

Configuración Ubuntu con ssh y dominio propio.

Alejandro David Pasaje Quinayás.

Instructor:

Sebastián Ruiz

CENTRO DE GESTION Y DESARROLLO SURCOLOMBIANO

ADSO

2692929

Pitalito – Huila

2024

## Table of Contents

Introducción.....	3
Objetivos.....	4
Objetivo general.....	4
Objetivos específicos .....	4
Creación de la máquina virtual. ....	5
Instalación de paquetes .....	9
Instalación de ssh.....	21
Configuración de dominio. ....	21
Despliegue del proyecto.....	31
Conclusiones.....	38
Fuentes de referencia .....	39

## Introducción.

En este documento, exploraremos el proceso de creación de una máquina virtual en Ubuntu Server. Abordaremos la instalación adecuada de la máquina virtual y también veremos cómo configurar **bind9** para establecer nuestro propio dominio. Además, discutiremos la configuración de **SSH** para facilitar la comunicación entre el cliente y la máquina virtual. Por último, exploraremos cómo desplegar un proyecto formativo utilizando **Visual Studio Code**.

## Objetivos

### Objetivo general

Crear instalar y desplegar el proyecto formativo usando virtual box junto con Ubuntu y visual estudio code.

### Objetivos específicos

#### 1. Configuración Eficiente de Máquinas Virtuales:

- Aprender a crear y personalizar máquinas virtuales en un entorno Ubuntu Server.
- Dominar la asignación de recursos y la configuración de red.

#### 2. Gestión Avanzada de DNS con bind9:

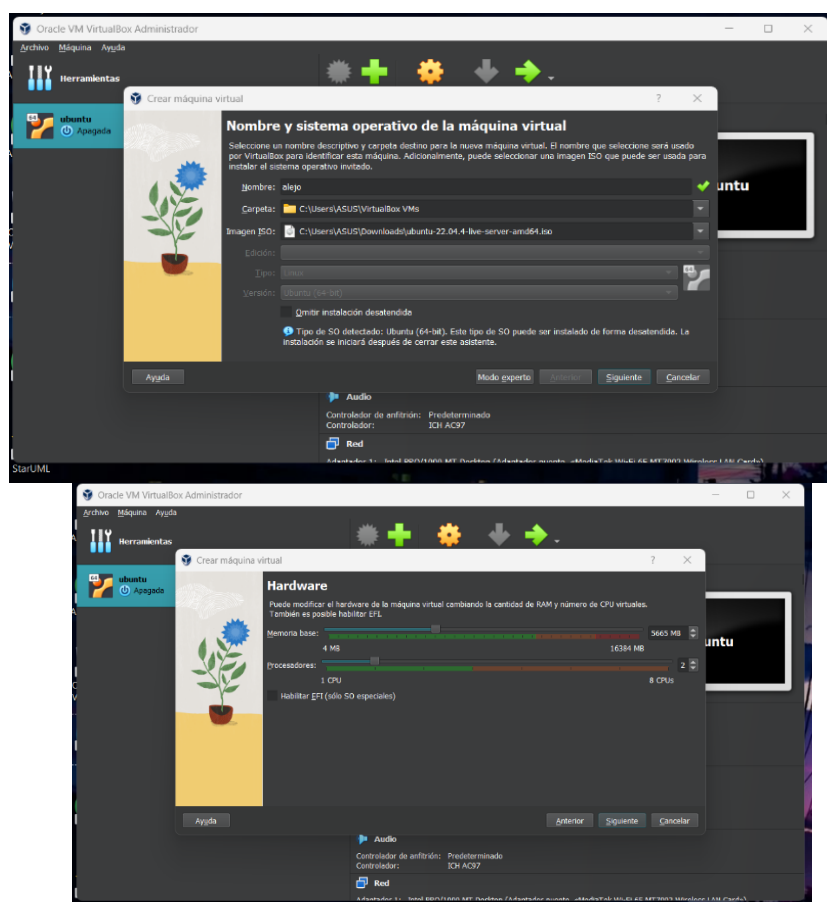
- Instalar y configurar el servicio DNS **bind9**.
- Crear y administrar zonas DNS para un dominio propio.

#### 3. Seguridad y Comunicación con SSH:

- Configurar conexiones seguras mediante SSH.
- Implementar autenticación basada en claves y prácticas de seguridad.

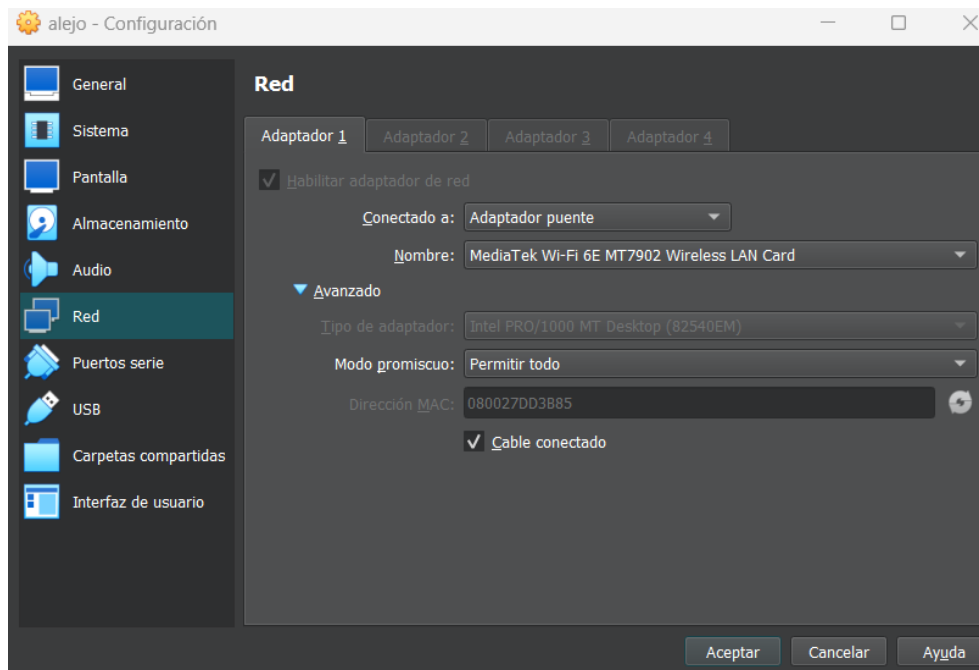
## Creación de la máquina virtual.

Como primer paso crearemos la máquina virtual para eso necesitaremos de virtual box un programa diseñado para la creación de máquinas virtuales simulando servidores.

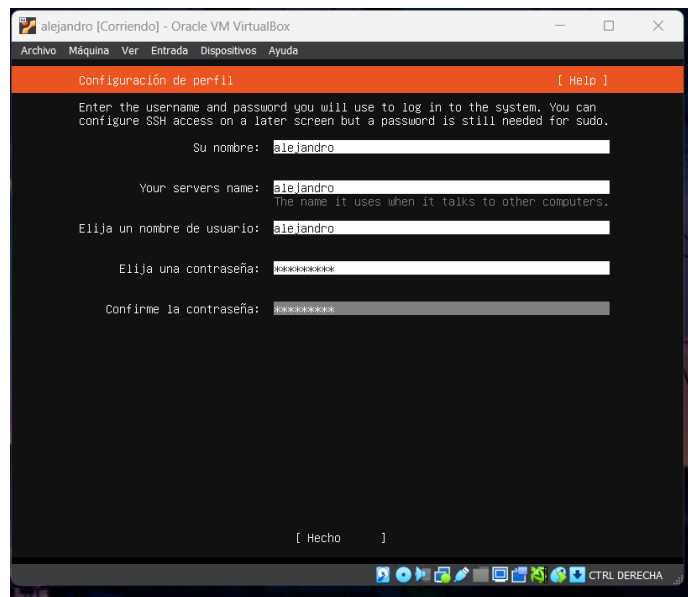


Para que no haya problemas más adelante procederemos a configurar la adaptación de red.

