

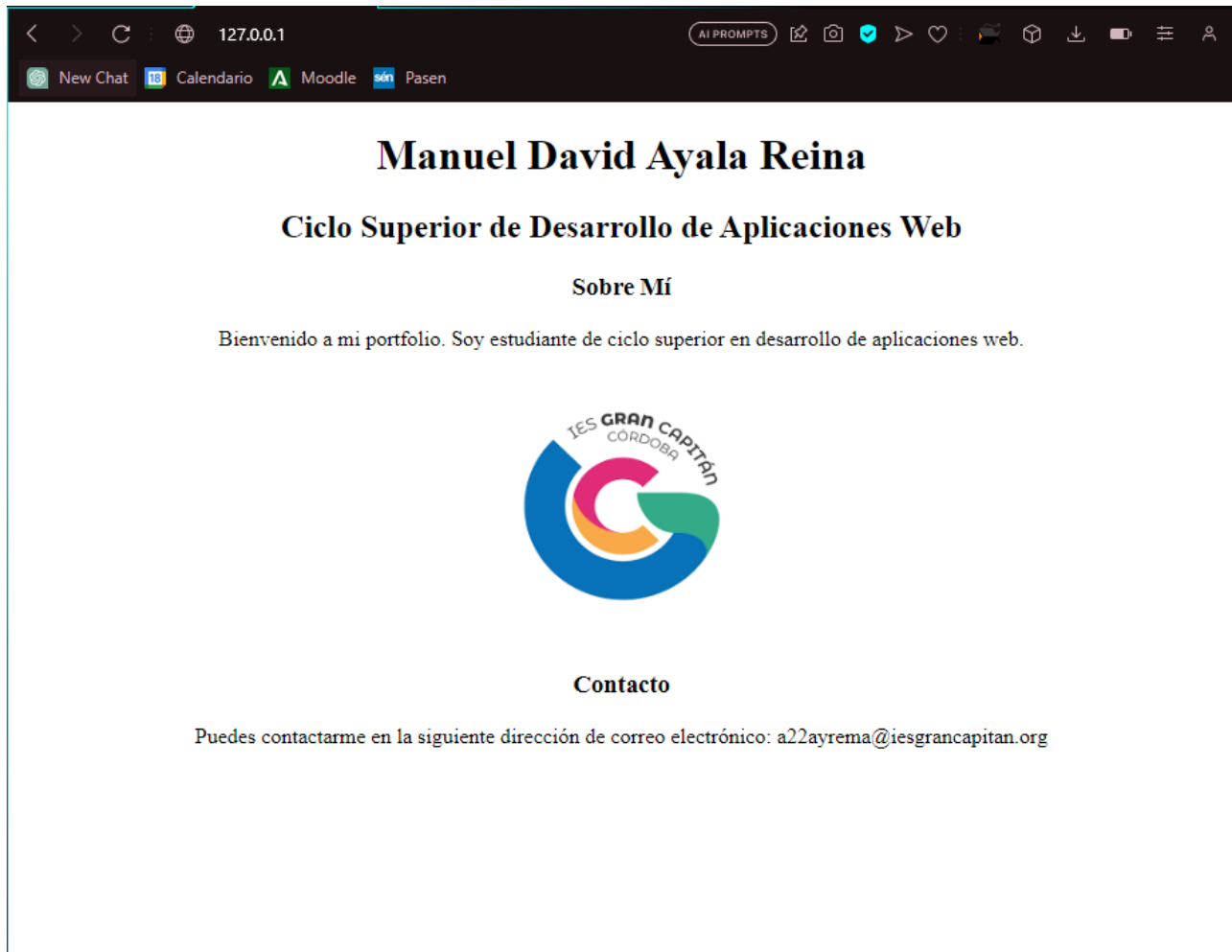
RA1_IT1_AE1

Indice

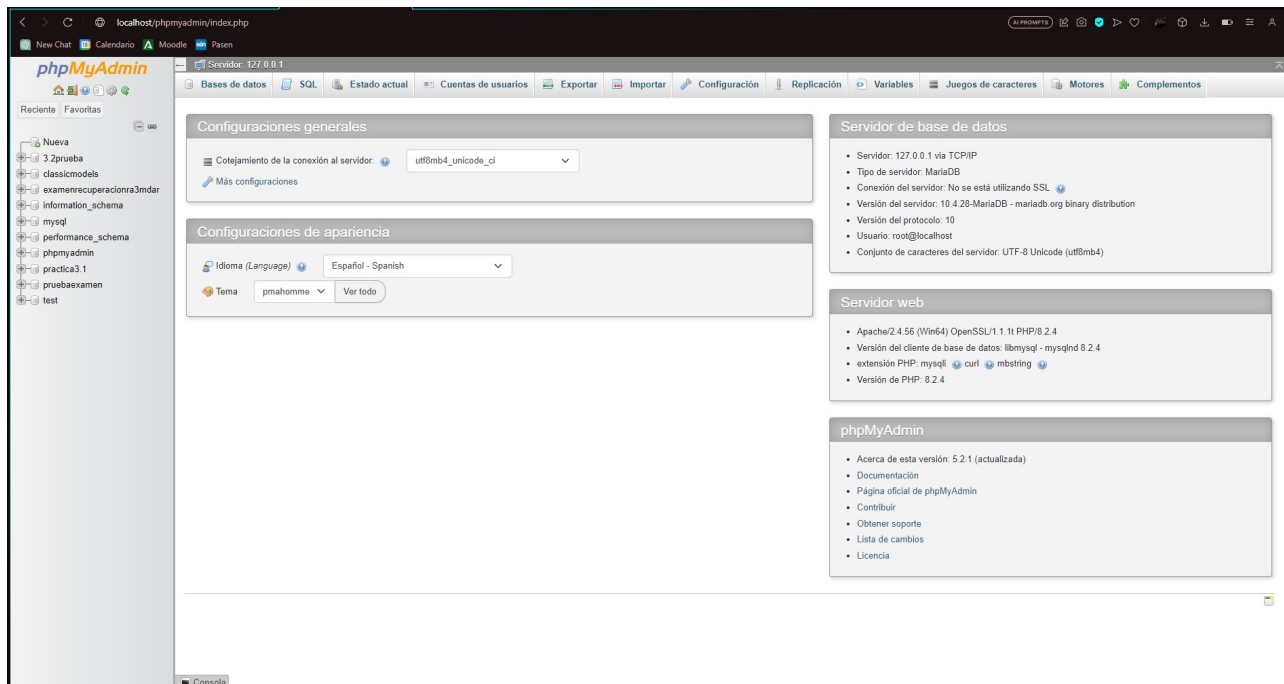
Actividad 1.....	2
Actividad 2.....	3
Actividad 3.....	14
Actividad 4.....	20

Actividad 1.

Como ya tenia instalado XAMPP, voy a mostrar el portfolio creado:



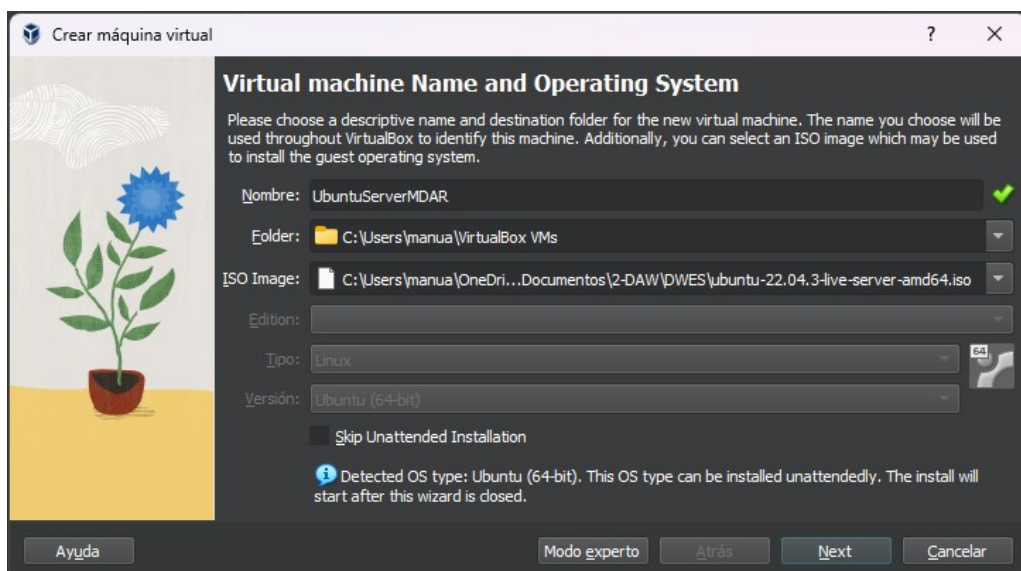
Ahora voy a mostrar phpmyadmin funcionando perfectamente:

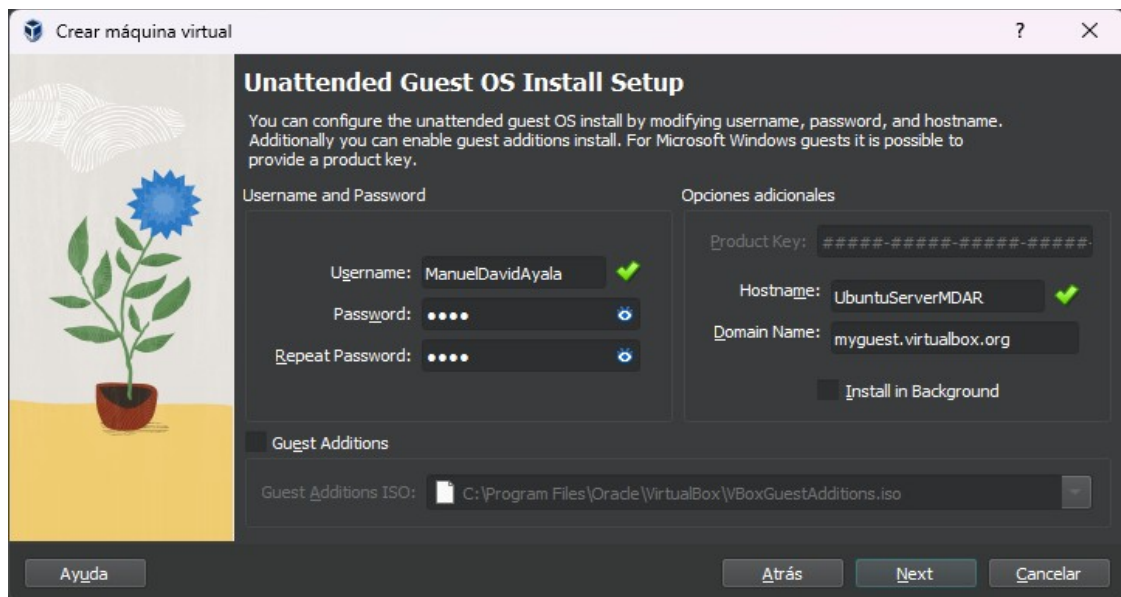


Actividad 2.

