

INSTALACIÓN APACHE

Despliegue de Aplicaciones Web

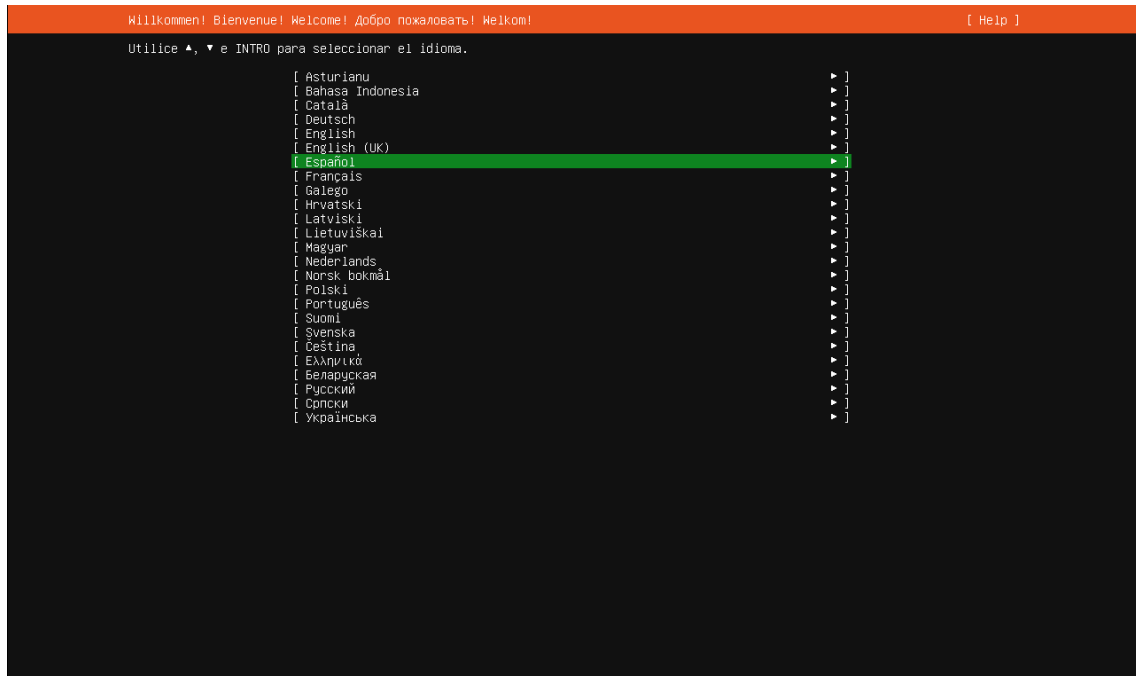
Práctica instalación Ubuntu Server

Tutorial para la instalación de apache en Ubuntu Server

