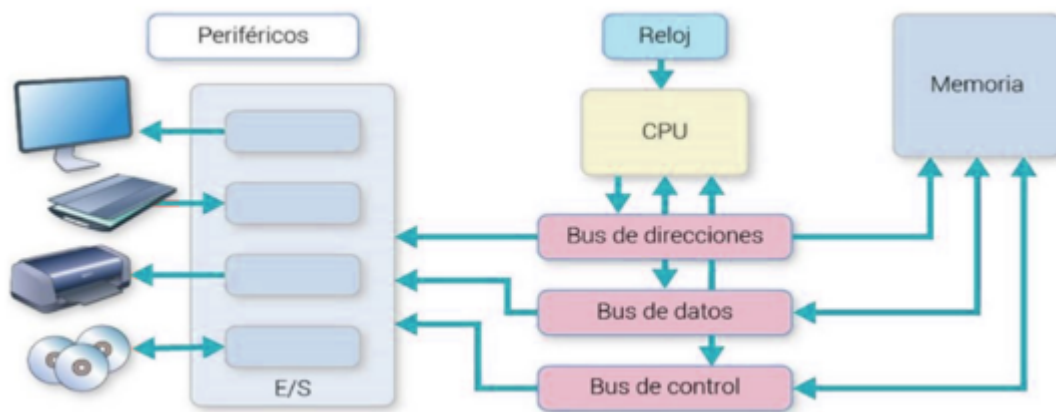


FUNDAMENTOS DE HARDWARE

0371

Administración de Sistemas
Informáticos en Red





Sistemas Informáticos. Estructura Funcional

Contenidos

■ 1. Esquema y estructura de un ordenador	3
■ 2. Elementos funcionales y subsistemas	7
■ 3. Tipos de aplicaciones y licencias	10
■ 4. Aplicaciones de propósito general	12
■ 5. Ejercicios	15
■ 6. Test de conocimientos	16

1. Esquema y estructura de un ordenador

1.1. El sistema informático

En este apartado aprenderás sobre los componentes principales de un sistema informático, cómo se organizan y su interacción para lograr el procesamiento de información. Un **sistema informático** puede ser cualquier dispositivo que procese información, como **teléfonos inteligentes, tabletas y ordenadores personales**.

El **hardware** es la parte tangible del ordenador, lo que puedes tocar. Piensa en la carcasa, el monitor, el teclado, el ratón... todos estos elementos son hardware. Dentro de la carcasa, encontramos componentes esenciales como:

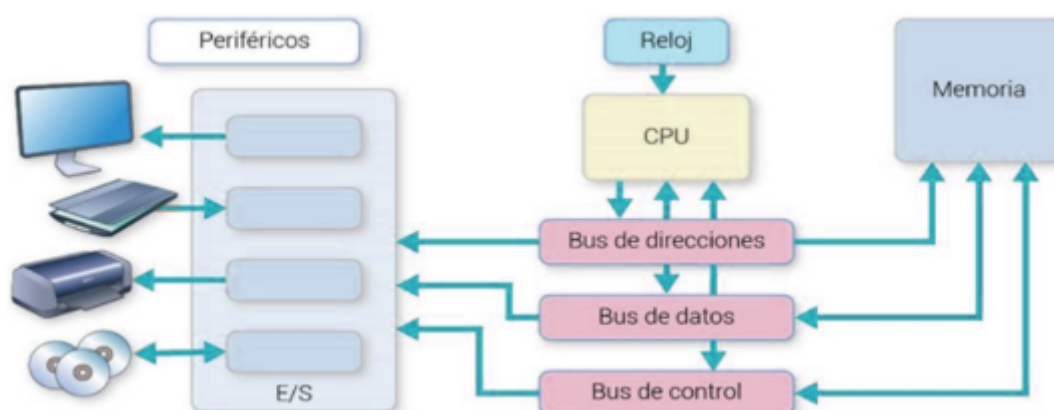
- **La placa base:** Es como la columna vertebral del ordenador, donde se conectan todos los demás componentes.
- **El procesador (CPU):** Es el cerebro del ordenador, el encargado de procesar la información y ejecutar las instrucciones.
- **La memoria RAM:** Es como la memoria a corto plazo del ordenador, donde se almacenan los datos que se están utilizando en ese momento.
- **El disco duro:** Es la memoria a largo plazo, donde se almacenan los programas, el sistema operativo y tus archivos personales.

Si el hardware es el cuerpo del ordenador, el **software** es su alma. El software son las instrucciones que le dicen al hardware qué hacer. Podemos distinguir diferentes tipos de software, como:

- **Sistemas operativos:** Windows, macOS o Linux son ejemplos de sistemas operativos. Son como el director de orquesta, que se encarga de que todos los componentes del ordenador funcionen correctamente.
- **Aplicaciones:** Son programas que nos permiten realizar tareas específicas, como navegar por internet (Chrome, Firefox), escribir documentos (Word, Google Docs) o editar fotos (Photoshop).
- **Juegos:** ¡El software también nos permite divertirnos! Los videojuegos son programas que nos permiten interactuar con mundos virtuales.