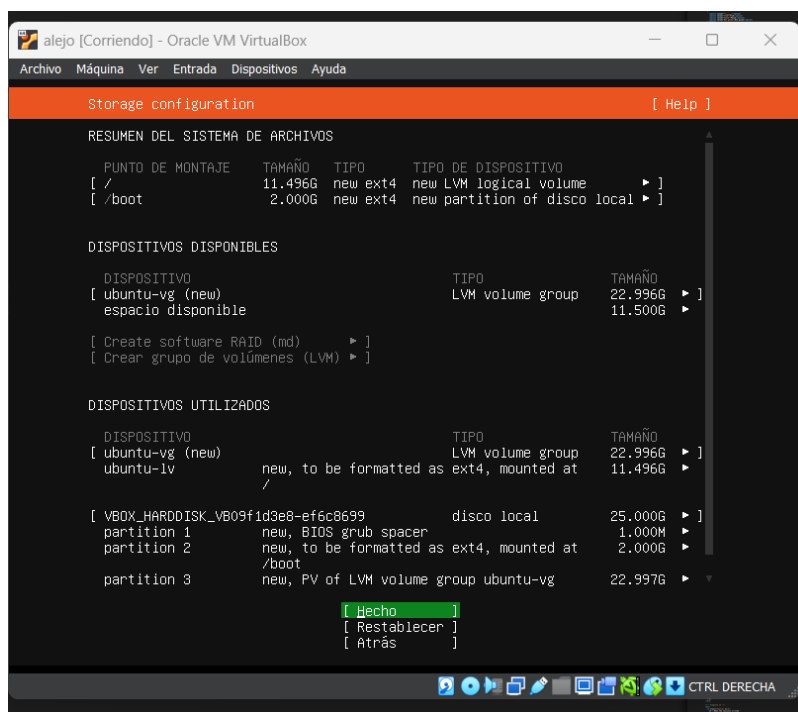
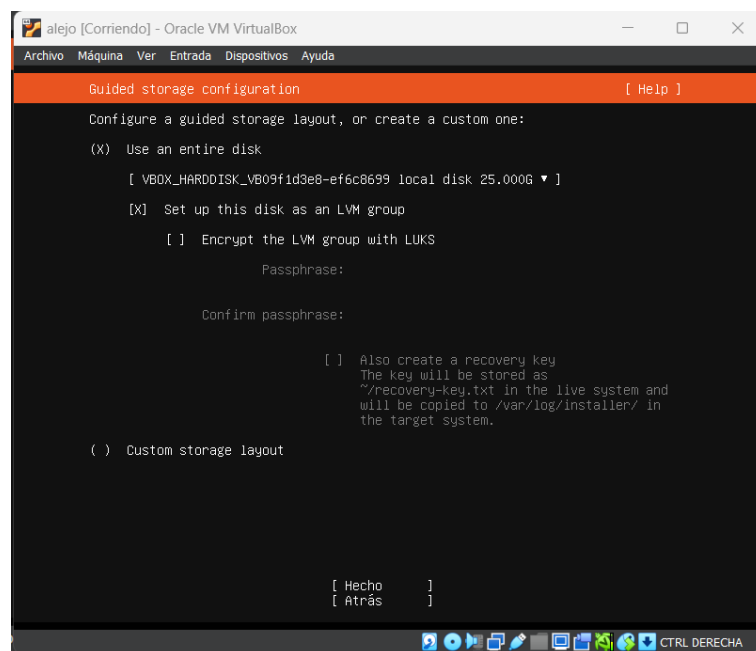
Esto se hace para que la conexión de la maquina no presente problemas.

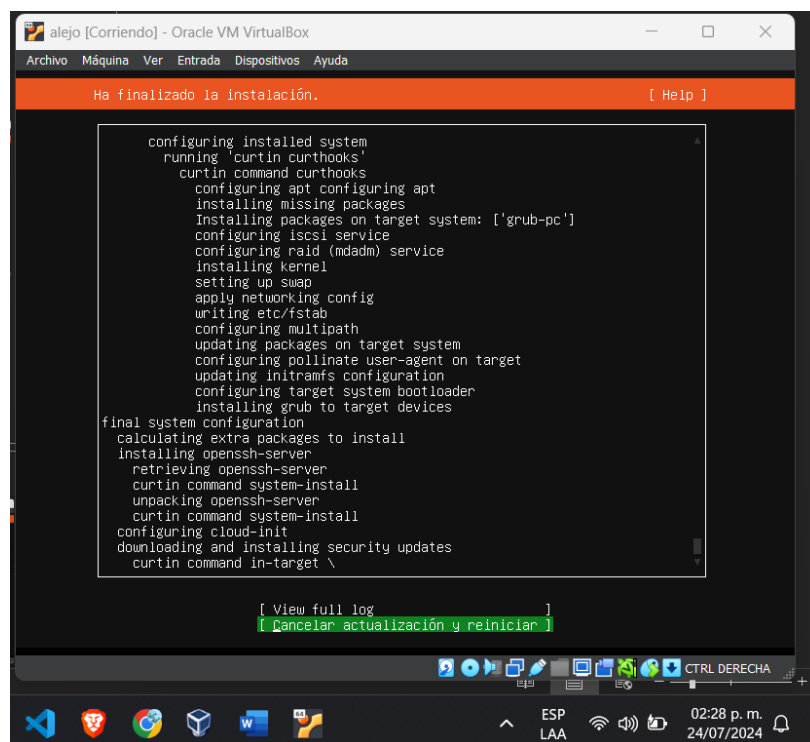
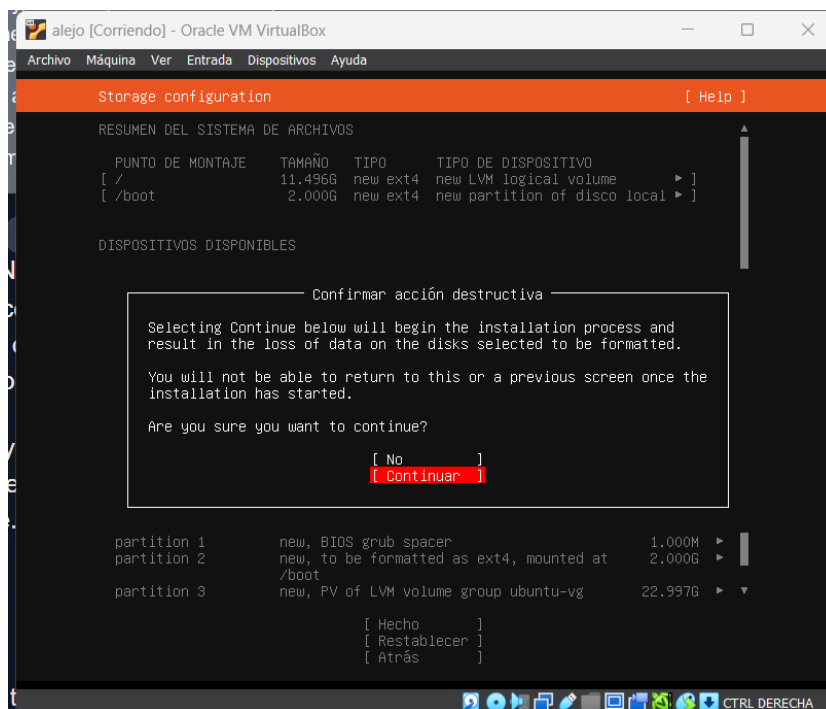


Crearemos un usuario y contraseñas.



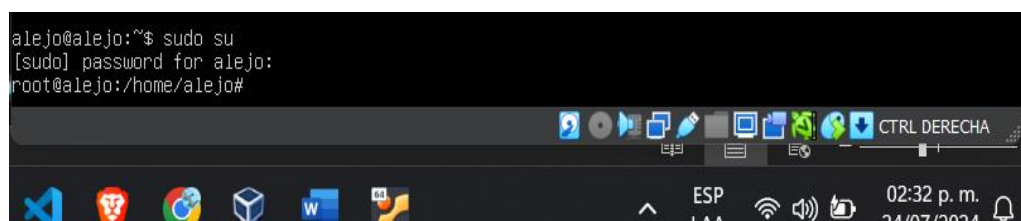
Para la instalacion de la maquina daremos ok y siguiente.







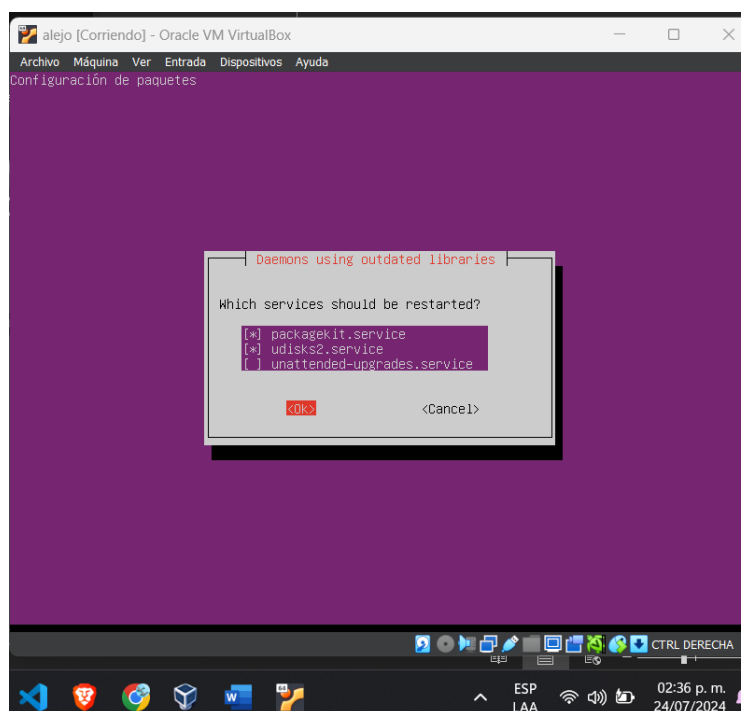
Al finalizar la carga nos saldrá la opción de reiniciar y daremos enter, seguido de eso ingresaremos nuevamente a la maquina y se vera reflejado el usuario y nos pedirá la contraseña que hayamos configurado, procederemos a ingresar como super administrador para seguido instalar los paquetes.

