

Virtualización con VirtualBox

Alumnos:

Diana Falla – diana.falla.cba@gmail.com

Claudio Fiorito – Claudio80.cf@gmail.com

Materia: Arquitectura y Sistemas Operativos

Profesor: Enferrel Ariel

Fecha de Entrega: 05/06/2025



Índice

1. Introducción
2. Marco Teórico
3. Caso Práctico
4. Metodología Utilizada
5. Resultados Obtenidos
6. Conclusiones
7. Bibliografía
8. Anexos
9. Links archivos de entrega

1. Introducción

Este trabajo fue realizado como parte del cierre del primer cuatrimestre de la materia Arquitectura y Sistemas Operativos. Somos estudiantes que estamos empezando en este camino, y elegimos el tema 'Virtualización' porque nos pareció interesante aprender a usar una herramienta como VirtualBox y conocer la forma en que se pueden simular distintos sistemas operativos dentro de una misma computadora.

Nuestro objetivo fue conocer los conceptos básicos de la virtualización, entender para qué sirve, y hacer una prueba concreta de instalación de una máquina virtual desde cero, ejecutando un pequeño programa en Python dentro del sistema operativo.

2. Marco Teórico

La virtualización es una tecnología que permite crear una representación lógica o virtual de un recurso físico, como un sistema operativo, un servidor, un dispositivo de almacenamiento o una red. En el contexto de este trabajo, nos centramos en la virtualización de sistemas operativos, que consiste en ejecutar uno o más sistemas operativos dentro de una máquina física utilizando máquinas virtuales (VMs).

¿Qué es la virtualización?

La virtualización permite que un solo equipo físico ejecute múltiples entornos operativos, cada uno aislado en su propia máquina virtual. Cada máquina virtual actúa como si fuera una computadora independiente, con su propio sistema operativo, recursos y archivos, pero compartiendo el hardware físico con otras VMs a través de un software llamado hipervisor.

¿Cómo funciona?

El componente clave que hace posible la virtualización es el hipervisor, que puede ser de dos tipos:

- Hipervisor Tipo 1 (bare-metal): se instala directamente sobre el hardware físico y controla múltiples sistemas operativos que se ejecutan encima de él.
- Hipervisor Tipo 2 (hosteado): se instala dentro de un sistema operativo anfitrión (host), como una aplicación más, y permite ejecutar sistemas operativos invitados (guest).

El hipervisor asigna y gestiona los recursos (como CPU, RAM y almacenamiento) que cada VM necesita, creando una capa de abstracción entre el hardware físico y las VMs.

Ventajas de la virtualización

- Ahorro de recursos físicos: reduce la necesidad de múltiples equipos físicos.
- Aislamiento: cada VM está separada del resto, lo que permite hacer pruebas o ejecutar software sin riesgo para el sistema principal.
- Flexibilidad: permite ejecutar múltiples sistemas operativos en un solo equipo.
- Facilidad de recuperación: mediante snapshots se puede restaurar una VM a un estado anterior fácilmente.
- Escalabilidad y eficiencia: especialmente útil en entornos empresariales para optimizar el uso de servidores.

Concepto de Sandbox

Un concepto muy relacionado con la virtualización es el de sandbox. Una sandbox es un entorno aislado donde se puede ejecutar software sin que este afecte al resto del sistema. Las máquinas virtuales funcionan como sandboxes en muchos aspectos: permiten experimentar, probar código o instalar aplicaciones sin poner en riesgo el sistema operativo anfitrión. Esto resulta particularmente útil en actividades educativas, desarrollo de software, pruebas de seguridad o incluso para ejecutar software potencialmente malicioso de forma segura.

Relación con la parte práctica

A través de la virtualización podemos simular entornos reales de trabajo sin tener que modificar directamente el equipo físico. En nuestro caso, utilizamos VirtualBox como herramienta para crear una máquina virtual, instalar el sistema operativo Ubuntu 24.04.2 LTS y ejecutar un pequeño programa en Python. Esta experiencia nos permitió aplicar los conceptos aprendidos en la teoría y entender en la práctica cómo se utilizan estas tecnologías.

3. Caso Práctico

Objetivo:

Instalar VirtualBox en un sistema operativo host, configurar una máquina virtual con Ubuntu 24.04.2 LTS como sistema operativo invitado, y probar su funcionamiento creando y ejecutando un programa simple en Python dentro de la máquina virtual.

Pasos:

1. Verificar condiciones para ejecutar un hipervisor de tipo 2
2. Descargar VirtualBox
3. Generamos la primera máquina virtual, instalación de Ubuntu 24.04.2.

Descarga del ISO desde el sitio oficial.

Configuración de la VM en VirtualBox con 3072 MB de RAM y 2 CPU.

5. Verificación de Instalación de Python con los comandos:

```
python3 --version
```

```
sudo apt update
```

```
sudo apt install python3
```

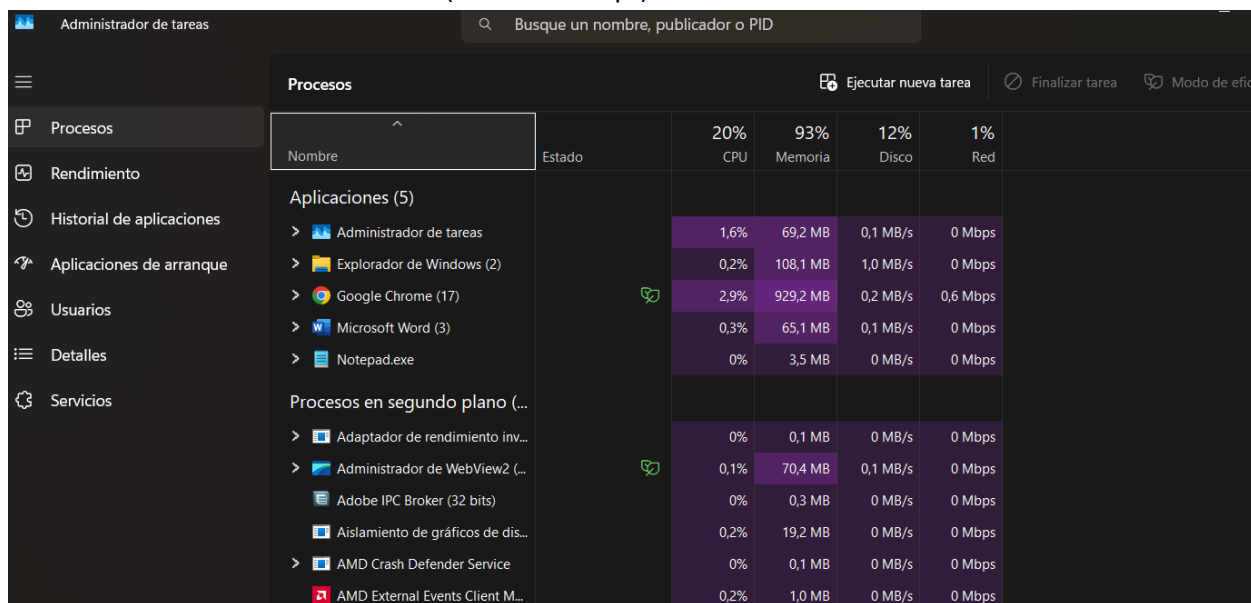
6. Creación del archivo Ejercicio1.py donde calculamos el promedio de notas solicitadas al usuario:

7. Ejecución del programa con: `python3 Ejercicio1.py`

A continuación, se anexan capturas de pantalla que documentan el proceso completo de instalación y configuración, desde la creación de la máquina virtual hasta la ejecución del programa en Python. Estas imágenes permiten visualizar de forma clara cada etapa del trabajo práctico realizado.

Verificamos si estamos en condiciones de ejecutar un hipervisor de tipo 2 – VirtualBox

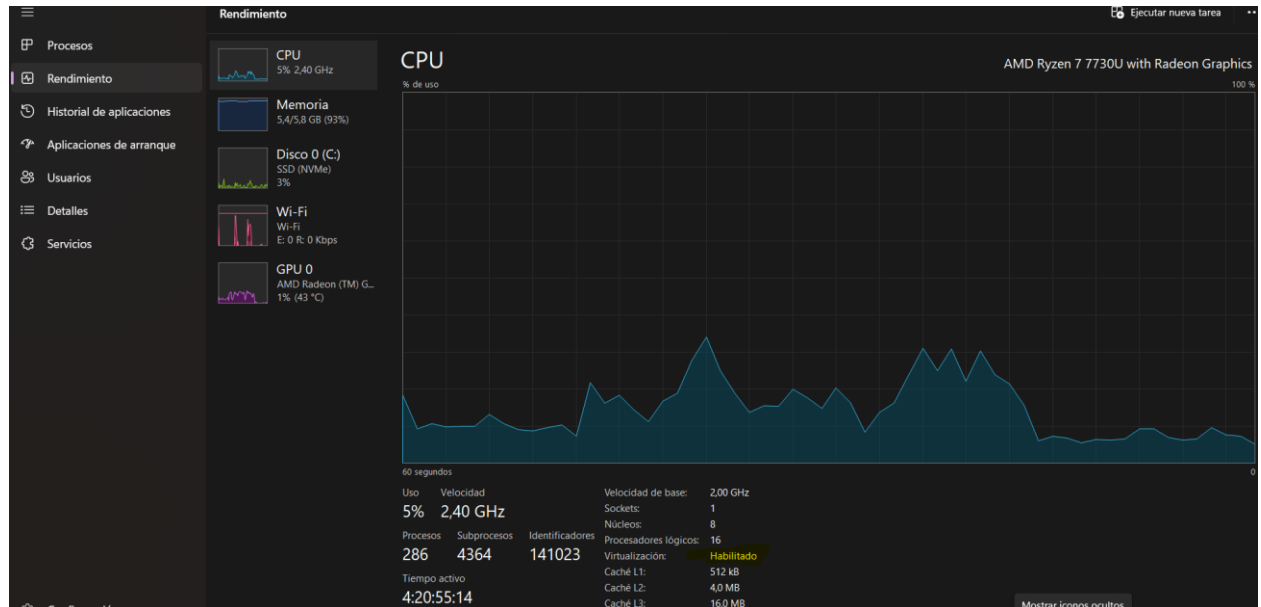
1. Abrimos el administrador de tareas: (Ctrl + Alt + Supr)



The screenshot shows the Windows Task Manager interface. The 'Procesos' tab is selected, displaying a list of running applications and background processes. The columns show the process name, status, and resource usage (CPU, Memory, Disk, Network). The 'Aplicaciones (5)' section lists the Task Manager, File Explorer, Google Chrome, Microsoft Word, and Notepad. The 'Procesos en segundo plano (...)' section lists various system services and background applications.

Nombre	Estado	20% CPU	93% Memoria	12% Disco	1% Red
Aplicaciones (5)					
Administrador de tareas		1,6%	69,2 MB	0,1 MB/s	0 Mbps
Explorador de Windows (2)		0,2%	108,1 MB	1,0 MB/s	0 Mbps
Google Chrome (17)		2,9%	929,2 MB	0,2 MB/s	0,6 Mbps
Microsoft Word (3)		0,3%	65,1 MB	0,1 MB/s	0 Mbps
Notepad.exe		0%	3,5 MB	0 MB/s	0 Mbps
Procesos en segundo plano (...)					
Adaptador de rendimiento inv...		0%	0,1 MB	0 MB/s	0 Mbps
Administrador de WebView2 (...)		0,1%	70,4 MB	0,1 MB/s	0 Mbps
Adobe IPC Broker (32 bits)		0%	0,3 MB	0 MB/s	0 Mbps
Aislamiento de gráficos de dis...		0,2%	19,2 MB	0 MB/s	0 Mbps
AMD Crash Defender Service		0%	0,1 MB	0 MB/s	0 Mbps
AMD External Events Client M...		0,2%	1,0 MB	0 MB/s	0 Mbps

2. Nos movemos a la opción de rendimiento, la cual nos permite poder visualizar si nuestro CPU tiene habilitada la virtualización.



Descargar VirtualBox

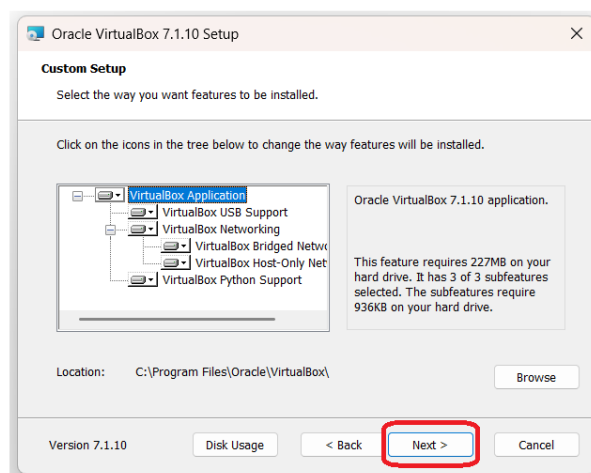
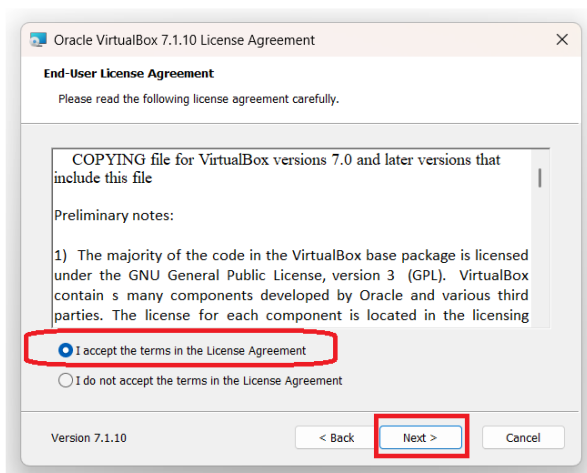
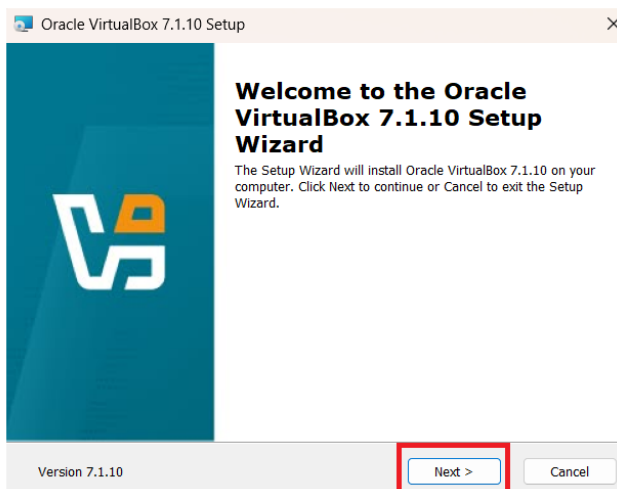
1. Ir a la pag web <https://www.virtualbox.org/wiki/Downloads>
2. Elegimos de acuerdo a nuestro SO, En nuestro caso seria Windows



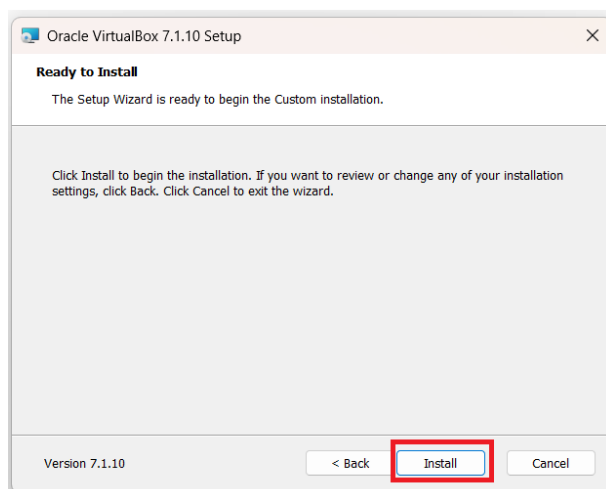
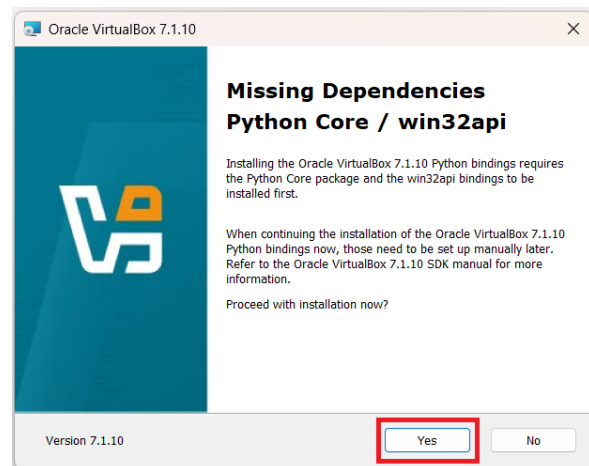
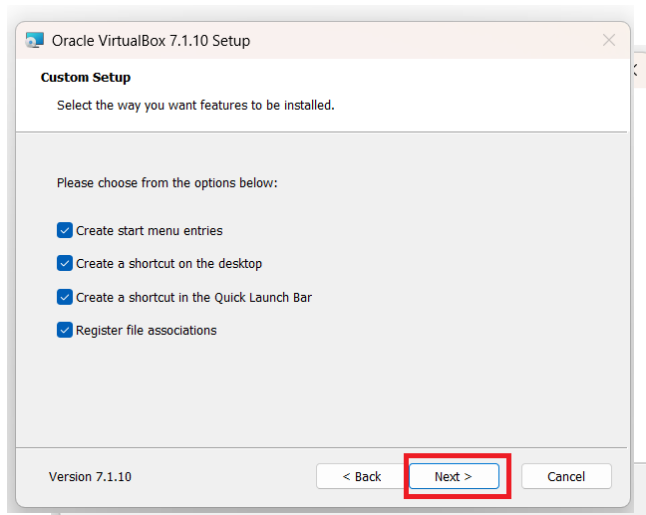
3. Lo descargamos, vamos a nuestra carpeta de descargas y ejecutamos el instalador



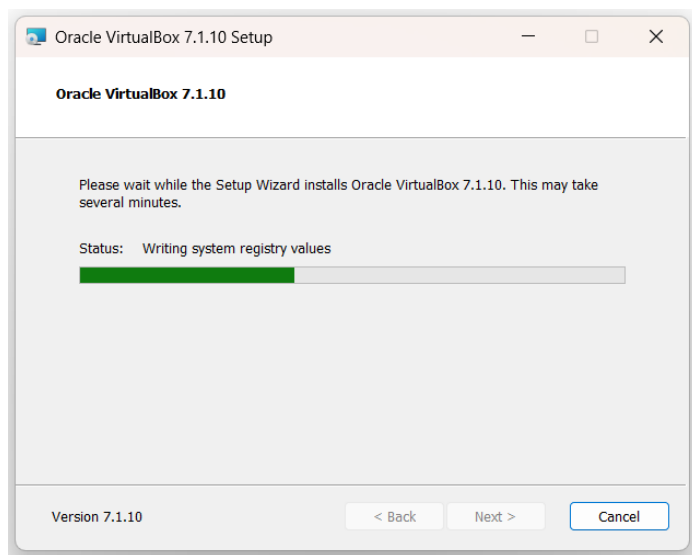
Le damos siguiente, luego aceptamos los términos y condiciones y siguiente nuevamente:



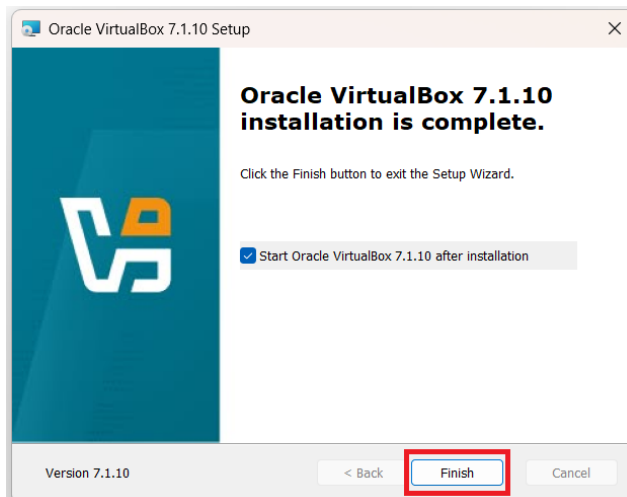
Para finalizar damos click como indicamos a continuación, e
Install:



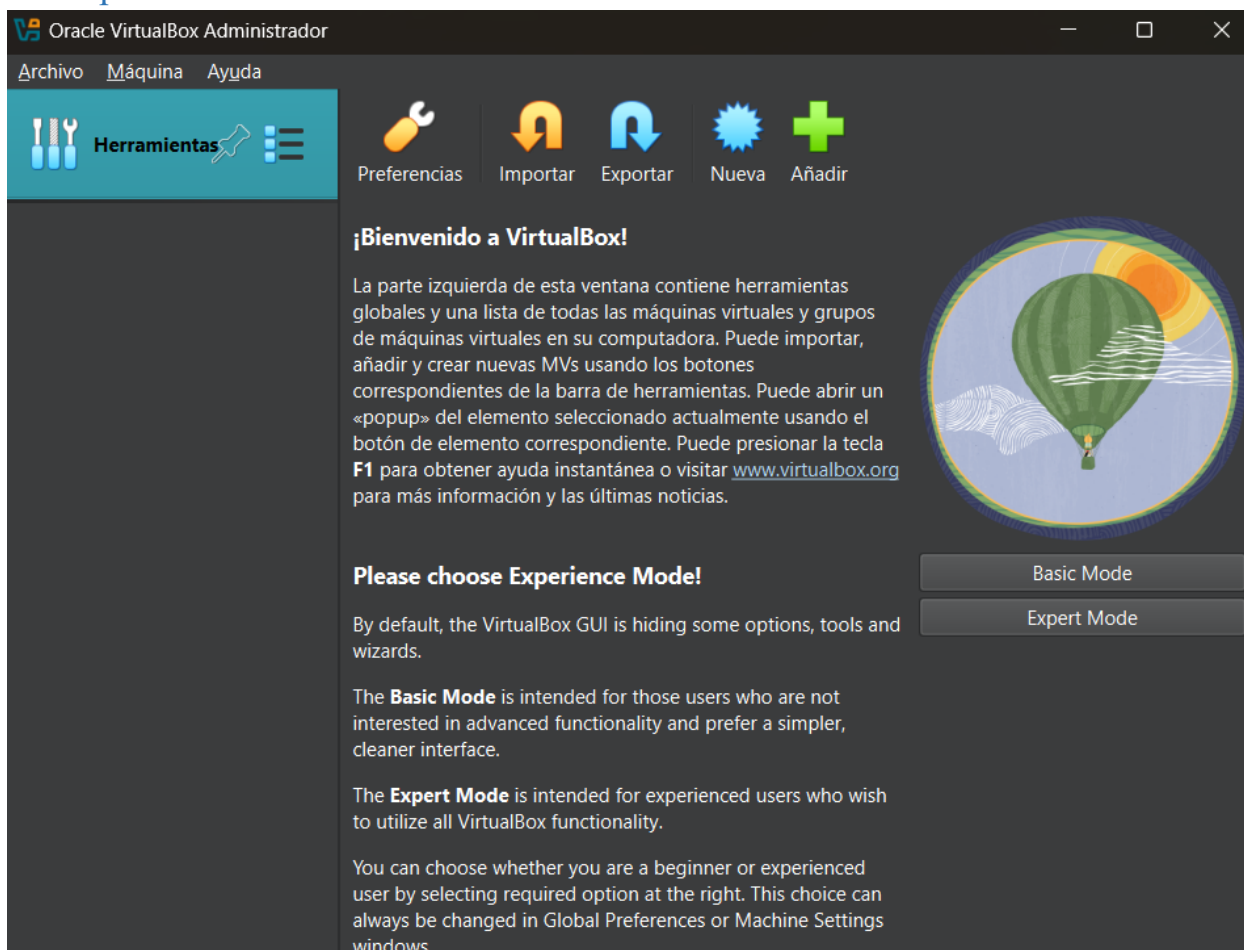
Y Comienza la descargar, vamos siguiendo el avance viendo el status:



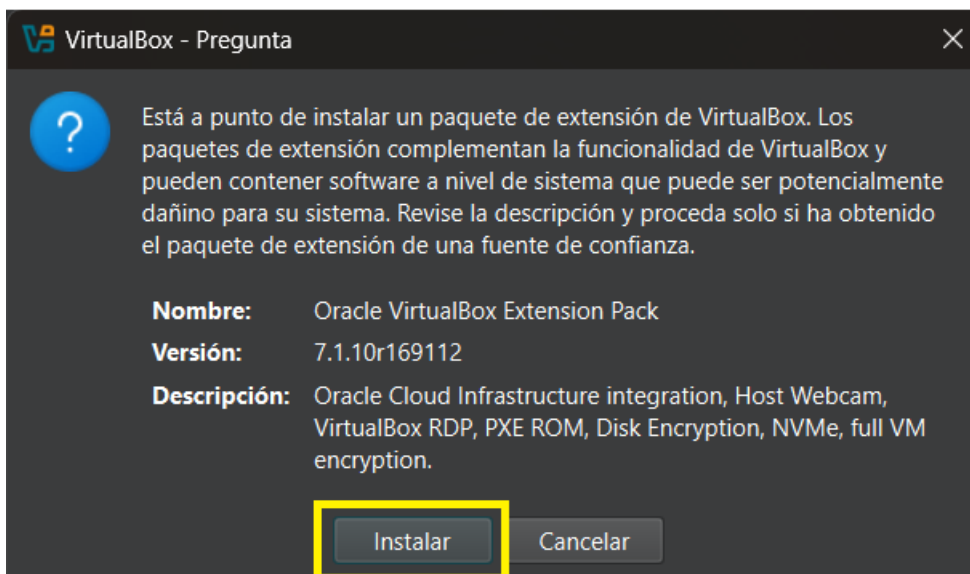
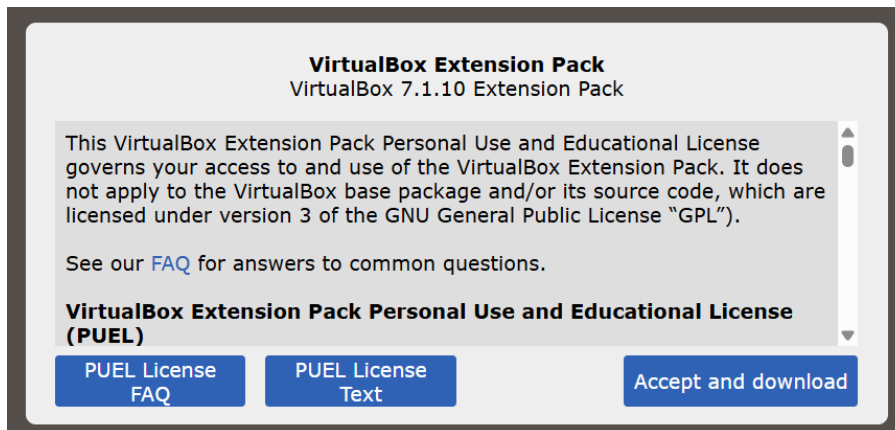
Y finalizo (Le damos click en Finish):



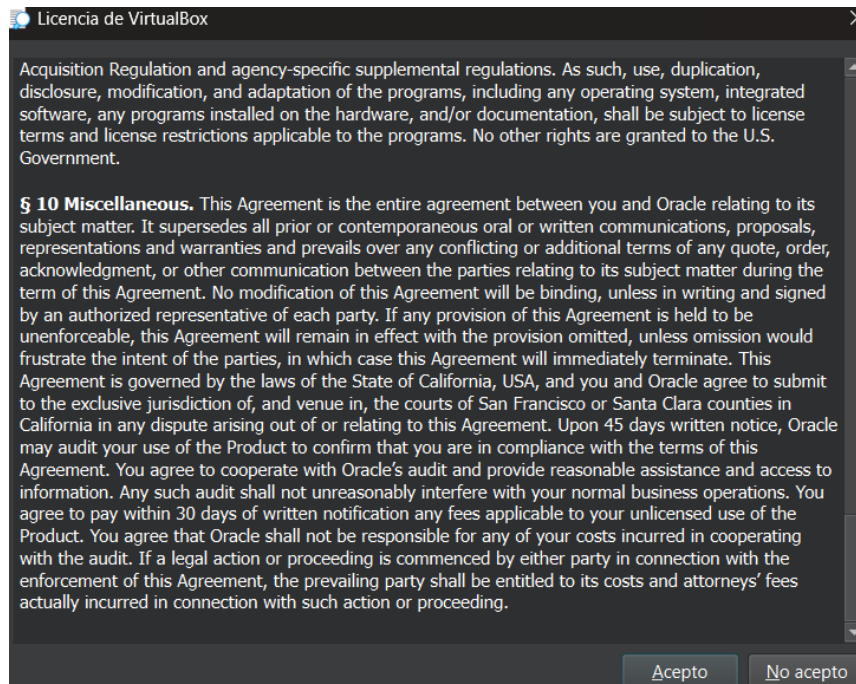
Nos aparece la BIENVENIDA a VM:



Después descargamos el paquete de expansión que también vamos a instalar:



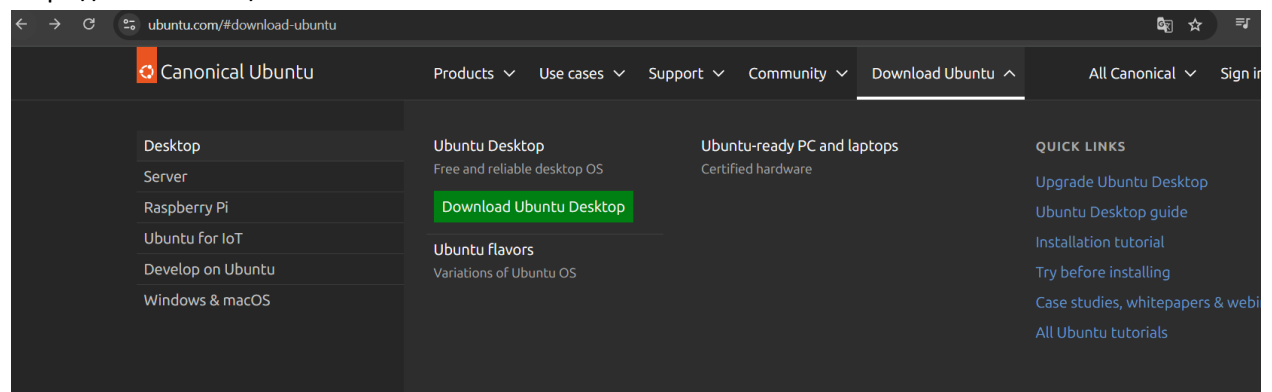
Le damos click en ACEPTO



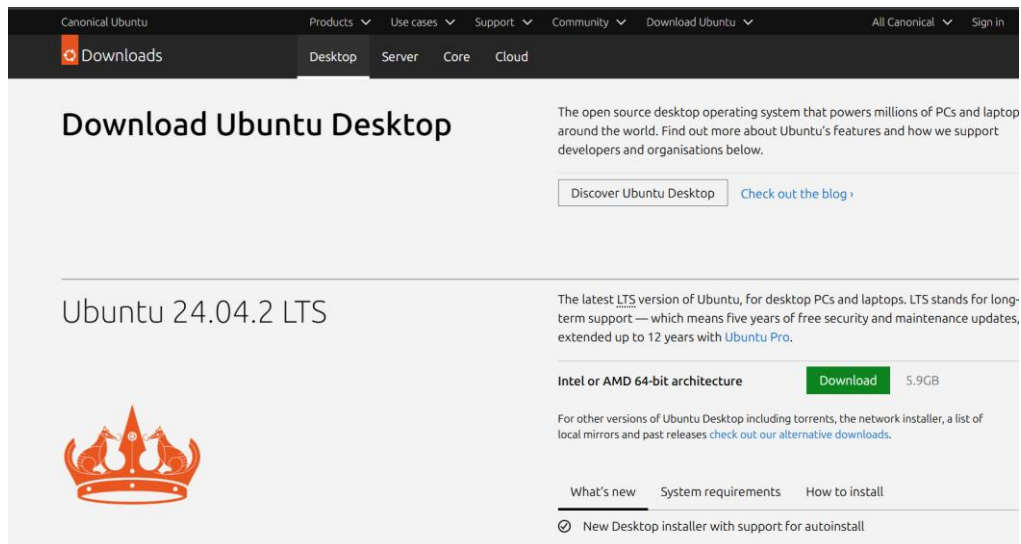
Generamos la primera Máquina Virtual

1. Descargamos de la pagina oficial de Ubuntu, el último sistema operativo
Le damos click en Download Ubuntu Desktop

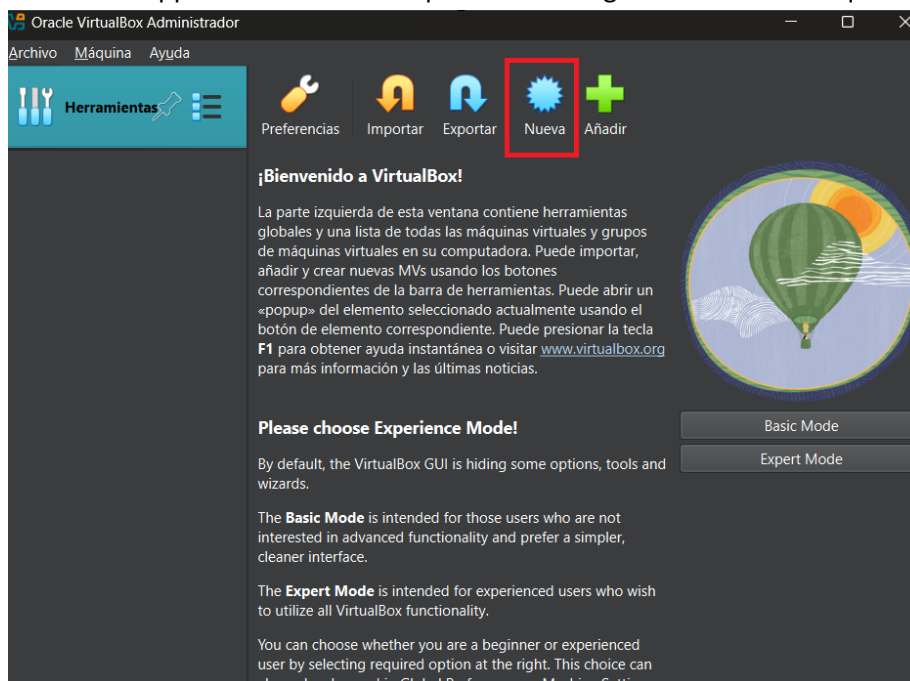
<https://ubuntu.com/#download-ubuntu>



2. Descargamos la última versión estable



2. Abrimos la app Oracle Virtual Box que esta descargada en nuestra maquina y damos click en NUEVA:

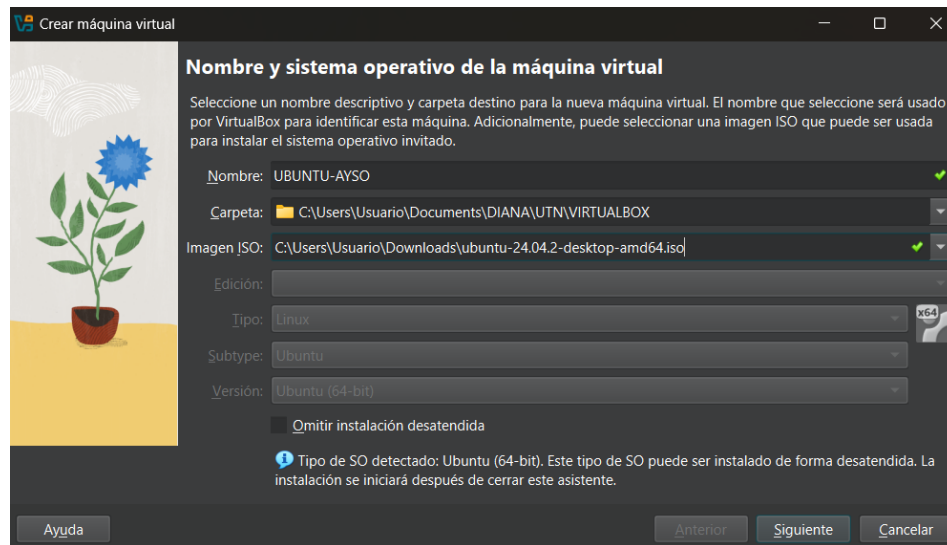


3. Completamos los campos:

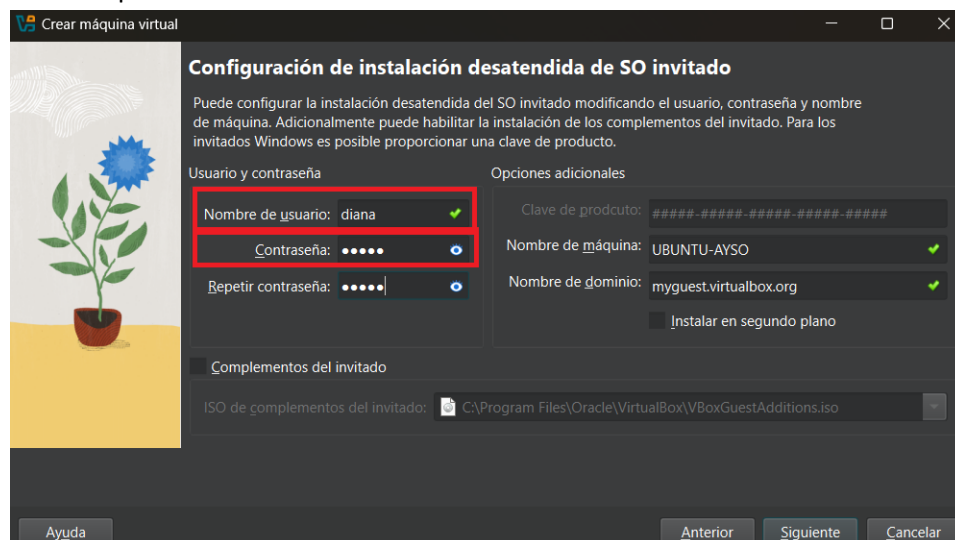
NOMBRE: UBUNTU-AYSO

CARPETA: elegimos la ruta en nuestra máquina

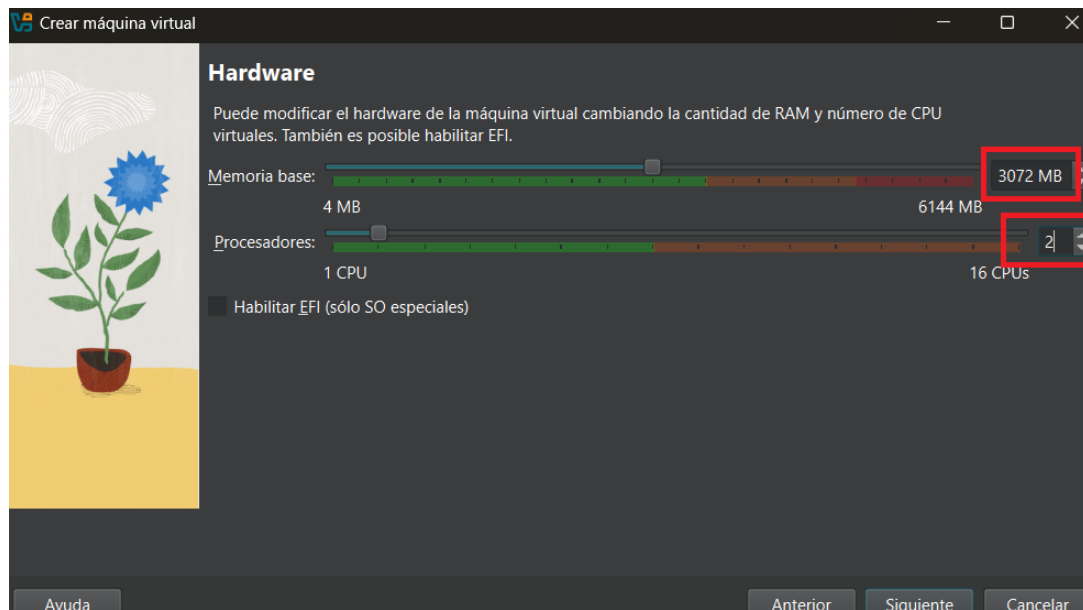
IMAGEN ISO: Buscamos la que habíamos descargado



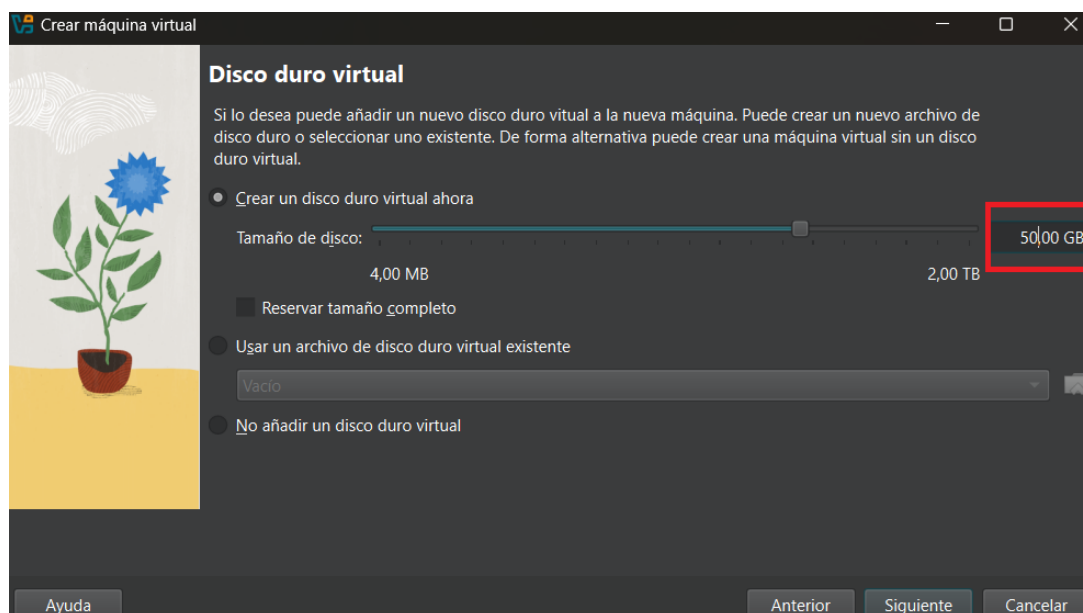
4. Luego al darle click a SIGUIENTE, cambiamos USUARIO y contraseña, sino nos quedaría la que se encuentra por defecto:



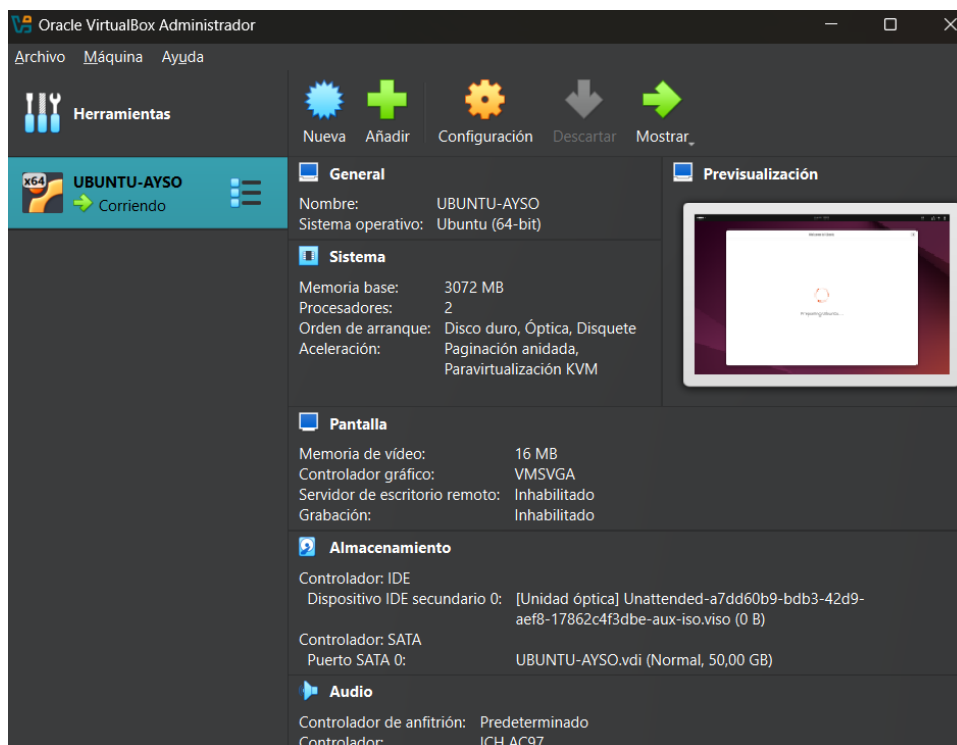
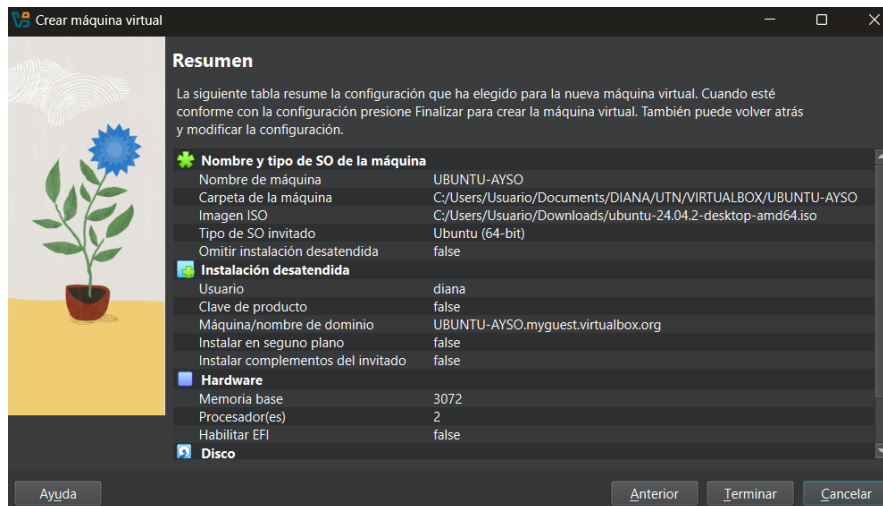
5. Al darle a SIGUIENTE, se nos abre la ventana de **HARDWARE**, en la cual cambiamos los siguientes valores, en nuestro caso le pusimos en **MEMORIA BASE** 3072 MB (que son 3 GB), y en **PROCESADORES** pusimos 2 núcleos.

