**Servidor LAMP en dos servidores (HTTPS + Web Avanzada)**

****

Version 1.0Revision: V1 Toni, Fecha 06/02/24

| **Revisor** | **Estado** | **Fecha** | **Notas** |
| --- | --- | --- | --- |
| [tonisuredacardona0@gmail.com](mailto:tonisuredacardona0@gmail.com) | **Aprobada** | **15/02(24** | Configuración de lamp en dos servidores |
| [Hasna Zalmat](mailto:hasnaehasnae063@gmail.com) | **Aprobada** | **15/02/24** | Actualización de detalles. |
|  | **No iniciada** |  |  |

**Índice**

[**Descripción 3**](#_heading=h.30j0zll)

[Finalidad del servicio : 3](#_heading=h.1fob9te)

[Máquinas en las que se aloja: 3](#_heading=h.3znysh7)

[Paquetes usados y versión de los mismos: 4](#_heading=h.2et92p0)

[**Configuración 4**](#_heading=h.tyjcwt)

[Configuraciones previas 4](#_heading=h.3dy6vkm)

[Configuración del servicio 5](#_heading=h.1t3h5sf)

[**Verificación funcionmiento 11**](#_heading=h.i7yarf7swoor)

[Pruebas y comprobaciones 11](#_heading=h.k0irvvxzl4z1)

[Puertos 14](#_heading=h.uhm1d77t8dhm)

[Proceso y estado del servicio 14](#_heading=h.yzcykjowmz1d)

[**Alternativas 15**](#_heading=h.jbdbudeae5ey)

[**Fuentes: 15**](#_heading=h.xch8vzvcro5m)

**Descripción**

**Finalidad del servicio :**

Separar los servidores Apache + PHP del servidor MySQL ofrece ventajas como un mejor rendimiento al distribuir la carga de trabajo, escalabilidad más flexible al permitir aumentar recursos de manera independiente para cada componente, mayor seguridad al reducir la superficie de ataque y facilitar la gestión y mantenimiento al especializar equipos en áreas específicas. Esta separación optimiza la eficiencia del sistema al permitir que cada servidor se enfoque en tareas específicas, mejorando así la experiencia del usuario y la robustez de la aplicación web.

**Máquinas en las que se aloja:**

Maquina 1 (Servidor con apache y PHP)

| S.O | Ubuntu server 22.04 |
| --- | --- |
| Hostname |  |
| Nucleos | 2 |
| RAM | 1 GB |
| Configuracion RED |  |
| VMBR | VMBR1 |

Maquina 2 (Servidor con MYSQL)

| S.O | Ubuntu server 22.04 |
| --- | --- |
| Hostname |  |
| Nucleos | 2 |
| RAM | 1 GB |
| Configuracion RED |  |
| VMBR | VMBR1 |

**Paquetes usados y versión de los mismos:**

mysql-server 8.0

php-8.1.2

apache 2.4.62

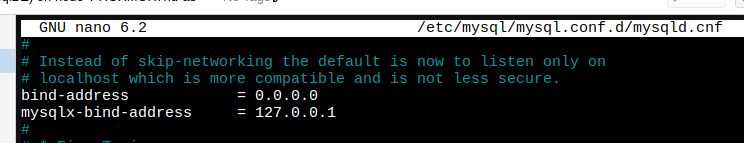
**Configuración**

**Configuraciones previas**

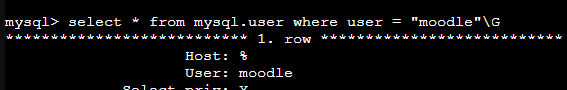
Primero actualizaremos los repositorios, a continuiacion, instalaremos los diferentes paquetes, en este caso mysql en nuestro BE, y PHP + Apache en nuestro FE.

**Configuración del servicio**

Iremos a la configuracion del mysql y permitiremos las conexiones desde cualquier ip poniendo 0.0.0.0 en bind-address.

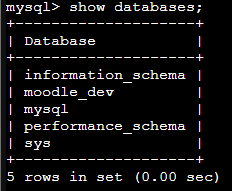
****

A continuacion crearemos un usuario y daremos permisos para que la aplicacion pueda acceder al MYSQL.





Solo nos quedara crear la BBDD que usara nuestra aplicacion.

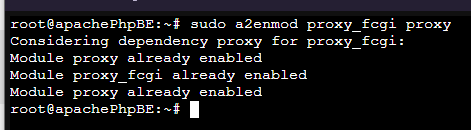


Ahora en el servidor FE instalaremos PHP, multiples funciones suyas y apache2.





Habiliraremos el modulo de proxy y proxy\_fcgi(necesario para el phpFPM)



<VirtualHost \*:80>: Esto indica que este VirtualHost va a escuchar en todas las direcciones IP (\*) en el puerto 80, que es el puerto predeterminado para las conexiones HTTP.

ServerName toni.com: Este es el nombre del servidor principal que este VirtualHost va a manejar.

ServerAlias www.toni.com: Esto especifica un alias para el servidor, es decir, si alguien accede al sitio web usando www.toni.com en lugar de toni.com, también será dirigido a este VirtualHost.

