3 de octubre de 2022

Lucia Mateos esteban

Daw 2

Desarrollo Web en Entorno Servidor

Manual instalación

Pasos necesarios para tener un servidor web

Contenido

[1 Instalar software: 2](#_Toc115646999)

[2 Crear la maquina: 2](#_Toc115647000)

[3 Instalar el sistema operativo: 3](#_Toc115647001)

[4 Configuración del sistema operativo: 3](#_Toc115647002)

[5 Actualizaciones de Ubuntu: 6](#_Toc115647003)

[6 Instalar servidor Apache2: 7](#_Toc115647004)

[7 Instalar PHP: 9](#_Toc115647005)

[8 Configuración de IP fija: 11](#_Toc115647006)

[9 Conexión servidor-VSCode (SSH) 13](#_Toc115647007)

[10 Vincular GitHub con el servidor: 15](#_Toc115647008)

## Instalar software:

* 1. 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.

## Crear la maquina:

1. 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.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Configuraremos la red antes de iniciar la máquina
   * + Configuración: Red: Conectado a: sustituimos **NAT** por **Adaptador puente**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

## Instalar el sistema operativo:

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

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

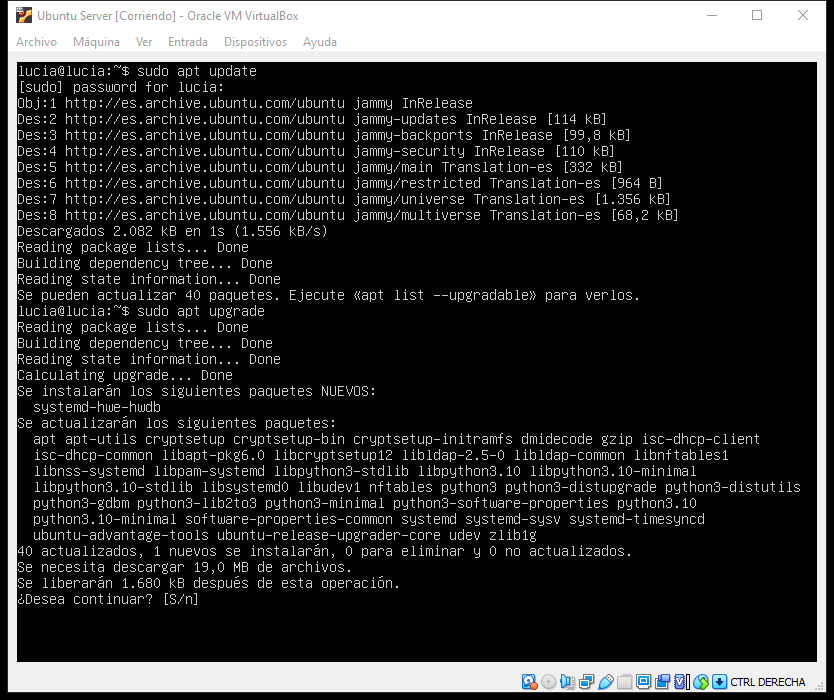
1. Si nos aparece en pantalla: *Try or Install Ubuntu Server*, podemos darle a enter o esperar.

## Configuración del sistema operativo:

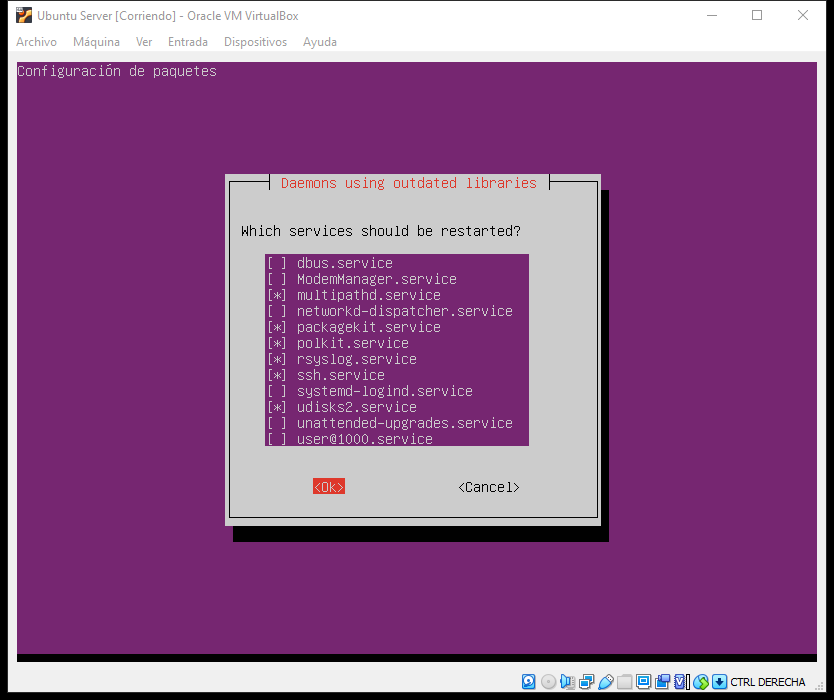
|  |  |
| --- | --- |
| ubuntu [Corriendo] - Oracle VM VirtualBox  .Elegimos un idioma | ubuntu [Corriendo] - Oracle VM VirtualBox  .Configuración del teclado |
| ubuntu [Corriendo] - Oracle VM VirtualBox  . Elegimos el tipo de instalación | ubuntu [Corriendo] - Oracle VM VirtualBox  .Conexión de red (la modificaremos más adelante) |
| ubuntu [Corriendo] - Oracle VM VirtualBox  .Configuración del proxy | ubuntu [Corriendo] - Oracle VM VirtualBox  .Configuración de la dirección Ubuntu |
| ubuntu [Corriendo] - Oracle VM VirtualBox  .Configuración de almacenamiento guiada | ubuntu [Corriendo] - Oracle VM VirtualBox  .Resumen de la configuración |
| ubuntu [Corriendo] - Oracle VM VirtualBox  .Confirmar acción destructiva | ubuntu [Corriendo] - Oracle VM VirtualBox  .Configuración de perfil |
| ubuntu [Corriendo] - Oracle VM VirtualBox  .Configuración de SSH | ubuntu [Corriendo] - Oracle VM VirtualBox  .Controladores de terceros |
| ubuntu [Corriendo] - Oracle VM VirtualBox  .Entornos de servidor | Ubuntu Server [Corriendo] - Oracle VM VirtualBox  .Instalación finalizada |

## Actualizaciones de Ubuntu:

1. 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)*
2. Si tenemos que instalar dichas actualizaciones usaremos el comando **sudo apt upgrade** *(nos pide confirmación para realizar la instalación)*

