

Instalando LAMP-SERVER en Ubuntu 9.10

(como trabajar en modo local con un CMS)

A petición de los compañeros de izalug aquí les presento un pequeño manual de cómo podemos trabajar con un CMS localmente (sin conexión al internet). El tutorial estará guiado con capturas de pantalla para mostrar el proceso paso a paso, además como la mayoría de los miembros del izalug usan ubuntu será dicha distro nuestro anfitrión :D

Lo primero es instalar el servidor independiente que nos permita trabajar sin conexión, voy a mencionar dos:

LAMP (Linux Apache Mysql Php)

En las tecnologías LAMP esto se consigue mediante la unión de las siguientes tecnologías:

Linux, el sistema operativo;

Apache, el servidor web;

MySQL, el gestor de bases de datos;

Perl, PHP, o Python, lenguajes de programación.

Para más información sobre LAMP ir a <http://es.wikipedia.org/wiki/LAMP>

XAMPP [X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl]

Actualmente XAMPP se encuentra disponible para Windows

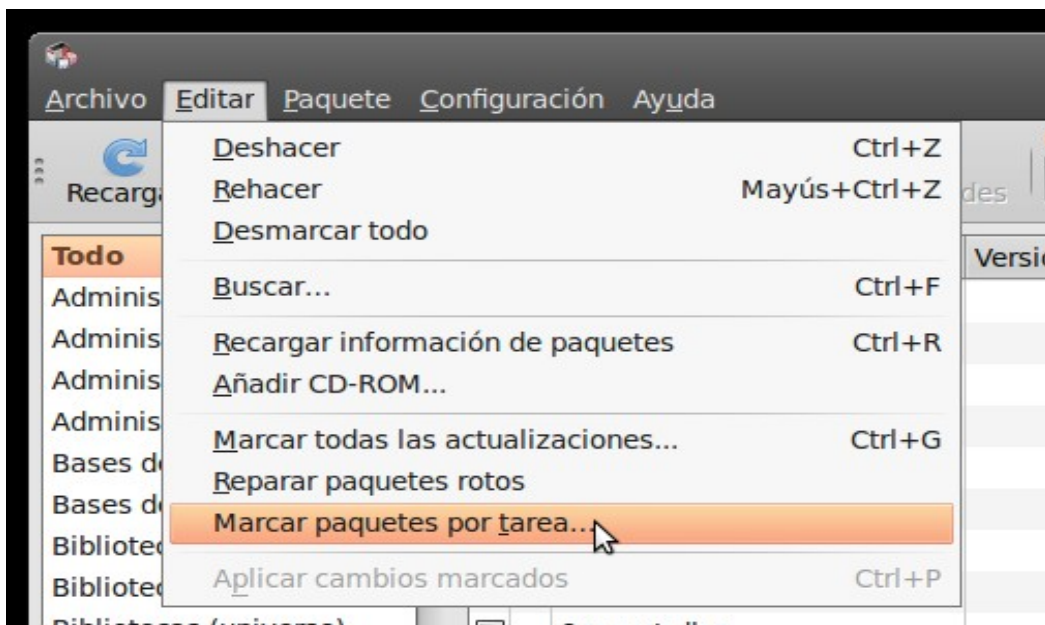
Para más información sobre XAMPP ir a <http://es.wikipedia.org/wiki/XAMPP>

Bién, nosotros vamos a trabajar con LAMP y lo vamos a instalar en unos sencillos pasos mediante el gestor de paquetes synaptic:

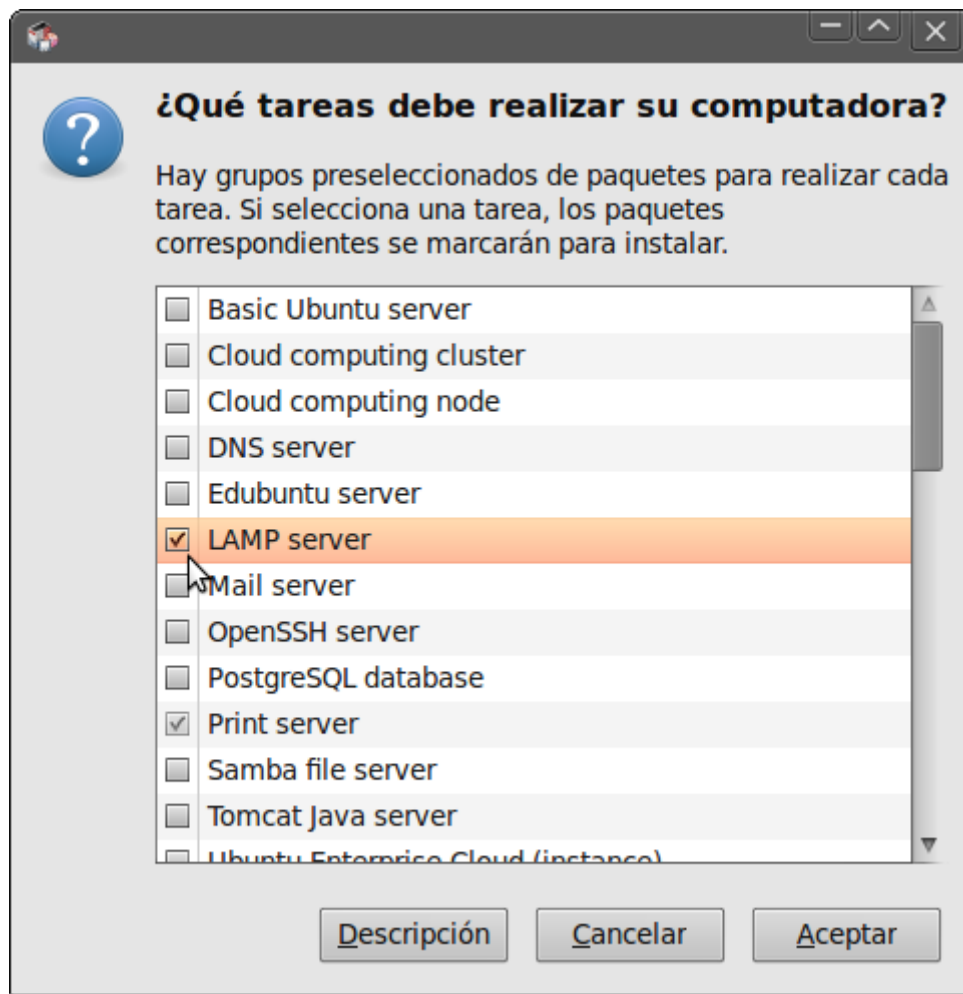
Paso 1 (sistema - Administración - Gestor de Paquetes Synaptic) Nos pedirá la contraseña del administrador.