DocumentRoot /home/moodle/public\_html: Este es el directorio raíz del sitio web. Apache buscará los archivos del sitio web dentro de este directorio.

<Directory /home/moodle/public\_html>: Aquí se especifica el directorio que será afectado por las directivas de configuración que siguen. En este caso, las directivas se aplican al directorio /home/moodle/public\_html.

Options -Indexes +FollowSymLinks +MultiViews: Estas son las opciones de configuración para el directorio.

-Indexes: Evita que los directorios muestren automáticamente su contenido si no hay un archivo de índice.

+FollowSymLinks: Permite que Apache siga enlaces simbólicos dentro del directorio.

+MultiViews: Habilita la búsqueda de archivos si no se especifica la extensión completa del archivo en la URL.

AllowOverride All: Esto permite que los archivos de configuración local (como .htaccess) anulen la configuración global definida en el archivo de configuración principal de Apache.

Require all granted: Esto especifica que todas las solicitudes al directorio están permitidas.

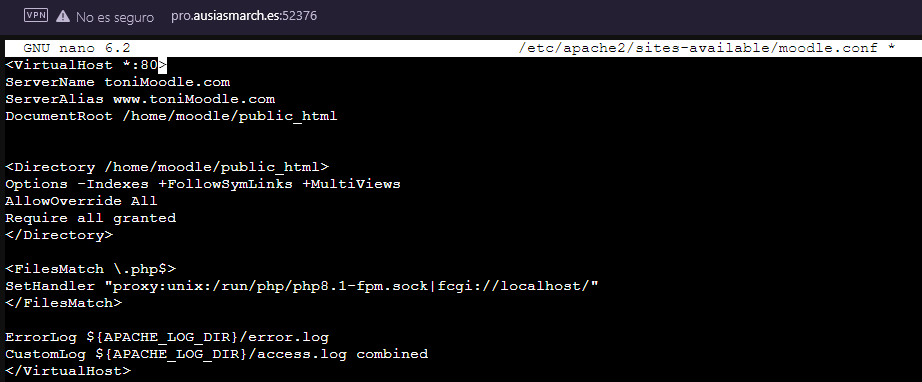
<FilesMatch \.php$>: Esta sección aplica la configuración a los archivos que coincidan con el patrón especificado, en este caso, archivos con extensión .php.

SetHandler "proxy:unix:/run/php/php8.1-fpm.sock|fcgi://localhost/": Esta directiva indica cómo manejar las solicitudes de archivos PHP. En este caso, está configurado para pasar las solicitudes PHP al daemon FastCGI a través del socket UNIX.

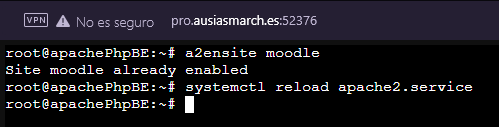
ErrorLog ${APACHE\_LOG\_DIR}/error.log: Especifica la ubicación del archivo de registro de errores para este VirtualHost.

CustomLog ${APACHE\_LOG\_DIR}/access.log combined: Especifica la ubicación del archivo de registro de acceso para este VirtualHost.

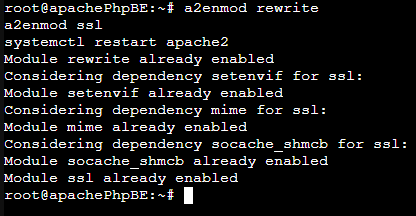
</VirtualHost>: Cierra la configuración de VirtualHost. Todo lo que está dentro de las etiquetas <VirtualHost> y </VirtualHost> se aplica solo a ese host virtual específico.



Habilitaremos la configuracion y recargaremos el servicio del Apache.



Habilitaremos los modulos de rewrite y ssl y reiniciaremos otra vez el servicio



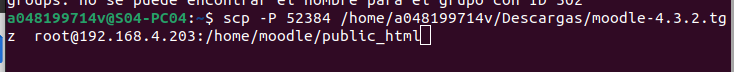
Crearemos un usuario llamado moodle con un home, y daremos permisos 711 a este documento



