP
	SGBD	MySQL-server

### Instalación del software

El software que necesitaremos instalar en esta máquina es openssh-server, apache2 y php7.0. Para ello utilizaremos los comandos

- **apt-get update** → para actualizar el repositorio
- **apt-get install openssh-server** → para instalar el servidor ssh y poder conectarnos de forma remota
- **apt-get install apache2** → para instalar apache y tener nuestro servidor web
- **apt-get install php7.0** → para poder programar en php

### Configuración del servidor SSH

Lo primero que haremos será crear en el directorio “/var/www” donde guardaremos nuestra página, en este caso lo llamaremos “public\_html”. Después le daremos los permisos para poder modificar en esta carpeta a nuestro usuario (que crearemos más adelante).

```
root@LRAUSED:~# mkdir public_html
root@LRAUSED:~# chmod 775 public_html/
```

Creamos un nuevo usuario, desde el que nos conectaremos:

```
miadmin@PCB-ED:~$ sudo adduser --home /var/www/html --ingroup www-data --shell /bin/bash operadorweb
```

Cambiamos el propietario del fichero:

```
miadmin@PCB-ED:~$ sudo chown www-data:www-data /var/www/html
```

Desde el cliente Filezilla comprobaremos que podemos conectarnos a nuestro servidor:

Servidor:	<input type="text" value="sftp://192.168.1.150"/>	Nombre de usuario:	<input type="text" value="operadorweb"/>	Contraseña:	<input type="password" value="...."/>
Estado:	Conectando a 192.168.1.150...				
Estado:	Connected to 192.168.1.150				
Estado:	Recuperando el listado del directorio...				
Estado:	Listing directory /var/www/html				
Estado:	Directory listing of "/var/www/html" successful				

## Configuración de Apache

En el directorio “/etc/apache2/sites-enabled” vemos que hay un sitio que se llama “000-default.conf”, que es el fichero en el que se encuentra la configuración de nuestro sitio por defecto.

En primer lugar observamos que puerto va a utilizar nuestro sitio por defecto. Si lo deseamos se podrá cambiar.

```
<VirtualHost *:80>
```

También observamos el “DocumentRoot”, que nos indica en donde se encuentra alojado nuestro sitio. También se puede cambiar.

```
<VirtualHost *:80>
    # The ServerName directive sets the r
    # the server uses to identify itself.
    # redirection URLs. In the context of
    # specifies what hostname must appear
    # match this virtual host. For the de
    # value is not decisive as it is used
    # However, you must set it for any fu
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., t
```

Si hemos realizado cambios, y deseamos que se apliquen, tendremos que reiniciar el servicio de apache, para ello ejecutaremos el siguiente comando:

```
miadmin@PCB-ED:~$ sudo service apache2 restart
```

Dado que nuestro servidor necesita estar preparado para que pueda leer el código PHP, es necesario que instalemos una librería de apache para interpretar dicho lenguaje:

```
miadmin@PCB-ED:~$ sudo apt-get install libapache2-mod-php
```

Después de instalar la librería, es necesario que activemos el módulo de php, para ello ejecutaremos el siguiente comando:

```
miadmin@PCB-ED:~$ sudo a2enmod php7.0.load
```

Después editaremos el fichero /etc/apache2/mods-enabled/dir.conf, para que reconozca los archivos "index":

```
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.css
</IfModule>
```


Para que se apliquen los cambios es necesario que se reinicie el servicio de apache, para ello ejecutaremos el siguiente comando:

```
miadmin@PCB-ED:~$ sudo service apache2 restart
```

Para comprobar que nuestro servidor interpreta correctamente el lenguaje PHP, crearemos un archivo llamado "info.php" en el que introduciremos el siguiente código:

```
1 <?php
2 phpinfo();
3 ?>
```

Vamos al navegador e introducimos "IP/info.php", y nos mostrará lo siguiente:

Versión PHP 7.0.25-0ubuntu0.16.04.1	
	
Sistema	Linux PCB-ED 4.4.0-87-generic # 110-Ubuntu SMP Martes 18 de julio 12:55:35 UTC 2017 x86_64
API del servidor	Apache 2.0 Handler
Soporte de directorio virtual	desactivado
Ruta del archivo de configuración (php.ini)	/etc/php/7.0/apache2
Archivo de configuración cargada	/etc/php/7.0/apache2/php.ini
Escanee este directorio para archivos .ini adicionales	/etc/php/7.0/apache2/conf.d
Archivos .ini adicionales analizados	/etc/php/7.0/apache2/conf.d/10-mysqld.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/15-xml.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-dom.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gd.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-mbstring.ini, /etc/php/7.0/apache2/conf.d/20-mcrypt.ini, /etc/php/7.0/apache2/conf.d/20-mysqli.ini, /etc/php/7.0/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.0/apache2/conf.d/20-phar.ini, /etc/php/7.0/apache2/conf.d/20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-simplexml.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.0/apache2/conf.d/20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini, /etc/php/7.0/apache2/conf.d/20-wddx.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini, /etc/php/7.0/apache2/conf.d/20-wddx.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini
PHP API	20151012

Consiste en una página en la que nos mostrará toda la configuración de PHP.

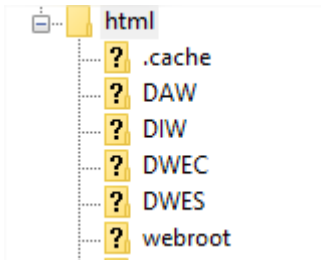
**XXX-W7****Configuración inicial**

Máquina: XXX-W7ED		
<b>Sistema Operativo</b>	Windows 7 profesional 32bits	
<b>Memoria</b>	1024MB	
<b>Usuarios y contraseñas</b>	admin	admin
	usuario	paso
<b>Configuración de red</b>	IP	192.168.3.120
	Máscara	255.255.255.0
	Puerta de enlace	192.168.3.1
	DNS	192.168.20.20
<b>Software</b>	Navegadores	Chrome Opera Safari Microsoft IE/Edge Firefox Opera
	IDE's	Netbeans Web/PhpStorm
	Ofimática	Libre Office Gimp
	Editores de texto	Notepad++ Komodo 10
	Otros	Filezilla, Putty