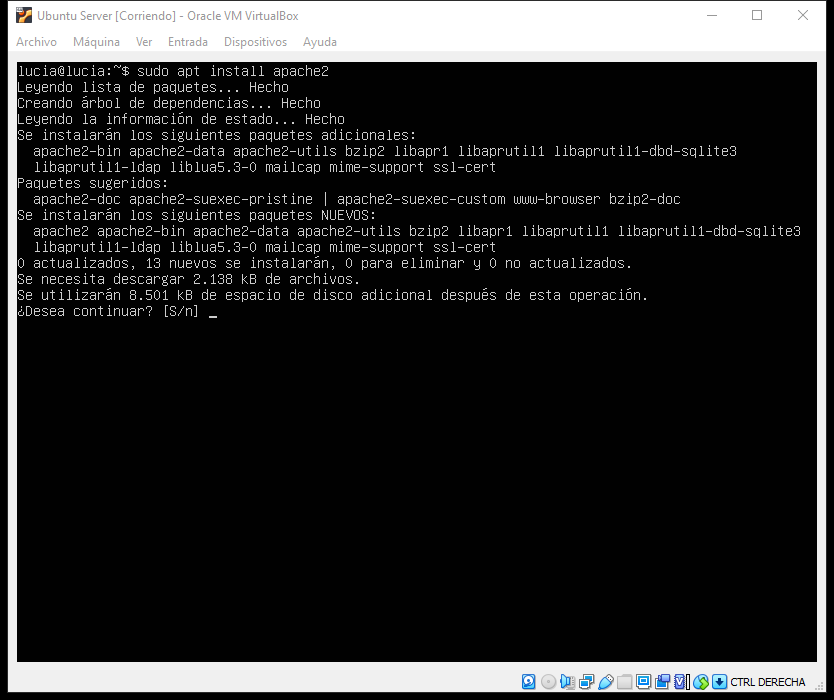


1. Cuando finalicen las actualizaciones nos pedirá reiniciar, dejaremos todo como aparece y le damos a **OK**

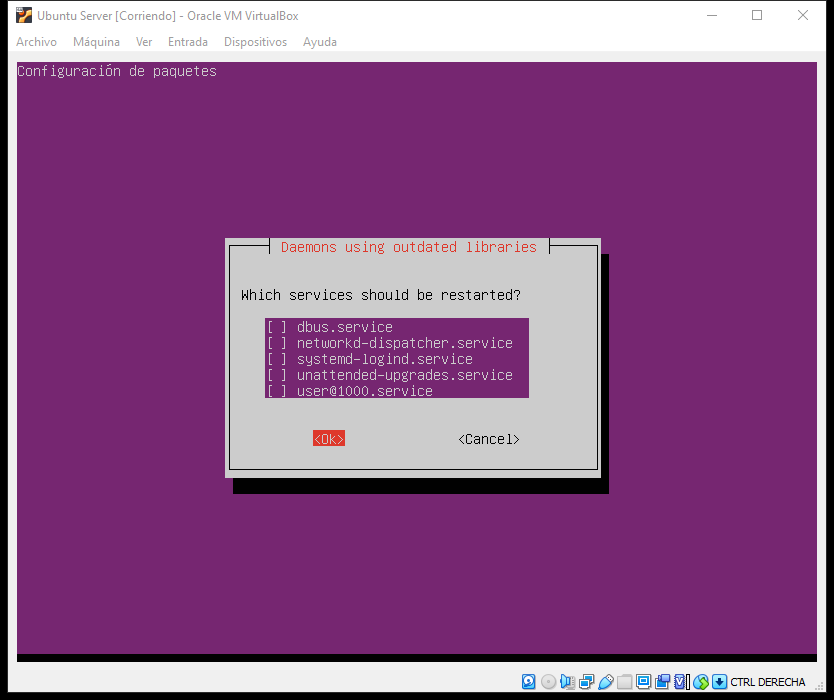


## Instalar servidor Apache2:

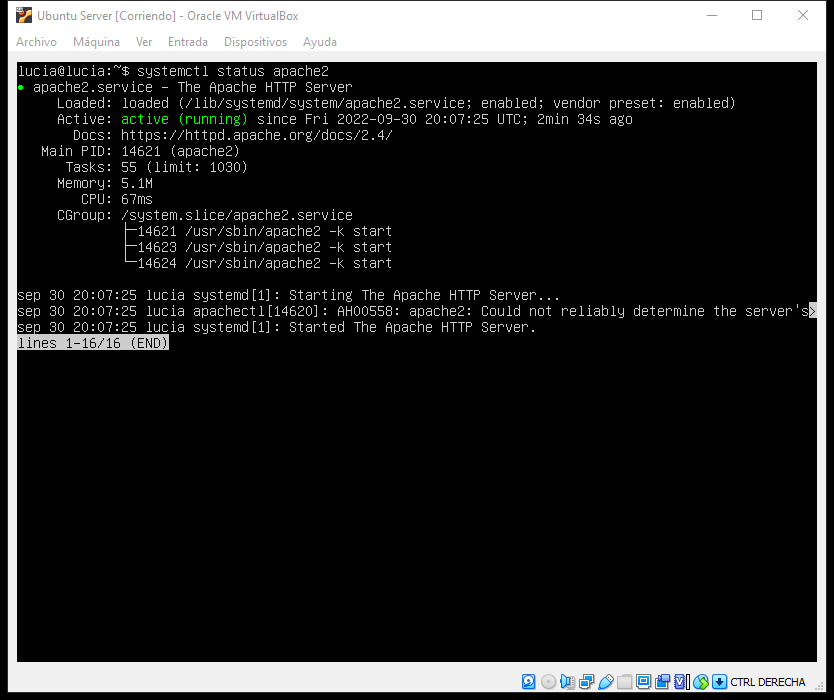
1. Para instalarlo usaremos el comando **sudo apt install apache2**



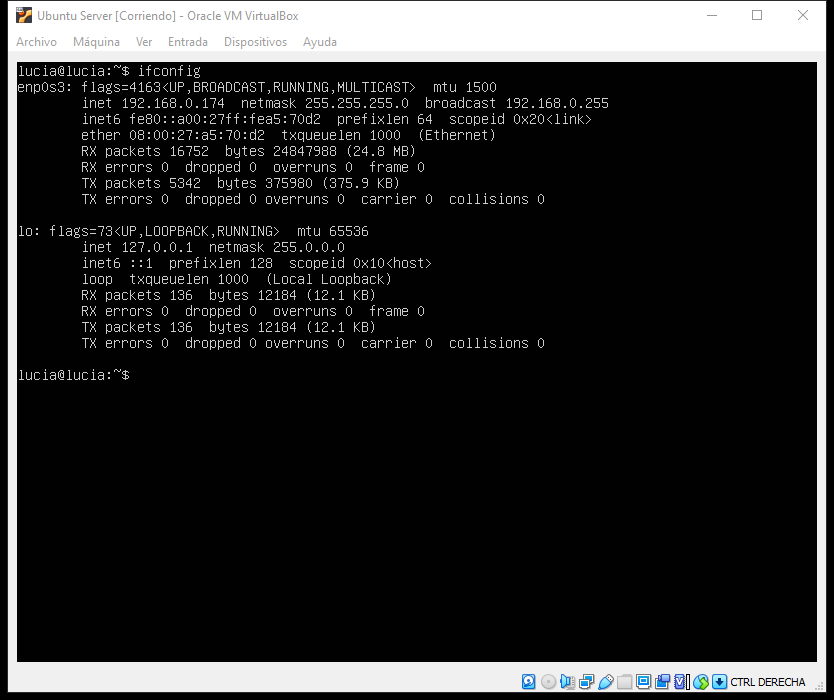
1. Cuando finalicen la instalación nos pedirá reiniciar, dejaremos todo como aparece y le damos a **OK**



