**UML UNIFIED MODELING LANGUAGE (LENGUAJE UNIFICADO DE**

**MODELADO**)



DISNEY ALEXANDRA SIABATO AVELLANEDA

DEMETRIO MAURICIO ESTUPIÑAN FINO

SERVICIO NACIONAL DE APRENDIZAJE

SENA

BOGOTA D.C

2020

Contenido

**1.1. conceptualice acerca de UML** ............................................................................................ 5 **a) Que es UML** ....................................................................................................................... 5

1. **Reseña histórica de UML** ................................................................................................. 5
2. **Ventajas de UML** ............................................................................................................... 6
3. **Características de UML**.................................................................................................... 6
4. **Como se clasifican los diagramas de UML**...................................................................... 6

**1.Diagramas de estructura:** .................................................................................................. 6

1. **Diagramas de interacción:** ................................................................................................ 6
2. **Diagramas de interacción:** ................................................................................................ 6 **f) Diagramas que componen UML** ....................................................................................... 7

**1.1.2 Diagrama de Clases** .......................................................................................................... 7

1. **Clase Abstracta** ............................................................................................................ 7
2. **Asociaciones** ....................................................................................................................... 7
3. **Multiplicidad:** .................................................................................................................... 8
4. **Composición y Agregación** ............................................................................................... 9
5. **Generalización:** .................................................................................................................. 9

**1.1.3 Diagrama de Objetos:** ...................................................................................................... 9

1. **Nombre de los objetos:** .................................................................................................... 10
2. **Atributos:** ......................................................................................................................... 10

**1.1.4 Diagrama de casos de Uso:** ............................................................................................ 10

**1.Sistema:** ............................................................................................................................. 10

**2.Casos de uso:** ..................................................................................................................... 11

**3.Actores:** .............................................................................................................................. 11

**4. Relaciones** ......................................................................................................................... 11

**1.1.5 Diagrama de Estados:** .................................................................................................... 11

**1.Estado:** ............................................................................................................................... 11

**2. Transición** ......................................................................................................................... 12

**1.1.6 Diagrama de Secuencias** ................................................................................................. 12

1. **Rol de clase** ................................................................................................................. 12
2. **Activación** ................................................................................................................... 12

**3.Mensajes** ............................................................................................................................ 12

1. **Líneas de Vida:** ........................................................................................................... 13
2. **Destrucción de Objetos** .............................................................................................. 14
3. **Loops:** .......................................................................................................................... 14

**1.1.7 Diagrama de actividades** ................................................................................................ 15

1. **Estados de Acción** ...................................................................................................... 15
2. **Flujo de la Acción** ....................................................................................................... 15
3. **El flujo de objetos** ....................................................................................................... 16
4. **Ramificación:** .............................................................................................................. 16
5. **Sincronización:** ........................................................................................................... 16
6. **Marcos de Responsabilidad:** ..................................................................................... 17
7. **Rol de clase:** ................................................................................................................ 17
8. **Rol de las Asociaciones** .............................................................................................. 18
9. **Mensajes:** .................................................................................................................... 18

**1.1.9 Diagrama de Componentes** ............................................................................................ 18

1. **Componente:** .............................................................................................................. 18
2. **Interface** ...................................................................................................................... 18
3. **Dependencias:** ............................................................................................................. 18

**2.1.1 Diagrama de Distribución:** ............................................................................................ 19

**1.Nodo** ................................................................................................................................... 19

**2.Asociacion** .......................................................................................................................... 19

**3.componentes y nodos** ........................................................................................................ 19

# **3.2.2. cuadro descriptivo de los diferentes tipos de Diagramas UML, sus diferentes**

**funciones y un ejemplo.** ................................................................................................................ 19

**2.1.2 Utilización de UML en el desarrollo de sistemas de información.** ............................. 23

**2.1.3 ¿Por qué es necesario contar con diversos diagramas en el modelo de un sistema?** 23

# **3.1.1 Actividades de contextualización e identificación de conocimientos necesarios para**

**el aprendizaje.** ............................................................................................................................... 23

**3.1.2 interrogantes:** .............................................................................................................. 23

**3.3.3 Definiciones enfocadas en el proceso de programación de software**.......................... 24

# **4.1.1 Normas internacionales para documentación de sistemas de información.**.............. 26

**Bibliografías** ......................................................................................................................... 26

1.1. conceptualice acerca de UML

# a) Que es UML

El Lenguaje Unificado de Modelado (Unifield Modeling Lenguaje UML), es un lenguaje estándar para escribir planos de software, UML se puede utilizar para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. UML prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan.

