



3 DE OCTUBRE DE 2022


# MANUAL INSTALACIÓN

PASOS NECESARIOS PARA TENER UN SERVIDOR WEB

LUCIA MATEOS ESTEBAN

DAW 2

Desarrollo Web en Entorno Servidor



## Contenido

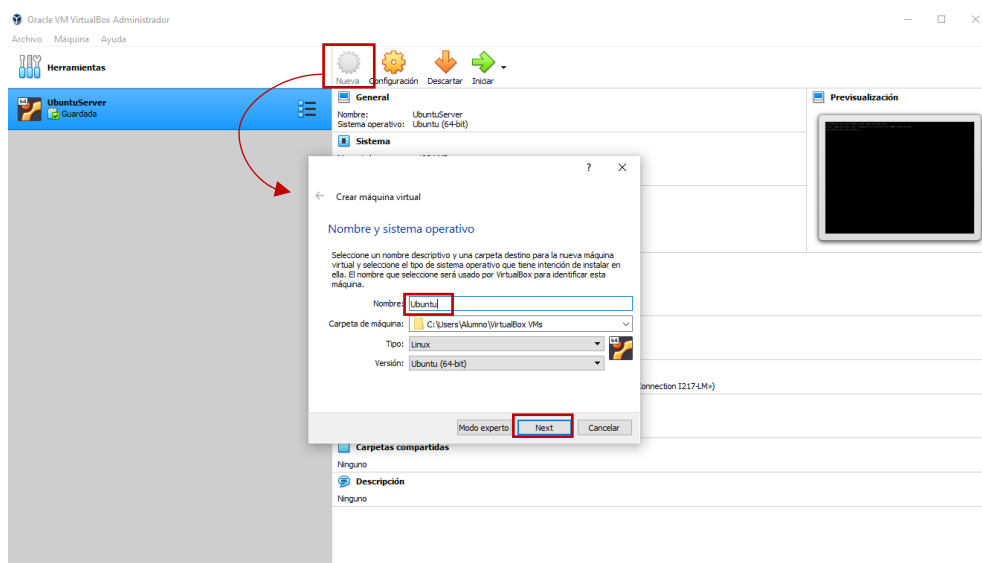
1	Instalar software:.....	2
2	Crear la maquina: .....	2
3	Instalar el sistema operativo: .....	3
4	Configuración del sistema operativo:.....	3
5	Actualizaciones de Ubuntu:.....	6
6	Instalar servidor Apache2:.....	7
7	Instalar PHP: .....	9
8	Configuración de IP fija:.....	11
9	Conexión servidor-VSCode (SSH).....	13
10	Vincular GitHub con el servidor:.....	15

## 1 Instalar software:

- a. Descargamos **Oracle VM VirtualBox** (<https://www.virtualbox.org/>) y la imagen con el sistema operativo **Ubuntu Server** (<https://ubuntu.com/download/server>), en nuestro caso Ubuntu Server 22.04.1.

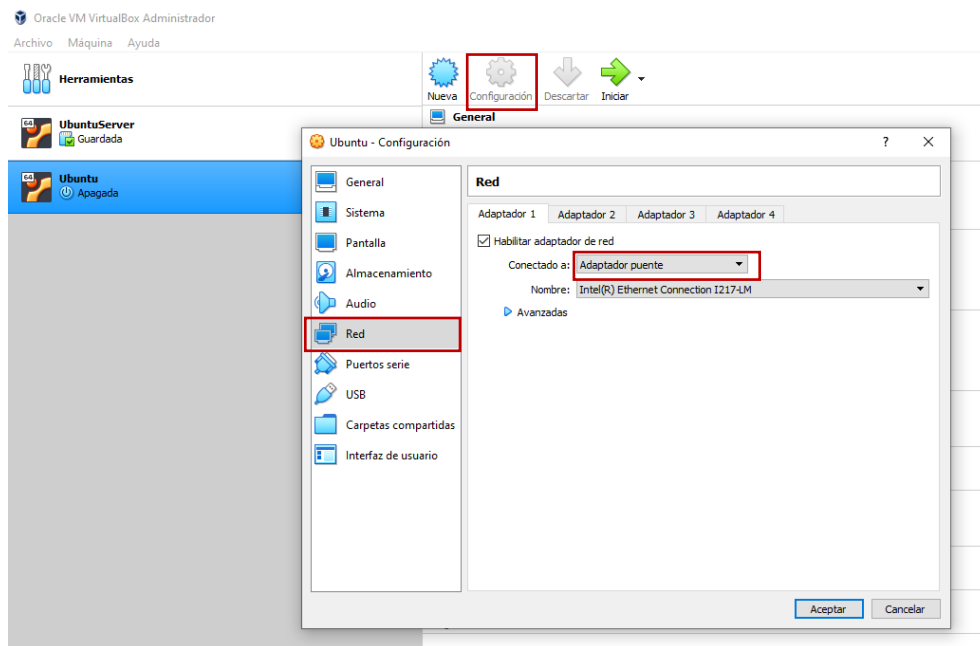
## 2 Crear la maquina:

- a. Abrimos **Oracle VM VirtualBox** y creamos una nueva máquina.
  - La llamaremos **Ubuntu** de tipo **Linux** y versión **Ubuntu(64bits)**
  - Pulsaremos **Next** aceptando todo lo que nos pida para crearla
    1. La memoria que usará nuestra máquina virtual. (*Crear un disco duro virtual ahora*)
    2. El tipo de archivo de disco duro. (*VDI*)
    3. El tamaño del disco que vamos a crear. (*Reservado dinámicamente*)
    4. Establecemos la carpeta donde se crea la unidad de disco duro virtual y su tamaño.



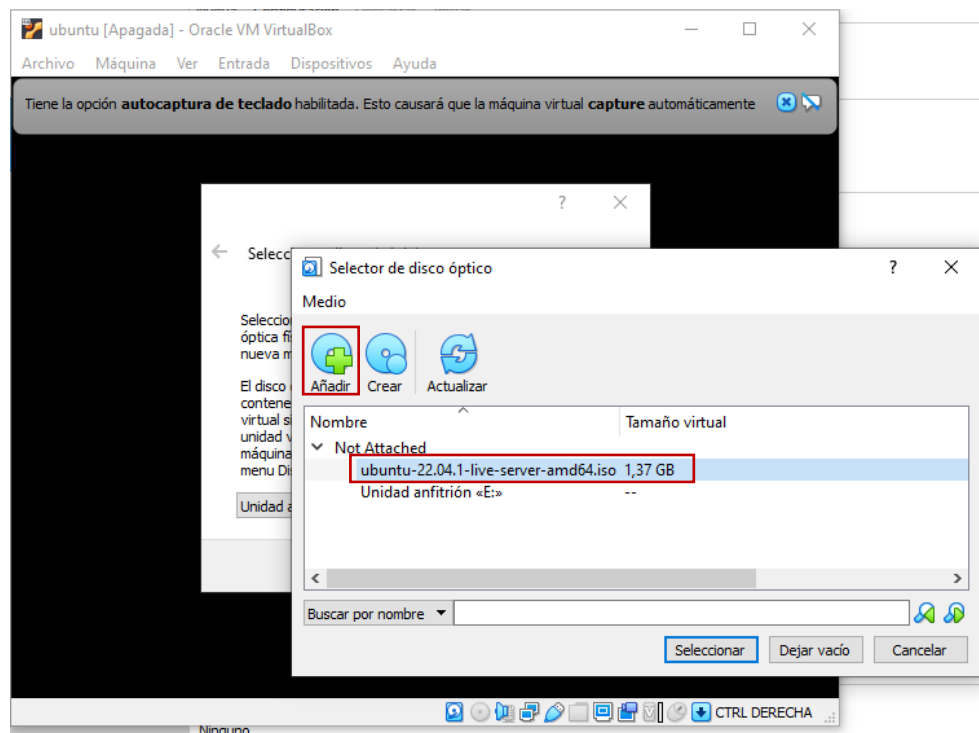
- b. Configuraremos la red antes de iniciar la máquina

