

FUNDAMENTOS Y DISEÑO DE BASE DE DATOS



Juan Vladimir
@juanvladimir13

UML

UML: Lenguaje Unificado de Modelado

Ayuda a especificar, visualizar y documentar modelos de sistemas de software, incluida su estructura y diseño.

Se puede modelar prácticamente cualquier tipo de aplicación que se ejecute en cualquier tipo y combinación de hardware, sistema operativo, lenguaje de programación y red.



CLASIFICACIÓN DE DIAGRAMAS UML

Diagramas estructurales

Diagrama de clases

Diagrama de objetos

Diagrama de despliegue

Diagrama de componentes

Diagrama de paquetes

Diagramas de comportamiento

Diagrama de actividades

Diagrama de caso de uso

Diagrama de máquina de estados

Diagrama de secuencia

Diagrama de comunicación

DIAGRAMAS DE CLASES

- Modela la estructura del software.
- Representa la estructura estática de un sistema.
- Se utiliza principalmente en el diseño orientado a objetos para modelar los componentes de un sistema y cómo se relacionan entre sí.
- Facilita la comprensión del diseño y la arquitectura del sistema.

QUE ES UNA CLASE

- Representa objetos reales o abstractos con características y comportamientos específicos
- Representa a un conjunto de objetos que comparten una estructura y un comportamiento comunes.
- Es un modelo que se utiliza para describir uno o más objetos del mismo tipo.

COMPONENTES DE UN DIAGRAMA DE CLASES

Clase: Se representan con rectángulos divididos en tres secciones:

Nombre de la clase: Aparece en la parte superior.

Atributos: Representa las propiedades de la clase.

Métodos: Describen el comportamiento o las funciones de la clase.

Estudiante
- nombres:string - apellidos:string - edad: number - fechaNacimiento: string
+ mostrarDatos():void + calcularEdad():void

Los **atributos** identifican las **características** de la clase y los **métodos** establecen su **comportamiento**

COMPONENTES DE UN DIAGRAMA DE CLASES

Relaciones

Multiplicidad

Asociación

1 . . *

De uno a muchos

Agregación

1 . . N

De **uno** a **N**

Composición

0 . . *

De **cero** a **muchos**

Herencia

0 . . N

De **cero** a **N**

Dependencia

M

Cantidad fija

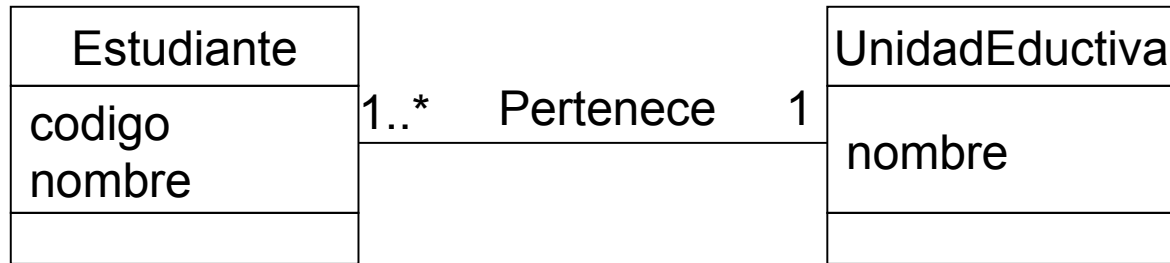
Realización

*** . . ***

De **muchos** a **muchos**

RELACIÓN DE ASOCIACIÓN

Es una relación entre clases, especificando que **los objetos de una clase deben conocer de alguna manera los objetos de la otra.**



Un estudiante pertenece a **una** Unidad Educativa
A **una** Unidad Educativa le pertenecen **1 o mas** estudiante

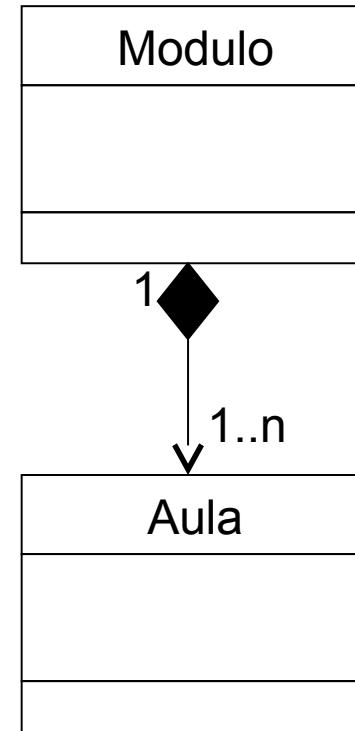
EJERCICIOS

RELACIÓN DE COMPOSICIÓN

Es un tipo de asociación fuerte entre dos clases, que se utiliza para representar una relación de "todo/parte".