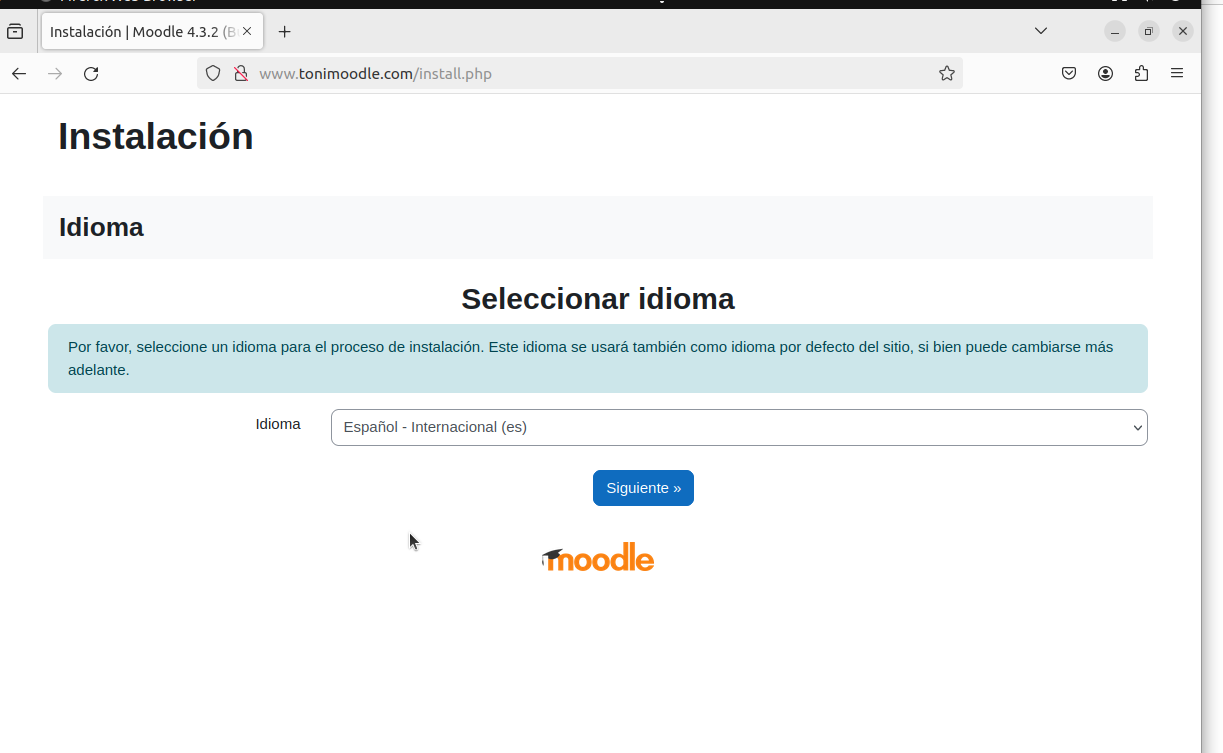
Ahora por el servicio SSH SCP pasaremos el comprimido del instalador del moodle a nuestro directorio public\_html que conseguiremos e esta pagina <https://download.moodle.org/releases/latest/>

Para eso tenemos que habilitar una regla SSH en el firewall.

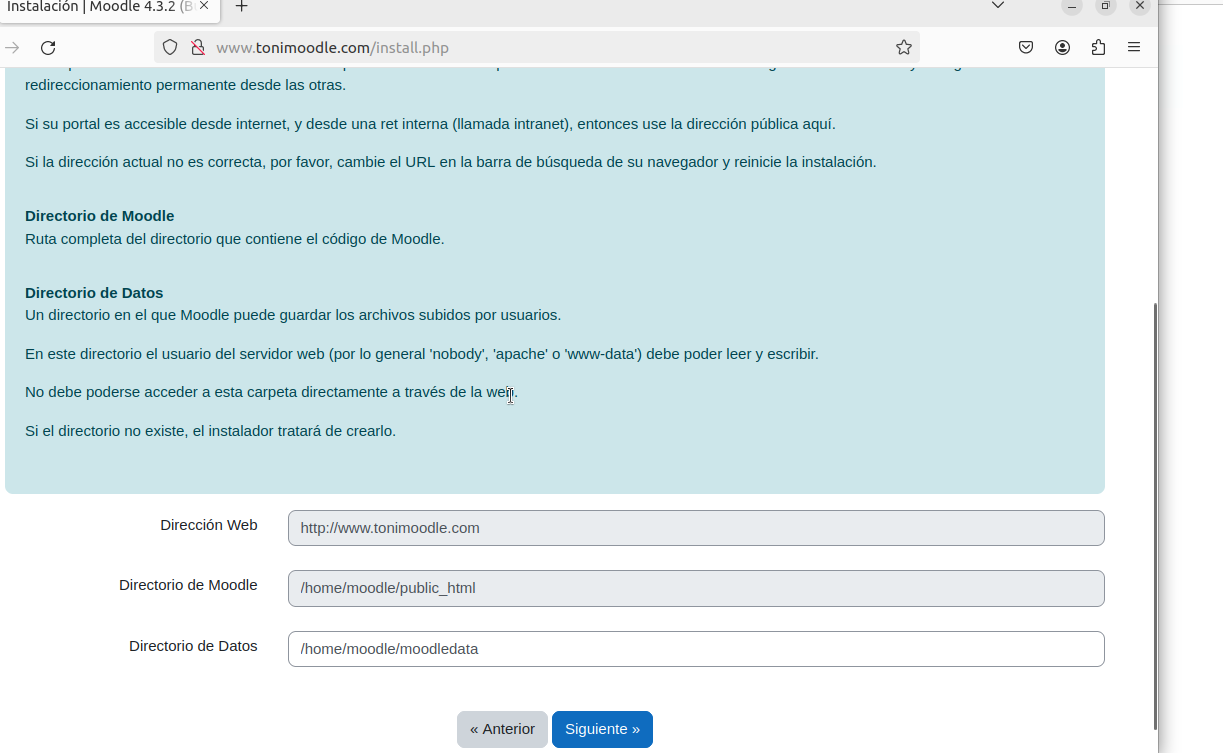




Asi podremos iniciar nuestra instalacion eligiendo el idioma en un pagina en la misma red editando el archivo hosts.