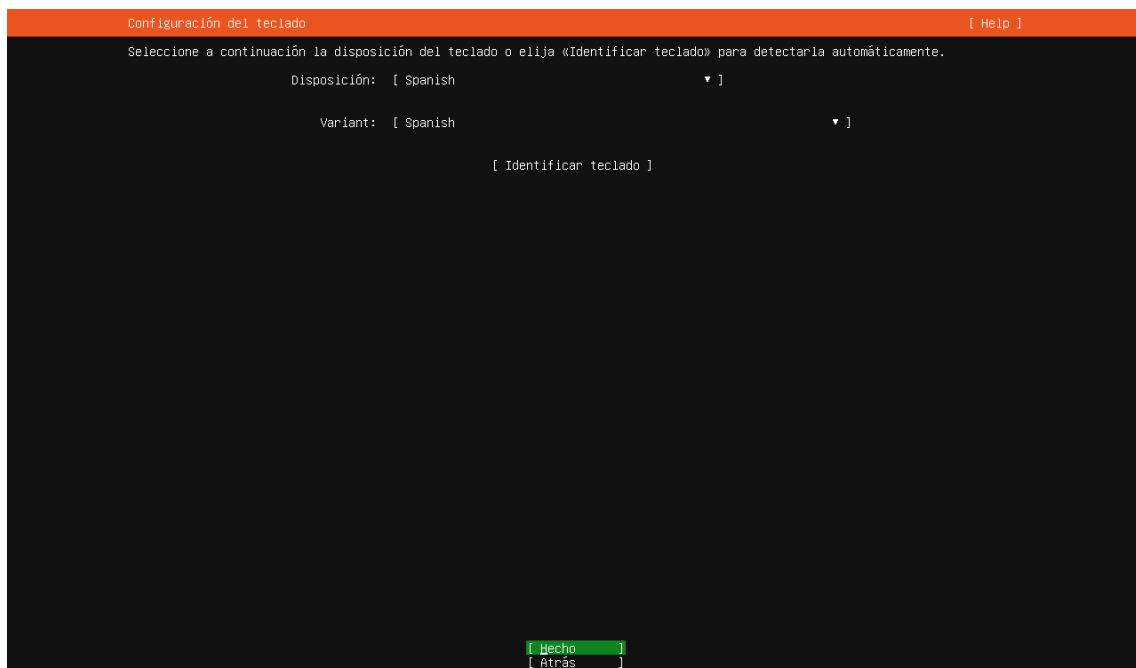
Javier Ruiz Muro

INSTALACIÓN UBUNTU SERVER

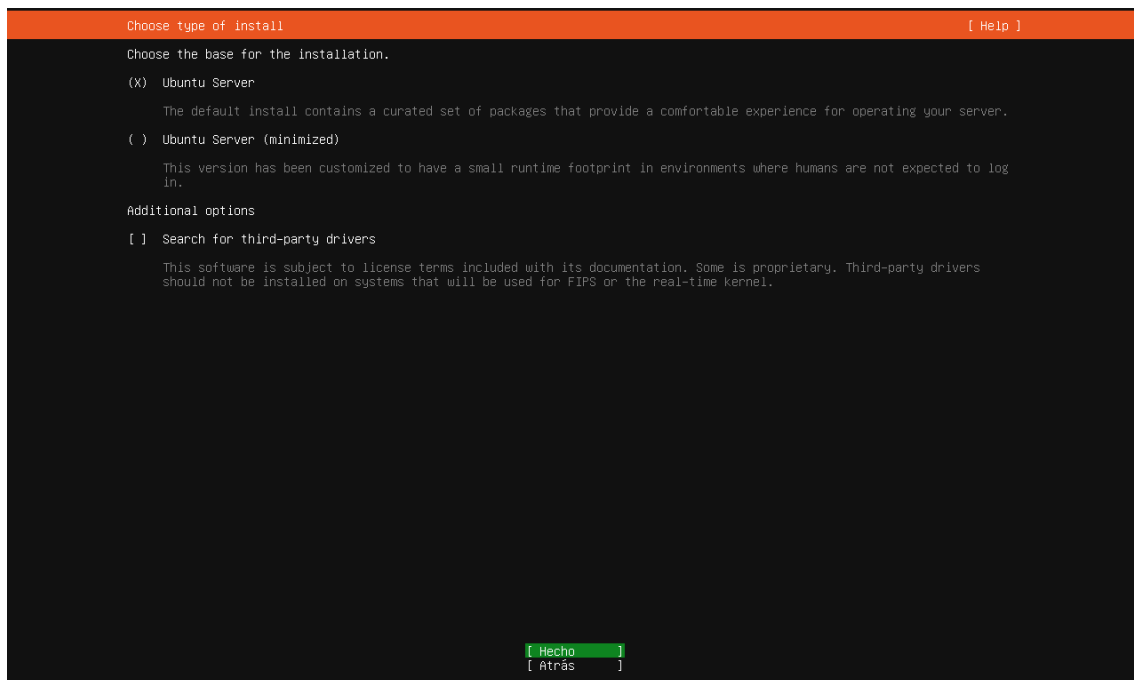
Seleccionamos el idioma de nuestra máquina:



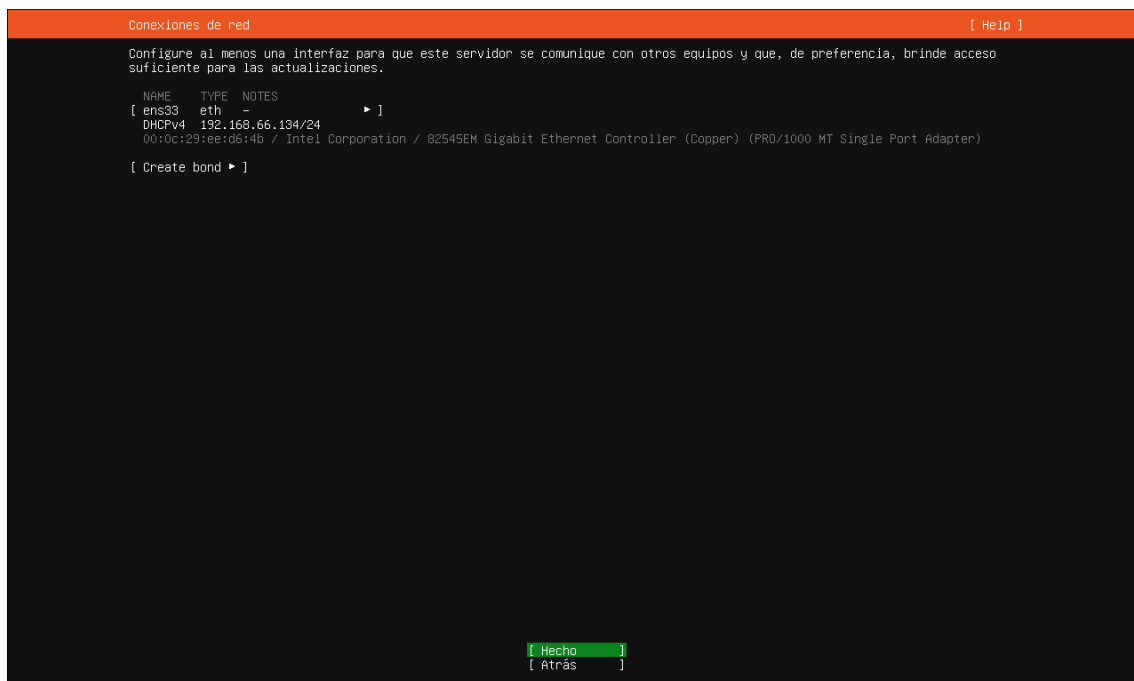
Además de la distribución de teclado:



Seleccionamos la instalación por defecto de *Ubuntu Server*.



Por defecto se nos configura la interfaz de red:



Estas cuatro pestañas las dejamos tal cual vienen por defecto:

Configure proxy [Help]

If this system requires a proxy to connect to the Internet, enter its details here.

Proxy address:

If you need to use a HTTP proxy to access the outside world, enter the proxy information here. Otherwise, leave this blank.

The proxy information should be given in the standard form of "http://[[user][:pass]@]host[:port]"/".

[Hecho]
[Atrás]

Configure Ubuntu archive mirror [Help]

If you use an alternative mirror for Ubuntu, enter its details here.

Mirror address:

You may provide an archive mirror that will be used instead of the default.

[Hecho]
[Atrás]

Guided storage configuration

[Help]

Configure a guided storage layout, or create a custom one:

(X)

Use an entire disk

[/dev/sda local disk 20.000G ▼]

(X)

Set up this disk as an LVM group

[]

Encrypt the LVM group with LUKS

Passphrase:

Confirm passphrase:

()

Custom storage layout

[Hecho]

[Atrás]

Storage configuration

[Help]

RESUMEN DEL SISTEMA DE ARCHIVOS

PUNTO DE MONTAJE	TAMAÑO	TIPO	TIPO DE DISPOSITIVO	
[/	10.000G	new ext4	new LVM logical volume	►]
[/boot	1.771G	new ext4	new partition of disco local	►]