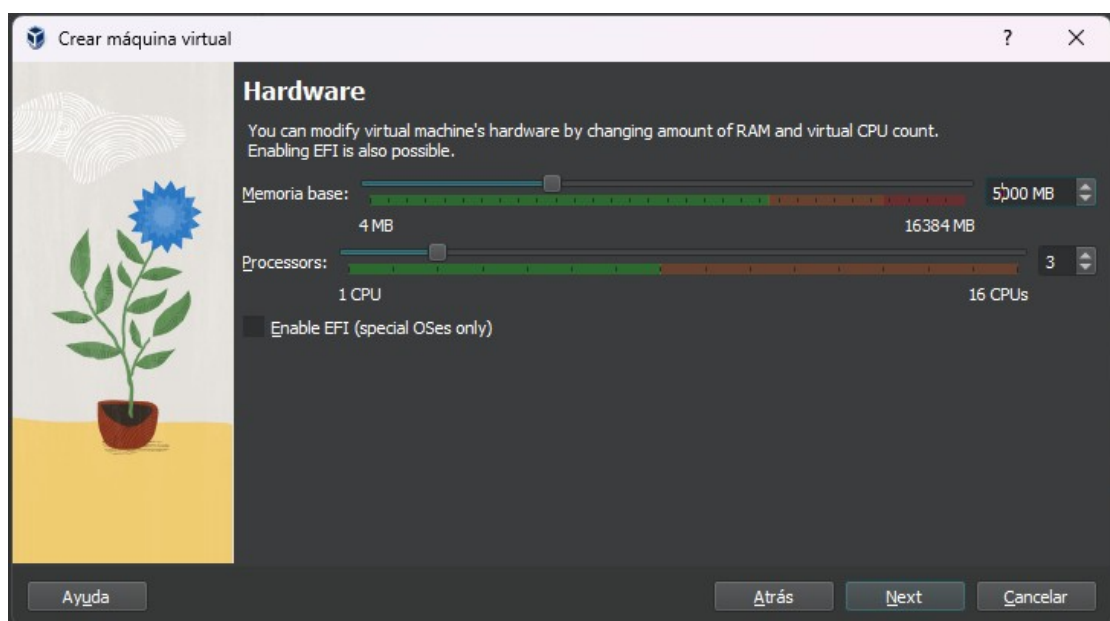
1. Crear una máquina virtual con Ubuntu Server:

- Abre VirtualBox y haz clic en "Nueva" para crear una nueva máquina virtual.
- Asigna un nombre a la máquina virtual y selecciona el tipo "Linux" y la versión "Ubuntu (64-bit)".

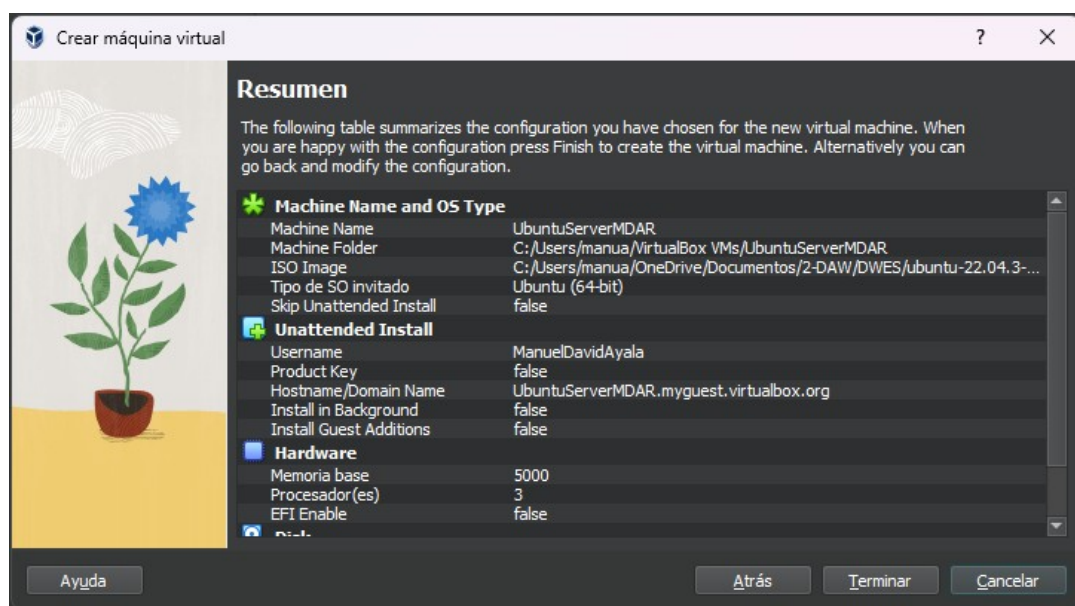
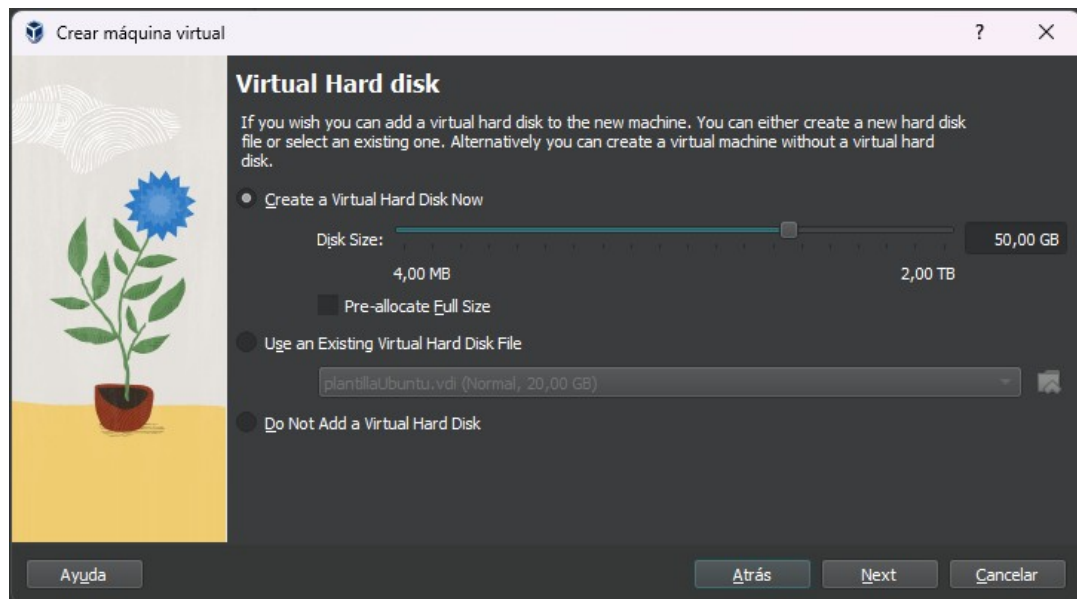




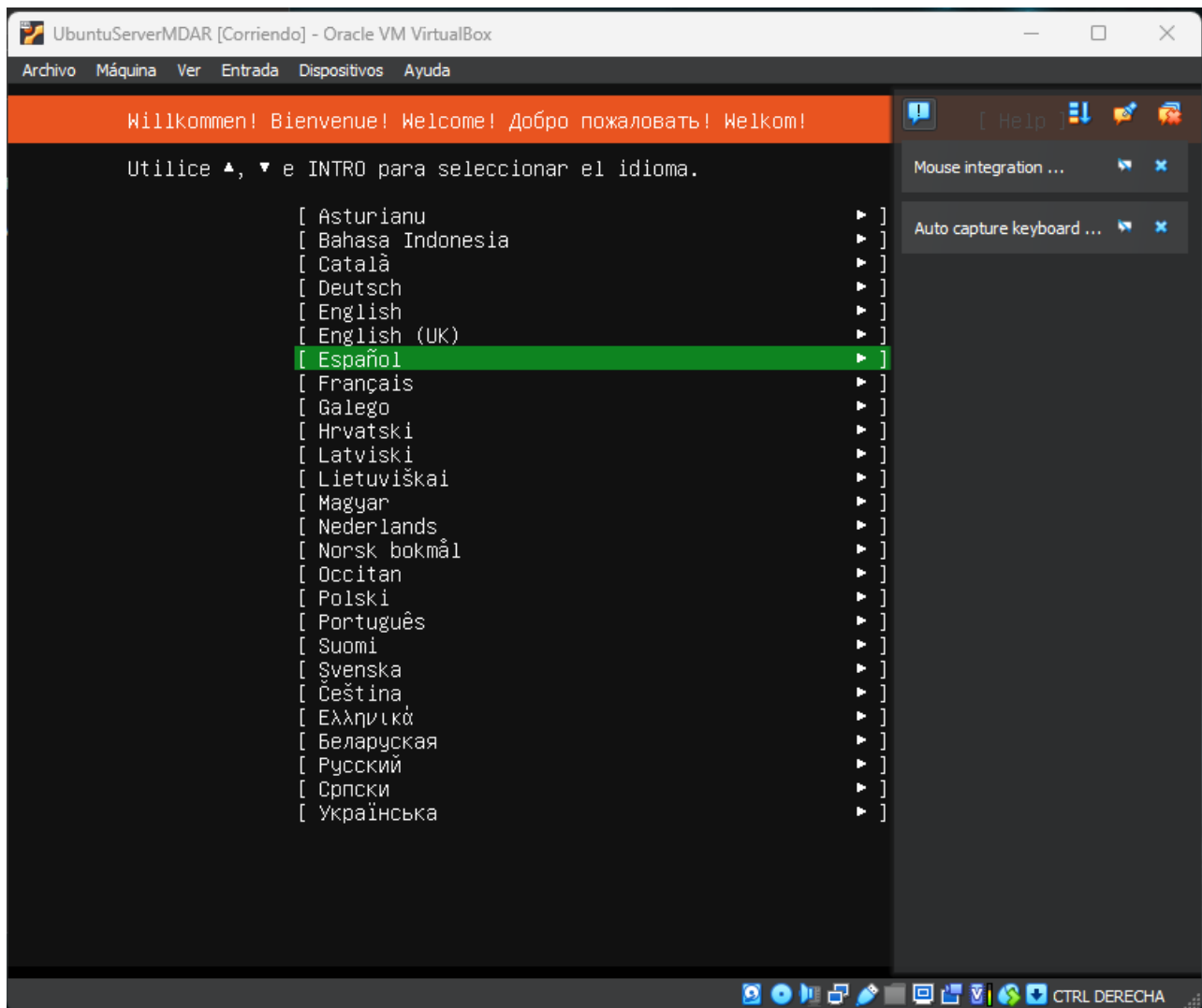
- Asigna la cantidad de memoria RAM que desees para la máquina virtual.