- Configuración: Red: Conectado a: sustituimos **NAT** por **Adaptador puente**



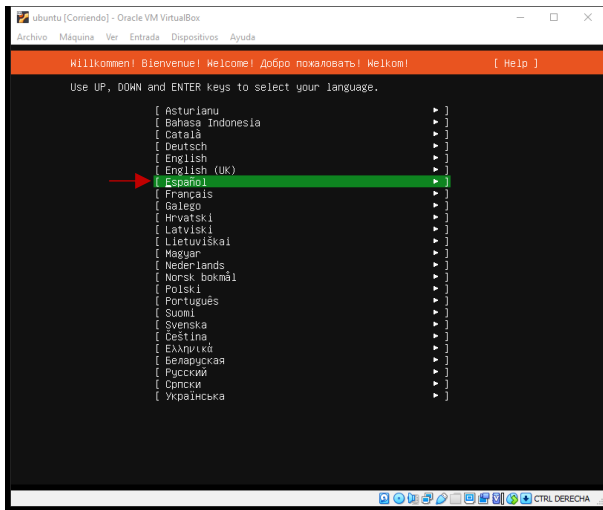
### 3 Instalar el sistema operativo:

- Ya creada la maquina y configurada, pulsamos **Iniciar** y se abrirá una nueva ventana donde nos pide seleccionar disco de inicio.

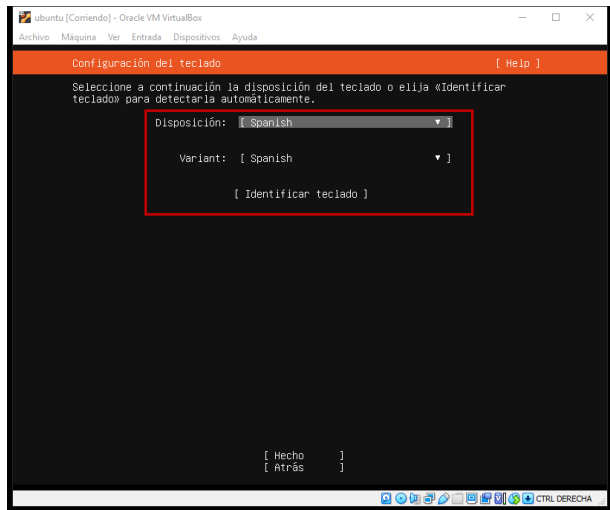


- Si nos aparece en pantalla: *Try or Install Ubuntu Server*, podemos darle a enter o esperar.

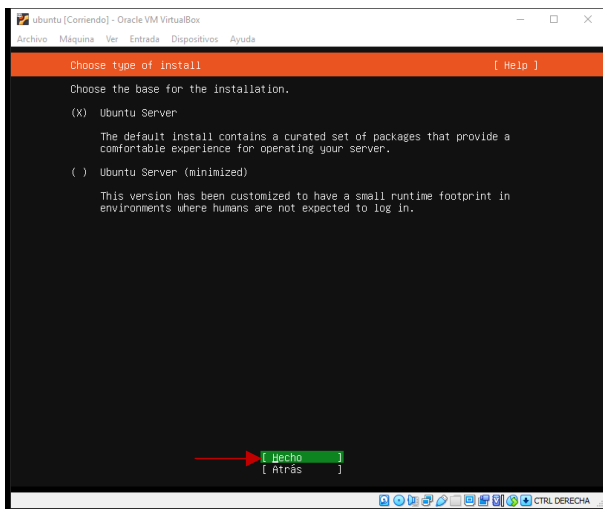
### 4 Configuración del sistema operativo:



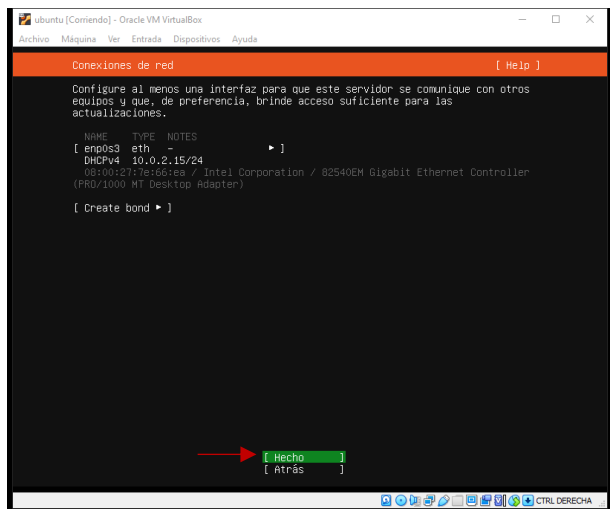
1. Elegimos un idioma



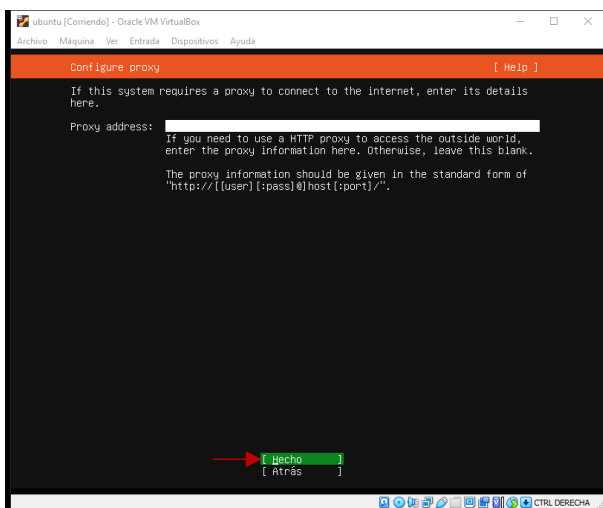
2. Configuración del teclado



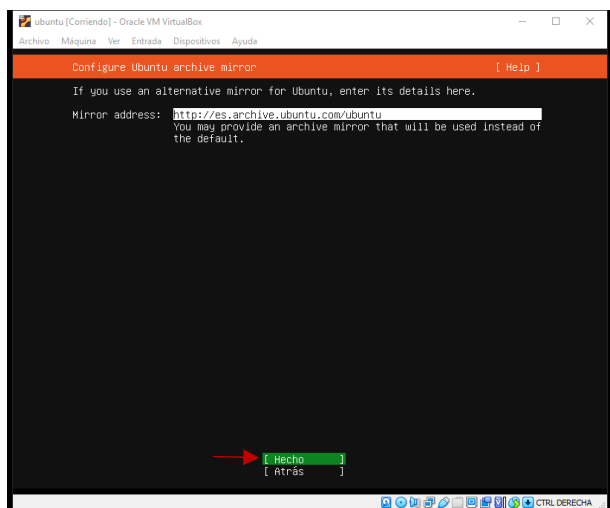
3. Elegimos el tipo de instalación



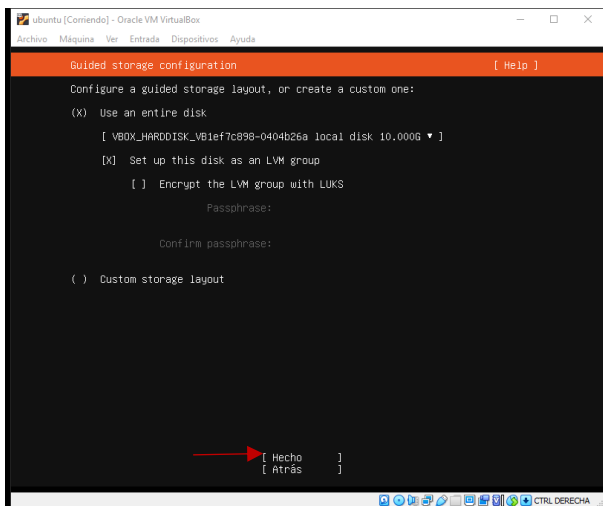
4. Conexión de red (la modificaremos más adelante)



