**¿Qué es un sistema operativo?**

Es el software principal que actúa como intermediario entre el hardware y el usuario. Permite que se ejecuten programas, administra recursos (memoria, CPU, dispositivos) y brinda una interfaz (gráfica o de texto) para interactuar con la computadora.

¿Qué sistemas operativos conocen?

Ejemplos: Windows, Linux, macOS, Android.

**Tipos de sistemas operativos :**

Monousuario: Solo permite un usuario a la vez (ej: MS-DOS).

Multiusuario: Varios usuarios pueden usar el sistema simultáneamente (ej: Linux, Unix).

Monotarea: Ejecuta una sola tarea por vez (ej: sistemas muy antiguos).

Multitarea: Puede ejecutar varios procesos al mismo tiempo (ej: Windows, Linux).

De tiempo real: Responden a eventos en tiempos muy precisos (ej: sistemas embebidos en autos, maquinaria,Equipos médicos, ).

Distribuidos: Un sistema operativo distribuido es un software que gestiona un conjunto de computadoras interconectadas (una red) y las presenta como si fueran un solo sistema unificado para el usuario.

De red: Administran recursos y servicios para múltiples computadoras conectadas a una red (ej: Windows Server, Ubuntu Server).

**Que es una red???**  
Una **red** es un conjunto de dispositivos (como computadoras, teléfonos, impresoras, servidores, etc.) conectados entre sí para compartir recursos e intercambiar información. Estas conexiones pueden ser **alámbricas** (por cables) o **inalámbricas** (como Wi-Fi).

**SO de red? que es eso**

Un sistema operativo de red es un software que permite a una computadora actuar como servidor y brindar servicios (como compartir archivos, impresoras, conexión a Internet, control de acceso, etc.) a otras computadoras conectadas a la red (clientes).

Administra usuarios y permisos desde un punto central.

Controla el acceso a archivos, carpetas y dispositivos compartidos.

Permite configurar servicios de red: como DNS, DHCP, FTP, HTTP, etc.

Gestiona la seguridad en red, como firewalls, usuarios, contraseñas.

Opera tanto en entornos cliente-servidor como en redes peer-to-peer.

| Sistema | Descripción breve |

|----------------------|---------------------------------------------------------|

| \*\*Windows Server\*\* | Versión de Windows diseñada para administrar redes. |

| \*\*Ubuntu Server\*\* | Distribución de Linux optimizada para servicios de red. |

| \*\*Debian\*\* | Usado en servidores por su estabilidad. |

| \*\*FreeBSD\*\* | Sistema derivado de Unix, potente en servidores. |

| \*\*Novell NetWare\*\* | Uno de los primeros NOS populares (años 80 y 90). |

**¿En qué se diferencia de un sistema operativo común (de escritorio)?**

| Característica | SO de Escritorio | SO de Red |

|---------------------------|------------------------------|----------------------------------|

| Enfocado en | Usuario individual | Usuarios en red |

| Prioriza | Interfaz gráfica y comodidad | Seguridad, rendimiento y acceso |

| Recursos compartidos | Limitados o ninguno | Totalmente configurables |

| Servicios de red | Opcionales | Nativos e integrados |

**INSTALAR UN ENTORNO VIRTUAL**

Ubuntu **22.04 LTS**

Una **máquina virtual (VM)** es un software que simula un sistema informático completo dentro de tu computadora. Es como tener una **computadora dentro de otra**, que puede ejecutar su propio sistema operativo (como Ubuntu) sin afectar el sistema principal (por ejemplo, Windows).

**VirtualBox** es un programa gratuito de Oracle que permite crear y ejecutar estas máquinas virtuales. Con él, podés instalar y usar sistemas operativos como **Ubuntu 22.04 LTS** sin necesidad de borrar ni modificar tu sistema actual.

**3: Primer contacto con Linux**

**¿Qué es Linux? ¿Qué lo diferencia de otros?**

Linux es un sistema operativo libre y de código abierto, basado en Unix.

Se diferencia por:

Ser gratuito.

Poder modificarse y distribuirse libremente.

Tener múltiples distribuciones (Ubuntu, Fedora, Debian, Arch).

Ser muy estable y seguro, ideal para servidores.

**¿Qué es la terminal?**

Es una interfaz de texto que permite al usuario dar órdenes directamente al sistema operativo escribiendo comandos. No tiene íconos ni ventanas: todo se hace con el teclado.

**¿Por qué se usa en Linux?**

Permite mayor control y precisión sobre el sistema.

Es más ligera que las interfaces gráficas.

Es ideal para administrar servidores, automatizar tareas, programar y trabajar en red.

Algunas herramientas solo están disponibles en terminal.

**¿Qué es un comando?**

Un comando es una instrucción que escribimos en la terminal para que el sistema operativo la ejecute.

**El entorno de comandos (CLI).**

CLI (Command Line Interface) es una interfaz de texto donde el usuario escribe comandos.

Ventajas:

Más control sobre el sistema.

Permite automatizar tareas.

Usado intensivamente en administración de servidores.

