



LENGUAJE UNIFICADO DE MODELADO

¿Qué es UML?

El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento.

Es comparable a los planos usados en otros campos y consiste en diferentes tipos de diagramas. En general, los diagramas UML describen los límites, la estructura y el comportamiento del sistema y los objetos que contiene. UML no es un lenguaje de programación, pero existen herramientas que se pueden usar para generar código en diversos lenguajes usando los diagramas UML. UML guarda una relación directa con el análisis y el diseño orientados a objetos.

¿Para que se usa UML?

#UML ES UNA DE LAS HERRAMIENTAS QUE SE USAN PARA TRANSFORMAR UNA IDEA EN UN MODELO CONCEPTUAL DE UN SISTEMA, PARA PODER COMUNICARLAS.

#SE USA PARA DOCUMENTAR UN ANÁLISIS BIEN PROFUNDO SOBRE EL PROBLEMA, PARA LUEGO PODER HACER UN PLAN Y TRANSMITIR LAS NECESIDADES.

¿Por qué es importante UML?

#EL MODELADO ES MUY IMPORTANTE EN EL DISEÑO DE UN SISTEMA PARA NO TERMINAR CONSTRUYENDO UNA COMPUTADORA DE ESCRITORIO CUANDO EN REALIDAD SE QUERÍA CONSTRUIR UNA NOTEBOOK.

#LA NECESIDAD DE CREAR MODELOS, SUCEDE EN TODOS LOS ÁMBITOS. ESTO SIGNIFICA MODELAR UNA IDEA, Y REALIZAR UN DISEÑO CONCEPTUAL ACORDE.

#EL ÉXITO DE LOS PROYECTOS DE DESARROLLO DE SISTEMAS SE DEBE AL ENLACE QUE EXISTE ENTRE QUIEN TIENE LA IDEA Y EL DESARROLLO, QUIEN TIENE QUE COMPRENDERLA PERFECTAMENTE.

#UML ES UNA HERRAMIENTA QUE AYUDA A CUMPLIR ESTA FUNCIÓN. CAPTURAR LA IDEA Y COMUNICARLA A QUIEN ESTÉ INVOLUCRADO EN EL PROCESO DE DESARROLLO.

Tipos de diagramas UML

Existen dos subcategorías de diagramas UML: diagramas estructurales y diagramas de comportamiento.

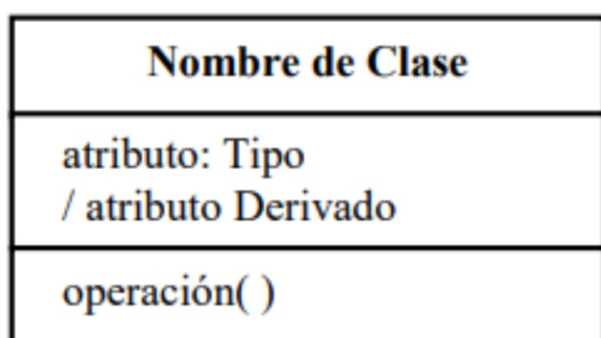
Los diagramas estructurales representan los componentes que forman un sistema y la relación entre dichos componentes. Estos diagramas muestran los aspectos estáticos de un sistema.

Los diagramas de comportamiento representan lo que ocurre dentro de un sistema. Muestran cómo interactúan todos los componentes entre sí y con otros sistemas o usuarios.