Indica que una clase contiene a otra y que la vida de la clase contenida está subordinada a la vida de la clase contenedora.

Si el objeto contenedor se destruye, los objetos que componen la clase también se destruyen.



EJERCICIOS

DIAGRAMA DE CASOS DE USO

- Definir los requisitos funcionales del sistema.
- Visualiza y modela cómo interactúan los usuarios con el sistema.
- Ayuda a identificar qué funciones debe tener el sistema y cómo están conectadas.
- Es una visión general de **alto nivel** de las relaciones entre los actores y los sistemas.

COMPONENTES DE UN CASO DE USO

Actores

Representan a los usuarios o sistemas externos que interactúan con el sistema.

Pueden ser personas, otros sistemas o entidades externas,

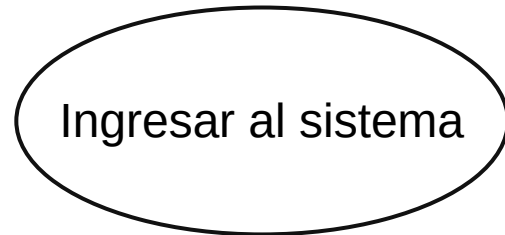


Docente

Casos de uso

Son las funcionalidades o acciones que el sistema realiza en respuesta a una interacción del actor.

Se representan con óvalos y se nombran con verbos que describen la acción



COMPONENTES DE UN CASO DE USO

Relaciones

Asociación: Une a los actores con los casos de uso.

Extensión: Representa una funcionalidad opcional que se añade a un caso de uso principal.

Inclusión: Indica que un caso de uso reutiliza la funcionalidad de otro.

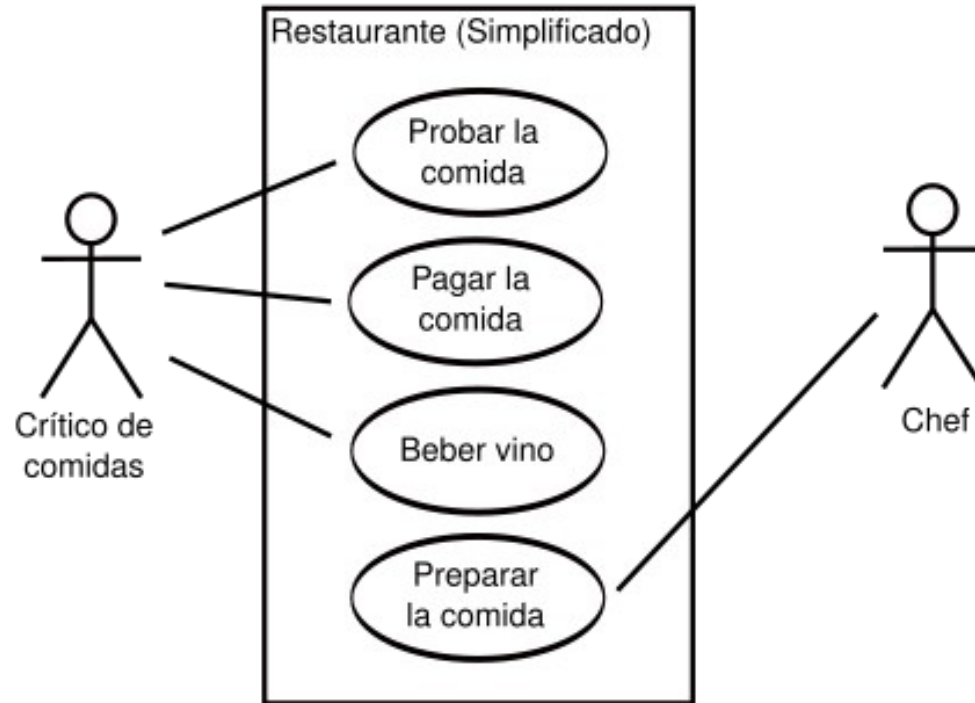


Sistema

Se representa como un rectángulo que contiene los casos de uso y define los límites del sistema modelado



Ejemplo



EJERCICIOS