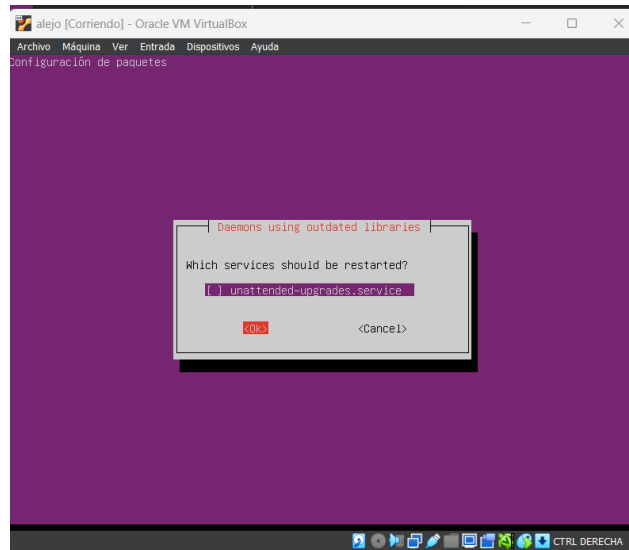


### Instalación de paquetes

Primero actualizaremos los paquetes de la maquina usando los comandos

apt update y udgrade



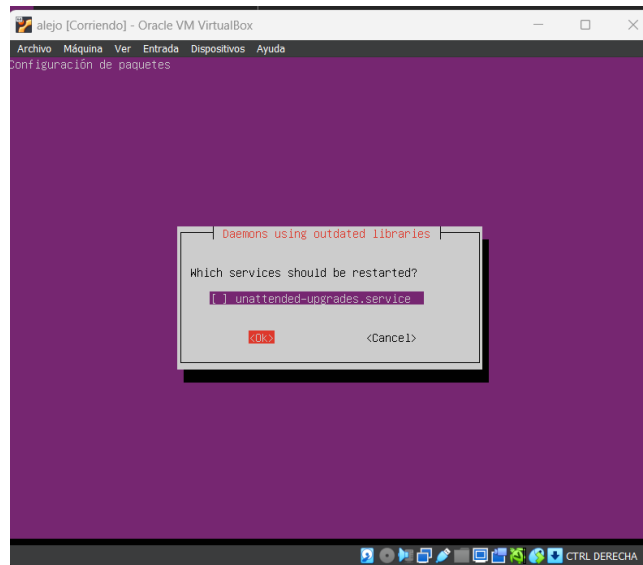


Instalar nginx en la maquina. Es nginx es un servicio es ampliamente utilizado como servidor web y también como proxy inverso.

```
root@alejo:/home/alejo# apt install nginx_
```

Instalar net tools, **Net Tools** es una colección de programas esenciales para controlar el subsistema de red en sistemas Linux.

```
root@alejo:/home/alejo# apt install net-tools
```



Instalación de openvswitch es una herramienta poderosa y versátil que se utiliza en entornos de virtualización y redes.

```
root@alejo:/home/alejo# apt install openvswitch-switch
```

Listamos los componentes

Usando este comando

```

root@alejo:/home/alejo# ufw app list
Available applications:
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
  OpenSSH
root@alejo:/home/alejo# ufw allow 'HTTP'
ERROR: Could not find a profile matching 'HTTP'
root@alejo:/home/alejo# ufw allow 'Nginx HTTP'
Rules updated
Rules updated (v6)
root@alejo:/home/alejo#

```

```

root@alejo:/home/alejo# ufw status
Status: inactive
root@alejo:/home/alejo# ufw enable
Firewall is active and enabled on system startup
root@alejo:/home/alejo# ufw status
Status: active

To Action From
--
Nginx HTTP ALLOW Anywhere
Nginx HTTP (v6) ALLOW Anywhere (v6)

root@alejo:/home/alejo#

```

Verificamos el estado de nginx.

```

root@alejo:/home/alejo# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2024-07-23 20:44:31 UTC; 3min 53s ago
     Docs: man:nginx(8)
  Process: 4509 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0)
  Process: 4510 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0)
   Main PID: 4606 (nginx)
      Tasks: 3 (limit: 6439)
     Memory: 5.3M
          CPU: 54ms
    CGroup: /system.slice/nginx.service
            └─4606 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
              └─4608 "nginx: worker process"
                └─4609 "nginx: worker process"

jul 23 20:44:31 alejo systemd[1]: Starting A high performance web server and a reverse proxy server: nginx.
jul 23 20:44:31 alejo systemd[1]: Started A high performance web server and a reverse proxy server: nginx.
lines 1-17/17 (END)

```

Verificaremos la route a la que estamos conectados

```

root@alejo:/home/alejo# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe9e:5bf8 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:9e:5b:f8 txqueuelen 1000 (Ethernet)
    RX packets 39304 bytes 59341540 (59.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 7138 bytes 448927 (448.9 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 120 bytes 10380 (10.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 120 bytes 10380 (10.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@alejo:/home/alejo# ip route show
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15 metric 100
10.0.2.2 dev enp0s3 proto dhcp scope link src 10.0.2.15 metric 100
192.168.0.1 via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
root@alejo:/home/alejo# _

```


Procederemos a configurar la ip única que vamos a usar. Usando este comando

```

root@alejo:/home/alejo# nano /etc/netplan/00-installer-config.yaml _

```

De manera inicial abrirá de esta manera.

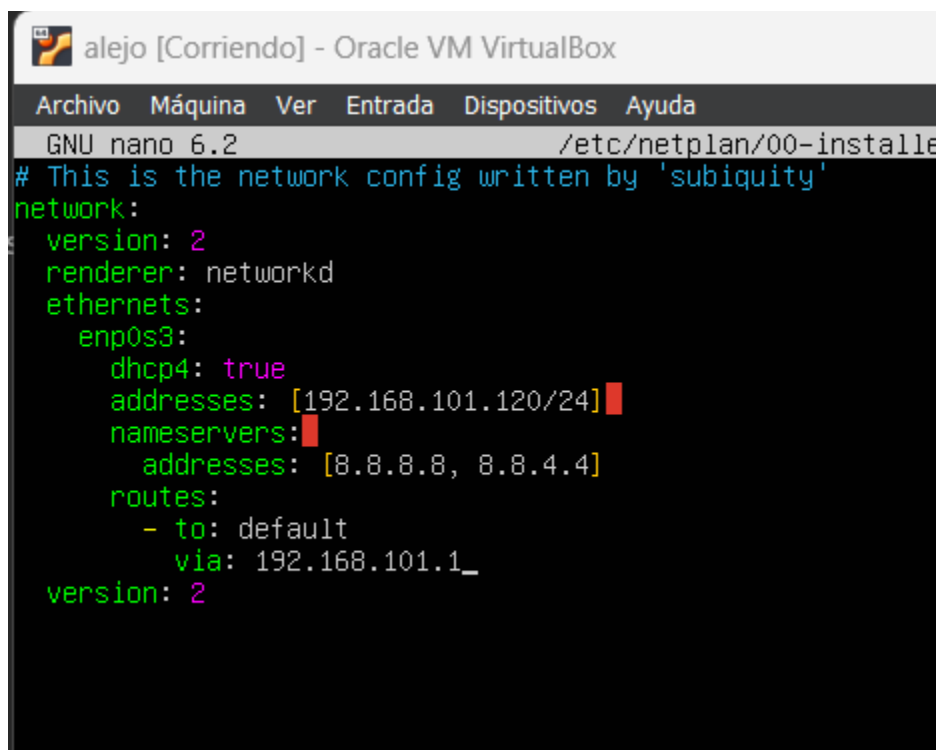


```

alejo [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
  version: 2

```

Procederemos a configurar y nos debe quedar así.



```

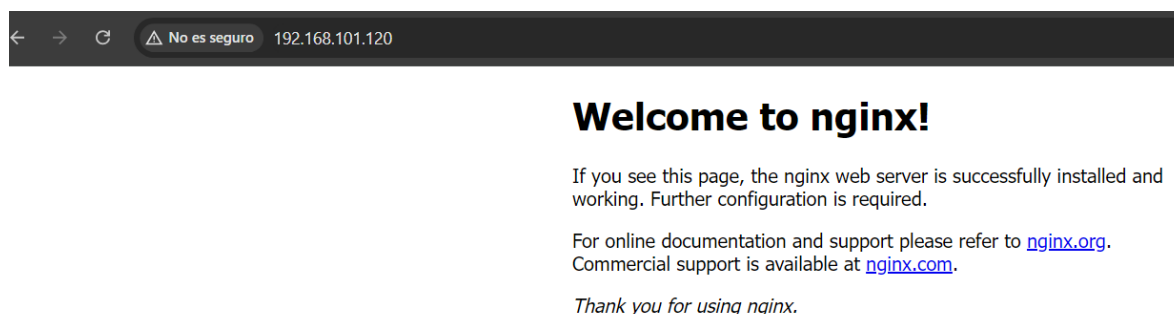
alejo [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 6.2 /etc/netplan/00-installe
# This is the network config written by 'subiquity'
network:
  version: 2
  renderer: networkd
  ethernet:
    enp0s3:
      dhcp4: true
      addresses: [192.168.101.120/24]
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
      routes:
        - to: default
          via: 192.168.101.1
  version: 2
  