# b) Reseña histórica de UML

El lenguaje UML comenzó a gestarse en octubre de 1994, cuando Rumbaugh se unió a la compañía Rational fundada por Booch (dos reputados investigadores en el área de metodología del software).

|  |  |
| --- | --- |
| El objetivo de ambos era unificar dos métodos que habían desarrollado: el método Booch y el OMT (Object Modelling Tool). El primer borrador apareció en octubre de 1995. En esa misma época otro reputado investigador, Jacobson, se unió a Rational y se incluyeron ideas suyas. Estas tres personas son conocidas como los “tres amigos”. Además, este lenguaje se abrió a la colaboración de otras empresas para que aportaran sus ideas. Todas estas colaboraciones condujeron a la definición de la primera versión de UML. | |
| En 1997 UML 1.1 fue aprobada por la OMG convirtiéndose en la notación estándar de facto para el análisis y el diseño orientado a objetos.  UML es el primer método en publicar un metamodelo en su propia notación, incluyendo la notación para la mayoría de la información de requisitos, análisis y diseño. Se trata pues de un metamodelo autorreferencial (cualquier lenguaje de modelado de propósito general debería ser capaz de modelarse a sí mismo). En 1997 UML 1.1 fue aprobada por la OMG convirtiéndose en la notación | |
| estándar de facto para el análisis y el diseño orientado a objetos.  UML es el primer método en publicar un metamodelo en su propia notación, incluyendo la notación para la mayoría de la información de requisitos, análisis y diseño. Se trata pues de un metamodelo autorreferencial (cualquier lenguaje de modelado de propósito general debería ser capaz de | |
| modelarse a sí mismo). |  |

# c) Ventajas de UML

Ventajas UML

Se puede usar para diferentes tipos de sistemas

Consolida muchas de las notaciones y conceptos más usadas orientados a objetos Es fácilmente entendible

# d) Características de UML

* un estándar para modelado de sistemas
* no es un estándar para procesos de software
* debe aplicarse en el contexto de un proceso de software

* es una anotación, no es un proceso
* establecido como estándar para documentar el proceso de ingeniería de software
* combina lo mejor del modelado de procesos, objetos, datos y componentes

# e) Como se clasifican los diagramas de UML

se clasifican en tres categorías distintas: Diagramas de estructura, de comportamiento o

interacción.

1.Diagramas de estructura: Representan un sistema y sus componentes de forma estática. Establecen las relaciones que existen entre elementos individuales o entre elementos y conceptos superiores. Un ejemplo de ellos es el diagrama de clases.

1. Diagramas de interacción: Pertenecen a la categoría de diagramas de comportamiento de un sistema dinámicamente. A diferencia de los diagramas de estructura, la secuencia de procesos y, por lo tanto, el tiempo, desempeñan un papel importante en la visualización un ejemplo de este tipo de diagramas es el diagrama de actividad.
2. Diagramas de interacción: Pertenecen a la categoría de diagramas de comportamiento. Se enumeran por separado porque modelan un comportamiento especifico, es decir, las interacciones entre los elementos del sistema. Los elementos básicos de las interacciones son las llamadas líneas de vida. Estas consisten en objetos (los bloques de construcción independientes mas pequeños de la programación orientada a objetos) que representa a participantes individuales en una interacción el diagrama de interacción mas utilizado es el diagrama de secuencia.