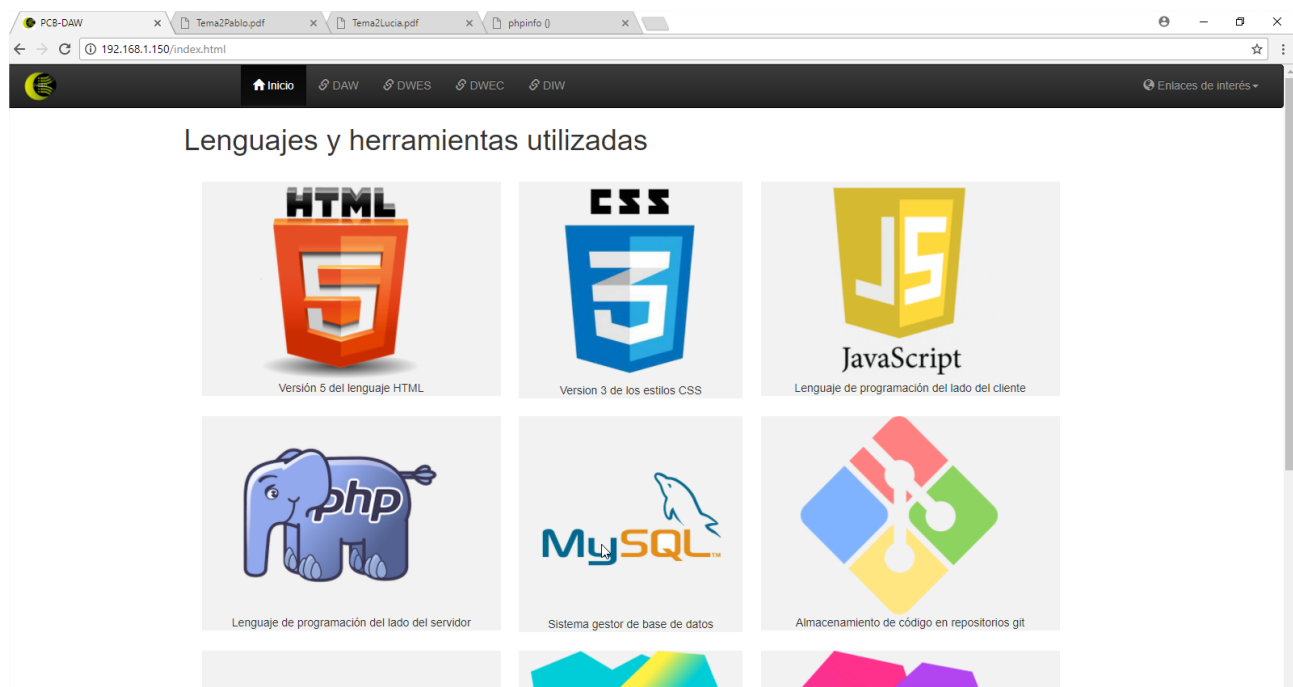
## Administración del sitio

Desde nuestro cliente, y con la ayuda de un IDE, podremos comenzar a crear nuestro sitio. Para subir los archivos a nuestro sitio podremos hacerlo desde Filezilla o desde nuestro propio IDE.

Crearemos nuestra estructura de directorios, en este caso será uno por cada uno de los módulos y otro para almacenar lo relativo al diseño de nuestro sitio.



Una vez creada toda la estructura podremos comenzar a crear nuestro sitio:

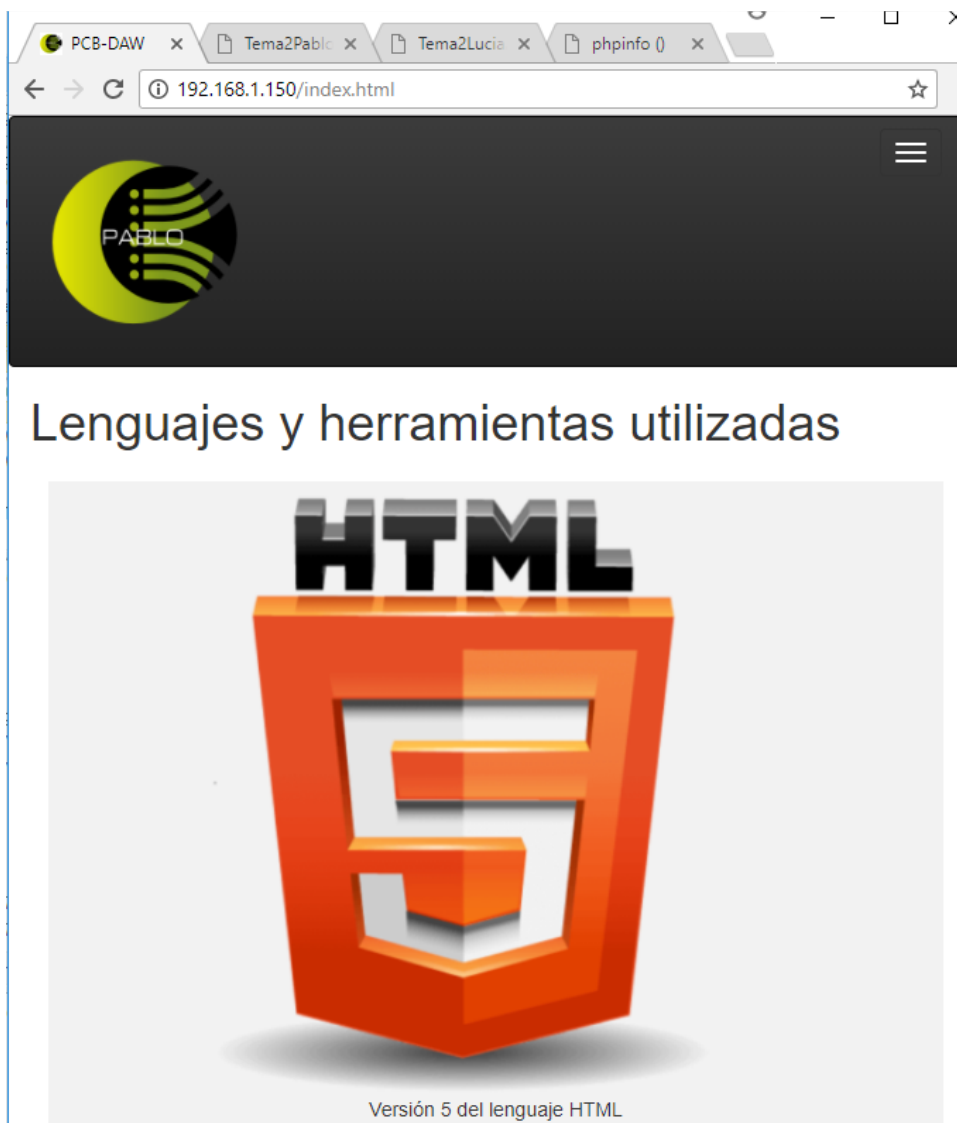




## Instalación y uso de aplicaciones

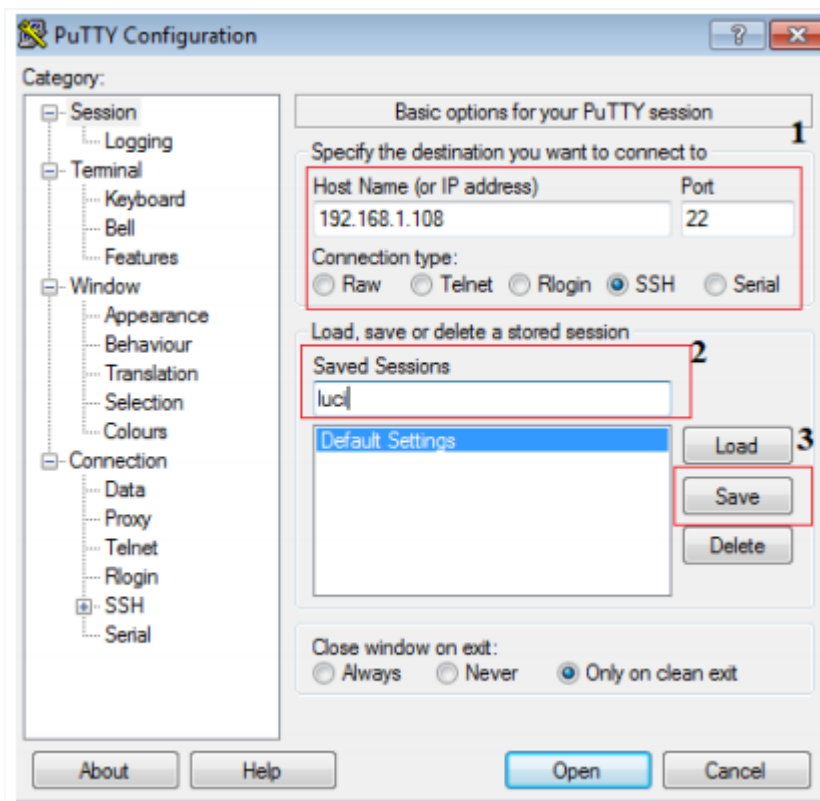
### Google Chrome:

Uno de los navegadores que utilizaremos para observar nuestras páginas. Realizaremos comprobaciones en otros navegadores ya que hay funcionalidades que no son compatibles con todos los navegadores.



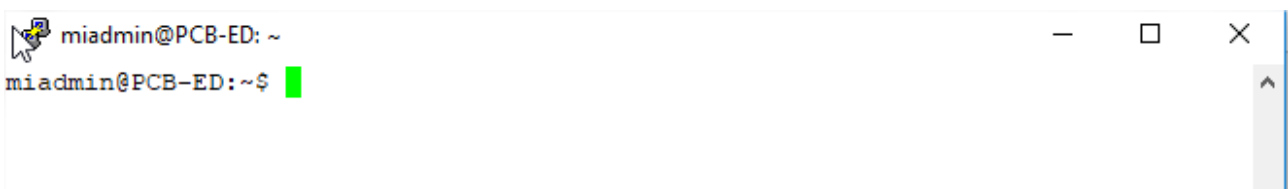
**Putty:**

Herramienta que utilizaremos para conectarnos de forma remota al servidor.



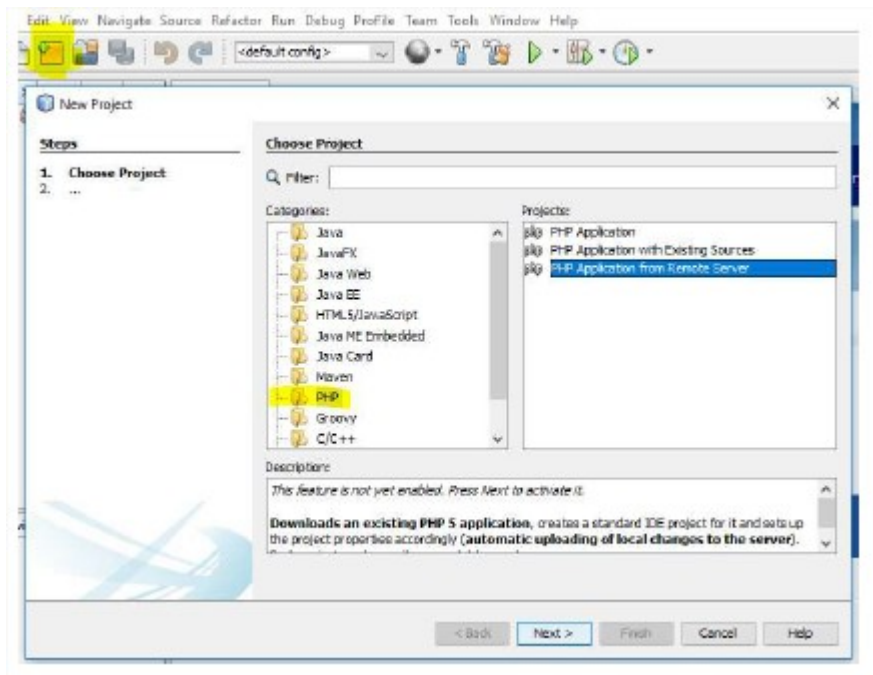
1. Indicaremos la ip del equipo al que queremos conectarnos, el puerto por el que nos conectaremos y el método en nuestro caso será por ssh.
2. Le pondremos un nombre a nuestra conexión
3. Guardaremos nuestra conexión para así la próxima vez entrar directamente solo con clicar en ella

Por defecto funciona con el fondo blanco, pero tiene la opción de cambiar los colores, por lo que en este caso está configurado para que esté con el fondo blanco.