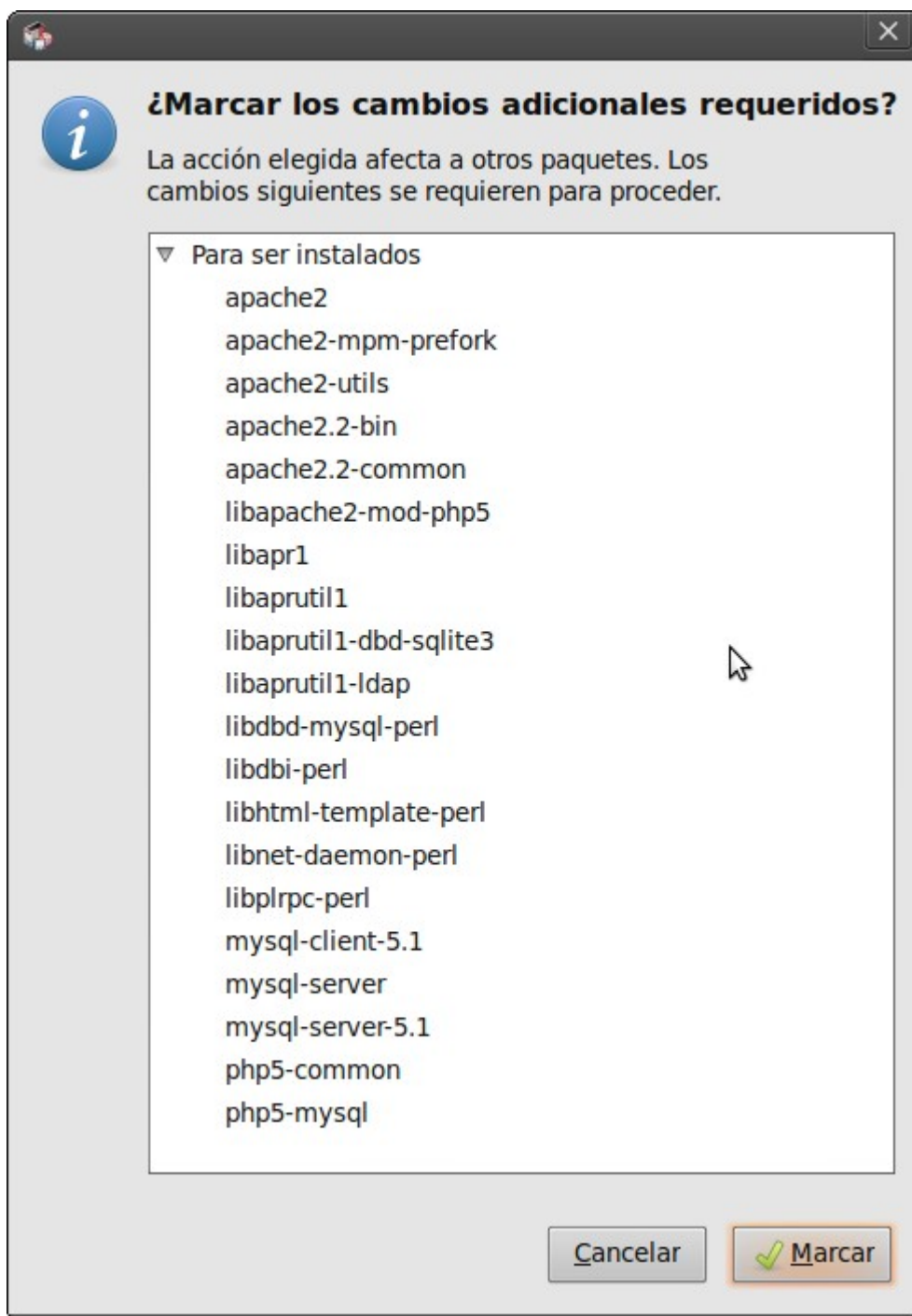
Paso 2 una vez abierto el gestor de paquetes nos vamos a (Editar - Marcar paquetes por tarea)