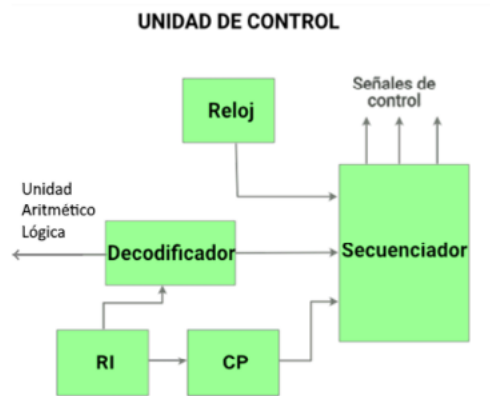
1.2. Arquitectura de Von Neumann

La **arquitectura de Von Neumann** es un modelo clásico de organización de ordenadores, propuesta en 1945 por **John von Neumann**, que todavía es usado en sistemas informáticos modernos. Esta arquitectura organiza el ordenador en cuatro partes fundamentales: **CPU, memoria, buses y dispositivos de entrada/salida (E/S)**. Su principal característica es el uso de un mismo **almacenamiento** para los datos y las instrucciones.

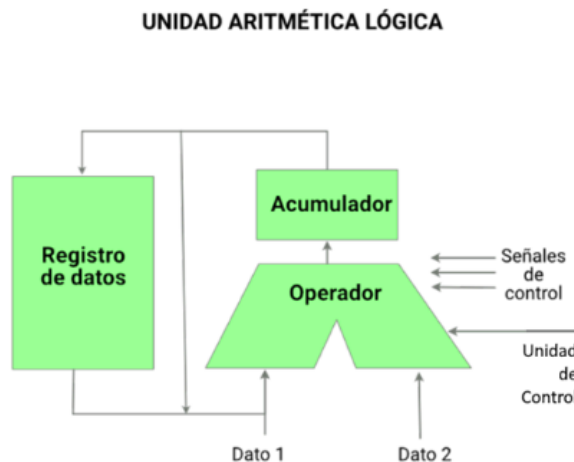


1.2.1. Componentes de la arquitectura

- **CPU (Unidad Central de Proceso):** Como ya hemos visto, la CPU es el cerebro del ordenador. Pero, ¿sabías que la CPU está compuesta por diferentes partes?
- **Unidad de Control (UC):** Es la que se encarga de dirigir y coordinar todas las operaciones del sistema. Es como el director de orquesta de la CPU.



- **Unidad Aritmético-Lógica (ALU):** Es la que se encarga de realizar las operaciones matemáticas y lógicas. Es como la calculadora del ordenador.

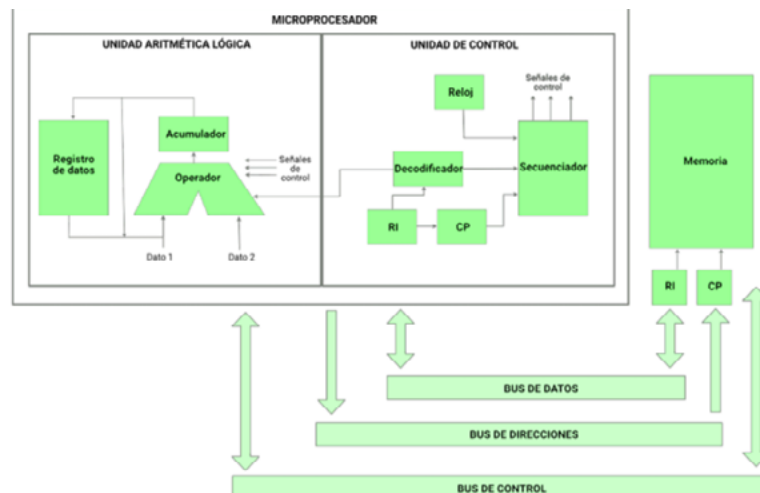


- **Registros:** Son pequeñas memorias que se encuentran dentro de la CPU y que se utilizan para almacenar datos e instrucciones temporalmente.
- **Memoria central:** Es donde se almacenan los programas y los datos que se están utilizando en ese momento. Es como la mesa de trabajo del ordenador.
- **Buses:** Son los canales de comunicación entre los diferentes componentes del ordenador. Son como las carreteras que conectan las diferentes ciudades.
- **Sistema de Entrada/Salida (E/S):** Son los dispositivos que permiten al ordenador interactuar con el mundo exterior, como el teclado, el ratón, el monitor, la impresora, etc.

1.2.2. Funcionamiento

La **arquitectura de Von Neumann** funciona en un ciclo de búsqueda-ejecución. Imagina que la CPU es un chef que está siguiendo una receta:

1. **Búsqueda:** El chef busca la siguiente instrucción de la receta (la CPU obtiene una instrucción de la memoria).
2. **Decodificación:** El chef lee la instrucción y la entiende (la CPU interpreta la instrucción).
3. **Ejecución:** El chef realiza la acción que indica la instrucción (la CPU realiza la operación).
4. **Almacenamiento:** El chef guarda el resultado de la acción (el resultado se guarda en la memoria).



1.2.3. Memorias

La **memoria** del ordenador es como un gran almacén donde se guarda la información. Existen diferentes tipos de memoria.



