INSTALACIÓN UBUNTU SERVER

Ultima actualización 26/11/2024

**Sumario**

[USED-Ubuntu server 3](#_bookmark0)

[Configuración inicial 3](#_bookmark1)

[Cuentas de administración 16](#_bookmark2)

[Apache 17](#_bookmark3)

[PHP 19](#_bookmark4)

[MySQL 20](#_bookmark5)

[XDebug 21](#_bookmark6)

[Cuentas de desarrollo y hosting virtual 22](#_bookmark7)

[GITHub-Internet 23](#_bookmark8)

[Cuentas de desarrollador 23](#_bookmark9)

[WXED-Windows X 25](#_bookmark10)

[Instalación y configuración inicial de la máquina 25](#_bookmark11)

[Cuentas administradoras y cuenta de desarrollador 26](#_bookmark12)

[Filezilla 27](#_bookmark13)

[NetBeans 28](#_bookmark14)

[Instalación y configuración inicial (plugins) 28](#_bookmark15)

[Creación de proyectos, modificación, borrado, prueba. 28](#_bookmark16)

[Conexión al servidor remoto SFTP. (Almacenamiento local/almacenamiento remoto) 33](#_bookmark17)

[Administración de la base de datos 34](#_bookmark18)

[Conexión al repositorio – versionado 40](#_bookmark19)

[Depuración - Configuración de la ejecución para la depuración 44](#_bookmark20)

[Creación de un proyecto nuevo 45](#_bookmark21)

[PHP Doc 46](#_bookmark22)

[CSS / JS / AJAX / XML / JSON 46](#_bookmark23)

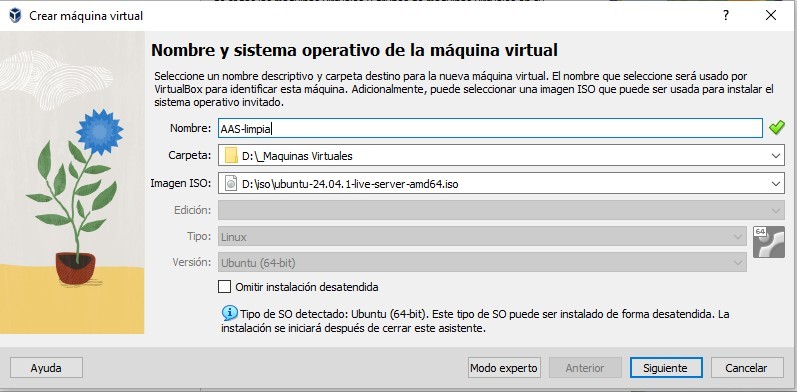
[Paso a explotación-PLESK 47](#_bookmark24)

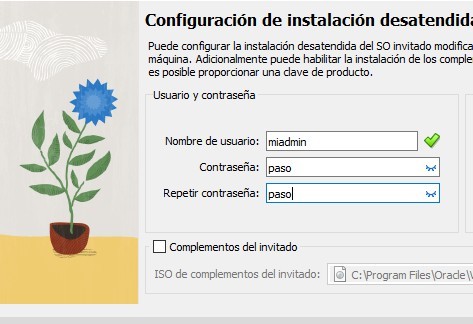
[Utilizando GitHub 47](#_bookmark25)

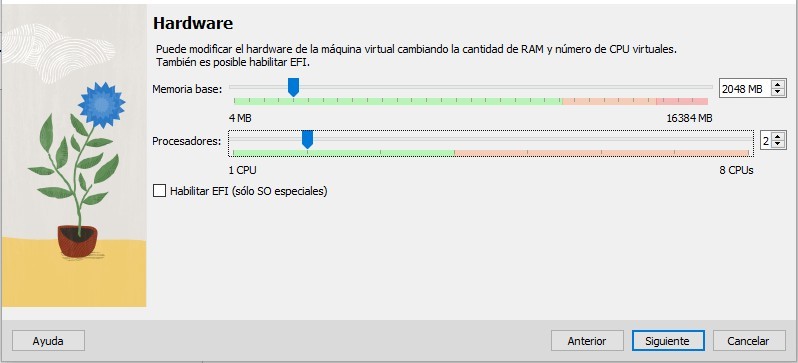
# USED-Ubuntu server

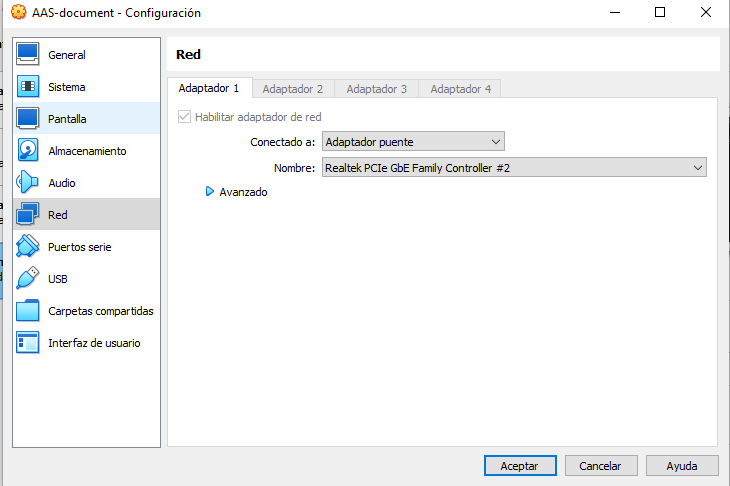
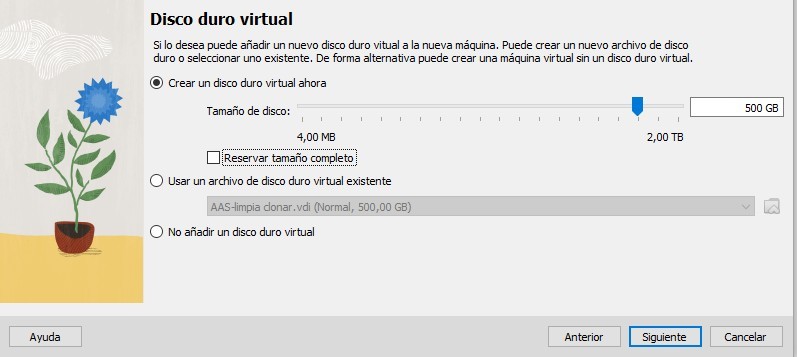
## Configuración inicial

La maquina contara con 2GB de ram y 2 CPU, 500GB de almacenamiento, ademas la configuracion de red estara colocada en adaptador puente