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



1. 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.**

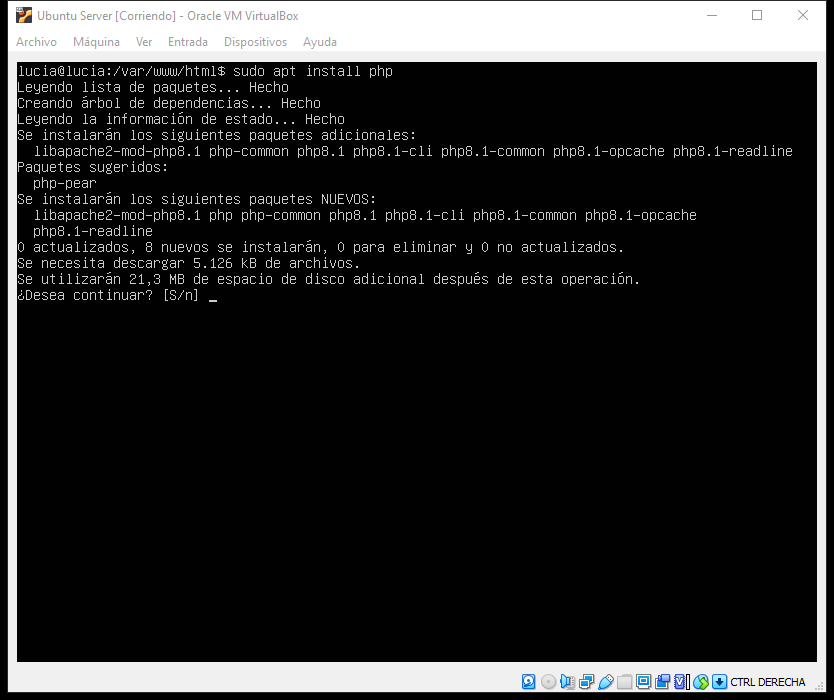


1. Así veríamos la página html (*http://192.168.0.214/index.html*)

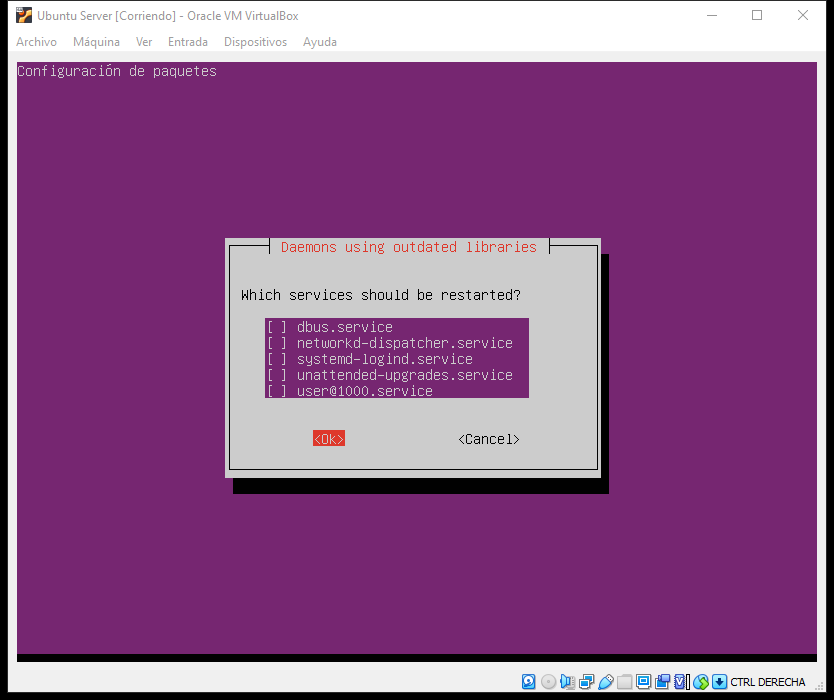


## Instalar PHP:

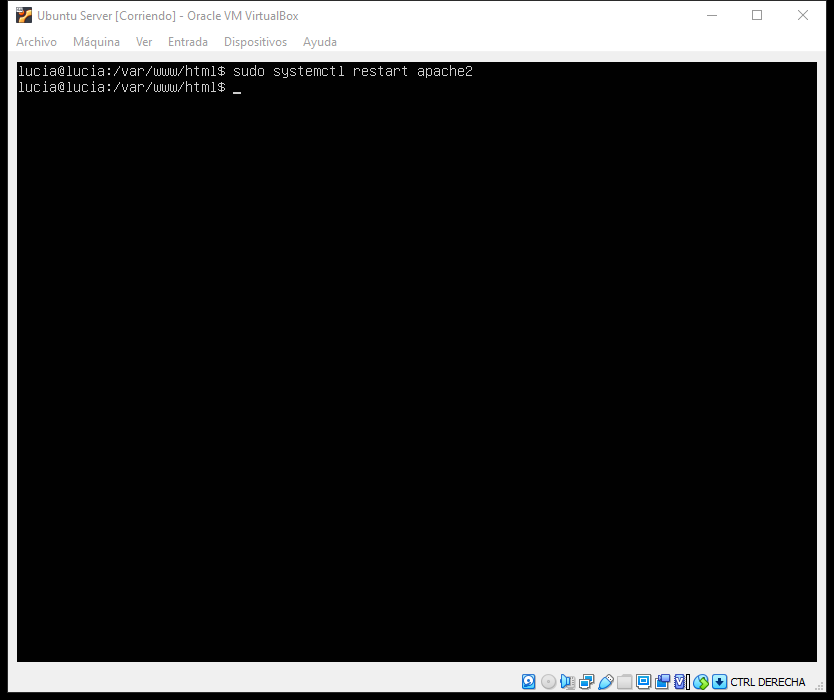
1. Para instalar el php usaremos el comando **sudo apt install php**



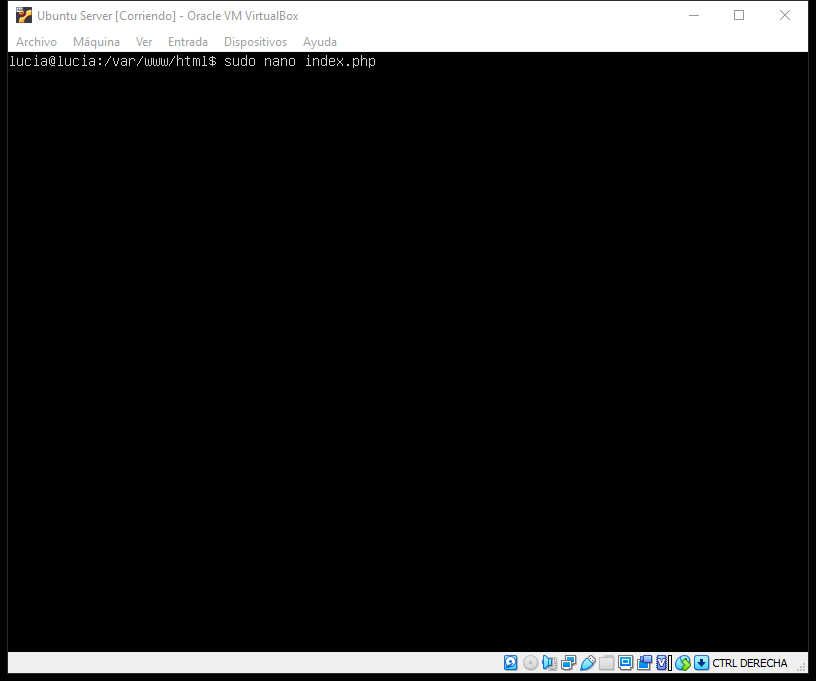
1. Cuando finalicen la instalación nos pedirá reiniciar, dejaremos todo como aparece y le damos a **OK**



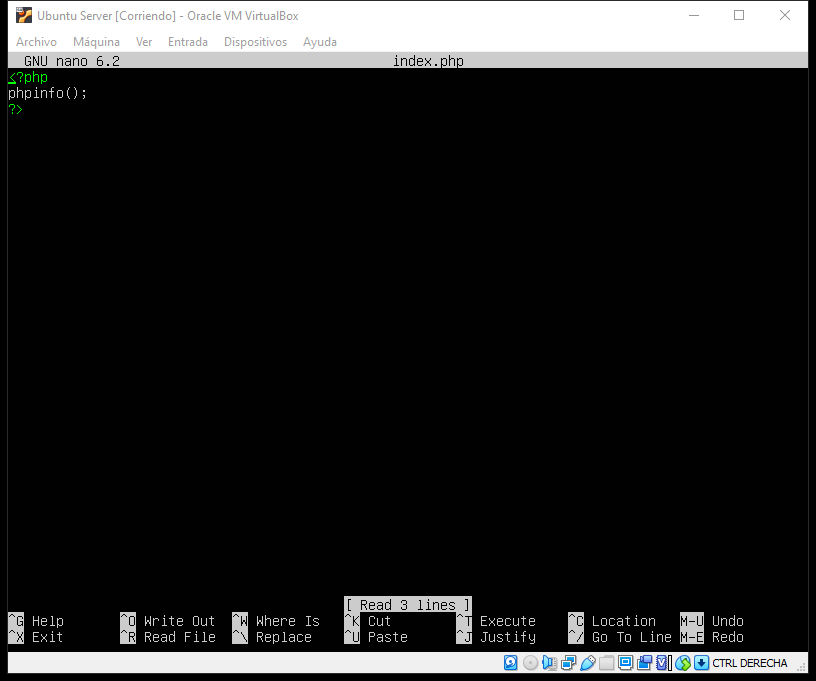
1. Para reiniciar usamos el comando **sudo systemctl restart apache2**



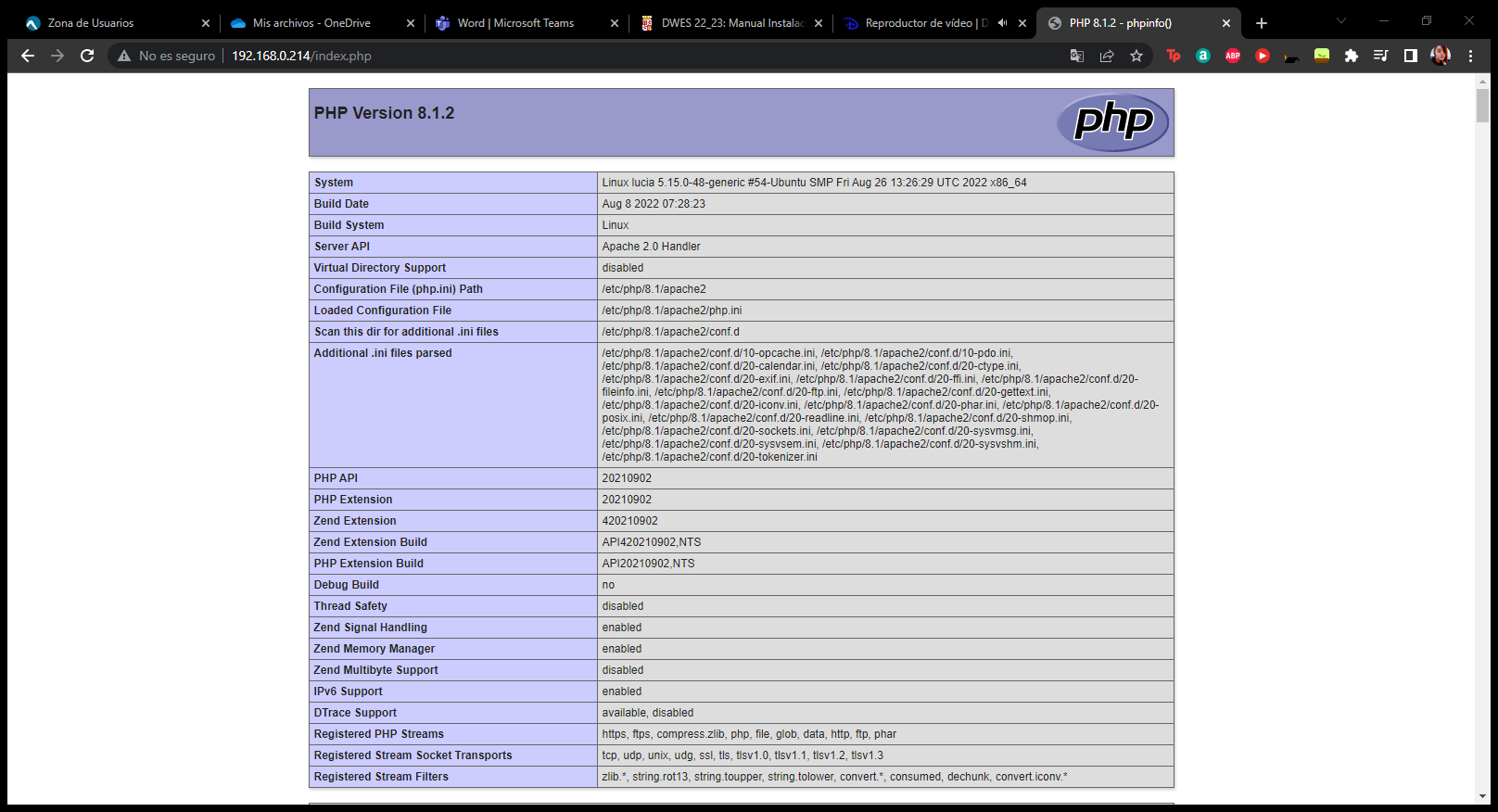
1. Comprobamos que la instalación es correcta creando un archivo **index.php** para ello usamos el comando **sudo nano index.php** *(creamos el archivo en la ruta var/www/html)*



