

Trabajo Integrador – Virtualización con VirtualBox

Alumnos:

María Verónica Biskupski – veronicabiskupski79@gmail.com - Comisión 7

Leylén Magalí Demarlenge - ldemarlenge@gmail.com - Comisión 4

Materia: Arquitectura y Sistemas Operativos

Profesor: Ariel Enferrel y Martín Aristiaran

Fecha de Entrega: 22 de octubre de 2025

Índice

Índice	1
1. Introducción	1
2. Marco Teórico	2
3. Caso Práctico	2
4. Metodología Utilizada	3
5. Resultados Obtenidos	3
6. Conclusiones	3
7. Bibliografía	4

1. Introducción

Se eligió el tema de virtualización porque permite explorar y utilizar otros sistemas operativos que, si bien se conocían por estudiarlo en la materia, no estaban instalados en las computadoras de las autoras. Además, la virtualización permite simular, crear, probar diferentes programas y/o configuraciones sin poner en riesgo la PC anfitriona.

Como técnicos en programación es crucial conocer diferentes sistemas operativos y herramientas que a futuro se puedan encontrar en el ámbito laboral, lo que forma profesionales más completos. Cuanto más herramientas se dominen, más competitivo será el técnico en programación.

En este trabajo se crea un programa en Python desde la VM invitada y tendrá como objetivo aplicar los conocimientos adquiridos en el curso e integrarlos con las competencias de programación que ya se han incorporado.

2. Marco Teórico

- Virtualización y Arquitectura
 - Virtualización de Hardware (Máquina Virtual o VM): permite ejecutar un sistema operativo completo y aislado (invitado) sobre un sistema operativo principal (anfitrión), simulando el hardware físico.
 - Hipervisor Tipo 2 (Hosted): software que se ejecuta como una aplicación dentro del sistema operativo anfitrión para gestionar las VMs. El software VirtualBox pertenece a esta categoría.
 - VirtualBox Guest Additions: Controladores instalados dentro del sistema operativo invitado, esenciales para:
 - Mejorar rendimiento gráfico.
 - Habilitar el Portapapeles Compartido Bidireccional y la función de Arrastrar y Soltar.
 - Mejorar la integración del hardware virtual.
- Entorno Operativo y Desarrollo
 - Sistema Operativo Anfitrión (Host): sistema operativo físico que aloja el software Virtual Box, Microsoft Windows en este caso.
 - Sistema Operativo Invitado (Guest): sistema operativo instalado dentro de la VM (Ubuntu), proporcionando un entorno de desarrollo aislado.
 - Red NAT: configuración de red por defecto que permite al invitado acceder a la red externa (internet) a través del anfitrión, manteniendo el aislamiento de red.
 - Python: lenguaje de programación utilizado para el desarrollo.
 - pip (Package Installer for Python): herramienta estándar para la gestión e instalación de librerías y módulos de terceros en Python.

3. Caso Práctico

Se utilizó VirtualBox para instalar una máquina virtual con Ubuntu Server 22.04 y generar un programa en Python.

El programa a desarrollar solicitará al usuario 3 números mediante un ciclo For y calculará el promedio.

4. Metodología Utilizada

Pasos:

1. Instalación de VirtualBox en Windows
2. Descarga de imagen ISO desde: <https://ubuntu.com/download>
3. Se crea una VM con 20 GB de disco, 3072 MB de memoria RAM, 2 núcleos.
4. Se monta el ISO y se instala Ubuntu
5. Se instala Python. Virtual Studio Code ya estaba instalado y es el que se utilizará como editor de código.
6. Se desarrolla el programa en Virtual Studio Code, se prueba y corrige hasta conseguir la versión definitiva.

Para el desarrollo del trabajo Verónica realizó la configuración de la VM, instaló Python y copió el código en Virtual Studio Code; mientras que Leylén generó el repositorio y organizó la documentación a presentar. Si bien el trabajo de cada integrante estaba bien definido, se crearon reuniones en meet para hacer consultas y brindar soporte. El código de Python se tomó de un ejercicio de la cátedra Programación 1 y fue mejorado para este trabajo.

El informe técnico, especialmente la conclusión fue hecha en conjunto mediante la puesta en común del proceso.

5. Resultados Obtenidos

- El sistema operativo invitado funcionó correctamente
- Se logró la correcta instalación del Python desde la consola de Ubuntu
- El código de Python funcionó correctamente
- Se logró interacción entre la VM invitada y la anfitriona

Errores corregidos:

Instalación de Virtualbox (le faltaba a la pc el C++).

Configuración de Ubuntu (el nombre de la máquina iba con minúscula)

6. Conclusiones

Surgieron dificultades durante el desarrollo del trabajo las cuales se detallan a continuación:

- Instalación de Ubuntu en VirtualBox:

Al iniciar la instalación de Ubuntu en Virtualbox no se podía dar al botón siguiente o finalizar ya que el nombre de la máquina virtual iba todo con minúscula. A este error lo buscamos en Chat GPT encontrando la solución.

Se intentó la solución planteada y aun así no funcionó, por lo que cerramos la sesión y se comenzó de nuevo desde el inicio tomando la precaución de las minúsculas en VM name. De esta manera se logró finalizar la instalación exitosamente.

- Capturas de pantalla en la VM invitada y cómo trasladarlas a la anfitriona

Al tomar capturas de pantalla de la VM invitada (necesarias para el registro del trabajo) no se podían pegar luego en la PC anfitriona. La solución más práctica que se encontró fue abrir el navegador FIREFOX y desde allí abrir Whatsapp Web en la VM enviar las capturas por ahí, para luego descargarlas en la PC anfitriona desde la misma aplicación.

- Instalación de Python por consola

Al verificar si en la VM estaban instalados los programas necesarios para el desarrollo del trabajo se encontró que Virtual Studio Code estaba instalado pero Python no.

Para poder instalarlo por consola consultamos antes a Chat GPT el cual nos dio un paso a paso que al final abandonamos y consultamos a Gemini que también nos guió paso a paso, esto nos ayudó a comprender mejor el funcionamiento de Ubuntu. Primero se verificó si el programa ya estaba instalado con el comando “python3 --version”.

- Otra de las dificultades fue el rendimiento, en varias ocasiones el sistema daba el error de “no responde”, lo cual nos llevó en varias oportunidades a tener que volver a reiniciar la VM, probamos de cerrar aplicaciones, ventanas de google que tenía abiertas en la PC anfitriona para poder corroborar si mejoraba, esto fue así en un momento dado pero luego volvía a suceder.

Como posible mejora para futuros trabajos, deberíamos profundizar en la configuración y manejo de entornos virtualizados, optimizando los procesos de instalación y transferencia de archivos entre la máquina virtual y la anfitriona. Esto permitiría evitar los inconvenientes técnicos experimentados y agilizar el desarrollo de las actividades en entornos de práctica.

7. Bibliografía

- Chat GPT
- Gemini: <https://gemini.google.com>
- Tecnicatura Universitaria en Programación. Arquitectura y Sistemas Operativos. Virtualización.