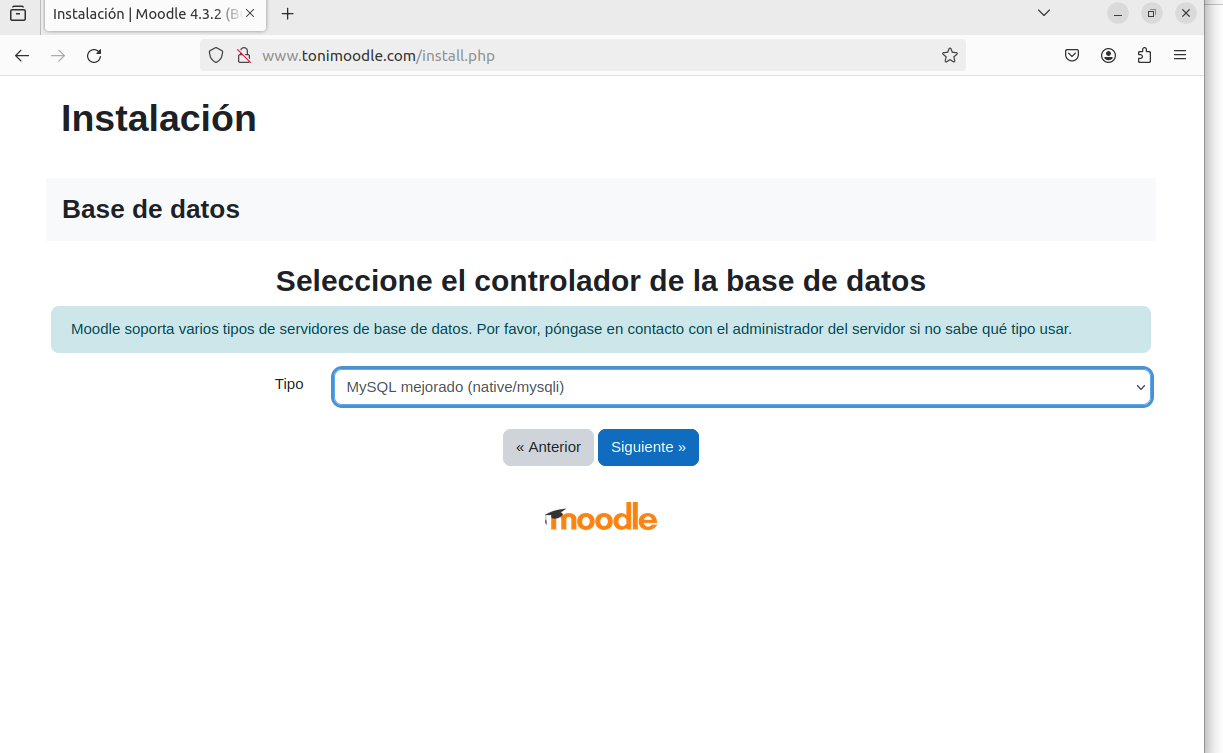
Ahora, nos indica que moodle necesita una carpeta para los datos, la crearemos y daremos permisos a Apache para continuar con la instalacion