1. Se nos abrirá el nuevo archivo y escribiremos el siguiente código:
   * + **<?php phpinfo (); ¿>**

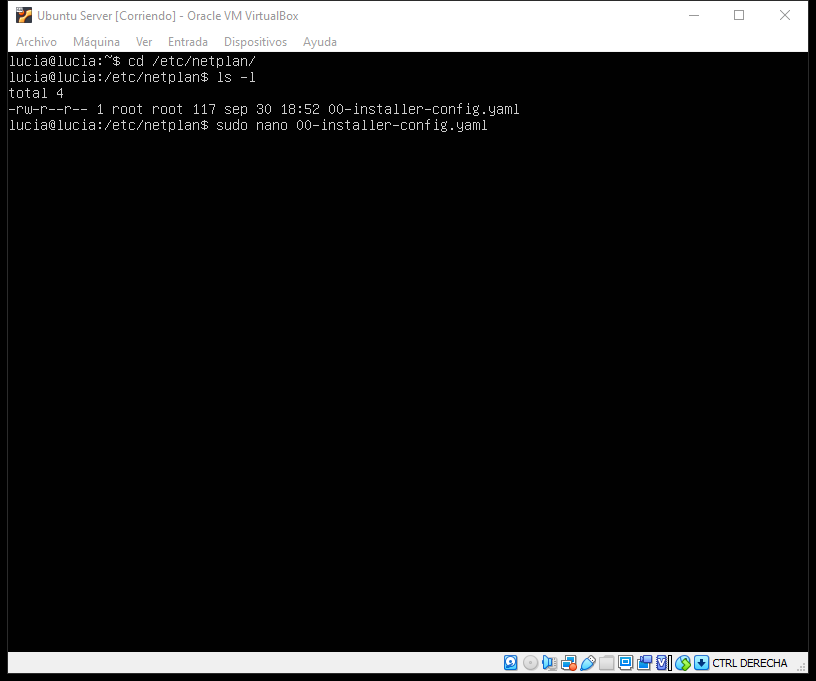


1. Para comprobar que funciona introducimos en nuestro navegador esta ruta:
   * + *http://192.168.0.214/index.php*

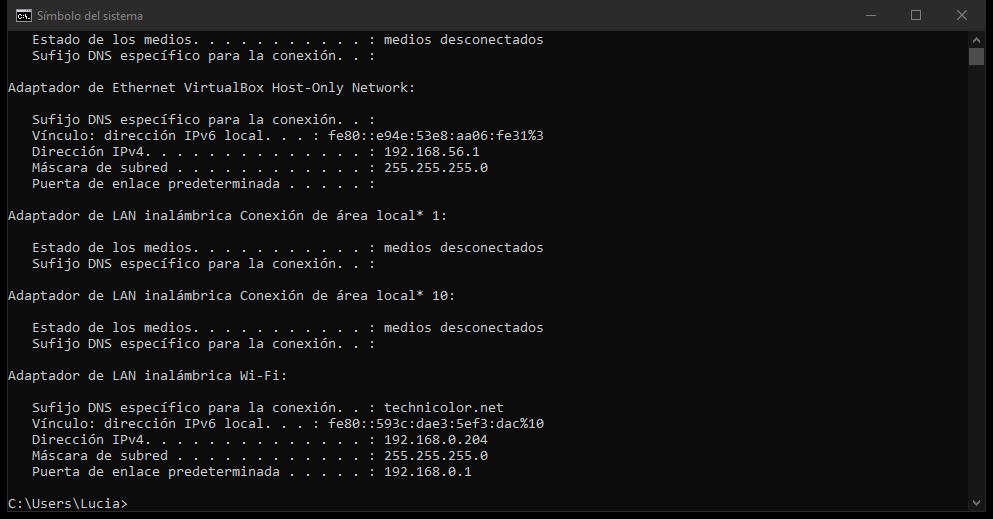


## Configuración de IP fija:

1. 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**



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



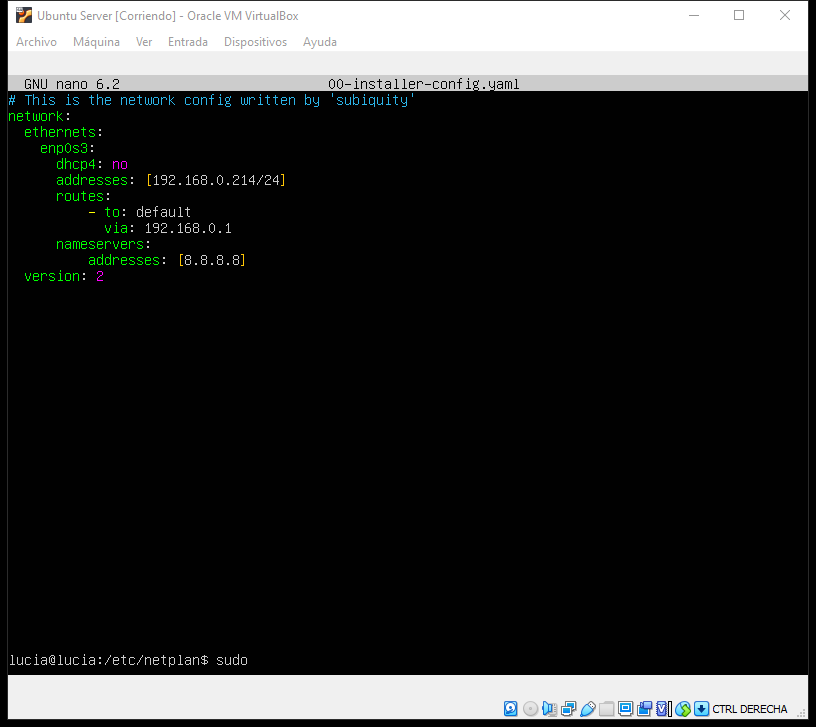
1. Teniendo ya toda la información del anterior paso, volvemos a la maquina y en el archivo **00-installer-config.yaml** introducimos los siguientes datos:
   1. dhcp4: **no**
   2. addresses: **[192.168.0.214/24]**
   3. routes:

- to: **default**

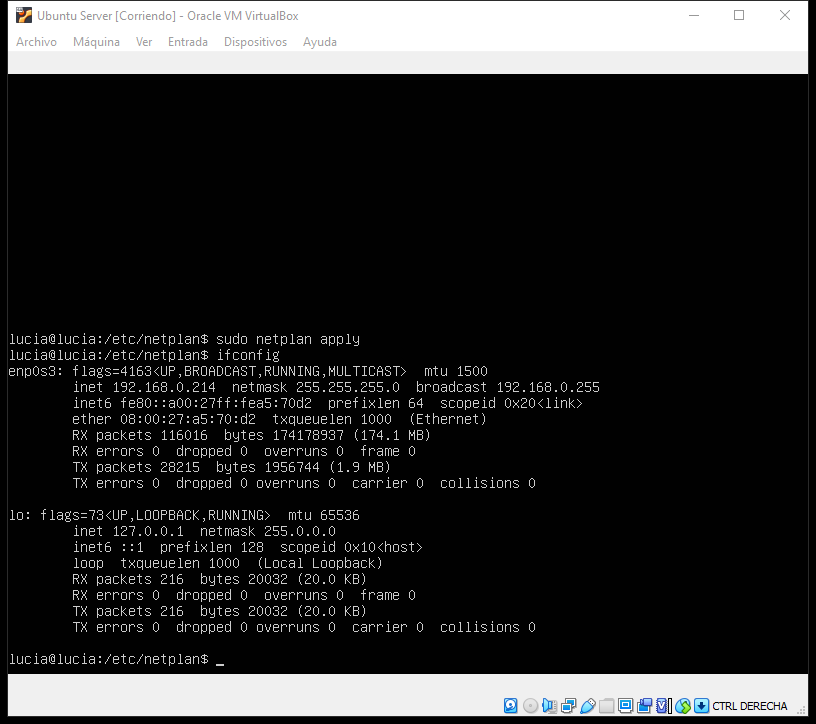
via: **192.168.0.1** (nuestra puerta de enlace predeterminada)