DISPOSITIVOS DISPONIBLES

DISPOSITIVO	TIPO	TAMAÑO	
[ubuntu-vg (new)	LVM volume group	18.222G	►]
espacio disponible		8.222G	►]

[Create software RAID (md)

►]

[Crear grupo de volúmenes (LVM)

►]

DISPOSITIVOS UTILIZADOS

DISPOSITIVO	TIPO	TAMAÑO	
[ubuntu-vg (new)	LVM volume group	18.222G	►]
ubuntu-iv	new, to be formatted as ext4, mounted at /	10.000G	►]

[/dev/sda

partition 1	new, BIOS grub spacer	20.000G	►]
partition 2	new, to be formatted as ext4, mounted at /boot	1.000M	►]
partition 3	new, to be formatted as ext4, mounted at /boot	1.771G	►]
	new, PV of LVM volume group ubuntu-vg	18.225G	►]

[Hecho]

[Restablecer]

[Atrás]

En la siguiente, introducimos nuestros datos:

Configuración de perfil [Help]

Proporcione el nombre de usuario y la contraseña que utilizará para acceder al sistema. Puede configurar el acceso SSH en la pantalla siguiente, pero aun se necesita una contraseña para sudo.

Su nombre:

El nombre del servidor:
The name it uses when it talks to other computers.

Elija un nombre de usuario:

Elija una contraseña:

Confirme la contraseña:

[Hecho]

Una vez hayamos añadido los datos mencionados anteriormente, marcamos la casilla para instalar *OpenSSH* y continuamos.

Configuración de SSH [Help]

You can choose to install the OpenSSH server package to enable secure remote access to your server.

☒ Instalar servidor OpenSSH

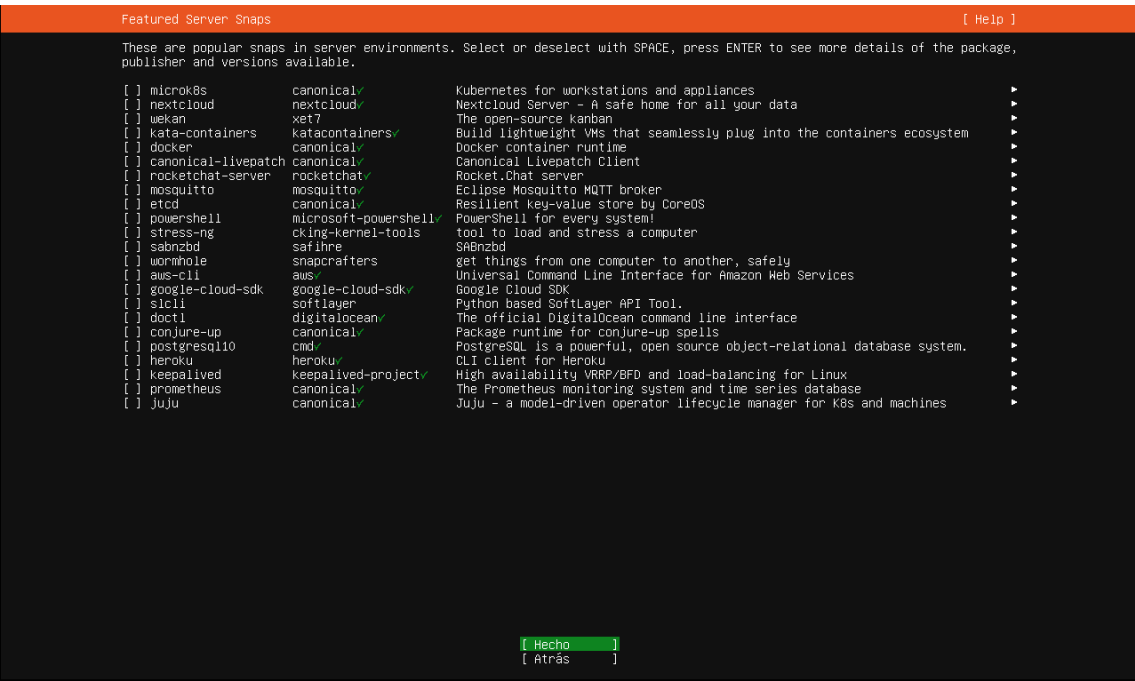
Importar identidad SSH:
You can import your SSH keys from GitHub or Launchpad.