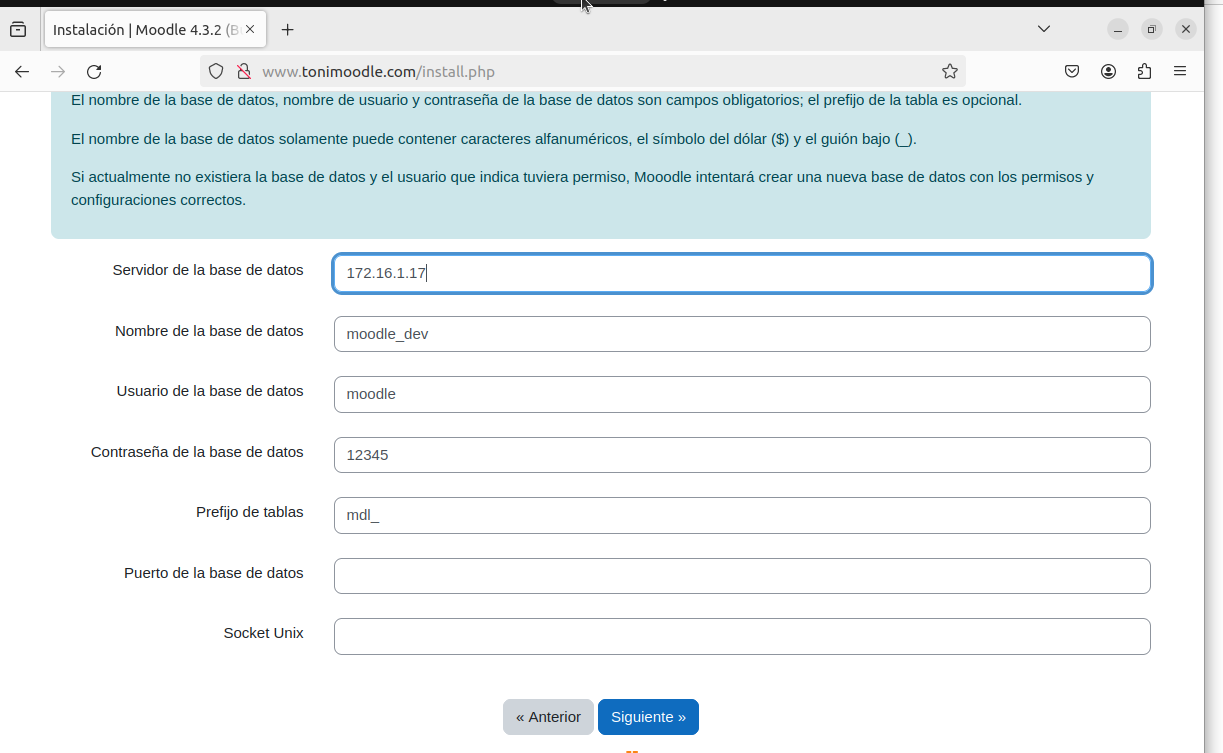




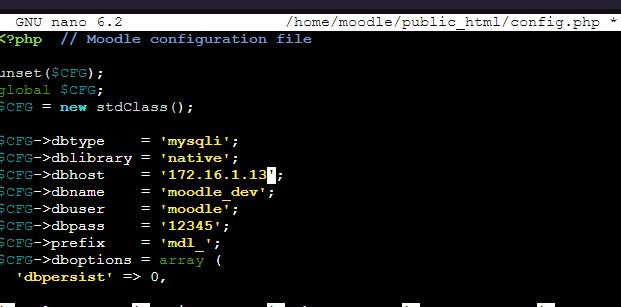
Elegimos nuestro sistema de BBDD.



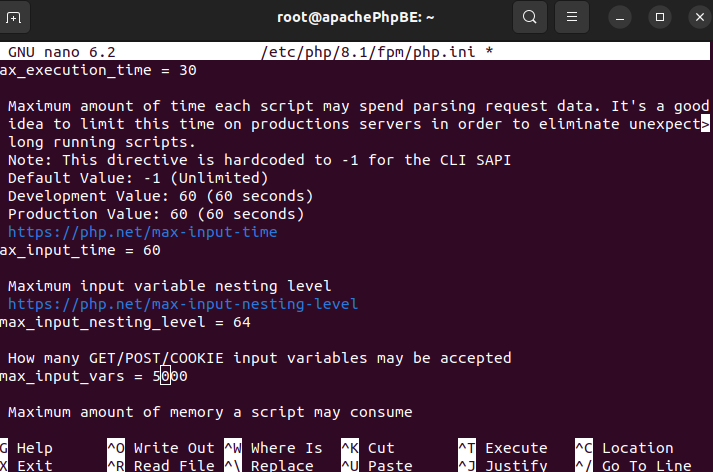
Aqui rellenaremos los datos de nuestra BBDD con la IP, la BBDD que creamos, el usuario y contraseña , tenemos en cuenta que la IP de la BBDD cambio y no es el visto en la captura, mas adelante se ve de manera correcta.



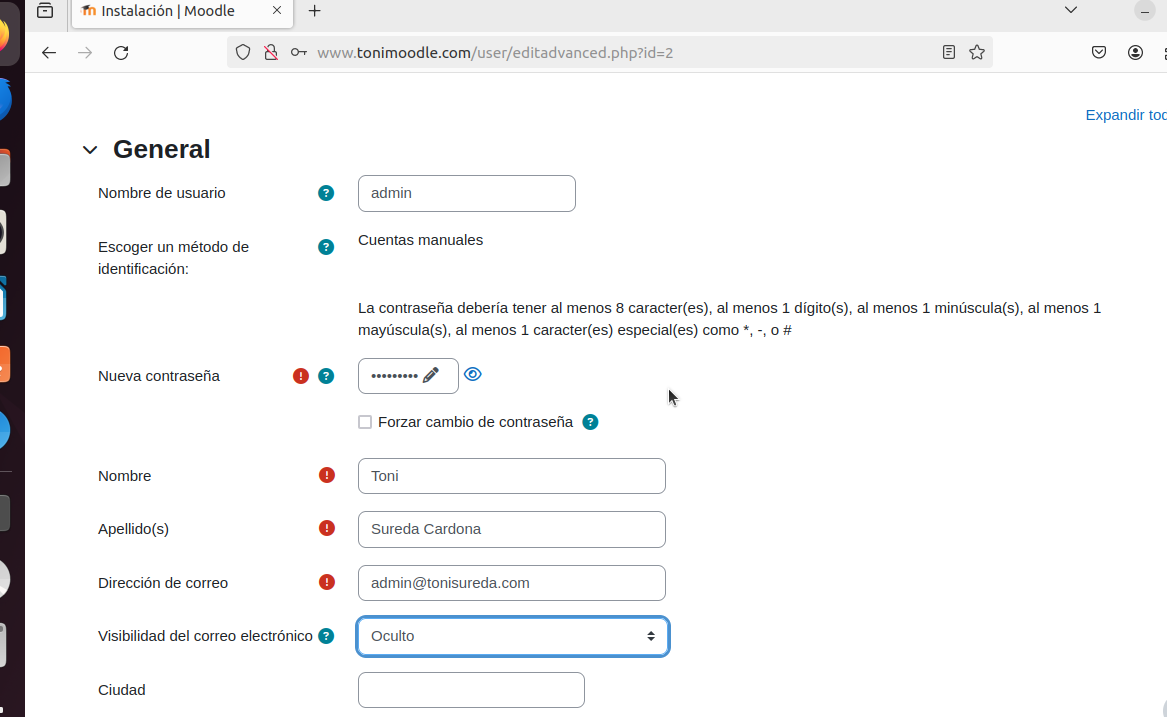
Ahora crearemos el archivo config.php con la configuración que nos da moodle en la carpeta public\_html