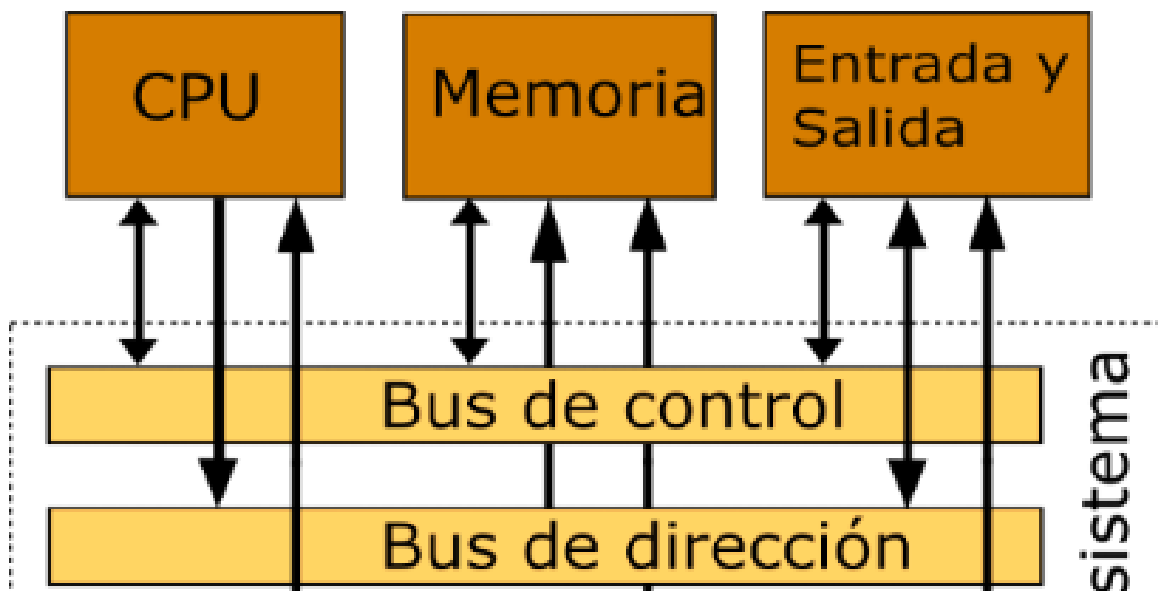
• Memoria Interna:

- **Registros:** Son las memorias más pequeñas y rápidas del ordenador, y se encuentran dentro de la CPU.
- **Memoria Caché:** Es una memoria intermedia entre la CPU y la memoria RAM. Almacena los datos que se utilizan con más frecuencia para que la CPU pueda acceder a ellos más rápidamente.
- **Memoria Principal (RAM):** Es la memoria principal del ordenador, donde se almacenan los programas y los datos que se están utilizando en ese momento. Es una memoria volátil, lo que significa que la información se pierde cuando se apaga el ordenador.
- **Memoria Secundaria:** Es la memoria que se utiliza para almacenar la información de forma permanente, incluso cuando el ordenador está apagado. Ejemplos de memoria secundaria son los discos duros, los SSD, las memorias USB, etc.

1.2.4. Buses

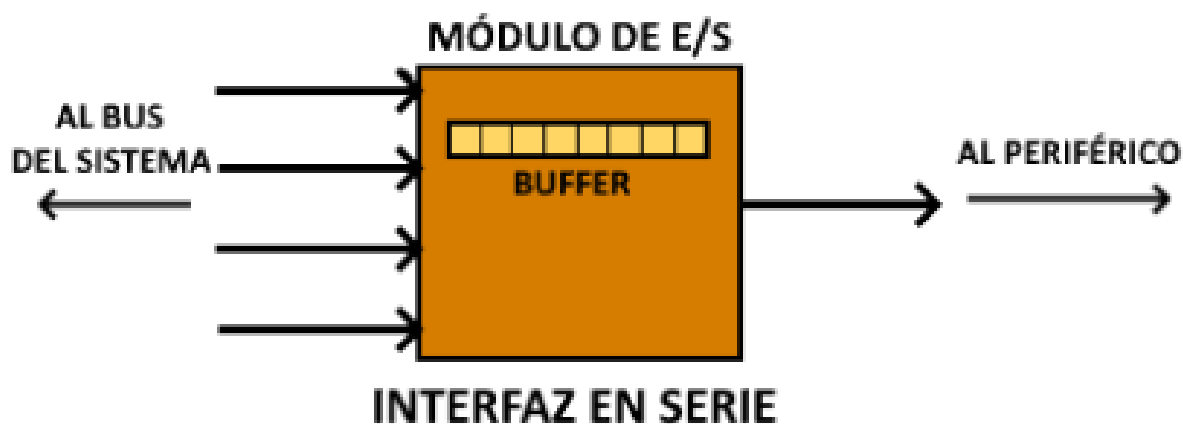
Los **buses** son las autopistas de la información dentro del ordenador. Son canales de comunicación que permiten que los diferentes componentes se comuniquen entre sí. Existen diferentes tipos de buses.

- **Bus de Datos:** Es el bus por donde viajan los datos.
- **Bus de Direcciones:** Es el bus por donde viajan las direcciones de memoria.
- **Bus de Control:** Es el bus por donde viajan las señales de control.



1.2.5. Interfaces

Una **interfaz** es como un traductor que permite que dos dispositivos se comuniquen entre sí. Por ejemplo, la interfaz USB permite que conectes tu memoria USB al ordenador. Existen diferentes tipos de interfaces.



- **Interfaces Paralelas:** Transmiten varios bits de información a la vez, como si fueran varias personas hablando al mismo tiempo.
- **Interfaces Seriales:** Transmiten un bit de información a la vez, como si fuera una sola persona hablando.

1.2.6. Sistemas de Entrada y Salida (E/S)

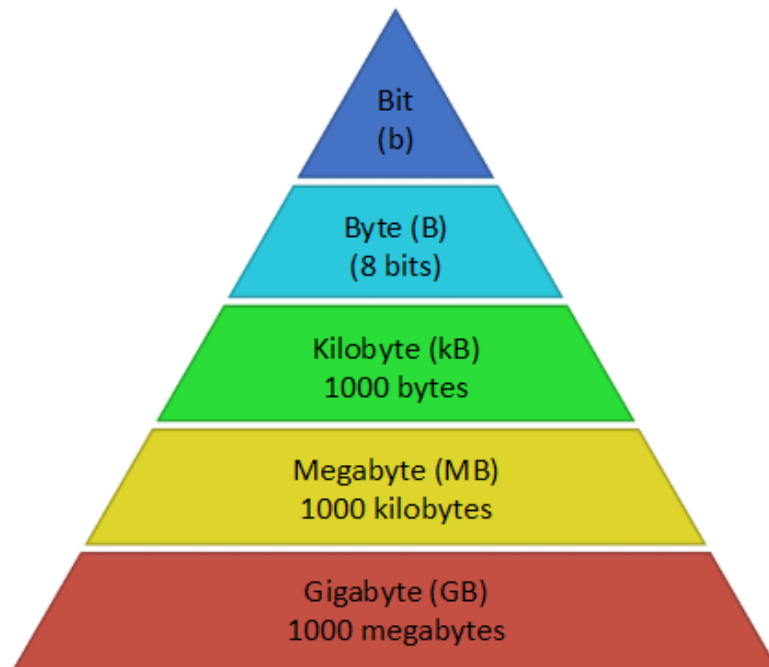
Los sistemas de E/S son los que permiten al ordenador interactuar con el mundo exterior. Son como los sentidos del ordenador. Algunos ejemplos son:

- **El teclado:** Permite introducir texto en el ordenador.
- **El ratón:** Permite controlar el cursor en la pantalla.
- **El monitor:** Permite visualizar la información.
- **La impresora:** Permite imprimir documentos.
- **El escáner:** Permite digitalizar documentos.
- **La webcam:** Permite capturar imágenes y vídeo.
- **El micrófono:** Permite grabar sonido.
- **Los altavoces:** Permiten escuchar sonido.

2. Elementos funcionales y subsistemas

2.1. Elementos de Almacenamiento

La memoria de un sistema informático está organizada en una jerarquía que varía desde registros dentro de la CPU hasta el almacenamiento en la nube, con el objetivo de almacenar datos de manera eficiente y accesible.



Tipos de Memoria:

- **Registros:** Son celdas de memoria dentro de la CPU utilizadas para operaciones rápidas. Son de alta velocidad pero de baja capacidad, y su tamaño depende de la arquitectura del procesador (por ejemplo, 64 bits en procesadores modernos).
- **Memoria Principal (RAM):** Almacena los datos y programas que la CPU necesita mientras realiza operaciones. Tiene mayor capacidad que los registros, pero es más lenta. Se divide en memoria volátil (RAM dinámica y estática) que pierde su contenido cuando el sistema se apaga.
- **Memoria Secundaria:** Es el almacenamiento permanente del sistema, utilizado para guardar grandes cantidades de datos incluso cuando el ordenador está apagado. Ejemplos incluyen discos duros (HDD), unidades de estado sólido (SSD) y discos ópticos.
- **Memoria Terciaria:** Usada principalmente para almacenamiento masivo de datos a largo plazo, como en el caso de cintas magnéticas o almacenamiento en la nube. Su acceso es más lento pero económico y confiable para el archivo de grandes volúmenes de datos.

Unidades de Medida:

- **Bit (b):** Unidad básica de información, representando 0 o 1.
- **Byte (B):** Compuesto por 8 bits, es la unidad fundamental de almacenamiento.
- **Kilobyte (KB):** 1,000 bytes.
- **Megabyte (MB):** 1,000 kilobytes.
- **Gigabyte (GB):** 1,000 megabytes.
- **Terabyte (TB):** 1,000 gigabytes.

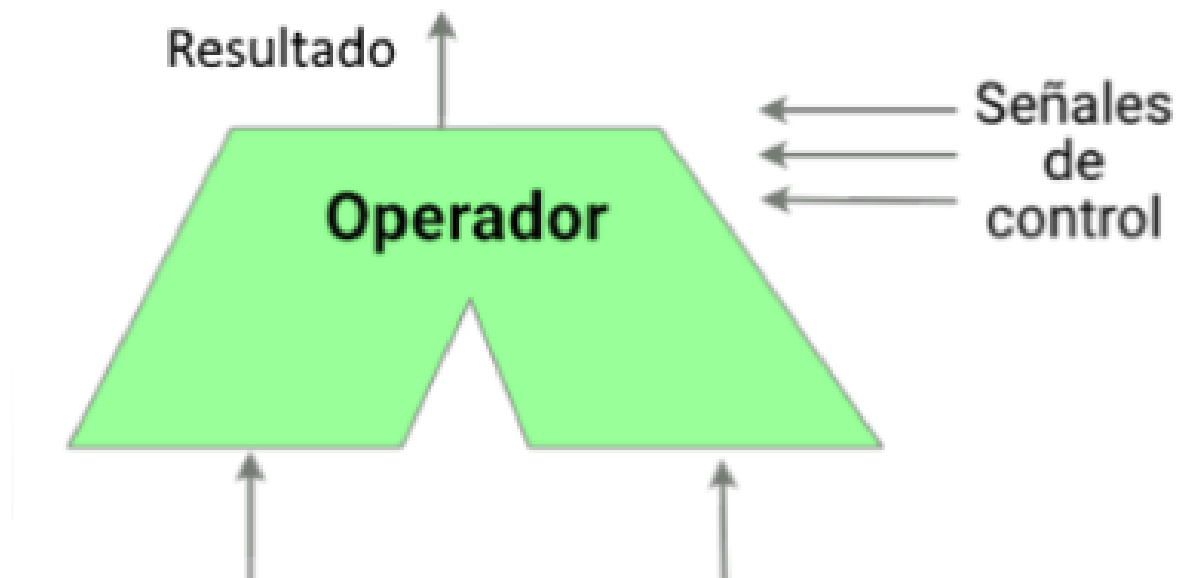
2.2. Mecanismos de direccionamiento

El direccionamiento es el proceso que permite localizar y acceder a la memoria de manera eficiente. Existen distintos métodos según la arquitectura de la memoria.

- **Direccionamiento Cableado:** Cada dirección se asigna a un único punto de memoria mediante señales directas. Este tipo de direccionamiento es rápido pero inflexible y se utiliza en sistemas con recursos limitados.
- **Direccionamiento 2D:** La memoria está organizada en una matriz bidimensional. Cada palabra se encuentra en una fila y columna, lo que permite un acceso rápido a las posiciones cercanas.
- **Direccionamiento 3D:** Utiliza una estructura de cubo para organizar las celdas de memoria en tres dimensiones. Este tipo de direccionamiento es útil en sistemas con grandes volúmenes de datos y acceso paralelo.
- **Memorias Asociativas (CAM):** En este tipo de memoria, la búsqueda se realiza por contenido, no por dirección. Se utilizan en sistemas que requieren acceso rápido y flexible, como en la traducción de direcciones de memoria en caché o en redes para almacenar direcciones MAC.

2.3. Elementos de Operación

Los elementos de operación ejecutan cálculos aritméticos, lógicos y de control dentro del sistema. Son esenciales para el procesamiento de datos y la toma de decisiones dentro de los circuitos del ordenador.



Clasificación de los Operadores

- **Por su ámbito:**

- **Generales:** Realizan operaciones aritméticas, lógicas y de comparación.
- **Específicos:** Están diseñados para realizar una operación particular, como multiplicación o división.

- **Por su electrónica:**

- **Combinacionales:** Su salida depende únicamente de las entradas actuales, sin necesidad de memoria.
- **Secuenciales:** Su salida depende no solo de las entradas, sino también del estado anterior de la operación, lo que requiere memoria.

- **Por el número de operandos:**

- **Monádicos:** Requieren un solo operando, como las operaciones de negación.
- **Diádicos:** Requieren dos operandos, como las operaciones de suma o resta.

- **Por el número de líneas:**

- **Paralelos:** Procesan todos los bits del operando a la vez, lo que permite un cálculo más rápido.
- **Serie:** Procesan un bit a la vez, lo que es más lento pero más simple y eficiente en términos de recursos.

2.4. Elementos de interconexión: Buses y Canales

Los elementos de interconexión son los que permiten la comunicación entre los diferentes componentes del ordenador. Son como las carreteras y las vías de tren que conectan las diferentes ciudades.

- **Buses:** Son como autopistas de la información, donde varios componentes comparten el mismo canal de comunicación.
- **Canales:** Son como vías de tren de alta velocidad, donde la comunicación es directa entre dos componentes específicos.

3. Tipos de aplicaciones y licencias

3.1. Tipos de aplicaciones: según el tipo de trabajo

Podemos clasificar las aplicaciones en función del tipo de trabajo que realizan:

- **Software de sistema:** Es esencial para la interacción entre el hardware y el usuario, actuando como intermediario. Incluye sistemas operativos y firmware, que gestionan los recursos del dispositivo y permiten la ejecución de otras aplicaciones. Ejemplos: **Windows, Ubuntu y macOS.**



- **Software de aplicación:** Son herramientas diseñadas para ayudar a los usuarios a realizar tareas específicas y automatizables. Son programas de uso diario, como procesadores de texto, navegadores o reproductores multimedia. Ejemplos: **Microsoft Office, Google Chrome, VLC.**



- **Software de programación:** Facilita la escritura y edición de código, soportando varios lenguajes de programación. Incluye herramientas como **editores de código, compiladores y depuradores.** Ejemplos: **Visual Studio Code, Android Studio.**



- **Software de virtualización:** Permite crear una capa de abstracción entre el hardware físico y el software, facilitando la ejecución de múltiples sistemas operativos en un solo dispositivo. Ejemplos: **Hyper-V, VMware Workstation.**



3.2. Tipos de aplicaciones: según la distribución y la licencia

Según la distribución y la licencia, el software se clasifica de la siguiente manera:

- **Software propietario:** El código fuente está cerrado y su acceso está restringido. A menudo se utiliza con fines comerciales, pero no necesariamente debe ser de pago. Ejemplo: **Google Chrome, Microsoft Office.**
- **Software libre:** Iniciado por **Richard Stallman**, fundador del proyecto **GNU**, permite a los usuarios ejecutar, estudiar, modificar y distribuir el software de forma libre. Se basa en cuatro libertades esenciales:

1. Ejecutar el programa para cualquier propósito.
2. Estudiar y modificar el programa.
3. Redistribuir copias del programa.
4. Distribuir versiones modificadas del programa.

Ejemplos: **Linux, LibreOffice, Firefox.**

- **Software de código abierto:** Similar al software libre, pero con licencias menos restrictivas. Fomenta el desarrollo colaborativo y permite la modificación y distribución del código fuente. Ejemplos: **Android, Chromium, Python.**

3.3. Tipos de licencias

Las licencias de software son acuerdos legales que determinan los términos de uso. Algunas de las más comunes son:

- **Dominio público:** El software no tiene restricciones de derechos de autor y puede ser usado, modificado y distribuido libremente.
- **Shareware:** Se ofrece gratuitamente por un tiempo limitado para evaluación. Después, se requiere pagar para continuar usándolo.
- **Freeware:** Se distribuye de manera gratuita y sin limitaciones de tiempo, pero no permite modificar ni redistribuir el software.
- **Adware:** Se ofrece de forma gratuita, pero incluye publicidad que puede ser molesta para el usuario.
- **Software de demostración:** Tiene funciones limitadas para permitir a los usuarios probarlo antes de adquirir la versión completa.
- **Copyleft:** Las versiones modificadas del software deben mantener la misma licencia que el original, garantizando que continúe siendo libre.
- **Copyright:** Protege la propiedad intelectual del software mediante derechos exclusivos de uso y distribución.
- **BSD:** Permite la creación de versiones modificadas del software bajo licencias diferentes a la original.
- **GPL (Licencia Pública General):** Exige que las versiones modificadas del software mantengan la misma licencia GPL.
- **MPL (Licencia Pública de Mozilla):** Las modificaciones deben ser comunicadas al autor original y seguir ciertas normas para su distribución.
- **OEM (Fabricante de Equipos Originales):** Licencia que se vende a fabricantes de hardware para ser preinstalada en dispositivos nuevos.
- **Retail:** Licencia que se vende directamente al público y puede ser instalada en varios dispositivos, pero no simultáneamente.

3.4. Tipos de aplicaciones: según su propósito

Según su propósito, las aplicaciones se clasifican de la siguiente forma:

- **Software de uso específico:** Diseñado para resolver problemas concretos en ámbitos especializados. Ejemplos incluyen software de **diseño industrial** (CAD), **simulación** (CAE) o **gestión empresarial** (ERP).
- **Aplicaciones ofimáticas:** Son herramientas utilizadas en tareas de oficina, como la edición de documentos, hojas de cálculo y presentaciones. Ejemplos: **Microsoft Office, LibreOffice, Google Workspace.**
- **Aplicaciones de seguridad:** Están diseñadas para proteger sistemas informáticos de amenazas, como virus y malware. Se dividen en:
 - **Mecanismos activos:** Previenen o eliminan amenazas, como **antivirus** y **firewalls**.
 - **Mecanismos pasivos:** Minimizar los daños tras un ataque, como **copias de seguridad** y sistemas de **cifrado**.

3.5. Criterios de seguridad en sistemas informáticos

Al establecer la seguridad en un sistema informático, se deben considerar los siguientes criterios:

- **Confidencialidad:** Asegurar que la información solo sea accesible para quienes tengan autorización.
- **Integridad:** Garantizar que los datos no sean alterados de manera no autorizada.
- **Disponibilidad:** Asegurar que los sistemas y la información estén disponibles cuando se necesiten.
- **Autenticidad:** Confirmar la identidad de los usuarios y sistemas involucrados.
- **No repudio:** Impide que un usuario niegue acciones realizadas en el sistema.

Establecer mecanismos de protección adecuados para cada uno de estos aspectos es clave para mantener la seguridad y confiabilidad del sistema.

4. Aplicaciones de propósito general

4.1. ¿Qué son?

Definición:

Las aplicaciones de propósito general son herramientas informáticas diseñadas para realizar una variedad de tareas sin estar limitadas a una función específica. Su versatilidad permite que sean adaptadas según las necesidades y preferencias del usuario final.

Contexto:

El sistema operativo actúa como el núcleo que gestiona los recursos de hardware y coordina el funcionamiento de las aplicaciones. Las aplicaciones de propósito general, como herramientas complementarias, extienden las capacidades del sistema operativo, mejorando la experiencia del usuario al ofrecer una gama de funcionalidades personalizables.

4.2. Instalación de Aplicaciones Informáticas

Es fundamental realizar un análisis del entorno antes de la instalación, considerando lo siguiente:

Modelo Cliente-Servidor:

Las aplicaciones pueden estar diseñadas para servidores corporativos o para estaciones de trabajo individuales. Las aplicaciones en servidores requieren configuraciones más complejas y su interacción con redes debe ser bien gestionada.

Consideraciones de Seguridad:

Es importante revisar la seguridad del entorno donde se instalará la aplicación, incluyendo la configuración de firewalls, redes y medidas adicionales de protección.

Requisitos de Hardware:

Antes de instalar una aplicación, se debe verificar la compatibilidad con los componentes esenciales del sistema, como el procesador, memoria RAM y almacenamiento.

Sistema Operativo:

Asegurarse de que la aplicación sea compatible con la versión específica del sistema operativo es crucial para su correcto funcionamiento.

Tipo de Licencia:

Las licencias de software determinan las condiciones de uso de una aplicación. Es importante entender si la licencia es **abierta**, **comercial**, **freeware** o **shareware**, para evitar posibles problemas legales.

4.3. Requisitos Previos para la Instalación

Especificaciones de Hardware:

- **Mínimas:** Requisitos básicos que permiten que la aplicación funcione, aunque con posibles limitaciones en el rendimiento.
- **Recomendadas:** Configuraciones óptimas que garantizan un funcionamiento fluido y eficiente de la aplicación.

Compatibilidad con el Sistema Operativo:

Es esencial asegurarse de que la aplicación sea compatible con el sistema operativo en uso. Por ejemplo, **Final Cut Pro** solo está disponible para **macOS**, mientras que otros programas como **Firefox** son multiplataforma.

Software Adicional:

Algunas aplicaciones requieren software adicional para funcionar correctamente, como:

- **Java Virtual Machine (JVM):** Necesaria para ejecutar aplicaciones basadas en Java.
- **Bibliotecas y componentes adicionales:** Conjuntos de código que la aplicación necesita para realizar sus funciones.

4.4. Métodos de instalación

Los métodos comunes para instalar aplicaciones incluyen:

Versión Portable:

Las aplicaciones portátiles no requieren instalación, ya que se pueden ejecutar directamente desde un directorio. Esto es útil para dispositivos donde no se puede realizar una instalación completa, como en **Notepad++** o **Advanced IP Scanner**.

Uso de Instalador:

Los instaladores automatizan el proceso de configuración y gestión de dependencias, como en los archivos **.exe** en **Windows** o **.dmg** en **macOS**.

Sistemas de Gestión de Paquetes:

Herramientas como **apt** en **Ubuntu** o **Homebrew** en **macOS** permiten instalar, actualizar y eliminar aplicaciones de manera eficiente.

Instalación en Línea:

Las aplicaciones que se descargan e instalan directamente desde Internet, como es común en dispositivos móviles, facilitan la instalación rápida sin necesidad de medios físicos.

4.5. Aplicaciones en la Nube



Las aplicaciones en la nube son accesibles a través de Internet, ofreciendo flexibilidad y escalabilidad. Se consideran **Software as a Service (SaaS)**, donde el proveedor gestiona el software y el hardware.

Ventajas:

- **Licencias Temporales:** Permiten pagar por uso o mediante suscripción, sin necesidad de compra permanente.
- **Ejecución en el Servidor:** Reduce los requisitos y carga del hardware del usuario final.
- **Actualizaciones Automáticas:** El proveedor se encarga de las actualizaciones, garantizando que siempre se disponga de la versión más reciente.

Desventajas:

- **Dependencia del Proveedor:** El servicio depende de la disponibilidad y políticas del proveedor.
- **Infraestructura de Internet:** Necesidad de una conexión estable para un rendimiento adecuado.
- **Seguridad y Privacidad:** Preocupaciones sobre el almacenamiento y manejo de datos sensibles.

Ejemplos de aplicaciones en la nube incluyen **Google Workspace**, **Microsoft 365** y **Netflix**.

4.6. La interfaz de usuario

La interfaz de usuario es el medio a través del cual interactuamos con una aplicación. Los tres tipos principales de interfaz son:

- **Interfaz de Línea de Comandos (CLI):**

Permite interactuar con la aplicación mediante comandos de texto. Proporciona mayor control y es eficiente en tareas repetitivas. Ejemplos incluyen **Terminal** en **Linux** y **PowerShell** en **Windows**.

- **Interfaz Gráfica de Usuario (GUI):**

Utiliza elementos visuales como menús, iconos y ventanas para facilitar la interacción. Este tipo de interfaz es común en sistemas operativos como **Windows** o **macOS**.

- **Interfaz Natural de Usuario (NUI):**

Se basa en gestos, voz y pantallas táctiles para la interacción. Ejemplos incluyen **Siri**, **Alexa** y dispositivos táctiles como **smartphones** y **tablets**.

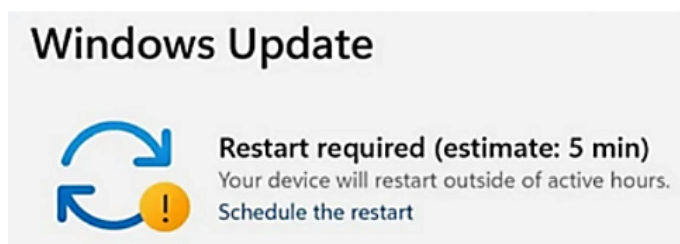
4.7. El mantenimiento de una aplicación

Las aplicaciones requieren actualizaciones periódicas para mejorar su funcionamiento y seguridad. Las actualizaciones pueden ser automáticas o requerir la intervención del usuario.

Motivos para actualizar una aplicación:

- **Actualización del Software:** Introducción de nuevas versiones con mejoras y nuevas funcionalidades.
- **Parches de Depuración:** Solución de problemas detectados durante el uso.
- **Mejoras de Seguridad:** Modificaciones para corregir vulnerabilidades.
- **Extensiones de Idioma:** Adaptación del software a diferentes idiomas según las necesidades del usuario.

Las aplicaciones se pueden clasificar en diversas categorías según su función, diseño y campo de actividad, como ofimática, multimedia, programación, educación, hogar, negocios, etc.



Campo de Actividad	Funcionalidades	Ejemplos
Ofimática y Documentación	Procesadores de texto, hojas de cálculo, bases de datos	LibreOffice, Microsoft Office, Google Suite
Imagen, Diseño y Multimedia	Edición de imágenes, video, sonido	Photoshop, GIMP, Audacity
Programación	Entornos de desarrollo integrados (IDE), editores de código	Visual Studio Code, Eclipse, NetBeans
Educación	Interacción con el alumnado en diversas áreas.	GeoGebra, Moodle, Google Classroom
Hogar y Ocio	Planificación de tareas diarias, entretenimiento.	OurHome, Planifood, Fortnite
Productividad y Negocios	ERP y CRM para la gestión empresarial.	Odoo, Microsoft Dynamics, Espacyo
Clientes para Servicios de Internet	Navegadores, correo electrónico, FTP, VoIP, VPN.	Google Chrome, Thunderbird, WinSCP
Compresores	Empaquetado y cifrado de ficheros.	7zip, WinRAR
Monitorización y Optimización del Sistema	Herramientas de diagnóstico y optimización.	AIDA64, Hwinfo, CrystalDiskInfo
Grabación	Grabación de datos en CD, DVD, Blu-Ray.	Nero, Brasero, ImgBurn
Mantenimiento	Predicción, prevención y corrección de fallos en equipos informáticos.	HD-Tune, Prime95

5. Ejercicios

Ejercicio 1: Define los siguientes términos: Hardware, Software, CPU, ALU, Memoria Cache.

Ejercicio 2: Explica cómo se organiza la jerarquía de memoria en un sistema informático y por qué es importante.

Ejercicio 3: Dibuja un diagrama de la arquitectura de Von Neumann, indicando y etiquetando los siguientes componentes: CPU, Unidad de Control, ALU, Memoria, Buses, Sistema de Entrada y Salida.

Ejercicio 4: En un ordenador abierto, identifica y describe los diferentes tipos de memoria que encuentres (registros, cache, RAM). Explica cómo cada uno contribuye al funcionamiento del sistema.

Ejercicio 5: Realiza una verificación de los buses en un sistema informático utilizando herramientas de diagnóstico de hardware. Describe los pasos que seguiste y los resultados obtenidos. Utiliza software de diagnóstico como HWMonitor o CPU-Z.

Ejercicio 6: Clasificación de Software. Investiga y elabora una lista de diez aplicaciones de software que utilices en tu vida diaria. Clasifica cada aplicación en una de las categorías de software vistas (software base, de aplicación, de programación, de virtualización). Identifica el tipo de licencia de cada aplicación (software propietario, libre, de código abierto, etc.).

Ejercicio 7: Análisis de Licencias de Software. Selecciona tres aplicaciones de software de código abierto (por ejemplo, LibreOffice, GIMP, Mozilla Firefox). Investiga las licencias bajo las cuales se distribuyen (GPL, MPL, BSD, etc.). Compara y contrasta las libertades y restricciones de cada licencia. ¿Cómo afectan estas licencias al uso, modificación y redistribución del software?

Ejercicio 8: Organiza tu espacio de trabajo según las recomendaciones de ergonomía. Ajusta la altura de tu silla y mesa, la distancia de la pantalla y el uso del teclado y ratón. Toma una foto de tu espacio antes y después de realizar los ajustes.

Ejercicio 9: Explica la diferencia entre las señales de advertencia y las señales de obligación, y proporciona un ejemplo de cada una.

Ejercicio 10: Planifica una rutina de trabajo que incluya pausas regulares y ejercicios de estiramiento. Lleva a cabo esta rutina durante una semana y escribe un informe sobre los beneficios observados.

6. Test de conocimientos

1. ¿Qué es la arquitectura de Von Neumann?

- a) Un tipo de sistema operativo.
- b) Un modelo de diseño de ordenadores que utiliza un único espacio de direcciones para instrucciones y datos.
- c) Un lenguaje de programación.
- d) Un tipo de memoria caché.

2. ¿Cuáles son los componentes principales de la arquitectura de Von Neumann?

- a) Hardware y software.
- b) CPU, memoria, buses y sistema de E/S.
- c) Aplicaciones y licencias.
- d) Sistemas operativos y firmware.

3. ¿Qué es la memoria caché?

- a) Un tipo de memoria principal.
- b) Una memoria de acceso lento.
- c) Una memoria que almacena datos de acceso frecuente.
- d) Un tipo de memoria secundaria.

4. ¿Qué son los buses en un ordenador?

- a) Dispositivos de entrada y salida.
- b) Canales de comunicación entre los componentes del ordenador.
- c) Tipos de interfaces.
- d) Sistemas operativos.

5. ¿Qué es una interfaz en un sistema informático?

- a) Un programa de aplicación.
- b) Un lenguaje de programación.
- c) Un punto de conexión entre dos componentes de hardware o entre hardware y software.
- d) Un tipo de memoria.

6. ¿Qué diferencia el software libre del software de código abierto?

- a) El software libre es siempre gratuito, mientras que el de código abierto puede ser de pago.
- b) El software libre tiene el código fuente disponible, mientras que el de código abierto no.
- c) El software libre se basa en cuatro libertades esenciales, mientras que el de código abierto puede tener licencias menos restrictivas.
- d) No hay diferencia entre ambos.

7. ¿Qué es una licencia de software?

- a) Un programa de aplicación.
- b) Un acuerdo legal que establece los términos de uso del software.
- c) Un tipo de sistema operativo.
- d) Un componente de hardware.

8. ¿Qué son las aplicaciones en la nube?

- a) Aplicaciones que se ejecutan en un servidor remoto y se accede a ellas a través de Internet.
- b) Aplicaciones que se instalan en el ordenador local.
- c) Aplicaciones que se utilizan para la gestión de la nube.
- d) Aplicaciones que requieren una conexión a Internet para funcionar.

9. ¿Qué es la ergonomía en el contexto del trabajo con ordenadores?

- a) El diseño del espacio de trabajo para prevenir lesiones y promover la comodidad.
- b) El uso de software ergonómico.
- c) La prevención de riesgos eléctricos.
- d) La gestión de residuos informáticos.

10. ¿Qué son las descargas electrostáticas?

- a) Un tipo de riesgo eléctrico.
- b) Un tipo de software malicioso.
- c) Un tipo de problema de hardware.
- d) Un tipo de fallo en el sistema operativo.