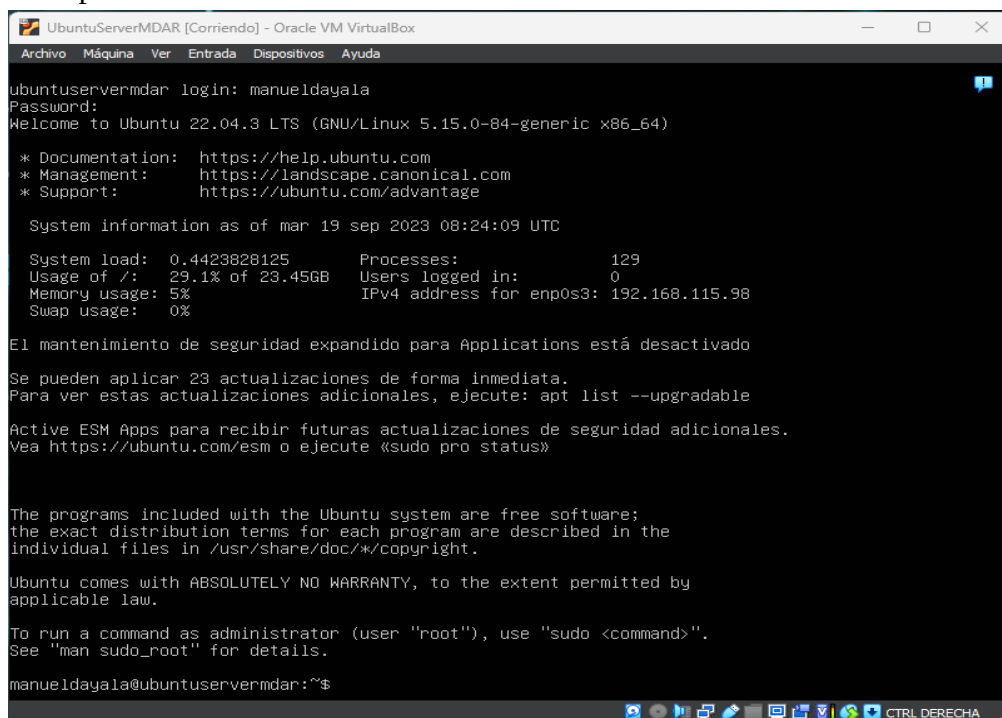
- Crea un nuevo disco duro virtual y asigna el espacio necesario.



- Inicia la máquina virtual.



En esta captura se ve como funciona el servidor correctamente.

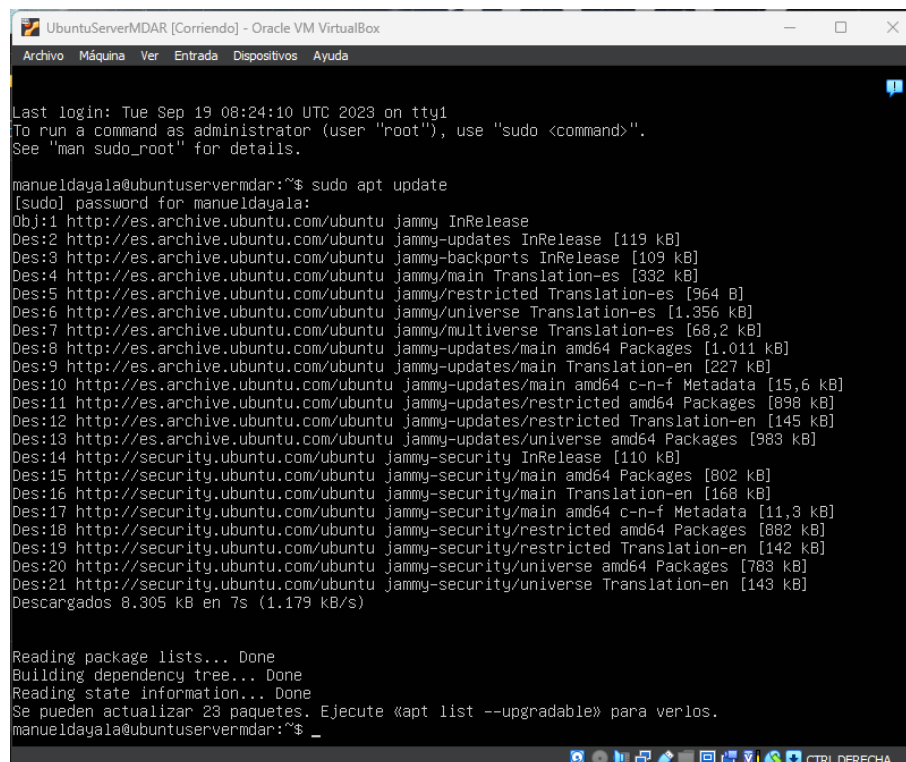


2.Instalacion

Paso 1: Instalación de Apache

2. Actualizar la lista de paquetes e instalar Apache con los siguientes comandos:

`sudo apt update`



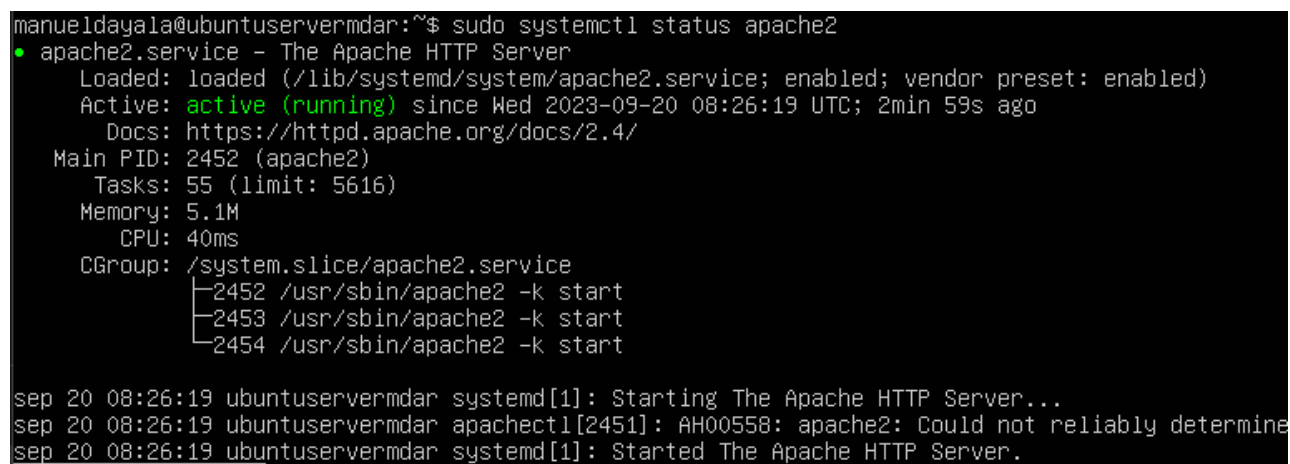
```
UbuntuServerMDAR [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

Last login: Tue Sep 19 08:24:10 UTC 2023 on tty1
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

manueldayala@ubuntuservermdar:~$ sudo apt update
[sudo] password for manueldayala:
Obj:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Des:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy/main Translation-es [332 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu jammy/restricted Translation-es [964 B]
Des:6 http://es.archive.ubuntu.com/ubuntu jammy/universe Translation-es [1.356 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu jammy/multiverse Translation-es [68,2 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1.011 kB]
Des:9 http://es.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [227 kB]
Des:10 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [15,6 kB]
Des:11 http://es.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [898 kB]
Des:12 http://es.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [145 kB]
Des:13 http://es.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [983 kB]
Des:14 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Des:15 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [802 kB]
Des:16 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [168 kB]
Des:17 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [11,3 kB]
Des:18 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [882 kB]
Des:19 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [142 kB]
Des:20 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [783 kB]
Des:21 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [143 kB]
Descargados 8.305 kB en 7s (1.179 kB/s)

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Se pueden actualizar 23 paquetes. Ejecute «apt list --upgradable» para verlos.
manueldayala@ubuntuservermdar:~$
```

`sudo apt install apache2`



```
manueldayala@ubuntuservermdar:~$ sudo systemctl status apache2
• apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-09-20 08:26:19 UTC; 2min 59s ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 2452 (apache2)
      Tasks: 55 (limit: 5616)
     Memory: 5.1M
        CPU: 40ms
    CGroup: /system.slice/apache2.service
            └─2452 /usr/sbin/apache2 -k start
              └─2453 /usr/sbin/apache2 -k start
                └─2454 /usr/sbin/apache2 -k start

sep 20 08:26:19 ubuntuservermdar systemd[1]: Starting The Apache HTTP Server...
sep 20 08:26:19 ubuntuservermdar apachectl[2451]: AH00558: apache2: Could not reliably determine
sep 20 08:26:19 ubuntuservermdar systemd[1]: Started The Apache HTTP Server.
```


3. Habilitar Apache para que se inicie automáticamente al arrancar:

`sudo systemctl enable apache2`

```
manueidayala@ubuntuservermdar:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
```

4. Inicia el servicio de Apache:

`sudo systemctl start apache2`

```
manueidayala@ubuntuservermdar:~$ sudo systemctl start apache2
```

5. Verificar que Apache esté en funcionamiento visitando la dirección IP de tu servidor en un navegador web. Debería verse la página de inicio de Apache.