Diagramas estructurales

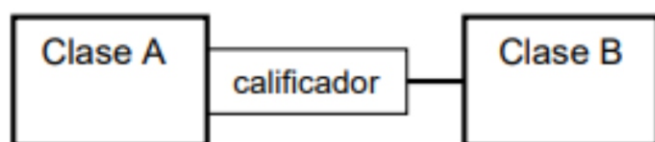
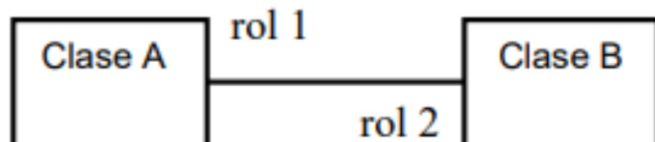
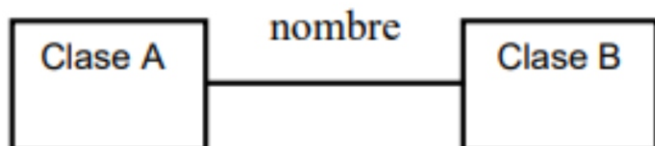
#Diagrama de clases.

Un diagrama de clases UML representa un sistema estático orientado a objetos. Define proyectos por clases, atributos y funciones; como tal, es un bloque de construcción fundamental de cualquier solución orientada a objetos. Muestra las clases de un sistema y las operaciones de cada una de ellas.



Clase Abstracta

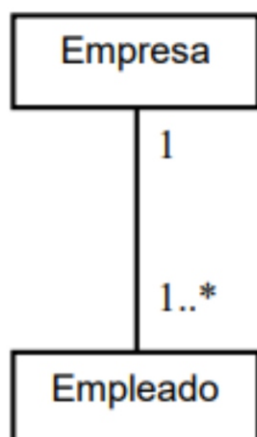
Las *clases* se representan con rectángulos divididos en tres áreas: la superior contiene el nombre de la clase, la central contiene los *atributos* y la inferior las *acciones*.



Asociaciones

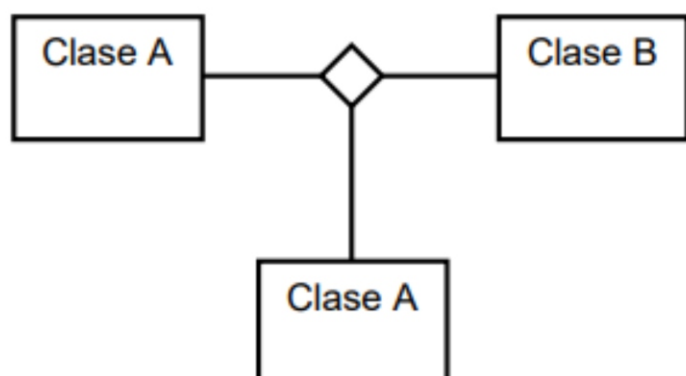
Las *asociaciones* son las que representan a las relaciones estáticas entre las clases. El nombre de la *asociación* va por sobre o por debajo de la línea que la representa. Una flecha rellena indica la dirección de la relación. Los *roles* se ubican cerca del final de una *asociación*. Los *roles* representan 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.

1	no mas de uno
0..1	cero o uno
*	muchos
0..*	cero o muchos
1..*	uno o muchos

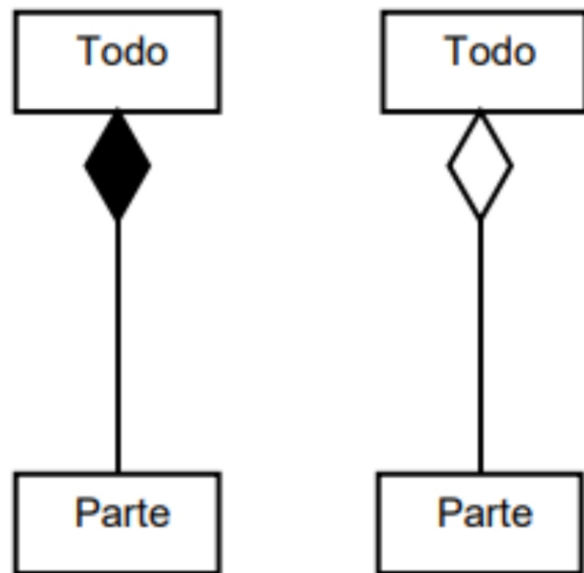


Multiplicidad

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.



