



**Tecnológico  
de Monterrey**

Escuela de Ingeniería y Ciencias

## **Cómputo en la Nube**

**Creación de una Máquina Virtual de manera  
Local**

**Armando Bringas Corpus**

**A01200230**

# 1 Introducción

La siguiente práctica tiene como objetivo la implementación de una máquina virtual que nos permita tener entornos diferentes, en este caso en particular un sistema operativo diferente al de la computadora donde se estará virtualizando un sistema operativo Linux el cual tendrá un servidor web. Esto nos permitirá poder analizar las ventajas y desventajas de aplicaciones en la nube sobre una infraestructura local.

## 2 Instalación de Oracle Virtual PC

El primer paso es la instalación de Oracle VirtualBox el cual nos permitirá el generar diferentes entornos donde podremos ejecutar diversos sistemas operativos en una misma computadora.

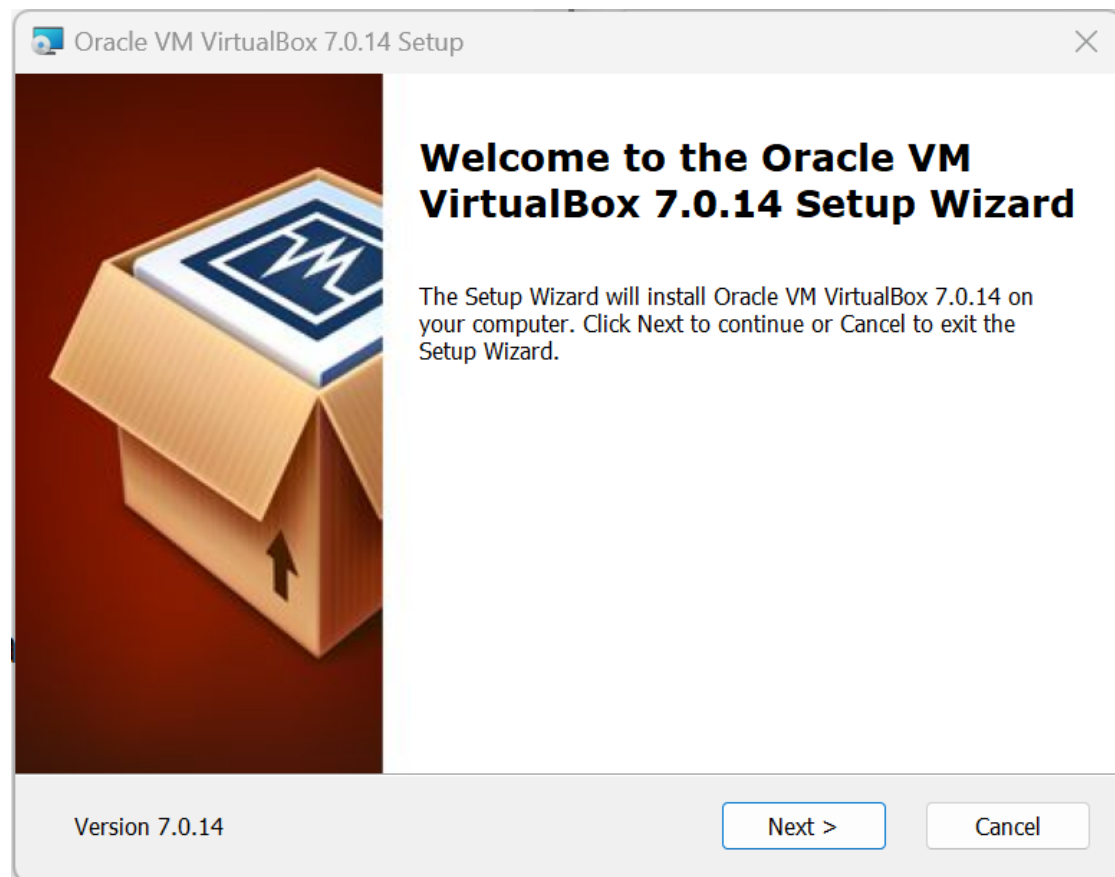


Figura 1: Instalación de VirtualBox 7.0.14

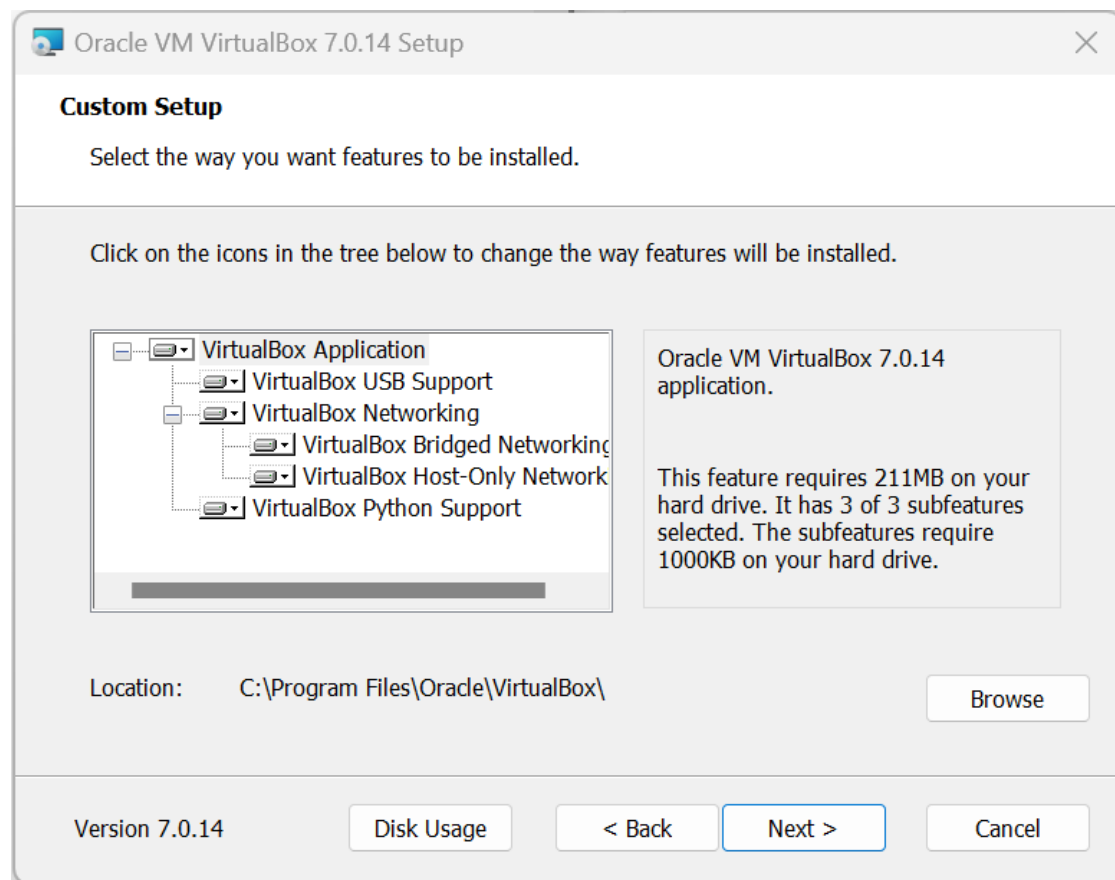


Figura 2: Instalación de VirtualBox 7.0.14

### 3 Creación de la Máquina Virtual de Debian

El siguiente paso es la creación de una Máquina Virtual donde montaremos un pequeño servidor web que estará basado en una distribución de Linux llamada Debian. Las siguientes imágenes muestran parte del proceso de instalación de Debian en la Máquina Virtual. Como configuraciones importantes para la creación de la Máquina Virtual enlistamos las siguientes:

- Nombre: MVDebianLinuxWebServer
- Tipo (Sistema Operativo): Linux
- Versión: Debian (64-bit)
- Tamaño de memoria RAM: 1024 MB

- Disco duro: Disco Virtual (VDI - VirtualBox Disk Image) de 5GB con Reservado Dinámico para permitir su expansión

Las siguientes imágenes 3, 4, son algunas capturas de pantalla que muestran la instalación de Debian a través de una interfaz gráfica (GUI) donde se realizaron diferentes configuraciones como el lenguaje, la zona horaria y localización del sistema, la configuración del servidor y cuentas, gestor de arranque (en este caso GRUB), la partición del disco e instalación de los paquetes.