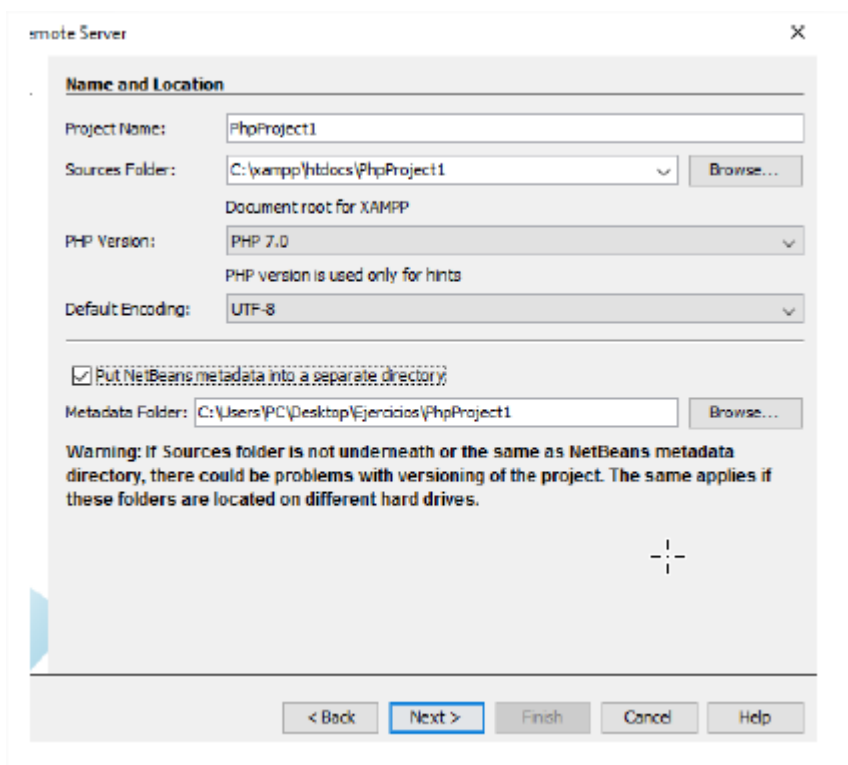


## Netbeans

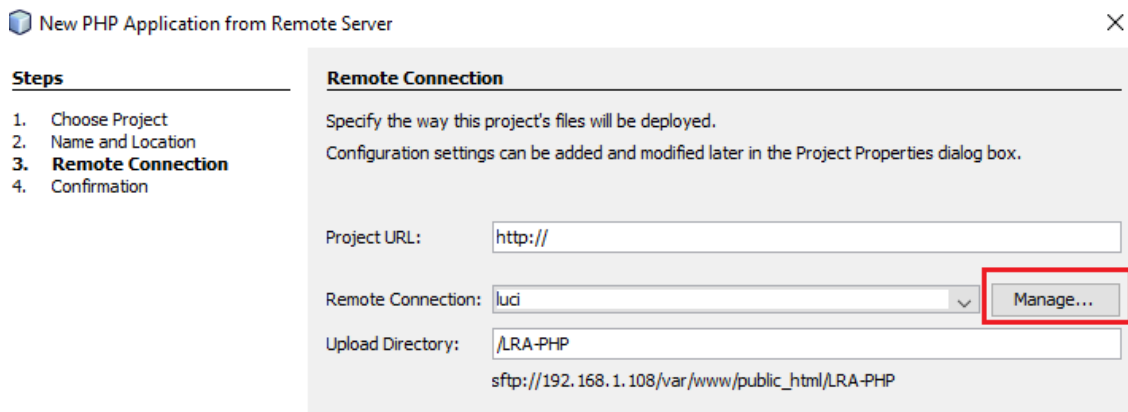
Creamos un nuevo proyecto, de PHP y en un servidor remoto.



Luego establecemos donde vamos a guardar el proyecto, los metadatos generados y la versión de PHP que se va a usar.



A continuación, creamos la conexión con el servidor.