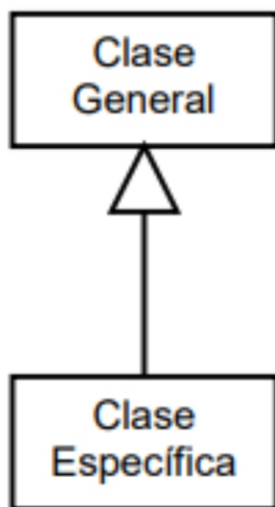
Asociación Tripartita



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 *agregación* es una relación en la que la Clase "Todo" juega 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".



Generalización

Generalización es otro 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". Por ejemplo, Honda es un tipo de auto, por lo que la Clase "Honda" va a tener una relación de *generalización* con la Clase "Auto".

#Diagrama de paquetes.

#Diagrama de objetos.

#Diagrama de componentes.

#Diagrama de estructura compuesta.

#Diagrama de despliegue.

Diagramas de comportamiento

#Diagrama de caso de uso.

Los diagramas de casos de uso proporcionan una visión gráfica de los actores que intervienen en un sistema de software. Ayudan a los desarrolladores a analizar las relaciones entre los casos de uso y los personajes.

Características:

- *REPRESENTAN EL COMPORTAMIENTO DE UN SISTEMA DESDE LA PERSPECTIVA DEL USUARIO PARA ALCANZAR LOS OBJETIVOS.
- *DESCRIBE LO QUE EL SOFTWARE HACE, Y NO LA MANERA EN QUE LO HACE.
- *SE COMPONEN DE UNA REPRESENTACIÓN GRÁFICA Y UNA ESPECIFICACIÓN FUNCIONAL.
- *DEBEN ESCRIBIRSE DESDE LA PERSPECTIVA DEL ACTOR.
- *DESCRIBEN LA INTERACCIÓN DEL ACTOR Y EL SISTEMA.
- *NO CONFUNDIR CON DESCOMPOSICIÓN FUNCIONAL.

Componentes:

1- Diagrama

Actores/Rol

#UN ACTOR ES ALGUIEN O ALGO QUE INTERACTÚA CON EL SISTEMA (PERSONA O SISTEMA DE HARDWARE O SOFTWARE).

#UN ACTOR ESTIMULA AL SISTEMA CON ALGÚN EVENTO O RECIBE INFORMACIÓN DEL SISTEMA.

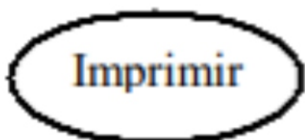
#UN ACTOR ES EXTERNO AL SISTEMA.

#UN ACTOR CUMPLE UN ROL FUNCIONAL DEFINIDO (CLIENTE, ENTIDAD, ETC).



Los *actores* son los usuarios de un sistema.

Casos de uso



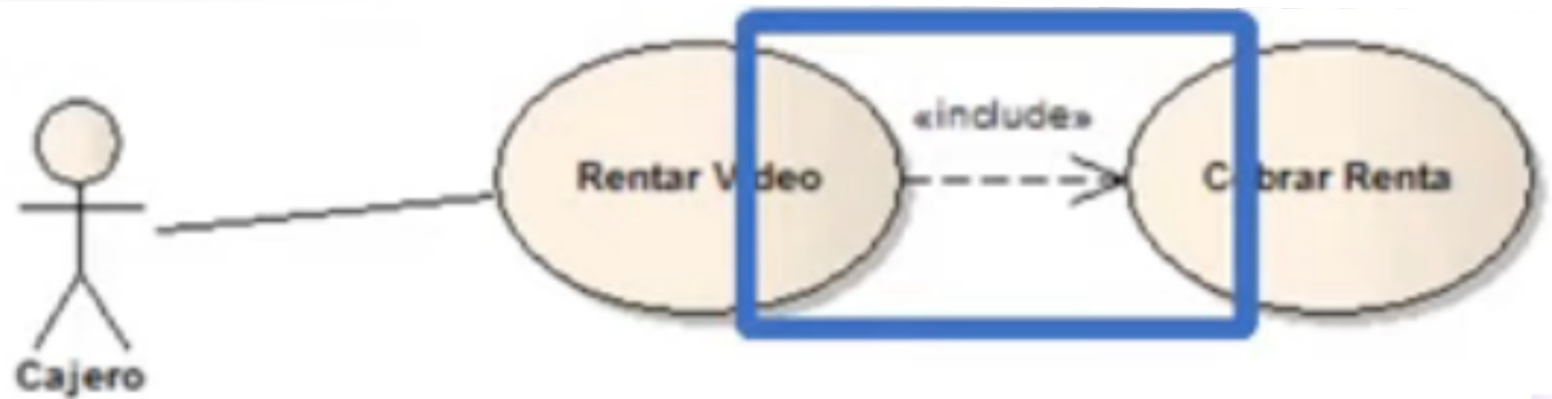
Se representan con óvalos. La etiqueta en el óvalo indica la función del sistema.