**Estructura del sistema de archivos.**

Linux organiza sus archivos como un árbol jerárquico, con / (raíz) como el comienzo.

| Carpeta | Contenido |

|-----------------|--------------------------------------------|

| `/bin` | Comandos básicos del sistema |

| `/etc` | Archivos de configuración |

| `/home` | Carpetas personales de los usuarios |

| `/root` | Carpeta del superusuario (root) |

| `/var` | Archivos variables como logs y correo |

| `/usr` | Aplicaciones y utilidades del sistema |

| `/dev` | Dispositivos conectados

Diagrama

El contenido generado por IA puede ser incorrecto.

**Terminal y comandos básicos**

Linux se maneja mucho desde la línea de comandos (CLI). Algunos comandos básicos:

**1. Navegación por el sistema de archivos**

| Comando | Función | Ejemplo |

|---------------|--------------------------------------------------|-----------------------------------|

| `pwd` | Muestra el directorio actual | `pwd` |

| `cd` | Cambia de directorio | `cd Documentos` |

| `cd ..` | Sube un nivel (al directorio padre) | `cd ..` |

| `cd /` | Va al directorio raíz | `cd /` |

| `ls` | Lista archivos y carpetas del directorio actual | `ls` |

| `ls -l` | Lista con detalles (permisos, tamaño, fecha) | `ls -l` |

| `ls -a` | Muestra también archivos ocultos | `ls -a` |

**2. Gestión de archivos y carpetas**

| Comando | Función | Ejemplo |

|---------------------|------------------------------------------------|------------------------------------|

| `mkdir` | Crea una nueva carpeta | `mkdir proyecto` |

| `rmdir` | Elimina una carpeta vacía | `rmdir carpeta\_vacía` |

| `rm -r` | Elimina una carpeta con todo su contenido | `rm -r carpeta` |

| `touch` | Crea un archivo vacío | `touch notas.txt` |

| `cp` | Copia archivos o carpetas | `cp archivo.txt copia.txt` |

| `mv` | Mueve o renombra archivos | `mv archivo.txt carpeta/` |

| `rm` | Elimina archivos | `rm archivo.txt` |

**3. Ver contenido de archivos**

| Comando | Función | Ejemplo |

|-------------|---------------------------------------------|----------------------------------|

| `cat` | Muestra el contenido completo | `cat archivo.txt` |

| `less` | Muestra contenido largo, permite desplazarse | `less archivo.txt` |

| `head` | Muestra las primeras líneas | `head archivo.txt` |

| `tail` | Muestra las últimas líneas | `tail archivo.txt` |

**4. Edición rápida de archivos (modo texto)**

| Comando | Función | Ejemplo |

|-------------|-----------------------------------------------------------|---------------------------------|

| `nano` | Editor de texto en terminal (fácil de usar) | `nano archivo.txt` |

| `vim` | Editor más avanzado (opcional para usuarios avanzados) | `vim archivo.txt` |

| `echo` | Imprime texto en pantalla o lo escribe en un archivo | `echo "Hola mundo" > hola.txt` |

echo > guarda el texto pero sobrescribe el texto existente

echo >> Añade al final

**5. Gestión de usuarios y permisos**

| Comando | Función | Ejemplo |

|---------------|--------------------------------------------|----------------------------------|

| `whoami` | Muestra el nombre del usuario actual | `whoami` |

| `id` | Muestra información del usuario y grupos | `id` |

| `chmod` | Cambia permisos de un archivo | `chmod 755 archivo.sh` |

| `chown` | Cambia el propietario del archivo | `chown usuario archivo.txt` |

| `ls -l` | Muestra permisos y propietarios | `ls -l` |

**6. Información del sistema**

| **Comando** | **Función** | **Ejemplo** |
| --- | --- | --- |
| uname -a | Información del kernel y sistema | uname -a |
| df -h | Uso del disco | df -h |
| free -h | Uso de memoria RAM | free -h |
| top | Muestra procesos activos | top |
| uptime | Tiempo encendido del sistema | uptime |

**7. Ayuda y manual**

| **Comando** | **Función** | **Ejemplo** |
| --- | --- | --- |
| man | Muestra el manual de un comando | man ls |
| --help | Muestra ayuda rápida de un comando | ls --help |
| history | Lista comandos ejecutados previamente | history |
| clear | Limpia la terminal | clear |

Actividad práctica: 🧱 "Construí tu barrio

Deben crear una estructura de carpetas como si fuera Avellaneda añadiendo localidades algunas localidades importantes, etc. Creando tmb archivos para introducir información adicional de las locaciones

Avellaneda

├──Avellaneda centro/

├── Wilde/

├── Dock Sud/

└── Gerli/

├── Piñeyro/

├── Sarandí/

└── Villa Domínico/

EJEMPLO

mkdir avellaneda

cd aventura

touch plazaAlsina.txt

echo "Está ubicada en el corazón del centro de Avellaneda, delimitada por la Av. Mitre y las calles Lavalle, San Martín y Alsina. Fue creada el 9 de agosto de 1877." > pista1.txt

mkdir wilde

cd wilde

touch estacionDeWilde.txt

**Permisos y usuarios**

* Cada archivo/directorio tiene **dueño**, **grupo** y **permisos**.
* Permisos: **r = lectura**, **w = escritura**, **x = ejecución**
* Se visualizan con ls -l
* Se modifican con chmod y chown