# f) Diagramas que componen UML

* Diagrama de Clases
* Diagrama de Objetos
* Diagrama de Casos de Uso
* Diagrama de Estados
* Diagrama de Secuencias
* Diagrama de Actividades
* Diagrama de Colaboraciones
* Diagrama de Componentes
* Diagrama de Distribución

## g) Explicación de los diagramas UML más utilizados

1.1.2 Diagrama de Clases**:** Los diagramas de clases describen la estructura estática de un sistema. Las cosas que existen y que nos rodean se agrupan naturalmente en categorías. Una clase es una categoría o grupo de cosas que tienen atributos (propiedades) y acciones similares

**Ejemplo:** Clase Abstracta Las clases se representan con rectángulos divididos en tres áreas: la superior contiene el nombre de la clase, la central contiene los atributos y la inferior las acciones.

1. Clase Abstracta**:** Las clases se representan con rectángulos divididos en tres áreas: la superior contiene el nombre de la clase, la central contiene los atributos y la inferior las acciones.

|  |
| --- |
| Nombre de clase |
| atributo: Tipo  / atributo Derivado |
| operación () |

1. Asociaciones**:** Las asociaciones son las que representan a las relaciones estáticas entre las clases. El nombre de la asociación va por sobre o por debajo de la línea que la representa. Una flecha rellena indica la dirección de la relación. Los roles se ubican cerca del final de una asociación. Los roles representan la manera en que dos clases se ven entre ellas. No es común el colocar ambos nombres, el de la asociación y el de los roles a la vez. Cuando una asociación es calificada, el símbolo correspondiente se coloca al final de la asociación, contra la clase que hace de calificador.



**Nombre**

**Rol 1**

**Rol 2**

3

.

Multiplici

dad:

Las notaciones utilizadas para señalar la multiplicidad se colocan cerca del

final de una asociación. Estos símbolos indican el

número de instancias de una clase vinculadas a

una de las instancias de la otra clase. Por ejemplo, una empresa puede tener

uno o más empleados,

pero cada empleado trabaja para una sola empresa solamente.

•

1

no más de uno

•

0..1

cero o uno

•

muchos

•

0

.. \* cer

o o muchos

•

1

.. \* uno o muchos

**1**

**1**

**.. \***

**Clase A**

**Clase B**

**Clase A**

**Clase B**

**Clase A**

**Clase A**

**Clase B**

**Clase B**

clasificador

**Empresa**

**Empleado**

**Asociación Tripartita:**



.

4

Composición y Agregación

**:**

Composición es un tipo especial de agregación que denota una

fuerte posesión de la Clase “Todo”, a la Clase “Parte”. Se grafica con un rombo diamante relleno

contra la clase que representa el todo. La agre

gación es una relación en la que la Clase “Todo”

ju

ega un rol más importante que la Clase "Parte", pero las dos clases no son dependientes una de

otra. Se grafica con un rombo diamante vacío contra la Clase “Todo”.

:

5

. Generalización

es ot

ro

nombre para herencia. Se refiere a una relación entre dos clases en

donde una Clase “Específica” es una versión especializada de la otra, o Clase “General”

**Clase B**

**Clase B**

**Clase B**

**Todo**

**Todo**

**Parte**

**Parte**

**Clase General**

**Clase Especifica**

1.1.3 Diagrama de Objetos: Los Diagramas de Objetos están vinculados con los Diagramas de Clases. Un objeto es una instancia de una clase, por lo que un diagrama de objetos puede ser visto como una instancia de un diagrama de clases. Los diagramas de objetos describen la estructura estática de un sistema en un momento particular y son usados para probar la precisión de los diagramas de clases.

1. Nombre de los objetos: Cada objeto es representado como un rectángulo, que contiene el nombre del objeto y su clase subrayadas y separadas por dos puntos.



2

. Atributos

:

Como con las clases, l

os atributos se listan en un área inferior. Sin embargo, los

atributos de los objetos deben tener un valor asignado.

1.1.

4

Diagrama de casos de Uso:

Un caso de uso es una descripción de las acciones

de un sistema desde el punto de vista del usuario

. Es una herramienta valiosa dado que

es una

técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto

de vista del usuario. Los diagramas de caso de uso modelan la funcionalidad del sistema usando

actores y casos

de uso. Los casos de uso son servicio

s o funciones provistas por el sistema para sus

usuarios.