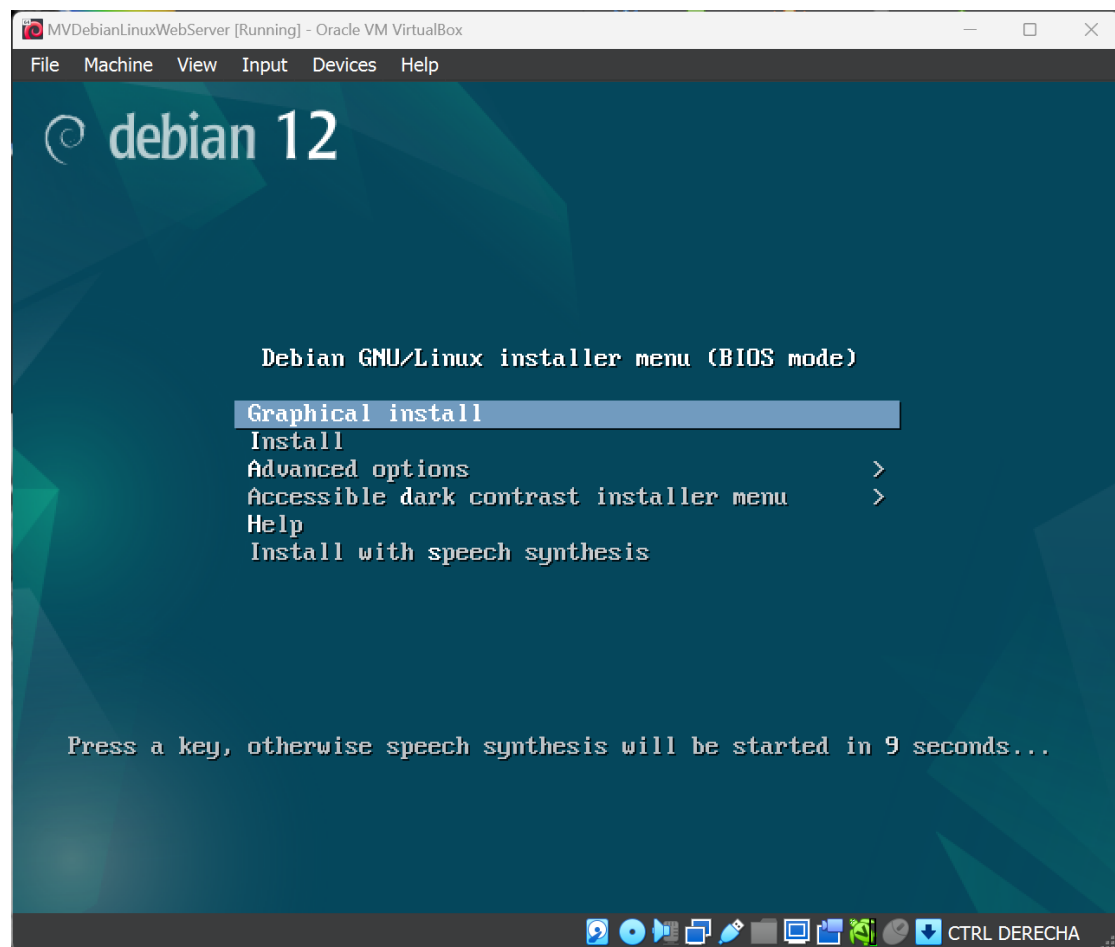


Figura 3: Instalación de Debian 12

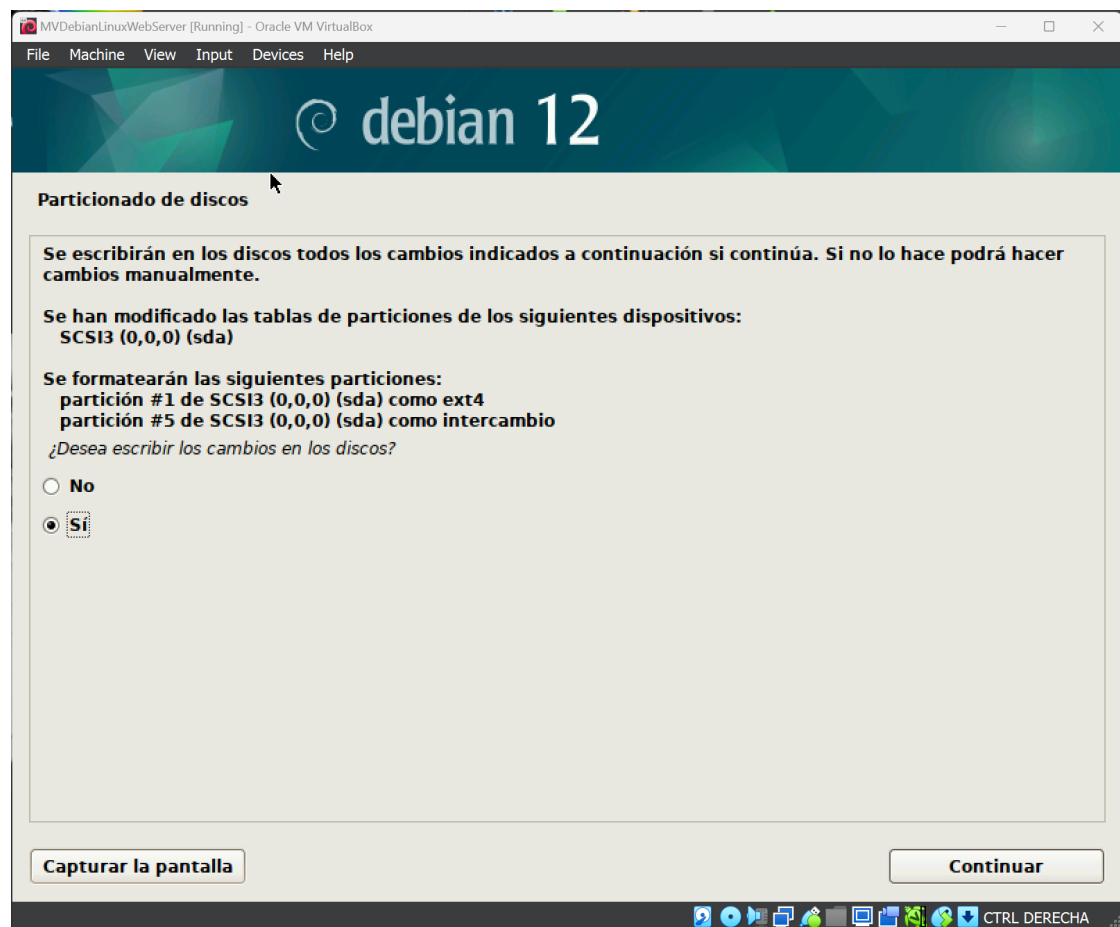


Figura 4: Instalación de Debian 12

Posteriormente, como se muestra en la imagen 5 arrancamos el sistema, el gestor de arranque GRUB automáticamente inicia nuestro sistema operativo Debian y una vez cargado para poder trabajar con el, se ingresa el usuario raíz y la contraseña.

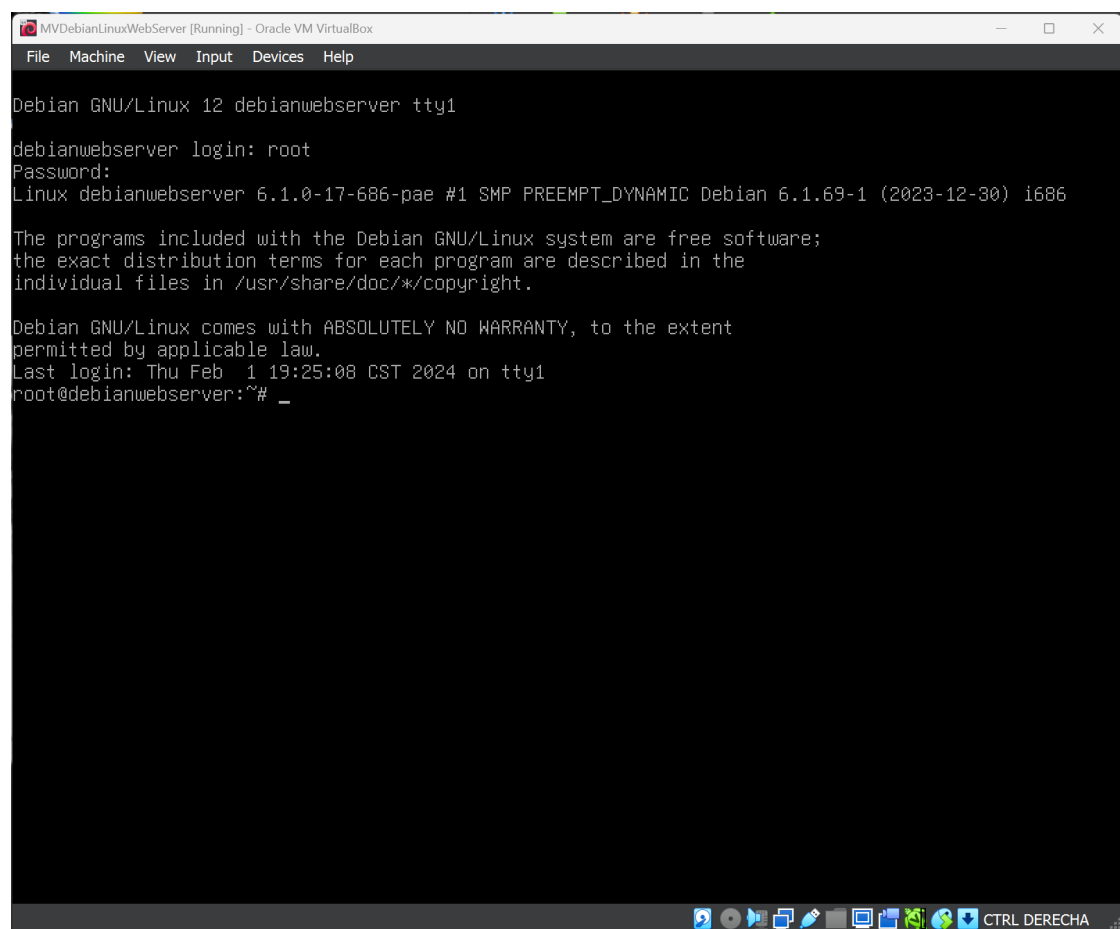
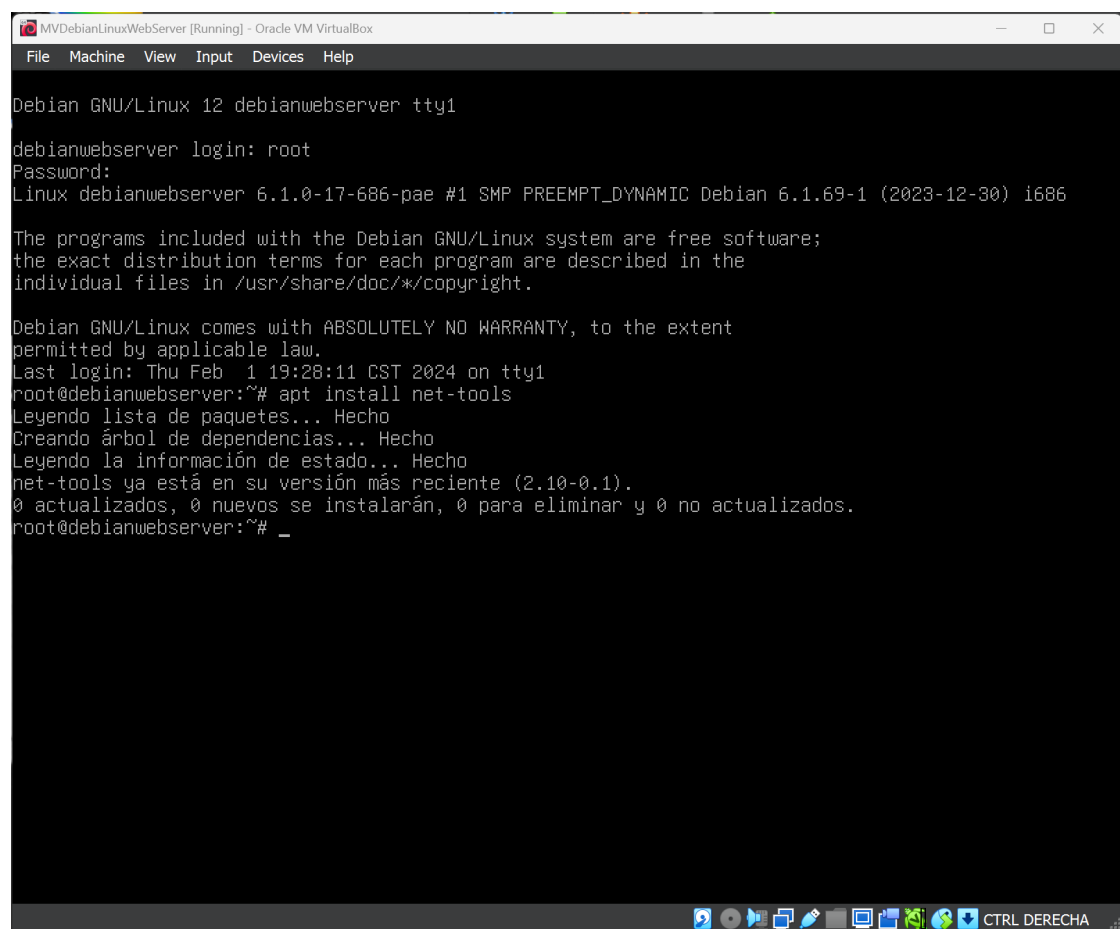


Figura 5: Instalación de Debian 12

## 4 Instalación y configuración de los servicios

Una vez que accedemos al sistema operativo de nuestra máquina virtual se procede a instalar los paquetes que nos permitirán conocer la IP de la Máquina Virtual, en este caso por medio de comando **apt install net-tools**. La imagen 6 muestra que los paquetes solicitados ya están instalados con su versión más reciente, esto es debido que previamente a la toma de captura ya se había ejecutado el comando.