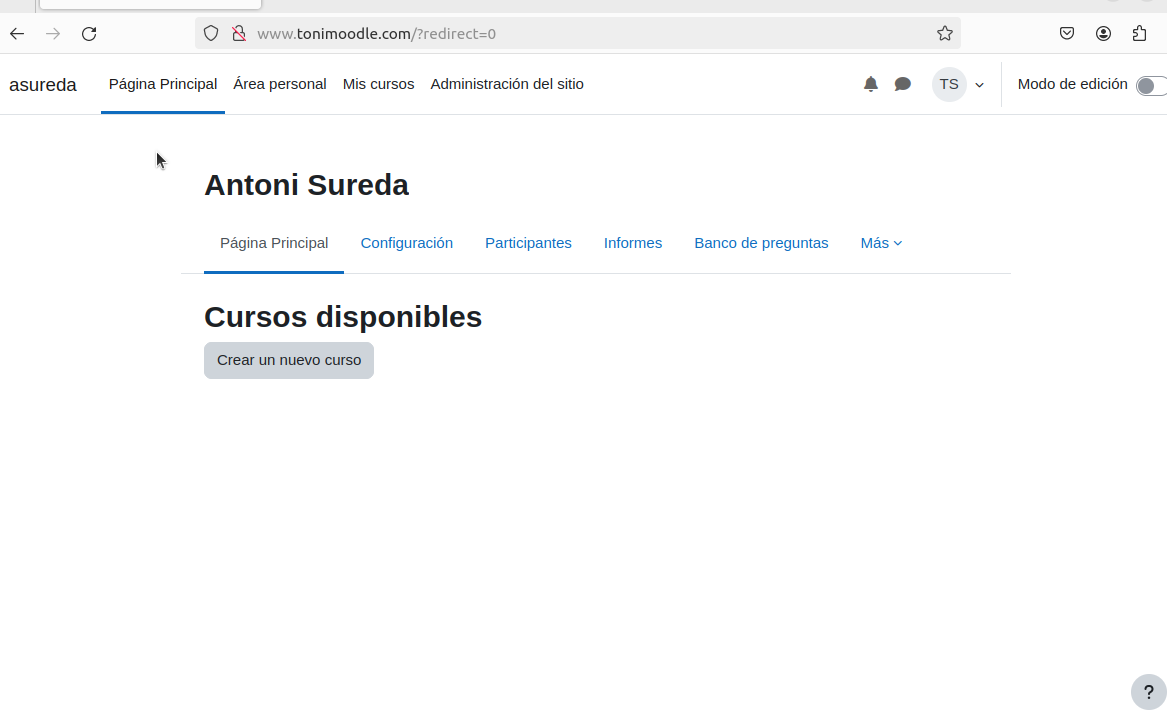
Para continuar la instalacion, cambiaremos el valor max\_input\_vars a un valor igual o superior a 5000 en la congiruacion del php-fpm



Crearemos nuestro usuario

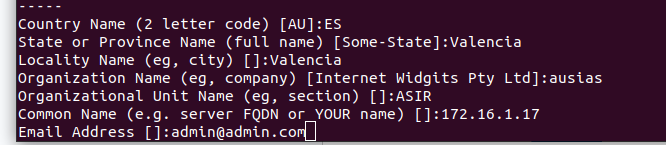


Y por ultimo veremos nuestra pagina creada.

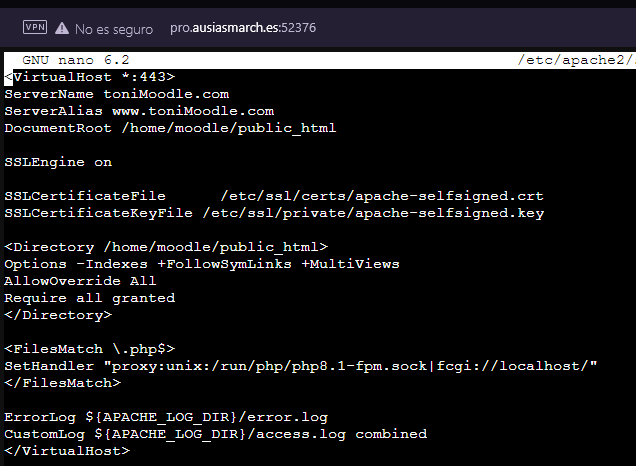


A continuación aparte de configurar los certificados SSL, configuraremos el servicio para poder acceder desde el exterior. Crearemos unos certificados autofirmados para nuestra pagina HTTPS.