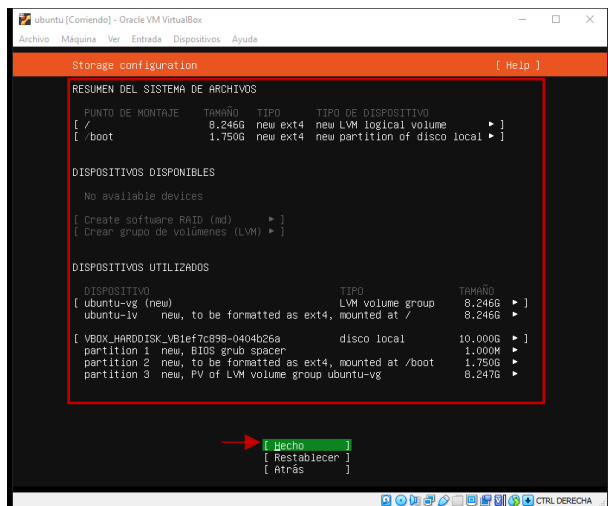
5. Configuración del proxy



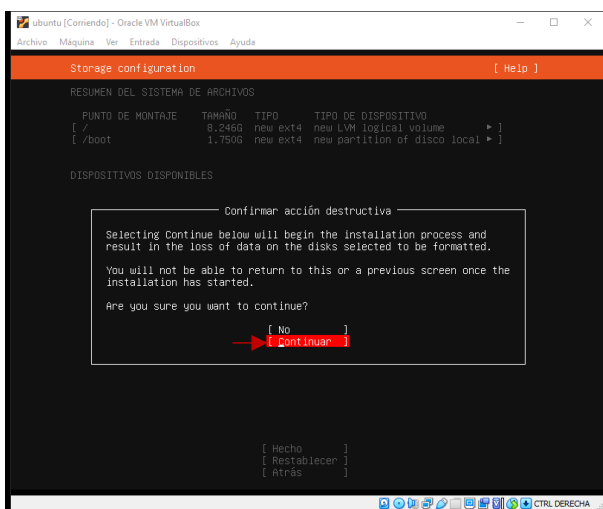
6. Configuración de la dirección Ubuntu



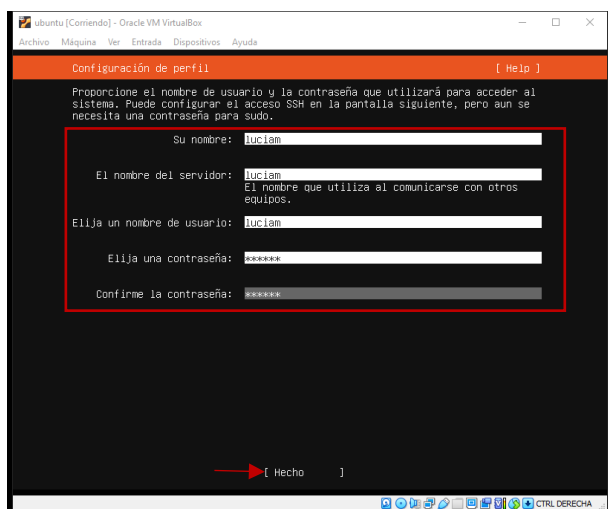
7. Configuración de almacenamiento guiada