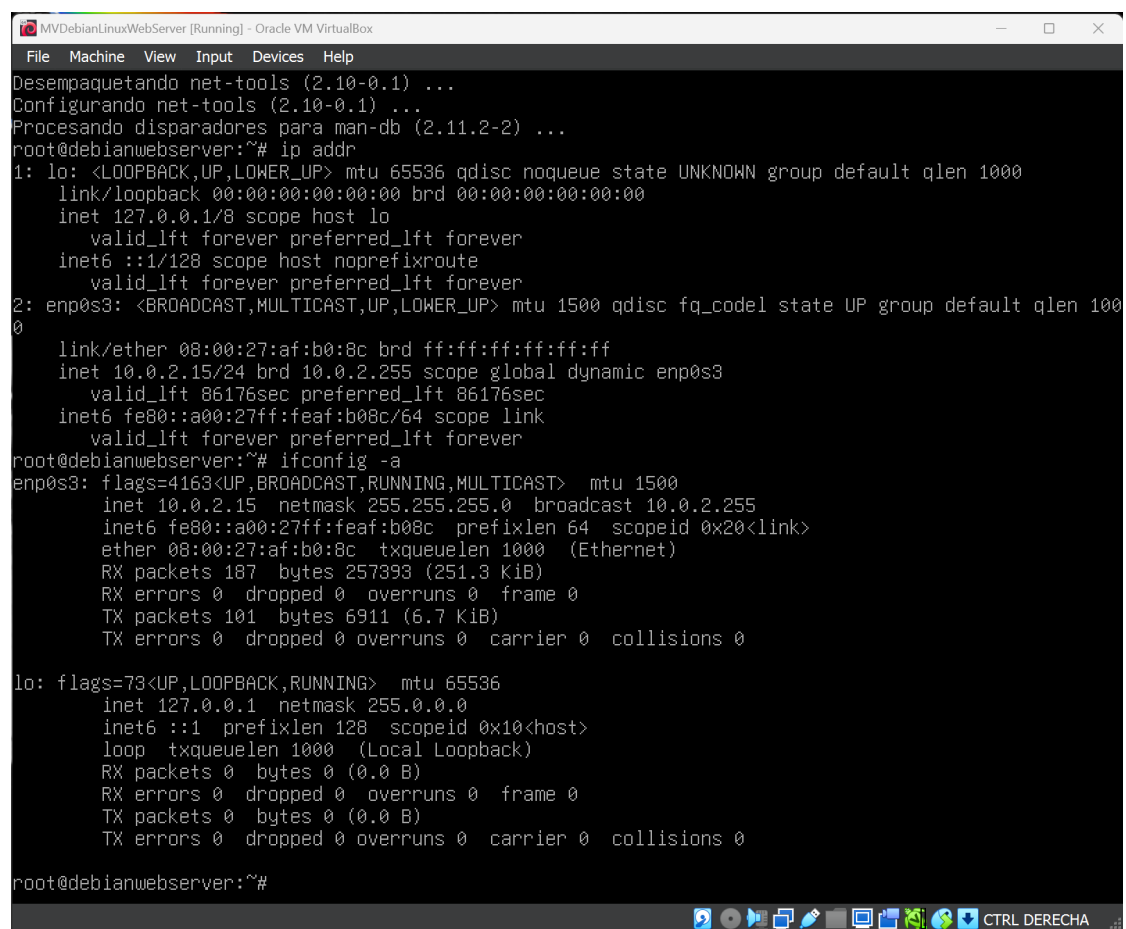
```
Debian GNU/Linux 12 debianwebserver tty1
debianwebserver login: root
Password:
Linux debianwebserver 6.1.0-17-686-pae #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Feb  1 19:28:11 CST 2024 on tty1
root@debianwebserver:~# apt install net-tools
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
net-tools ya está en su versión más reciente (2.10-0.1).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
root@debianwebserver:~# _
```

Figura 6: Configuración de los Servicios

A través del comando **ifconfig -a** podemos desplegar la configuración de la red del sistema operativo que se encuentra ejecutando la Máquina Virtual como se observa en la figura 7, en donde más adelante ocuparemos la dirección IP para poder acceder al pequeño servidor que se estará levantando



```
Desempaquetando net-tools (2.10-0.1) ...
Configurando net-tools (2.10-0.1) ...
Procesando disparadores para man-db (2.11.2-2) ...
root@debianwebserver:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:af:b0:8c brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86176sec preferred_lft 86176sec
    inet6 fe80::a00:27ff:feaf:b08c/64 scope link
        valid_lft forever preferred_lft forever
root@debianwebserver:~# ifconfig -a
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:feaf:b08c prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:af:b0:8c txqueuelen 1000 (Ethernet)
    RX packets 187 bytes 257393 (251.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 101 bytes 6911 (6.7 KiB)
    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 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@debianwebserver:~#
```

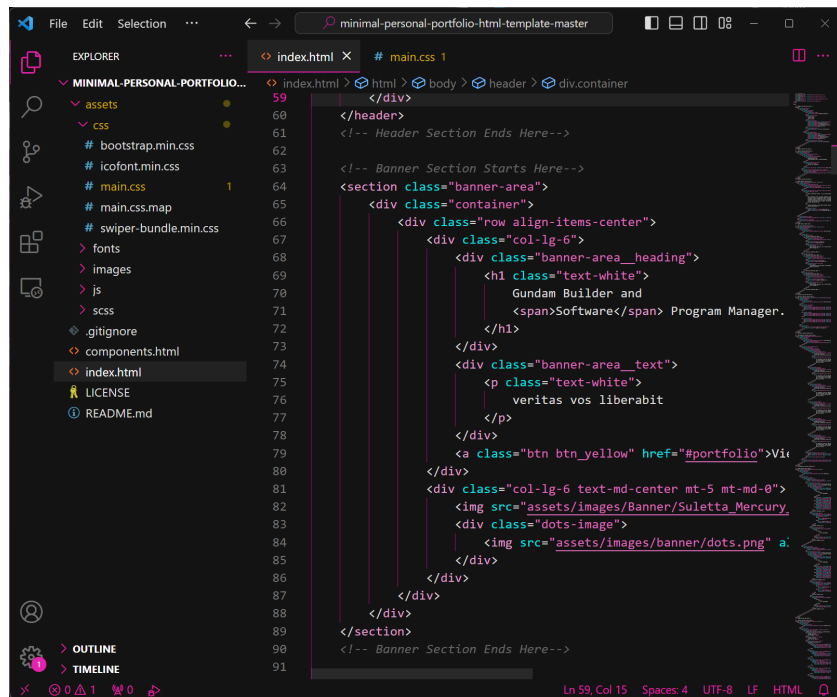
Figura 7: Configuración de los Servicios

## 5 Personalización del sitio web

Descargamos la plantilla web del siguiente sitio <https://htmltemplates.co/free-website-templates/minimal-personal-portfolio-free-portfolio-html-template> y procedemos a realizar algunas modificaciones y personalizaciones, este caso utilizamos Visual Studio Code como se muestra en la figura 8. En la figura ?? se pueden observar las modificaciones que estuvimos realizando como por ejemplo en el color de fondo, la imagen, descripción, etc.

Las modificaciones realizadas se pueden consultar en el siguiente repositorio de Github: [https://github.com/armandoBringas/Cloud\\_Computing/tree/main/M3\\_Virtualizaci%C3%B3n\\_y\\_Contenedores](https://github.com/armandoBringas/Cloud_Computing/tree/main/M3_Virtualizaci%C3%B3n_y_Contenedores)





```
59 </div>
60 </header>
61 <!-- Header Section Ends Here-->
62
63 <!-- Banner Section Starts Here-->
64 <section class="banner-area">
65   <div class="container">
66     <div class="row align-items-center">
67       <div class="col-lg-6">
68         <div class="banner-area_heading">
69           <h1 class="text-white">
70             Gundam Builder and
71             <span>Software</span> Program Manager.
72           </h1>
73         </div>
74         <div class="banner-area_text">
75           <p class="text-white">
76             veritas vos liberabit
77           </p>
78         </div>
79         <a class="btn btn_yellow" href="#portfolio">View Project</a>
80       </div>
81       <div class="col-lg-6 text-md-center mt-5 mt-md-0">
82         
83         <div class="dots-image">
84           
85         </div>
86       </div>
87     </div>
88   </div>
89 </section>
90 <!-- Banner Section Ends Here-->
91
```

Figura 8: Personalización del Sitio Web

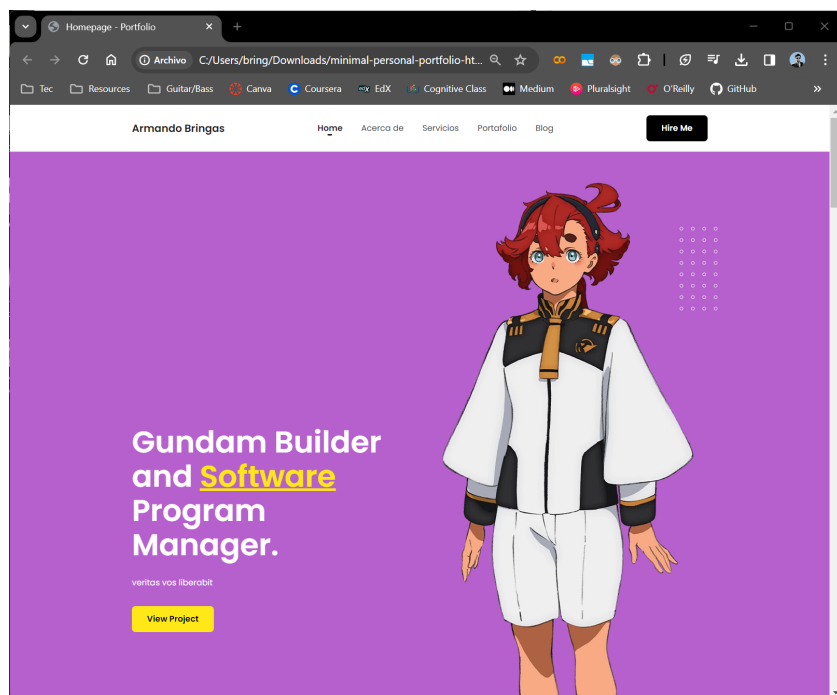


Figura 9: Personalización del Sitio Web

## 6 Carga de Sitio Web en la Máquina Virtual

Posteriormente cargamos nuestro sitio web a la Máquina Virtual, para este caso utilizamos un gestor de archivos para sitios remotos llamado FileZilla, en donde primeramente en VirtualBox habilitamos el permitir la conexión entre nuestra computadora y la máquina virtual. La figura 10 muestra la configuración para establecer la conexión y realizar la transferencia de archivos entre la computadora y la Máquina Virtual.

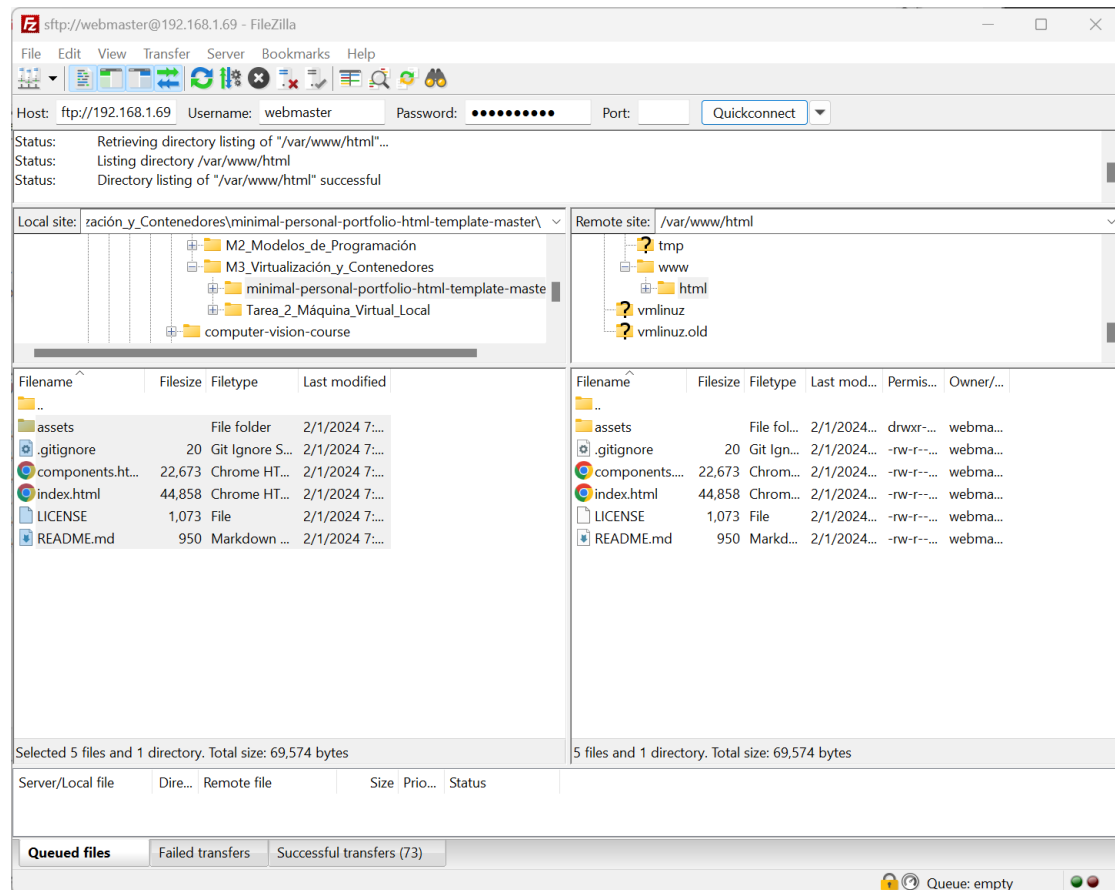
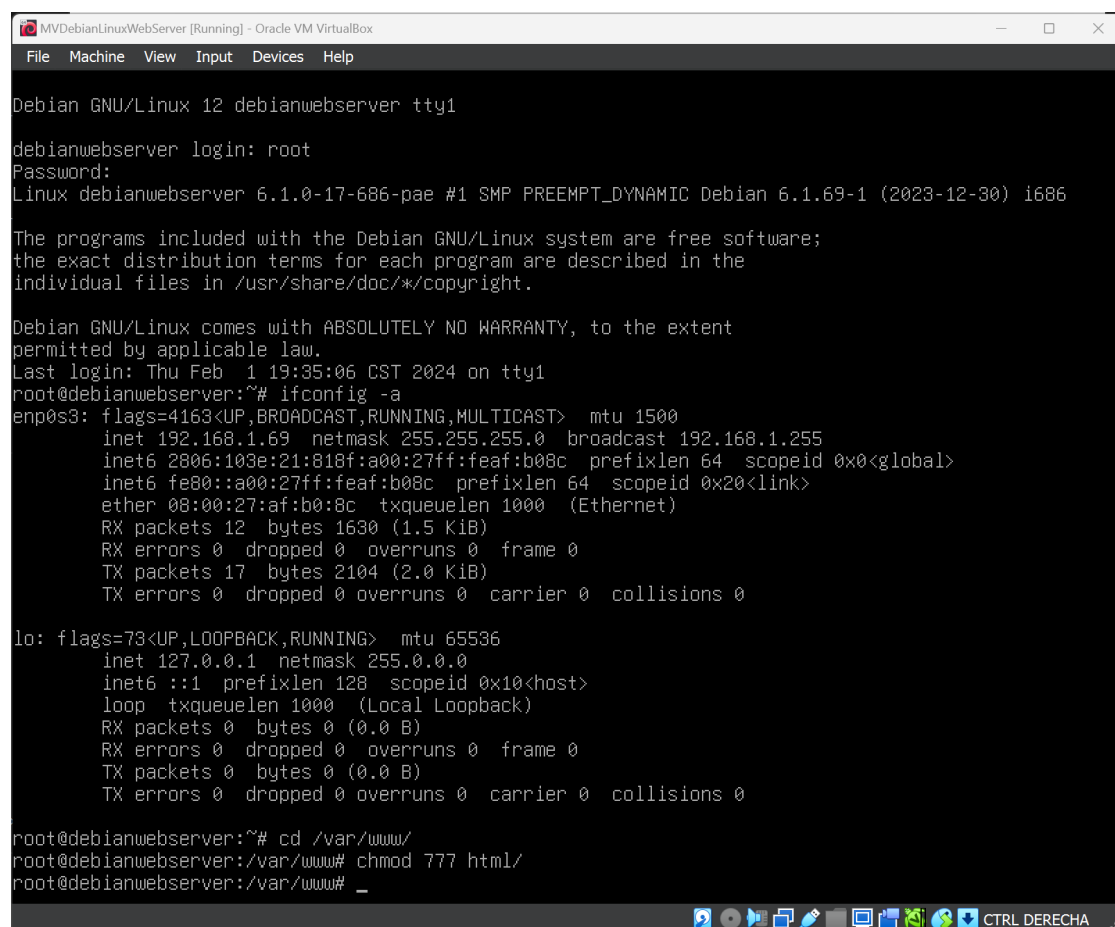


Figura 10: Cargado de sitio web en la Máquina Virtual

Finalmente ejecutamos nuestra Máquina Virtual con nuestro usuario raíz y ejecutamos el comando **chmod 777 html/** para poder escribir en el directorio hml y así poder copiar a través de FileZilla los archivos en el sitio remoto de la Máquina Virtual.



```
Debian GNU/Linux 12 debianwebserver tty1

debianwebserver login: root
Password:
Linux debianwebserver 6.1.0-17-686-pae #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Feb  1 19:35:06 CST 2024 on tty1
root@debianwebserver:~# ifconfig -a
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.1.69  netmask 255.255.255.0  broadcast 192.168.1.255
    inet6 2806:103e:21:818f:a00:27ff:feaf:b08c  prefixlen 64  scopeid 0x0<global>
    inet6 fe80::a00:27ff:feaf:b08c  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:af:b0:8c  txqueuelen 1000  (Ethernet)
    RX packets 12  bytes 1630 (1.5 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 17  bytes 2104 (2.0 KiB)
    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 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@debianwebserver:~# cd /var/www/
root@debianwebserver:/var/www# chmod 777 html/
root@debianwebserver:/var/www# _
```

Figura 11: Cargado de sitio web en la Máquina Virtual

## 7 Descripción de los resultados obtenidos

A través de la IP que se obtiene ejecutando el comando **ifconfig** abrimos un explorador web y procedemos a ingresar la IP para visualizar si se carga la página web y estamos accediendo a nuestro pequeño servidor en la máquina virtual, como se puede ver en la figura 12 la página carga de forma correcta, incluso procedemos a interactuar con la misma para poder ver determinar si la interacción con la misma funciona de forma correcta como se muestra en la figura 13. En este caso es exitoso el resultado obtenido, no se presentan problemas con la renderización de la página web, ni con la interacción con la misma.

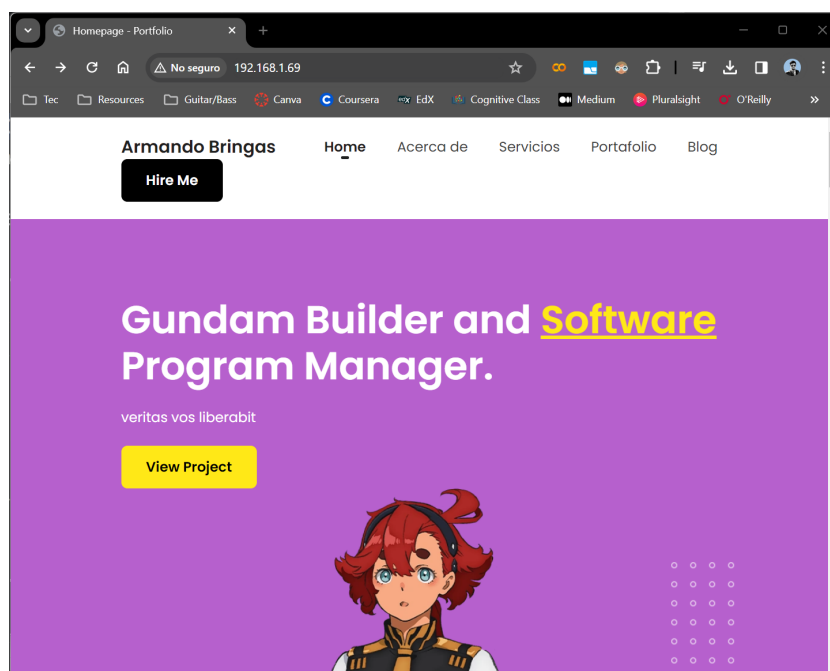


Figura 12: Resultados obtenidos

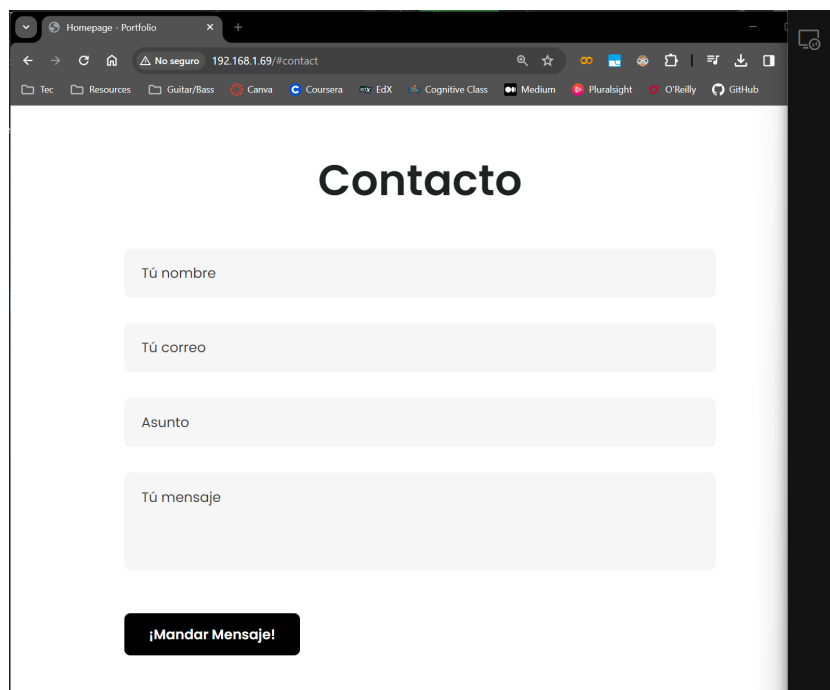


Figura 13: Resultados obtenidos

## 8 Reflexión sobre las Máquinas Virtuales

Como pudimos experimentar con la práctica pudimos experimentar con uno de los servicios más básicos con los que comenzó el cómputo en la nube que nos da como principal ventaja el simular diferentes entornos en un mismo equipo, como pueden ser diferentes sistemas operativos.

Vemos como ventaja que la virtualización de diferentes entornos nos permite experimentar con diferentes sistemas operativos sin tener que hacer una partición de nuestro disco duro de la computadora y hacer configuraciones un tanto tediosas para poder por ejemplo desde el BIOS ejecutar dos sistemas operativos a la vez, también observamos que a través de la virtualización nos dejamos relativamente de preocupar por configurar drivers y la capacidad de asignar los recursos que queramos destinar nuestra Máquina Virtual.

Ahora bien, una de las desventajas que observamos es que el uso de Máquinas Virtuales con el proceso que realizamos ya no se adapte a las prácticas actuales de Cómputo en la Nube, los recursos están limitados a la capacidad de nuestra computadora y en una última instancia nos puede resultar más conveniente el contratar un servicio IaaS a través de algún proveedor donde no nos tengamos que preocupar por ir expandiendo horizontal y verticalmente nuestros recursos y la capacidad de los mismos.

Por último, con anterioridad ya habíamos tenido oportunidad de emplear Máquinas Virtuales, en este caso para la emulación de videojuegos en donde se han generado diferentes distribuciones basadas en Linux para poder ejecutar emuladores de diferentes sistemas de videojuegos, lo cual resulta muy práctico en muchos casos sencillo ya que también hay una comunidad detrás que se ha encargado de optimizar estos sistemas.

Sin embargo, en alguna ocasión tratamos de utilizar una Máquina Virtual para poder ejecutar Tensorflow (un paquete de Deep Learning) cuya instalación es más sencilla en Linux, ya que su última versión ya no es de forma nativa compatible con Windows, en este caso no fue exitosa nuestra instalación ya que no encontramos la manera de configurar la tarjeta gráfica (GPU) para su utilización. Por el contrario, nos fue más práctico el utilizar WSL el cual es un subsistema de de Windows que nos permitió la instalación de Linux, y al momento de instalar TensorFlow en el mismo pudimos configurar exitosamente la tarjeta gráfica.