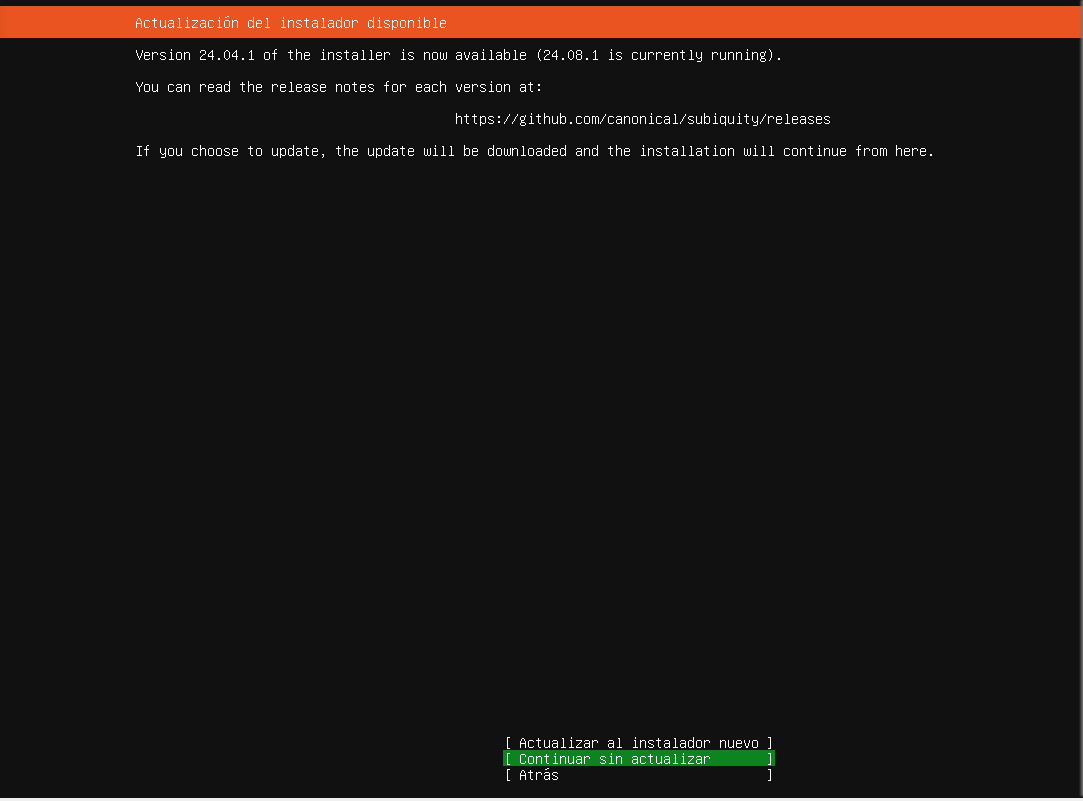




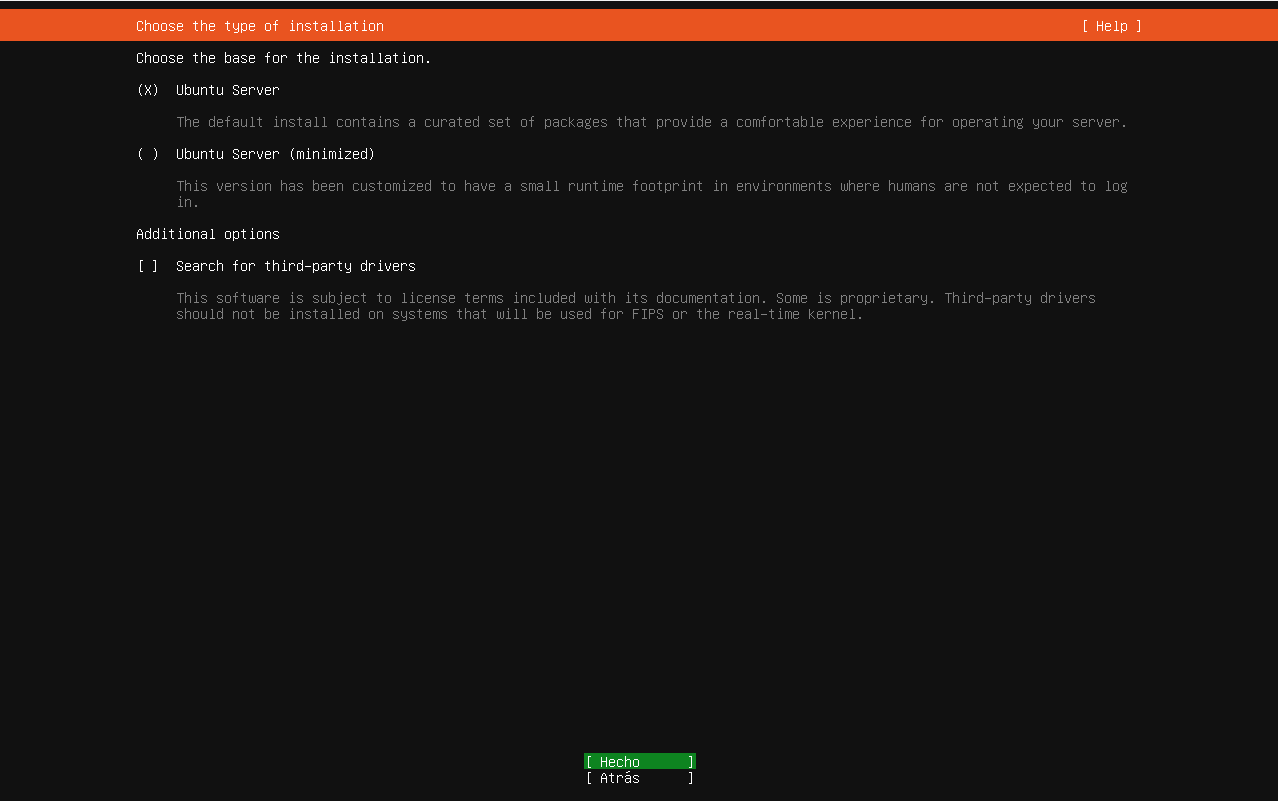




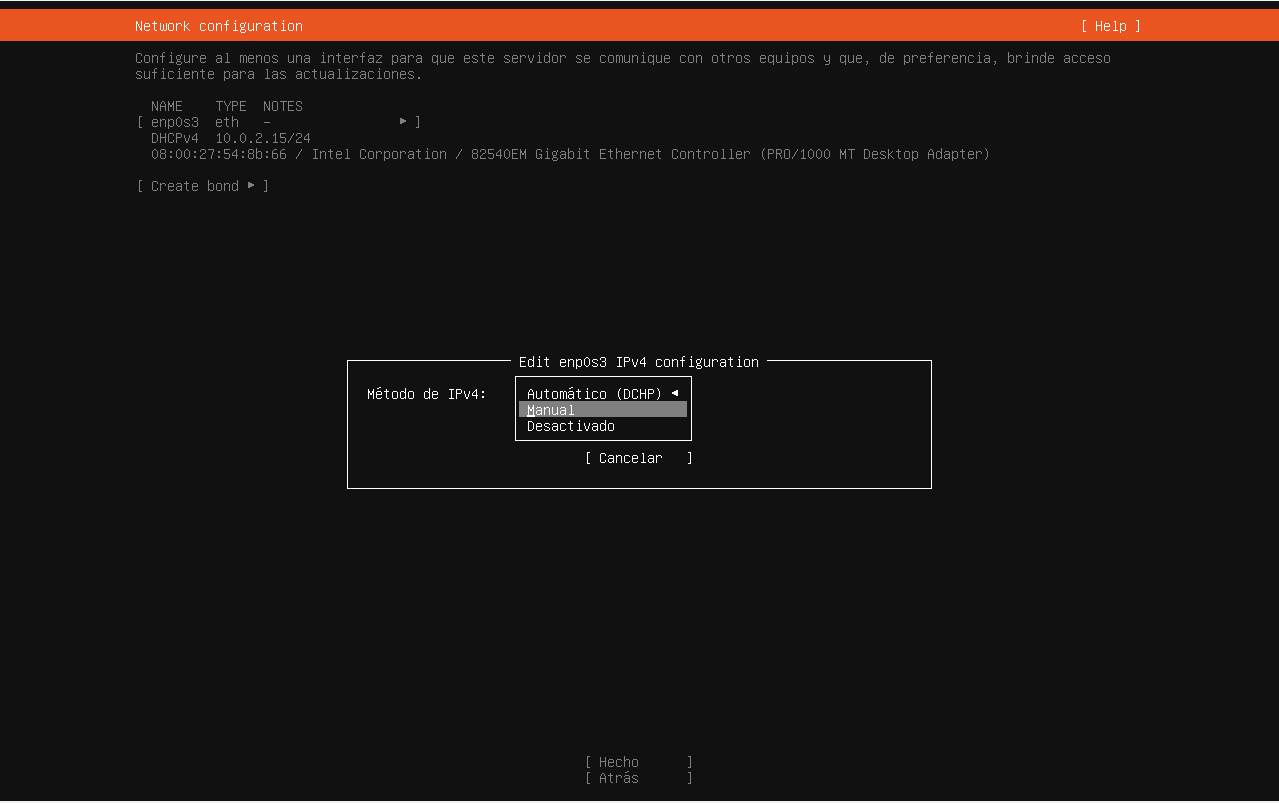
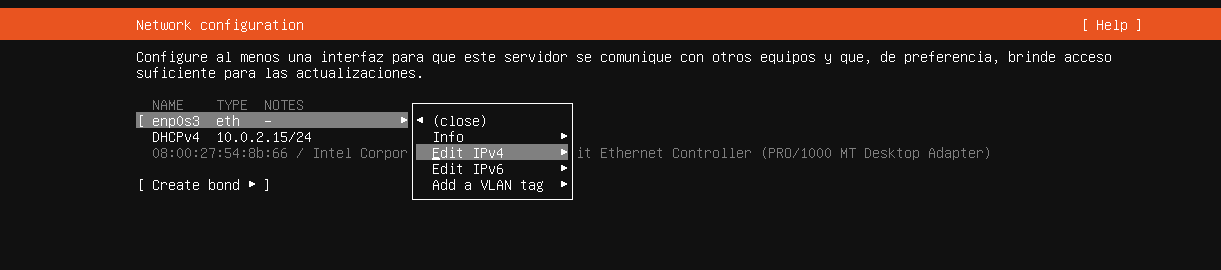
Ya podremos arrancar la maquina, seleccionamos el idioma que queramos. Continuamos sin actualizar



Elegimos instalar Ubuntu server

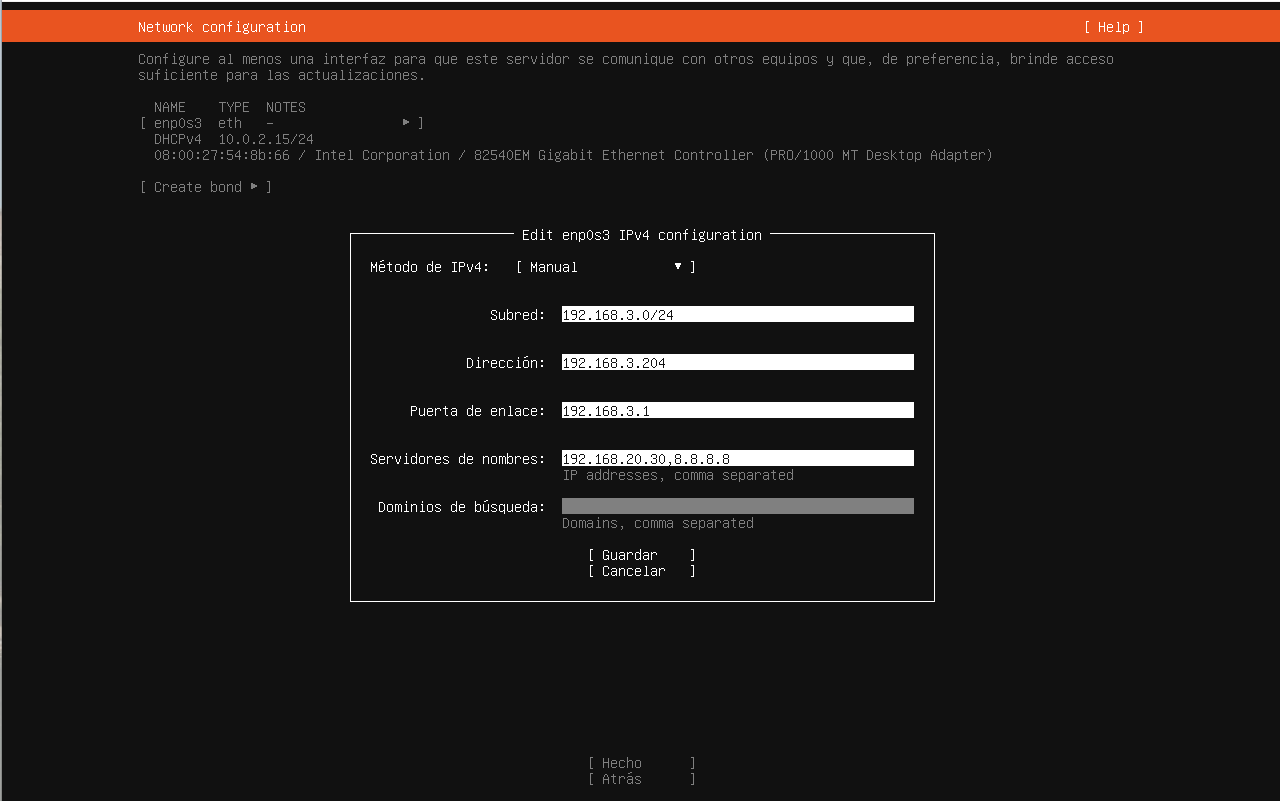


En la configuración de red seleccionaremos una red ipv4 personalizada e introducimos los siguientes valores

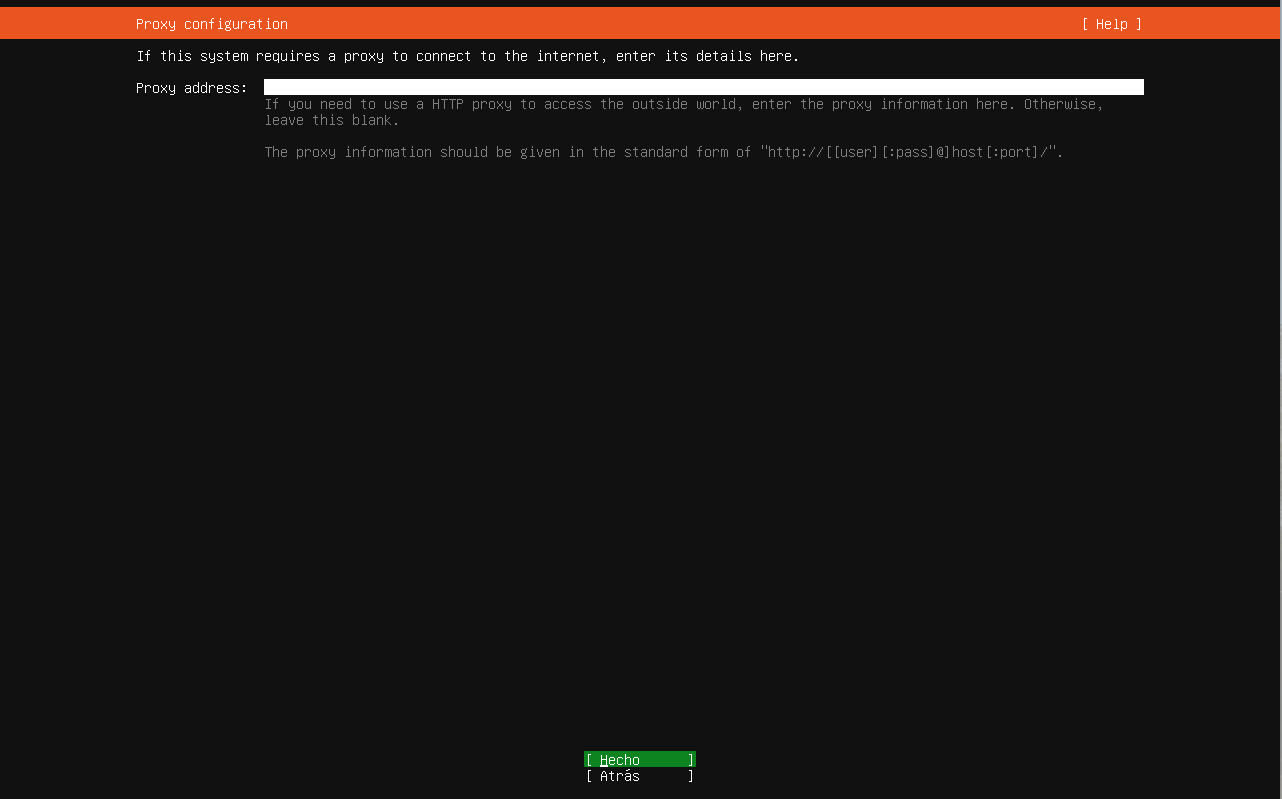


En caso de hacer esto en casa:

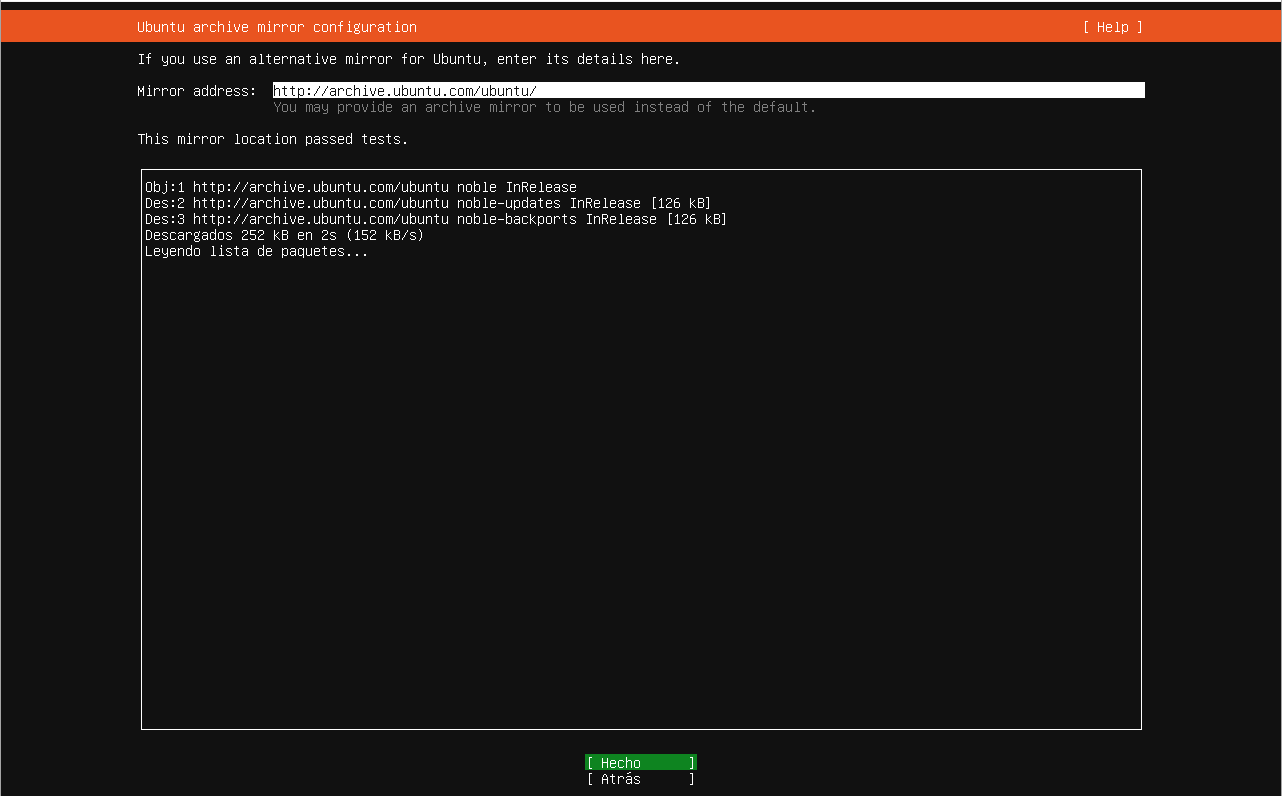
La IP no se pisara con la nuestra, la mascara de red sera de 24 bits usando la ip de la maquina anfitriona (xxx.xxx.xxx.0/24), y la puerta de enlace sera la misma que la maquina anfitriona

Esperaremos a que la opción “continuar sin red” cambie a “continuar” y la seleccionaremos

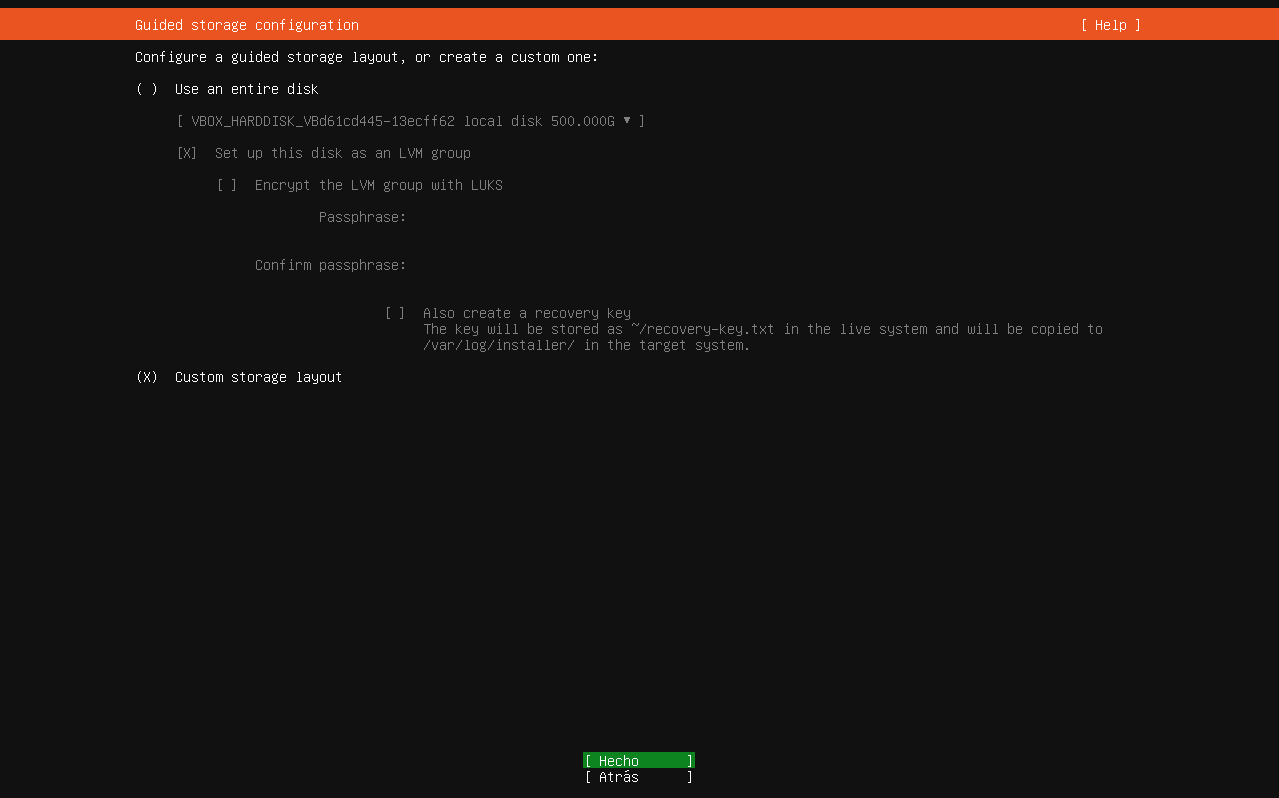
No haremos nada con esto



Ni con esto, simplemente continuamos



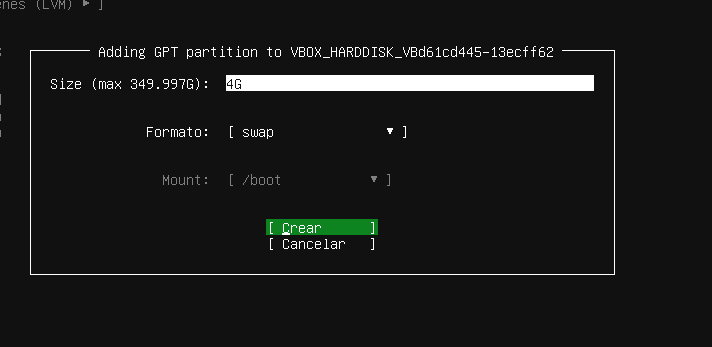
Seleccionaremos la opción “custom store layout”



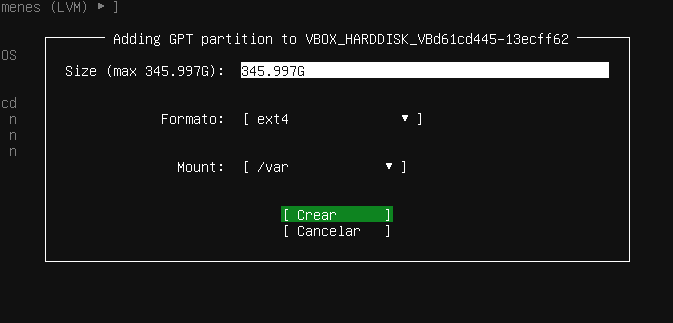
Crearemos 3 particiones, una de 150GB para el sistema



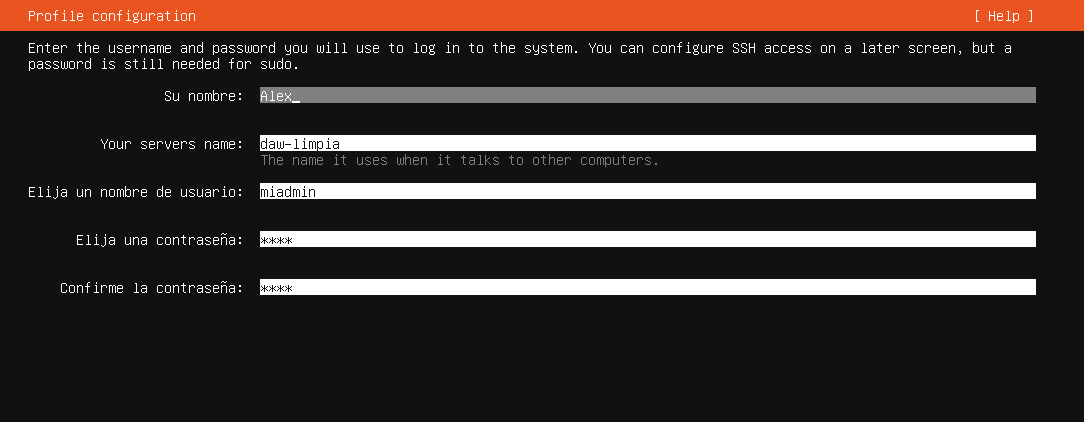
Una de 4GB de tipo swap



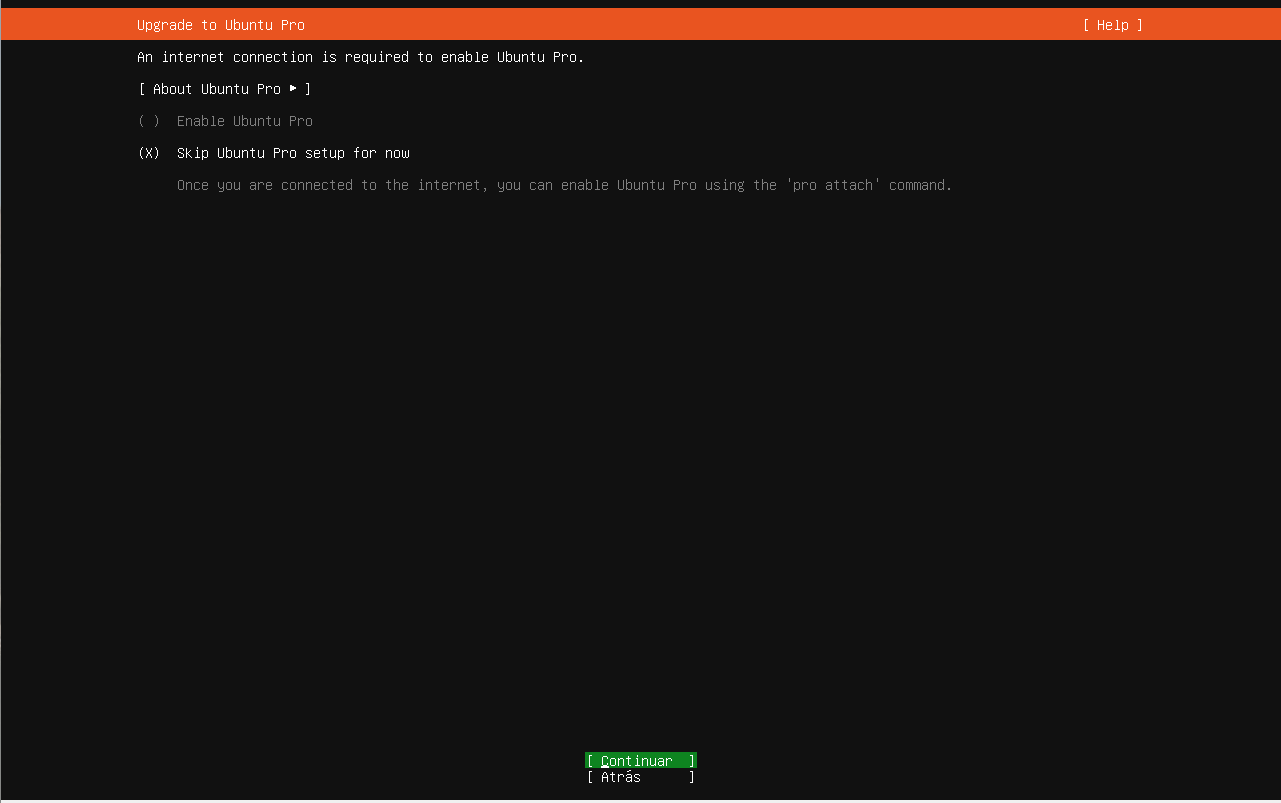
Y una con el espacio restante para /var



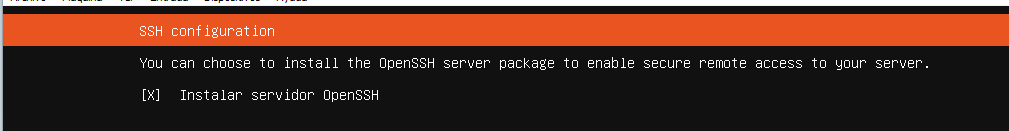
Introducimos nuestros credenciales



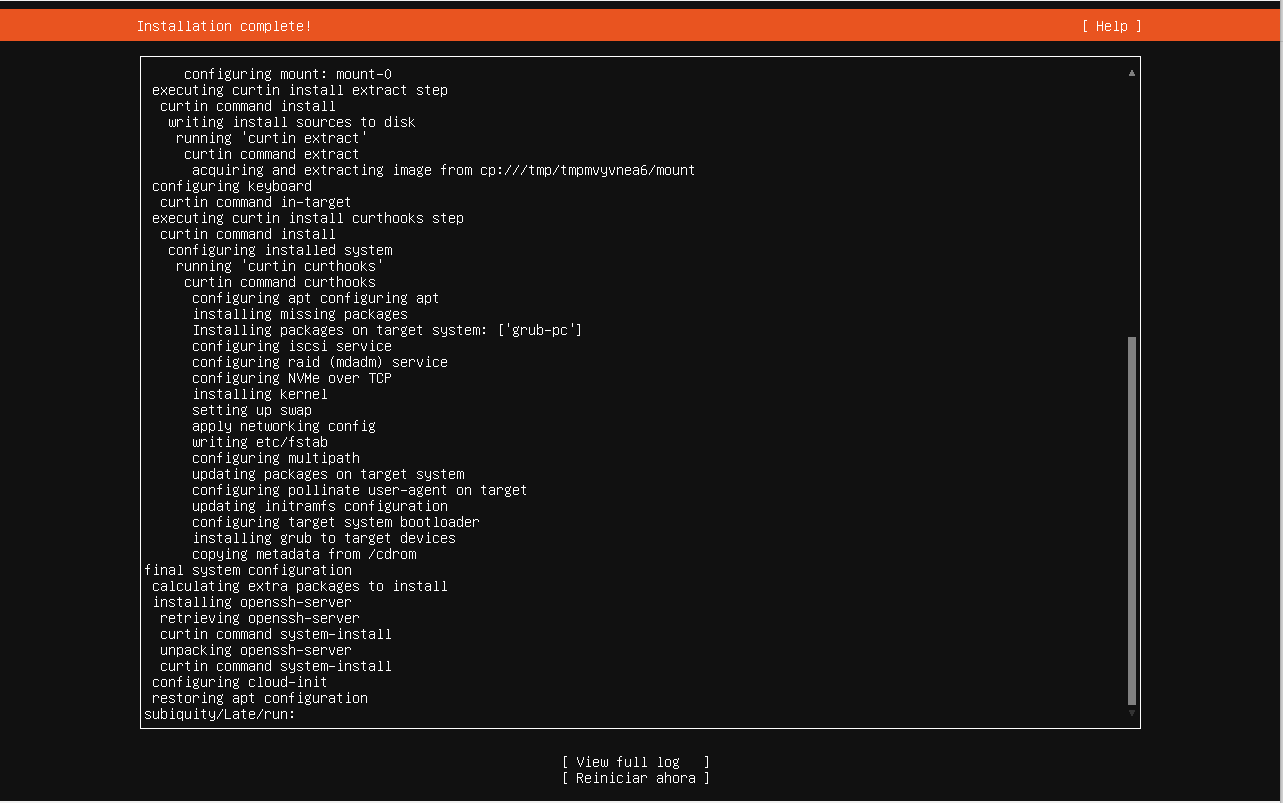
Nos saltamos la instalacion de ubuntu pro



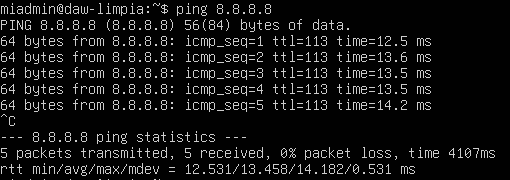
Y, muy importante, instalaremos ssh (la casilla viene sin marcar por defecto)



Dejamos que se instale la maquina y la reiniciaremos cuando termine



Ya reiniciada, comprobaremos que tenemos acceso a internet haciendo ping

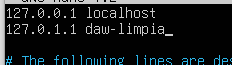


Si es así, haremos un “sudo apt update” y “sudo apt upgrade”, que descargan e instalan las ultimas actualizaciones

A continuación cambiaremos el nombre de la maquina



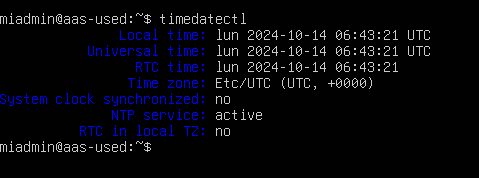
Cambiamos el nombre



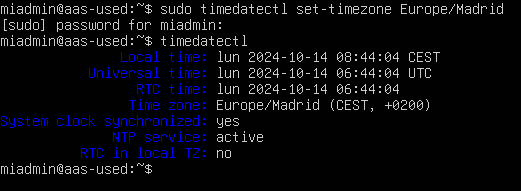
Introducimos el siguiente comando y reiniciamos la maquina



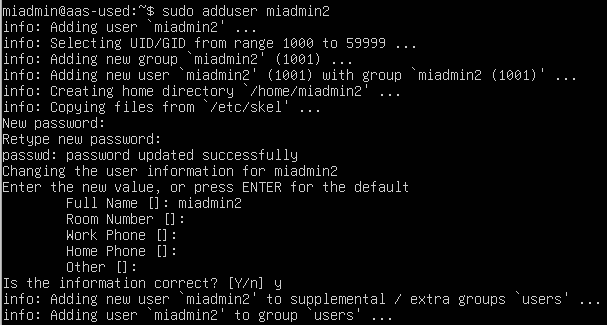
Para comprobar la zona horaria de la maquina introduciremos el comando “timedatectl”



Si queremos una zona horaria en especifico, introduciremos el siguiente comando: “sudo timedatectl set-timezone Continente/Ciudad”



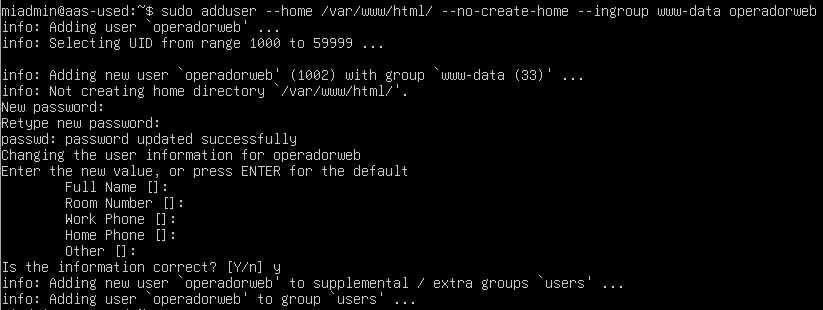
Crearemos 2 usuarios, el primero sera uno con permisos de administrador llamado miadmin2



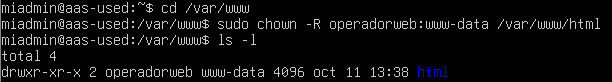
Le concedemos permisos



Ahora añadimos un usuario al directorio /var/www/html y al grupo www-data



Le concedemos permisos solo en el directorio html



con ls -l podemos ver que el usuario aparece

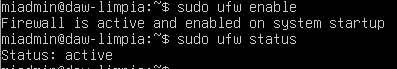


## Cuentas de administración

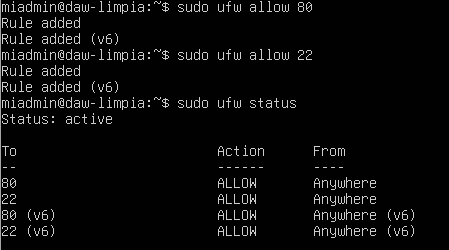
Maquina anfit, administrador y usuario

## Apache

Primero habilitamos el cortafuegos (comprobaremos su estatus con ufw enable y ufw status)



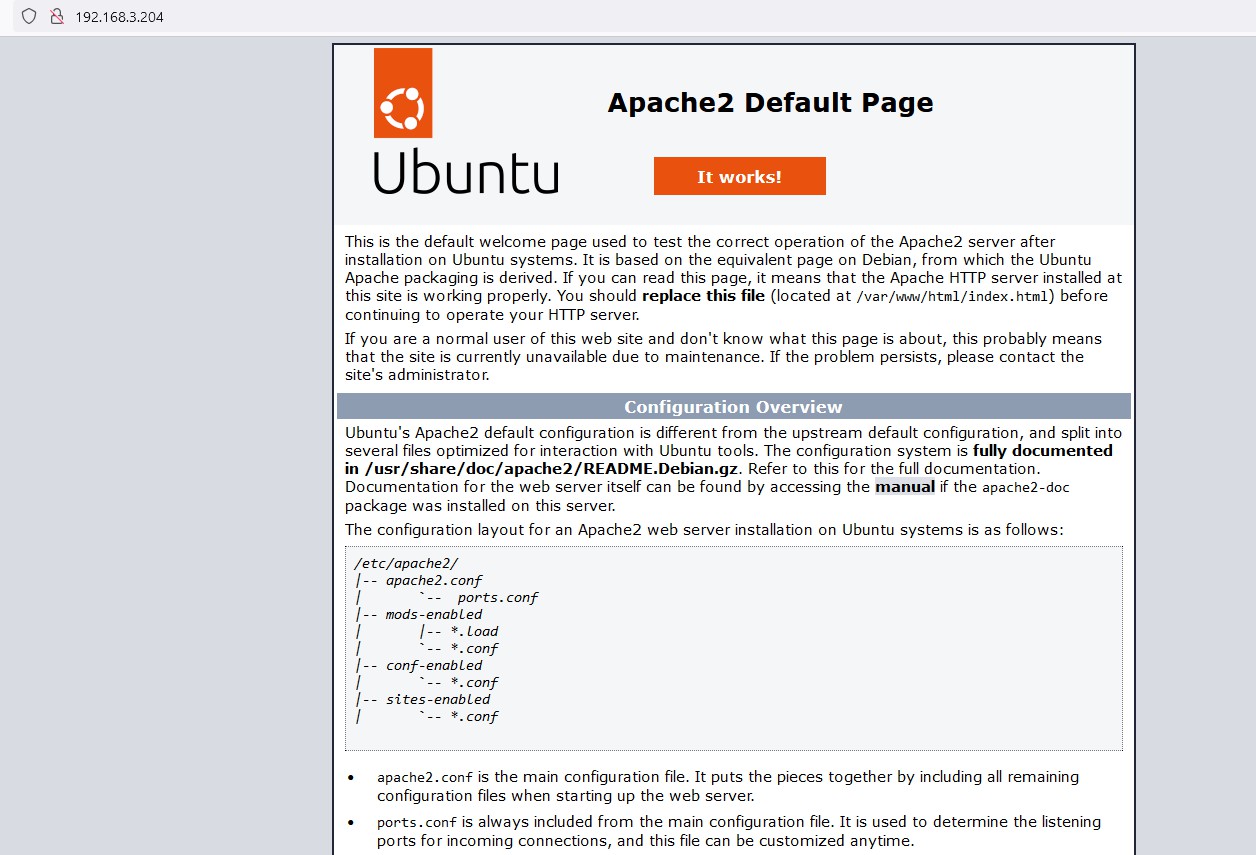
Abrimos los puertos 80 y 22



Ahora instalaremos el servidor apache, introduciremos el siguiente comando



Para comprobar que funciona, introduciremos la ip de la maquina en cualquier buscador, deberia aparecer esta pagina:



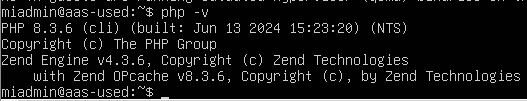
## PHP

Para instalar PHP en nuestra maquina ejecutaremos el siguiente comando



(El comando instala la ultima versión de php, si quisiéramos una en especifico podríamos indicarselo así: “sudo apt install php8.3”)

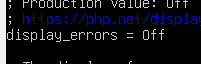
Tras esto, comprobaremos que se ha instalado correctamente y que version se ha instalado



Ahora editaremos el archivo php.ini, que se encuentra en el siguiente directorio



Cambiaremos el parametro “display\_errors” de “off” a “on”



Haremos lo mismo con el parametro “display\_startup\_errors”



Al acabar, guardaremos los cambios con ctrl+x, reiniciaremos el servicio apache y comprobaremos que esta en marcha



Instalaremos una serie de modulos para php:





(Si este comando nos dice que no puede encontrar el paquete para instalar mbstring, introduciremos el siguiente comando)

sudo apt-get install php-mbstring

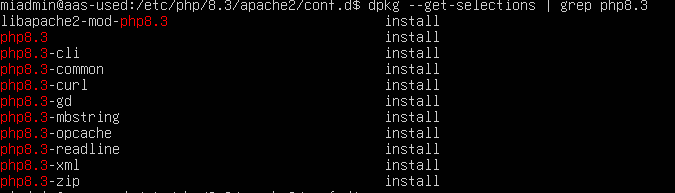








Podemos comprobar todos los modulos instalados de la siguiente manera



Para acabar, editaremos de nuevo el archivo php.ini que se encontraba en la ruta

/etc/php/8.3/apache2/php.ini

Aqui indicaremos aumentaremos el limite de memoria de 128M a 256M

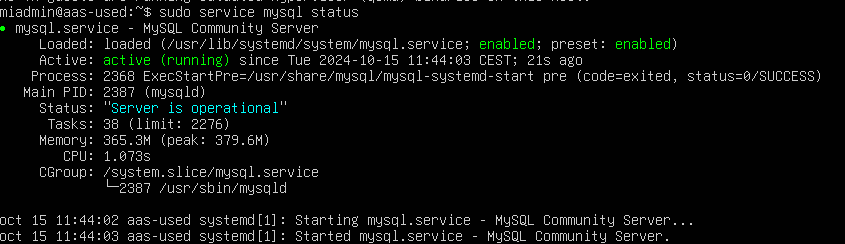


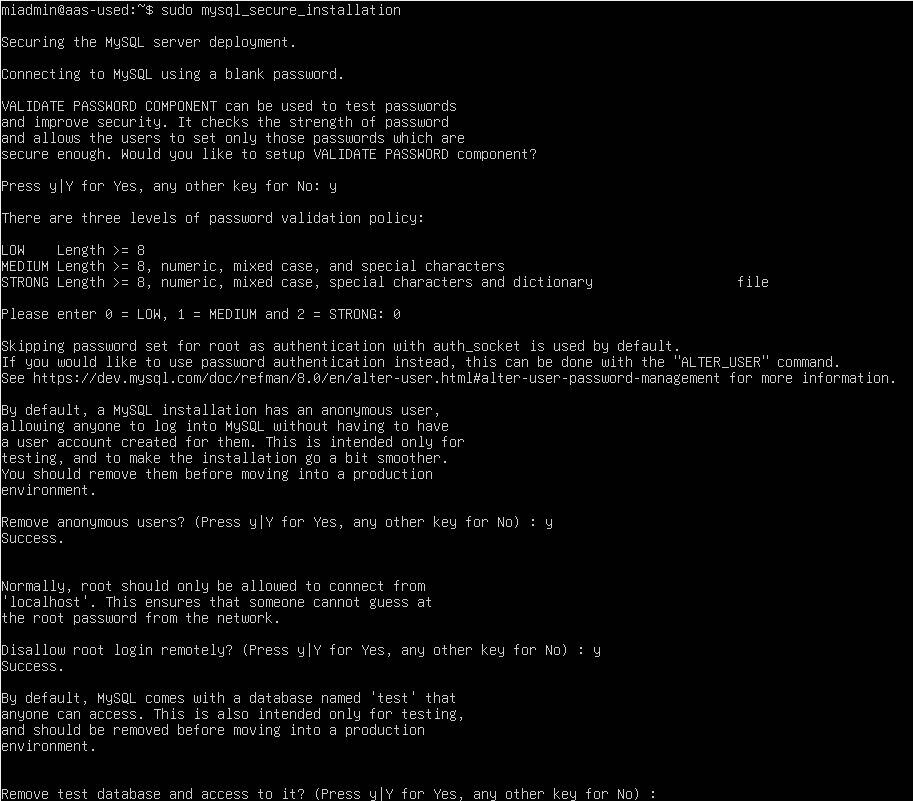
## MySQL

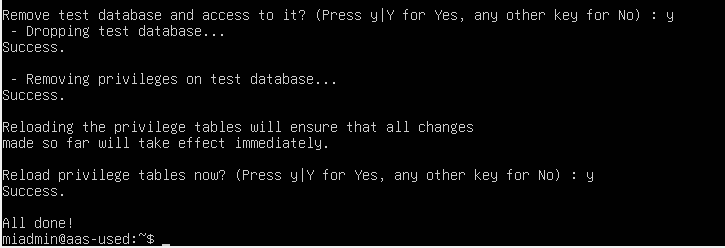
Instalamos mysql-server



Comprobamos que esta activo







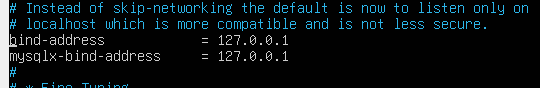
Posteriormente abrimos el puerto 3306 (el que utiliza MySql)

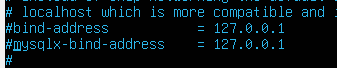


#### Copiaremos el archivo de configuración de mysql (para tener una copia de seguridad)



#### Editaremos este mismo archivo y comentaremos estas dos líneas



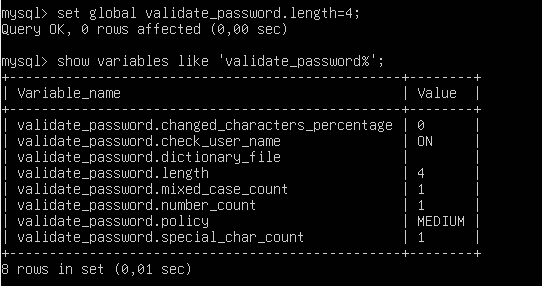


Reiniciamos el servicio de MySQL



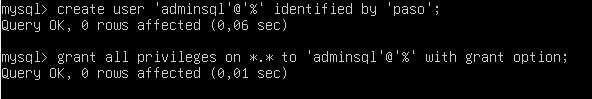
A continuación entramos en Mysql con el comando “sudo mysql”

Ejecutamos los siguiente comando





Creamos un usuario administrador y le concedemos todos los permisos en todas las bases de datos



Salimos de mysql



Intentamos entrar a mysql usando ese usuario para comprobar si se ha creado correctamente



Ya habriamos instalado MySQL

## XDebug

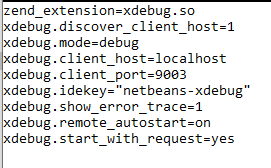
Instalamos xdebug



Editamos el archivo de configuración de xdebug

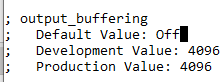


En este archivo escribiremos lo siguientes



Una vez hecho esto editaremos el archivo php.ini y en el apartado “output\_buffering” cambiaremos el valor de “Default value” a On





Reiniciamos Apache y ya tendríamos instalado xdebug

## Cuentas de desarrollo y hosting virtual

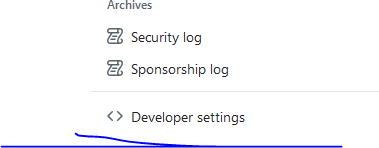
# GITHub-Internet

## Cuentas de desarrollador

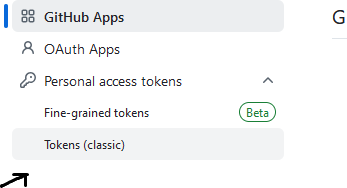
Necesitaremos una cuenta de github, nos registraremos (no tiene perdida) y obtendremos un token personal

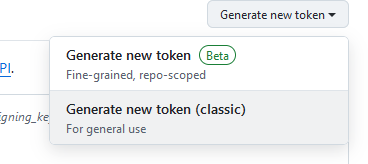
Dicho token nos permitirá conectarnos a un repositorio de GitHub a través de NetBeans

Esta opcion se encuentra abajo del todo en la pestaña de configuracion de nuestro perfil, en developer settings

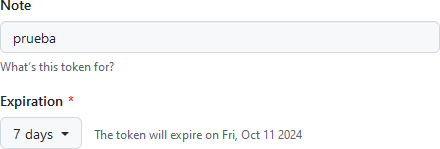


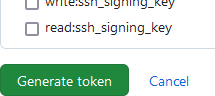
Despues iremos a la pestaña de claves de acceso, a la opcion clasic



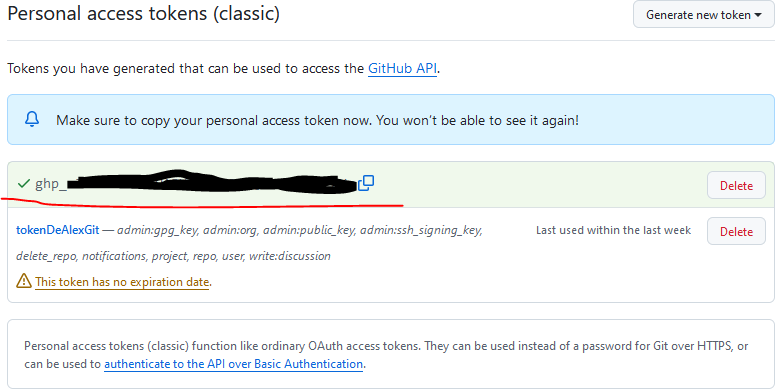


Indicaremos para que usaremos el token (puede ser cualquier cosa, con no dejar el cuadro en blanco sirve) y cuando expirara el token, el minimo son 7 dias y el maximo es que no expire





Y listo, ya tendremos nuestro token, debemos guardarlo bien pues esta sera la unica vez que github nos lo muestre



# WXED-Windows X

## Instalación y configuración inicial de la máquina

## Cuentas administradoras y cuenta de desarrollador

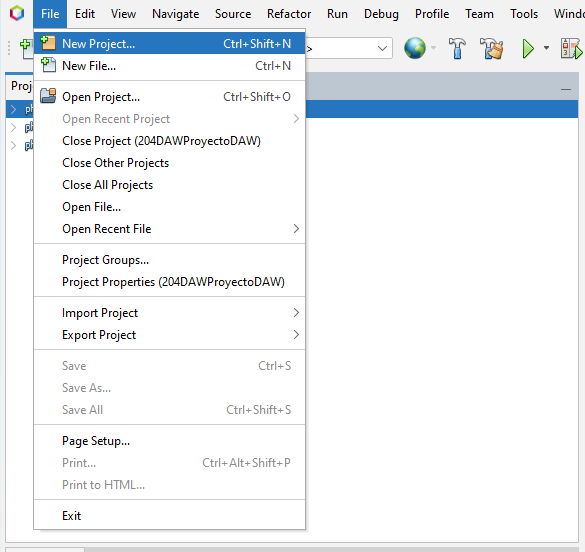
## Filezilla

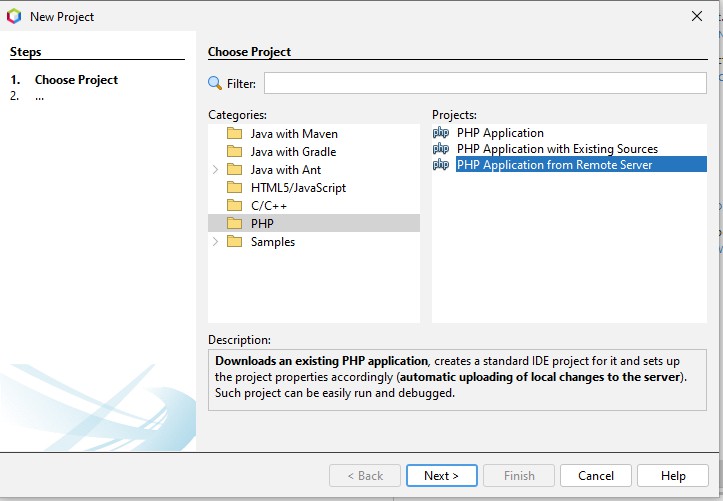
## NetBeans

***Instalación y configuración inicial (plugins)***

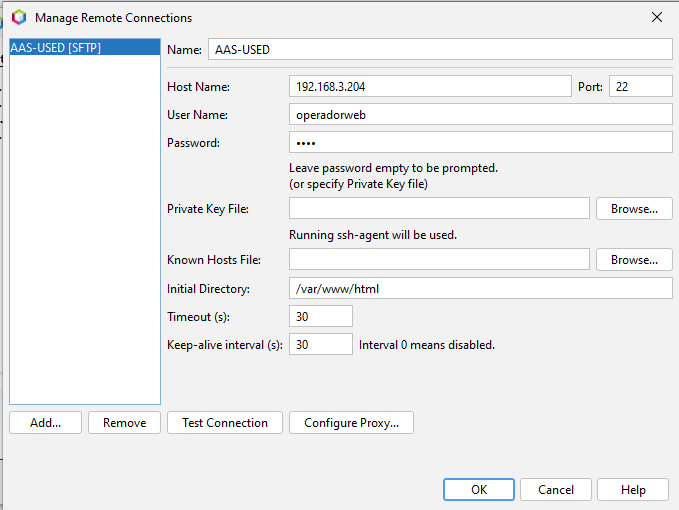
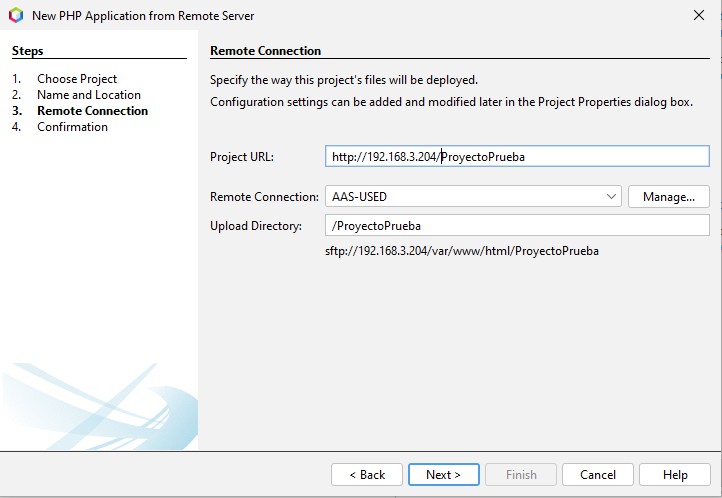
***Creación de proyectos, modificación, borrado, prueba.***

Lo primero sera crear un proyecto de php

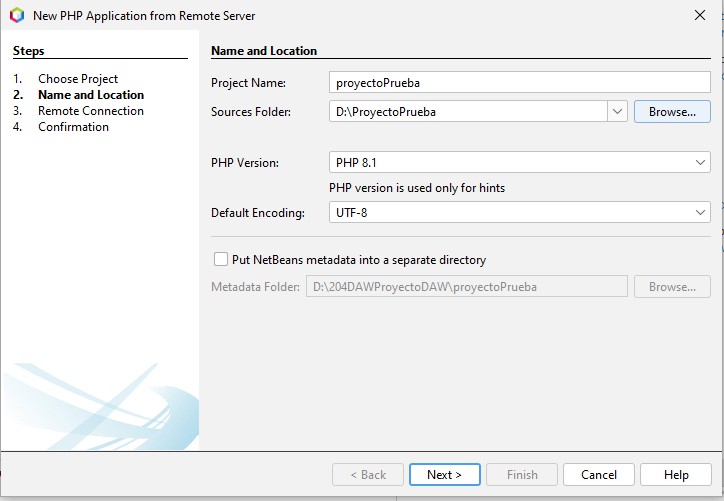


Para crear un proyecto desde 0 seleccionaremos la opción “aplicación desde servidor remoto”

30 / 58

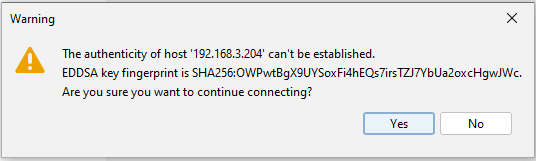


Elegimos el nombre y la ruta del proyecto

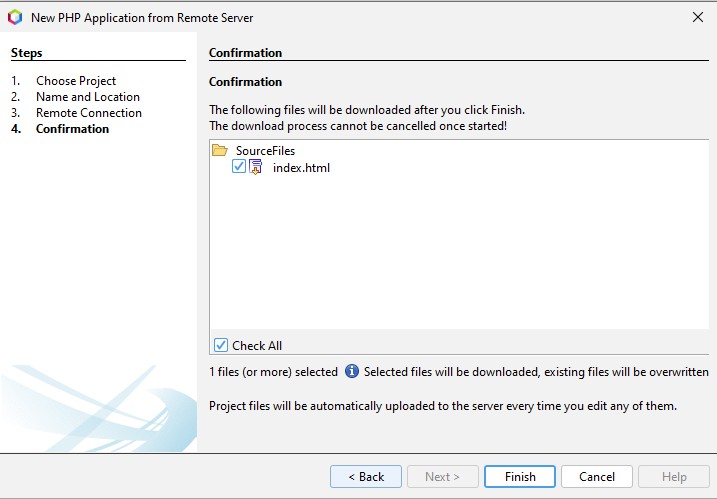


Indicamos la url del directorio del servidor en el que deseamos subir los archivos, también añadimos una conexión remota al servidor por sftp

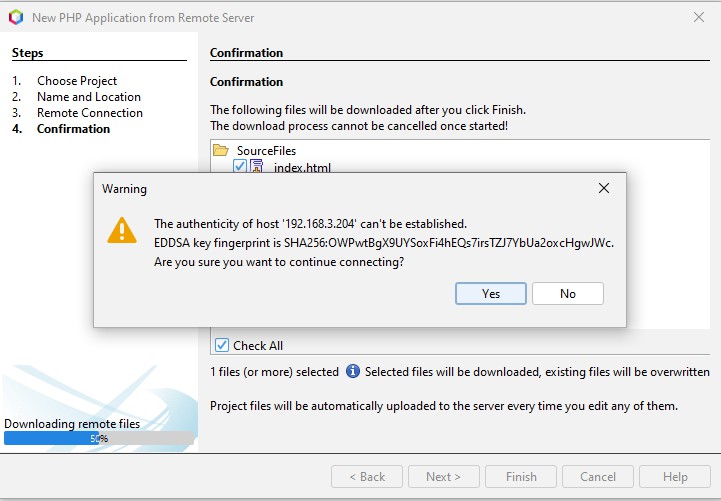
Tras esto seleccionamos que “si”



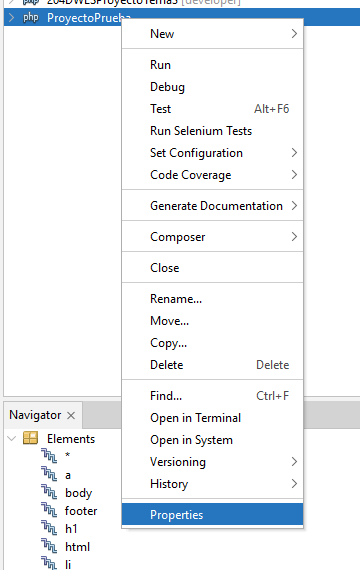
Seleccionamos los archivos que queremos descargar del servidor



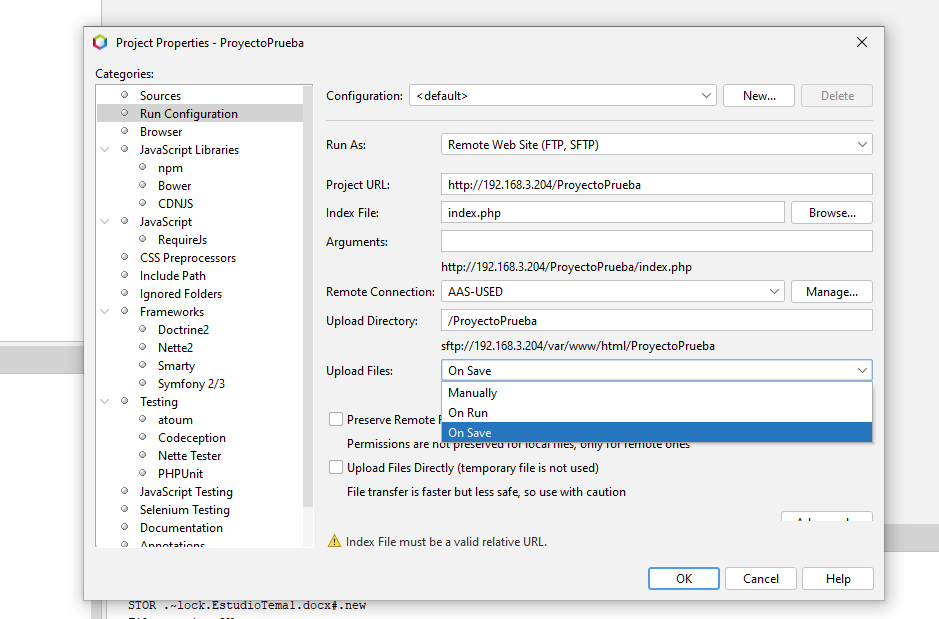
Seleccionamos que si de nuevo



Para configurar el que los archivos se suban al servidor cada vez que guardemos, iremos a la configuración del proyecto



En la pestaña “run configuration” seleccionaremos “upload on save”

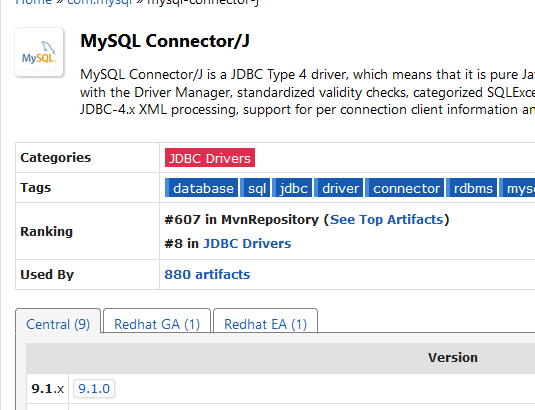


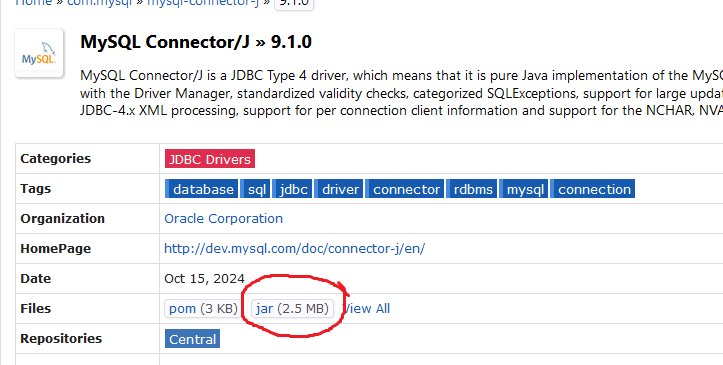
***Conexión al servidor remoto SFTP. (Almacenamiento local/almacenamiento remoto)***

***Administración de la base de datos***

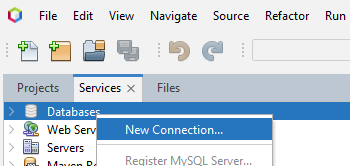
Lo primero que haremos sera descargar el driver conector de mysql desde la pagina de maven: <https://mvnrepository.com/artifact/com.mysql/mysql-connector-j>

en esta ocasión, descargaremos la versión 9.1

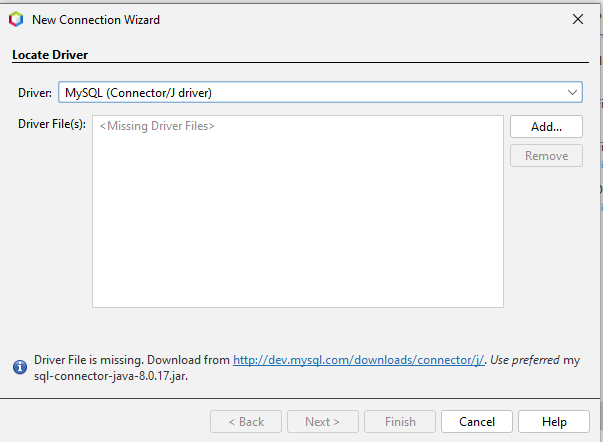




Ya descargado, iremos a NetBeans, en el apartado de la izquierda (en el que aparecen los proyectos) iremos a el apartado “services”, haremos click derecho en “databases” y en “new connection”

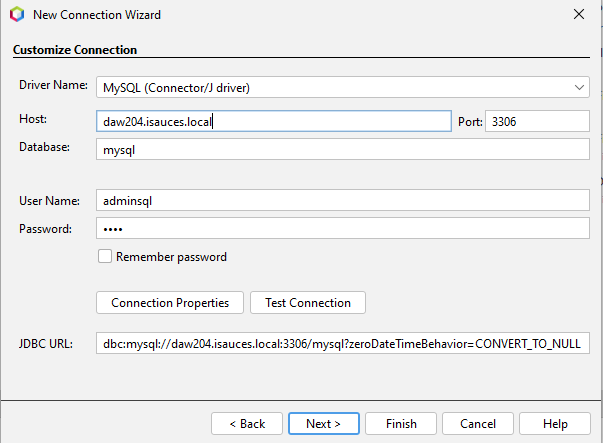


En la siguiente pestaña, seleccionaremos la opción “mysql connector/J driver” y selecionaremos “add”

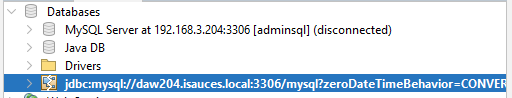


Seleccionaremos el driver que descargamos hace un momento y continuaremos. En esta nueva pestaña configuraremos la conexión

En host indicaremos la ip/alias de la maquina en la que este la base de datos, la base de datos puede ser la generica (mysql) o una ya creada, el puerto sera el predefinido para mysql (3306) y el usuario sera uno ya creado por el administrador de la base de datos

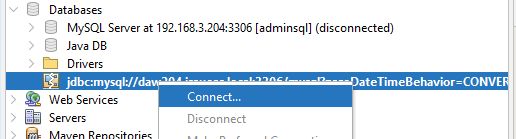


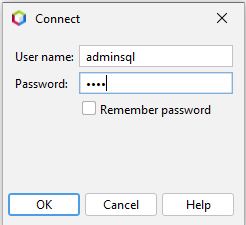
Si todo ha ido bien, en el desplegable “databases” debería aparecer nuestra conexión



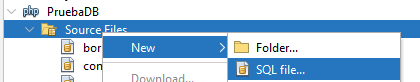
Si desplegamos esta conexión, podremos ver las bases de datos disponibles (solo podremos hacerlo si estamos conectados)

Como conectarse:

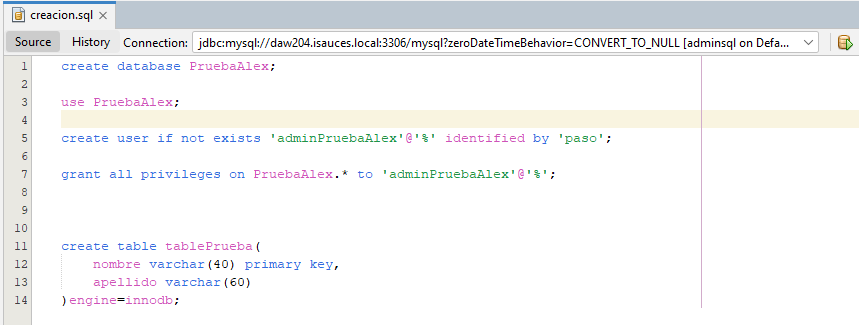




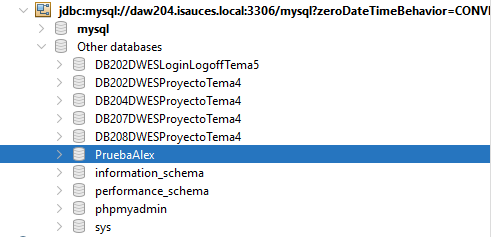
Para probar esta conexión, crearemos una base de datos y jugaremos con ella usando scripts, crearemos 4 archivos .sql



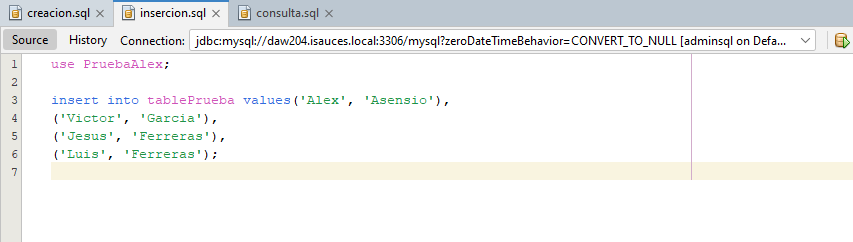
Escribimos el script de creación de una base de datos y un administrador para la misma, en la parte de arriba seleccionaremos la conexión que queramos usar

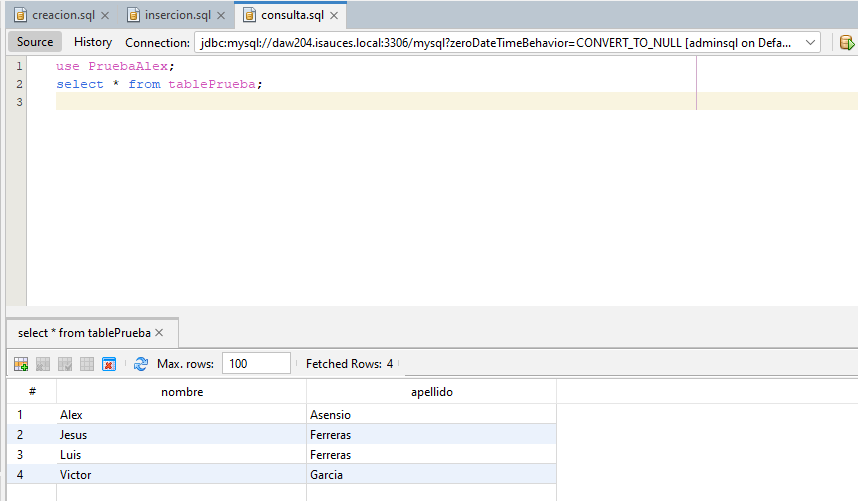


Haremos click en este boton (o pulsaremos control+hift+e) =>  Tras ejecutar, podemos ver que la base de datos se ha creado

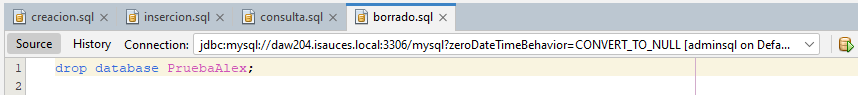


Ahora insertaremos datos y haremos una consulta



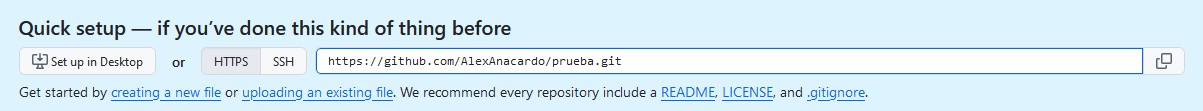


Para terminar, borramos la base de datos de prueba

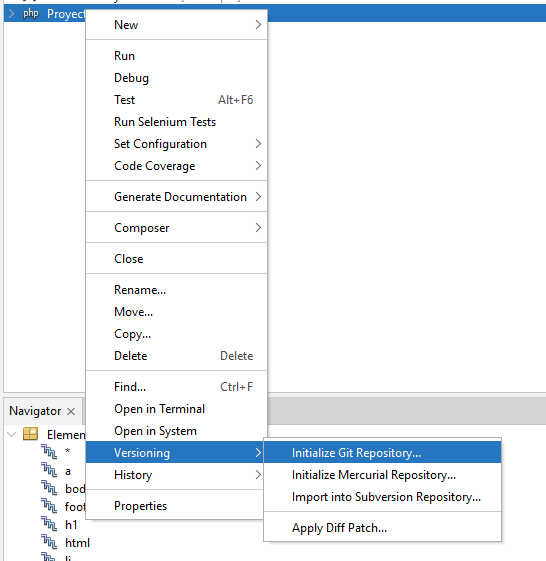


***Conexión al repositorio – versionado***

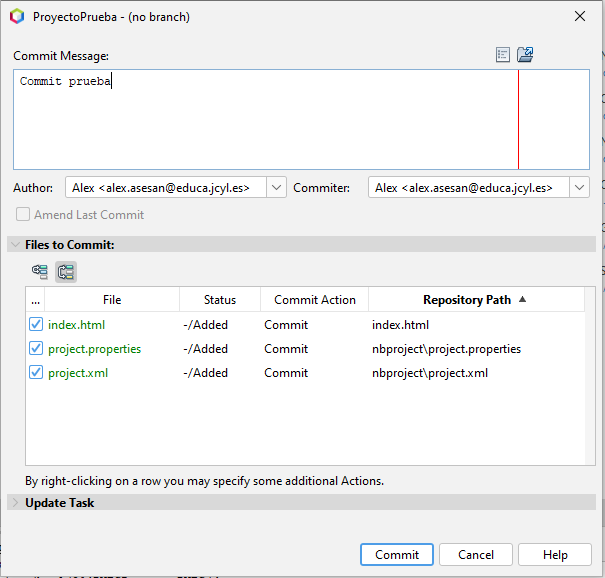
Creamos un repositorio (no obligatorio, puede ser uno ya creado) y copiamos su url



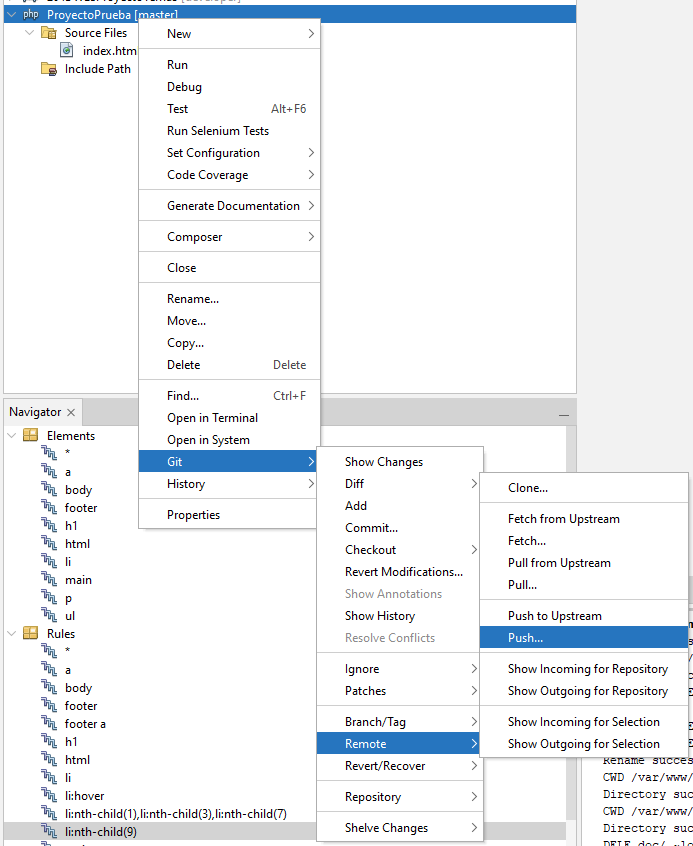
En el proyecto que creamos anteriormente, inicializamos un repositorio de git



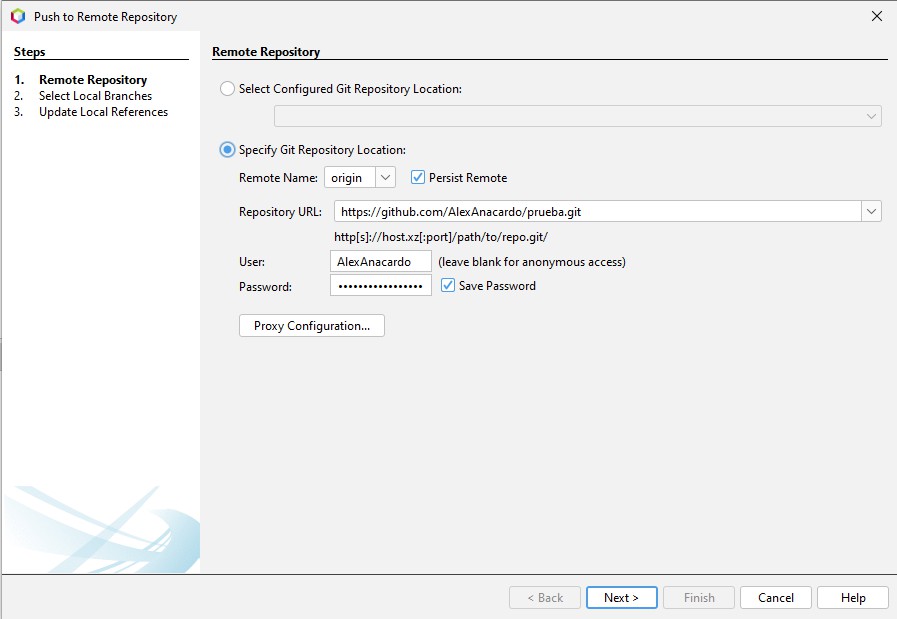
Haremos un commit de los archivos del proyecto



Hacemos un push (ojo, esto lo hago por que el repositorio remoto esta vacío, si el repositorio tiene archivos que no están en local o si ha sido actualizado por una fuente externa, es importante hacer un “pull” para descargar todo lo anterior mencionado en nuestro repositorio local)



Aquí introducimos la URL del repositorio, el nombre de nuestra cuenta, y en el apartado “password” introduciremos nuestro **token**



Y ya habríamos subido nuestros archivos a remoto

***Depuración - Configuración de la ejecución para la depuración***

***Creación de un proyecto nuevo***

***PHP Doc***

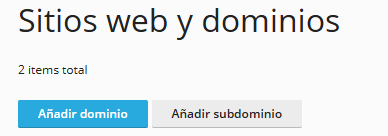
***CSS / JS / AJAX / XML / JSON***

# Paso a explotación-PLESK

## Utilizando GitHub

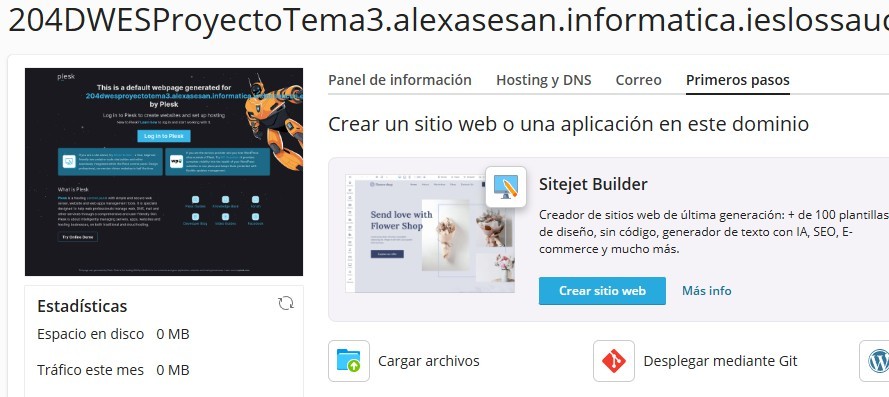
Primero deberemos crear un nuevo subdirectorio, en este caso su nombre será el nombre del proyecto + la extensión del dominio

La carpeta en la que se guardaran los archivos estará dentro de la carpeta httpdocs y su nombre será el mismo que el del proyecto

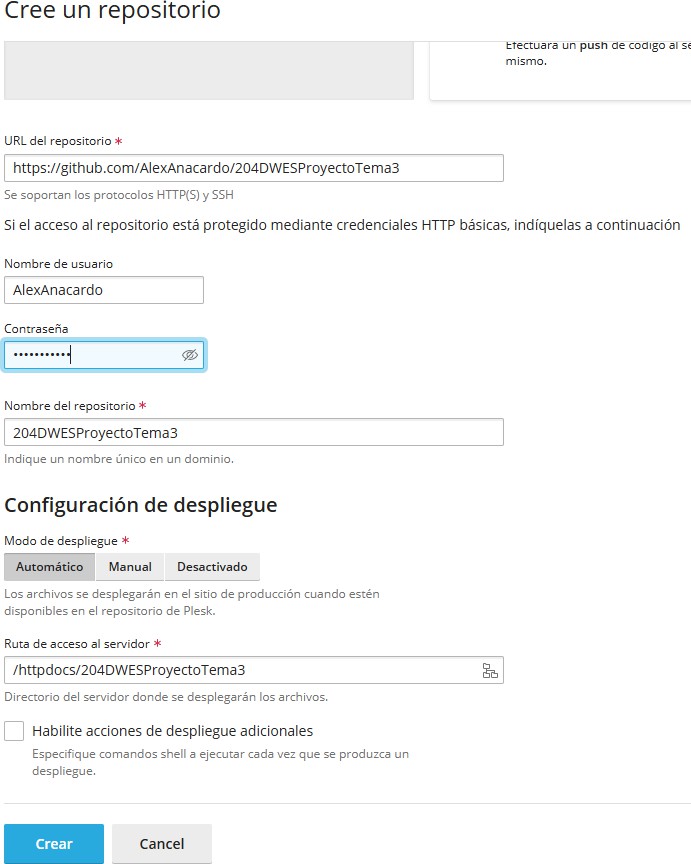




Tras esto seleccionaremos la opción ‘desplegar mediante git’



Seleccionaremos la opción ‘repositorio remoto’, introduciremos la url del repositorio, nuestro usuario y contraseña (que no token, ojo) de git. Dejaremos lo demás tal y como esta y le daremos a crear

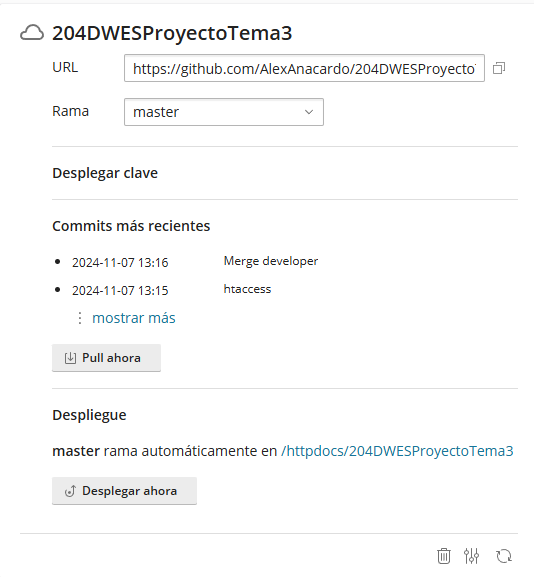


Importante, para poder entrar a esa subpágina a través del subdominio, deberemos incluir un archivo htaccess indicándole cual es el índice del proyecto

Ejemplo:



Nos aseguraremos de que la rama que se usara para subir los archivos es la rama master y seleccionamos “desplegar ahora”

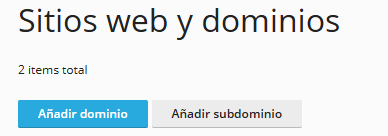


Ya hecho esto, nuestros archivos habrán sido subidos al entorno de explotación



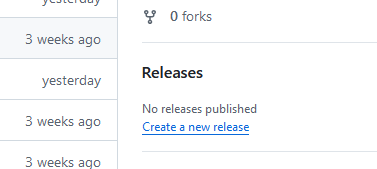
## Subiendo los archivos manualmente

Creamos un subdominio tal y como en el anterior apartado

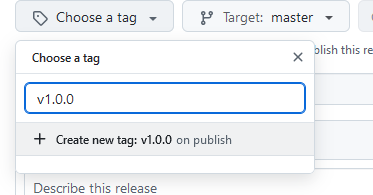




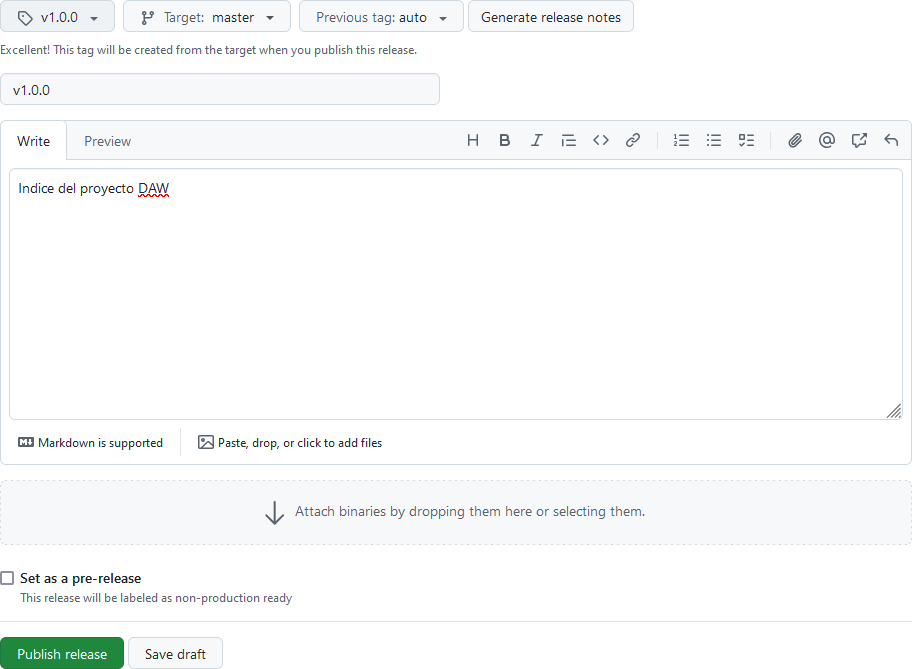
Vamos a nuestro repositorio de github, entraremos en la pestaña de “releases”



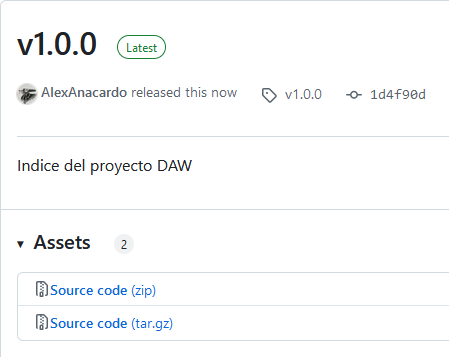
Crearemos una nueva etiqueta, como esta es la primera release sera la version 1.0.0



Elegiremos la rama master como la rama a usar y como titulo pondremos el nombre de la version, publicamos la release



Ahora tenemos la opcion de descargar nuestros archivos como un zip

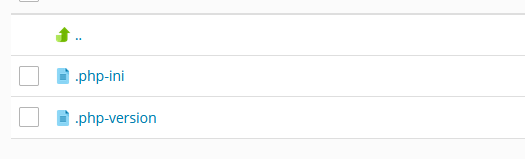


Descargaremos dicho zip y lo descomprimimos

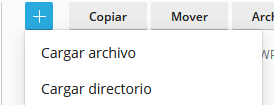
De vuelta a Plesk, entraremos en nuestro subdirectorio y en la pestaña “Primeros pasos” selecionaremos “cargar archivos”



Aqui borraremos el index generado por defecto y dejaremos los otros dos archivos ahi



Haremos click en el + azul y seleccionaremos “cargar archivo” (puede que para subir carpetas sea obligatorio usar la opcion “cargar directorio”)



Seleccionamos los archivos a subir

