**TEMA 1**

**Hardware**

Es la parte tangible del sistema informático, equipamiento físico

Generalmente se base en la arquitectura de Von Neumann

**Memoria principal:**

Es el componente básico de un ordenador ya que controla la ejecución de las operaciones y dirige el funcionamiento de todos los demás componentes

**Unidad de control**:

Controla la ejecución de las operaciones y dirige el funcionamiento de todos los demás componentes.

**Unidad aritmética lógica**:

Es la parte encargada de realizar las operaciones aritméticas.

**Dispositivos de entrada y salida**:

Son los dispositivos que se conectan al equipo para facilitar el trabajo al usuario.

**Ciclo de una instrucción:**

La UC indica a la memoria principal la siguiente instrucción que quiere ejecutar. Luego recibe la instrucción la UC. Bajo las directrices de UC, la UAL realiza la operación y se guarda el resultado en su destino.

Una vez ejecutada la instrucción se incrementa al contador del programa y se pasa a realizar de la siguiente instrucción

**Software**

Es la parte intangible del sistema informático

Se encarga de traducir las órdenes dadas por el usuario en órdenes comprensibles por el hardware.

**Características:**

Es lógica, se desarrolla (no se fabrica), no se estropea y se puede construir a medida.

**Tipos:**

**Software de sistema:** ha de estar instalado y configurado en nuestro ordenador (Windows, Linux)

**Software de programación:** es el conjunto de herramientas que nos permiten desarrollar programas informáticos. (Eclipse, netbenans)

**Software de aplicación:** son los programas que tienen una finalidad (office)

**Lenguajes de programación:**

Todos los programas están escritos en algún lenguaje

Permiten comunicarnos con el hardware del ordenador.

Tienen una sintaxis y un conjunto de normas y palabras reservadas

**Tipos:**

**Lenguaje maquina:** son instrucciones complejas, compuesta de unos y ceros – no necesita ser traducido.

**Lenguaje de nivel medio:** sustituye al lenguaje máquina y utiliza mnemotécnicos. – difícil de comprender y programar.

Lenguaje nivel alto: la mayoría de lenguajes pertenecen a esta categoría- ingles – orientado a objetos.

**Clasificarlos:**

Lenguaje compilados, lenguajes interpretados y lenguaje virtuales

**Tipos de lenguajes:**

JAVA – PYTHON – C /C++ JavaScript - PHP

JAVA funciona en todos los procesadores y es uno de los más utilizados en la actualidad

Característica: simple – orientado a objetos – fácil de entender

Ventajas: es fácil de entender

Desventajas: rinde poco y cuesta

PYTHON: es multiplataforma y es interpretado

Características: multiplataforma, no compilado

Ventaja: flexible, tiene muchas herramientas

Desventaja: es lento y no soporta varios servidores

C: lenguaje estructurado y códigos cortos

Característica: facilidad de aprender

Ventaja: gran facilidad

Desventaja: lenguaje no visual

**Codificación**:

Los traductores son programas cuya finalidad es traducir lenguajes de alto nivel a lenguajes de bajo nivel.

Un intérprete traduce el código fuente línea a línea.

**Depuración:**

Es el proceso de identificar y corregir errores de programa

**Ingeniería de software**

Es una disciplina formada por un conjunto de métodos que se utiliza en el software y mantenimiento

**Desarrollo de software**

Construirlo simplemente mediante su descripción

**TEMA 2**

**IDE:**

* Es un software compuesto por una serie de herramientas que utilizan los programadores para desarrollar código.
* Soporta un único lenguaje de programación o varios

**Las herramientas que componen un IDE:**

Editor del código de programación - compilador – interprete – depurador – constructor del interfaz grafico

**Funciones de un entorno de desarrollo**:

Un editor de código fuente – un compilador – automatización de generación de herramientas - depurador.

**Entornos integrados libres:**

Netbeans eclipse gambas

**Entorno integrados propietarios:**

Microsoft visual studio – jbuilder – J creator

**Estructura de Entornos de desarrollo:**

Editor de textos

Compilador / interprete

Depurador

Generador automático de herramientas

Interfaz grafica

**Gestión de módulos:**

Es un componente software que contiene clases de Java que pueden interactuar con las APIs del entorno de desarrollo y el manifest file, que es un archivo especial que lo identifica como módulo.

**TEMA 3**

**Modelo:**

Es una simplificación de la realidad.

Es resultado de un proceso de abstracción y ayuda a razonar y comprender la realidad.

**Modelo software**

Es una descripción de un aspecto del sistema, expresada en un lenguaje bien definido.

Modera la complejidad – independencia del lenguaje de codificación – evaluar el modelo

**¿Por qué es útil de modelar?**

Permite utilizar un lenguaje para facilita la comunicación entre los miembros del equipo de desarrollo.

Que permiten especificar aspectos del sistema como los requisitos.

**Propósitos de los modelos:**

Pensar sobre un diseño de un sistema

Capturar decisiones

Documentar

**Estructura del modelo:**

Conjunto de decisiones – selección de elementos - comportamiento – composición

**UML:**

Permite definir los objetivos que debe cumplir un proyecto software.

Sirve para visualizar, especificar, construir y documentar

Lenguaje unificado de modelado

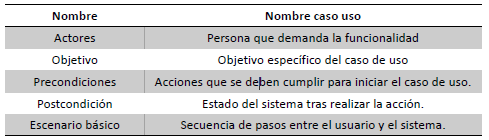
Diagrama estructurado

Diagrama de clase, objetos, De componentes, De paquetes, De despliegue,

Diagrama de comportamiento:

Secuencia, Colaboración, estados, actividades, caso de usos.

DIAGRAMA DE CASO DE USO



Actores: cliente Banco

Objetivo: obtener la cantidad de saldo disponible en su cuenta

Precondiciones: tiene que tener cuenta de banco

Postcondicion: saldo mostrado

Escenario básico: insertar tarjeta, introducir la contraseña, retirar tarjeta

**Metodología AGIL**

Se entiende un conjunto de valores, principios y prácticas que constituyen la base sobre la que cada organización añade sus prácticas para implementarlo.

* Enfoque para tomar de decisiones de proyectos.
* Desarrollo iterativo e incremental.
* Requisitos y soluciones evolucionan.

**Metodología SCRUM**

Se trata de un marco de gestión que permite hacer frente a problemas complejos de manera adaptativa, contribuyendo al desarrollo de productos/soluciones del más alto valor para los clientes.

* Marco o framework
* Conjunto de buenas practicas
* Trabajo colaborativo y tener el mejor resultado