:

1

.Sistema

El rectángulo representa los límites del sistema que contiene los casos de uso. Los

actores se ubican fuera de los límites del sistema.

**Nombre Objeto:**

**Clase**

**Atributo tipo =**

**´Valor´**

**Atributo tipo = ´Valor´**

**Atributo tipo = ´Val**

**or´**

**Nombre Objeto:**

**Clase**

Caso de uso 1

Caso de uso 3

Caso de

uso 2

2.Casos de uso:Se representan con óvalos. La etiqueta en el óvalo indica la función del sistema.



.Actores

:

3

Los actores son los usuarios de un sistema.

**Autor**

. Relaciones

4

**:**

Las relaciones entre un actor y un caso de uso, se

dibujan con una línea simple.

Para relaciones entre casos de uso, se utilizan flechas etiquetadas "incluir" o "extender." Una

relación "incluir" indica que un caso de uso es necesitado por otro para poder

cumplir una tarea.

Una relación "extender" indica

opciones

para un cierto caso de uso.

**>>**

**incluir**

**<<**

**<<**

**extender**

**>>**

1.1.5

Diagrama de Estados:

En cualquier momento, un objeto se encuentra en un estado

particular, la luz está encendida o apagada, e

l auto en movimiento o detenido, la persona leyendo

o cantando, etc. El

diagrama de estados UML captura esa pequeña realidad.

1

.Estado

:

El estado representa situaciones durante la vida de un objeto. Se representa con un

rectángulo que tiene sus esquinas r

edondeadas

**Caso**

**Caso**

**Caso**

**Caso**

**Estado**

1. Transición**:** Una flecha representa el pasaje entre diferentes estados de un objeto. Se etiqueta con el evento que lo provoca y con la acción resultante.

3.Mensajes**:** Los mensajes son flechas que representan comunicaciones entre objetos. Las medias flechas representan mensajes asincrónicos. Los mensajes asincrónicos son enviados desde un objeto que no va a esperar una respuesta del receptor para continuar con sus tareas.



Evento/Acción

**Estado Inicial**

**Estado Final**

1.1.6

Diagrama de Secuencias

**:**

Lo

s diagramas de clases y los de objetos

representan

información estática. No obstante, en un sistema funcional, los objetos interactúan entre sí, y tales

interacciones suceden con el tiempo. El diagrama de secuencias UML muestra la mecánica de la

interacció

n con base en tiempos.

**1.**

Rol de clase

**:**

El rol de la clase describe la manera en que un objeto se va a comportar en el

contexto. No se listan los atributos del objeto.

**2.**

Activación

**:**

Los cuadros de activación representan el tiempo que un objeto necesita para

completar

una

tarea.

**Actor**

**Activaciones**

**Objeto: Cla**

**se**

**Objeto: Clase**

**Objeto: Clase**



**Actor**

**Mensajes**

FLECHA

TIPO DE

MENSAJE

**SIMPLE**

**SINCRONICO**

**ASINCRÓNICO**

**RECHAZADO**

**3.**

Líneas de Vida:

L

as líneas de vida son verticales y en línea de puntos, ellas indican la

presencia del objeto durante el tiempo



**Actor**

**Líneas de vida**

**4.**

Destrucción de Objetos

**:**

Los objetos pueden

ser eliminados tempranamente usando una

flecha etiquetada "<>" que apunta a una X.

**Actor**

>>

Destruir

<<

**5.**

Loops:

Una repetición o loop

en un diagrama de secuencias, es representado como un

rectángulo. L

a condición para abandonar el loop se coloca en la parte inferior entre corchetes

].

[

**Objeto: Cla**

**se**



**Actor**

**[**

**condición para salir]**

**LOOP**

1.1.7

Diagrama de actividades

**:**

Un d

iagrama de actividades ilustra la naturaleza

dinámica de un sistema mediante el modelado del flujo ocurrente de

actividad en actividad.

Una actividad representa una operación en alguna clase del sistema y que resulta en un

cambio en el estado del sistema.

Típicamente, los diagramas de actividad son utilizados para

modelar el flujo de trabajo interno de una operación

.

**1.**

Estados de Acción

**:**

Los estados de acción representan las acciones no interrumpidas

de los objetos.

