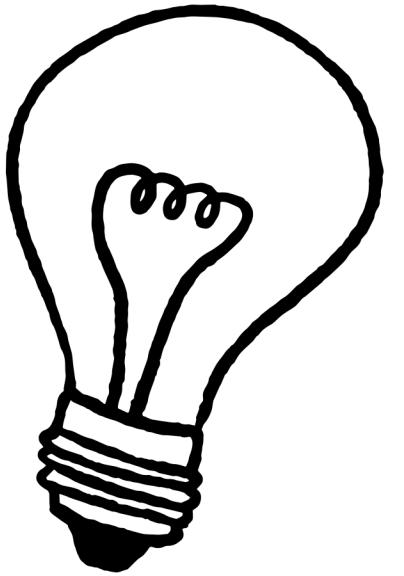
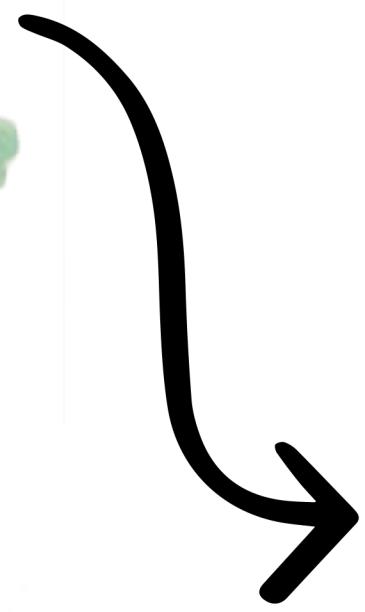




# SISTEMAS OPERATIVOS



Por: karen Chacón

# INTRODUCCIÓN

Un sistema operativo es un software que actúa como intermediario entre el hardware de una computadora y los programas de aplicación. Se encarga de gestionar los recursos del sistema, como el procesador, la memoria, los dispositivos de almacenamiento y los periféricos, para que los programas puedan ejecutarse de manera eficiente. Además, proporciona una interfaz de usuario que permite a los usuarios interactuar con la computadora y ejecutar aplicaciones de manera fácil y cómoda.

## IMPORTANCIA DE LOS SISTEMAS OPERATIVOS

Los sistemas operativos son fundamentales en la informática moderna por varias razones:

- Permiten la gestión eficiente de los recursos del sistema, lo que maximiza el rendimiento y la utilización de la computadora.
- Proporcionan una interfaz de usuario intuitiva que facilita la interacción con la computadora y el uso de programas y aplicaciones.
- Garantizan la compatibilidad y la interoperabilidad entre hardware y software de diferentes fabricantes.
- Facilitan la ejecución de múltiples programas de manera simultánea, lo que permite la multitarea y aumenta la productividad.
- Proporcionan funciones de seguridad para proteger el sistema y los datos contra amenazas externas e internas. En resumen, los sistemas operativos son el corazón de cualquier dispositivo informático y son indispensables para su funcionamiento adecuado y eficiente.

# OBJETIVO DE LA PRESENTACIÓN

El objetivo de esta presentación es proporcionar una visión general de los sistemas operativos, explicando qué son, por qué son importantes en la informática moderna y cómo funcionan. Se abordarán aspectos clave como funciones, características, tipos, evolución y tendencias futuras de los sistemas operativos. Además, se buscará ofrecer ejemplos prácticos y casos de uso para ilustrar su relevancia en el mundo de la tecnología.

# FUNCIONES Y CARACTERÍSTICAS PRINCIPALES:

## 1. Gestión de recursos:

- Administración de la CPU: Asigna tiempo de procesador a los diferentes procesos en ejecución y gestiona la ejecución de programas.
- Gestión de memoria: Controla el acceso a la memoria RAM, asigna y libera espacio de memoria para programas y gestiona la memoria virtual.
- Gestión de dispositivos: Controla la comunicación entre la CPU y los dispositivos de entrada/salida, administrando la transferencia de datos entre ellos.

## 2. Interfaz de usuario:

- Interfaz de línea de comandos (CLI): Proporciona un entorno basado en texto donde los usuarios pueden interactuar con el sistema a través de comandos.
- Interfaz gráfica de usuario (GUI): Ofrece una interfaz visual que permite a los usuarios interactuar con el sistema a través de ventanas, iconos y menús.

## 3. Gestión de archivos:

- Sistema de archivos: Organiza y gestiona la información almacenada en el disco, permitiendo la creación, modificación, eliminación y búsqueda de archivos.
- Directorios: Estructura jerárquica que organiza los archivos en el sistema de archivos, facilitando su gestión y ubicación.

## 4. Comunicación y red:

- Protocolos de red: Proporciona soporte para la comunicación entre dispositivos en una red, incluida la configuración de conexiones, el intercambio de archivos y la gestión de recursos compartidos.
- Servicios de red: Ofrece servicios como DHCP, DNS, servidor de archivos, firewall, etc., para garantizar la conectividad y la seguridad de la red.

## 5. Gestión de procesos:

- Planificación de procesos: Decide qué proceso se ejecutará a continuación y durante cuánto tiempo, utilizando algoritmos de planificación de CPU.
- Creación y terminación de procesos: Permite la creación y finalización de procesos, así como la gestión de su estado y recursos asociados.

## 6. Seguridad:

- Control de acceso: Determina quién tiene permiso para acceder a recursos específicos del sistema, como archivos, directorios y dispositivos.
- Autenticación: Verifica la identidad de los usuarios mediante credenciales como nombres de usuario y contraseñas.
- Protección de datos: Garantiza la integridad y confidencialidad de los datos almacenados en el sistema mediante técnicas como la encriptación y el control de acceso.

# TIPOS DE SISTEMAS OPERATIVOS:

## 1. Computadoras personales (PCs):

- Ejemplos: Microsoft Windows, macOS, Linux.

## 2. Servidores:

- Ejemplos: Windows Server, Linux (Ubuntu Server, CentOS), Unix (AIX, Solaris).

## 3. Dispositivos móviles:

- Ejemplos: Android, iOS.

## 4. Dispositivos embebidos:

- Ejemplos: Embedded Linux, FreeRTOS, Windows IoT.

## 5. Consolas de videojuegos:

- Ejemplos: Xbox OS, PlayStation OS

# ARQUITECTURA DE UN SISTEMA OPERATIVO

## Estructura interna:

- Núcleo (kernel): Gestiona los recursos del hardware y proporciona servicios básicos.
- Gestor de procesos: Controla la ejecución de procesos y la asignación de recursos.
- Gestor de memoria: Administra la memoria disponible y la asignación de memoria a procesos.
- Gestor de archivos: Organiza y controla el acceso a los archivos en el sistema de archivos.
- Otros componentes: Scheduler, gestor de dispositivos, gestor de red, entre otros.

# INTERFAZ DE USUARIO:

1. Tipos de interfaces de usuario:
2. Interfaz gráfica de usuario [GUI].
3. Interfaz de línea de comandos [CLI].
4. Interfaces táctiles y gestuales.
5. Ejemplos de interfaces de usuario en diferentes sistemas operativos:
  - 6. Windows: Interfaz gráfica con ventanas, iconos y menús.
  - 7. macOS: Interfaz intuitiva con dock, barra de menú y Spotlight.
  - 8. Linux: Variedad de entornos de escritorio como GNOME, KDE, etc.

# SEGURIDAD EN SISTEMAS OPERATIVOS

- Importancia de la seguridad:
  - Protección de datos, confidencialidad e integridad.
  - Prevención de accesos no autorizados y ataques informáticos.
- Medidas de seguridad comunes:
  - Control de acceso: Autenticación, autorización y auditoría.
  - Cifrado de datos: Protección de información sensible mediante algoritmos de cifrado.
  - Detección de intrusiones: Monitoreo de actividades anómalas en el sistema

# EVOLUCIÓN Y TENDENCIAS FUTURAS

- Evolución histórica:
  - Desde los primeros sistemas simples hasta los complejos sistemas modernos.
- Tendencias futuras:
  - Virtualización, contenerización y computación en la nube.
  - Internet de las Cosas (IoT) y sistemas conectados.
  - Inteligencia Artificial (IA) y automatización.

# EJEMPLOS

- Interfaz de Usuario:
  - Breve demostración de cómo interactuar con diferentes interfaces de usuario, como Windows, macOS y Linux.
  - Ejemplo: Abrir y cerrar aplicaciones, navegar por el menú de inicio o el dock, y ejecutar comandos en la terminal.
- Configuración de Red:
  - Configurar una conexión de red en diferentes sistemas operativos, como Windows, macOS y Linux.
  - Ejemplo: Configurar una conexión Wi-Fi desde el centro de redes y recursos compartidos en Windows.
- Seguridad:
  - Ejemplificar medidas de seguridad como configuración de firewall en Windows, habilitar el cifrado de disco en macOS y actualizaciones automáticas en Linux.
- Ejemplos de Casos Prácticos:
  - Mostrar casos prácticos como el desarrollo de software en Visual Studio, configuración de servidores web en Linux y automatización de tareas mediante scripts.
- Virtualización:
  - Demostrar la ejecución de varios sistemas operativos en una misma máquina utilizando software de virtualización.
- Ejemplos en Dispositivos Móviles:
  - Ejemplificar el uso de aplicaciones como Google Maps en Android y Siri en iOS para ilustrar las capacidades de los sistemas operativos móviles.
-