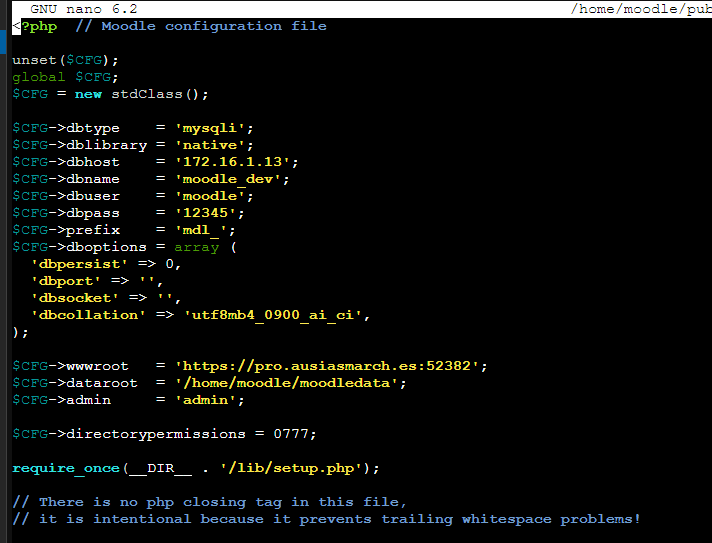




Cambiaremos la configuracion el puerto al 443, indicaremos que el sistema SSL esta activo con SSLEngine ON. Despues indicaremos el archivo llave SSL y el certificado con SSLCertificateFile y SSLCertificateKeyFile.



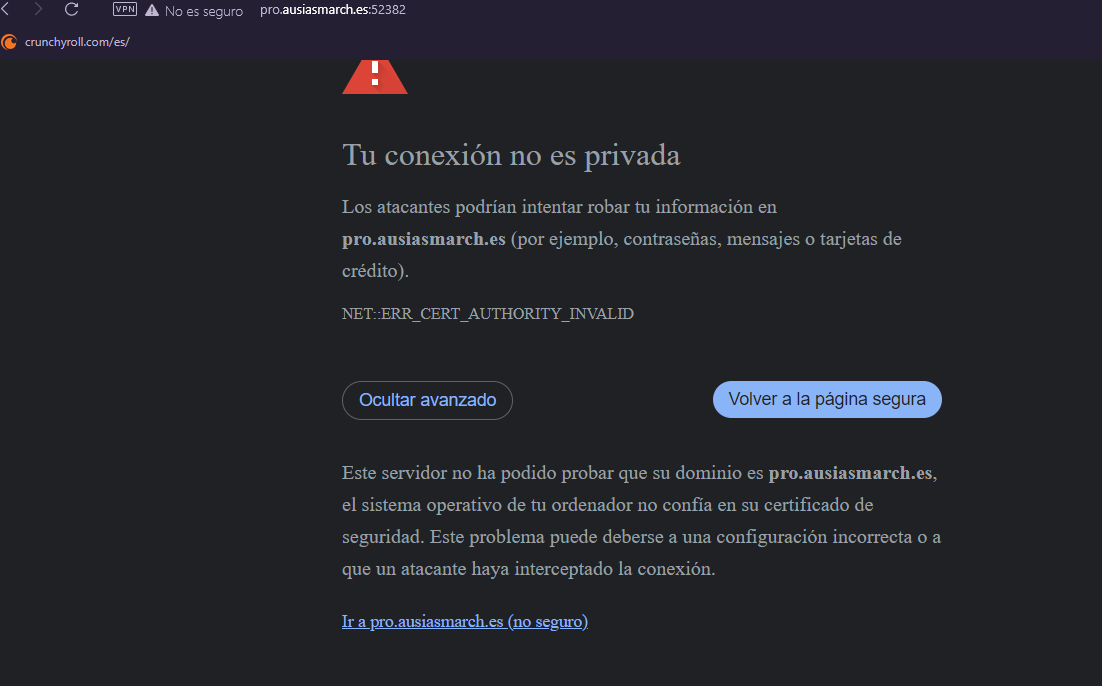
Cambiaremos el wwwroot con la IP externa que tenemos para acceder con HTTPS a nuestro servidor, aparte de crear esta misma norma