New PHP Application from Remote Server

**Steps**

1. Choose Project
2. Name and Location
3. **Remote Connection**
4. Confirmation

**Remote Connection**

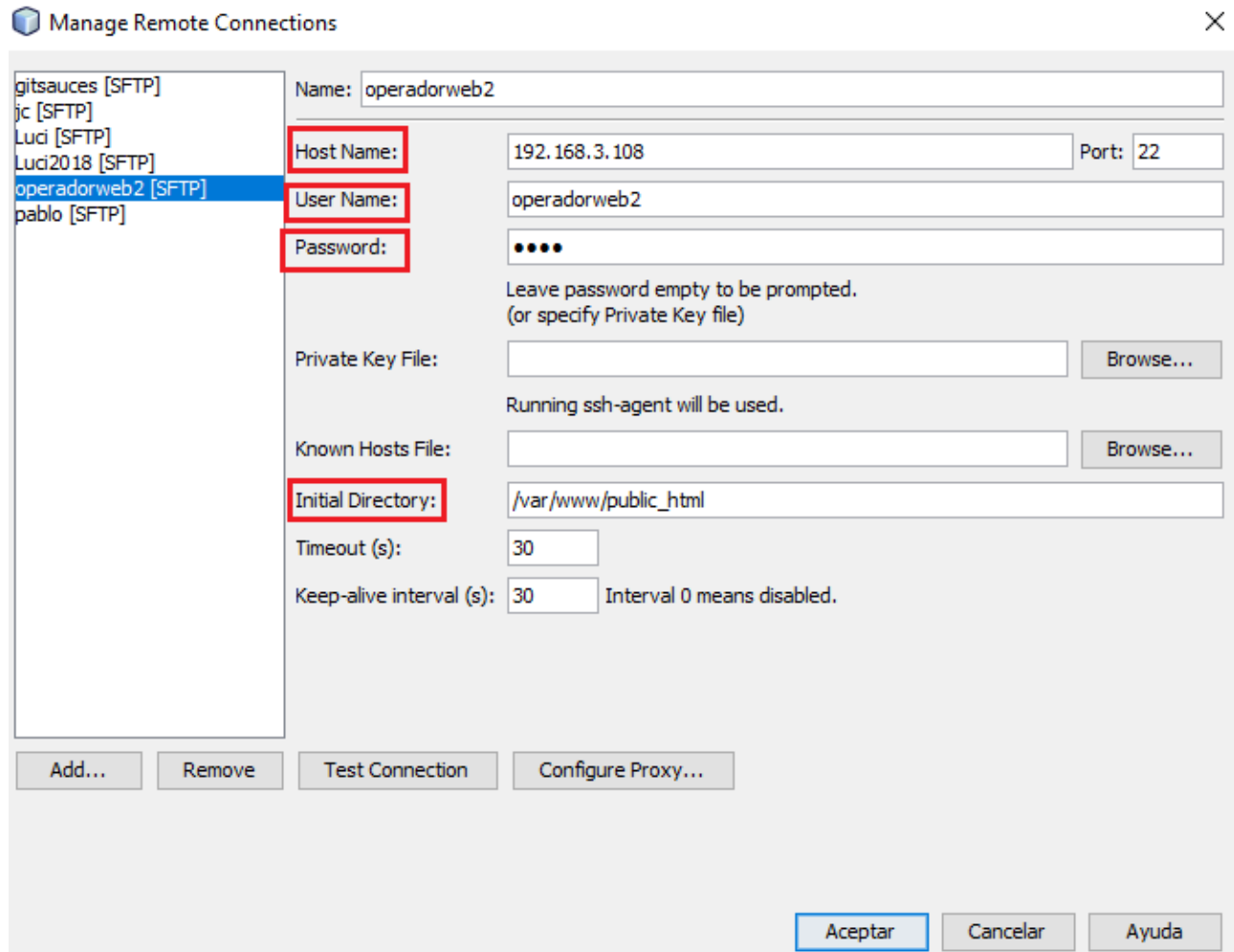
Specify the way this project's files will be deployed.  
Configuration settings can be added and modified later in the Project Properties dialog box.

Project URL:

Remote Connection:  Manage...

Upload Directory:   
sftp://192.168.1.108/var/www/public\_html/LRA-PHP

Indicamos el host, el nombre de usuario, la contraseña y el lugar donde se va a guardar dentro del servidor.



Manage Remote Connections

gitsaucos [SFTP]  
jc [SFTP]  
Luci [SFTP]  
Luci2018 [SFTP]  
**operatorweb2 [SFTP]**  
pablo [SFTP]

Name:

Host Name:  Port:

User Name:

Password:

Leave password empty to be prompted.  
(or specify Private Key file)

Private Key File:  Browse...

Running ssh-agent will be used.

Known Hosts File:  Browse...

Initial Directory:

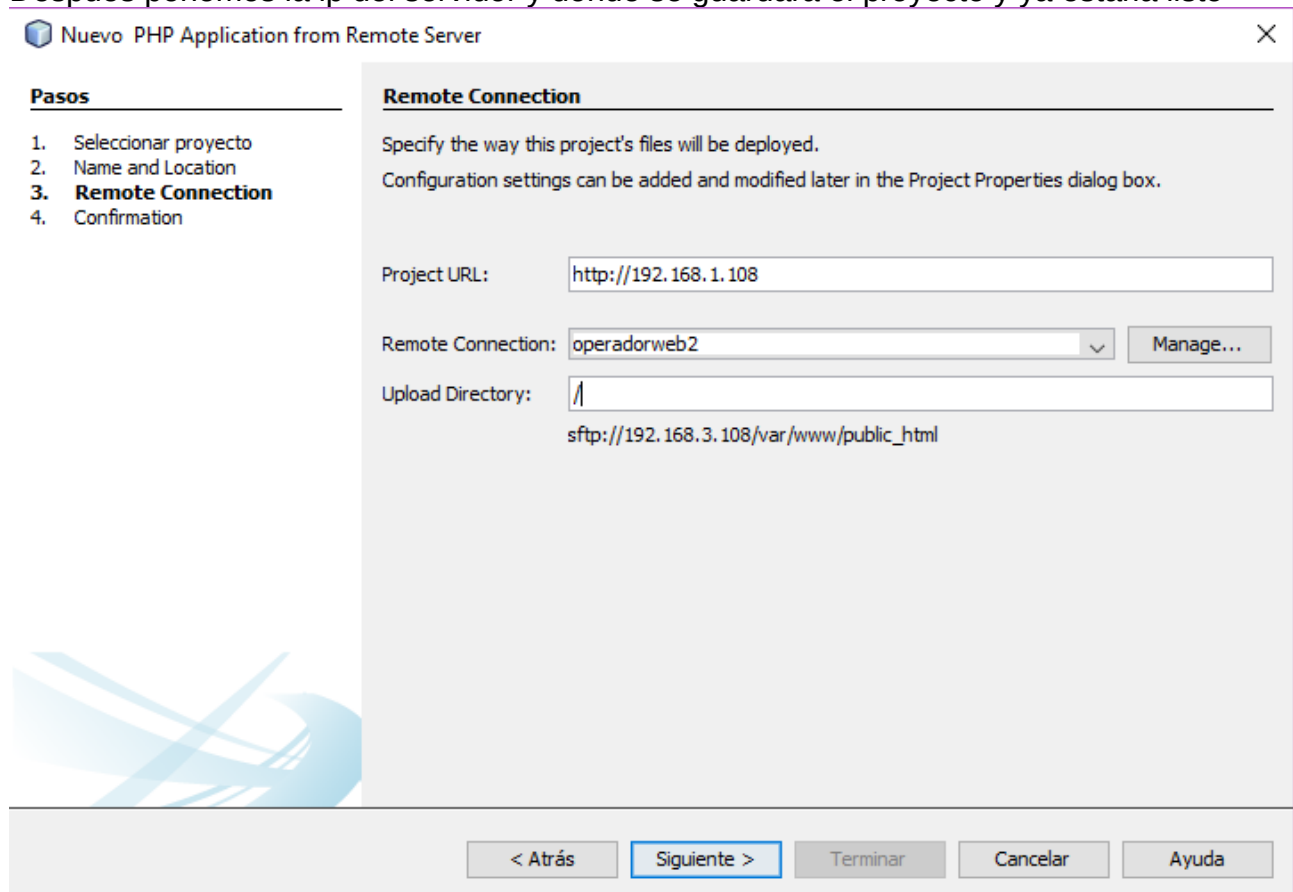
Timeout (s):

Keep-alive interval (s):  Interval 0 means disabled.