* 1. Nameservers:

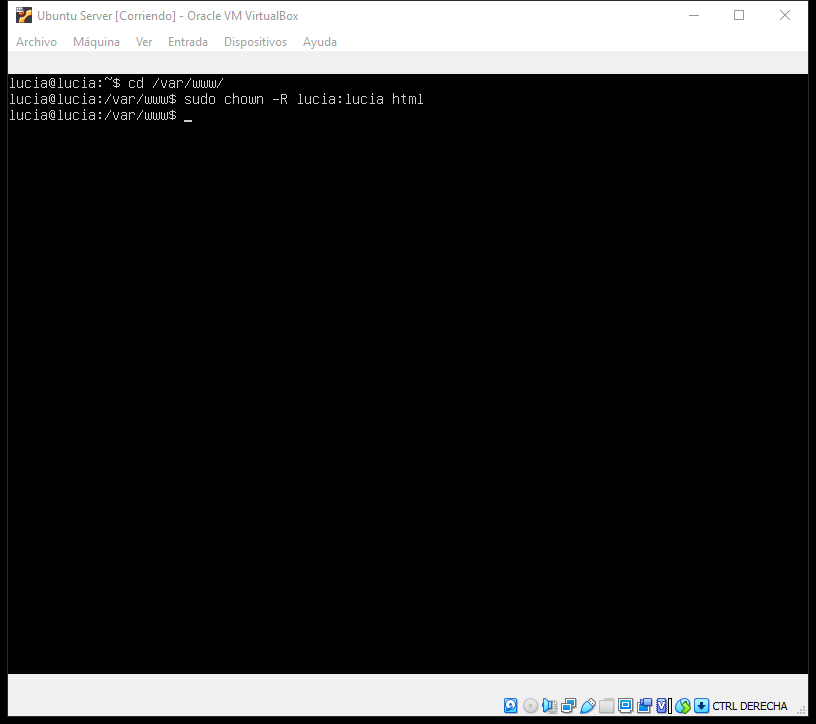
addresses: **[8.8.8.8]** (dns de Google)



1. 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.



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



## Conexión servidor-VSCode (SSH)

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

|  |  |
| --- | --- |
| Code - Visual Studio Code | **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. |

1. En la parte inferior izquierda pulsaremos en este icono:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. 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**