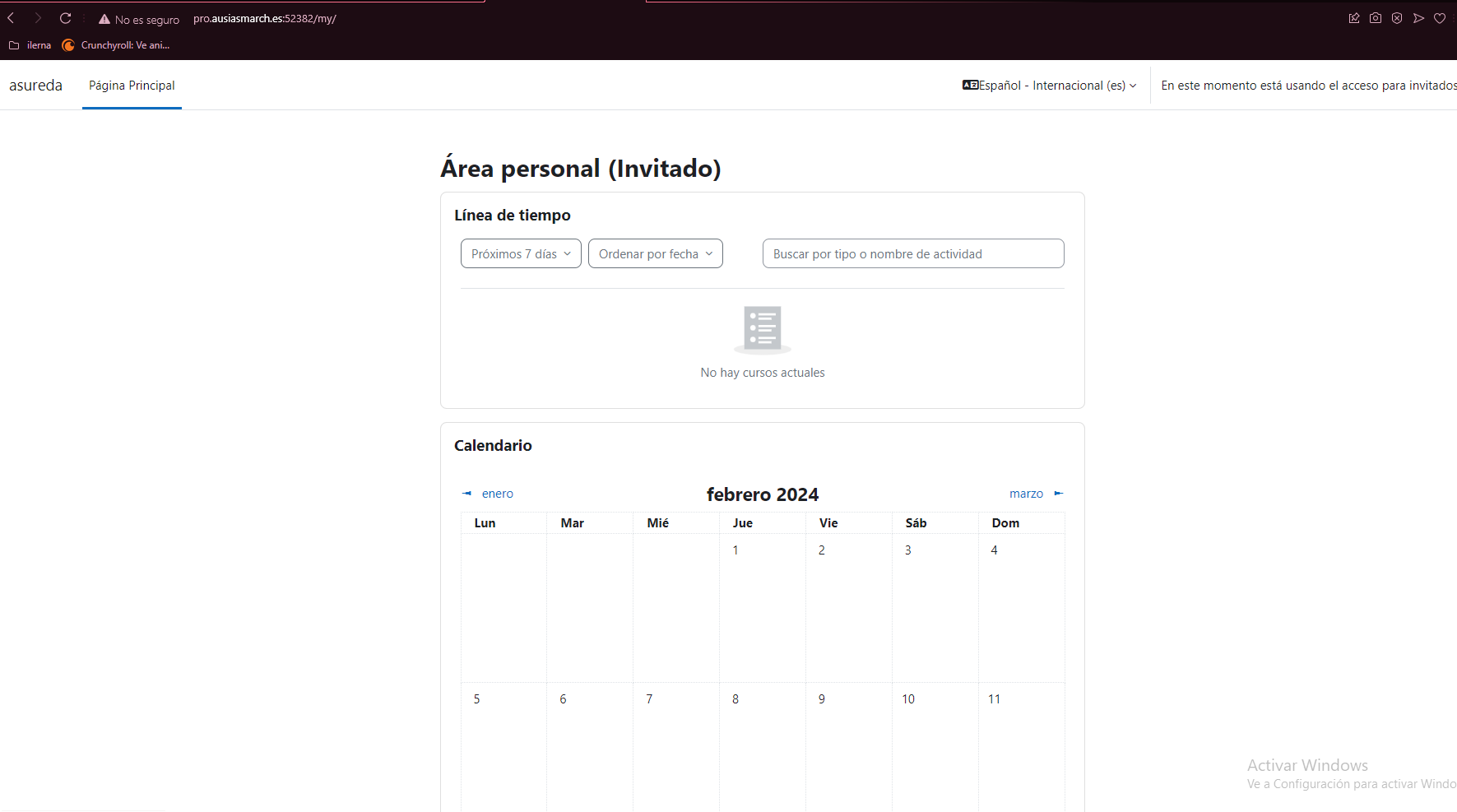


**Verificación funcionmiento**

**Pruebas y comprobaciones**

Entramos a nuestro moodle, veremos que nos da un error porque no puede verificar los certificados, accedemos igualmente y vemos como nos muestra la pagina del moodle.

****



**Puertos**

El servicio HTTPS usa el puerto 443 por defecto.

Servidor BE

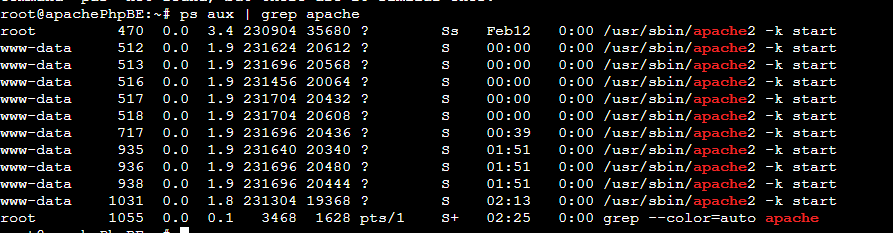


Servidor FE

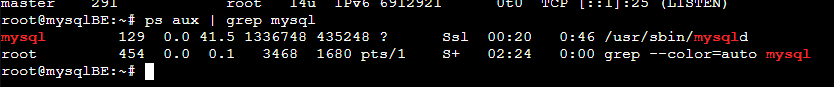
****

**Proceso y estado del servicio**

Servidor FE



Servidor BE



**Alternativas**

Alternativas a Apache para la creacion de servidores FE + BE:

Nginx es una opción sólida y versátil para implementar servidores Front-End (FE) y Back-End (BE). Funciona como un servidor web rápido y ligero que puede servir archivos estáticos de manera eficiente, además de actuar como un potente proxy inverso para enrutar y equilibrar cargas entre varios servidores backend. Su capacidad para manejar solicitudes HTTP, actuar como balanceador de carga y servir como capa frontal para aplicaciones web dinámicas lo convierte en una herramienta fundamental para infraestructuras web modernas, ofreciendo rendimiento, escalabilidad y flexibilidad en una sola solución.

**Fuentes:**

* https://www.webhostingzone.org/blog/how-to-install-moodle-on-ubuntu-22-04
* Documentos de clase