Add... Remove Test Connection Configure Proxy...

Aceptar Cancelar Ayuda

Despues ponemos la ip del servidor y donde se guardara el proyecto y ya estaria listo



The screenshot shows a dialog box titled "Nuevo PHP Application from Remote Server" with a close button (X) in the top right corner. On the left, under the heading "Pasos", there is a list of steps: 1. Seleccionar proyecto, 2. Name and Location, 3. **Remote Connection**, and 4. Confirmation. The main area of the dialog is titled "Remote Connection" and contains the following text: "Specify the way this project's files will be deployed. Configuration settings can be added and modified later in the Project Properties dialog box." Below this text are three input fields: "Project URL:" with the value "http://192.168.1.108", "Remote Connection:" with a dropdown menu showing "operadorweb2" and a "Manage..." button to its right, and "Upload Directory:" with a text field containing a forward slash "/" and a suggested path "sftp://192.168.3.108/var/www/public\_html" below it. At the bottom of the dialog, there are five buttons: "< Atrás", "Siguiete >" (highlighted with a blue border), "Terminar", "Cancelar", and "Ayuda".

**Nuevo PHP Application from Remote Server**

**Pasos**

1. Seleccionar proyecto
2. Name and Location
- 3. Remote Connection**
4. Confirmation

**Remote Connection**

Specify the way this project's files will be deployed.  
Configuration settings can be added and modified later in the Project Properties dialog box.

Project URL:

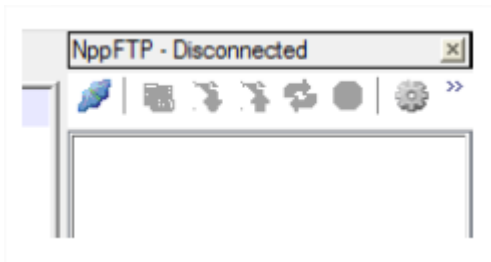
Remote Connection:  Manage...

Upload Directory:   
sftp://192.168.3.108/var/www/public\_html

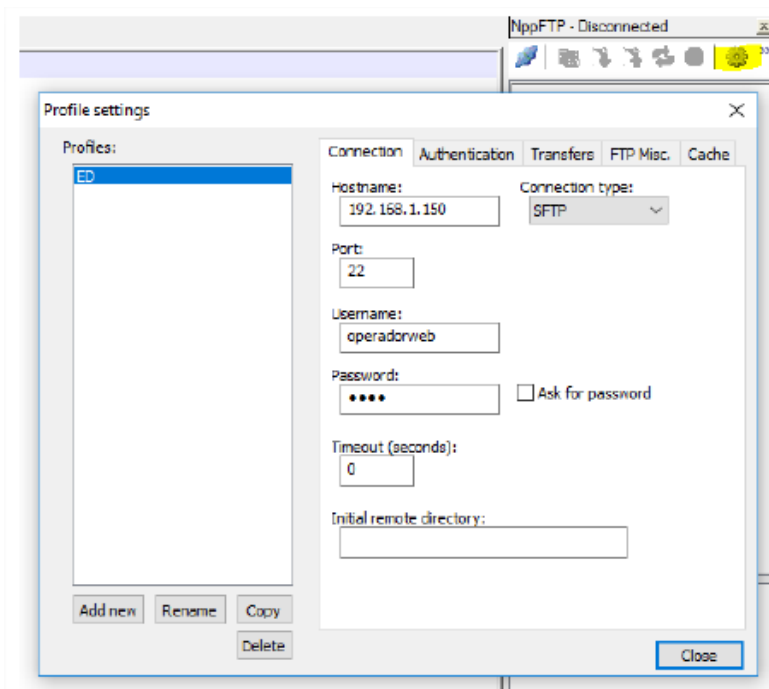
< Atrás Siguiete > Terminar Cancelar Ayuda

**Notepad ++**

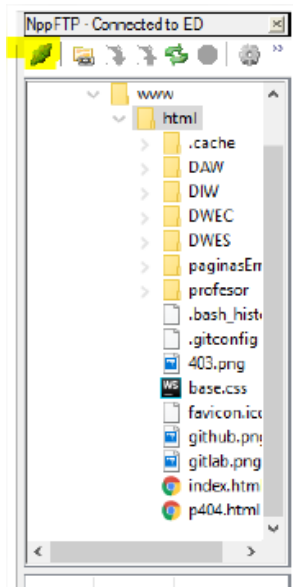
En el menú, vamos a plugins, luego a NPP\_FTP y seleccionamos la opción de mostrar la ventana FTP:



Luego vamos a settings y nos aparecerá un diálogo el que introduciremos los datos para establecer la conexión.

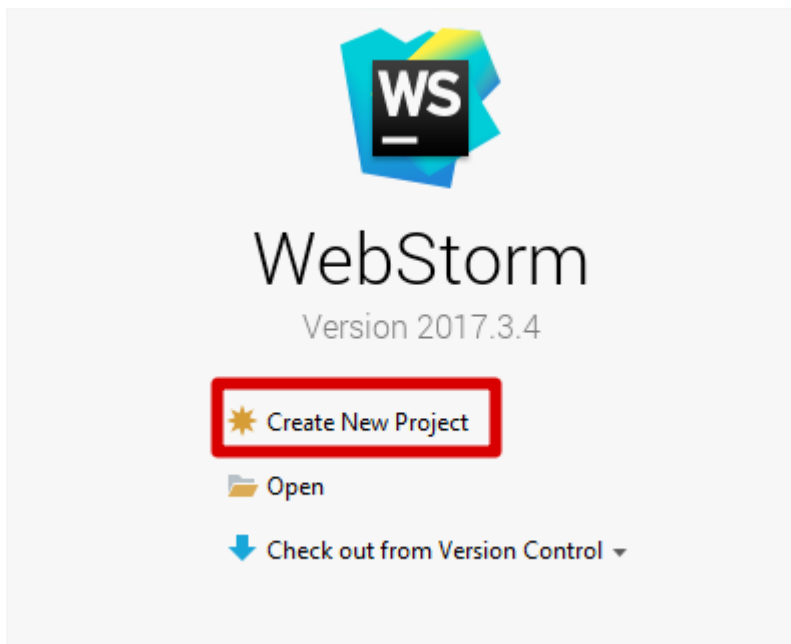


Una vez hecho probamos si funciona.