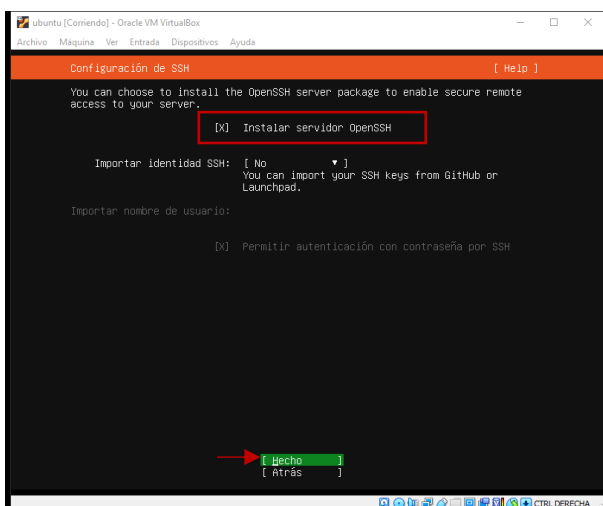
8. Resumen de la configuración



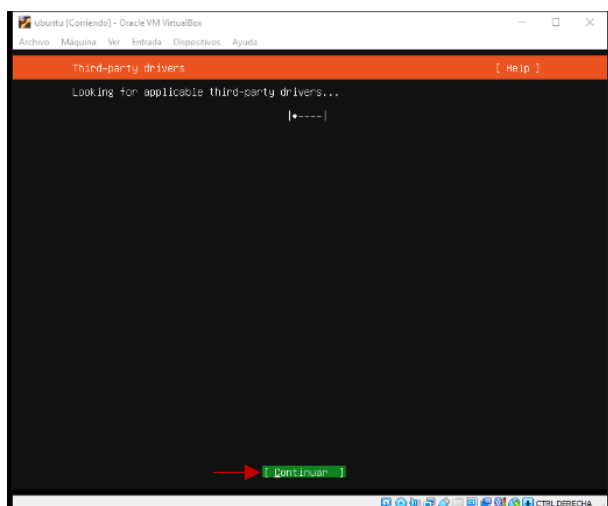
9. Confirmar acción destructiva



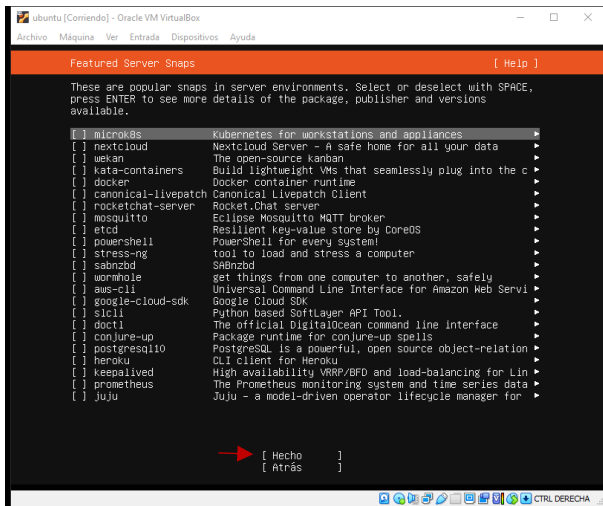
10. Configuración de perfil



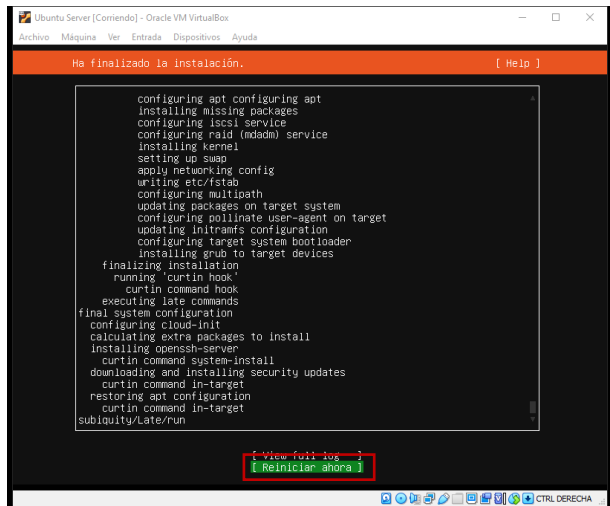
11. Configuración de SSH



12. Controladores de terceros



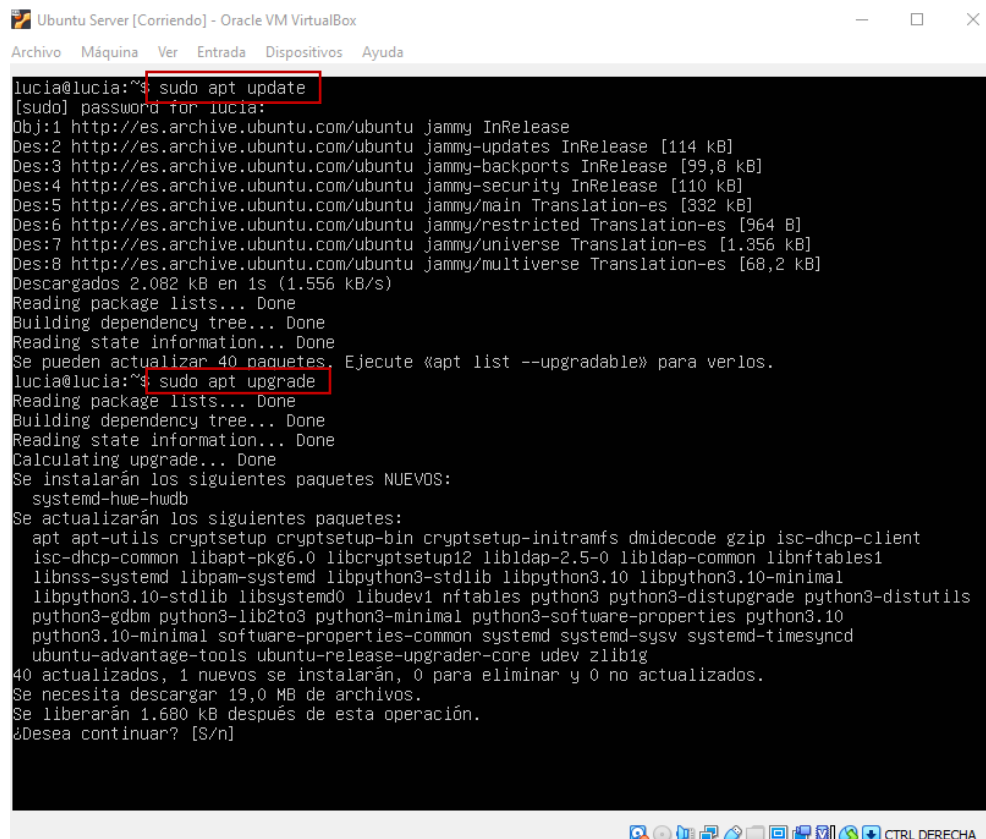
13. Entornos de servidor



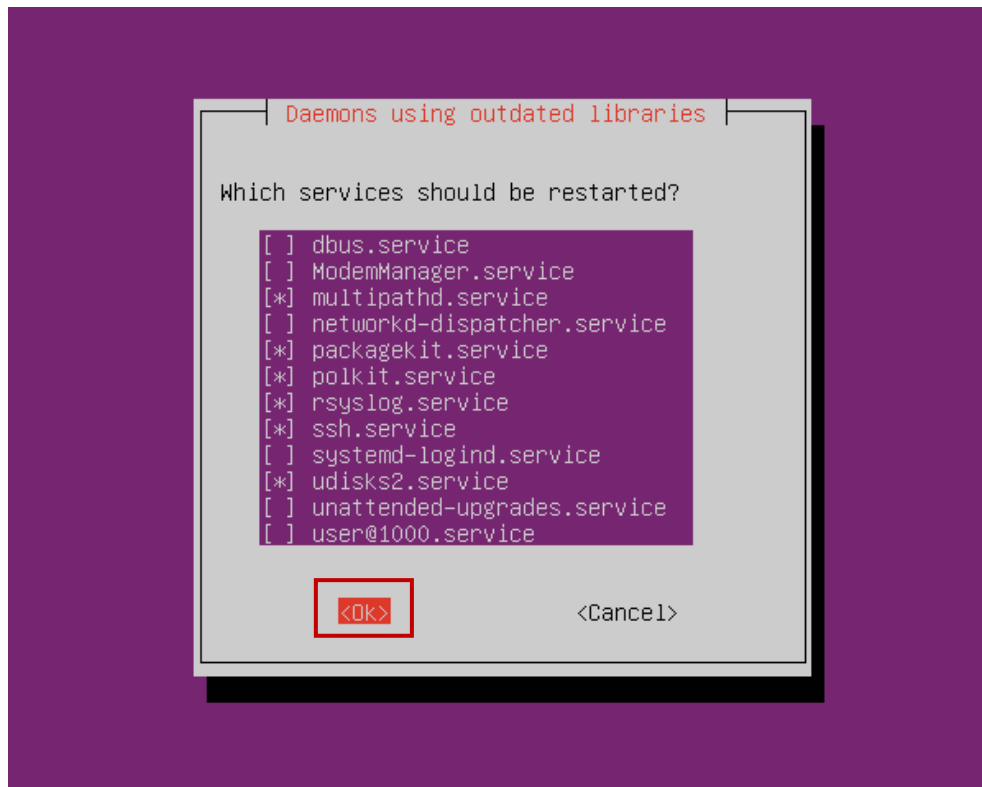
14. Instalación finalizada

## 5 Actualizaciones de Ubuntu:

- Iniciado el sistema operativo, debemos ejecutar el comando **sudo apt update** para saber si tenemos actualizaciones para Ubuntu. *(nos pedirá la contraseña para ejecutar dicho comando)*
- Si tenemos que instalar dichas actualizaciones usaremos el comando **sudo apt upgrade** *(nos pide confirmación para realizar la instalación)*



- c. Cuando finalicen las actualizaciones nos pedirá reiniciar, dejaremos todo como aparece y le damos a **OK**



## 6 Instalar servidor Apache2:

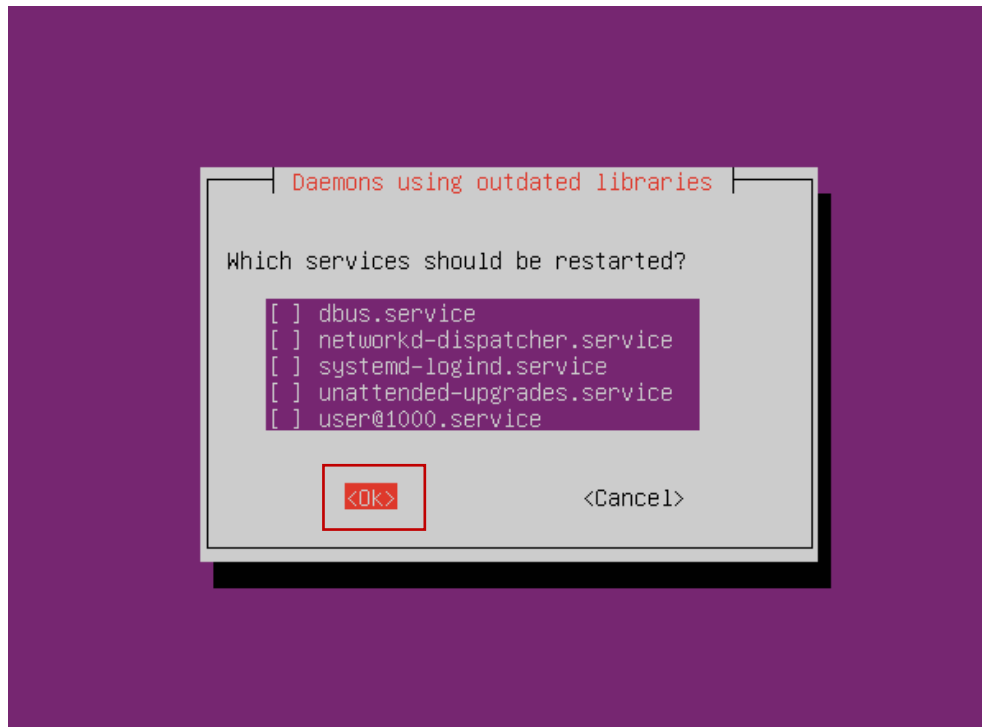
- a. Para instalarlo usaremos el comando **sudo apt install apache2**

```
Ubuntu Server [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

lucia@lucia:~$ sudo apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  apache2-bin apache2-data apache2-utils bzip2 libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap liblua5.3-0 mailcap mime-support ssl-cert
Paquetes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser bzip2-doc
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-utils bzip2 libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap liblua5.3-0 mailcap mime-support ssl-cert
0 actualizados, 13 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 2,138 kB de archivos.
Se utilizarán 8,501 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] _
```

- b. Cuando finalicen la instalación nos pedirá reiniciar, dejaremos todo como aparece y le damos a **OK**