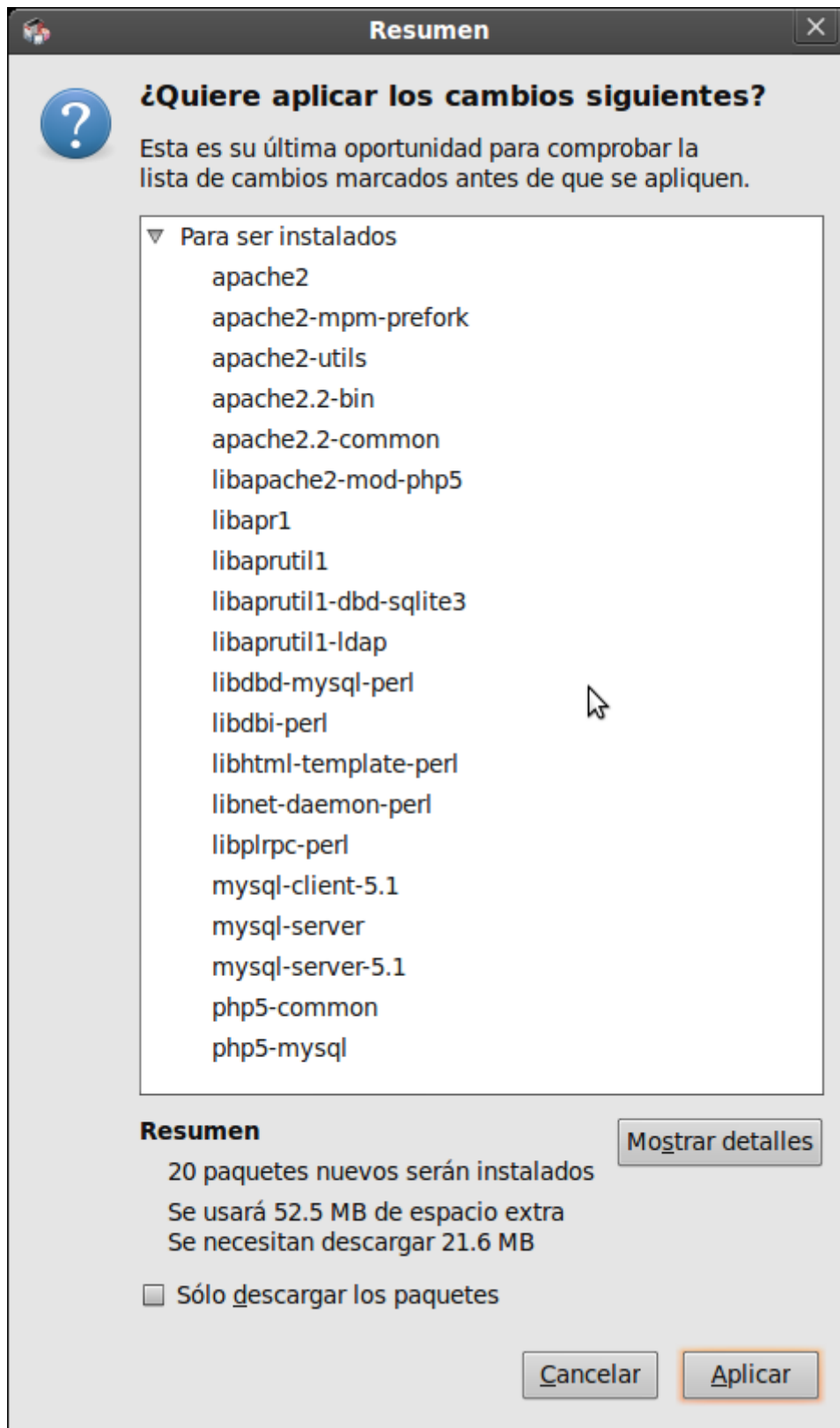
Paso 3 Nos muestra las tareas que podemos marcar, nosotros queremos LAMP-SERVER, lo marcamos en la casilla a continuación y luego Aceptar.