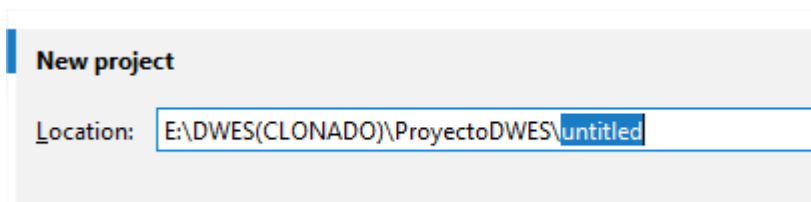


## WebStorm/PhpStorm

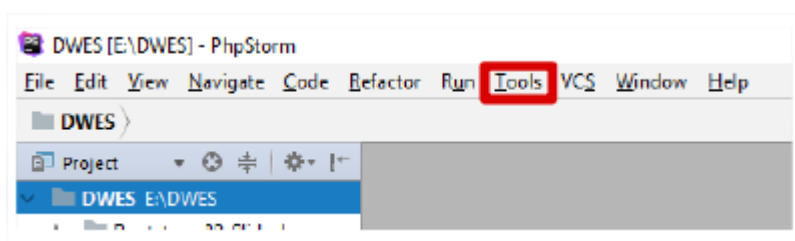
En primer lugar, crearemos nuestro proyecto:



Luego establecemos la ruta en la que se va a encontrar y le damos un nombre:

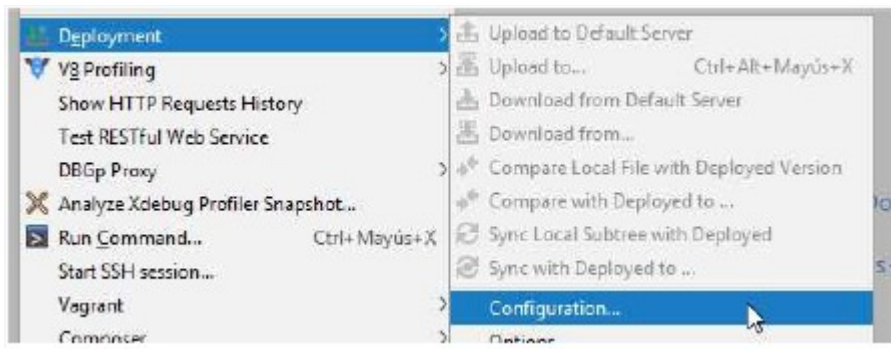


Para establecer la conexión con nuestro servidor vamos a la pestaña de tools.

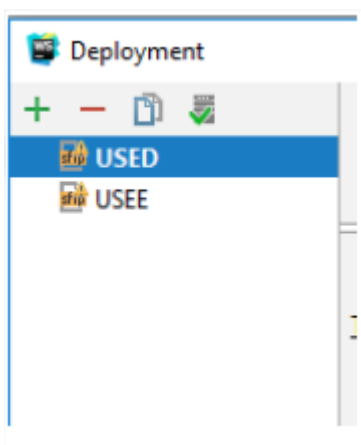


A continuación vamos a la opción de deployment y luego a configuración.

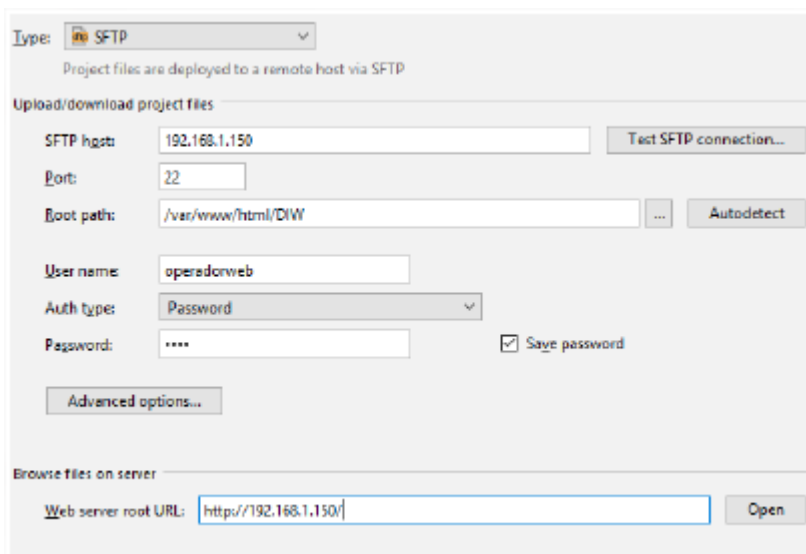




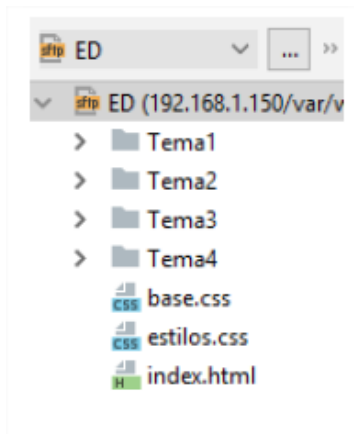
Luego nos aparecerá una ventana con las configuraciones que tenemos, para crear una nueva pulsaremos en +.



Luego completamos la conexión.



Finalmente comprobamos que se ha realizado correctamente.



## 2. Instalación y configuración del entorno de explotación personal para la asignatura de Desarrollo web del lado servidor sobre una máquina virtual xxx-USEE.