**2.**

Flujo de la Acción

**:**

Los flujos de acc

ión, representados con flechas, ilustran las

relaciones entre los estados de acción

**Actividad**

**Actividad**

**Actividad**

1. El flujo de objetos: se refiere a la creación y modificación de objetos por parte de actividades. Una flecha de flujo de objeto, desde una acción a un objeto, significa que la acción está creando o influyendo sobre dicho objeto. Una flecha de flujo de objeto, desde un objeto a una acción, indica que el estado de acción utiliza dicho objeto.



**4.**

Ramificación:

Un rombo representa una decisión con caminos alternati

vos. Las salidas

alternativas deben estar etiquetadas con una condición

**Condición 1**

**Condición 2**

**5.**

Sincronización:

Una barra de sincronización ayuda a ilustrar la ocurrencia de

transiciones paralelas, así

quedan representadas las accion

es concurrentes.

**Actividad**

**Nombre**

**Objeto: Clase**

**Actividad**

**Actividad**

**Actividad**

entre los objetos en términos de mensajes secuenciados. Los diagramas de colaboración representan una combinación de información tomada de los diagramas de clases, de secuencias y



**6.**

Marcos de Responsabilidad:

Los marcos de responsabilidad agrupan a las actividades

relacionadas en una misma columna.

**MARCO 1**

**MARCO 2**

**1.1.8**

**Diagrama de Colaboraciones**

**:**

El diagrama de colaboraciones describe l

as interacciones

de casos de uso, describiendo el comportamiento, tanto de l

a estructura estática, como de la

**Actividad**

**Actividad**

**Actividad**

**Actividad**

**Actividad**

**Objeto: Clase**

**Actividad**

estructura dinámica de un sistema.

**1.** Rol de clase: El rol de la clase describe cómo se comporta un objeto. Los atributos del objeto no se listan.

\

## Objeto: Clase

1. Rol de las Asociaciones**:** Los roles de asociación describen cómo se va a comportar una asociación en una situación particular. Se usan líneas simples etiquetadas con un



estereotipo

**>>**

**<<**

**Global**

**3.**

Mensajes:

Contrariamente a los diagramas de secuencias, los diagramas de

colaboración no tienen una manera explí

cita para denotar el tiempo, por lo que entonces

numeran a los mensajes en orden de ejecución.

La numeración puede anidarse; por

ejemplo, para mensajes anidados al mensaje número 1: 1.1, 1.2, 1.3, etc. La condición

para un mensaje se suele colocar entre co

rchetes. Para indicar un loop se usa \* después

de la numeración.

**[**

**Condición]**

**nombre del mensaje**

1.1.9

Diagrama de Componentes

**:**

Un diagrama de componentes describe la

organización de los componentes físicos de un sistema.

**1.**

Compone

nte:

Un componente es un bloque de construcción física del

sistema.

**2.**

Interface

**:**

Una interface describe a un grupo de operaciones usada o creada por

componentes.

**Comp**

**onente**

1. Dependencias: Las dependencias entre componentes se grafican usando flechas de puntos



2.1.1

Diagrama de Distribución:

El diagrama de distribución UML muestra la

arquitectura física de un sistema informático. Puede representar a los equipos y a los dispositivos,

y también mostrar sus

interconexiones y el softwa

re que se encontrará en cada máquina.

.Nodo

1

**:**

Un nodo es un recurso físico capaz de ejecutar componentes de código.(Procesador)

2

.Asociacion

**:**

La asociación se refiere a la conexión física entre los nodos, como por

ejemplo

Ethernet

.componentes y nodos

3

3.2.2

. cuadro descriptivo de los diferentes tipos de Diagramas UML,

sus

diferentes funciones y un ejemplo.