- c. Comprobamos si lo hemos instalado correctamente con el comando **systemctl status apache2** (nos tendrá que aparecer *active: active (running)*)

```
Ubuntu Server [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

lucia@lucia:~$ systemctl status apache2
• apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-09-30 20:07:25 UTC; 2min 34s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 14621 (apache2)
      Tasks: 55 (limit: 1030)
     Memory: 5.1M
        CPU: 67ms
    CGroup: /system.slice/apache2.service
            └─14621 /usr/sbin/apache2 -k start
              └─14623 /usr/sbin/apache2 -k start
                └─14624 /usr/sbin/apache2 -k start

sep 30 20:07:25 lucia systemd[1]: Starting The Apache HTTP Server...
sep 30 20:07:25 lucia apachectl[14620]: AH00558: apache2: Could not reliably determine the server's
sep 30 20:07:25 lucia systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

- d. También podemos comprobar el anterior paso entrando en **index.html** (este archivo lo encontraremos en la ruta */var/www/html*) desde nuestro navegador con la IP de nuestra maquina que obtenemos con el comando **ipconfig**.

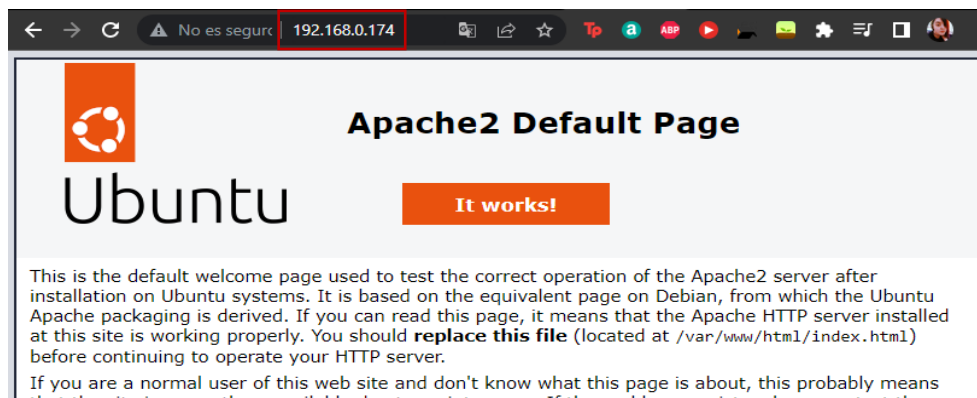
```
Ubuntu Server [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

lucia@lucia:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.174 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::a00:27ff:fea5:70d2 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:a5:70:d2 txqueuelen 1000 (Ethernet)
    RX packets 16752 bytes 24847988 (24.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5342 bytes 375980 (375.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 136 bytes 12184 (12.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 136 bytes 12184 (12.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lucia@lucia:~$
```

- e. Así veríamos la página html (<http://192.168.0.214/index.html>)



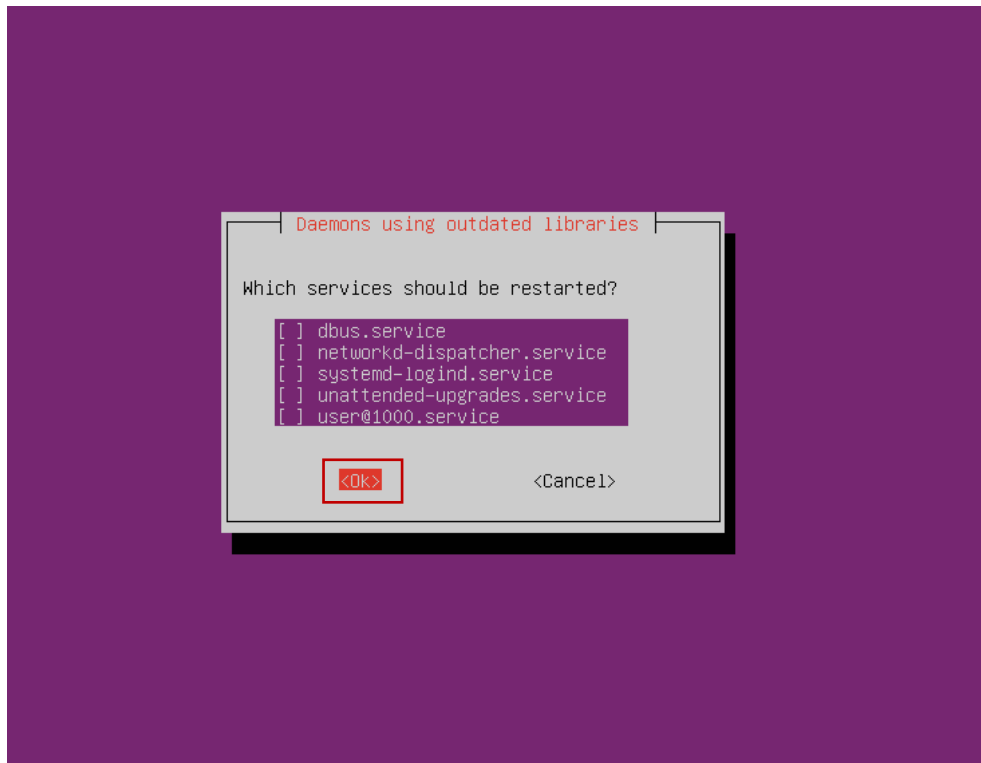
## 7 Instalar PHP:

- a. Para instalar el php usaremos el comando **sudo apt install php**

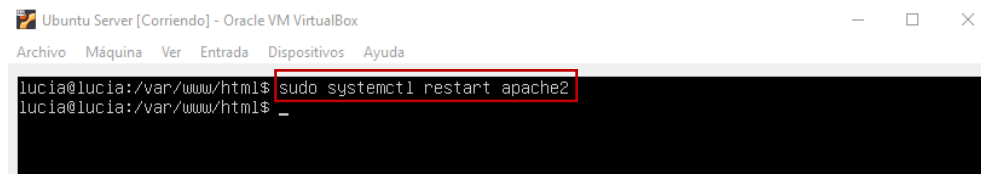
```
Ubuntu Server [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

lucia@lucia:/var/www/html$ sudo apt install php
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libapache2-mod-php8.1 php-common php8.1 php8.1-cli php8.1-common php8.1-opcache php8.1-readline
Paquetes sugeridos:
  php-pear
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php8.1 php php-common php8.1 php8.1-cli php8.1-common php8.1-opcache
  php8.1-readline
0 actualizados, 8 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 5.126 kB de archivos.
Se utilizarán 21,3 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] _
```

- b. Cuando finalicen la instalación nos pedirá reiniciar, dejaremos todo como aparece y le damos a **OK**



- c. Para reiniciar usamos el comando **sudo systemctl restart apache2**

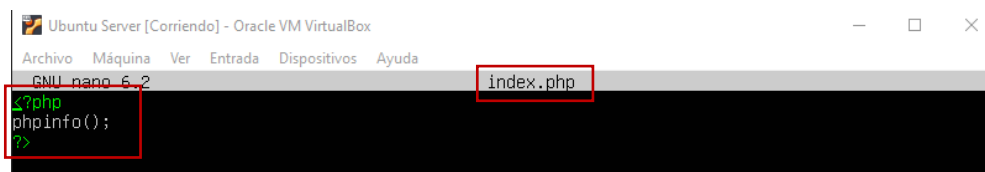


- d. Comprobamos que la instalación es correcta creando un archivo **index.php** para ello usamos el comando **sudo nano index.php** (creamos el archivo en la ruta *var/www/html*)



- e. Se nos abrirá el nuevo archivo y escribiremos el siguiente código:

- **<?php phpinfo (); ¿>**



- f. Para comprobar que funciona introducimos en nuestro navegador esta ruta:

- ***http://192.168.0.214/index.php***

PHP Version 8.1.2	
System	Linux lucia 5.15.0-48-generic #54-Ubuntu SMP Fri Aug 26 13:26:29 UTC 2022 x86_64
Build Date	Aug 8 2022 07:28:23
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/apache2
Loaded Configuration File	/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/conf.d/10-pdo.ini, /etc/php/8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20-ctype.ini, /etc/php/8.1/apache2/conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-fileinfo.ini, /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/20-gettext.ini, /etc/php/8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-imagick.ini, /etc/php/8.1/apache2/conf.d/20-ldap.ini, /etc/php/8.1/apache2/conf.d/20-mbstring.ini, /etc/php/8.1/apache2/conf.d/20-mcrypt.ini, /etc/php/8.1/apache2/conf.d/20-mysqlnd.ini, /etc/php/8.1/apache2/conf.d/20-openssl.ini, /etc/php/8.1/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.1/apache2/conf.d/20-sysvsem.ini, /etc/php/8.1/apache2/conf.d/20-sysvshm.ini, /etc/php/8.1/apache2/conf.d/20-tokenizer.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902.NTS
PHP Extension Build	API20210902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.*

## 8 Configuración de IP fija:

- Para nuestra comodidad a la hora de trabajar estableceremos la IP, para ello debemos cambiar el archivo **00-installer-config.yaml** que la encontraremos en la ruta **/etc/netplan**