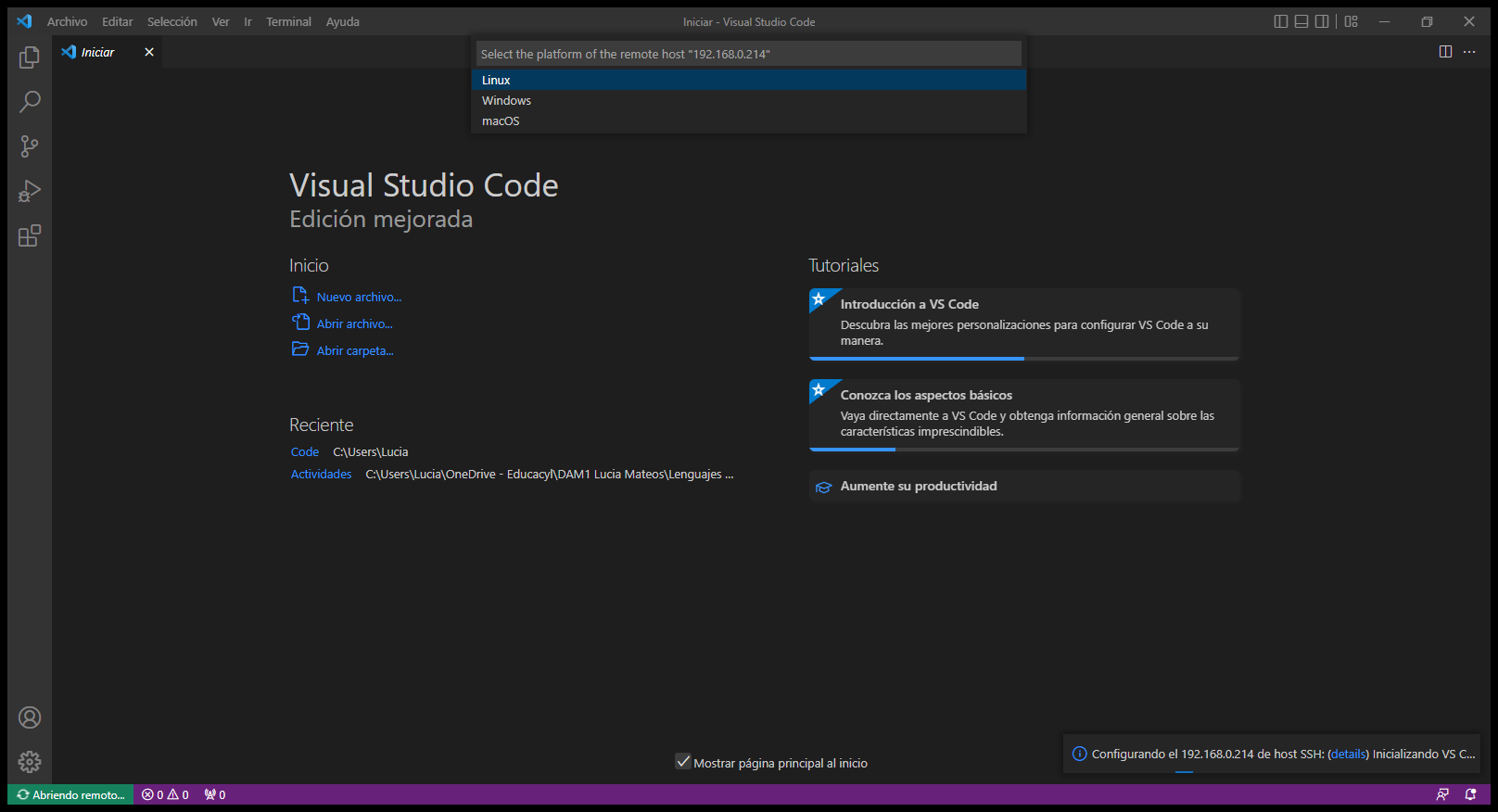
**HostName 192.168.0.214**

**User lucia**

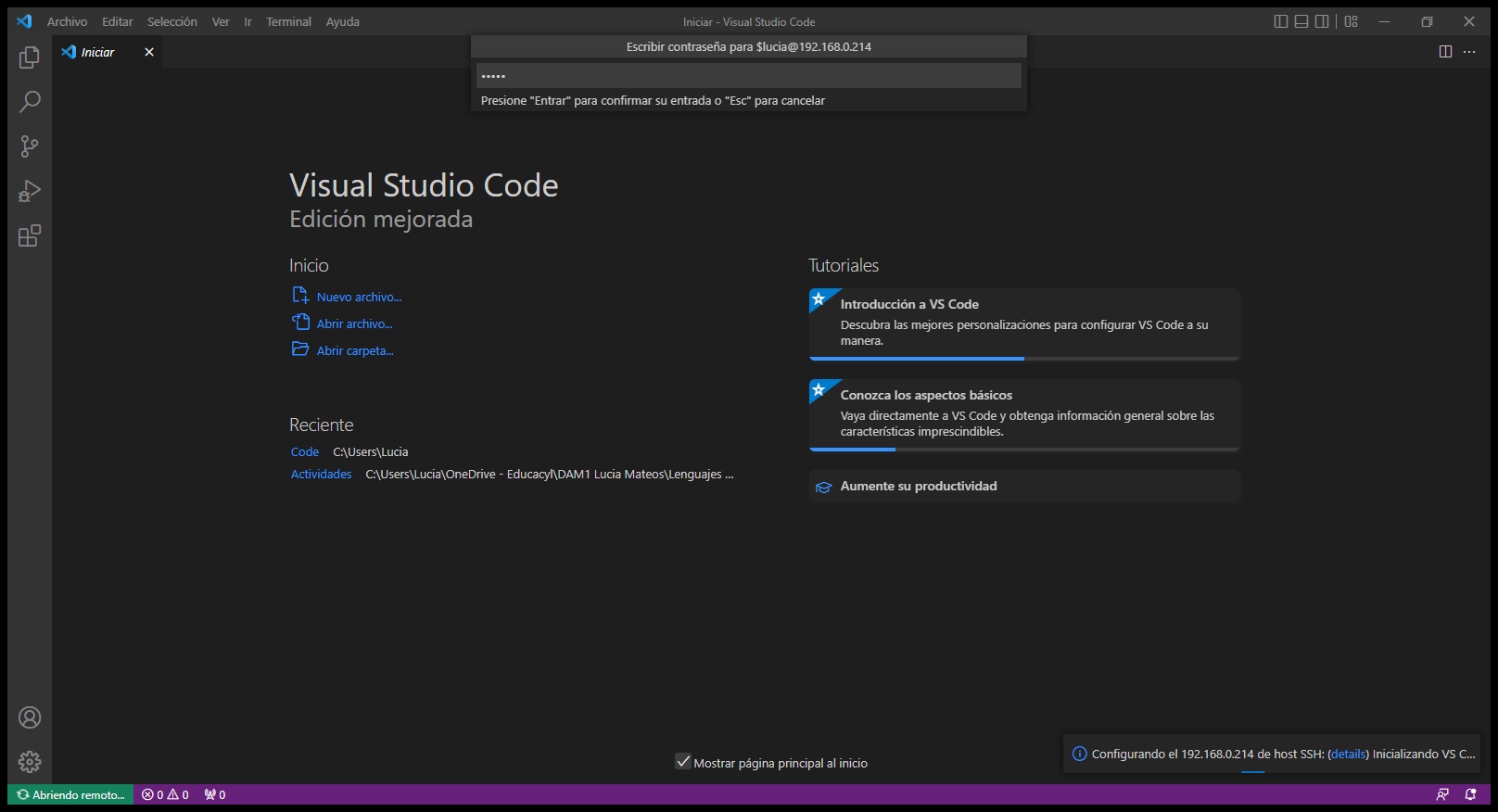
**Protocol 2**

**Port 22**

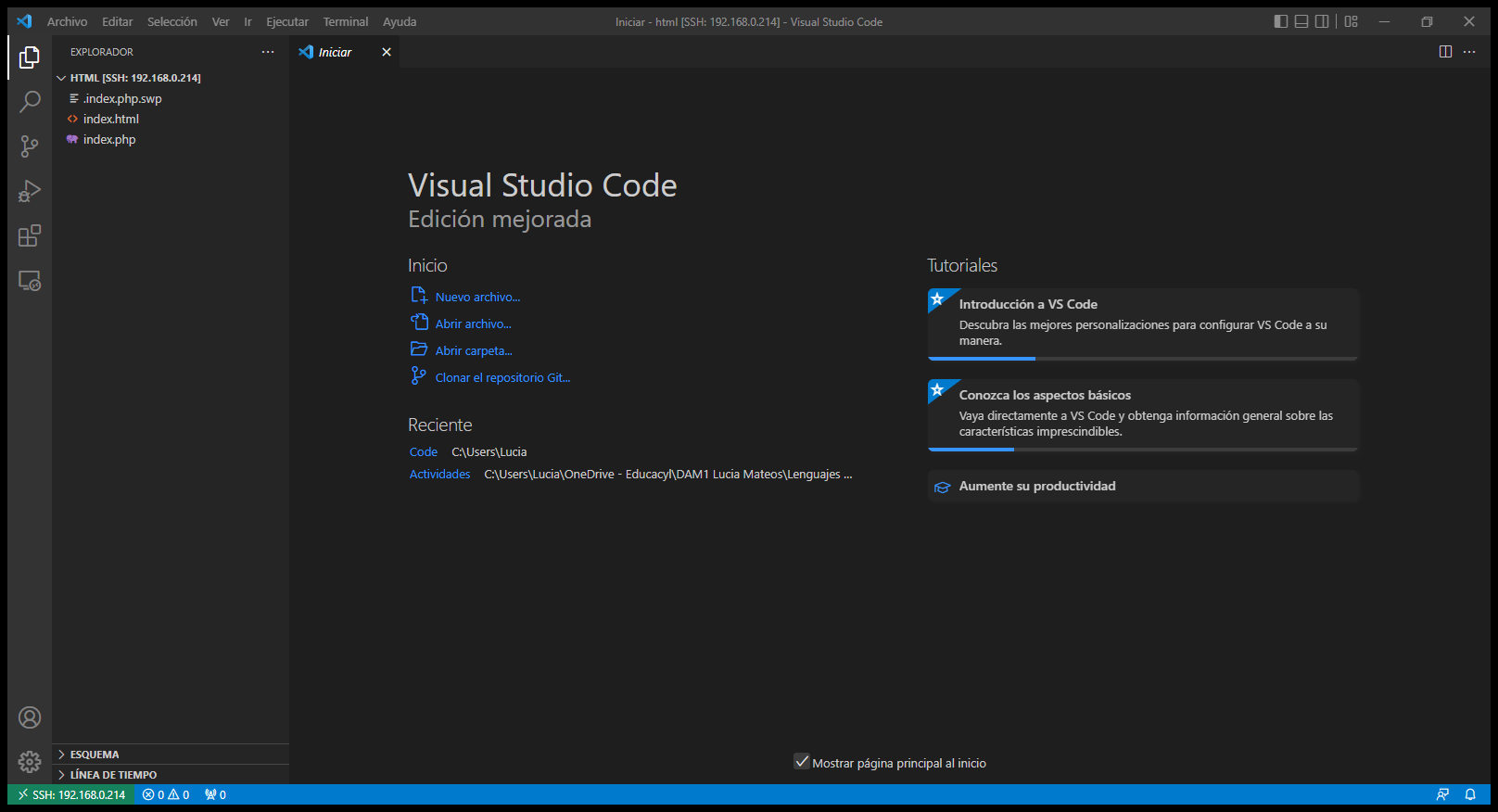
* + - Se abrirá una ventana nueva de VSCode que nos pedirá seleccionar la plataforma del host remoto **(Linux)**



* + - Confirmaremos y nos pedirá contraseña para aceptar la conectarnos

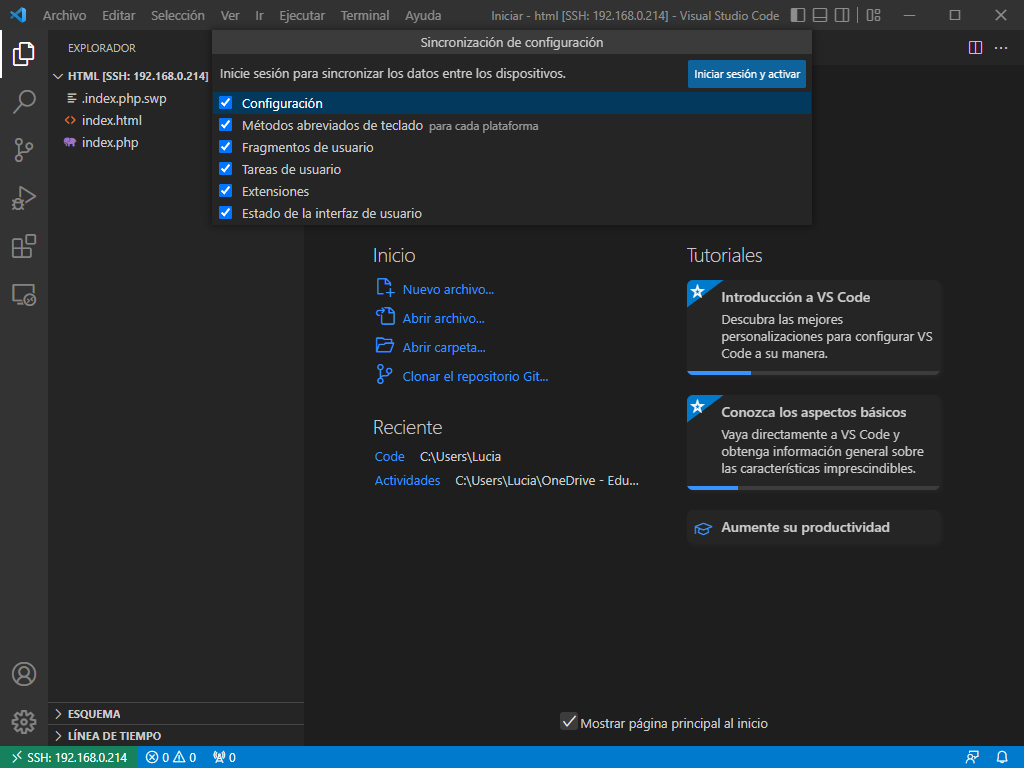


* + - Ya conectados nos aparecerá en la parte inferior izquierda la IP, el siguiente paso será abrir la carpeta de **/var/www/html/**, nos volverá a pedir la contraseña y nos mostrará todo lo que tenemos en esa carpeta