### XXX-USEE

#### Configuración de la máquina

Máquina XXX-USED		
<b>Sistema Operativo</b>	Ubuntu Server 16.04 64bits	
<b>Memoria</b>	512 MB	
<b>Usuarios y contraseñas</b>	miadmin	paso
	operadorweb	paso
<b>Configuración de red</b>	IP	192.168.3.10X
	Máscara	255.255.255.0
	Servidor DNS	192.168.20.20
<b>Software</b>	Servidor Web	Apache (con módulo PHP)
	Servidor SSH	OpenSSH-server
	Servicio SFTP	SFTP
	SGBD	MySQL-server

#### Instalación del software

El software que necesitaremos instalar en esta máquina es openssh-server, apache2 y php7.0. Para ello utilizaremos los comandos

- **apt-get update** → para actualizar el repositorio

- **apt-get install openssh-server** → para instalar el servidor ssh y poder conectarnos de forma remota
- **apt-get install apache2** → para instalar apache y tener nuestro servidor web
- **apt-get install php7.0** → para poder programar en php

## Configuración del servidor SSH

Lo primero que haremos será crear en el directorio “/var/www” donde guardaremos nuestra página, en este caso lo llamaremos “public\_html”. Después le daremos los permisos para poder modificar en esta carpeta a nuestro usuario (que crearemos más adelante).

```
root@LRAUSED:~# mkdir public_html
root@LRAUSED:~# chmod 775 public_html/
```

Creemos un nuevo usuario, desde el que nos conectaremos:

```
miadmin@PCB-ED:~$ sudo adduser --home /var/www/html --ingroup www-data --shell /bin/bash operadorweb
```

Cambiamos el propietario del fichero:

```
miadmin@PCB-ED:~$ sudo chown www-data:www-data /var/www/html
```

Desde el cliente Filezilla comprobaremos que podemos conectarnos a nuestro servidor:

Servidor:	sftp://192.168.1.150	Nombre de usuario:	operadorweb	Contraseña:	••••
Estado:	Conectando a 192.168.1.150...				
Estado:	Connected to 192.168.1.150				
Estado:	Recuperando el listado del directorio...				
Estado:	Listing directory /var/www/html				
Estado:	Directory listing of "/var/www/html" successful				

## Configuración de Apache

En el directorio “/etc/apache2/sites-enabled” vemos que hay un sitio que se llama “000-default.conf”, que es el fichero en el que se encuentra la configuración de nuestro sitio por defecto.

En primer lugar observamos que puerto va a utilizar nuestro sitio por defecto. Si lo deseamos se podrá cambiar.

```
<VirtualHost *:80>
```

También observamos el “DocumentRoot”, que nos indica en donde se encuentra alojado nuestro sitio. También se puede cambiar.

```
<VirtualHost *:80>
    # The ServerName directive sets the r
    # the server uses to identify itself.
    # redirection URLs. In the context of
    # specifies what hostname must appear
    # match this virtual host. For the de
    # value is not decisive as it is used
    # However, you must set it for any fu
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., t
```

Si hemos realizado cambios, y deseamos que se apliquen, tendremos que reiniciar el servicio de apache, para ello ejecutaremos el siguiente comando:

```
miadmin@PCB-ED:~$ sudo service apache2 restart
```

Dado que nuestro servidor necesita estar preparado para que pueda leer el código PHP, es necesario que instalemos una librería de apache para interpretar dicho lenguaje:

```
miadmin@PCB-ED:~$ sudo apt-get install libapache2-mod-php
```

Después de instalar la librería, es necesario que activemos el módulo de php, para ello ejecutaremos el siguiente comando:

```
miadmin@PCB-ED:~$ sudo a2enmod php7.0.load
```

Después editaremos el fichero /etc/apache2/mods-enabled/dir.conf, para que reconozca los archivos "index":

```
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index$
</IfModule>
```


Para que se apliquen los cambios es necesario que se reinicie el servicio de apache, para ello ejecutaremos el siguiente comando:

```
miadmin@PCB-ED:~$ sudo service apache2 restart
```

Para comprobar que nuestro servidor interpreta correctamente el lenguaje PHP, crearemos un archivo llamado "info.php" en el que introduciremos el siguiente código:

```
1 <?php
2 phpinfo();
3 ?>
```

Vamos al navegador e introducimos "IP/info.php", y nos mostrará lo siguiente:

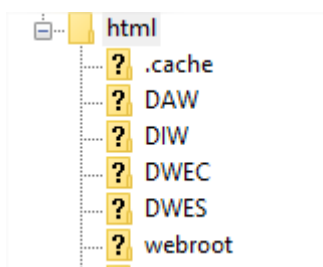
Versión PHP 7.0.25-0ubuntu0.16.04.1 	
Sistema	Linux PCB-ED 4.4.0-87-generic # 110-Ubuntu SMP Martes 18 de julio 12:55:35 UTC 2017 x86_64
API del servidor	Apache 2.0 Handler
Soporte de directorio virtual	desactivado
Ruta del archivo de configuración (php.ini)	/etc/php/7.0/apache2
Archivo de configuración cargada	/etc/php/7.0/apache2/php.ini
Escanee este directorio para archivos .ini adicionales	/etc/php/7.0/apache2/conf.d
Archivos .ini adicionales analizados	/etc/php/7.0/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/15-xml.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-dom.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gd.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-mbstring.ini, /etc/php/7.0/apache2/conf.d/20-mcrypt.ini, /etc/php/7.0/apache2/conf.d/20-mysqli.ini, /etc/php/7.0/apache2/conf.d/20-pdo_mysqli.ini, /etc/php/7.0/apache2/conf.d/20-phar.ini, /etc/php/7.0/apache2/conf.d/20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-simplexml.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.0/apache2/conf.d/20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini, /etc/php/7.0/apache2/conf.d/20-wddx.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini, /etc/php/7.0/apache2/conf.d/20-wddx.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini
PHP API	20151012

Consiste en una página en la que nos mostrará toda la configuración de PHP.

## Administración del sitio

Desde nuestro cliente, y con la ayuda de un IDE, podremos comenzar a crear nuestro sitio. Para subir los archivos a nuestro sitio podremos hacerlo desde Filezilla o desde nuestro propio IDE.

Crearemos nuestra estructura de directorios, en este caso será uno por cada uno de los módulos y otro para almacenar lo relativo al diseño de nuestro sitio.



Una vez creada toda la estructura podremos comenzar a crear nuestro sitio:

