**SISTEMAS OPERATIVOS:  
Es el soporte lógico que controla el funcionamiento del equipo físico.** Desde el punto de vista usuario, es un conjunto de programas y funciones que ocultan los detalles del hardware, ofreciendo al usuario una vía sencilla y flexible de acceso al mismo.Software que comprende un conjunto de programas. El sistema operativo administra los recursos ofrecidos por el hardware y actúa como un intermediario entre la computadora y su usuario, ofreciéndole un entorno amigable y sencillo de interpretar.  
Software de comunicación usuario-dispositivo.  
Controlador de hardware del sistema.  
Un sistema operativo empieza a funcionar en el momento que encendemos nuestro dispositivo. Y deja de funcionar cuando lo apagamos. Podemos encontrar sistemas operativos en gran parte de los aparatos tecnológicos que usan un microprocesador para funcionar, como puede ser: computadoras, celulares, heladeras inteligentes. Por ejemplo, cuando accedemos a Instagram desde nuestros celulares estamos utilizando la interfaz del sistema operativo porque es este quien les da las ordenes al procesador para que lo ejecute. El sistema operativo también administra los dispositivos de entrada y salida, la cola de procesos y los recursos del dispositivo.   
Servidores: en ellos también encontramos sistemas operativos mayormente heredados del sistema operativo UNIX como Red Hat, Window server. ¿En que se diferencian? En su uso. Los sistemas operativos de servidores son multiusuarios, es decir, varios usuarios están conectados al mismo tiempo trabajando sobre el mismo núcleo. En cambio, en las computadoras domésticas, tienden a ser monousuarios.   
Clasificación de los SO según su tipo de licencia:  
-Open source: permiten modificar, usar y adaptar un sistema operativo a voluntad del usuario. Ubuntu y Red hat.  
-Proprietary software: son de propietarios y no permiten modificaciones, como Windows. Limitaciones del propietario.

**Recursos** administrados por el sistema operativo:

- **Gestionar la memoria** de acceso aleatorio y ejecutar las aplicaciones, designando los recursos necesarios.  
- **Administrar la CPU,** gracias al algoritmo de programación.  
- **Direccionar las entradas y salida**s de datos (a través de drives), por medio de los periféricos de entrada y salida.  
- **Administrar la información** para el buen funcionamiento de la PC.  
- **Dirigir las autorizaciones** de uso para el usuario.  
- **Administrar los archivos**.

Los tipos de sistemas operativos varían según el hardware y la función de cada dispositivo.  
**\*Según el usuario:**  
-Multiusuario: S.O. que permite que varios usuarios ejecuten simultáneamente sus programas.  
-Monousuario: S.O. que solamente permite ejecutar los programas de un usuario a la vez.  
**\*Según la Gestión de tareas:**  
-Multitarea: S.O. que puede ejecutar varios procesos al mismo tiempo.  
-Monotarea: S.O. que solamente permite ejecutar un proceso a la vez.  
**\*Según la Gestión de recursos:**  
-Centralizado: S.O. solo permite utilizar los recursos de un solo ordenador.  
-Distribuido: S.O. que permite ejecutar los procesos de mas de un ordenador al mismo tiempo.

Generaciones del S.O.:  
-Generacion Cero.  
-Primera Generación.  
-Segunda Generación.  
-Tercera Generación.  
-Cuarta Generación.

**CALIFICACION Y COMPARACION DEL S.O.**

Podemos clasificarlos según:  
**-La administración de tareas:**   
\*Monotarea: Poder realizar una sola tarea a la vez sin que se pueda interrumpir (Windows Me, Windows Vista)  
\*Multitarea: nos permite realizas varias tareas al mismo tiempo. Mucho más comunes.  
**-La administración de usuarios**:  
\*Monousuarios: Soportan a un usuario por vez, no importa cuantos procesadores tenga la computadora o cuantas tareas realice el usuario solo podrá dar servicio a uno. Ejemplo: Todas las versiones de Windows para computadores domésticas.  
\*Multiusuarios: Pueden dar servicio a varios usuarios al mismo tiempo ya sea por medio de terminales conectadas a la computadora o por sesiones remotas en una red de comunicaciones. Ejemplos: Unix, Linux o Solaris.  
-**La organización/estructura interna del S.O.**  
\*Monolítica: Constituida por un solo programa compuesto de una serie de rutinas entrelazadas entre si de tal forma que pueden comunicarse entre ellas. Estos sistemas operativos suelen estar hecho a medidas por lo que son muy rápidos, pero no tienen flexibilidad para soportar diferentes tipos de aplicaciones.   
\*Estructura jerárquica: A medida que fueron creciendo las necesidades de los usuarios y se perfeccionaron los sistemas se hizo necesario una mayor organización del software del S.O. donde una parte del sistema contenía subpartes y esto organizado en forma de niveles. Subdivide en capas o anillos perfectamente definidos y con una clara interfaz con respecto al resto de los recursos.  
\*Máquina Virtual: separan dos conceptos que suelen estar unidos en los otros sistemas. La multiprogramación y la maquina extendida. Los objetivos es el integrar distintos S.O. dando la sensación de ser varias maquinas diferentes.  
\*Cliente-servidor: Sirve para toda clase de aplicaciones. Por lo tanto, es de propósito general y cumplen con las mismas actividades que los S.O. convencionales. La idea es mantener la visión que tiene un usuario de un computador personal pero la red le permite compartir el espacio del disco o impresora con el fin de optimizar los recursos. Estos S.O. presentan una desventaja: No resuelven los problemas de compartir información lo que dificulta el desarrollo en grupo.  
El sistema operativo se puede elegir según nuestras necesidades. Por ejemplo, lo que necesitamos es una computadora para trabajar desde casa, lo mas recomendable es que elijamos un S.O. monousuario.

**KERNEL**

KERNEL: el cerebro del Sistema operativo. Es una parte del sistema operativo el cual es el encargado de interactuar entre las diferentes aplicaciones y sus necesidades según los recursos que posee para ejecutarlos.  
Capa fundamental de un sistema operativo, es el encargado de comunicar y administrar los recursos de la computadora, como la RAM o el uso del procesador.  
  
APLICACIONES 🡪 KERNEL 🡪 CBU, MEMORIA, DISPOSITIVOS.

Por ejemplo, al hacer clic en guardar un documento, el kernel es quien interactúa con la memoria secundaria para resguardar la información. A su vez, el kernel es quien decida cuando asignar o quitar recursos de hardware a las aplicaciones que se ejecutan en el software. (Kernel asigna y prioriza recursos). Como así también asignan prioridades según las diferentes necesidades del sistema operativo.

¿Como se llevan a cabo estas interacciones?   
A través de las llamadas al sistema. Las llamadas son el método que tienen las aplicaciones para solicitar un servicio o un recurso. Manera en la cual un programa solicita una acción al SO con el que interactúa.  
Esta acción es el punto de enlace entre el modo usuario y el modo privilegiado del SO. Lo que permite a las aplicaciones utilizar recursos de hardware. El objetivo de las llamadas al sistema:  
Es la forma en la que los SO diferencian las acciones que puede realizar un usuario (modo usuario) de las que no ya que algunas pueden llegar a ser muy dañinas para el SO que solo debe controlarlas el mismo (modo privilegiado).  
\*El SO autoriza y administra todas las acciones potencialmente peligrosas.  
\*Un usuario que modifica aspectos claves del SO puede causar grandes danos en el mismo.

Clasificación de llamadas al sistema: Si bien todas trabajan como unidades de control para el SO, se establecieron cinco tipos de llamadas al sistema.  
\*Gestión de control: Supervisa el inicio, creación, detención y finalización de los procesos.  
\*Gestión de archivos: Incluyen la creación, eliminación, apertura, cierre, escritura y lectura de archivos.  
\*Gestión de dispositivos: Administra los recursos disponibles, como ser el almacenamiento.  
\*Gestión de información: Asegura la puntualidad e integridad de la información.  
\*Comunicación entre procesos: Coordina la interacción entre los distintos procesos y aplicaciones.

El kernel se encuentra alojado dentro del SO y es una parte esencial del mismo. Tanto Windows como Linux o Mac tienen su propio Kernel, y hay otros sistemas operativos que utilizan o se basan en kernels ya creados, como por ejemplo Android, ¿y el kerral de linus?  
Existen varios modelos (o tipos de estructura) de Kernel que varían según el creador del sistema operativo. Pero en general hay dos ramas que dividen en dos columnas su metodología de trabajo.

TIPOS DE KERNEL:  
-Monolítico: Con el cual trabaja Linux, es un código de muchas líneas que está alojado en un solo espacio de memoria y que posee todos los drivers, los servicios y los métodos de administración de recursos. Como desventaja de este diseño podemos decir que se desperdicia mucho espacio en memoria porque cuando se carga el kernel completo se cargan drivers y diferentes métodos para todo tipo de dispositivos que el sistema operativo puede operar.   
\*El más veloz ya que se comunica con llamadas al sistema.   
\*El 70% del kernel no es utilizado.  
\*Si un sistema falla, todo el núcleo falla.   
-Microkernel: El cual solo posee las instrucciones básicas de la administración en un pequeño espacio de memoria y deja los diferentes dispositivos su propio manejo. Es decir, se encarga de las tareas más básicas de administración. La desventaja de este modelo es que un microkernel pertenece únicamente a un dispositivo y al ser el kernel una parte fundamental del SO hay que diseñar un sistema operativo por cada dispositivo con su microkernel determinado. El microkernel de una PC de escritorio funciona en una notebook siempre y cuando el hardware sea de las mismas características podrá funcionar correctamente.  
\*Mas lento debido a que se comunica con paso de mensajes.  
\*Mas fácil agregar nuevas funcionalidades.  
\*Requiere más líneas de código.

Existen variantes sobre las ramas del kernel, los cuales buscan explotar aun mas sus fortalezas y mejorar sus debilidades.  
-Kernel hibrido:  
\*En esencia es un microkernel con más código “no esencial”, pero menor al de un monolítico puro.  
\*Agiliza la velocidad de un microkernel  
\*Compatible para gran variedad de dispositivos.  
-Nanokernel:  
\*El código es aun mas reducido que en microkernel, pero más difícil crear.  
\*Todos los servicios se comunican con paso de mensajes.  
\*Fácil modificación del sistema operativo.

Cuando un dispositivo o proceso falla, la función del kernel es interrumpir todo lo que está haciendo la computadora para evitar un daño en el sistema operativo.  
Todas las acciones que lleva a cabo una computadora pasan a través de un kernel y la performance de un sistema operativo dependen mucho del mismo.