# Actividad: Tutorial montar servidor

Contenido

[Actividad: Tutorial montar servidor 1](#_Toc115386473)

[0. Punto de partida. 1](#_Toc115386474)

[1. Crear la máquina. 2](#_Toc115386475)

[Configuración de Red nuestro servidor 3](#_Toc115386476)

[2. Instalar el sistema operativo 3](#_Toc115386477)

[Configuración del sistema operativo 4](#_Toc115386478)

[Instalar actualizaciones de Ubuntu 7](#_Toc115386479)

[3. Instalar servidor Apache2. 8](#_Toc115386480)

[4. Instalar php en el servidor Apache2 9](#_Toc115386481)

[5. Cambiar IP del servidor 9](#_Toc115386482)

[6. Conectar el servidor al VSCode a través de “SSH” 10](#_Toc115386483)

[Agregar y conectar servidor a VSCode 10](#_Toc115386484)

[Permisos Linux 11](#_Toc115386485)

[7. Autenticar con nuestra cuenta GitHub 12](#_Toc115386486)

[8. Sincronizar con Git 13](#_Toc115386487)

[Clonar el repositorio 13](#_Toc115386488)

[Cambios desde VSCode 14](#_Toc115386489)

[Cambios desde GitHub 14](#_Toc115386490)

## 0. Punto de partida.

Para montar la máquina virtual donde tendremos nuestro servidor debemos tener instalado el programa "Virtual Box" y la imagen con el sistema operativo que instalaremos (ubuntu-22.04.1-live-server-amd64.iso).

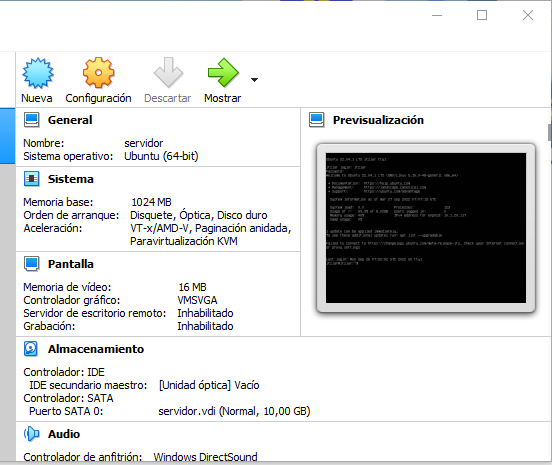
Además, debemos tener instalado el programa Visual Studio Code sobre el que se hará el desarrollo del software a desplegar en el servidor que montemos. Debemos tener también instaladas en Visual Studio Conde las siguientes extensiones:

* Remote – SSH
* GitHub Pull Requests and Issues
* PHP Debug
* PHP Intelephense

También debemos disponer una cuenta GitHub con la que sincronizaremos nuestro trabajo.

## 1. Crear la máquina.

Abrimos Virtual Box y pulsamos "Nueva".



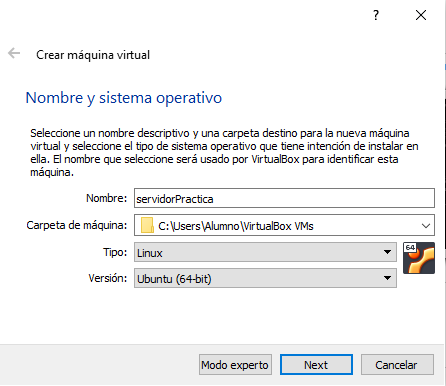
Automáticamente se abre una ventana donde debemos indicar el nombre del servidor y el sistema operativo que queremos instalar en él.

En el apartado "Nombre" le establecemos un nombre. Por ejemplo: "servidor1"

En el apartado "Carpeta de máquina" se rellena automáticamente. Indica la carpeta en la que se va a crear.

En el apartado "Tipo" le establecemos: "Linux".

En el apartado "Versión" establecemos: "Ubuntu (64-bit)"



Pulsamos “Next” cuando tenemos los datos establecidos.

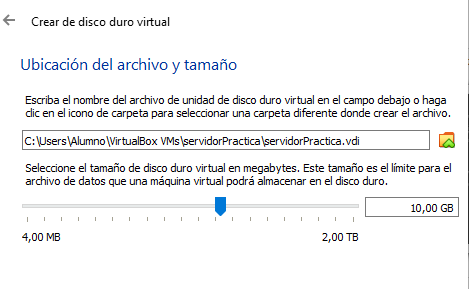
A continuación, indicamos la memoria que usará nuestra máquina virtual. Dejamos la configuración que nos propone y pulsamos "Next".

Después indicamos si deseamos un disco duro virtual, dejamos marcada la opción que aparece por defecto "Crear un disco duro virtual ahora" y pulsamos "Next".

En la siguiente ventana indicamos el tipo de archivo de disco duro, dejamos marcada la opción que aparece por defecto "VDI (VirtualBox Disck Imagen)" y pulsamos "Next".

A continuación, indicamos el tamaño del disco que vamos a crear, dejaremos marcada la opción por defecto que nos viene seleccionada "Reservado dinámicamente" y pulsamos "Next".

Después establecemos la carpeta donde se crea la unidad de disco duro virtual y su tamaño. Dejamos las opciones que aparecen por defecto y pulsamos "Next".



Aquí ya tendremos nuestra máquina virtual creada.

### Configuración de Red nuestro servidor

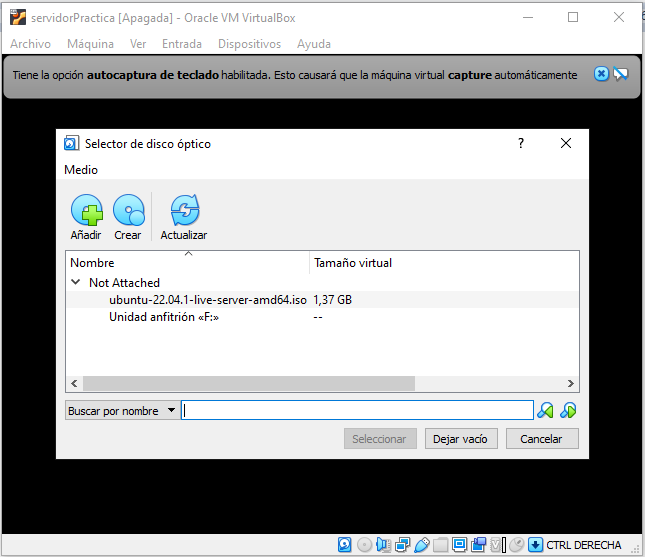
Con el servidor seleccionado desde virtual box, cambiamos la configuración de red indicando en el apartado “Conectado a:” establecemos “Adaptador puente” y le damos a “Aceptar”.

## 2. Instalar el sistema operativo

Para instalar el sistema operativo iniciamos la máquina pulsando sobre el botón “Iniciar”. Al iniciar nos pedirá la imagen con nuestro sistema opertivo que habremos descargado previamente (imagen iso con el sistema operativo Ubuntu).



Seleccionamos la ubicación de nuestra imagen. Si no aparece por defecto, debemos indicar la ubicación en la que está pulsando el botón “Añadir”. Tras ubicarla, la seleccionamos y pulsamos "Seleccionar".



A partir de aquí, se abrirá una consola con la que trabajaremos con el teclado, por lo que, si decimos pulsar en algo, quiere decir que los seleccionamos con el teclado y pulsamos "Hecho" o "Enter".

En la siguiente pantalla se nos muestra si queremos instalar el sistema o cancelar. Seleccionamos la opción: "Try or Install Ubuntu Server" y pulsamos “Enter”. Y comenzará la instalación de Ubuntu en nuestra máquina virtual.

### Configuración del sistema operativo

Trans instalar Ubuntu, nos pedirá configurar algunas opciones del sistema operativo. En la siguiente pantalla nos pedirá que indiquemos el idioma, en nuestro caso seleccionamos "español" o el que deseemos.

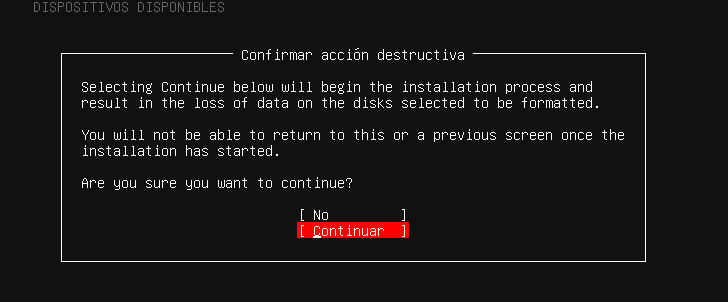
Después nos pide la disposición del teclado. En nuestro caso seleccionamos "Spanish" y se nos configuran el resto de opciones automáticamente. A continuamos pulsando "Hecho".

En las siguientes pantallas que nos muestre, seguimos pulsando "Hecho" para dejar los valores por defecto

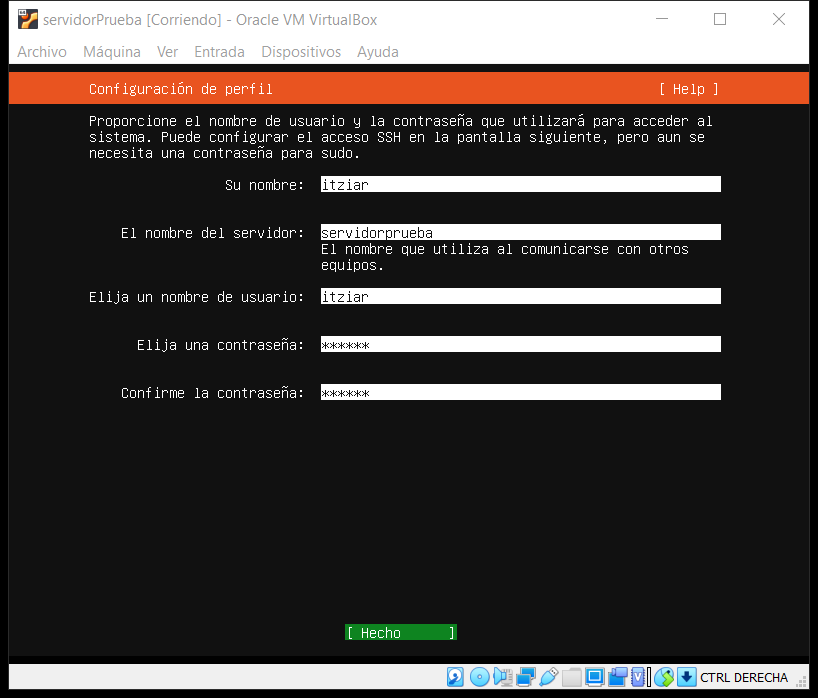
* “Choose type of install”
* “Conexiones de red”
* “Configure proxy”
* “Configure Ubuntu archive mirror”
* “Guide storage configuration”
* “Storage configuration”

Cuando hayamos avanzado estas pantallas, llegará a una ventana de confirmación puesto que va a instalar ya las configuraciones y puede dar lugar a la perdida de información que tuviéramos en el disco sobre el que estamos montando. Pulsamos “Continuar”.

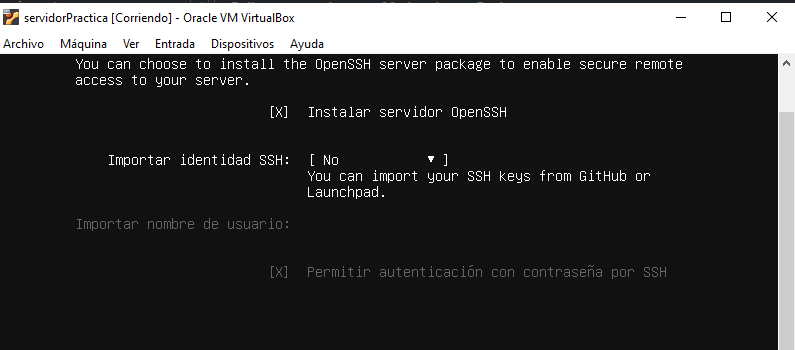
Siguiente muestra información establecida y si queremos cambiar. Pulsamos "Hecho" y nos pregunta si queremos hacer la instalación sobre el disco. Pulsamos "Continuar"



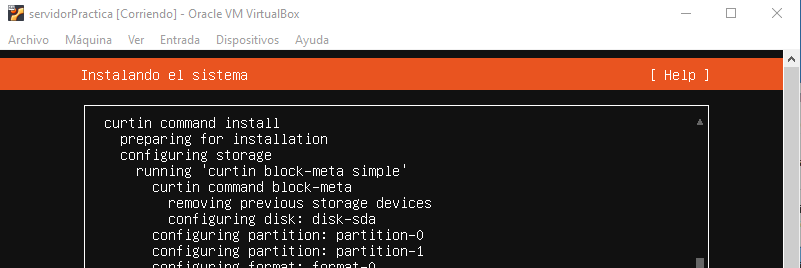
En la siguiente pantalla, haremos la “configuración del perfil” o usuario, para que podamos entrar al sistema y ejecutar como sudo o administrador. Cuando hayamos rellenado estos campos, seleccionamos “Hecho” y pulsamos “Enter” para continuar.



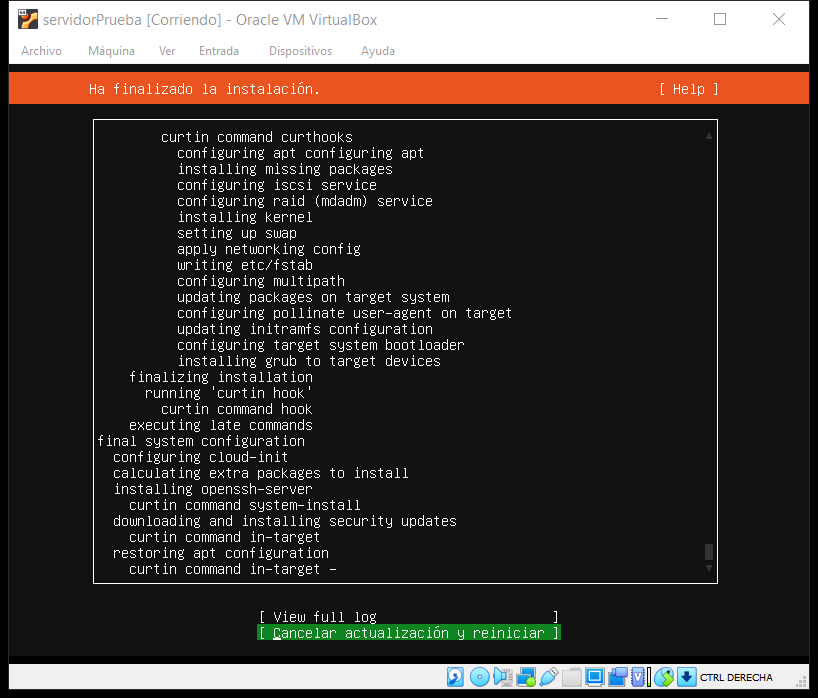
En la siguiente pantalla marcaremos la opción de "Instalar servidor OpenSSH" (pulsamos “Enter” cuando estemos sobre esa opción). Sirve para realizar conexiones seguras con el servidor. Dejaremos el resto de opciones como aparece por defecto. En un futuro lo haremos con GitHub. Pulsamos "Hecho".



A continuación, nos pide si queremos añadir algunas herramientas, que no será nuestro caso, por lo que pulsamos "Hecho" y empezará a hacer la instalación automáticamente.



Cuando haya finalizado la instalación (nos lo indica con en el mensaje naranja que aparece en la parte superior de la consola “Ha finalizado la instalación”), pulsamos sobre "Cancelar actualización y reiniciar".



Cuando reinicie, nos muestra una pantalla con los logs. Pulsamos “Enter” para continuar.

En este momento ya tenemos Ubuntu instalado, configurado y reiniciado, por lo que nos pedirá las claves de acceso que establecimos anteriormente. Tras establecer estos datos, ya podemos trabajar con el sistema operativo.

### Instalar actualizaciones de Ubuntu

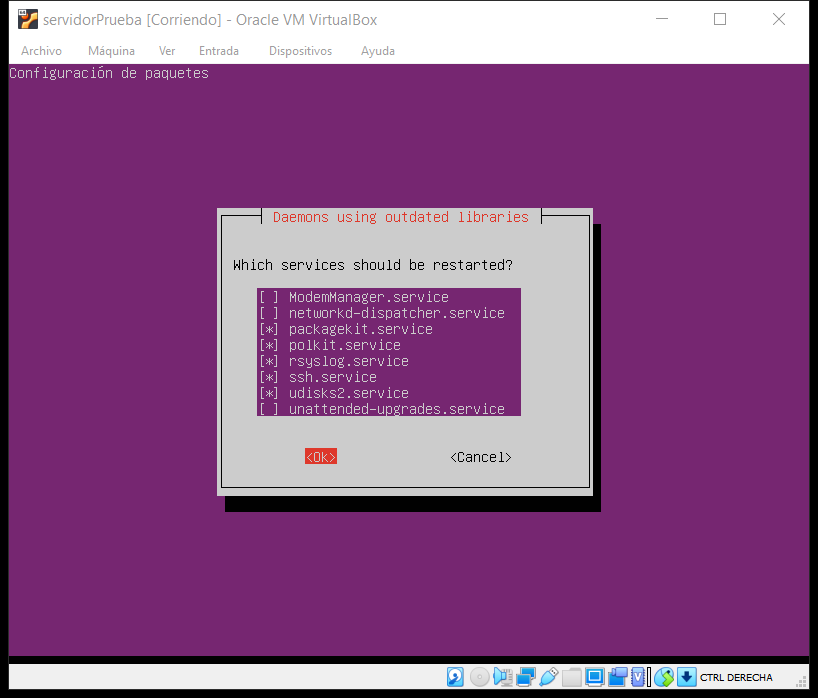
Para saber si tenemos actualizaciones para Ubuntu, cuando tengamos iniciado el sistema operativo debemos ejecutar el comando update y lo haremos como administrador con sudo, por lo que introducimos el siguiente comando:

“sudo apt update”

Tras ejecutarlo nos pedirá la contraseña de administrador. La introducimos y nos mostrará si existen actualizaciones. Si las queremos instalar, introducimos el siguiente comando:

“sudo apt upgrade”

Cuando nos pida continuar, escribimos “s” y damos a “Enter”. Por último, nos pide reiniciar los servicios instalados. Dejamos marcados los que aparecen por defecto y pulsamos “Ok”.

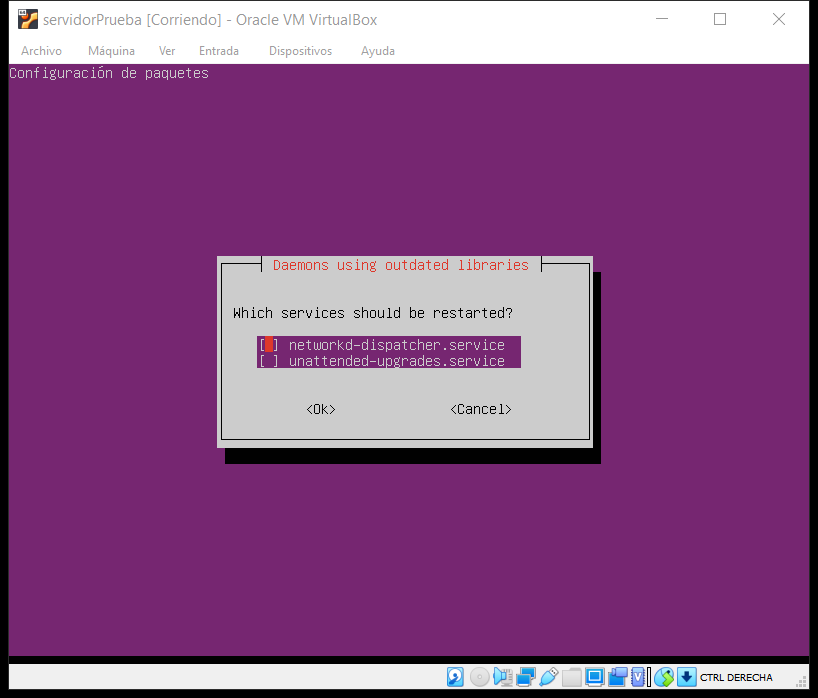


## 3. Instalar servidor Apache2.

Para instalar Apache2 debemos ejecutar el siguiente comando:

“sudo apt install apache2”

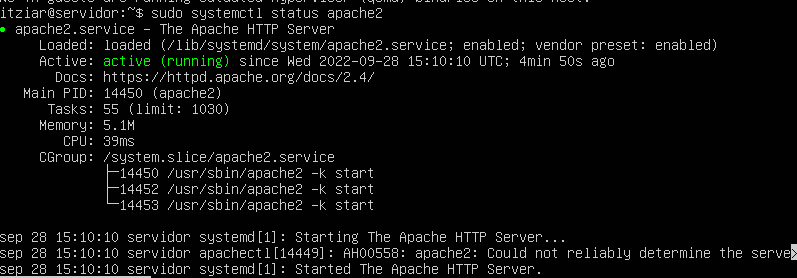
Nos pedirá continuar, por lo que pondremos: “s” y pulsamos “Enter”. En el siguiente paso nos indica los servicios que deseamos reiniciar. Lo dejamos como está por defecto y pulsamos “Ok”

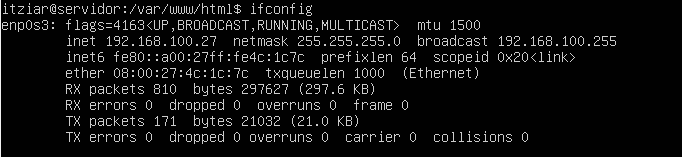


Para comprobar si está instalado ejecutamos el siguiente comando:

“sudo systemctl status apache2”

Si está instalado, nos los mostrará en pantalla como indica la siguiente imagen (si no es así, indicara “stopped”).



O también, podemos hacer una consulta a la página index.html que se crea en la ruta /var/www/html a través de nuestro navegador escribiendo la IP del servidor. Para saber la IP ejecutamos el comando “ifconfig” y nos lo muestra en el apartado inet. En este caso, la dirección que escribimos en el navegador es 192.168.100.27

También podemos hacer algunos cambios en el archivo index.html (como introducir algo de texto) y ver si se muestran los cambios desde el navegador (“sudo nano index.html”).

## 4. Instalar php en el servidor Apache2

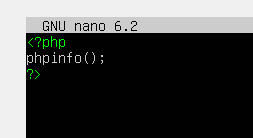
Para instalar PHP debemos ejecutar el siguiente comando:

“sudo apt install php”

Nos pedirá continuar, por lo que pondremos: “s” y pulsamos “Enter”. En el siguiente paso reiniciaremos el servidor con el comando:

“sudo systemctl restart apache2”

Para comprobar si está instalado crearemos un pequeño archivo .php, que ejecutaremos desde el servidor. Para ello creamos en la ruta en la ruta /var/www/html el archivo index.php con el comando “sudo nano index.php” y le añadimos el siguiente código:



Para ver que funciona, escribimos en el navegador la IP/index.php, en este caso: “192.168.100.27/index.php” y nos mostrará la configuración como le indicamos en el archivo que creamos antes.

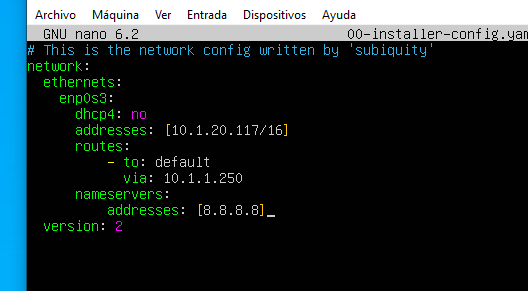
## 5. Cambiar IP del servidor

Por comodidad a la hora de trabajar, debemos establecer la IP del servidor a una IP fija. Si no, es probable que cuando volvamos a trabajar en él esta haya cambiado. Para ello debemos cambiar el archivo donde tenemos nuestras configuraciones “00-installer-config.yam” que se encuentra en la ruta /etc/netplan.

A la hora de editar este archivo debemos evitar el uso de tabulaciones y utilizar espacios en su lugar. También, hay que tener cuidado con el número de espacios, para que la configuración sea la correcta. En este caso, pondremos como IP fija la IP que tenemos ahora, pero sumando 100 al último cuarteto de la dirección. En este caso, la IP actual es 192.168.100.27, y en el archivo “.yam” estableceremos 192.168.100.127. Además, debemos establecer los siguientes valores:

* dhcp4: no
* addresses: [192.168.100.127/16]
* routes:
  + - to: default
  + Via: 192.168.100.1 (nuestra puerta de enlace predeterminada)
* Nameservers:
  + Addresses: [8.8.8.8] (dns de Google)

Para editar el archivo, nos situamos con el comando “cd” en la ruta “/etc/netplan” y lo editaremos con el comando “sudo nano 00-installer-config.yam”. Nos debe quedar como en la siguiente imagen:



Cuando hayamos editado, ejecutamos “sudo netplan apply”. Si no indica nada la línea de comandos tras ejecutar ese comando, está correcto. Si aparecieran errores, ejecutar “reboot”.

## 6. Conectar el servidor al VSCode a través de “SSH”

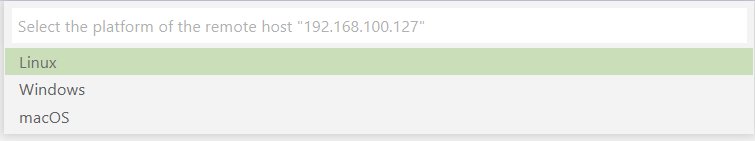
### Agregar y conectar servidor a VSCode

Previamente debemos instalar en VSCode la extensión SSH. Cuando esté veremos en la izquierda inferior del VSCode el siguiente icono:



Picar sobre este icono y selecciono del desplegable que sale en la parte superior de VSCode la primera opción “Connect to Host”. Y después seleccionamos la opción “+ Agregar nuevo host SSH”. A continuación, nos pide que escribamos el comando de conexión SSH, y pondremos el siguiente comando: “ssh usuario@IP -p22” (establecemos el nuestro usuario y la IP de servidor). Ejemplo: “ssh [itziar@192.168.100.127](mailto:itziar@192.168.100.127) -22. Y pulsamos “Enter”.

En la esquina inferior derecha nos mostrará que se ha agregado el servidor a VSCode que daremos a “aceptar”. Tras aceptar, nos pide establecer la configuración del servidor que hemos instalado. Indicamos: “Linux”

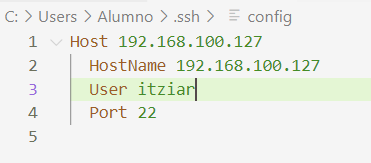


En la siguiente nos informa de la huella , seleccionamos “Continuar”.

Nos pedirá la contraseña para el usuario, se la pondremos y pulsamos “Continuar”. Sabremos que estamos conectados, cuando veamos en la esquina inferior izquierda la IP de nuestro servidor.



Establecemos la configuración, pulsamos sobre el icono de la imagen anterior y seleccionamos “open configuration ssh file” y le pondremos los datos de la siguiente imagen.



Si estamos conectados, podremos abrir la carpeta /var/www/html. Para ello vamos a File/open folder y en la parte superior de la pantalla le indicamos la anterior ruta.

### Permisos Linux

Para poder trabajar con el servidor ( modificar ficheros desde VSCode, crear carpetas, clonar el contenido de git, etc) debemos dar permisos desde Linux a nuestro usuario.

Desde la consola de Linux nos situamos en /var/www/ y establecemos los permisos a la carpeta html con el siguiente comando, en nuestro caso el usuario que usamos es itziar: “sudo chown -R itziar:itziar html”

Podemos comprobar los permisos creando en VSCode un fichero.html, y ver desde el servidor si se ha creado (desde /var/www/html ejecutamos ls y debe aparecer el fichero.html).

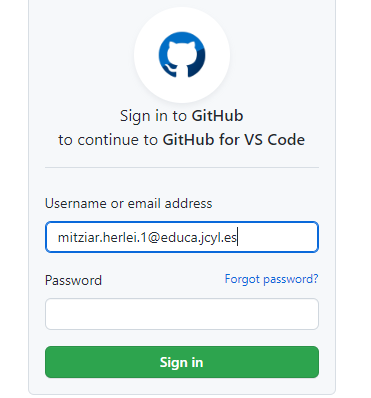
## 7. Autenticar con nuestra cuenta GitHub

Previamente debemos tener una cuenta GitHub. Esto nos sirve para poder trabajar con nuestros repositorios y tenerlos actualizados (sincronizar el trabajo que vamos haciendo).

Pulsamos sobre el icono de la activity bar que representa un usuario y seleccionamos “Turn on settings Sync”



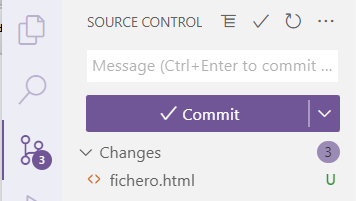
Después pulsamos “Sing in & Turn on” y luego seleccionamos “Sing in with GitHub”. Esto nos lleva a la página de autenticación de GitHub, donde pondremos los datos de nuestra cuenta y pulsamos “Sing in”.



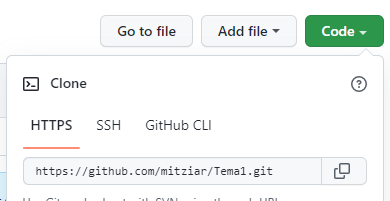
Cuando nos hayamos autenticado, veremos en el icono de usuario de la activity bar nuestra cuenta.

## 8. Sincronizar con Git

Pulsamos sobre el icono del control de versiones de la activity bar de VSCode, y seleccionamos el menú superior (tres puntitos)



En ese menú, seleccionamos “Remote” y después la opción “Add remote”, en el input que nos muestra, debemos introducir la url de nuestro repositorio que encontraremos en GitHub: “<https://github.com/mitziar/Tema1.git>” y pulsamos “Enter”. Nos pedirá que le demos un nombre y pulsamos “Enter”.



Después, nos pide una confirmación en la esquina inferior derecha del VSCode, donde le daremos a “YES”.

### Clonar el repositorio

Para disponer en VSCode del repositorio que necesitemos debemos clonarlo. Para ello debemos estar en apartado de control de código de la activity bar. A continuación, pulsamos sobre sobre los tres puntitos del explorador y seleccionamos Clone. Nos pedirá que indiquemos el repositorio a clonar, donde introducimos la dirección que nos proporciona GitHub (como en la imagen anterior). Después, establecemos el fichero a clonar y pulsamos “Ok”. En la esquina inferior derecha nos saldrá un mesaje de confirmación, donde pulsaremos “Open”. No pedirá la contraseña, la establecemos y pulsamos “Enter”.

Tras enlazar nuestro repositorio, desde el Terminal de VSCode configuraremos nuestro usuario y nuestro email de github ejecutando los siguientes comandos:

git config –global user.name “itziar”

git config –goblar user.email [mitziar.herlei.1@educa.jcyl.es](mailto:mitziar.herlei.1@educa.jcyl.es)

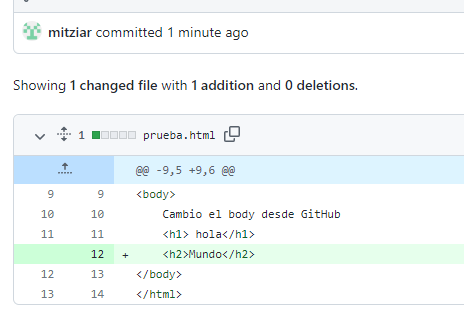
Podemos comprobar que todo ha ido bien, haciendo cambios en los archivos de nuestro repositorio desde VSCode y viceversa, y ver si se reflejan.

### Cambios desde VSCode

Por ejemplo, podemos introducir algo nuevo en algún fichero que exista desde VSCode, cuando demos a guarda, el icono de control de versiones nos indicará el cambio. Si queremos que se guarde nuestro cambio debemos hacer commit y tendremos la opción de añadir un mensaje que nos oriente en un futuro del cambio realizado. También nos permite dehacer cambios, guardar los cambios sin hacer commit (Stage change) pulsando sobre el icono +, o hacer commit y subirlo a la rama (commit & push).

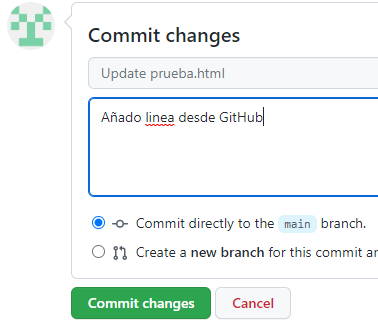


En nuestro caso, haremos commit & push, desde GitHub lo veremos reflejado.

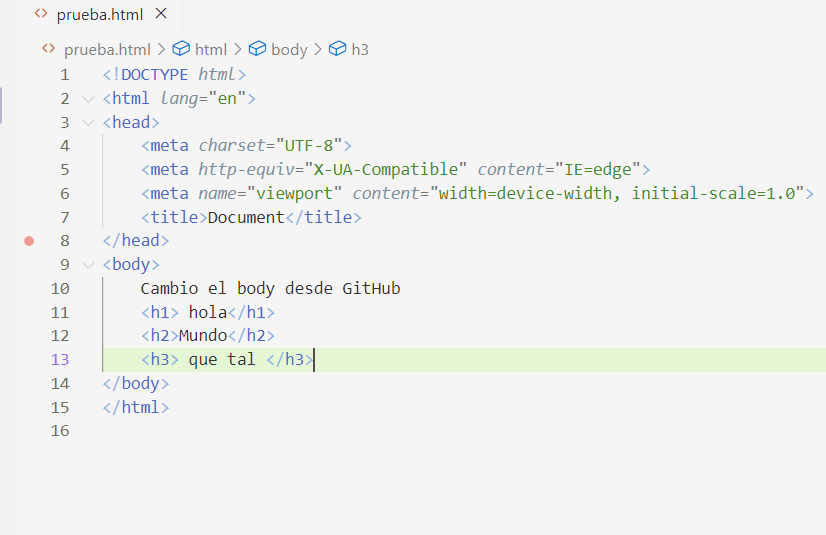


### Cambios desde GitHub

Editamos el archivo prueba desde GitHub y pulsamos “Commit changes”, que como tenemos seleccionado se hará directamente a la rama.



Para traer estos cambios a VSCode debemos hacer “Pull”, y se nos actualiza automáticamente.



Es buena práctica añadir mensajes que nos orienten sobre nuestros cambios para futuras ocasiones, tanto si cambiamos algo desde GitHub, como si lo hacemos desde VSCode. Cabe la posibilidad que exista conflicto porque hemos hecho cambios en ambos lados sin haber ejecutado los commits de forma correcta, en ese caso emplearemos el comando “pull rebase”.