Paso 2: Instalación de MariaDB (MySQL)

6. Instalar el servidor de bases de datos MariaDB y seguir las instrucciones para configurar una contraseña de root:

```
sudo apt install mariadb-server
```

```
manueldayala@ubuntu-servermdar:~$ sudo apt install mariadb-server
[sudo] password for manueldayala:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Se instalarán los siguientes paquetes adicionales:
  galera-4 libcgi-fast-perl libcgi-pm-perl libclone-perl libconfig-inifiles-perl libdaxctl1 libdbd-mysql-perl
  libdbi-perl libencode-locale-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmariadb3
  libmysqlclient21 libndctl6 libpmem1 libsnappy1v5 libtimedate-perl liburi-perl liburing2 mariadb-client-10.6
  mariadb-client-core-10.6 mariadb-common mariadb-server-10.6 mariadb-server-core-10.6 mysql-common socat
Paquetes sugeridos:
  libmldbm-perl libnet-daemon-perl libsql-statement-perl libdata-dump-perl libipc-sharedcache-perl
  libbusiness-isbn-perl libwww-perl mailx mariadb-test
Se instalarán los siguientes paquetes NUEVOS:
  galera-4 libcgi-fast-perl libcgi-pm-perl libclone-perl libconfig-inifiles-perl libdaxctl1 libdbd-mysql-perl
  libdbi-perl libencode-locale-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmariadb3
  libmysqlclient21 libndctl6 libpmem1 libsnappy1v5 libtimedate-perl liburi-perl liburing2 mariadb-client-10.6
  mariadb-client-core-10.6 mariadb-common mariadb-server mariadb-server-10.6 mariadb-server-core-10.6 mysql-common
  socat
0 actualizados, 35 nuevos se instalarán, 0 para eliminar y 23 no actualizados.
Se necesita descargar 18,5 MB de archivos.
Se utilizarán 164 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] |
```

Paso 3: Instalación de PHP

8. Instalar PHP y algunas extensiones necesarias:

```
sudo apt install php libapache2-mod-php php-mysql
```

```
manueldayala@ubuntu-server X + -
Remove test database and access to it? [Y/n] n
... skipping.

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] n
... skipping.

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
manueldayala@ubuntu-servermdar:~$ sudo apt install php libapache2-mod-php php-mysql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Se instalarán los siguientes paquetes adicionales:
  libapache2-mod-php8.1 php-common php8.1 php8.1-cli php8.1-common php8.1-mysql php8.1-opcache php8.1-readline
Paquetes sugeridos:
  php-pear
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php libapache2-mod-php8.1 php php-common php-mysql php8.1 php8.1-cli php8.1-common php8.1-mysql
  php8.1-opcache php8.1-readline
0 actualizados, 11 nuevos se instalarán, 0 para eliminar y 23 no actualizados.
Se necesita descargar 5.265 kB de archivos.
Se utilizarán 21,8 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

9. Reiniciar Apache para que se apliquen los cambios:

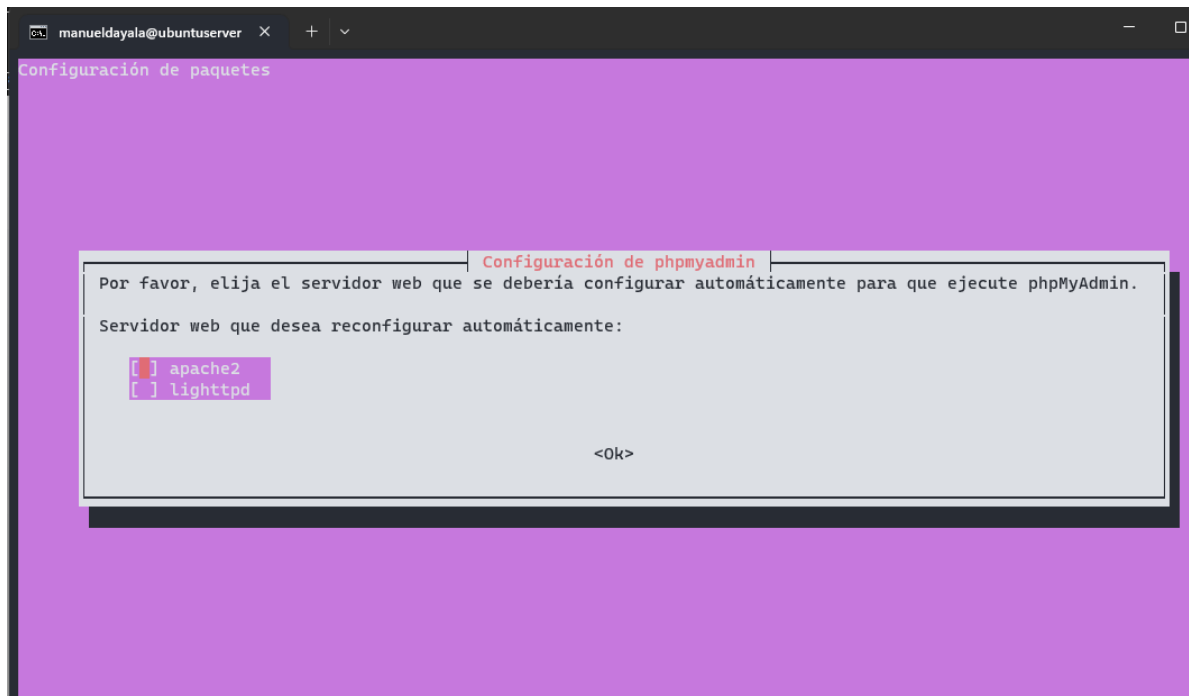
```
sudo systemctl restart apache2
```

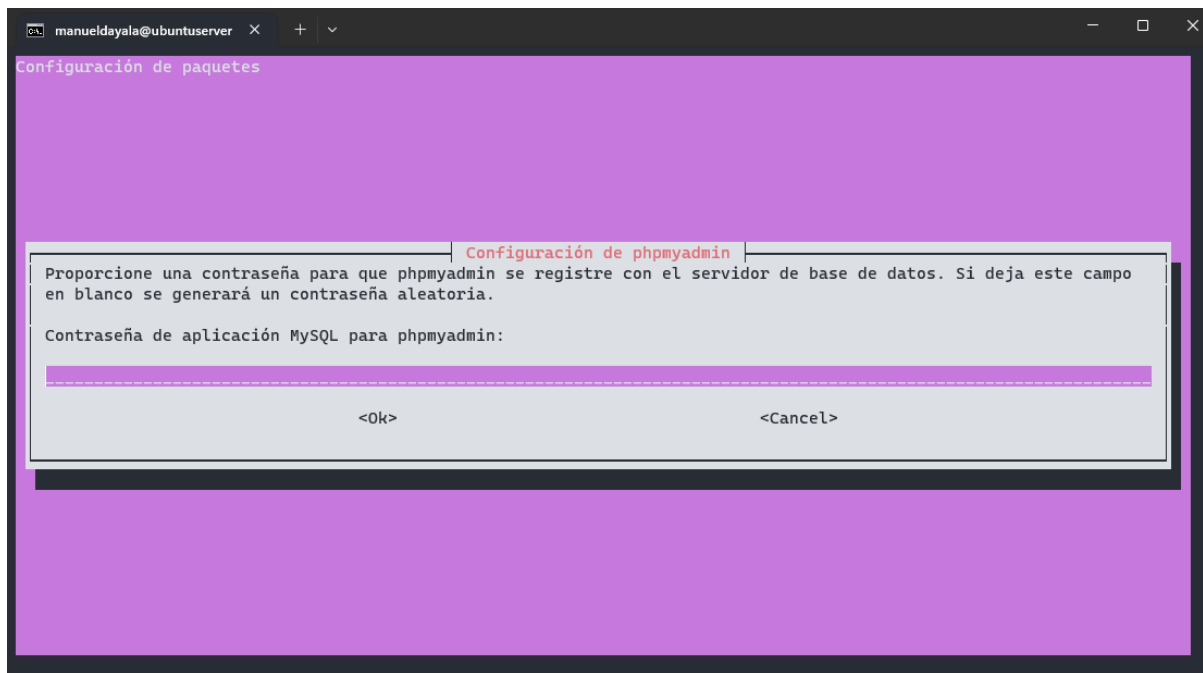
```
manueldayala@ubuntuservermdar:~$ sudo systemctl restart apache2
```

Paso 4: Instalación de phpMyAdmin

10. Instala phpMyAdmin con el siguiente comando:

```
sudo apt install phpmyadmin
```





11. Configura phpMyAdmin para que funcione con Apache:

`sudo phpenmod mbstring`

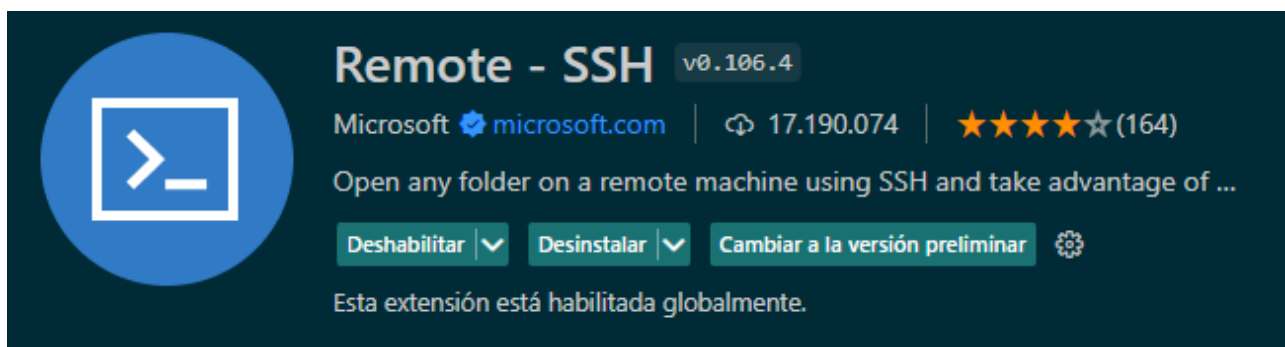
```
manueldayala@ubuntu-servermdar:~$ sudo phpenmod mbstring
```

`sudo systemctl restart apache2`

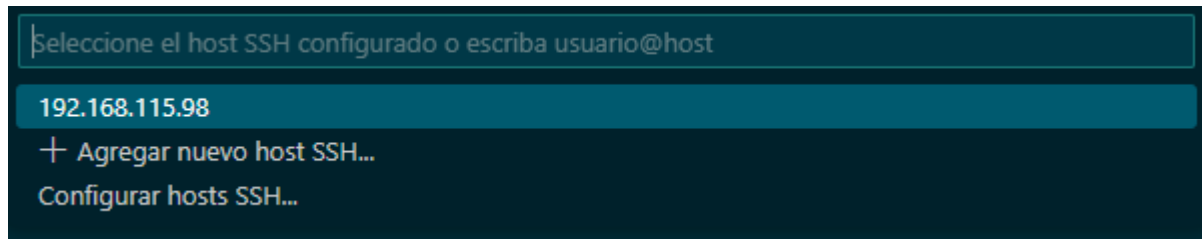
```
manueldayala@ubuntu-servermdar:~$ sudo systemctl restart apache2
```

Paso 5: Configurar la conexión entre el entorno de trabajo y el servidor.

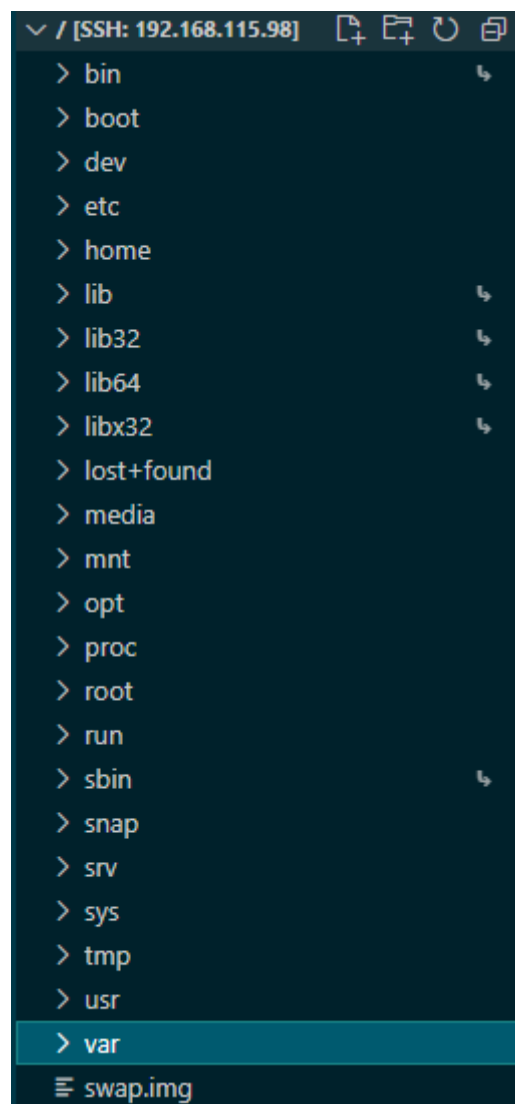
En este caso voy a utilizar VisualStudio code, para usar ssh en este entorno hay que instalar este plugin.



Para conectarlo, el servidor tiene que estar en la misma red de la maquina física o de trabajo.



Yo en mi caso ya tengo agregada la ip del servidor y es solo seleccionar la ip y poner la contraseña del mismo.



Paso 6: Probar la aplicación creando un portfolio personal básico.

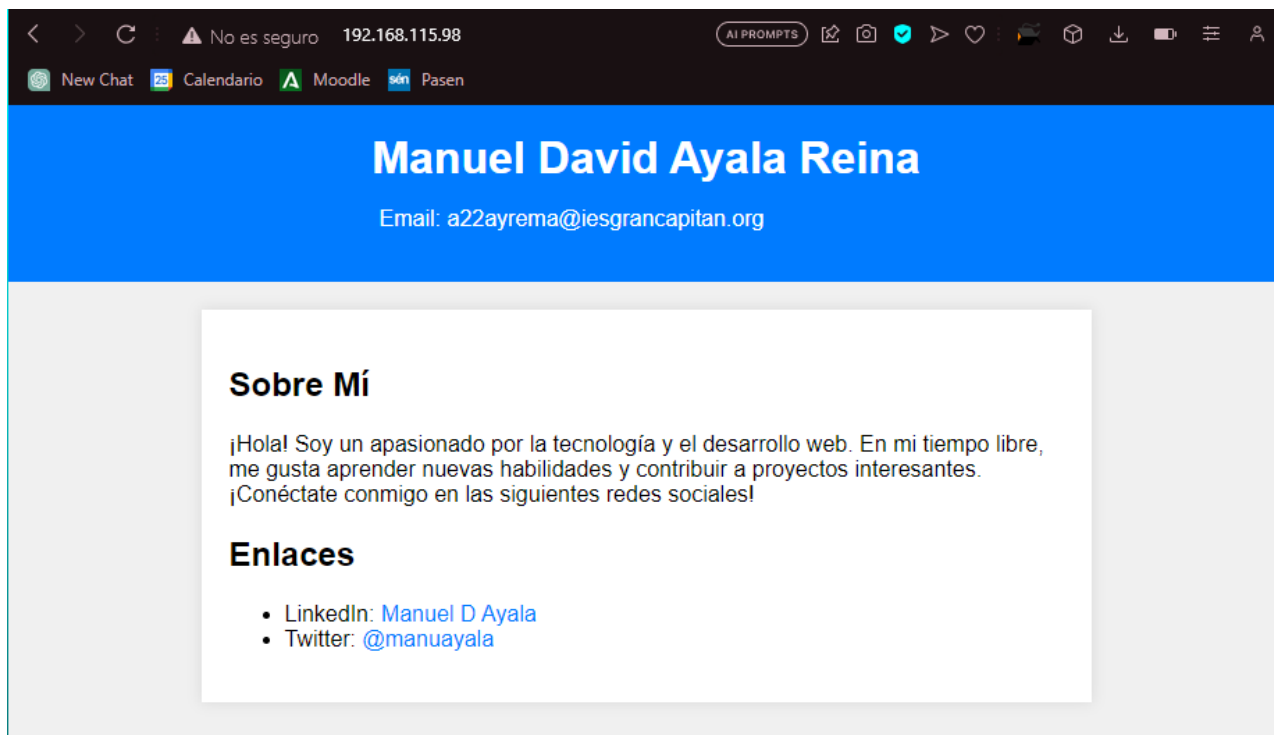
Para ello hay que modificar el archivo `/var/www/html`, para ello antes hay que dar permisos al usuario para que pueda modificarlo.

```
manueldayala@ubuntuservermdar:~$ sudo chown manueldayala -R /var/www/html
```

Ahora modificando y poniendo el código html se puede crear el portfolio.

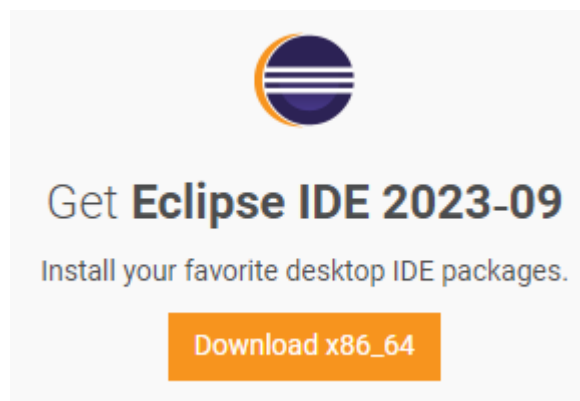
```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mi Portfolio Personal</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f0f0f0;
      margin: 0;
      padding: 0;
    }
    header {
      background-color: #007bff;
      color: #fff;
      text-align: center;
      padding: 20px;
    }
    h1 {
      margin: 0;
    }
    .container {
      max-width: 600px;
      margin: 20px auto;
      padding: 20px;
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    a {
      text-decoration: none;
      color: #007bff;
    }
  </style>
</head>
<body>
  <header>
    <h1>Manuel David Ayala Reina</h1>
    <p>Email: a22ayrema@iesgrancapitan.org <a href="mailto:tu@email.com">tu@email.com</a></p>
  </header>
  <div class="container">
    <h2>Sobre Mí</h2>
    <p>¡Hola! Soy un apasionado por la tecnología y el desarrollo web. En mi tiempo libre, me gusta aprender nuevas habilidades y contribuir a proyectos interesantes. ¡Conéctate conmigo en las siguientes redes sociales!</p>
    <h2>Enlaces</h2>
    <ul>
      <li>LinkedIn: <a href="https://www.linkedin.com" target="_blank">Manuel D Ayala</a></li>
      <li>Twitter: <a href="https://www.twitter.com" target="_blank">@manuayala</a></li>
    </ul>
  </div>
</body>
</html>
```

Con este código se crea este portfolio básico accediendo desde la ip del servidor.

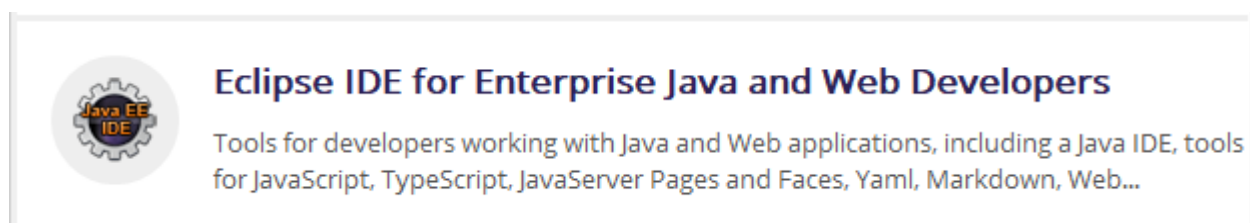


Actividad 3.

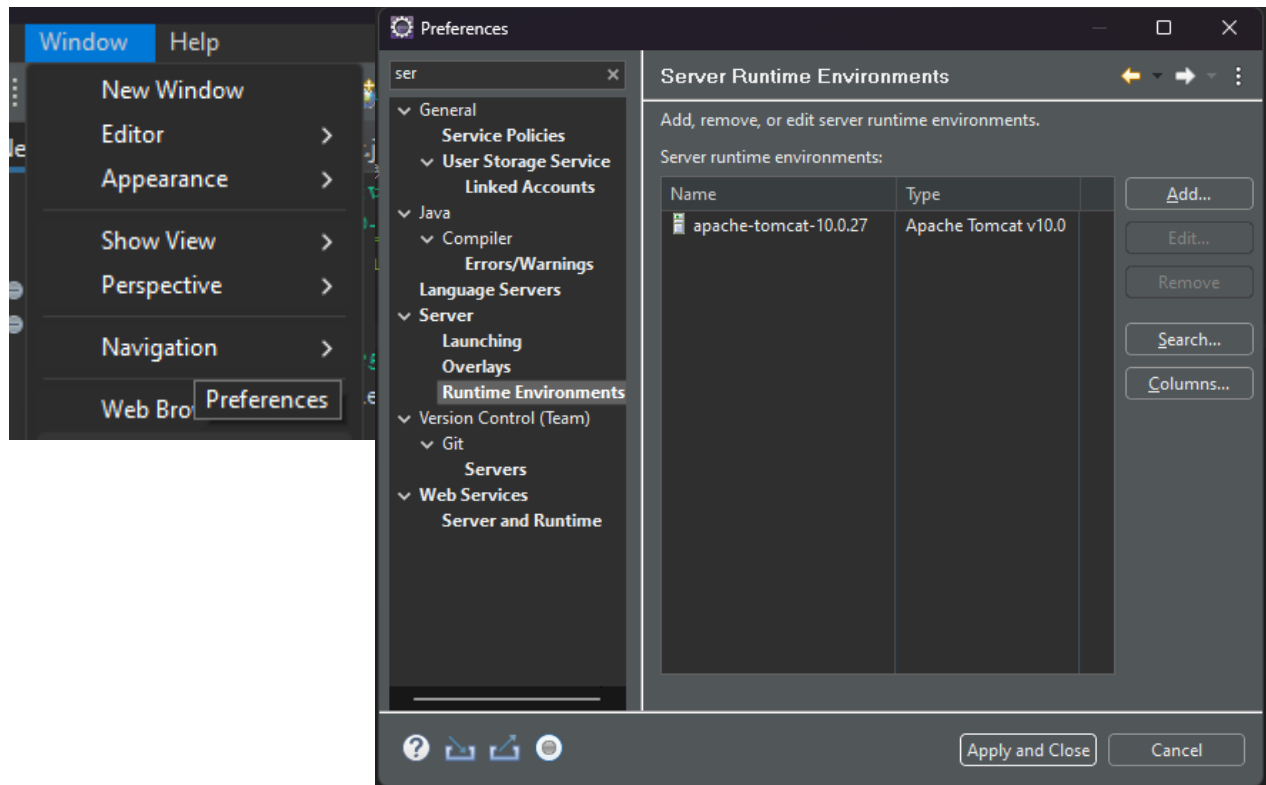
1. Instalar y configurar el entorno de trabajo.



En la instalación hay que seleccionar esta versión:



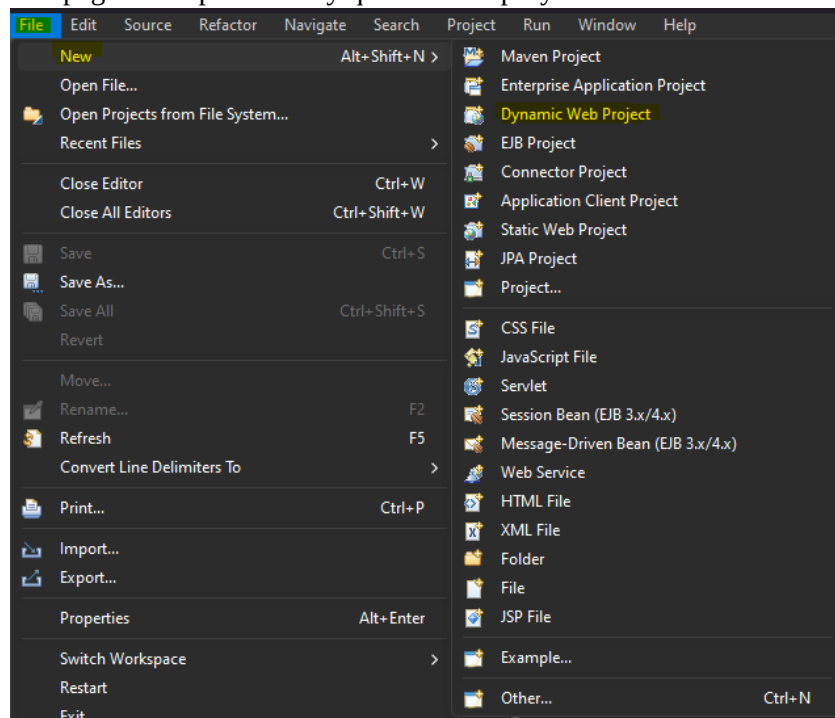
2. Instalar y configurar el servidor de servlets.



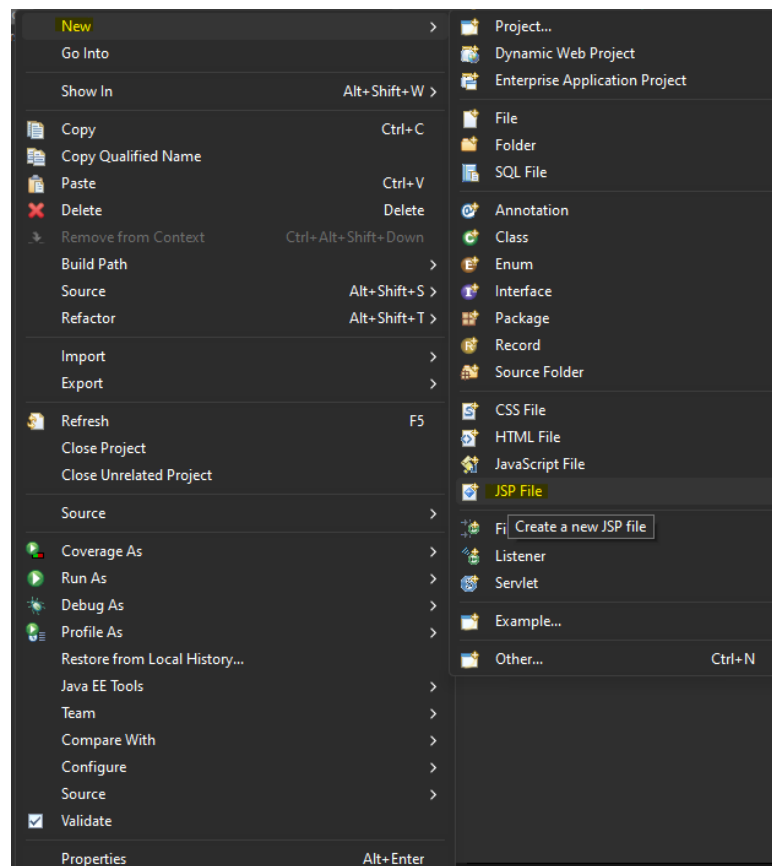
En el apartado “window > preferencias > server > runtime environments” hay que instalar apache Tomcat, en este caso ya lo tenia instalado pero si tuviera que instalarlo solo hay que darle al botón add y seleccionar apache y seleccionar la version 10 de Tomcat.

3. Desarrollar una página JSP que muestre el mensaje “Hello World”.

Para desarrollar una página JSP primero hay que crear un proyecto dinámico.



Después crear el fichero JSP.



4. Desarrollar una aplicación que calcule el doble de un número.

Primero hay que añadir el formulario para que permita introducir el número.