Paso 4 nos muestra la lista de paquetes que contiene la tarea para instalar y nos pide confirmación para proceder:



Paso 5 y con esto ya casi terminamos con la instalación del servidor LAMP, nos pedirá confirmación para aplicar los cambios, nos muestra un resumen de lo que será descargado y posteriormente instalado en nuestro sistema:



Y así comienza la descarga, al terminar comienza a instalar y el ayudante de MySQL nos pedirá que introduzcamos una contraseña nueva para el *superusuario* (root) quien es el que controlará todas las bases de datos con todos los permisos, **recuerden bien esta contraseña** que uds. Mismos asignarán ya que nos servirá mas adelante, (bueno siempre) XD

por cierto, si ven que la apariencia cambia... digamos que tuve problemas técnicos X(

y aquí es donde seguí el tuto.

Descargando paquetes

Descargando el archivo 12 de 21

Tasa de descarga: 32.5 kB/s - quedan 2min 12s

▼ Mostrar los archivos

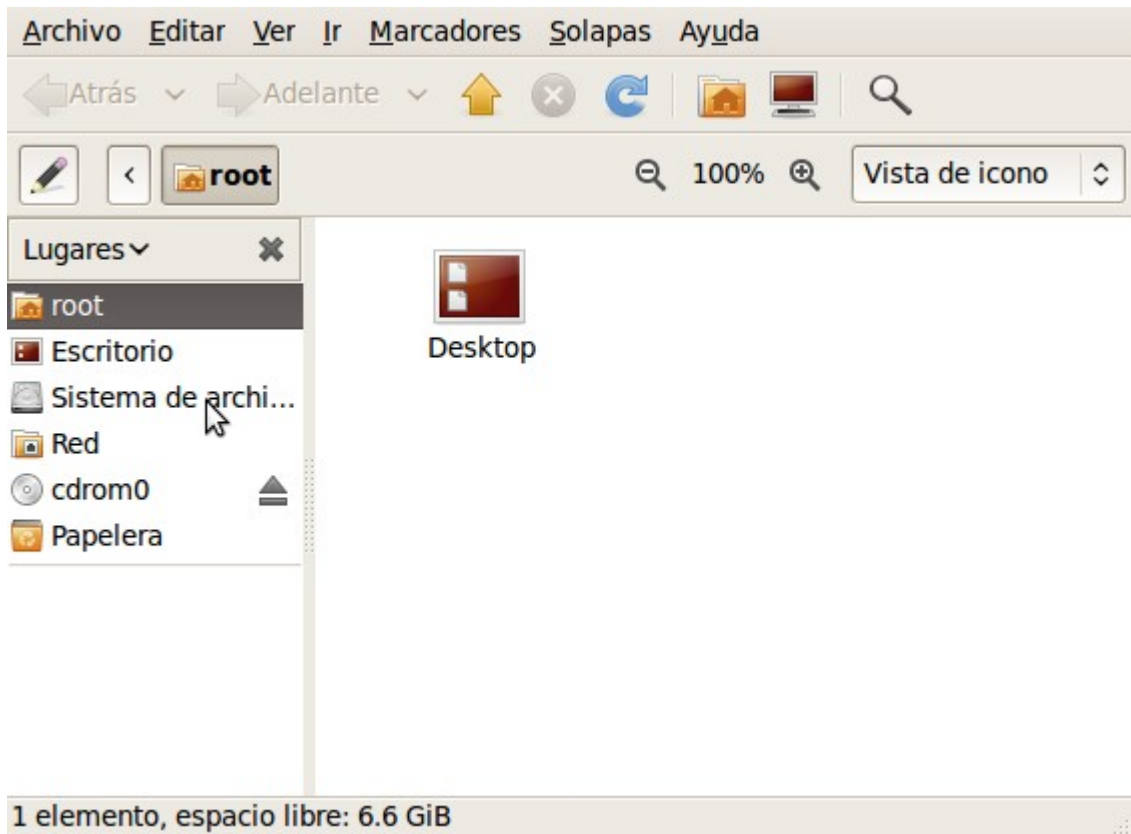
Estado	Tamaño	Paquete	URI
Hecho	3838 kB	mysql-server-core-5.1	http://archive.
Hecho	7186 kB	mysql-server-5.1	http://archive.
Hecho	116 kB	libapr1	http://archive.
Hecho	91.1 kB	libaprutil1	http://archive.
Hecho	27.5 kB	libaprutil1-dbd-sqlite3	http://archive.
Hecho	25.2 kB	libaprutil1-ldap	http://archive.
44%	1309 kB	apache2.2-bin	http://archive.

Cancelar

Ok, una vez instalado nuestro servidor local vamos a proceder a hacer unas pequeñas configuraciones, primero debemos saber que al momento de instalar estos paquetes se han creado nuevos directorios en nuestro sistema, el que mas nos interesa es /www que se encuentra en /var/www para ir ahi con permisos de escritura abrimos la terminal y escribimos:

sudo nautilus

(nos pedirá la clave de root y abrirá el administrador de archivos que ya todos conocemos)



Una vez ingresemos en él con permisos de root nos vamos a la carpeta /var/www y dentro de ella vamos a crear un pequeño archivo con extensión *.php que quedaría así: **phpinfo.php** con el siguiente contenido:

```
<?php
```

```
phpinfo();
```

```
?>
```

Lo guardamos en /var/www luego nos vamos al navegador y escribimos <http://localhost/phpinfo.php>

Esto nos mostrará toda la info necesaria, lo que muestra es algo como esto:

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

← → ↻ × 🏠 📄 http://localhost/phpinfo.php ☆ Google 🔍

Más visitados Getting Started Latest Headlines

phpinfo() +

PHP Version 5.2.10-2ubuntu6.1

System	Linux bjs 2.6.31-14-generic #48-Ubuntu SMP Fri Oct 16 14:04:26 UTC 2009 i686
Build Date	Oct 29 2009 19:02:41
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
additional .ini files parsed	/etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/mysqli.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, data, http, ftp, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed

This server is protected with the Suhosin Patch 0.9.7
Copyright (c) 2006 [Hardened-PHP Project](#) 수호신

This program makes use of the Zend Scripting Language Engine: Powered By

Terminado

Para verificar que nuestro servidor apache esta funcionando OK escribimos `http://localhost/` en el navegador y esto nos arrojará un mensaje como este:

It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Si en algún momento les diera problemas al llamar el `phpinfo.php` en el navegador solo necesitan reiniciar los servicios y listo!