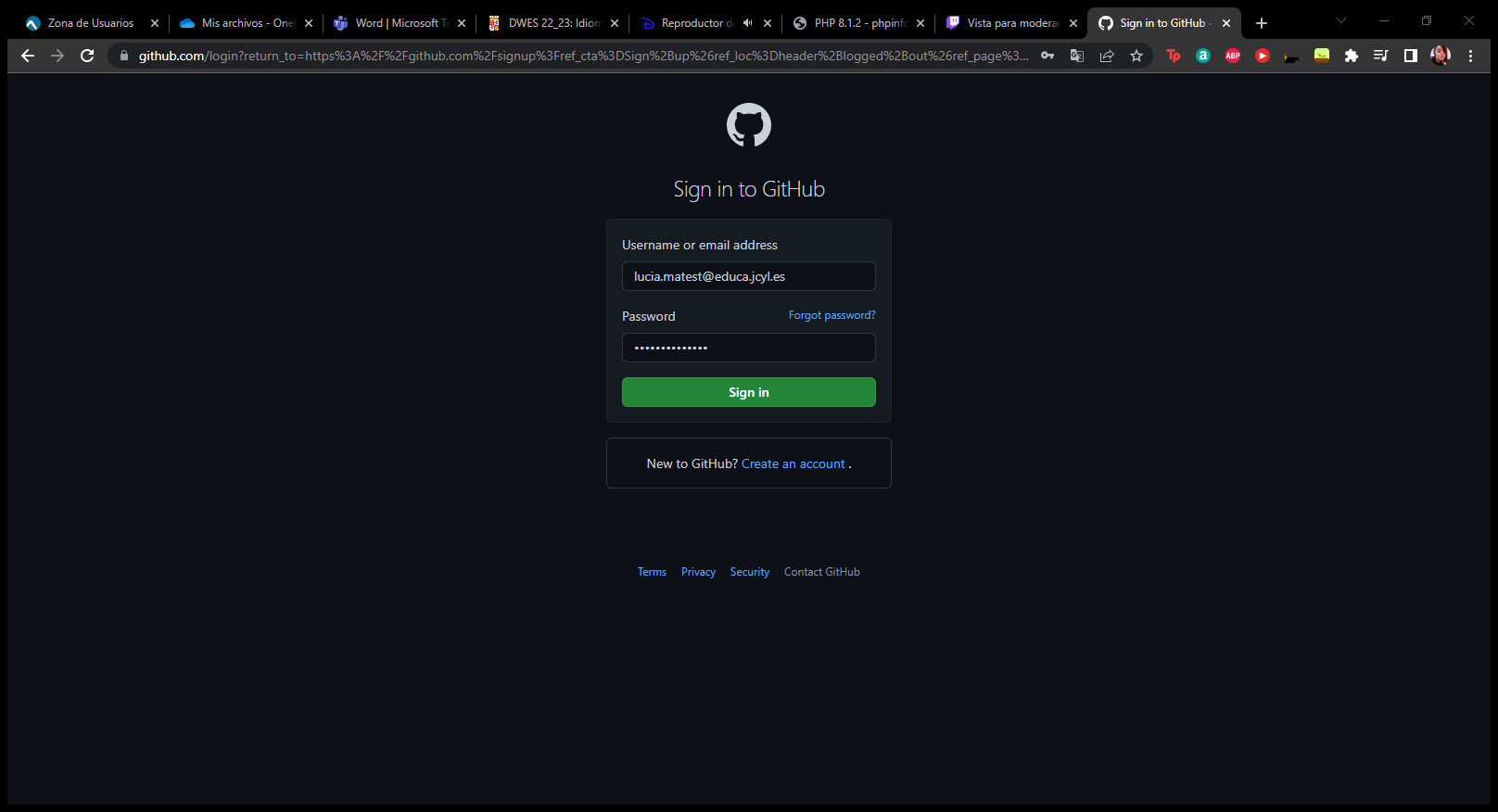


## Vincular GitHub con el servidor:

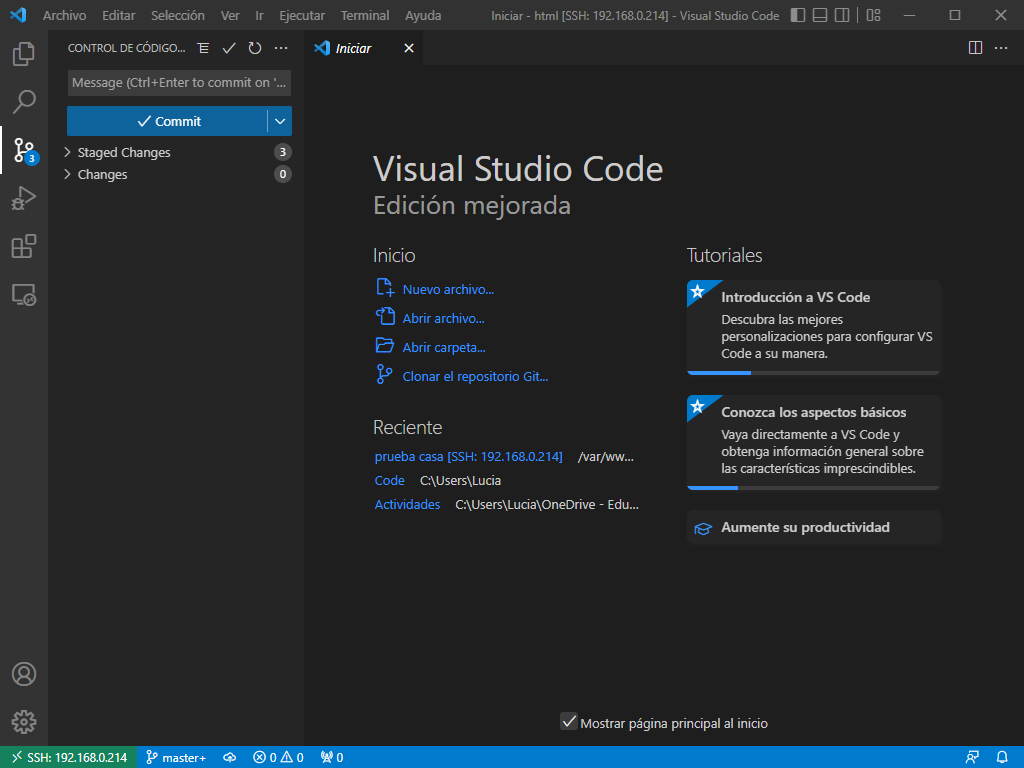
1. En VS Code vamos a **cuentas** y pulsamos el botón **iniciar sesión y activar**



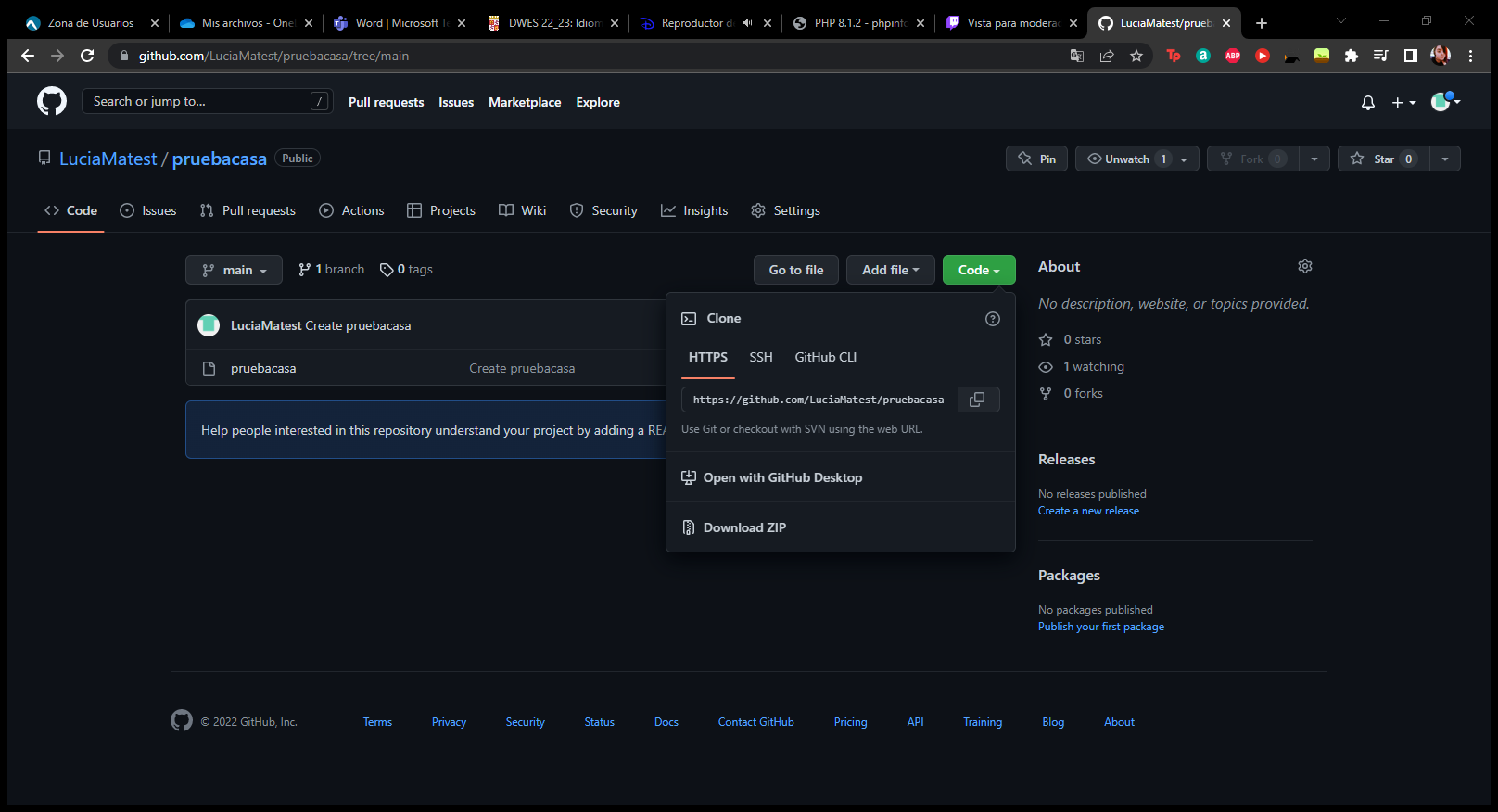
1. Desde la página **https://github.com/** creamos una cuenta *(esto nos servirá para poder sincronizar el trabajo que vamos haciendo).*