```

Aplicaremos los cambios con el comando

```

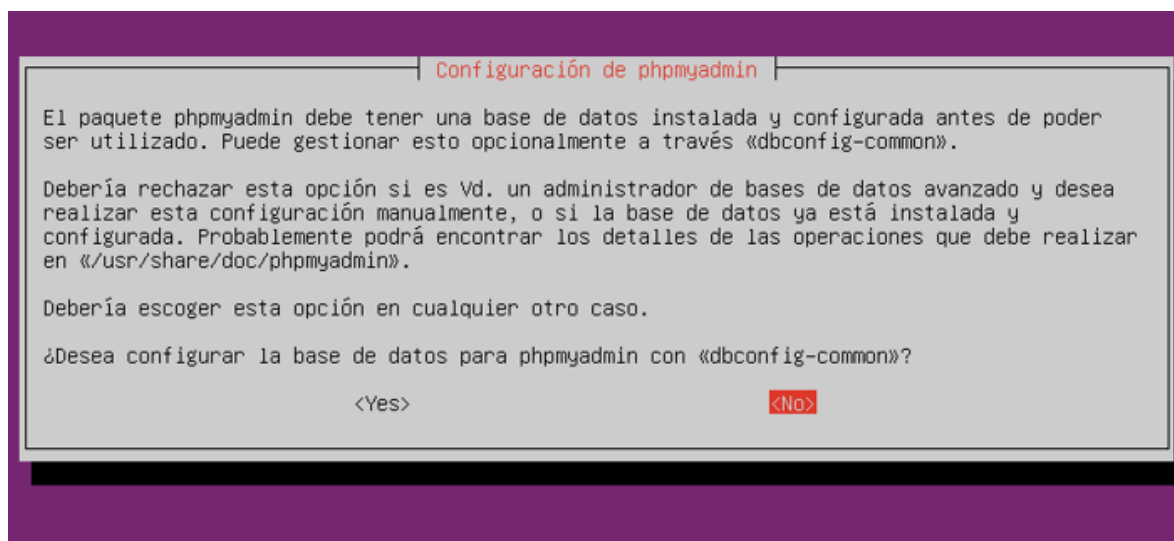
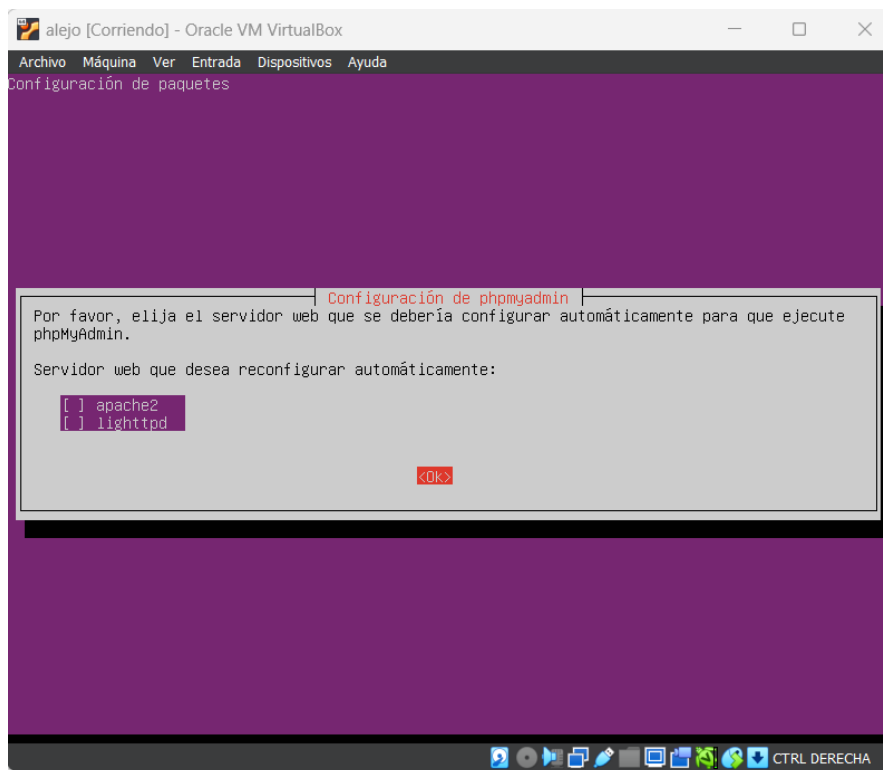
root@alejo:/home/alejo# netplan apply
root@alejo:/home/alejo# apt install mysql-server -y
  
```

Abriremos el navegador y nos deberá cargar la vista de bienvenida de nginx.



Instalaremos los programas php myadmin y mysql que nos permitirán la conexión con nuestra base de datos.

```
root@alejo:/home/alejo# apt install php-fpm php-mysql -y
```



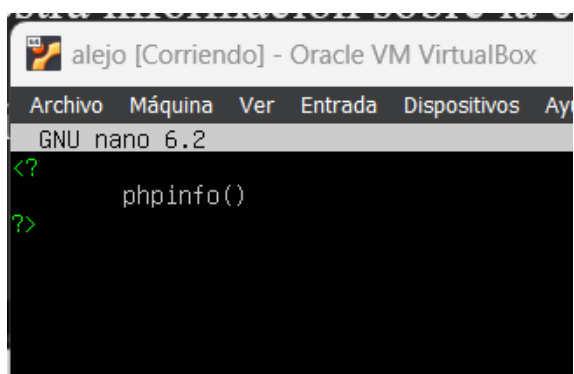
Verificaremos su estado para saber que se instaló de manera correcta

```
root@alejo:/home/alejo# systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset:
   Active: active (running) since Wed 2024-07-24 20:10:43 UTC; 40s ago
     Process: 6993 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=e
   Main PID: 7001 (mysqld)
    Status: "Server is operational"
     Tasks: 38 (limit: 6364)
    Memory: 365.5M
       CPU: 2.164s
    CGroup: /system.slice/mysql.service
           └─7001 /usr/sbin/mysqld

Jul 24 20:10:42 alejo systemd[1]: Starting MySQL Community Server...
Jul 24 20:10:43 alejo systemd[1]: Started MySQL Community Server.
lines 1-14/14 (END)
```

Creamos el archivo de la información de php para comprobar que este bien.

```
root@alejo:/home/alejo# nano info.php
```



The screenshot shows a web browser window titled "alejo [Corriendo] - Oracle VM VirtualBox". The browser displays the output of the `phpinfo()` function, which is rendered in a dark-themed nano editor interface. The nano editor shows the PHP version and other configuration details. The browser's address bar and menu bar are visible at the top.



## Creación de usuario phpmyadmin usando mysql.

**Ingresaremos a mysql para la creación de nuestro usuario autorizado para ingresar a phpmyadmin.**

```
root@alejo:/home/alejo# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.37-0ubuntu0.22.04.3 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

```
mysql> CREATE USER 'alejo'@'localhost' IDENTIFIED BY 'alejo';
Query OK, 0 rows affected (0,02 sec)

mysql>
```

Establecer privilegios.

```
mysql> GRANT ALL PRIVILEGES ON *.* TO 'alejo'@'localhost' WITH GRANT OPTION;
Query OK, 0 rows affected (0,02 sec)

mysql>
```

```
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,01 sec)
```

```
mysql> SELECT user, HOST FROM mysql.user;
+-----+-----+
| user          | HOST          |
+-----+-----+
| alejo         | localhost     |
| debian-sys-maint | localhost     |
| mysql.infoschema | localhost     |
| mysql.session  | localhost     |
| mysql.sys      | localhost     |
| root          | localhost     |
+-----+-----+
6 rows in set (0,00 sec)

mysql> exit
```

### Configuración de php myadmin

```
root@alejo:/home/alejo# nano /etc/nginx/snippets/phpmyadmin.conf_
```

Abre en blanco y se debe escribir esto este comando nos permitirá el correcto uso de mph el comando **test** es útil para realizar pruebas lógicas y condicionales en scripts de shell y automatización de tareas.

```
location /phpmyadmin {
    root /usr/share/;
    index index.php index.html index.htm;
    location ~ ^/phpmyadmin/(.+\.php)$ {
        try_files $uri =404;
        root /usr/share/;
        fastcgi_pass unix:/run/php/php8.1-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        include /etc/nginx/fastcgi_params;
    }

    location ~* ^/phpmyadmin/(.+\. (jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
        root /usr/share/;
    }
}
```

### Aceptar a php desde nginx

```
root@alejo:/home/alejo# nano /etc/nginx/sites-available/default
```

Se establece la conexión con php

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;
    include snippets/phpmyadmin.conf;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

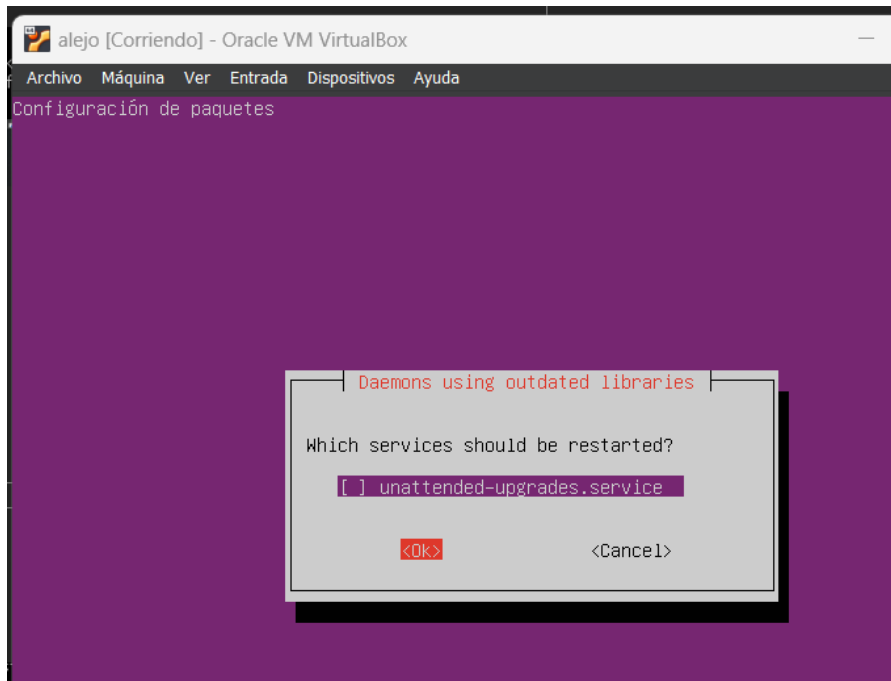
    # Add index.php to the list if you are using PHP
    index index.html index.htm index.php index.nginx-debian.html;

    server_name _;

    location / {
        # First attempt to serve request as file, then
        # as directory, then fall back to displaying a 404.
    }
}
```

Instalación de nodejs para manejar el desarrollo de nuestra api mas adelante.

```
root@alejo:/home/alejo# apt install nodejs-y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
```



Versión instalada se instalará una versión por defecto

```
root@alejo:/home/alejo# node -v
v12.22.9
```

Esta no funcionara ya que esta desactualizada instalar la mas nueva posible

```
root@alejo:/home/alejo# curl -sL https://deb.nodesource.com/setup_20.x -o nodesource_setup.sh
```

```
root@alejo:/home/alejo# bash nodesource_setup.sh
```

```
root@alejo:/home/alejo# apt-get install nodejs -y
```

```
root@alejo:/home/alejo# nodejs -v
v20.16.0
root@alejo:/home/alejo# _
```

## Instalación de ssh

Esta se utilizara para la creación de conexión con cliente

```
root@alejo:/home/alejo# apt install -y ssh
```

Verificar su estado

```
root@alejo:/home/alejo# systemctl status ssh
• ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-07-26 01:47:10 UTC; 15h ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 911 (sshd)
      Tasks: 1 (limit: 6364)
     Memory: 4.2M
        CPU: 1.225s
    CGroup: /system.slice/ssh.service
            └─911 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

jul 26 13:47:38 alejo sshd[39744]: Accepted password for alejo from 10.193.157.95 port 50546 ssh2
jul 26 13:47:38 alejo sshd[39744]: pam_unix(sshd:session): session opened for user alejo(uid=1000)
jul 26 14:36:30 alejo sshd[43792]: Accepted password for alejo from 10.193.144.230 port 52448 ssh2
jul 26 14:36:30 alejo sshd[43792]: pam_unix(sshd:session): session opened for user alejo(uid=1000)
jul 26 14:37:03 alejo sshd[43983]: Accepted password for alejo from 10.193.144.230 port 52496 ssh2
jul 26 14:37:03 alejo sshd[43983]: pam_unix(sshd:session): session opened for user alejo(uid=1000)
jul 26 14:47:11 alejo sshd[44262]: Accepted password for alejo from 10.193.144.230 port 52207 ssh2
jul 26 14:47:11 alejo sshd[44262]: pam_unix(sshd:session): session opened for user alejo(uid=1000)
jul 26 15:00:33 alejo sshd[44500]: Accepted password for alejo from 10.193.144.230 port 52821 ssh2
jul 26 15:00:33 alejo sshd[44500]: pam_unix(sshd:session): session opened for user alejo(uid=1000)

root@alejo:/home/alejo# apt install -y ssh
```

## Configuración de dominio.

### Instalación de bind9.

Bind9 nos ayudara para la configuración de dominio.

```
root@alejo:/home/alejo# apt install bind9 bind9-utils nano_
```

Activar bind9 con el comando

```
root@alejo:/home/alejo# ufw allow bind9
Rule added
Rule added (v6)
root@alejo:/home/alejo# _
```

este comando te permite acceder y modificar la configuración de **bind9** para ajustar su comportamiento según tus necesidades específicas.

```
root@alejo:/home/alejo# nano/etc/bind/named.conf.options_
```

```
GNU nano 6.2 /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    allow-query { any; };
    listen-on { any; };
    allow-recursion { any; };
    forwarders {
        8.8.8.8;
        8.8.4.4;
    };

    // forwarders {
    //     0.0.0.0;
    // };

    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys. See https://www.isc.org/bind-keys
    //=====
    #dnsssec-validation auto;

    dnssec-validation no;

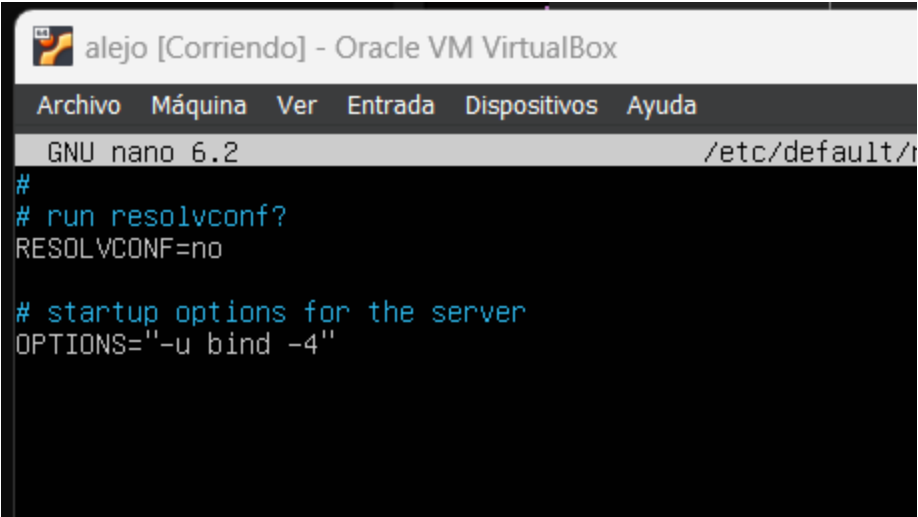
    #listen-on-v6 { any; };

    auth-nxdomain no; # conform to RFC1035
    listen-on-v6 { any; };
};
```

Ingresaremos a configurar el uso de IPv4 en ves IPv6 esto lo haremos ingresando a la siguiente ruta:

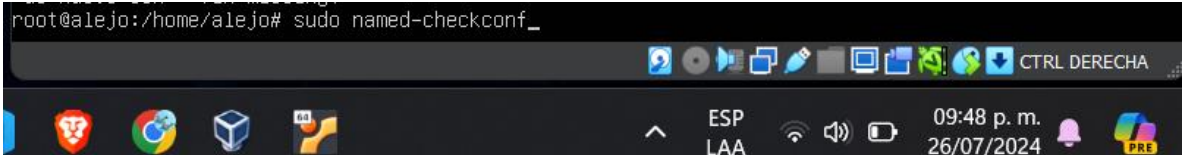
```
root@alejo:/home/alejo# nano /etc/default/named
```

Donde se configurará el número 4.



```
alejo [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 6.2 /etc/default/named
#
# run resolvconf?
RESOLVCONF=no
# startup options for the server
OPTIONS="-u bind -4"
```

Verificaremos que este correcto con el comando si este no muestra nada se abra configurado correctamente



```
root@alejo:/home/alejo# sudo named-checkconf_
```

Verificaremos su estado.

```

root@alejo:/home/alejo# sudo named-checkconf
root@alejo:/home/alejo# systemctl restart bind9
root@alejo:/home/alejo# systemctl status bind9
● named.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-07-24 23:59:01 UTC; 18s ago
     Docs: man:named(8)
  Process: 21627 ExecStart=/usr/sbin/named $OPTIONS (code=exited, status=0/SUCCESS)
    Main PID: 21628 (named)
      Tasks: 6 (limit: 6364)
     Memory: 5.3M
        CPU: 80ms
    CGroup: /system.slice/named.service
            └─21628 /usr/sbin/named -u bind -4

jul 24 23:59:01 alejo named[21628]: configuring command channel from '/etc/bind/rndc.key'
jul 24 23:59:01 alejo named[21628]: command channel listening on 127.0.0.1#953
jul 24 23:59:01 alejo named[21628]: managed-keys-zone: loaded serial 2
jul 24 23:59:01 alejo named[21628]: zone 0.in-addr.arpa/IN: loaded serial 1
jul 24 23:59:01 alejo named[21628]: zone localhost/IN: loaded serial 2
jul 24 23:59:01 alejo named[21628]: zone 255.in-addr.arpa/IN: loaded serial 1
jul 24 23:59:01 alejo named[21628]: zone 127.in-addr.arpa/IN: loaded serial 1
jul 24 23:59:01 alejo named[21628]: all zones loaded
jul 24 23:59:01 alejo named[21628]: running
jul 24 23:59:01 alejo systemd[1]: Started BIND Domain Name Server.
root@alejo:/home/alejo#

```

Ingresaremos al archivo de configuración de zonas

```

run `npm fund` for details
root@alejo:/home/alejo# nano /etc/bind/named.conf.local_

```

Nos abrirá de esta manera y debemos configurar lo que necesitemos.

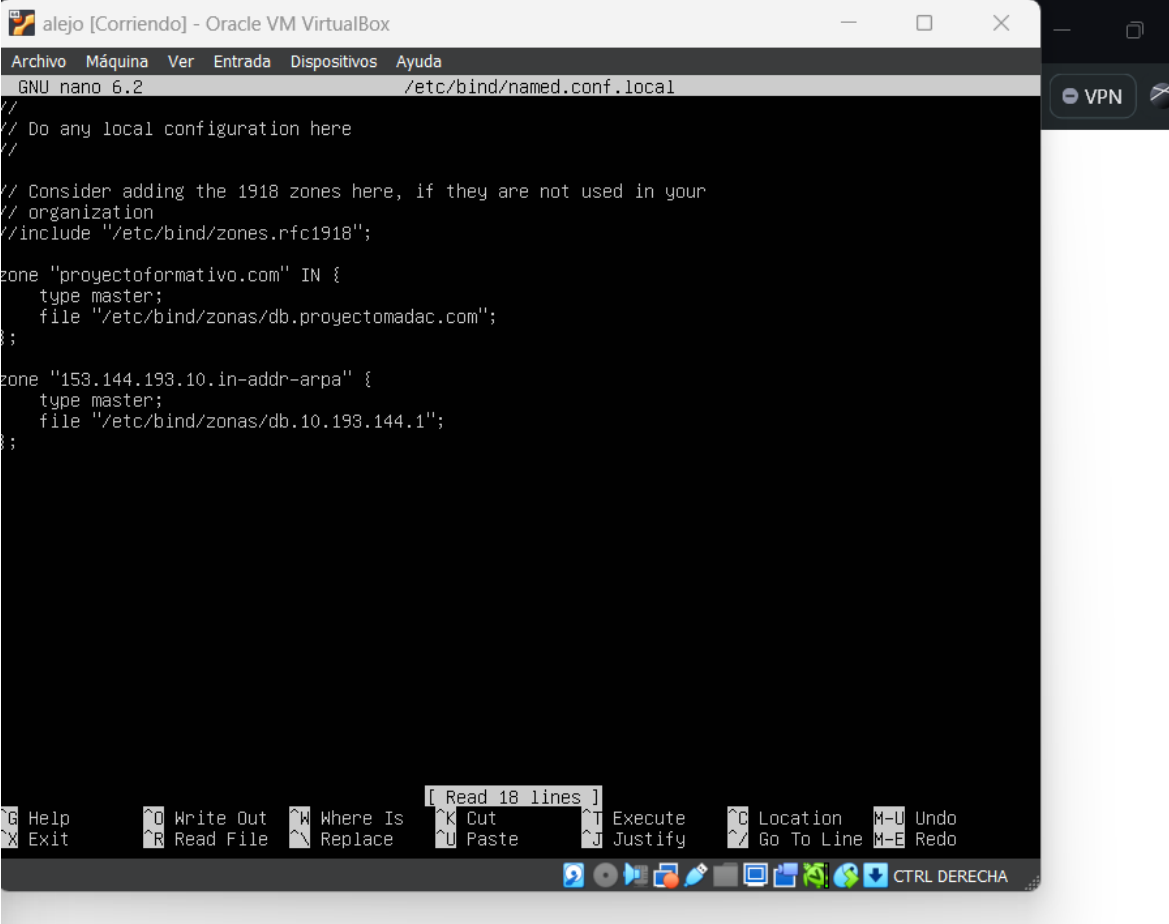
```

alejo [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 6.2 /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

```

Se creará el nombre del dominio que usaremos el archivo a donde estará dirigido y la configuración de la ruta inversa.





```
alejo [Corriendo] - Oracle VM VirtualBox
GNU nano 6.2 /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "proyectoformativo.com" IN {
    type master;
    file "/etc/bind/zonas/db.proyectomadac.com";
};

zone "153.144.193.10.in-addr-arpa" {
    type master;
    file "/etc/bind/zonas/db.10.193.144.1";
};

[ Read 18 lines ]
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
CTRL DERECHA
```

Configurar las zonas directa e inversa

- Zona directa sudo nano /etc/bind/zonas/db.proyectoformativo.com

```

GNU nano 6.2 /etc/bind/zonas/db.projetoformativo.com
$TTL      86400
@          IN      SOA      ns.projetoformativo.com. admin.projetoformativo.com. (
                                2024072401 ; Serial
                                3600       ; Refresh
                                1800       ; Retry
                                1209600   ; Expire
                                86400 )    ; Negative Cache TTL

; Name servers
@          IN      NS       ns.projetoformativo.com.

; A records for name servers
ns         IN      A        192.168.101.110

; A records for the domain
@          IN      A        192.168.101.110

```

Zona inversa.

```

alejo [Corriendo] - Oracle VM VirtualBox
GNU nano 6.2 /etc/bind/zonas/db.192.168.101
$TTL      604800
@          IN      SOA      ns1.projetoformativo.com. admin.projetoformativo.com. (
                                2024072301 ; Serial
                                604800     ; Refresh
                                86400      ; Retry
                                2419200   ; Expire
                                604800 )    ; Negative Cache TTL

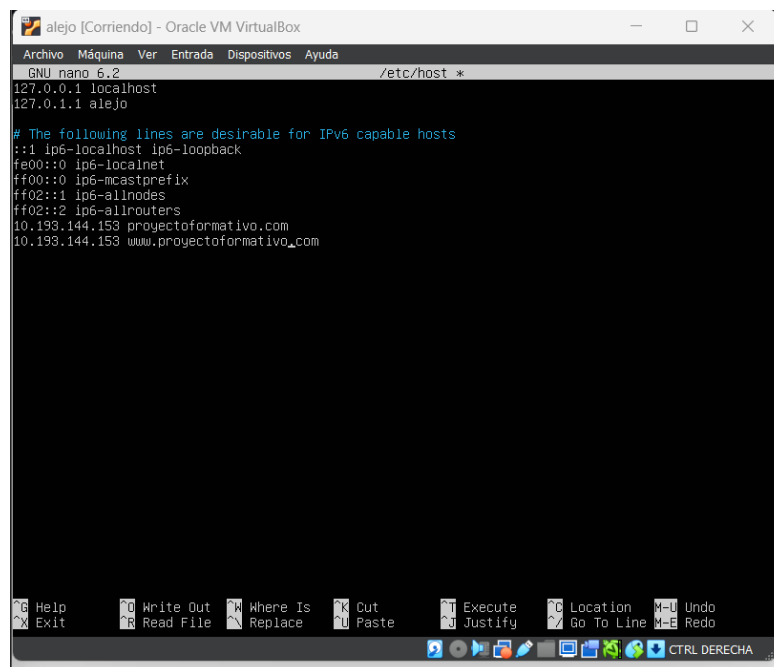
; Registros NS
@          IN      NS       ns1.projetoformativo.com.
@          IN      NS       ns2.projetoformativo.com.

; Registros PTR
9         IN      PTR       proyectoformativo.com.

```

## Configuración del host para la conexión

```
root@alejo:/home/alejo# nano /etc/host
root@alejo:/home/alejo# nano /etc/host
```



## Check de la zona

Este paso se hace para la revisión de las configuraciones del nombre

```
root@alejo:/home/alejo# sudo named-checkzone proyectomadac.com /etc/bind/zonas/db.proyectomadac.com
zone proyectomadac.com/IN: loaded serial 1
OK
root@alejo:/home/alejo#
```

```
root@alejo:/home/alejo# sudo named-checkzone 120.101.168.192.in-addr.arpa /etc/bind/zonas/db.192.
.101
zone 120.101.168.192.in-addr.arpa/IN: loaded serial 20210222
OK
root@alejo:/home/alejo#
```

Luego de esto debemos reiniciar el servicio para que tome los cambios.

```
root@alejo:/home/alejo# sudo systemctl restart bind9
root@alejo:/home/alejo#
```

## Configurar

```
root@alejo:/home/alejo# nano /etc/resolv.conf_
```

Para que tome la URL de nuestro usuario ip.

```
GNU nano 6.2 /etc/resolv.conf
# Do not edit.
#
# This file might be symlinked as /etc/resolv.conf. If you're looking at
# /etc/resolv.conf and seeing this text, you have followed the symlink.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs should typically not access this file directly, but only
# through the symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a
# different way, replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 192.168.101.110
nameserver 127.0.0.1
```

Para saber si quedo bien configurado.

```

root@alejo:/home/alejo# dig www.proyectomadac.com

; <<> DiG 9.18.28-0ubuntu0.22.04.1-Ubuntu <<> www.proyectomadac.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 1042
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.proyectomadac.com.      IN      A

;; AUTHORITY SECTION:
com.      900      IN      SOA     a.gtld-servers.net. nstld.verisign-grs.com. 1
95 1800 900 604800 86400

;; Query time: 247 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Thu Jul 25 04:15:12 UTC 2024
;; MSG SIZE rcvd: 123

root@alejo:/home/alejo#

```

```

root@alejo:/home/alejo# sudo ufw enable
Firewall is active and enabled on system startup
root@alejo:/home/alejo# _

```

Habilitar puerto 53 estandarizado para el trafico de datos

```

root@alejo:/home/alejo# sudo ufw allow 53
Rule added
Rule added (v6)
root@alejo:/home/alejo#

```

Puerto para el SSH

```

root@alejo:/home/alejo# sudo ufw allow 22/tcp
Rule added
Rule added (v6)
root@alejo:/home/alejo#

```

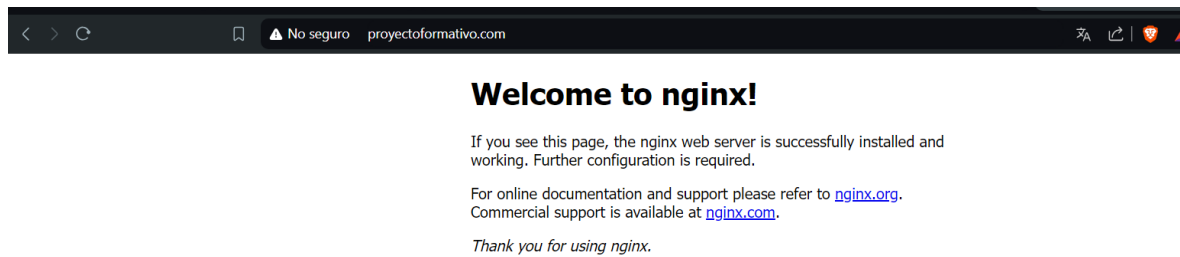
```

root@alejo:/home/alejo# ufw status
Status: active

To Action From
--
Nginx HTTP ALLOW Anywhere
Bind9 ALLOW Anywhere
53 ALLOW Anywhere
22/tcp ALLOW Anywhere
Nginx HTTP (v6) ALLOW Anywhere (v6)
Bind9 (v6) ALLOW Anywhere (v6)
53 (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)

```

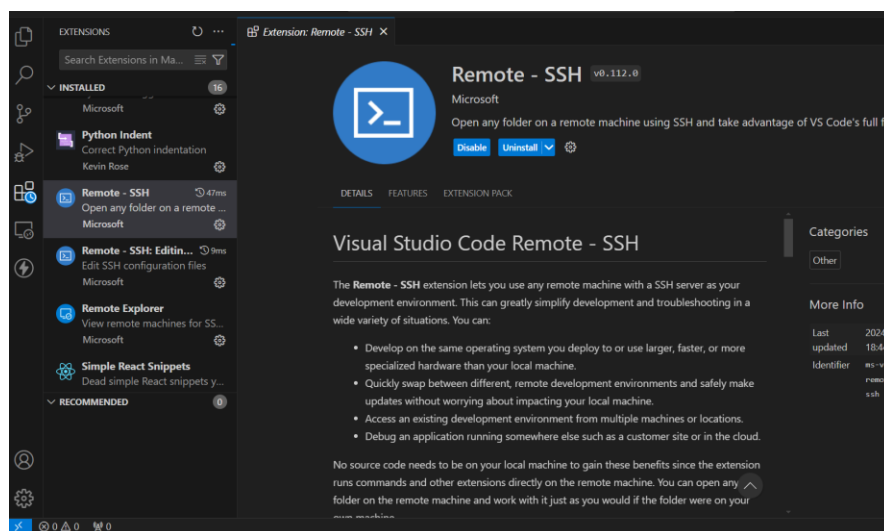
En el navegador verificaremos si quedo de buena manera.



## Despliegue del proyecto

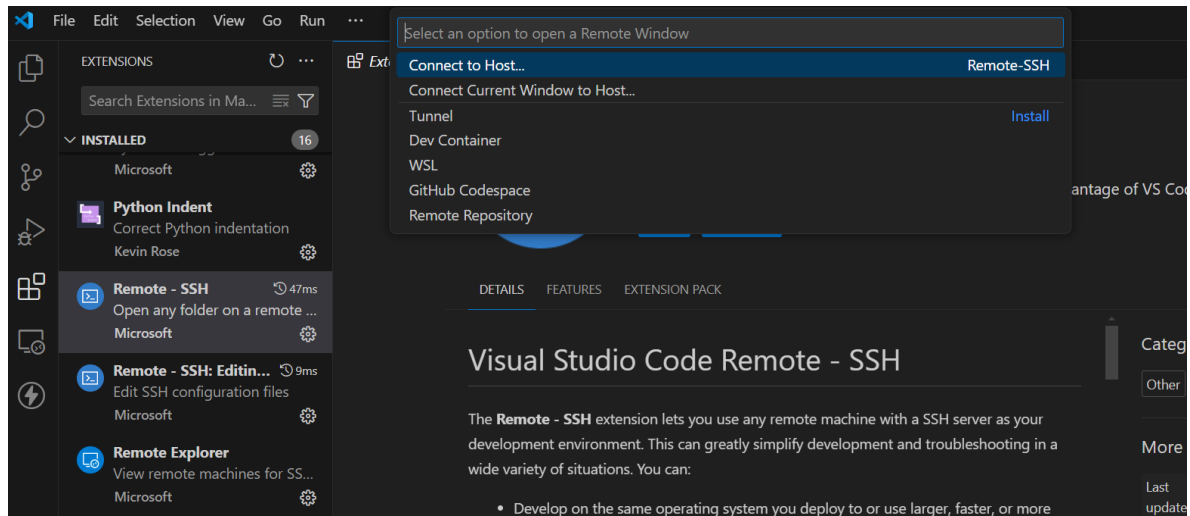
En visual estudio code, usaremos este editor para subir los archivos de nuestro proyecto

Instalamos una dependencia llamada remote SSH de visual estudio code.



Le daremos conectar con nueva host.

Esta extensión nos creada en la parte inferior izquierda un icono donde nos abrirá esta aparte  
donde crearemos la conexión con SSH.

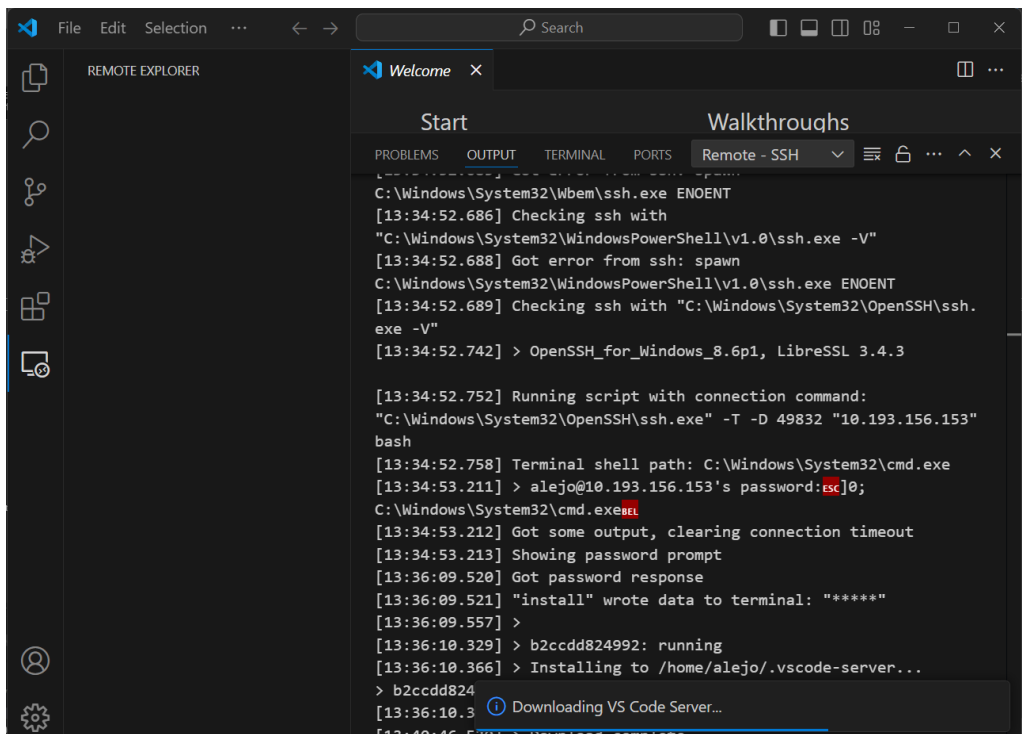


Daremos click en conectar host e ingresaremos el comando ssh [alejo@10.193.156.153](mailto:alejo@10.193.156.153)

Donde alejo es el nombre de usuario de la maquina junto con la URL única creada en la primera configuración.

Nos pedirá la contraseña del usuario la ingresaremos nos pedirá la contraseña y empezará la instalación





```

C:\Windows\System32\Wbem\ssh.exe ENOENT
[13:34:52.686] Checking ssh with
"C:\Windows\System32\WindowsPowerShell\v1.0\ssh.exe -V"
[13:34:52.688] Got error from ssh: spawn
C:\Windows\System32\WindowsPowerShell\v1.0\ssh.exe ENOENT
[13:34:52.689] Checking ssh with "C:\Windows\System32\OpenSSH\ssh.
exe -V"
[13:34:52.742] > OpenSSH_for_Windows_8.6p1, LibreSSL 3.4.3

[13:34:52.752] Running script with connection command:
"C:\Windows\System32\OpenSSH\ssh.exe" -T -D 49832 "10.193.156.153"
bash
[13:34:52.758] Terminal shell path: C:\Windows\System32\cmd.exe
[13:34:53.211] > alejo@10.193.156.153's password:***0;
C:\Windows\System32\cmd.exe
[13:34:53.212] Got some output, clearing connection timeout
[13:34:53.213] Showing password prompt
[13:36:09.520] Got password response
[13:36:09.521] "install" wrote data to terminal: "*****"
[13:36:09.557] >
[13:36:10.329] > b2ccdd824992: running
[13:36:10.366] > Installing to /home/alejo/.vscode-server...
> b2ccdd824
[13:36:10.3 > Downloading VS Code Server...
[13:40:46.5 > Download complete

```

Nos pedirá la contraseña de nuestro usuario y daremos click nos creara la conexión con nuestra maquina aquí lo que haremos será crear una carpeta donde se almacene nuestro backend y frontend

A ellos instalaremos el node modules con el comando npm install y también

Pm2 y correremos para saber si están bien habilitaremos el puerto del backend y el frontend

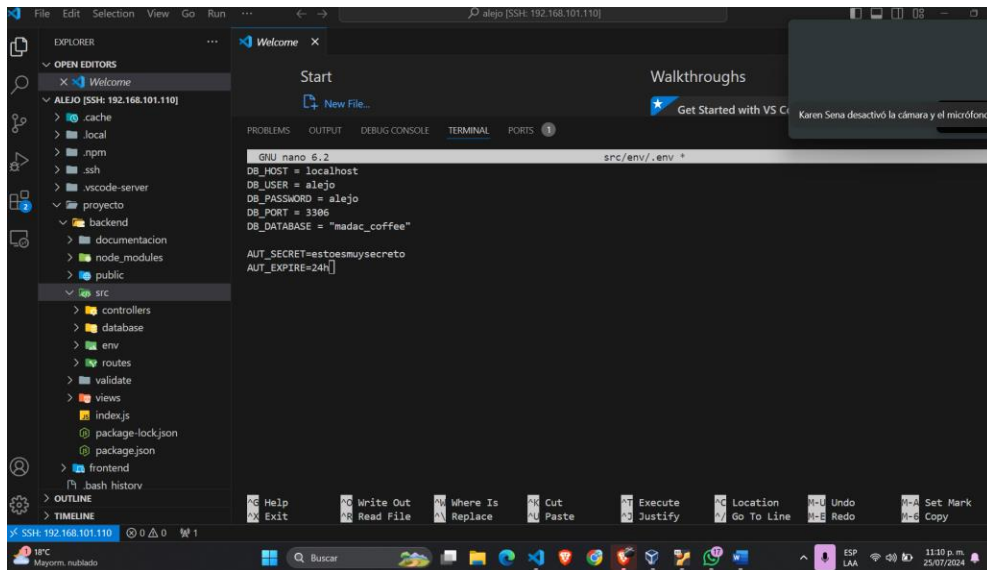


```

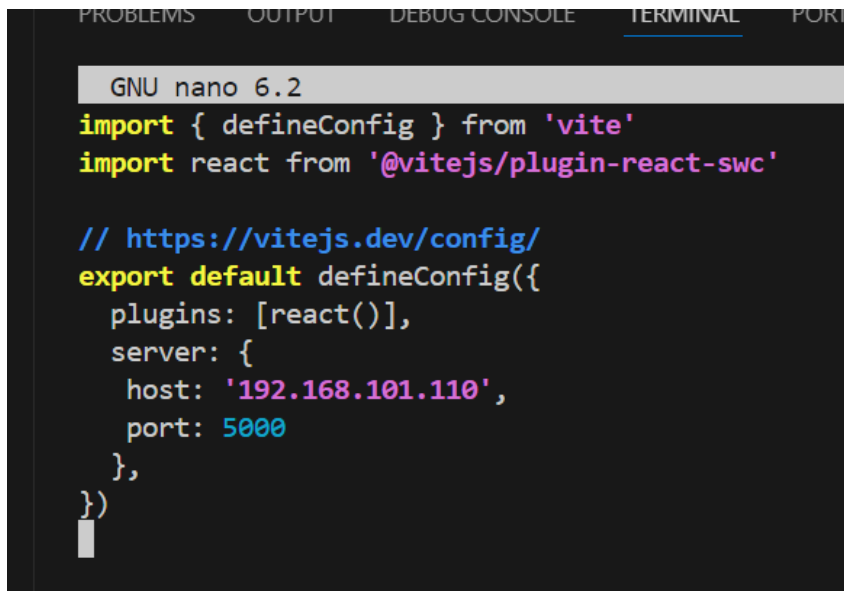
root@alejo:/home/alejo# ufw allow 3310/tcp_
root@alejo:/home/alejo# ufw allow 5510/tcp

```

Procederemos a configurar el archivo .env por lo general se almacena en la carpeta src/env/.env del backend con nuestro usuario y contraseña.



Ahora en frontend configuraremos el archivo vite.config.js



Activación de servicios pm2.

```
pm2 start npm --name 'frontend' -- run dev
```

```

root@alejo:/home/alejo/proyecto/frontend# pm2 start npm --name 'frontend' -- run dev
[PM2] Applying action restartProcessId on app [npm](ids: [ 0 ])
[PM2] [frontend](0) ✓
[PM2] Process successfully started

```

id	name	mode	🔄	status	cpu	memory
0	frontend	fork	75	online	0%	20.6mb

pm2 start index.js --name 'backend'

```

root@alejo:/home/alejo/proyecto/backend# pm2 start index.js --name 'backend'
[PM2] Starting /home/alejo/proyecto/backend/index.js in fork_mode (1 instance)
[PM2] Done.

```

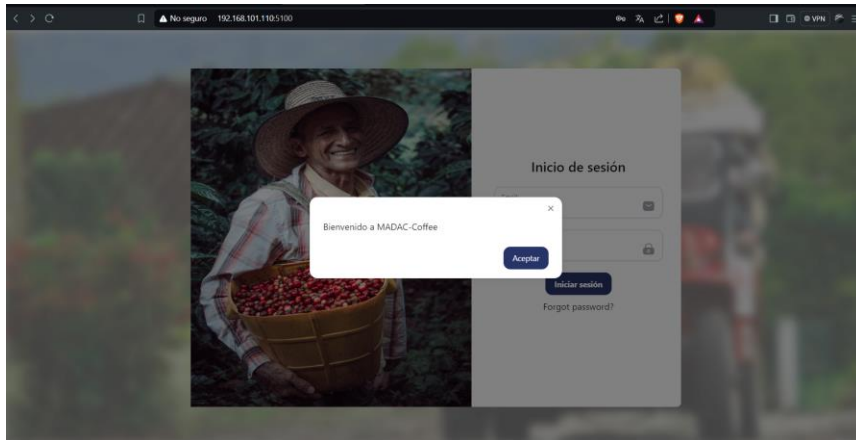
id	name	mode	🔄	status	cpu	memory
2	backend	fork	0	online	0%	7.9mb
0	frontend	fork	75	online	0%	55.8mb
1	index	fork	0	online	0%	63.6mb

Al terminar la activación se mostrará las vistas en el navegador.

<div> <span>&lt;</span> <span>&gt;</span> <span>🔍</span> </div> <div> <span>⚠️ No seguro</span> <span>192.168.101.110:3000/document</span> </div>	
<b>Documentacion API MADAC-Coffee</b>	
<b>Modulo Usuarios</b>	
<b>Requerimiento registrar usuarios</b>	
Numero	1
Nombre del servicio	Registrar usuarios
Url	/usuarios/registrar
Descripción	Permite registrar nuevos usuarios al sistema
Verbo HTTP	Post
Header	Token: token
Usuario	Administrador
Petición body-form-data	Json : { "identificacion": 1080934, "telefono": 3119547883, "nombre": "Aristobulo", "correo_electronico": "aristobulo@gmail.com", "tipo_usuario": 1, "password": "ari123", "estado": 1 }
Respuesta Json	Correcto: Status: 201 Json: { "mensaje": "Usuario registrado con exito!!" }, Incorrecto: Status: 403 Body Json: { "mensaje": "No se pudo registrar el usuario!" }, Error: Status: 500 Body Json: { "mensaje": "Error del servidor" + error }
Programador	Cesar David Carrillo

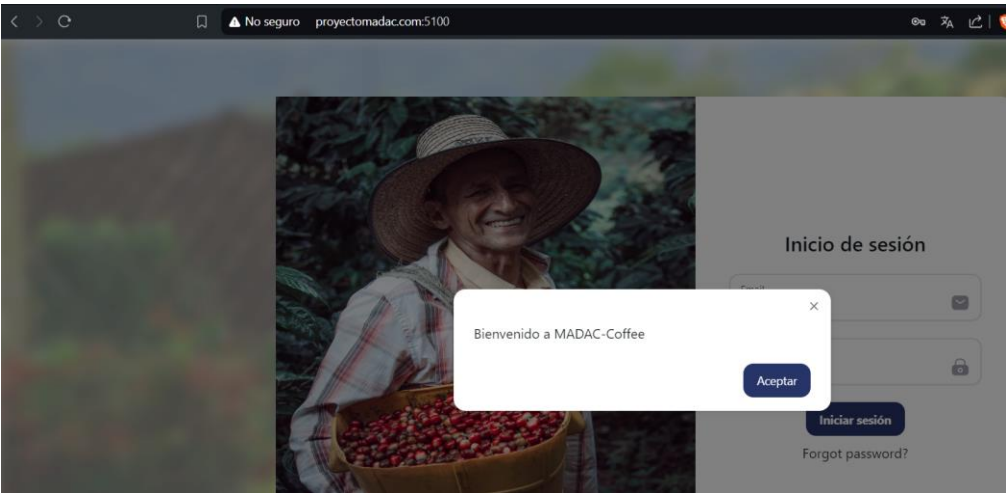
Vista frontend

Para llegar a esta vista debemos de cambiar el localhost por la url de nosotros la propia para poder iniciar sesión esto en todos los archivos que tengas.



Reiniciamos con el comando `systemctl restart bind`) en caso de cambios de deberan realizar los pasos de configuración de los nombres y usar los check y reiniciar para tomar los cambios.

Escribimos el dominio junto con el puerto y debería salirnos tanto backend como frontend



Documentacion API MADAC-Coffee

Modulo Usuarios

Requerimiento registrar usuarios

Numero	1
Nombre del servicio	Registrar usuarios
Url	/usuarios/registrar
Descripción	Permite registrar nuevos usuarios al sistema
Verbo HTTP	Post
Header	Token: token
Usuario	Administrador

## Conclusiones.

En resumen, este documento proporciona una guía paso a paso para crear y configurar una máquina virtual en Ubuntu Server. Al dominar estos conceptos, estarás preparado para desarrollar y desplegar proyectos en un entorno virtualizado de manera eficiente y segura.

## Fuentes de referencia

1. [https://fp.josedomingo.org/sri/6\\_dns/taller1.html](https://fp.josedomingo.org/sri/6_dns/taller1.html)
2. <https://fp.josedomingo.org/sri/pdf/dns.pdf>
3. [https://mundowin.com/como-configurar-bind9-en-ubuntu-server/?expand\\_article=1](https://mundowin.com/como-configurar-bind9-en-ubuntu-server/?expand_article=1)
4. <https://skilly.gitbook.io/dns/v/ejercicios-5/ejer07>
5. <https://skilly.gitbook.io/dns/v/ejercicios-5>
6. <https://skilly.gitbook.io/dns/v/ejercicios-5/ejer03>

<https://trotahosting.com/blog/como-cambiar-los-dns-de-mi-dominio/#:~:text=Iniciamos%20sesi%C3%B3n%20en%20donde%20registramos%20nuestro%20dominio.%20Luego%2C,%E2%80%9Cnameservers%E2%80%9D%20para%20personalizarlos%20e%20ingresaremos%20los%20c%C3%B3digos%20DNS>