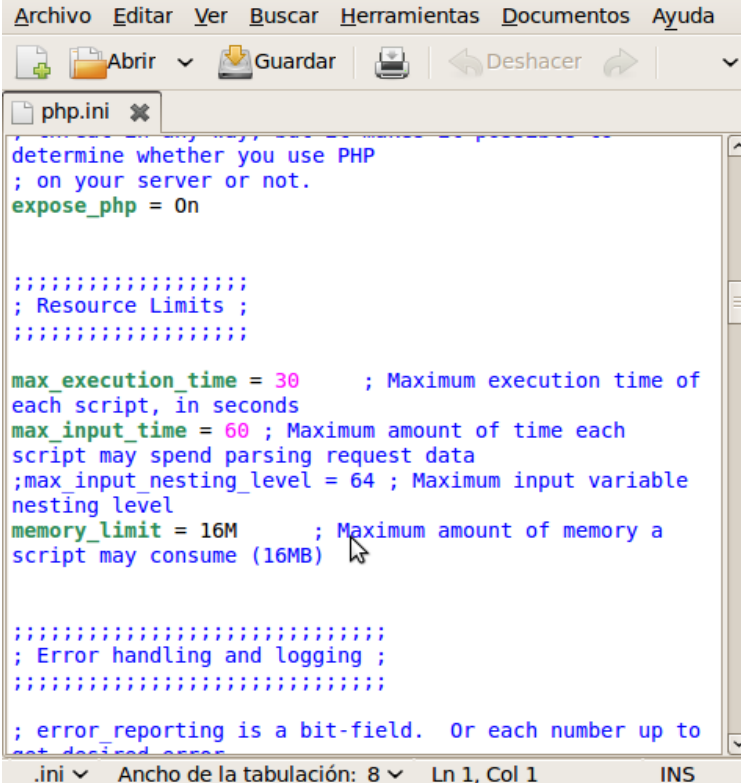
Ejecutar en la terminal:

```
sudo /etc/init.d/apache2 restart
```

```
sudo /etc/init.d/mysql restart
```

Otro posible error (a mí me pasó), es que por ejem. con Drupal me lanzaba un error donde me explicaba que necesita almacenar info en un archivo x pero que el límite del servicio era 16MB, googleando y con un poco de ayuda de @kaasi logré solucionar el problema de la sig. Manera:

Cuando llamamos al `phpinfo.php` en el navegador veremos que ahí nos muestra unos directorios, el que nos interesa es el **php.ini** que se encuentra en `/etc/php5/apache2` Nos vamos al directorio (como root) y editamos el archivo con gedit; buscamos la línea que diga `memory_limit = 16M` y lo cambiamos por 128M



```
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Abrir  Guardar  Deshacer
php.ini
determine whether you use PHP
; on your server or not.
expose_php = On

;;;;;;;;;;;;;;;;;;;;;;;;;
; Resource Limits ;
;;;;;;;;;;;;;;;;;;;;;;;;;

max_execution_time = 30    ; Maximum execution time of
each script, in seconds
max_input_time = 60 ; Maximum amount of time each
script may spend parsing request data
;max_input_nesting_level = 64 ; Maximum input variable
nesting level
memory_limit = 16M        ; Maximum amount of memory a
script may consume (16MB)

;;;;;;;;;;;;;;;;;;;;;;;;;
; Error handling and logging ;
;;;;;;;;;;;;;;;;;;;;;;;;;

; error_reporting is a bit-field. Or each number up to
get desired error
```

Con esto ya tenemos todo para poder trabajar con nuestro CMS en Ubuntu, solo creamos la respectiva Base de Datos y creamos usuarios para las mismas, o bien pueden usar a root para todas ;) **No olviden colocar siempre sus carpetas en /var/www para luego llamarlas con el navegador.**

Para crear una base de datos primero nos conectamos a MySQL (por medio de la terminal)

```
sudo mysql -u root -p
```

Con éste comando nos conectamos a MySQL primero nos va a pedir la clave del sudo osea superusuario del sistema, luego nos pedirá la clave de root para MySQL (la que le dimos al momento de la instalación a mysql).

Si todo anda bien y se lograron conectar verán algo como esto:

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
Your MySQL connection id is 35
```

```
Server version: 5.1.37-1ubuntu5 (Ubuntu)
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql>
```

Creamos la base de datos:

```
CREATE DATABASE ejemplo;
```

Creamos el usuario:

```
CREATE USER usuario@localhost IDENTIFIED BY 'contraseña';
```

Damos permisos al usuario en la nueva base de datos 'ejemplo':

```
GRANT ALL PRIVILEGES ON ejemplo.* TO usuario@localhost IDENTIFIED BY 'contraseña';
```

Aplicamos cambios:

```
flush privileges;
```

```
quit (para salir de mysql)
```

Con ésto ya le damos los datos al CMS y a trabajar!

Espero les haya servido de mucho este tutorial y también que lo hayan logrado hacer todo sin ningún problema.

Para izalug
Jorge Rojas, (tochi)