

METODOLOGÍA DE DESARROLLO DE SISTEMAS II

GUÍA DE APRENDIZAJE 2.2

DIAGRAMA DE CLASES

Alumnos:

Bruno Francisco

Falcone Renzo

Marani Ignacio

Pascual Tomás

Sebes Ramiro

2022

1. ¿Cómo se representa una clase en UML? ¿Y un objeto?

Las clases son plantillas para crear objetos e implementar un comportamiento en un sistema. En UML, una clase representa un objeto o un conjunto de objetos que comparte una estructura y un comportamiento comunes. Se representan con un rectángulo que incluye filas del nombre de la clase, sus atributos y sus operaciones.

Los objetos se representan con un rectángulo con su nombre y el de su clase en la parte superior subrayados y separados por dos puntos. En caso de ser un objeto anónimo no se escribe su nombre, dejando solo el de la clase.

2. ¿Cómo se representa un atributo y una operación

Un atributo es una propiedad de una clase que es compartida por todos los elementos de esa clase. Los atributos de las instancias se modelan con diagramas a nivel de las instancias. En lugar de un diagrama de clases, se dibuja por ejemplo un diagrama de objetos.

Una operación es una abstracción de algo que se puede hacer a un objeto y que es compartido por todos los objetos de la clase.

Una operación UML se representa con un rectángulo que incluye filas del nombre de la clase, sus atributos y sus operaciones.

3. ¿Cuál es el objetivo de establecer una visibilidad en los atributos y operaciones de una clase? ¿Cómo vincula este concepto con el de encapsulamiento?

Se pueden establecer distintos niveles de encapsulación para los miembros de una clase (atributos y operaciones) en función de desde dónde queremos que se pueda acceder a ellos.

El encapsulamiento se refiere a la agrupación de datos con los métodos que operan en esos datos, o la restricción del acceso directo a algunos de los componentes de un objeto. La encapsulación se utiliza para ocultar los valores o el estado de un objeto de datos estructurados dentro de una clase,

4. Explique los ámbitos de visibilidad privada, publica y restringida de atributos y operaciones

Visibilidad	Significado	UML
Pública	Se puede acceder al miembro de la clase desde cualquier lugar.	+
Restringida	Sólo se puede acceder al miembro de la clase desde la propia clase o desde una clase que herede de ella.	#

Privada	Sólo se puede acceder al miembro de la clase desde la propia clase.	-
---------	---	---

5. ¿Qué es una clase abstracta? ¿Cómo se representa? ¿Para qué se utiliza?

Son clases a las que no puede realizarse una instancia, se utilizan en jerarquías de generalización. Las clases abstractas tienen, al menos, una operación abstracta. Una operación abstracta tiene que ser implementada por algún método en un nivel más bajo de abstracción.

6. ¿Qué es una interfaz? ¿Qué diferencia tiene con una clase abstracta?

Una interfaz es una colección de operaciones que especifican un servicio de una clase o un componente.

También especifica el comportamiento visible de forma externa a la clase.

Una interfaz define un conjunto de especificaciones de operaciones, pero no su implementación.

A pesar de que la interfaz y la clase abstracta no pueden tener instancias directas, una clase abstracta puede tener operaciones concretas.

7. ¿Qué es una dependencia?

Es una relación de uso entre clases, declara que un elemento utiliza información de otro elemento.

En general se utiliza el contexto de las clases para indicar que una utiliza las operaciones de otra o que utiliza variables o parámetros cuyo tipo viene dado por la otra clase.

En los diagramas de clase, se utilizan para describir la visibilidad entre clases que no es de tipo atributo.

8. ¿Qué representa una asociación entre clases?

Una asociación es una relación estructural que especifica que los objetos de una clase están conectados con los de otra.

Son relaciones entre clases que indican alguna conexión significativa que deseamos preservar a lo largo del tiempo.

a. Explique nombre de rol y multiplicidad

- Nombre de rol: implica que cada objeto juega un rol específico en la asociación
- Multiplicidad: define cuántas instancias de una clase A pueden asociarse con las de una clase B. Representa un rango de enteros que especifican el tamaño posible del conjunto de objetos relacionados (v gr.: 0..1; 1..1; 0..*; 1..*). El valor de la multiplicidad indica cuántas instancias se pueden asociar con otras en un momento concreto.