Importar nombre de usuario:

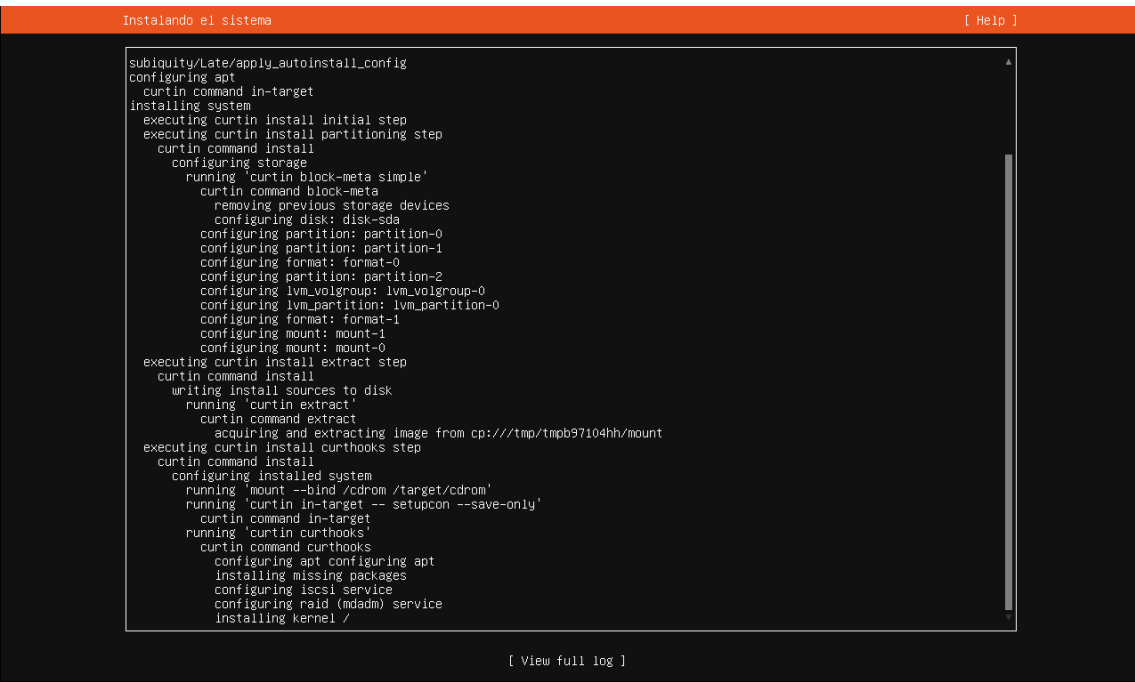
☒ Permitir autenticación con contraseña por SSH

[Hecho]
[Atrás]

La siguiente ventana la dejamos tal cual y avanzamos.



A continuación, se procederá a la instalación del sistema operativo.



INSTALACIÓN DE APACHE

Comenzamos actualizando los repositorios:

```
javierrmserver@javierrm:~$ sudo apt-get update
```

Instalamos apache:

```
javierrmserver@javierrm:~$ sudo apt-get install apache2
```

Configuramos el firewall:

```
javierrmserver@javierrm:~$ sudo ufw app list
Available applications:
  Apache
  Apache Full
  Apache Secure
  OpenSSH
javierrmserver@javierrm:~$ _
```

Activamos el firewall:

```
javierrmserver@javierrm:~$ sudo ufw enable
Firewall is active and enabled on system startup
javierrmserver@javierrm:~$
```

Permitimos el tráfico en el puerto 80:

```
javierrmserver@javierrm:~$ sudo ufw allow 'Apache' _
```