```
NewFile.jsp x dobleNServlet.java
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="ISO-8859-1">
7 <title>Insert title here</title>
8 </head>
9 <body>
10 <h1>Hola Manuel David Ayala Reina</h1>
11 <form action="dobleNServlet" method="post">
12   Introduce un número:
13   <input type="text" name="numero" size="20">
14   <input type="submit" value="Enviar">
15 </form>
16 </body>
17 </html>
```

Crear el server con el nombre dobleNServlet.

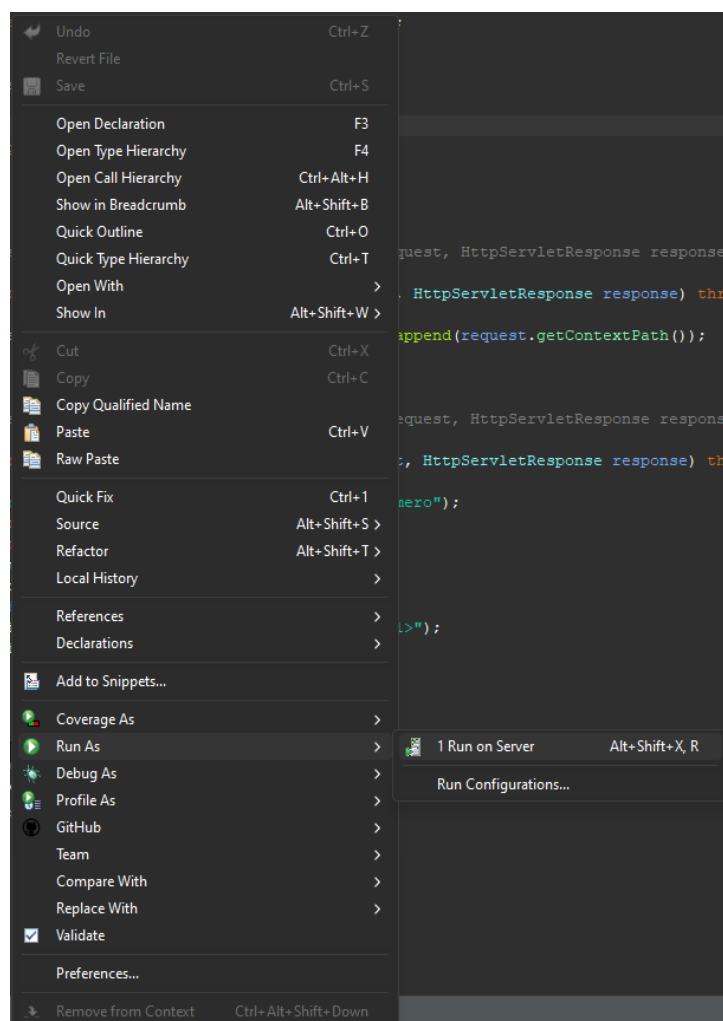
Modificar el servlet para que pueda hacer el doble del número.

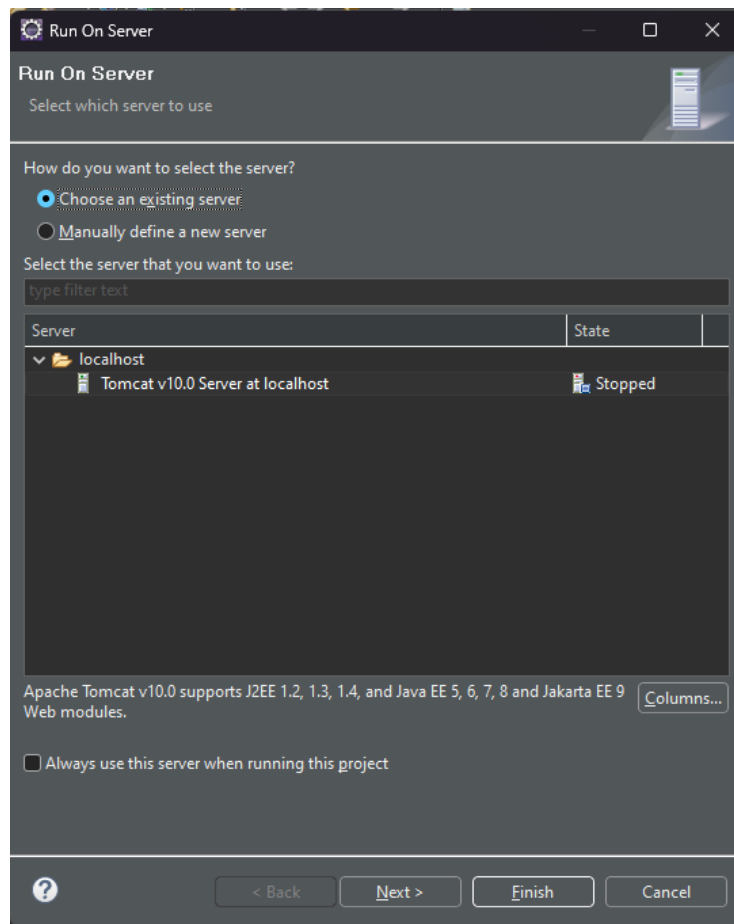
```