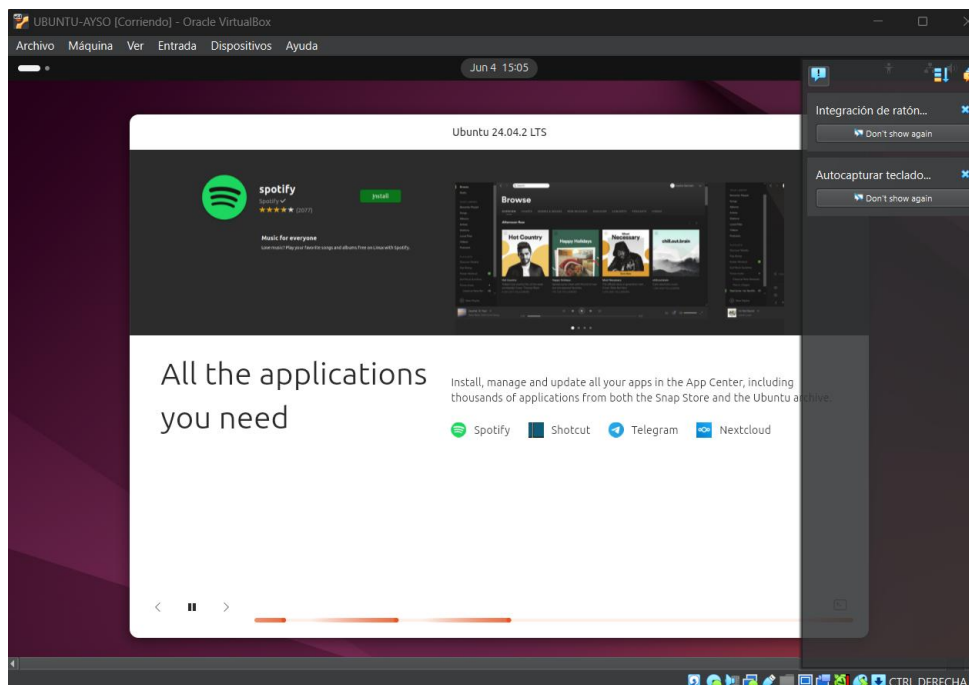
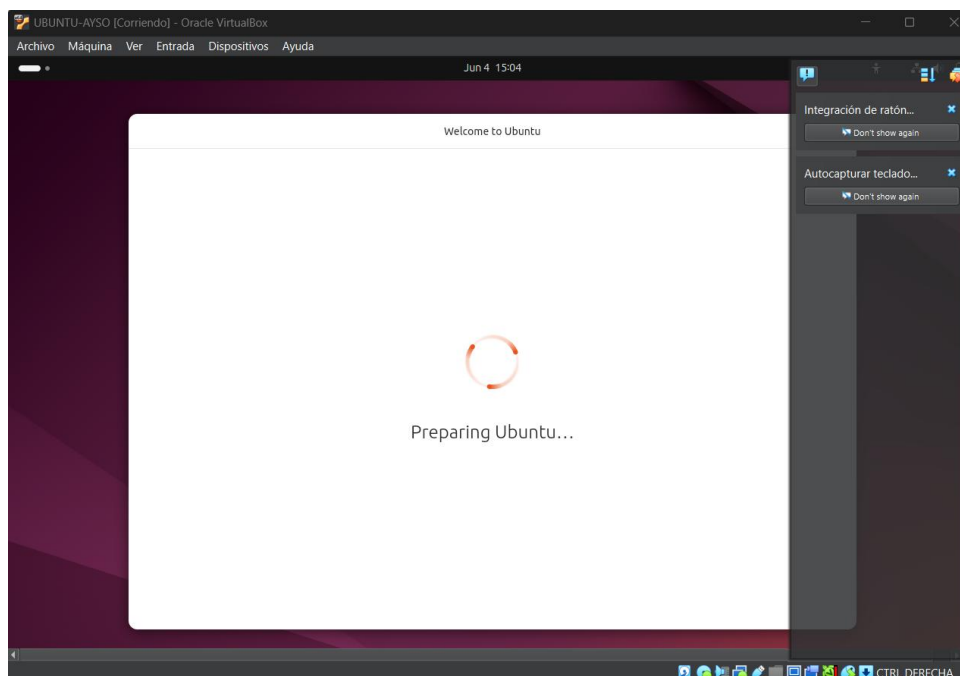


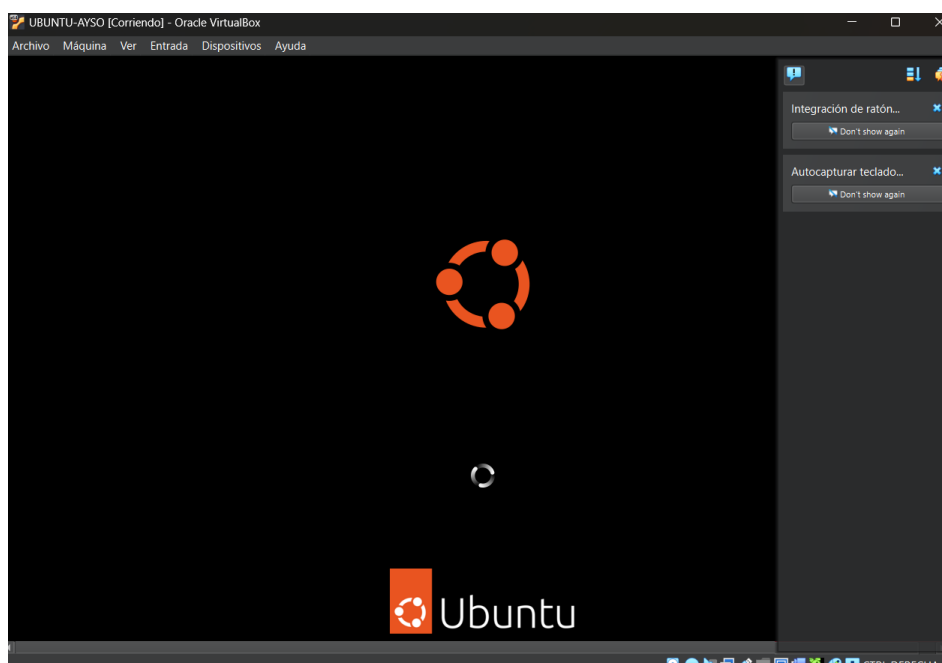
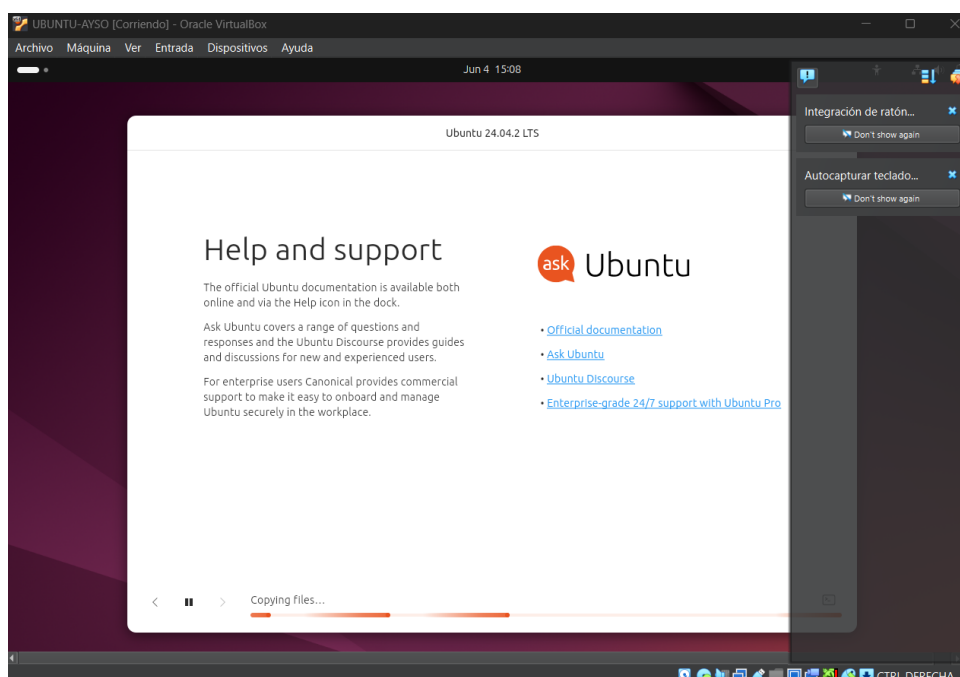
6. Siguiendo, se nos abre la ventana de **DISCO DURO VIRTUAL**, el cambio que realizamos es del tamaño a 50GB