**Componente**

**Componente**

**Componente**

|  |  |  |
| --- | --- | --- |
| Tipos de  diagramas | Función | Ejemplo |
| Diagrama de Clases | está formado por varios rectángulos de este tipo conectados por líneas que representan las asociaciones o maneras en que las clases se  relacionan entre si |  |
| Diagrama de Objetos | los diagramas de objetos describen la estructura estática de un sistema en un  momento particular y son usados para probar la precisión de los diagramas de clases |  |



|  |  |  |
| --- | --- | --- |
| Diagrama de Casos de  Uso | Los  diagramas de uso se  suelen utilizar el modelado del sistema desde el punto de vista de sus usuarios para representar las acciones que realiza cada tipo de usuario. |  |
| Diagrama de  Secuencias | muestra la mecánica de  la interacción con base en tiempos |  |
| Diagrama de Actividades | Muestran el orden en el que se van realizando tareas dentro un sistema. |  |
| Diagrama de Colaboracio nes | Permite  observar adecuadamen  te la  interacción de un objeto con respecto a los demás |  |
| Diagrama de Componente  s | Organizan lógica de la implementaci  ón de un  sistema |  |
| Diagrama de Distribución | Es nuestra denominació n genérica para nuestros equipos |  |

2.1.2 Utilización de UML en el desarrollo de sistemas de información.

**R=** UML nos permite presentar diversas perspectivas de un sistema, describiendo en una visión más amplia lo que hará el sistema, visualizando, especificando y construyendo, cubriendo las decisiones de análisis, diseño e implementación, cabe resaltar que no es un lenguaje de programación, pero sus modelos pueden conectarse a lenguajes de programación.

2.1.3 ¿Por qué es necesario contar con diversos diagramas en el modelo de un sistema?

**R=** El modelado es importante porque ayuda que el equipo de desarrollo visualice, especifique, construya y documente la estructura y comportamiento de la arquitectura de un sistema.

3.1.1 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje.

A continuación, encontrará una actividad, donde podrá establecer e identificar los conceptos previos para la comprensión del tema. El objetivo de este cuestionario es evidenciar la lectura del material que se ha publicado y el proceso de investigación implementada por usted para la obtención de conocimiento, escriba respuestas propias basadas en comprensión de textos y síntesis sobre el material. Le sugiero indagar e interpretar los siguientes,

temas:

3.1.2 interrogantes:

* ¿Considera que el informe de acuerdo con el análisis del problema es fundamental para el desarrollo e implementación de su proyecto, explique por qué?

**R=** Es fundamental ya que, si deseamos capturar el comportamiento deseado de un sistema UML para los desarrolladores o expertos en el dominio, este nos ayudara a que nuestro sistema marche y lleve un análisis de proyección adecuado, además siendo este un medio de comprensión de nuestro sistema, ayudándonos a validar la arquitectura y a verificar el sistema en el transcurso de desarrollo de este.

* ¿Cree que en el resultado del análisis mediante un informe se deben abarcar todos los procesos, además de la normatividad y necesidades del cliente, explique por qué?

**R=** Si se lleva un adecuado proceso de concepción, indagación, elaboración, priorización y validación teniendo en cuenta un debido proceso como lo aproximarnos por primera vez al sistema, identificando el problema, indagando y saber si se está llevando un entendimiento del problema empezando a realizar preguntas para empezar a rellenar interrogantes, indagando si realmente estamos entendiendo el problema identificando las necesidades de todos los actores. En elaboración el objetivo es documentar el proceso de análisis para que más adelante se empiece hacer el proceso de diseño, especificando lo que nuestro sistema va a hacer y no, teniendo en cuenta priorización del sistema, finalmente validando nuestra documentación de tal manera que no haya ambigüedades, inconsistencias o errores, para que cuando el sistema llegue a su etapa final haga utilización de casos de uso pasando a la etapa de diseño con muchos de los elementos ya identificados.

* ¿Investigue y mencione mínimo tres tipos de herramientas y/o metodologías que permitan realizar modelos que garanticen la construcción de un software de calidad?

**R= 1. Real Academia Española:** Esta propiedad o conjunto de propiedades inherentes a una cosa, que permiten apreciarla como igual, mejor o peor que las restantes de su especie, es decir que sirve para compararlo.

**2. ISO:** Que cuando se habla de la calidad del software el conjunto de características de una entidad que le confieren su aptitud para satisfacer necesidades expresadas y las implícitas.