NewFile.jsp  dobleNServlet.java X
3 import jakarta.servlet.ServletException;
10
11 /**
12  * Servlet implementation class dobleNServlet
13  */
14 public class dobleNServlet extends HttpServlet {
15     private static final long serialVersionUID = 1L;
16
17     /**
18      * @see HttpServlet#HttpServlet()
19      */
20     public dobleNServlet() {
21         super();
22         // TODO Auto-generated constructor stub
23     }
24
25     /**
26      * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
27      */
28     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
29         // TODO Auto-generated method stub
30         response.getWriter().append("Served at: ").append(request.getContextPath());
31     }
32
33     /**
34      * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
35      */
36     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
37         // TODO Auto-generated method stub
38         String strNumero = request.getParameter("numero");
39         int numero;
40         int doble;
41         numero = Integer.parseInt(strNumero);
42         doble = numero*2;
43         PrintWriter writer = response.getWriter();
44         writer.println("<html>DOBLE: " + doble + "</html>");
45         writer.close();
46     }
47

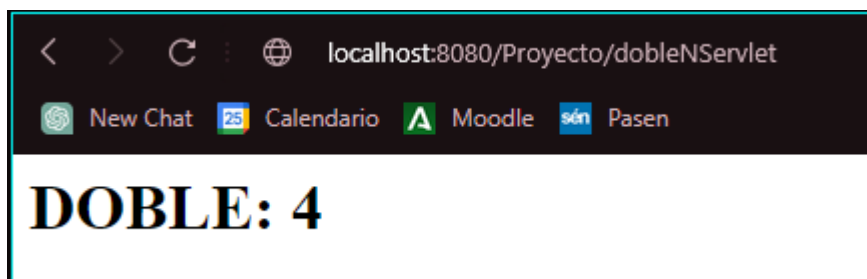
```

Para ejecutarlo en el server seleccionas Run As y seleccionas el seleccionas el Tomcat.





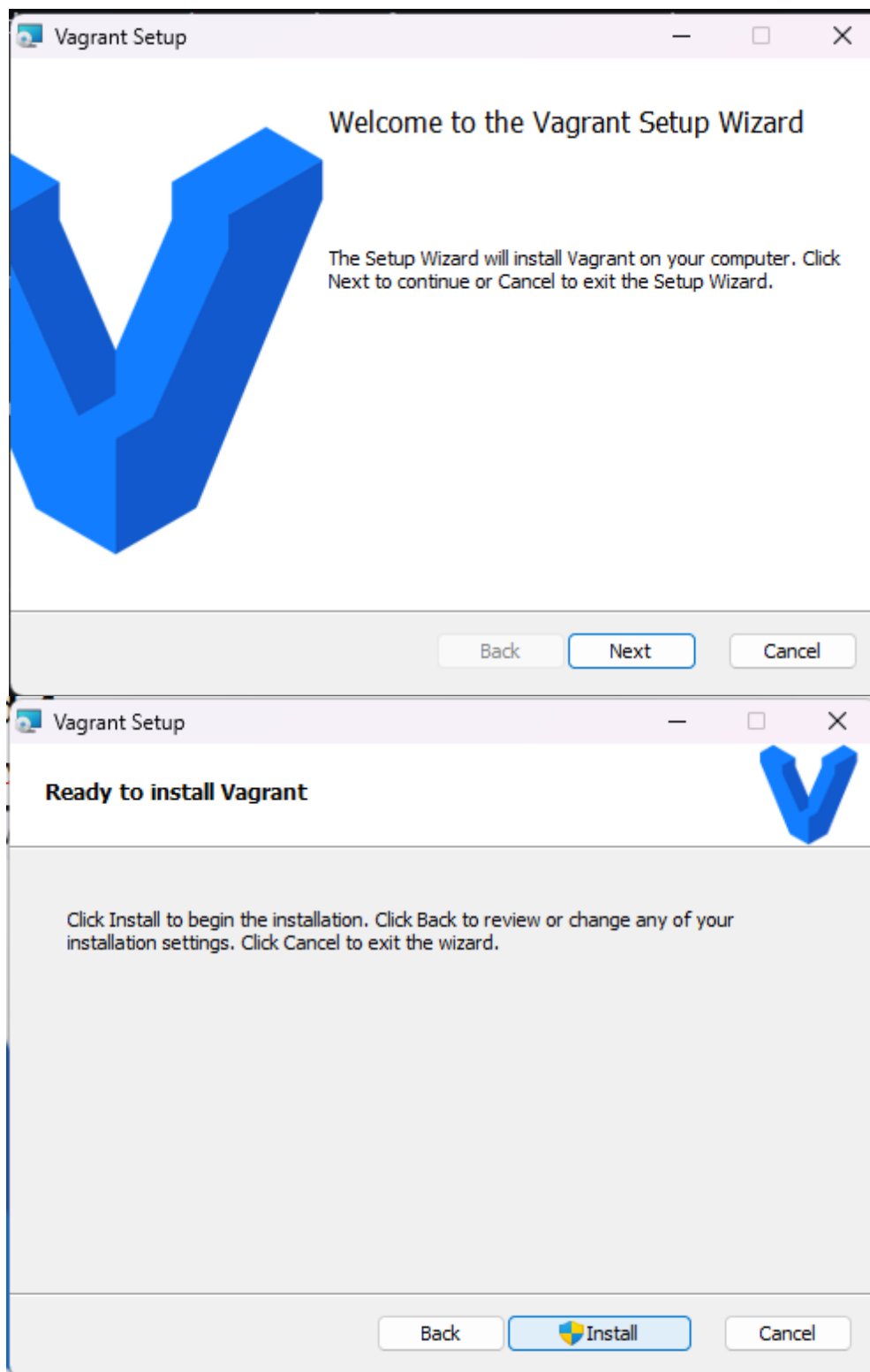
5.Ejecución:



Se puede ver como hace el doble de 2.

Actividad 4.

Paso 1. Instalar vagrant en la máquina física.



Paso 2. Comprobar la instalación.

```
(Manuela)-26/09/2023 10:16:17,31-C:\Users\manua>
vagrant --version
Vagrant 2.3.7
```

Paso 3. Añade el proyecto Laravel Homestead a tu entorno Vagrant seleccionando VirtualBox como proveedor de virtualización. Puedes hacerlo ejecutando el siguiente comando en tu terminal:

```
manua@DESKTO==> box: Loading metadata for box 'laravel/homestead'
box: URL: https://vagrantcloud.com/laravel/homestead
This box can work with multiple providers! The providers that it
can work with are listed below. Please review the list and choose
the provider you will be working with.

1) libvirt
2) parallels
3) virtualbox
```

```
=> box: Adding box 'laravel/homestead' (v13.0.0) for provider: virtualbox
box: Downloading: https://vagrantcloud.com/laravel/boxes/homestead/versions/
13.0.0/providers/virtualbox.box
Progress: 0% (Rate: 0/s, Estimated time remaining: --:--:--)
box:
box: Calculating and comparing box checksum...
==> box: Successfully added box 'laravel/homestead' (v13.0.0) for 'virtualbox'!
```

Paso 4. Clonar el proyecto de Laravel Homestead en el directorio de trabajo:

```
manua@DESKTOP-TKI556G MINGW64 ~
$ git clone https://github.com/laravel/homestead.git
Cloning into 'homestead'...
remote: Enumerating objects: 6330, done.
remote: Counting objects: 100% (107/107), done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 6330 (delta 68), reused 91 (delta 60), pack-reused 6223
Receiving objects: 100% (6330/6330), 1.55 MiB | 1.08 MiB/s, done.
Resolving deltas: 100% (3987/3987), done.
```

Paso 5. Ve al directorio donde clonaste el repositorio Homestead:

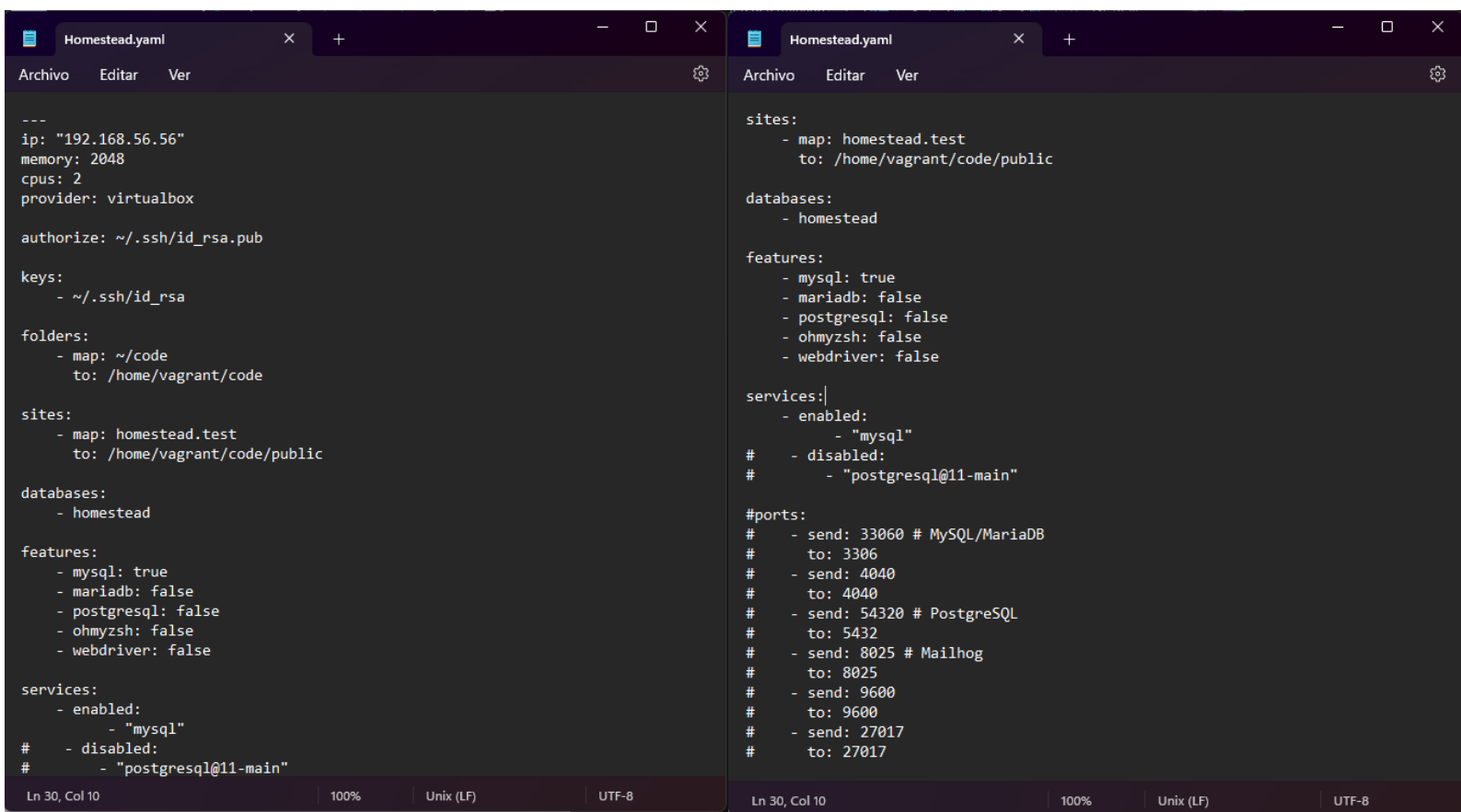
```
manua@DESKTOP-TKI556G MINGW64 ~
$ cd homestead

manua@DESKTOP-TKI556G MINGW64 ~/homestead (main)
$
```

Paso 6. Ejecuta el script de inicialización `init.bat` o `init.sh`, dependiendo de tu sistema operativo. Este script configurará Homestead para tu proyecto.

```
manua@DESKTOP-TKI556G MINGW64 ~/homestead (main)
$ bash init.sh
Homestead initialized!
```

Paso 7. Observar el archivo de configuración Homestead.yaml, especialmente en folders, y sites. Es posible tener múltiples proyectos en la misma máquina.



```
---
ip: "192.168.56.56"
memory: 2048
cpus: 2
provider: virtualbox

authorize: ~/.ssh/id_rsa.pub

keys:
  - ~/.ssh/id_rsa

folders:
  - map: ~/code
    to: /home/vagrant/code/public

sites:
  - map: homestead.test
    to: /home/vagrant/code/public

databases:
  - homestead

features:
  - mysql: true
  - mariadb: false
  - postgresql: false
  - ohmyzsh: false
  - webdriver: false

services:
  - enabled:
    - "mysql"
  # - disabled:
  #   - "postgresql@11-main"
```

```
sites:
  - map: homestead.test
    to: /home/vagrant/code/public

databases:
  - homestead

features:
  - mysql: true
  - mariadb: false
  - postgresql: false
  - ohmyzsh: false
  - webdriver: false

services:|
  - enabled:
    - "mysql"
  # - disabled:
  #   - "postgresql@11-main"

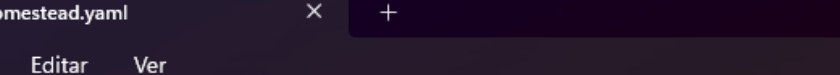
#ports:
# - send: 33060 # MySQL/MariaDB
#   to: 3306
# - send: 4040
#   to: 4040
# - send: 54320 # PostgreSQL
#   to: 5432
# - send: 8025 # Mailhog
#   to: 8025
# - send: 9600
#   to: 9600
# - send: 27017
#   to: 27017
```

Paso 8. Lanza la máquina.

Antes de lanzarla tienes que conseguir una clave SSH con este comando:

```
(Manuela)-26/09/2023 12:29:18,45-C:\Users\manua>
ssh-keygen -t rsa -C "manuayala27@gmail.com"
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\manua/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\manua/.ssh/id_rsa
Your public key has been saved in C:\Users\manua/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:hLGMFhKSfVm/3JvCwgKTbmUuQ/ytj2Cv06P1t/JHJj8 manuayala27@gmail.com
The key's randomart image is:
+---[RSA 3072]-----+
|.00..00|
|...0+ =|
|.0 + 0|
|. 0 0 0|
|* 0 S .|
|o B o .. oo|
|B.= + o=o|
|o.*o=...E|
|o++oo+oo .|
+---[SHA256]-----+
```

Y despues poner la ubicación del archivo en el apartado autorize en el Homestead.yaml:



```
---
ip: "192.168.56.56"
memory: 2048
cpus: 2
provider: virtualbox

authorize: C:\Users\manua/.ssh/id_rsa.pub
```

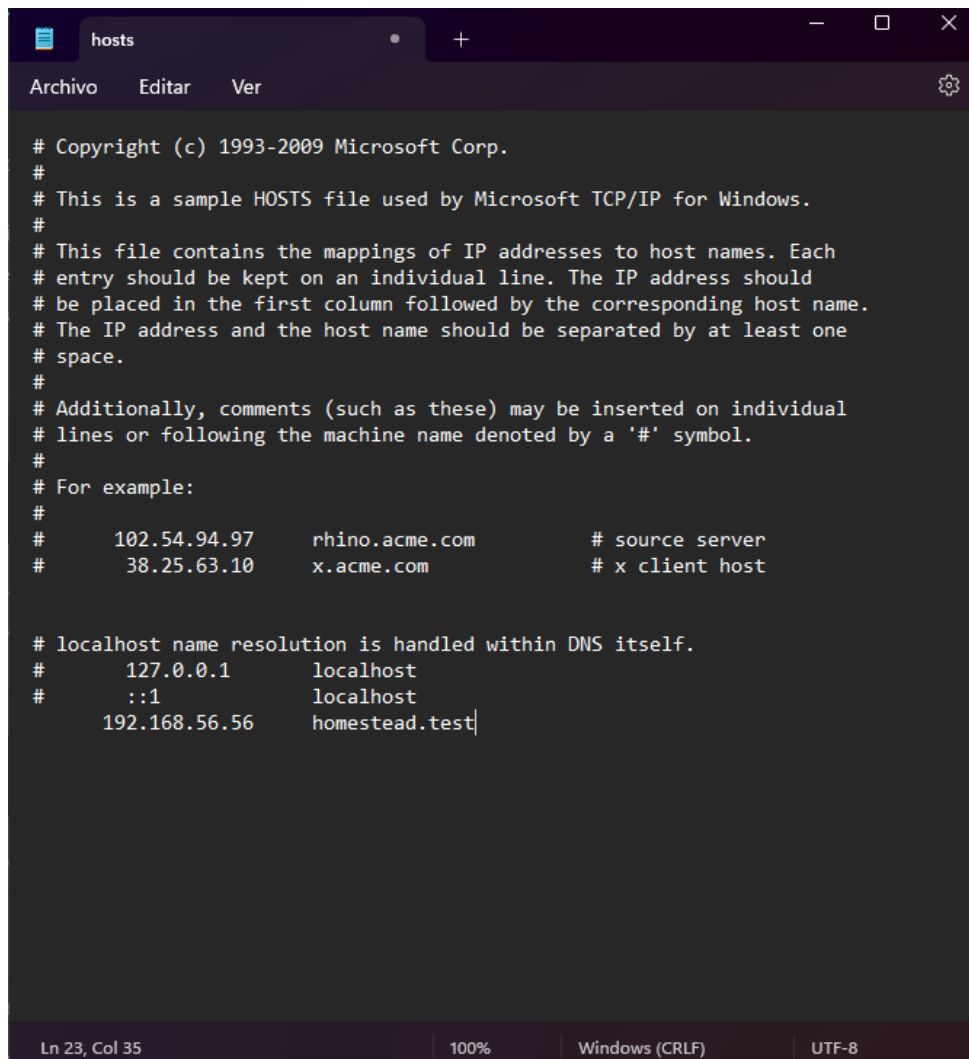
Despues de esto ya puedes hacer vagrant up:

```
manua@DESKTOP-TKI556G MINGW64 ~/homestead (main)
$ vagrant up
Bringing machine 'homestead' up with 'virtualbox' provider...
==> homestead: Importing base box 'laravel/homestead'...
==> homestead: Matching MAC address for NAT networking...
==> homestead: Checking if box 'laravel/homestead' version '13.0.0' is up to dat
e...
```

Y en este momento ya estas correctamente conectado a vagrant:

[illegible]

Paso 9. Modifica el fichero de hosts para asignar nombre y dirección ip del archivo de configuración Homstead.yaml.



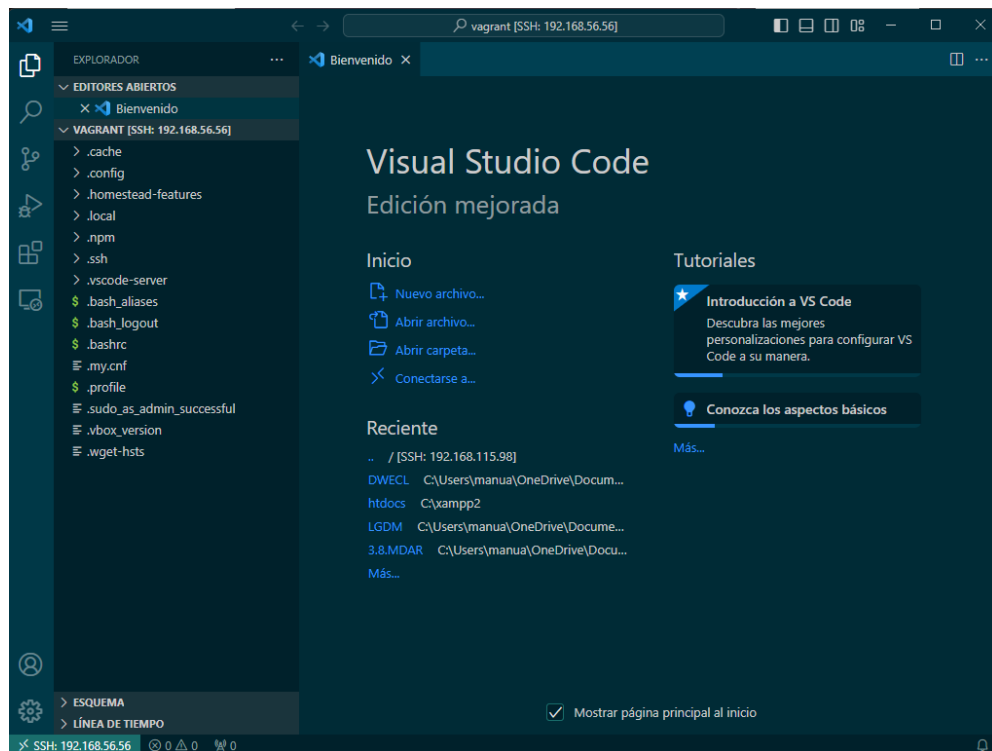
```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com       # source server
#       38.25.63.10       x.acme.com          # x client host


# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
192.168.56.56            homestead.test|
```

Ln 23, Col 35 | 100% | Windows (CRLF) | UTF-8

Paso 10. Crea una página que muestre el mensaje “Hello Vagrant”.

En este caso voy a hacerlo con visual por ssh:



Para crear la página hay que crear las siguientes carpetas:

```
mkdir -p /home/vagrant/code/public
```

Ahora puedes ver tu página introduciendo en el navegador: “**homestead.test**”