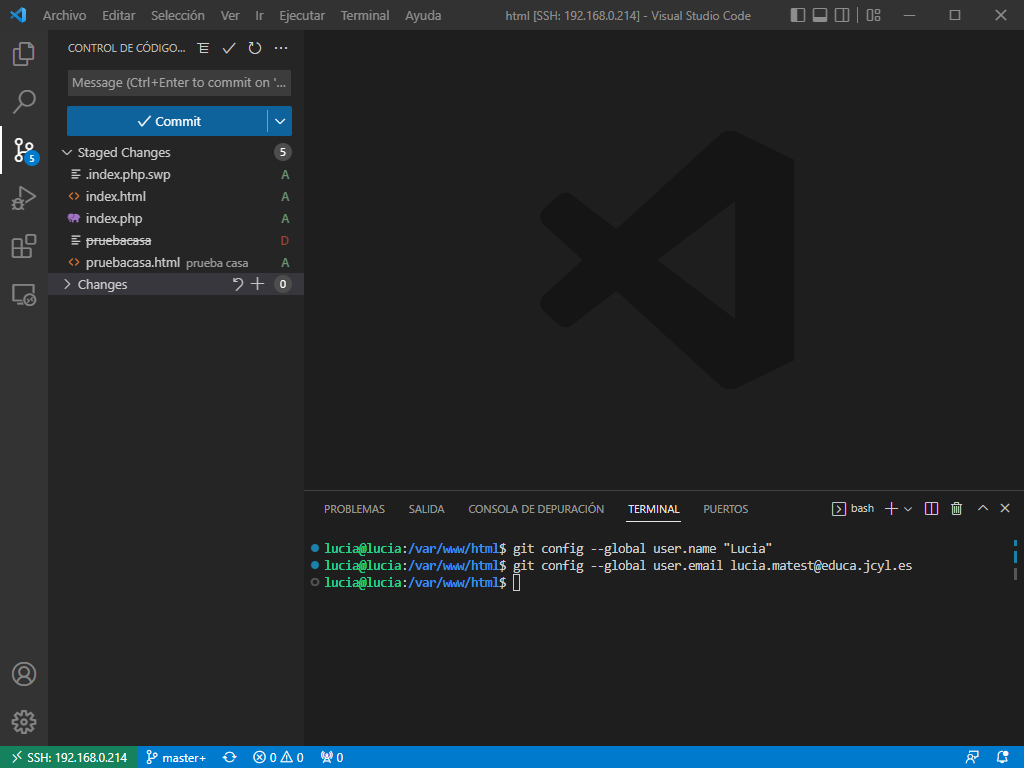
1. Para poder sincronizarnos iremos al apartado **Control de código,** vamos al menú () y seleccionamos la opción **Remote – Add remote.**



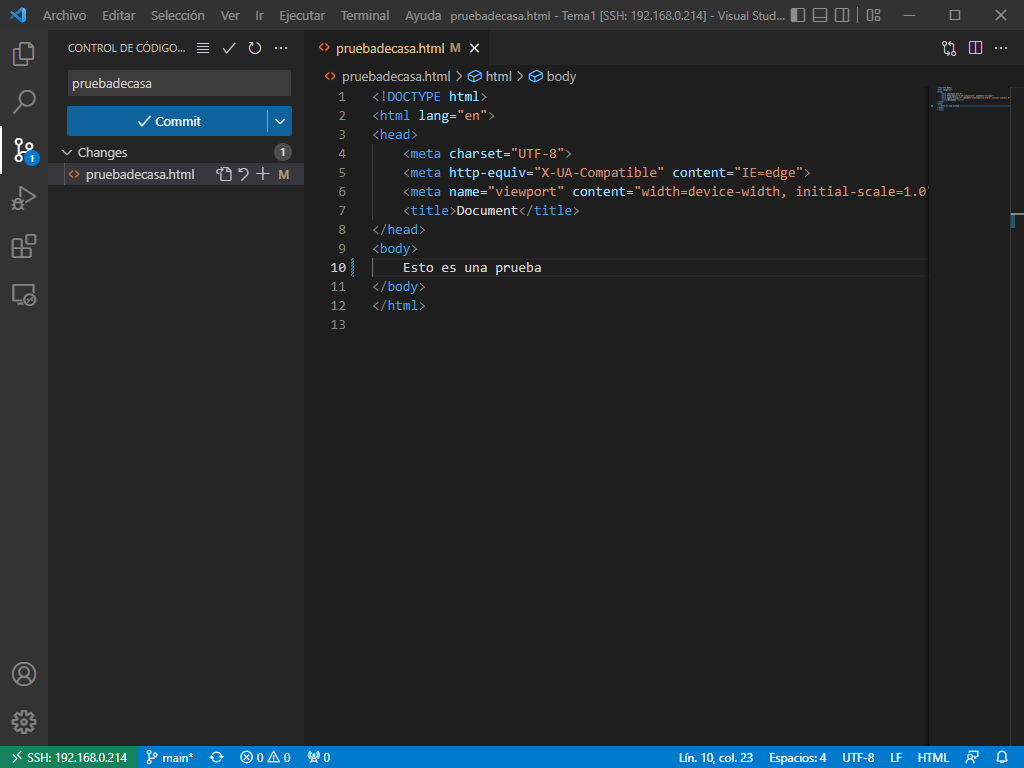
1. Debemos introducir la **url** de nuestro repositorio que encontraremos en **GitHub**:



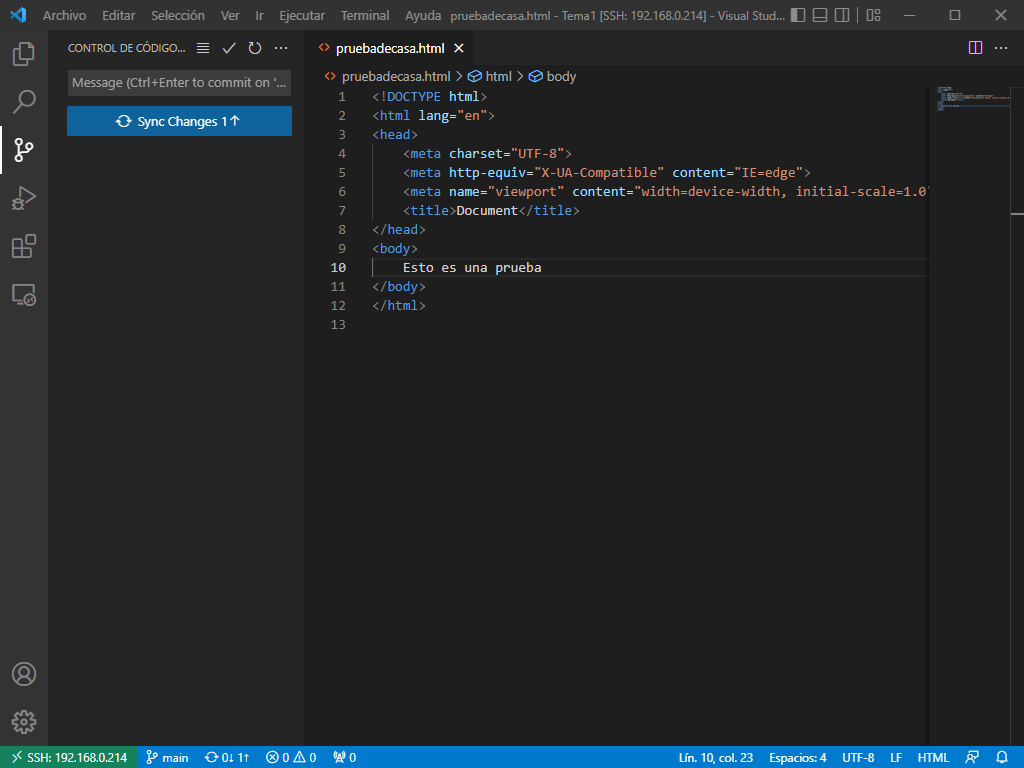
1. Desde el **Terminal** de **VSCode** configuraremos *usuario* y *email* de github ejecutando los siguientes comandos:



1. 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**



1. Al hacer esto, cambiará el boton a **Sync Changes** y podemos comprobar el cambio desde GitHub



1. Si hacemos cambios al mismo tiempo desde VSCode y Github en la misma linea, al sincronizarlos de nuevo, nos aparecera este aviso:

