



Unidad 1: Desarrollo del Software

Actividad propuesta Act 1.4

R1. Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento.

Criterios de evaluación:

- a) Se ha reconocido la relación de los programas con los componentes del sistema informático, memoria, procesador, periféricos, entre otros.
- b) Se han clasificado los lenguajes de programación.
- c) Se han diferenciado los conceptos de código fuente, objeto y ejecutable.
- g) Se han identificado las fases de desarrollo de una aplicación informática.

Completar las definiciones siguientes:

1) ¿Qué es el software? Tipos de software.

Es un conjunto de programas, instrucciones y normas informáticas para ejecutar de manera segura los datos de un ordenador. Su objetivo es transferir los datos a los programas que se ejecutan en un ordenador con la finalidad de realizar determinadas tareas sobre el hardware, y los datos necesarios para el resto de programas. Existen distintos tipos de software en función de su uso:

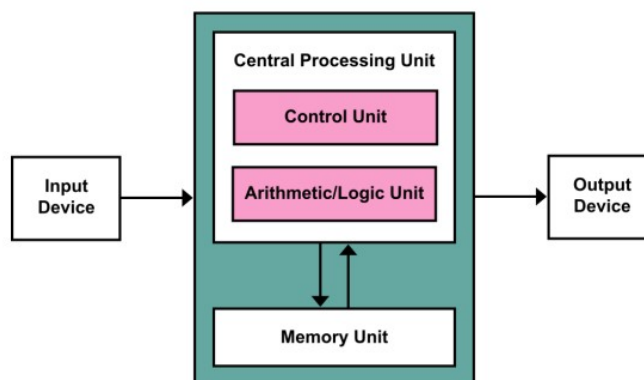
- Software de sistemas: son programas que permiten gestionar y controlar el hardware del ordenador facilitando otros programas y el usuario pueda interactuar con los componentes físicos.
- Software de programación o desarrollo: proporcionan herramientas necesarias para la creación de software. Incluyen editores de código, compiladores, depuradores y otras utilidades. Lo más común es utilizar entornos de desarrollo integrados (IDE), que agrupan todas estas herramientas en un solo programa para facilitar el proceso de desarrollo.
- Software de aplicación: son los programas que ayudan a los usuarios a realizar algún tipo de tarea y permiten interactuar con el usuario, a través de un entorno gráfico u otras interfaces de usuario facilitando su uso para las personas que lo utilizan.

2) Explica la relación entre hardware y software.

El hardware es un conjunto de componentes físicos que permiten la comunicación con el equipo (ratón, pantalla, teclado, altavoces,...). El software es un conjunto de programas o aplicaciones y reglas informáticas que hacen posible el funcionamiento del equipo (Chrome, LibreOffice, Spotify,...).

3) ¿Qué has entendido de la arquitectura de Von Neumann?

Se compone de una Unidad Central de Proceso (CPU) para procesar las instrucciones aritméticas y lógicas, una memoria para almacenar los datos y programas, dispositivos de entrada y salida y canales de comunicación para enviar y recibir mensajes.



Cada uno de los componen hacen referencia al hardware del ordenador:

1. Unidad Central del Proceso (CPU), se encarga de ejecutar los programas, controlando su secuencia, interpretando y ejecutando sus instrucciones. También controla el resto de componentes; como los periféricos, la memoria, la información que hay que procesar, ..., según lo necesiten las instrucciones.
 1. Unidad Aritmético-Lógica realiza los cálculos matemáticos y lógicos necesarios para su funcionamiento.
 2. Unidad de Control (UC), recoge las instrucciones incluidas en la memoria principal y ordena su ejecución a través de envíos de señales a la ALU y a los registros.
2. La memoria principal - RAM (Random Access Memory), se encarga de almacenar datos e instrucciones de los programas que se deben ejecutar el que necesite su funcionamiento. El grupo de registros por el que está construida son capaces de retener esa información en su interior mientras se encuentre el ordenador encendido.
3. Unidad de entrada / salida permite la comunicación del ordenador con el exterior transfiriendo la información mediante los periféricos. Pueden ser de varios tipos:
 1. Entrada introducen la información o los datos desde el exterior a la memoria principal. Ejemplo el teclado.
 2. Salida son encargadas de sacar al exterior los datos o resultados de los procesos. Ejemplo la pantalla.
 3. Entrada / Salida utilizan tanto la entrada como para la salida de información. Manejan la información en formato binario pero necesitan otros procesos de conversión para comunicarse con otros dispositivos. Ejemplo las tarjetas de red inalámbricas.



4) *¿Qué entiendes por desarrollo del software? ¿Cuáles son las etapas en el desarrollo del software? Explica con tus palabras qué se hace en cada una de ellas.*

Son procesos que se realizan el cual sigue una serie de procesos establecidos para la obtención o implementación de dispositivos electrónicos, ordenadores, móviles,... Tiene una serie de fases en la que se realizan una serie de trabajos, siguiendo unas pautas:

- Análisis su finalidad es analizar las necesidades de los usuarios del software para resolver qué debe hacer la aplicación, y conforme ello escribir una explicación del sistema. En resumen conseguir una especificación del sistema, basado en una documentación descriptiva del software, pero no cómo. Lo principal es que sea una documentación comprensible, complementa y fácil de comprobar y modificar para facilitar el trabajo anterior
- Diseño su finalidad es analizar el sistema, para de esta forma saber como se va a resolver el problema planteado. Los requisitos serán los resultados de la base de análisis en partes del software. Con respecto a la fase anterior tomaremos conciencia para decidir qué sistema gestor de datos se deberá utilizar para administrar la información.
- Programación consiste en traducir los resultados de la fase de diseño en una forma legible para la máquina de manera que el código fuente sea utilizado en un lenguaje definido. Y el código resultado de esa será el código ejecutable.
- Pruebas es donde se realiza el funcionamiento correcto de la aplicación. Se comienza probando cada componente software por separado, se integran poco a poco los componentes, hasta probar el programa por completo. Su finalidad es comprobar que la aplicación satisfaga los requisitos del cliente. En esos resultados es donde descubrirán errores, donde será necesario modificar el código o rehacer algunas de las otras tareas anteriores.
- Explotación se llevara a cabo la instalación y comienzo del software en el entorno de trabajo del cliente.
- Mantenimiento el software sufrirá cambios después de que haya sido entregado al cliente. Consistirá en realizar los cambios del software existente para cada uno de las actividades que corresponde con el nuevo ciclo de vida.

5) ¿Qué tenemos que evitar en el desarrollo de un software?

- No definir específicamente los requisitos del proyecto, puede falta de interpretación y malas interpretaciones, eso puede afectar a la hora de traspasar la información a la calidad que se obtenga el software y su funcionalidad.
- No utilizar las pruebas adecuadas como es asegurarse de la funcionalidad específica del software para que cumpla con los requisitos del proyecto.
- No tener en cuenta la flexibilidad para adaptarse a las medidas necesarios de los usuarios y las empresas que cambian. También puede estar relacionado con la utilización de las bases de datos y la creación de un diseño modular que permita diferentes funcionalidades.
- No seguir las mejores prácticas de seguridad al no seguir las pautas puede poner en riesgo el desarrollo de todo el proceso. Esto puede incluir la utilización de técnicas de configuración segura y la realización de pruebas exhaustivas.

Información buscada en [Errores comunes del desarrollo software y cómo evitarlos](#).

6) Explica los distintos modelos de ciclos de vida.

Es el orden en el que se deben llevar a cabo las tareas para realizar el paso a la siguiente tarea a la siguiente. Consiste en la obtención de ciertos productos intermedios que indican características en las que debe distribuirse los productos para progresar a la fase anterior. Este modelo es muy significativo en la forma decisiva en su éxito o fracaso.

7) ¿Qué es un lenguaje de programación? Explica lenguaje máquina, lenguaje ensamblador, lenguaje de alto nivel y lenguaje visual.

Son instrucciones de los programas que se van a ejecutar en el ordenador para resolver un problema, de manera que el ordenador entienda el lenguaje binario (ceros y unos). Como ese lenguaje tiene dificultad, facilitan la creación de programas para resolver lenguajes más complejos, llamados lenguajes de programación de alto nivel.

Se puede realizar una clasificación de los lenguajes de programación, establecidos en tres grupos:

- Lenguaje máquina o de bajo nivel es aquel que utiliza el alfabeto binario para establecer la comunicación entre el hardware y la máquina. Fue el primer lenguaje que se usó, pero debido a su dificultad fue sustituido por el lenguaje natural, más fáciles de aprender y utilizar.



- Lenguaje ensamblador o intermedio son aquellos lenguajes que no utilizan códigos numéricos ni direcciones de memoria, se representan mediante instrucciones de forma simbólica. A cada dato se le asigna un código en la que guarda relación con la operación o el dato al que representa. Utilizan direcciones binarias para referirse a esos datos. Los lenguajes ensambladores son más fáciles de entender por el ser humano, a través de esas instrucciones por ello se denomina lenguaje ensamblador.
- Lenguaje de alto nivel o lenguajes evolucionados su finalidad es liberar al programador de tareas aburridas y dificultosas en las que pueda frenar su productividad y eficiencia. Ofrecen que se conozcan ciertas características de la máquina y lo representen en estos lenguajes respecto al lenguaje natural. Sus características son:
 - Se expresan mediante caracteres alfanuméricos, numéricos y especiales.
 - Se pueden definir variables para recoger los datos de manera que formen estructuras de datos, lo que facilita su gran cantidad en el ámbito de la programación.
 - Pueden incluirse líneas de comentarios.
 - Establece instrucciones muy potentes de tipo aritmético, lógicas, de tratamiento de caracteres,...
 - El tiempo de codificación está puesto en los programas que se ha requerido por anteriores lenguajes.
 - Suelen resultar fáciles de corregir y modificar, con la finalidad de reducir el tiempo de programación.
 - El índice de aprendizaje en las personas responsables de la programación es más corta y menos pronunciada que con los lenguajes de máquina y ensambladores.

También tiene desventajas, son las siguientes:

- Los programas escritos no pueden ejecutarse directamente por lo que su duración es más corta que anteriormente.
- La ocupación de la memoria se incrementa tanto por el traductor como por el programa traducido. Actualmente nos ofrecen el hardware y los sistemas operativos reducen la importancia en ello.
- Lenguaje visual es el que utiliza elementos gráficos y figuras para desarrollar un programa. Permite desarrollar programas de software a través de elementos gráficos visuales; ya pueden ser de varios tipos basados en iconos, lenguaje esquema y basados en formularios.