Relaciones

#SON RELACIONES QUE USAMOS PARA UNIR GRÁFICAMENTE DOS CASOS DE USO, CUYOS FLUJOS DE EVENTOS ESTÁN UNIDOS, NORMALMENTE EN UNA SOLA SESIÓN DEL USUARIO.

*RELACIÓN <<INCLUDE>>

#UN RELACIÓN DE INCLUSIÓN SE REPRESENTA COMO UNA DEPENDENCIA QUE DECLARA QUE UN CASO DE USO UTILIZA INFORMACIÓN Y SERVICIOS DE OTRO.



RELACIÓN <<EXTEND>>

#LA FUNCIONALIDAD DE UN CASO DE USO INCLUYE UN CONJUNTO DE PASOS QUE OCURREN SOLO EN ALGUNAS OPORTUNIDADES.



RELACIÓN <<USO>>

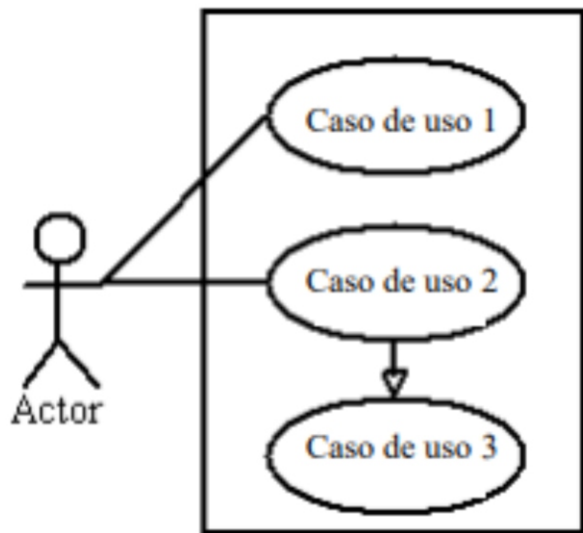
RELACIÓN <<GENERALIZACIÓN>>

2- Escenario

#ES UN TEXTO MUY SIMPLE CON CIERTO FORMATO QUE DESCRIBE COMO SE DEBERÍA COMPORTAR UN SISTEMA ANTE LA INTERACCIÓN CON UNO O MAS USUARIOS.

#CADA SECUENCIA ESPECIFICA DEL CASO DE USO SE DENOMINA ESCENARIO.

#UN ESCENARIO ES UNA SECUENCIA ESPECIFICADA DE ACCIONES ENTRE LOS ACTORES Y EL SISTEMA (INSTANCIA DE UN CASO DE USO).



El rectángulo representa los límites del sistema que contiene los *casos de uso*. Los *actores* se ubican fuera de los límites del sistema.

#Diagrama de actividades.

#Diagrama de secuencia.

#Diagrama de comunicación.

#Diagrama de interacción.

#Diagrama de temporización.

#Diagrama de máquina de estados.

#Diagrama de perfil