**3.IEEE:** Grado con el cual el cliente o usuario percibe que el software satisface sus expectativas.

* ¿Los diagramas UML permiten el análisis de los requerimientos de un sistema de información, explique por qué?

**R=** Los requerimientos nos permitirán documentar las necesidades de nuestro sistema teniendo en cuenta comportamiento, restricciones, y en cuanto la calidad que se espera o se necesita que tenga, en conjunto con los diagramas UML cumpliendo con lo planteado. haciendo utilidad de diagramas con el que se documentara el funcionamiento del sistema de la forma más general posible, pasando de la etapa de diseño con muchos de los elementos identificados de como deseamos que nuestro sistema funcione en un futuro.

3.3.3 Definiciones enfocadas en el proceso de programación de software

* **Modelo:** Es el diseño de aplicaciones de software antes de la codificación

* **Codificación**: Es más que una estrategia predeterminada, el resultado de una falta de experiencia o presión que se ejerce sobre los desarrolladores para cumplir con una fecha de entrega
* **Software**: Soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas.

* **Caso de uso**: es una técnica para capturar información de como un sistema o negocio trabaja actualmente, o como desea que trabaje.

* **Diagrama:** Es la representación grafica de un algoritmo

* **Prototipo:** es el propio mockup, pero en el que ya se pueden tocar distintas interfaces e intercambiar pantallas en la aplicación.

* **Objeto:** Los objetos se crean a partir de una clase.

* **Clase**: Definiciones de elementos de tipo homogéneo

* **Diagramas UML:** El UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. Debido a que el UML es un lenguaje, cuenta con reglas para combinar tales elementos.

* **Atributo:** Define los valores que se puedan asociar a instancias de clase o interfaz.

* **Versiones de UML:** UML1.X, UML2.X, UML.3X

* **Análisis:** Entender el proceso de análisis y diseño de aplicaciones usando buenas prácticas.

* **Herramientas CASE:** Son diversas aplicaciones informáticas o programas informáticos destinadas a aumentar el balance en el desarrollo de software reduciendo el costo de estas, en términos d tiempo y de dinero.

* **que es cite 3**: Representa a una cita del titulo de una obra. Este elemento puede ser usado para citar títulos de libros, artículos científicos, ensayos, pinturas, esculturas, obras de teatro, canciones, películas, series de TV, video juegos, etc..

* **Agregación: Es un tipo de asociación que indica que una clase es parte de otra clase, este se representa con un diamante de color blanco colocado en el extreme de la clase que representa el todo.**

* **Composición:** Es una forma fuerte de composición donde la vida de la clase contenida debe coincidir con la clase del contenedor

* **Herencia**: Es un mecanismo que hace posible que una clase herede todo el componente y atributos de otra clase más específica.

* **Encapsulamiento:** ocultamiento del funcionamiento interno de las diferentes operaciones.

* **polimorfismo (P.O.O):** se refiere a la propiedad por la que es posible enviar mensajes sintácticamente iguales a objetos de tipos distintos.

* **Cardinalidad:** Especifica cuantas instancias de una clase están asociadas a una instancia de la otra clase, los tipos de multiplicidad son: uno uno, uno a muchos, y muchos a muchos

4.1.1 Normas internacionales para documentación de sistemas de información.

Las normas ISO son estándares internacionales, para la optimización de los procesos de la empresa, elaboradas por la organización de la estandarización, las normas ISO no tienen reglas específicas o listas de control, los estándares ISO ofrecen la posibilidad de que cada empresa pueda crear su propia estrategia ISO. En esta estrategia una empresa registrará, por ejemplo, como archivará o protegerá su propia información empresarial

# Bibliografías

<https://desarrolloweb.com/articulos/flujo-trabajo-habitual-diseno-web-apps.html>

<https://desarrolloweb.com/articulos/260.php>

<https://desarrolloweb.com/>

<http://www.agilemodeling.com/>

<http://asanchez.cs.buap.mx/ceneval/codificacionSistema.pdf>

<http://www.uml.org/>

<http://www.agilemodeling.com/essays/umlDiagrams.htm>

<http://stadium.unad.edu.co/ovas/10596_9839/qu_es_uml.html>