```

Ubuntu Server [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
lucia@lucia:~$ cd /etc/netplan/
lucia@lucia:/etc/netplan$ ls -l
total 4
-rw-r--r-- 1 root root 117 sep 30 10:52 00-installer-config.yaml
lucia@lucia:/etc/netplan$ sudo nano 00-installer-config.yaml

```

- Tendremos en cuenta la IP de nuestro ordenador, para obtenerla:
  - Pulsamos **[Windows] + R** que abrirá el cuadro de dialogo **“Ejecutar”**
  - En el cuadro de dialogo, buscaremos **CMD**
  - Nos abrirá **cmd.exe**, donde escribiremos **IPCONFIG**
  - Obtendremos toda la información y datos de la configuración del equipo y entre ella buscaremos **Dirección IPv4**

```

Símbolo del sistema
Estado de los medios. . . . . : medios desconectados
Sufijo DNS específico para la conexión. . :

Adaptador de Ethernet VirtualBox Host-Only Network:

Sufijo DNS específico para la conexión. . :
Vínculo: dirección IPv6 local. . . : fe80::e94e:53e8:aa06:fe31%3
Dirección IPv4. . . . . : 192.168.56.1
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . :

Adaptador de LAN inalámbrica Conexión de área local* 1:

Estado de los medios. . . . . : medios desconectados
Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Conexión de área local* 10:

Estado de los medios. . . . . : medios desconectados
Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Wi-Fi:

Sufijo DNS específico para la conexión. . : technicolor.net
Vínculo: dirección IPv6 local. . . : fe80::592c:dae2:5ef2:dac%10
Dirección IPv4. . . . . : 192.168.0.204
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.0.1

C:\Users\Lucia>

```

c. Teniendo ya toda la información del anterior paso, volvemos a la maquina y en el archivo **00-installer-config.yaml** introducimos los siguientes datos:

e. **dhcp4: no**

f. **addresses: [192.168.0.214/24]**

g. **routes:**

- **to: default**

via: **192.168.0.1** (nuestra puerta de enlace predeterminada)

h. **Nameservers:**

**addresses: [8.8.8.8]** (dns de Google)

```

GNU nano 6.2 00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    ethp0s3:
      dhcp4: no
      addresses: [192.168.0.214/24]
      routes:
        - to: default
          via: 192.168.0.1
      nameservers:
        addresses: [8.8.8.8]
  version: 2

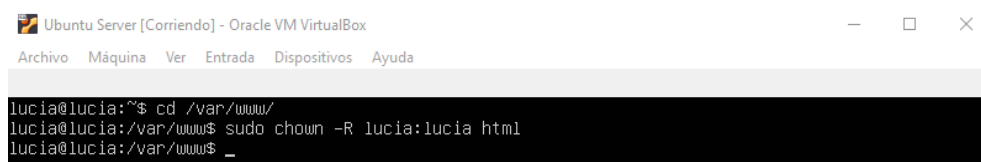
```

d. Comprobamos que todo está bien con el comando **sudo netplan apply**, si no da errores, reiniciamos y con el comando **ifconfig** podemos ver el cambio.

```
lucia@lucia:/etc/netplan$ sudo netplan apply
lucia@lucia:/etc/netplan$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.214 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::a00:27ff:fea5:70d2 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:a5:70:d2 txqueuelen 1000 (Ethernet)
    RX packets 116016 bytes 174178937 (174.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 28215 bytes 1956744 (1.9 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 216 bytes 20032 (20.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 216 bytes 20032 (20.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

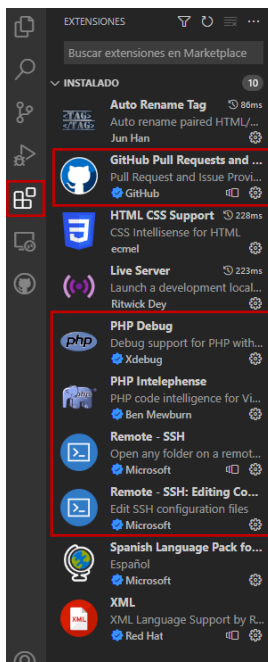
- e. Para poder trabajar con el servidor debemos dar permisos desde Linux a nuestro usuario **sudo chown -R lucia:lucia html**.



```
lucia@lucia:~$ cd /var/www/
lucia@lucia:/var/www$ sudo chown -R lucia:lucia html
lucia@lucia:/var/www$ _
```

## 9 Conexión servidor-VSCode (SSH)

- Debemos instalar el programa **Visual Studio Code**.
- Abrimos el programa y añadimos estas extensiones:



**GitHub Pull Requests and Issues:** Esta extensión le permite revisar y administrar solicitudes de extracción y problemas de GitHub en Visual Studio Code

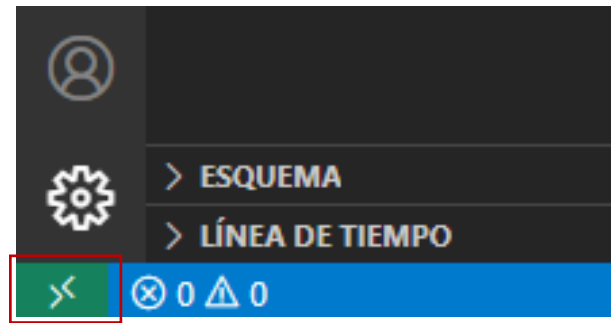
**PHP Debug:** Permite ejecutar el código paso por paso. Evaluar el contenido de las variables en tiempo real. Establecer puntos de corte.

**PHP Intelephense:** Analiza todos los archivos de tu proyecto y proporciona información necesaria para el autocompletado.

**Remote - SSH:** La extensión Remote - SSH le permite usar cualquier máquina remota con un servidor SSH como su entorno de desarrollo.

**Remote - SSH: Editing Configuration Files:** Esta extensión complementa la extensión Remote - SSH con colorización de sintaxis, inteligencia de palabras clave y fragmentos simples al editar archivos de configuración SSH.