9. ¿Cómo se instancias las clases y las asociaciones?

Se instancian mediante una asociación (enlace) de una tupla de referencias a objetos.

10. ¿Cómo se representa una asociación?

a. unaria

Se representa mediante un bucle que sale y entra a la misma clase.

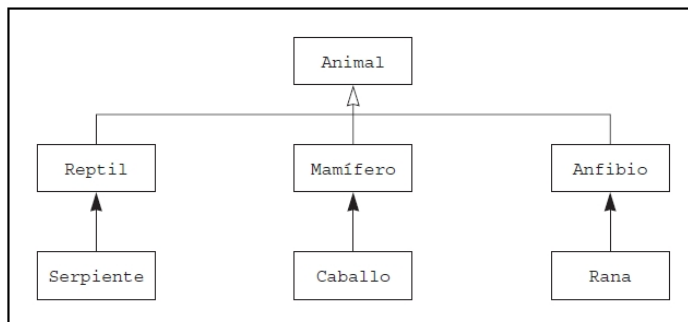
b. binaria

Se representa mediante una flecha que sale de una clase y llega a otra clase, ambas clases que se quieran referenciar.

11. ¿Cómo se representa una generalización?

La generalización es una relación entre dos clases de modo que una, la subclase o clase hija, se considera como forma especializada de la otra, la superclase o clase padre. La superclase se considera como generalización de la subclase y la subclase se considera una especialización de la superclase.

También se suele denominar relación de herencia ya que representa el mismo concepto del paradigma de orientación a objetos.



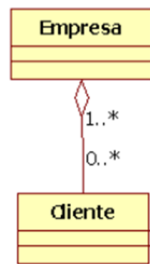
a. ¿En qué casos la utilizaría?

La herencia o generalización la utilizaría en programación orientada a objetos con el fin de definir nuevas clases basadas de unas ya existentes a fin de reutilizar un código.

12. ¿Cómo se representa agregación y una composición?

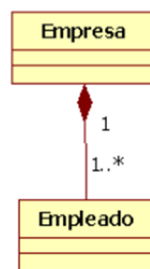
La agregación se representa con un rombo vacío ◇ mientras que la composición con un rombo relleno ◆

Ejemplo de agregación:



- Tenemos una clase Empresa.
- Tenemos una clase Cliente.
- Una empresa agrupa a varios clientes.

Ejemplo de composición:



- Tenemos una clase Empresa.
- Un objeto Empresa está a su vez compuesto por uno o varios objetos del tipo empleado.
- El tiempo de vida de los objetos Empleado depende del tiempo de vida de Empresa, ya que si no existe una Empresa no pueden existir sus empleados.

a. ¿Cuándo usaría una u otra?

La agregación se utilizaría para aquellas relaciones que no posee dependencia existencial, mientras que la composición sería utilizada para aquellas relaciones en las que si existe una dependencia existencial.

13. ¿Cómo se representa una clase asociación?

Las asociaciones representan las relaciones más generales entre clases, es decir, las relaciones con menor contenido semántico.

La forma de representar las asociaciones binarias en UML es mediante una línea que conecta las dos clases. En general, las asociaciones son bidireccionales, esto es, no tienen un sentido asociado.

a. ¿Para qué se utiliza?

Para UML una asociación va a describir un conjunto de vínculos entre las instancias de las clases.

Las asociaciones pueden ser binarias (conectan dos clases) o n-arias (conectan n clases), aunque lo más normal en un modelo es utilizar sólo relaciones binarias.

¿Qué es un paquete y para que se usa?

Los paquetes son una forma de agrupar clases es interfaces asociadas. Habilitan los grupos de clases para estar disponibles sólo si son necesarios y eliminan los conflictos potenciales entre los nombres de clases en grupos diferentes de clases.