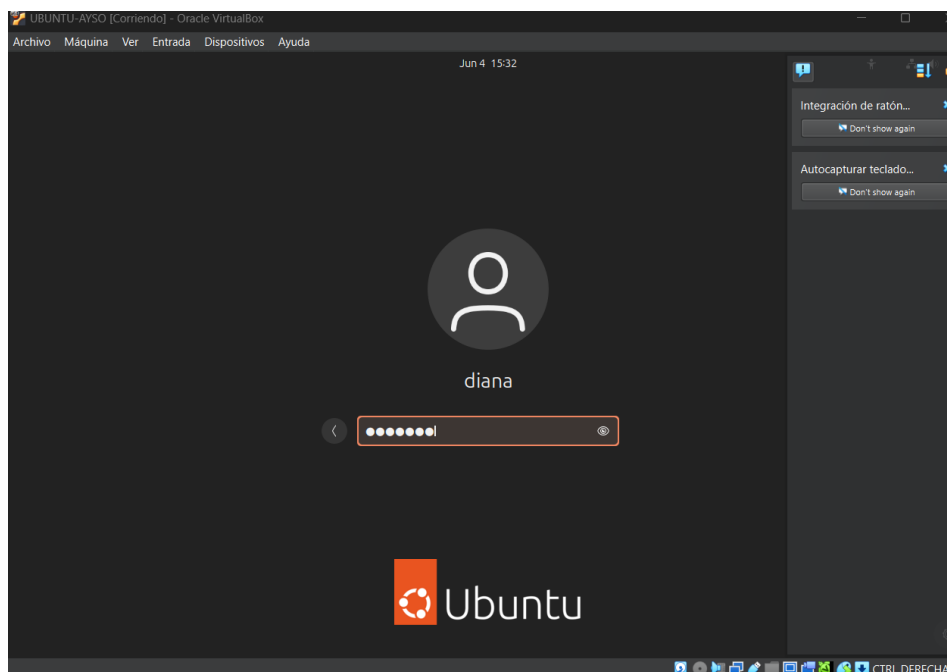
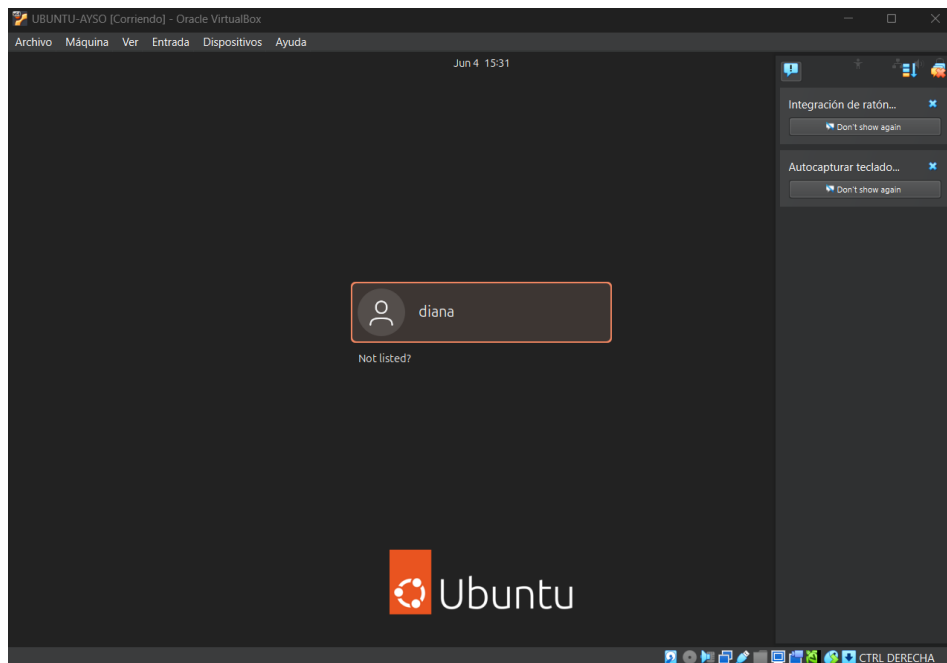
7. Por último le damos click en TERMINAR:



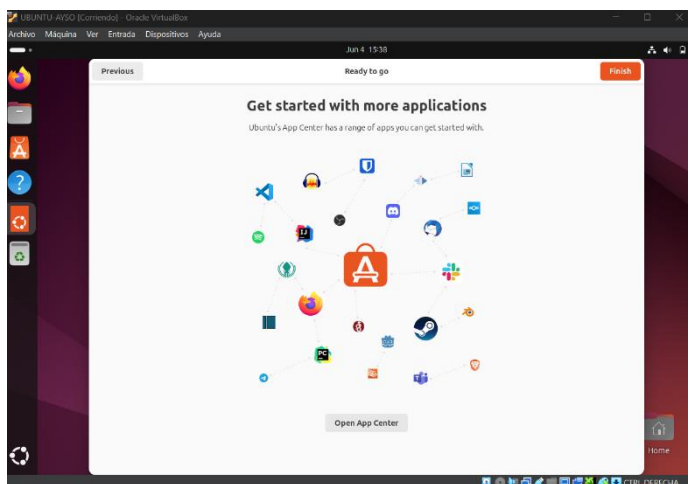
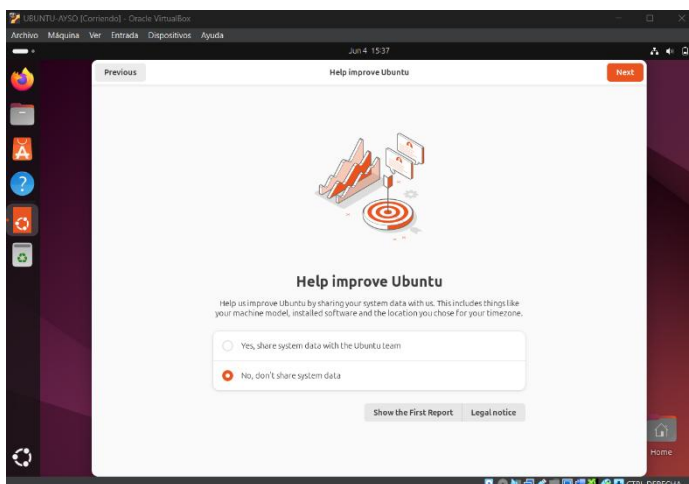
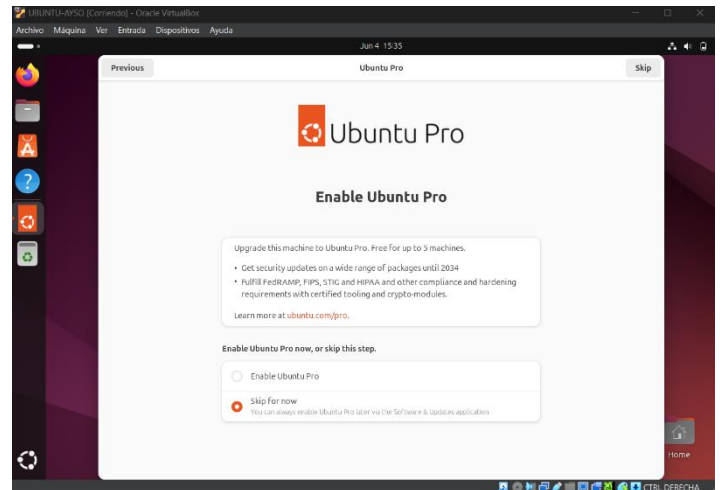
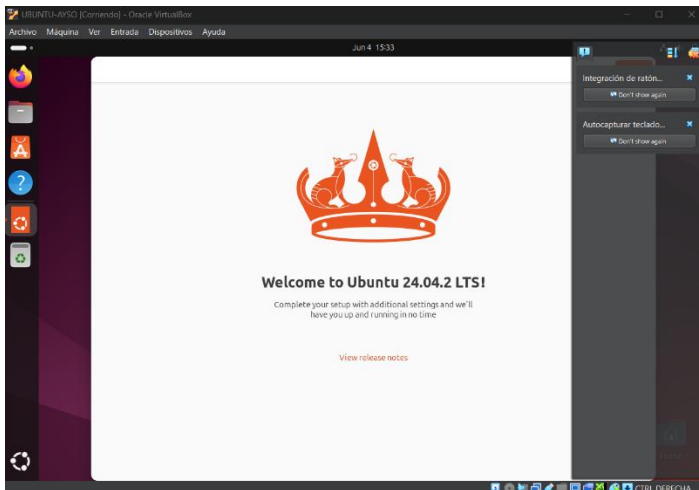




Tecnicatura Universitaria en Programación a Distancia



Tecnicatura Universitaria en Programación a Distancia

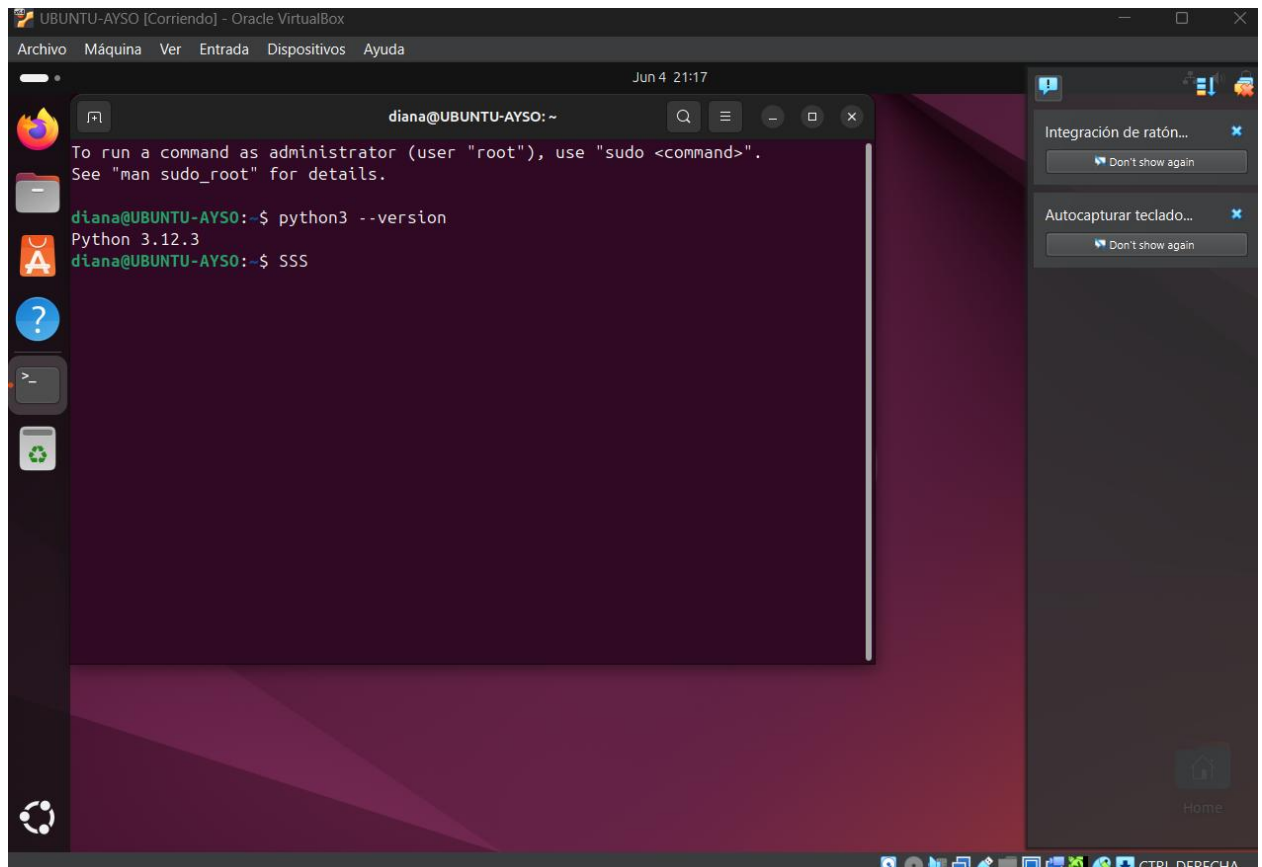


Le damos click en FINISH

Ahora ya nos encontramos dentro de nuestra máquina virtual:

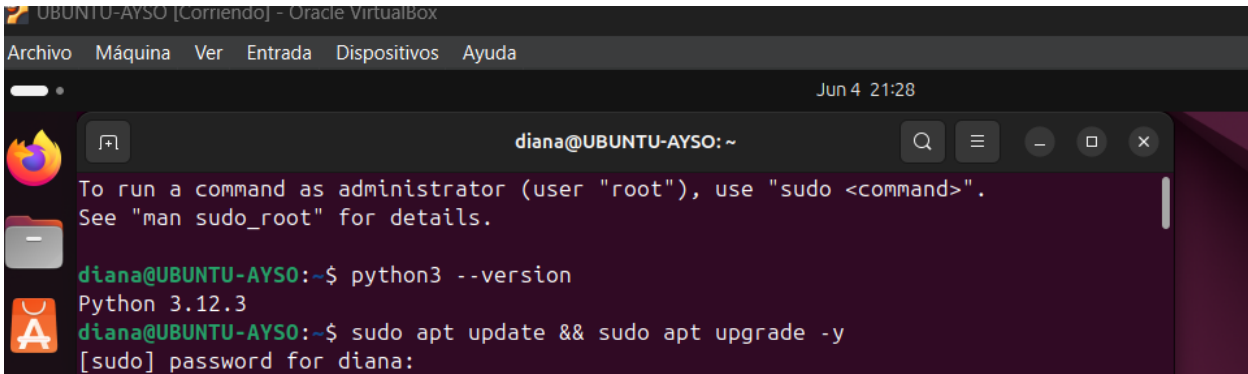
Vamos a consultar si Python ya está instalado en la VM ejecutando en la terminal:

```
python3 --versión
```



Como podemos observar ya lo tenemos instalado, lo que procederemos a realizar es Actualizar los repositorios:

```
sudo apt update && sudo apt upgrade -y
```

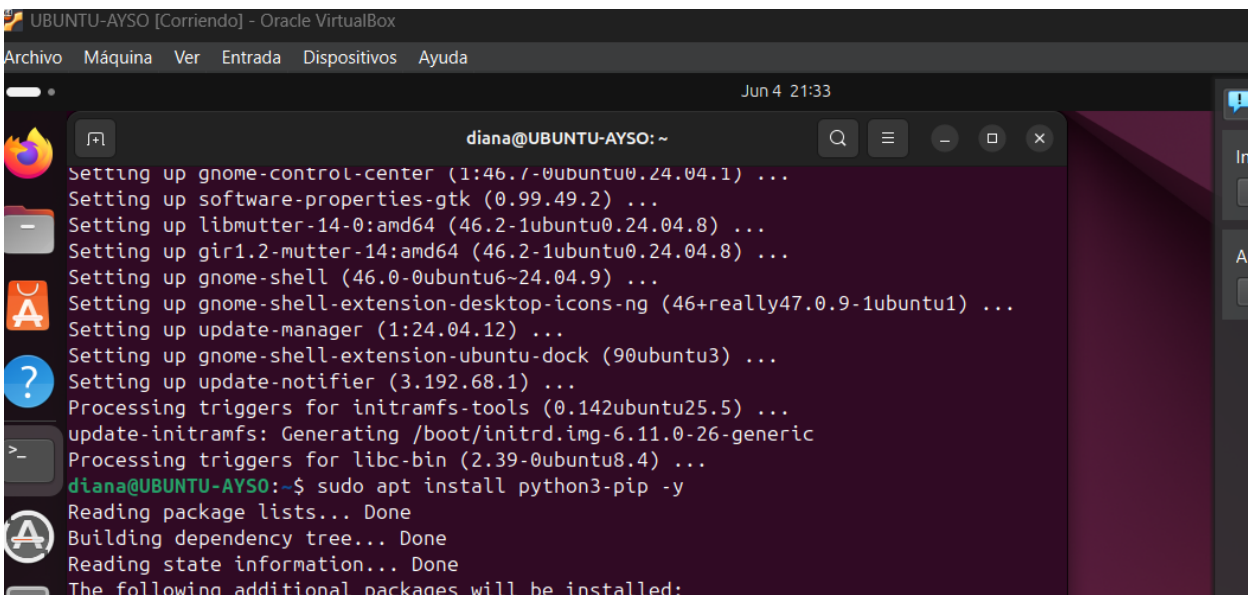


```
UBUNTU-AYSO [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Jun 4 21:28
diana@UBUNTU-AYSO: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
diana@UBUNTU-AYSO:~$ python3 --version
Python 3.12.3
diana@UBUNTU-AYSO:~$ sudo apt update && sudo apt upgrade -y
[sudo] password for diana:
```

Cuando nos pide la contraseña, la escribimos y le damos click a enter, y veremos que empieza el progreso, hasta finalizar.

Luego procedemos a **Instalar pip (gestor de paquetes de Python)**

```
sudo apt install python3-pip -y
```



```
UBUNTU-AYSO [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Jun 4 21:33
diana@UBUNTU-AYSO: ~
Setting up gnome-control-center (1:46.7-0ubuntu0.24.04.1) ...
Setting up software-properties-gtk (0.99.49.2) ...
Setting up libmutter-14-0:amd64 (46.2-1ubuntu0.24.04.8) ...
Setting up gir1.2-mutter-14:amd64 (46.2-1ubuntu0.24.04.8) ...
Setting up gnome-shell (46.0-0ubuntu6~24.04.9) ...
Setting up gnome-shell-extension-desktop-icons-ng (46+really47.0.9-1ubuntu1) ...
Setting up update-manager (1:24.04.12) ...
Setting up gnome-shell-extension-ubuntu-dock (90ubuntu3) ...
Setting up update-notifier (3.192.68.1) ...
Processing triggers for initramfs-tools (0.142ubuntu25.5) ...
update-initramfs: Generating /boot/initrd.img-6.11.0-26-generic
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
diana@UBUNTU-AYSO:~$ sudo apt install python3-pip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

Por último verificamos:

```
python3 --version
```

```
pip3 --version
```

```
diana@UBUNTU-AYSO:~$ python3 --version
Python 3.12.3
diana@UBUNTU-AYSO:~$ pip3 --version
pip 24.0 from /usr/lib/python3/dist-packages/pip (python 3.12)
```

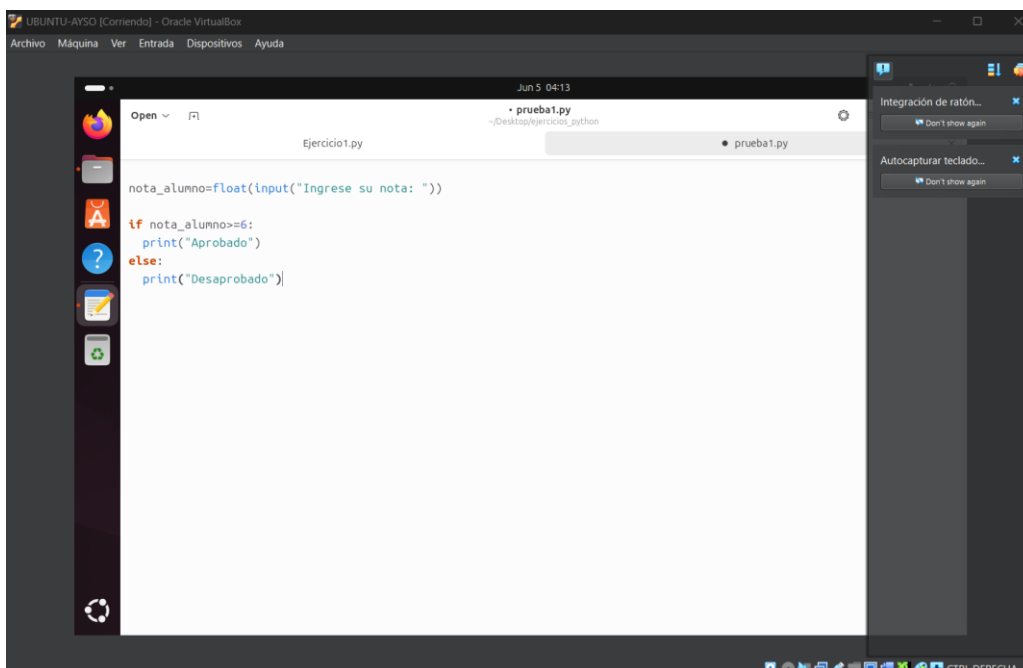
Con esto tenemos listo PYTHON para poder programar en nuestra máquina virtual.

Ahora probaremos códigos de ejemplo en PYTHON:

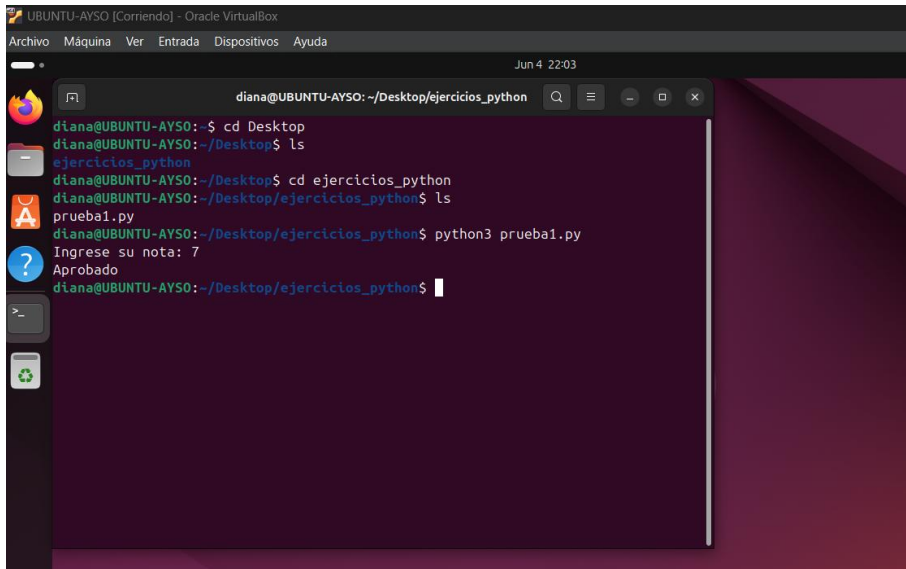
1. Abrimos el editor de texto dentro de nuestra máquina virtual y escribimos el código python:

En esta primer prueba cree un programa que pida al usuario su nota y que si es mayor o igual a 6 diga Aprobado o de lo contrario desaprobado.

Una vez creado el código lo guardamos en una carpeta creada en el escritorio llamada ejercicios_python, al nombre del archivo, le pusimos prueba1, le colocamos la extensión al final .py, y quedaría prueba1.py



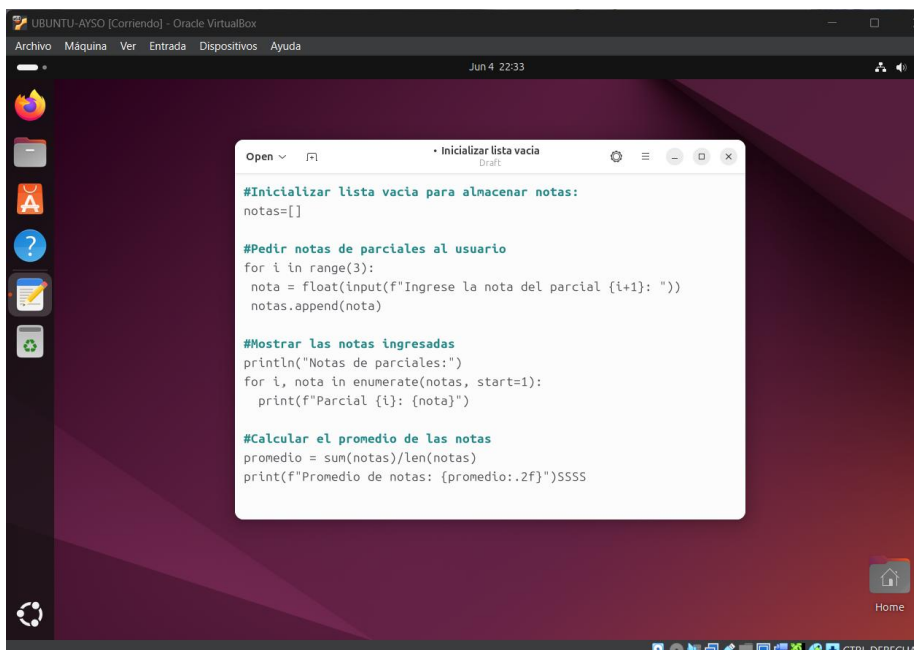
Probaremos este código, para ello ingresamos en la terminal, y ponemos los siguientes comandos, y cómo podemos observar funciona correctamente



```
diana@UBUNTU-AYSO:~$ cd Desktop
diana@UBUNTU-AYSO:~/Desktop$ ls
ejercicios_python
diana@UBUNTU-AYSO:~/Desktop$ cd ejercicios_python
diana@UBUNTU-AYSO:~/Desktop/ejercicios_python$ ls
prueba1.py
diana@UBUNTU-AYSO:~/Desktop/ejercicios_python$ python3 prueba1.py
Ingrese su nota: 7
Aprobado
diana@UBUNTU-AYSO:~/Desktop/ejercicios_python$
```

Para finalizar creamos un código de ejemplo en PYTHON:

1. Abrimos nuestro editor de texto y codificamos el programa.



```
Open ▾ | Inicializar lista vacia
Draft

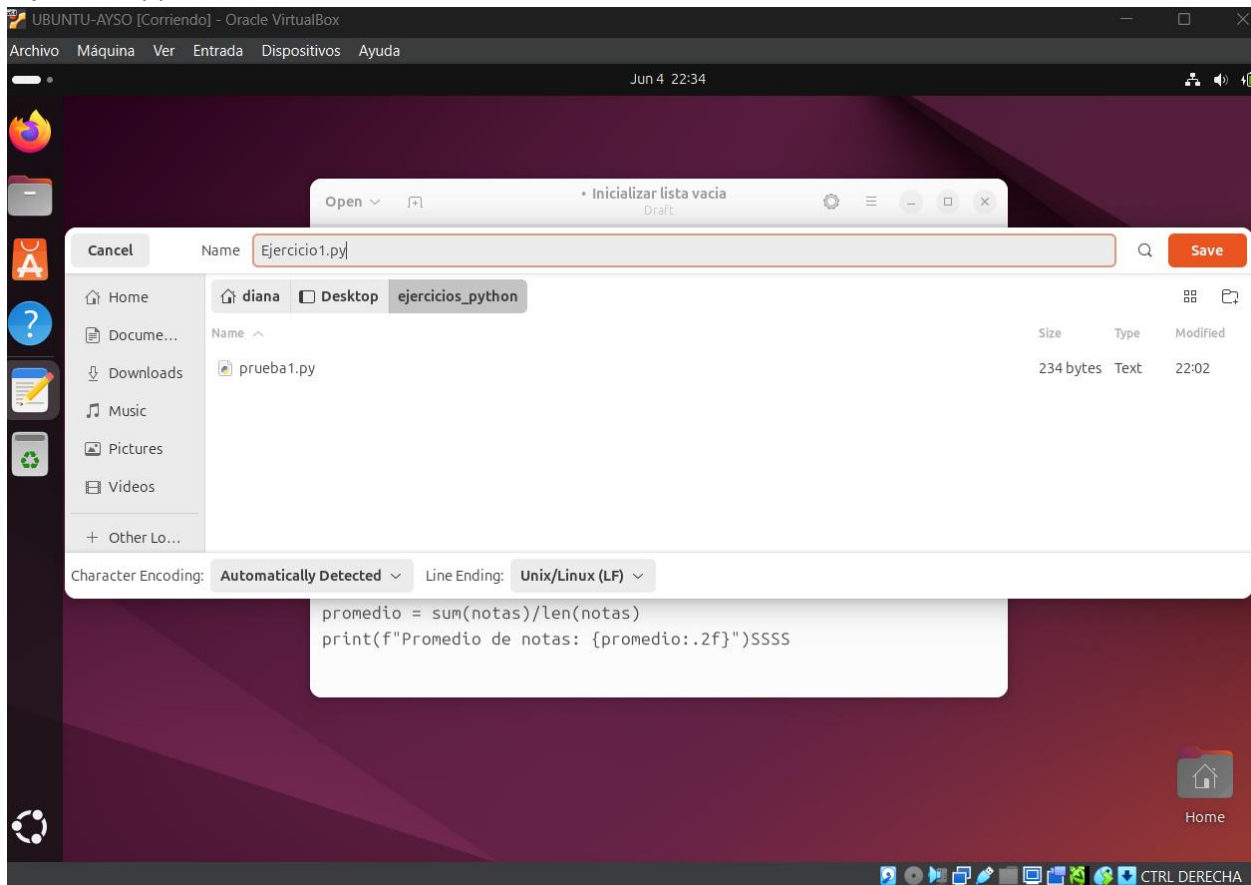
#Inicializar lista vacia para almacenar notas:
notas=[]

#Pedir notas de parciales al usuario
for i in range(3):
    nota = float(input(f"Ingrese la nota del parcial {i+1}: "))
    notas.append(nota)

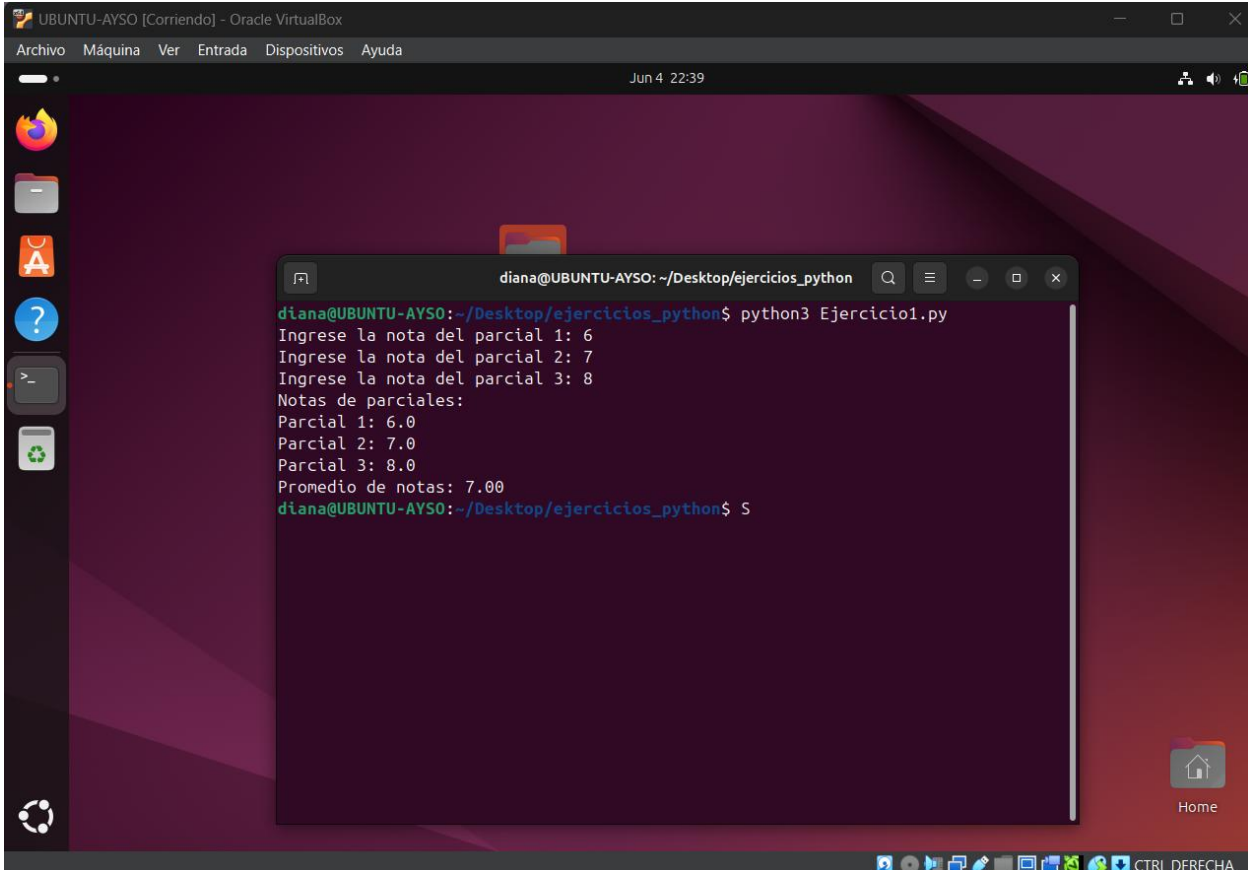
#Mostrar las notas ingresadas
println("Notas de parciales:")
for i, nota in enumerate(notas, start=1):
    print(f"Parcial {i}: {nota}")

#Calcular el promedio de las notas
promedio = sum(notas)/len(notas)
print(f"Promedio de notas: {promedio:.2f}")SSSS
```

2. Luego lo guardamos en nuestra carpeta ya creada ("ejercicios_python") en el escritorio, con el nombre "Ejercicio1.py"



3. Por último abrimos el Terminal y ejecutamos nuestro programa:



```
diana@UBUNTU-AYSO: ~/Desktop/ejercicios_python
diana@UBUNTU-AYSO:~/Desktop/ejercicios_python$ python3 Ejercicio1.py
Ingrese la nota del parcial 1: 6
Ingrese la nota del parcial 2: 7
Ingrese la nota del parcial 3: 8
Notas de parciales:
Parcial 1: 6.0
Parcial 2: 7.0
Parcial 3: 8.0
Promedio de notas: 7.00
diana@UBUNTU-AYSO:~/Desktop/ejercicios_python$
```

4. Metodología Utilizada

- Instalación de VirtualBox en Windows 10.
- Descarga de Ubuntu 24.04.2 LTS
- Configuración de la VM.
- Instalación de Python y ejecución del script.
- Trabajo colaborativo en la planificación y documentación del proceso.

5. Resultados Obtenidos

- La máquina virtual se creó y funcionó correctamente.
- Ubuntu fue instalado sin problemas.
- Python se ejecutó exitosamente.
- Aprendimos a manejar VirtualBox y el entorno de terminal de Ubuntu.

6. Conclusiones

Este trabajo nos permitió aplicar conocimientos básicos de sistemas operativos y programación en un entorno virtual. Aprendimos a configurar una VM, instalar un sistema operativo y trabajar con la terminal para ejecutar código. Fue una experiencia muy útil para nuestra formación.

7. Bibliografía

- Documentación de VirtualBox: <https://www.virtualbox.org/manual/>
- Guía de Ubuntu Server: <https://ubuntu.com/server/docs>
- Video tutorial: <https://www.youtube.com/watch?v=Ne-IS7m9HEw>
- Kaspersky. (s.f.). *¿Qué es una sandbox?*. Recuperado de <https://latam.kaspersky.com/resource-center/definitions/sandbox>

8. Anexos

- Captura de pantalla del sistema corriendo Ubuntu.
- Imagen del script Ejercicio1.py en ejecución.
- Configuración de la máquina virtual en VirtualBox.

9. Links archivos de entrega

- Video : <https://www.youtube.com/watch?v=jl-DSbE2Oss>
- Repositorio GITHUB: <https://github.com/Diana-Falla/TP-Integrador-AYSO>