- En la parte inferior izquierda pulsaremos en este icono:



d. Seguiremos los siguientes pasos:

- Conectar a hosts...
- Agregar nuevo host SSH...
- Nos pide que escribamos el comando de conexión SSH y pondremos **ssh lucia@192.168.0.214 -p 22**. Podemos ajustar esta información en el archivo ssh file:

**Host 192.168.0.214**

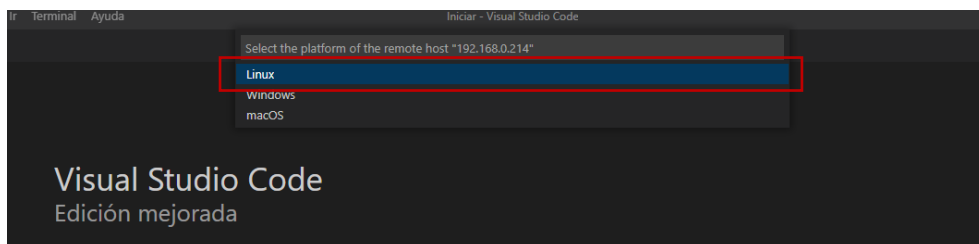
**HostName 192.168.0.214**

**User lucia**

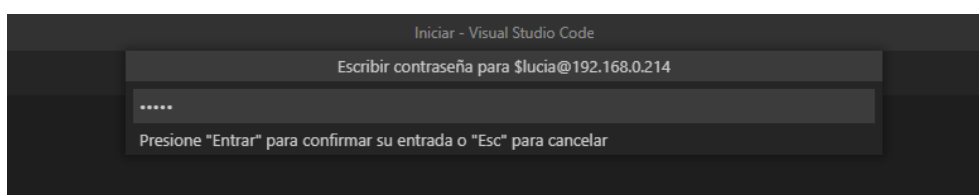
**Protocol 2**

**Port 22**

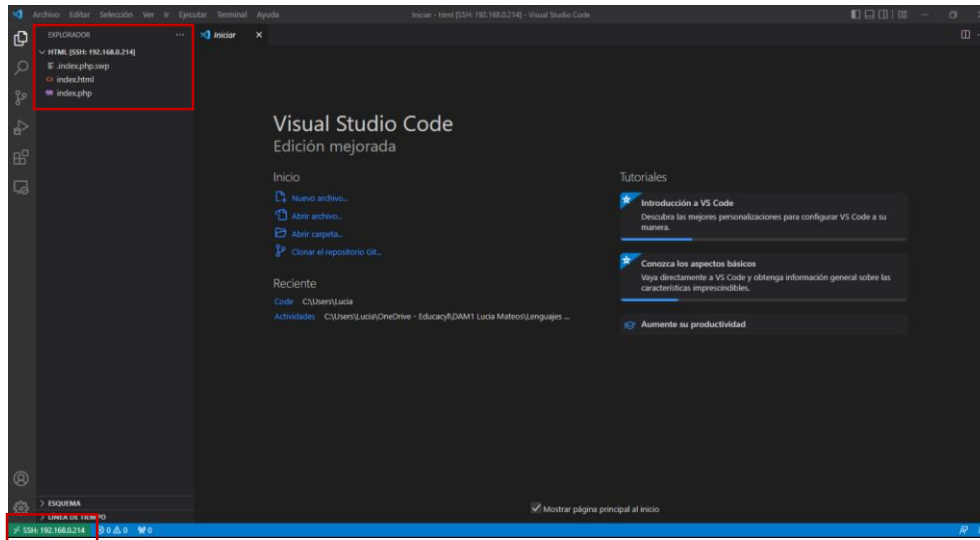
- Se abrirá una ventana nueva de VSCode que nos pedirá seleccionar la plataforma del host remoto (**Linux**)



- Confirmaremos y nos pedirá contraseña para aceptar la conectarnos

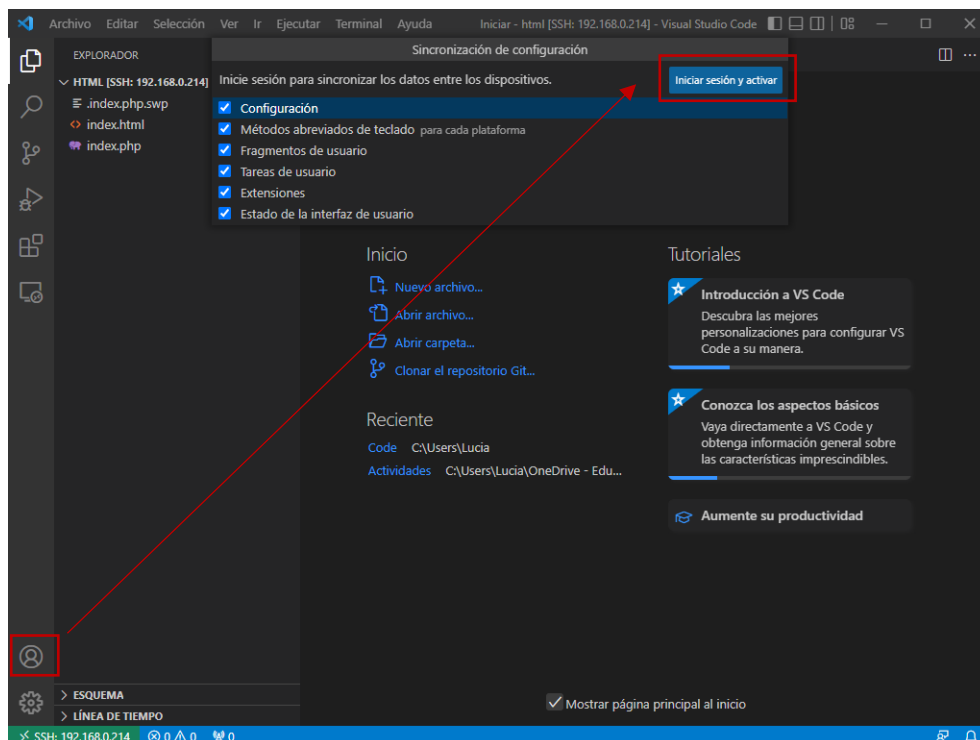


- Ya conectados nos aparecerá en la parte inferior izquierda la IP, el siguiente paso será abrir la carpeta de `/var/www/html/`, nos volverá a pedir la contraseña y nos mostrará todo lo que tenemos en esa carpeta



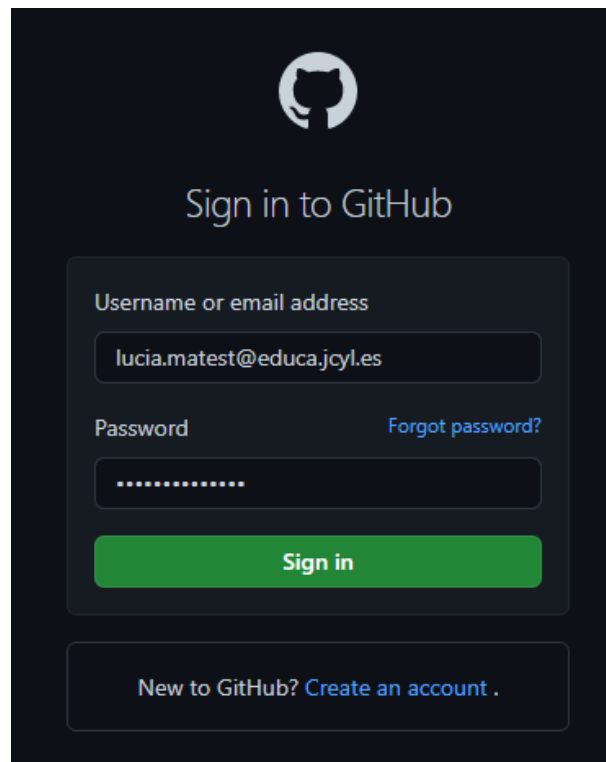
## 10 Vincular GitHub con el servidor:


- a. En VS Code vamos a **cuentas** y pulsamos el botón **iniciar sesión y activar**