Comprobamos el estado con el siguiente comando:

```
javierrmserver@javierrm:~$ sudo ufw status
Status: active

To Action From
--
Apache ALLOW Anywhere
Apache (v6) ALLOW Anywhere (v6)

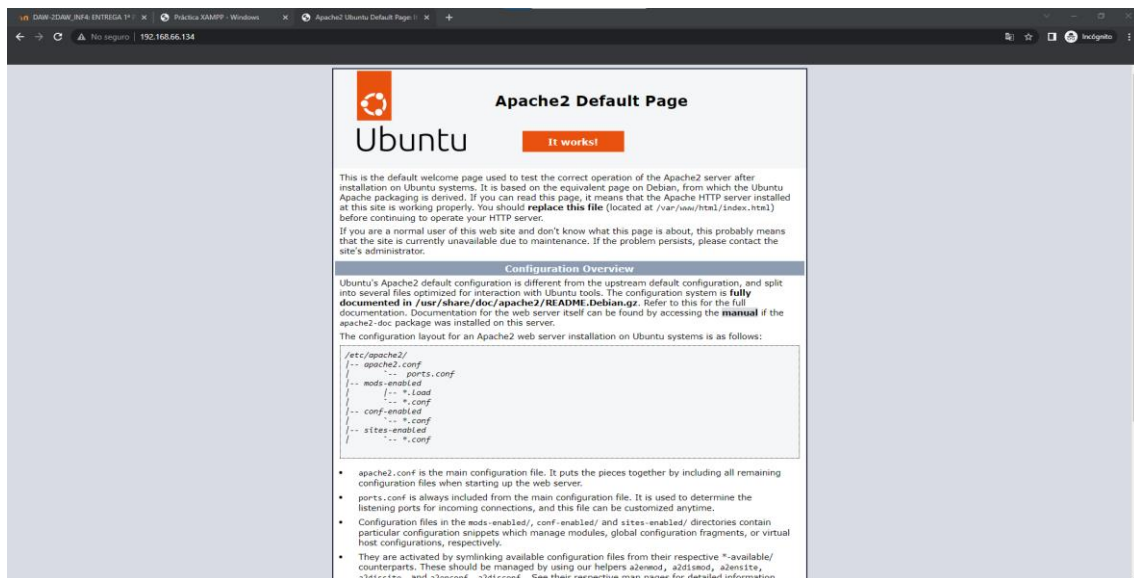
javierrmserver@javierrm:~$ _
```


Verificamos el estado de apache (que este corriendo).

```
javierrmsrver@javierrm:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Mon 2023-01-30 16:25:02 UTC; 4min 1s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2454 (apache2)
    Tasks: 55 (limit: 4524)
   Memory: 5.1M
      CPU: 65ms
   CGroup: /system.slice/apache2.service
           └─2454 /usr/sbin/apache2 -k start
             └─2455 /usr/sbin/apache2 -k start
               └─2456 /usr/sbin/apache2 -k start

ene 30 16:25:02 javierrm systemd[1]: Starting The Apache HTTP Server...
ene 30 16:25:02 javierrm apache2[2453]: AH00558: apache2: Could not reliably determine the server's
ene 30 16:25:02 javierrm systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

Comprobamos que efectivamente funciona:



Para detener el servidor web introducimos lo siguiente:

```
javierrmsrver@javierrm:~$ sudo systemctl stop apache2
javierrmsrver@javierrm:~$
```

Para iniciar el servidor web introducimos lo siguiente:

```
javierrmsrver@javierrm:~$ sudo systemctl start apache2
javierrmsrver@javierrm:~$ _
```

Para reiniciar el servidor web introducimos lo siguiente:

```
javierrmsrver@javierrm:~$ sudo systemctl restart apache2
javierrmsrver@javierrm:~$ _
```

Para realizar cambios de configuración el servidor web introducimos lo siguiente:

```
javierrmsrver@javierrm:~$ sudo systemctl reload apache2
javierrmsrver@javierrm:~$
```

Para deshabilitar y habilitar el inicio automático escribimos los siguientes comandos:

```
javierrmserver@javierrm:~$ sudo systemctl disable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable apache2
Removed "/etc/systemd/system/multi-user.target.wants/apache2.service".
javierrmserver@javierrm:~$ 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
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
javierrmserver@javierrm:~$ _
```

Una vez hemos terminado de instalar y configurar apache nos disponemos a configurar los hosts virtuales (creando nuestro sitio virtual).

En primer lugar, creamos y daremos permisos al directorio donde alojaremos nuestro sitio:

```
javierrmserver@javierrm:~$ sudo mkdir /var/www/webvinosrioja
javierrmserver@javierrm:~$ sudo chown -R $USER:$USER /var/www/webvinosrioja
javierrmserver@javierrm:~$ sudo chmod -R 755 /var/www/webvinosrioja
javierrmserver@javierrm:~$ _
```

A continuación, creamos una página de ejemplo (obviamente dentro de la carpeta que acabamos de crear):

```
GNU nano 6.4 /var/www/webvinosrioja/index.html *
<html>
  <head>
    <title>Página Web 1 - SMR</title>
  </head>
  <body>
    <h1>Enhorabuena, has accedido al sitio virtual que muestra información sobre los vi<
  </body>
</html>
```

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo
^X Exit	^R Read File	^_ Replace	^U Paste	^J Justify	^_ Go To Line	M-E Redo

Añadimos las siguiente líneas indicando el dominio de nuestro sitio web:

```
GNU nano 6.4 /etc/hosts *
127.0.0.1 localhost
127.0.1.1 javierrm

127.0.0.1 www.webvinosrioja.com
127.0.0.1 www.webturismorural.com
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

Copiamos el siguiente fichero para crear un archivo de host virtual con las siguientes directivas:

```
javierrmserver@javierrm:~$ cd /etc/apache2/sites-available/
javierrmserver@javierrm:/etc/apache2/sites-available$ sudo cp 000-default.conf webvinosrioja.conf
javierrmserver@javierrm:/etc/apache2/sites-available$
```

Dichas directivas:

```
GNU nano 6.4 /etc/apache2/sites-available/webvinosrioja.conf *
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    ServerName www.webvinosrioja.com
    ServerAlias www.webvinosrioja
    DocumentRoot /var/www/webvinosrioja

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

Habilitamos el archivo creado anteriormente con la herramienta *a2ensite* y deshabilitamos el archivo predeterminado:

```
javierrmsrver@javierrm:/etc/apache2/sites-available$ sudo a2ensite webvinosrioja.conf
Enabling site webvinosrioja.
To activate the new configuration, you need to run:
  systemctl reload apache2
javierrmsrver@javierrm:/etc/apache2/sites-available$ sudo a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
javierrmsrver@javierrm:/etc/apache2/sites-available$
```

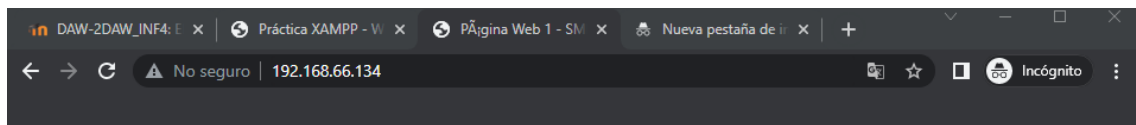
Realizamos una prueba para verificar que no haya errores en la configuración:

```
javierrmsrver@javierrm:/var/www/html$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
javierrmsrver@javierrm:/var/www/html$
```

Por último, reiniciamos Apache para implementar los cambios:

```
javierrmsrver@javierrm:/var/www/html$ sudo systemctl restart apache2
javierrmsrver@javierrm:/var/www/html$ _
```

Con todo esto, Apache debería ser el servidor del nombre de dominio que hemos creado. Para comprobar que efectivamente funciona he accedido a la dirección donde está alojado el sitio y funciona perfectamente como se puede observar en la siguiente fotografía.



Enhorabuena, has accedido al sitio virtual que muestra información sobre los vinos de la Rioja