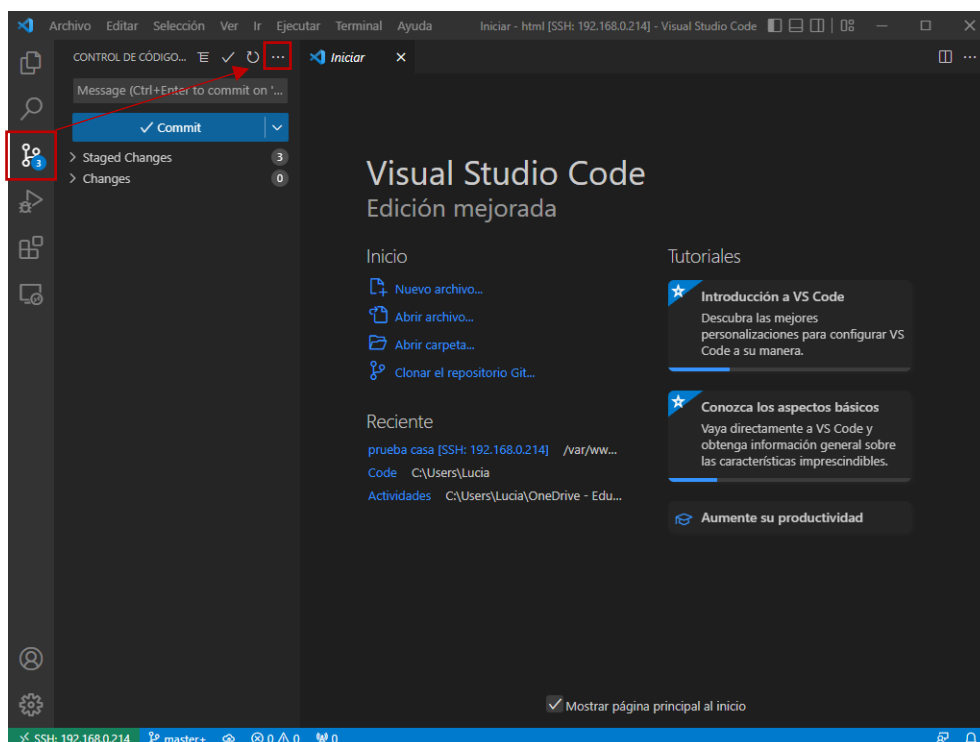


- b. Desde la página <https://github.com/> creamos una cuenta (esto nos servirá para poder sincronizar el trabajo que vamos haciendo).

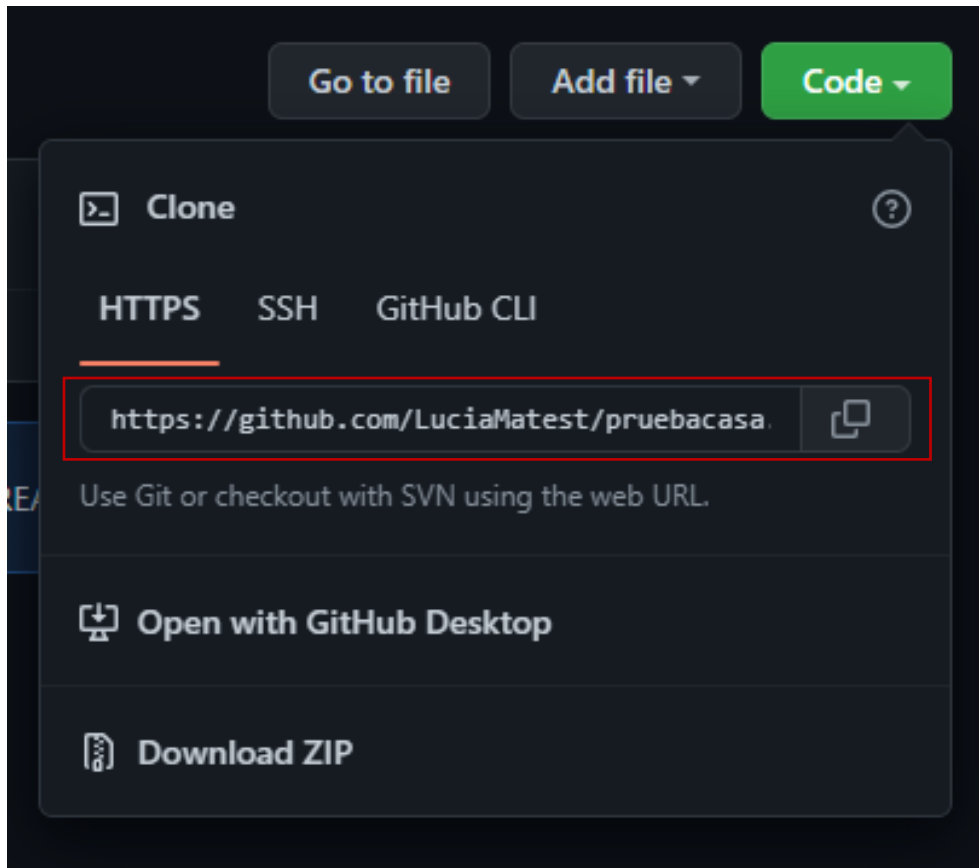




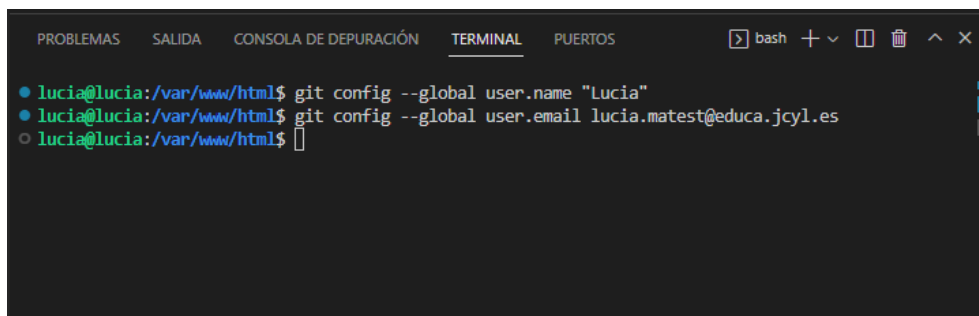
- c. Para poder sincronizarnos iremos al apartado **Control de código**, vamos al menú (  ) y seleccionamos la opción **Remote – Add remote**.



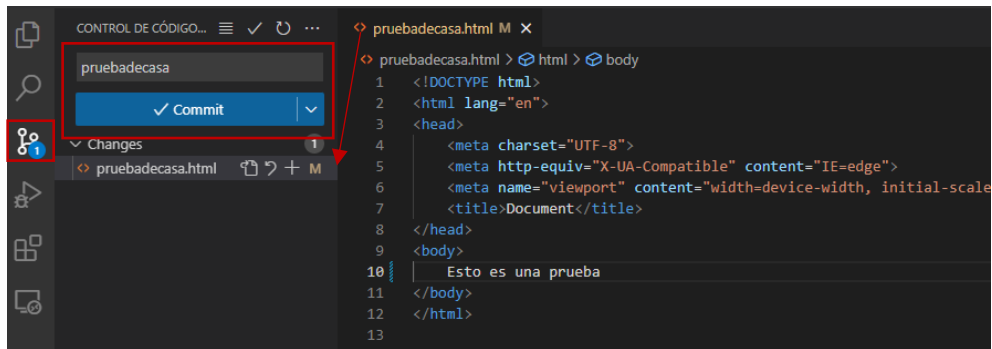
- d. Debemos introducir la **url** de nuestro repositorio que encontraremos en **GitHub**:



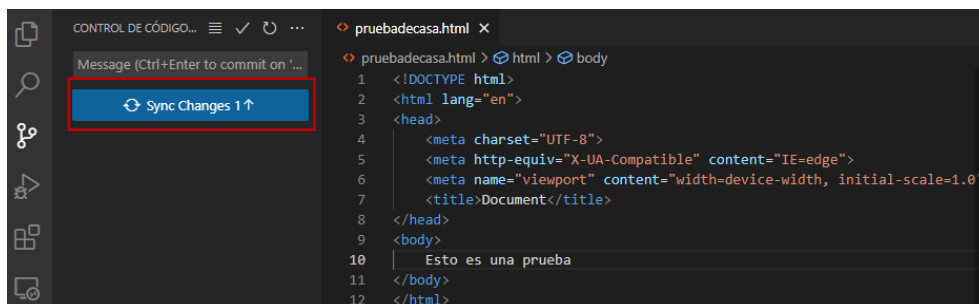
- e. Desde el **Terminal** de **VSCode** configuraremos *usuario* y *email* de github ejecutando los siguientes comandos:



- f. Para comprobar que todo esté sincronizado y funcionando escribimos en un archivo vinculado desde VSCode, guardamos y nos vamos a el control de codigos escribimos el nombre en la barra y pulsamos el botón **Commit** o entramos en el menú y pulsamos **Push**



- g. Al hacer esto, cambiará el botón a **Sync Changes** y podemos comprobar el cambio desde GitHub



- h. Si hacemos cambios al mismo tiempo desde VS Code y Github en la misma línea, al sincronizarlos de nuevo, nos aparecerá este aviso:

