Método orientado a objetos

- **Carrera:** Licenciatura en Informática
- **Asignatura:** Programación con objetos I
- **Profesores:**
  - » **Mendoza, Dante.**

UNIVERSIDAD NACIONAL DEL OESTE

Temario

- Modelado de un sistema.
- Diagramas UML.
- Vistas del modelado de un sistema.
- Diagrama de clases.
- Herramientas UML.

Modelado de un sistema

Modelado de un sistema

El modelado es una parte importante en la construcción de un sistema ya que se desarrollan modelos abstractos del mismo, cada modelo representa una perspectiva diferente de ese sistema.

El modelado nos brinda una visualización del sistema a desarrollar, entender mejor sus funcionalidades y es utilizado para comunicarse con el cliente.

¿Cómo lo representamos?

Diagramas UML

Diagramas UML

La representación de un sistema se logra mediante la aplicación de un estándar como el Lenguaje de Modelado Unificado (UML).

Es posible desarrollar modelos tanto del sistema existente para ayudarnos a comprender lo que hace, como también del sistema a diseñar, es decir el sistema nuevo. UML es el estándar utilizado en el modelado orientado a objetos.

Vistas del modelado de un sistema

Vistas del modelado de un sistema

En el modelado del sistema existen cuatro vistas diferentes, cada una de las cuales enfatiza ciertos aspectos que están estrechamente relacionados entre sí.

La Vista Externa: Aquí se modela el entorno o contexto del sistema. Por ejemplo los diagramas de casos de uso.

La Vista Estructural: Se modelan la estructura de los datos del sistema. Por ejemplo diagramas de clases.

Vistas del modelado de un sistema

La Vista de Comportamiento: Se muestra cual es el comportamiento del sistema y como responde a eventos. Por ejemplo diagramas de estados y diagramas de actividades

La Vista de Interacción: Nos ilustra la interacción del sistema con su entorno o entre componentes. Por ejemplo los diagramas de secuencia.

Diagrama de clases

Diagrama de clases

Antes de comenzar con la explicación de los diagramas de clases debemos entender el concepto de objeto, por lo tanto preguntaremos

¿Qué es un objeto?

Diagrama de clases

Un objeto es una representación de un elemento del mundo real o entorno que nos rodea.

Para realizar dicha representación se utilizan en UML los diagramas de clases

Diagrama de clases

Un diagrama de clases nos muestra la estructura que tendrá un sistema mostrando las clases que este contendrá.

Las clases están conformadas por atributos, operaciones (acciones o métodos) y las relaciones con otros objetos.

Elementos de una clase

Nombre: El nombre de la clase se encuentra en la parte superior y debe respetar el comienzo de la primera letra en mayúscula.

En caso de contener dos o más palabras, se las une y cada una comienza con la primera letra que la conforma en mayúscula.

Ejemplo: Lavadora

LavadoraIndustrial

Elementos de una clase

Atributos: Los atributos son una característica o propiedad de la clase. Deben de respetar estar escritas en minúsculas, en caso de contener dos o más palabras debe de escribirse la primera letra en minúscula y la o las palabras siguientes con su primera letra en mayúscula (Camel Case).

Ejemplo: teléfono

telefonoPersonal

Elementos de una clase

Elementos de una clase

Operaciones (Métodos): Las operaciones son las acciones que puede realizar la clase.

Al igual que los atributos estas deben de respetar el tipo de escritura en Camel Case.

Elementos de una clase

Tanto los atributos como las operaciones incluyen al principio de su descripción la visibilidad que tendrá.

Esta visibilidad se identifica escribiendo un símbolo que podrá ser alguna de las siguientes:

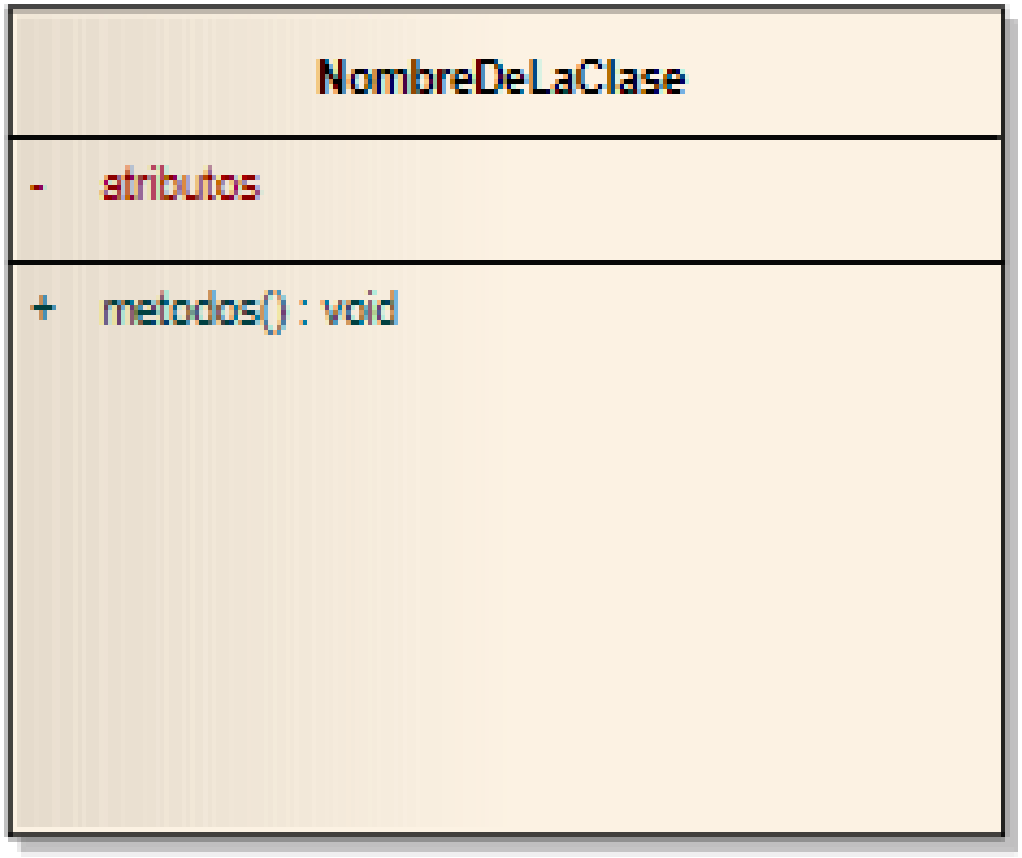
Elementos de una clase

- **(+) Pública.** Representa que se puede acceder al atributo o función desde cualquier lugar de la aplicación.

- **(-) Privada.** Representa que se puede acceder al atributo o función únicamente desde la misma clase.

- **(#) Protegida.** Representa que el atributo o función puede ser accedida únicamente desde la misma clase o desde las clases que hereden de ella (clases derivadas).

Esqueleto de una clase



Ejemplo

Imaginemos que deseamos representar mediante un diagrama de clases el objeto perro. Pensemos los siguientes puntos importantes:

- Características de un perro.
- Acciones que realiza un perro.

Ejemplo

Ejemplo

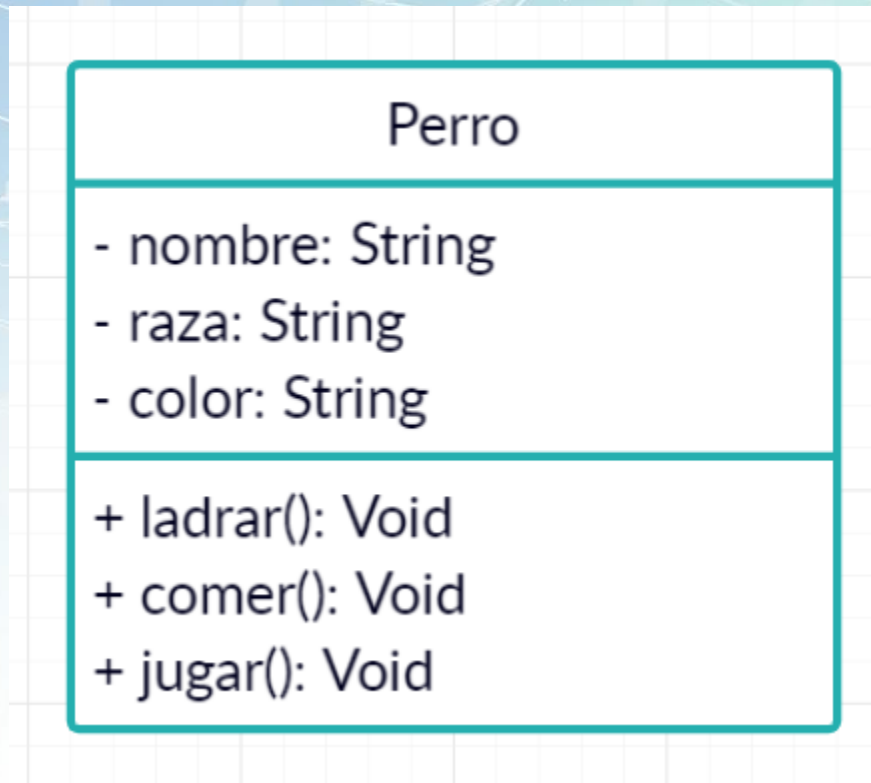
Características:

- Nombre.
- Raza.
- Color.

Acciones:

- Ladrar.
- Comer.
- Jugar.

Ejemplo


Perro
- nombre: String
- raza: String
- color: String
+ ladrar(): Void
+ comer(): Void
+ jugar(): Void

¿Camel Case o Snake Case?

Ambos tienen su particular forma de escribirse pero depende del ámbito del proyecto y de las decisiones tomadas respecto a que estándar aplicar.

Camel Case: lavarRopa()

Snake Case: lavar_ropa()

Tipos de relaciones

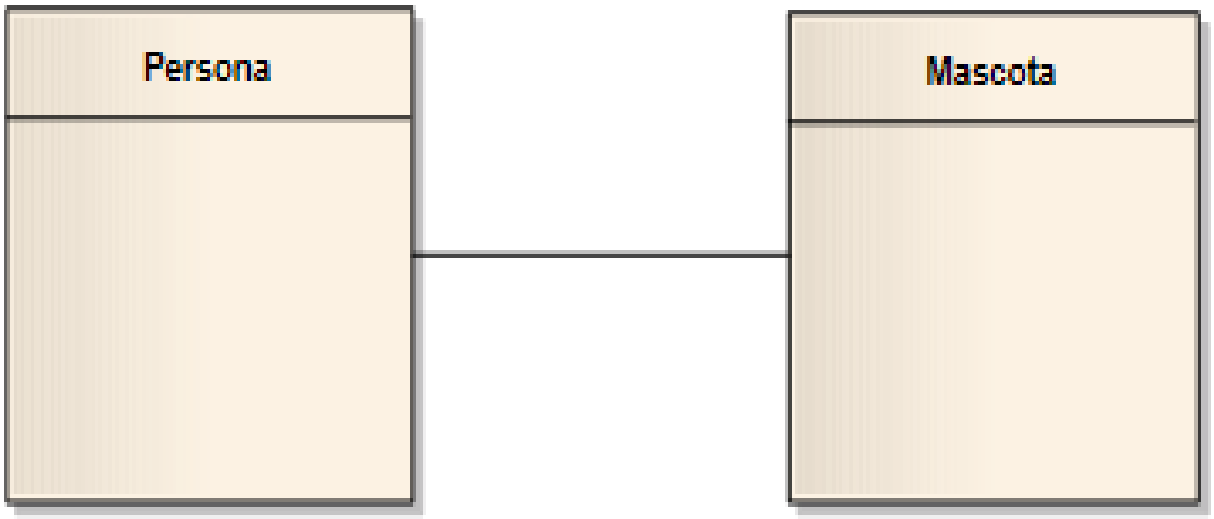
Los diagramas de clases se relacionan mediante distintos tipos de relaciones:

- Asociación.
- Agregación.
- Composición.
- Dependencia.
- Generalización.

No veremos todas ellas pero si las mas relevantes.

Tipos de relaciones

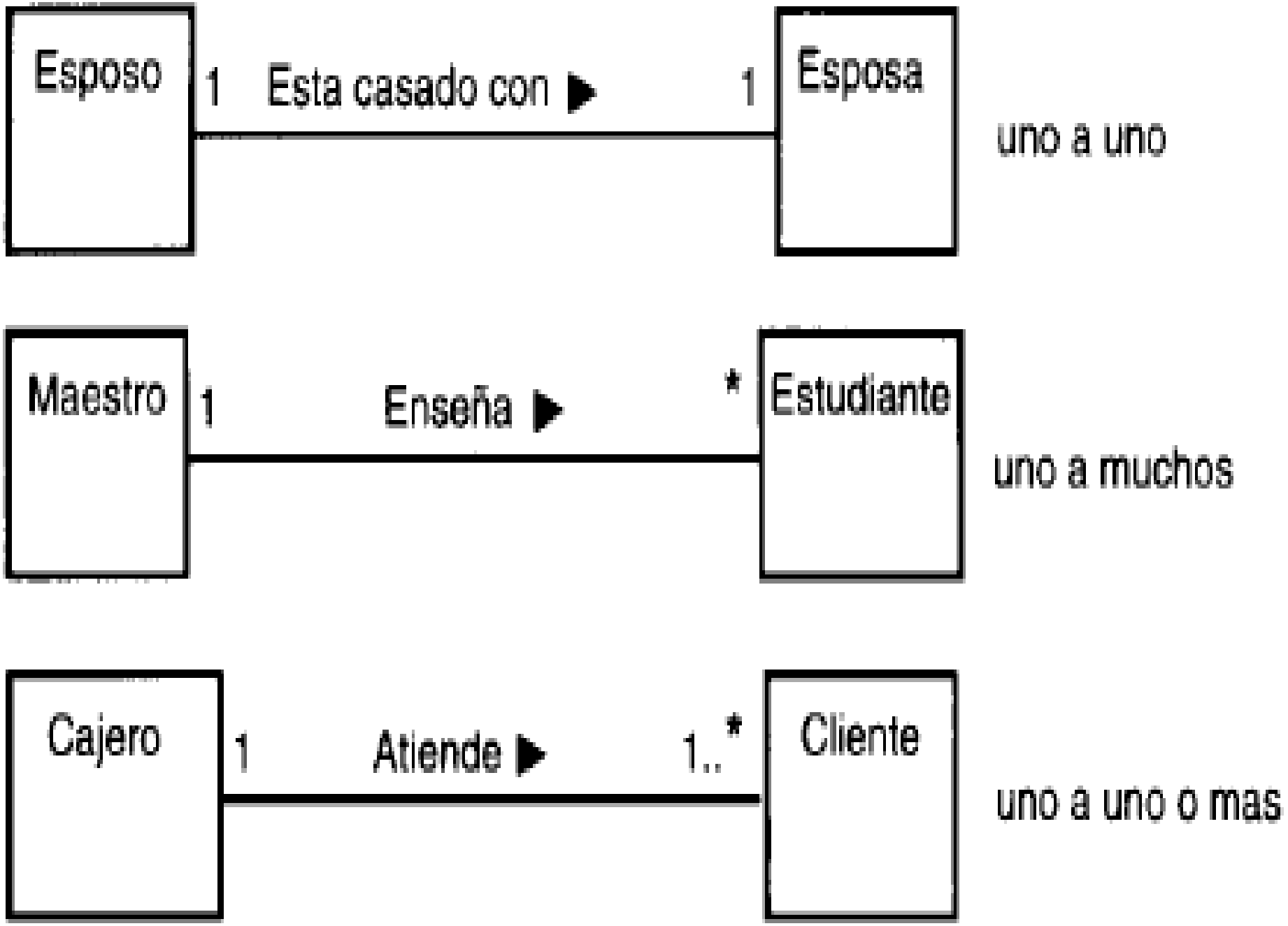
Asociación: Se representa mediante una línea continua entre las clases que conforman la asociación.



Tipos de relaciones

Multiplicidad: En el ejemplo anterior existe el problema que una persona puede tener una o muchas mascotas, por lo tanto la representación anterior no sería correcta.

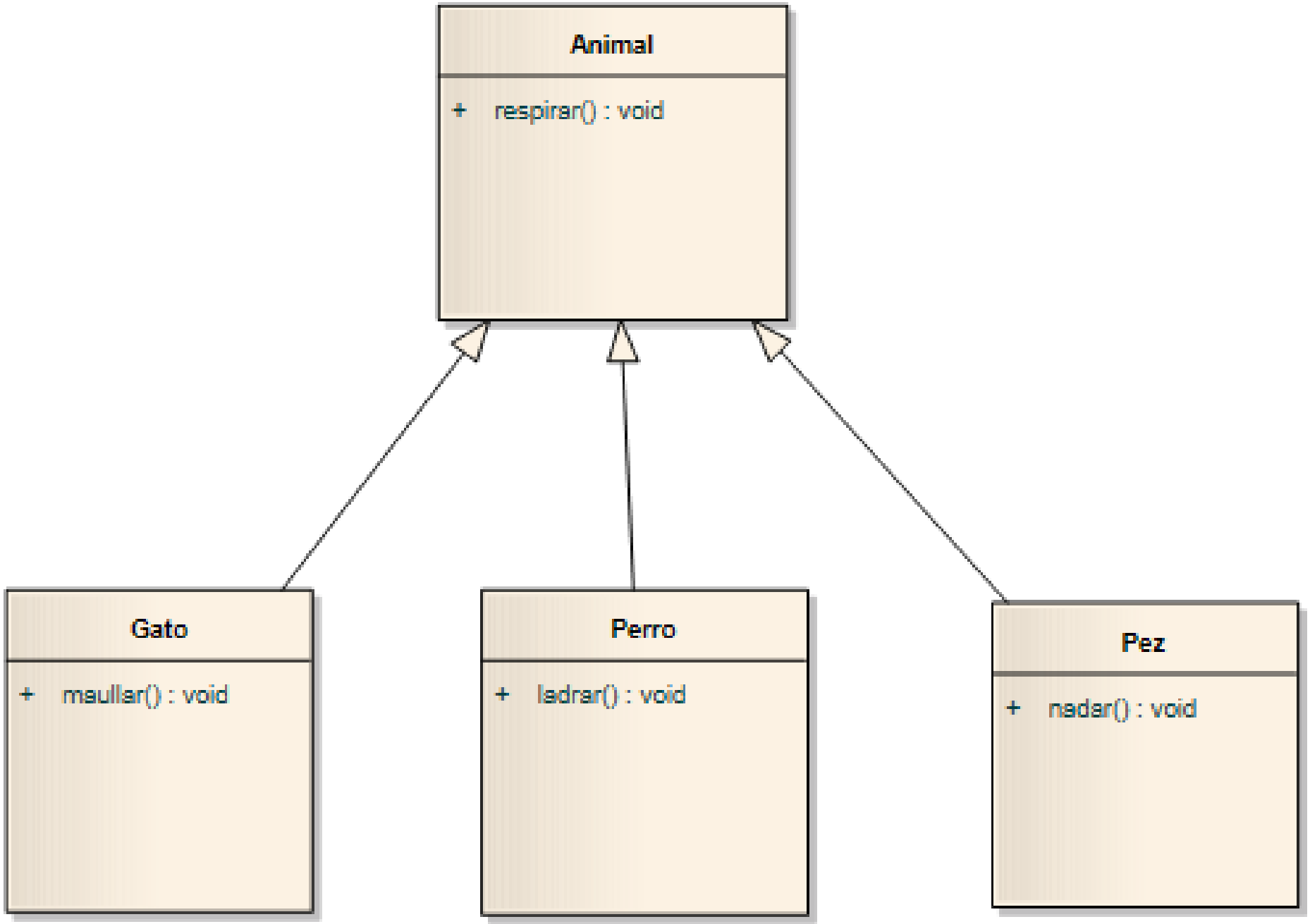
Es aquí donde la multiplicidad se hace evidente en donde nos muestra la cantidad de objetos que se relacionan de una clase con otra.


Esposo
1
Esta casado con
1
Esposa
uno a uno
Maestro
1
Enseña
*
Estudiante
uno a muchos
Cajero
1
Atiende
1..*
Cliente
uno a uno o mas

Tipos de relaciones

Generalización: Este tipo de relaciones permiten que una clase (clase hija o subclase) reciba los atributos y métodos de otra clase (clase padre o superclase).

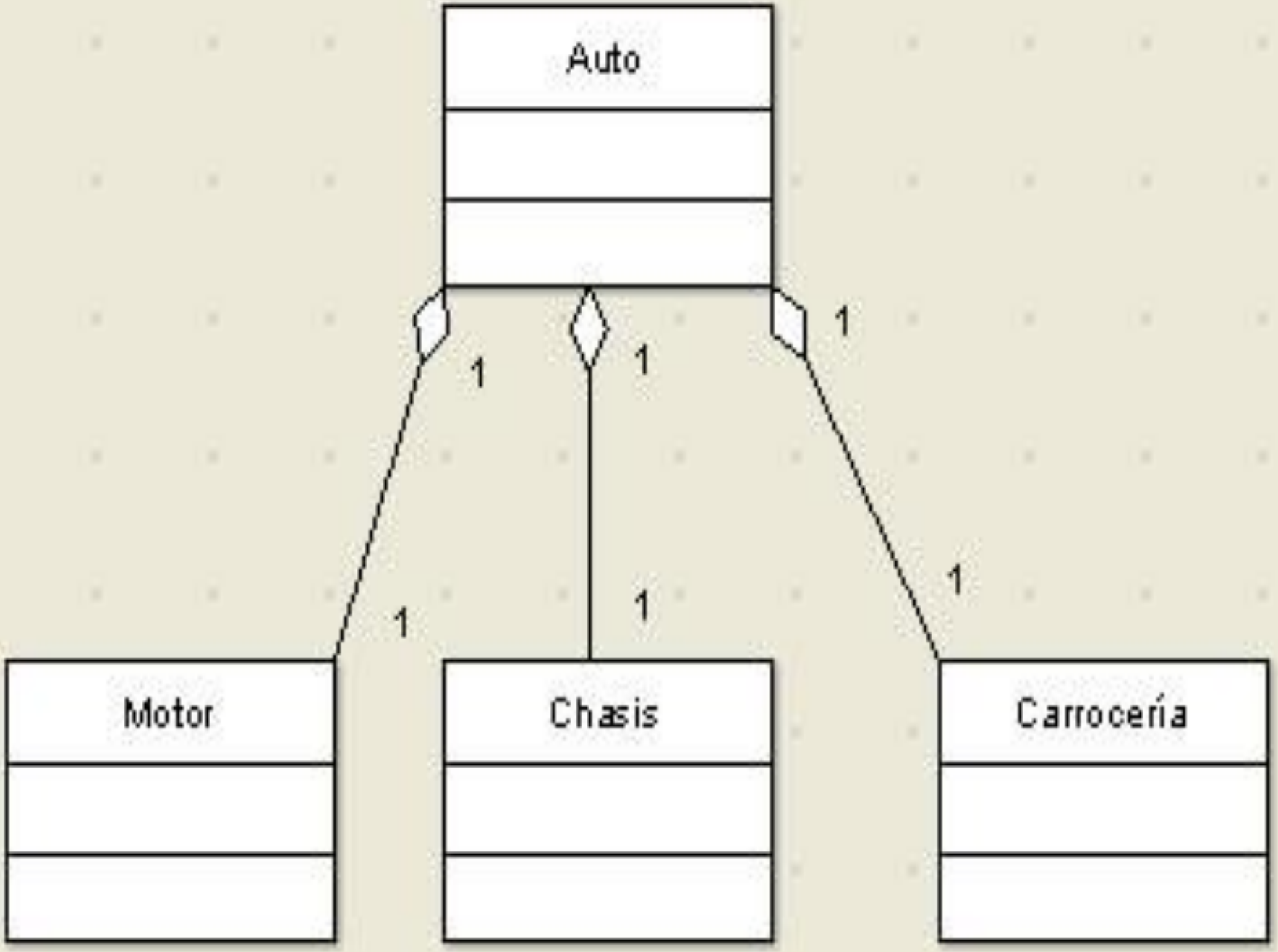
Estos atributos y métodos recibidos se suman a los que la clase tiene por sí misma. Se utiliza en relaciones “es un”.


Animal
+ respirar() : void
Gato
+ maullar() : void
Perro
+ ladrar() : void
Pez
+ nadar() : void

Tipos de relaciones

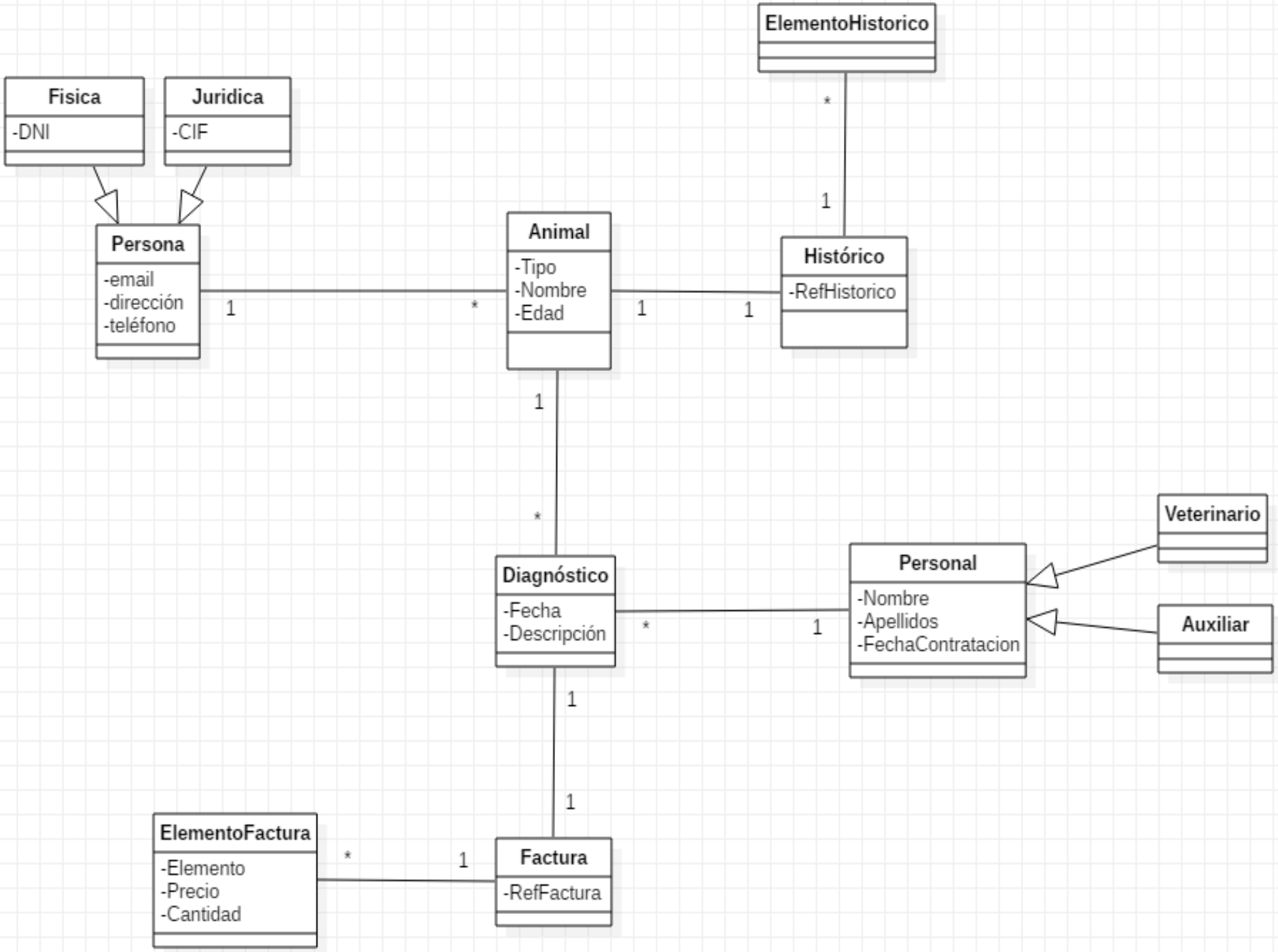
Agregación: Es una representación jerárquica que indica que una clase puede estar compuesta por otras.

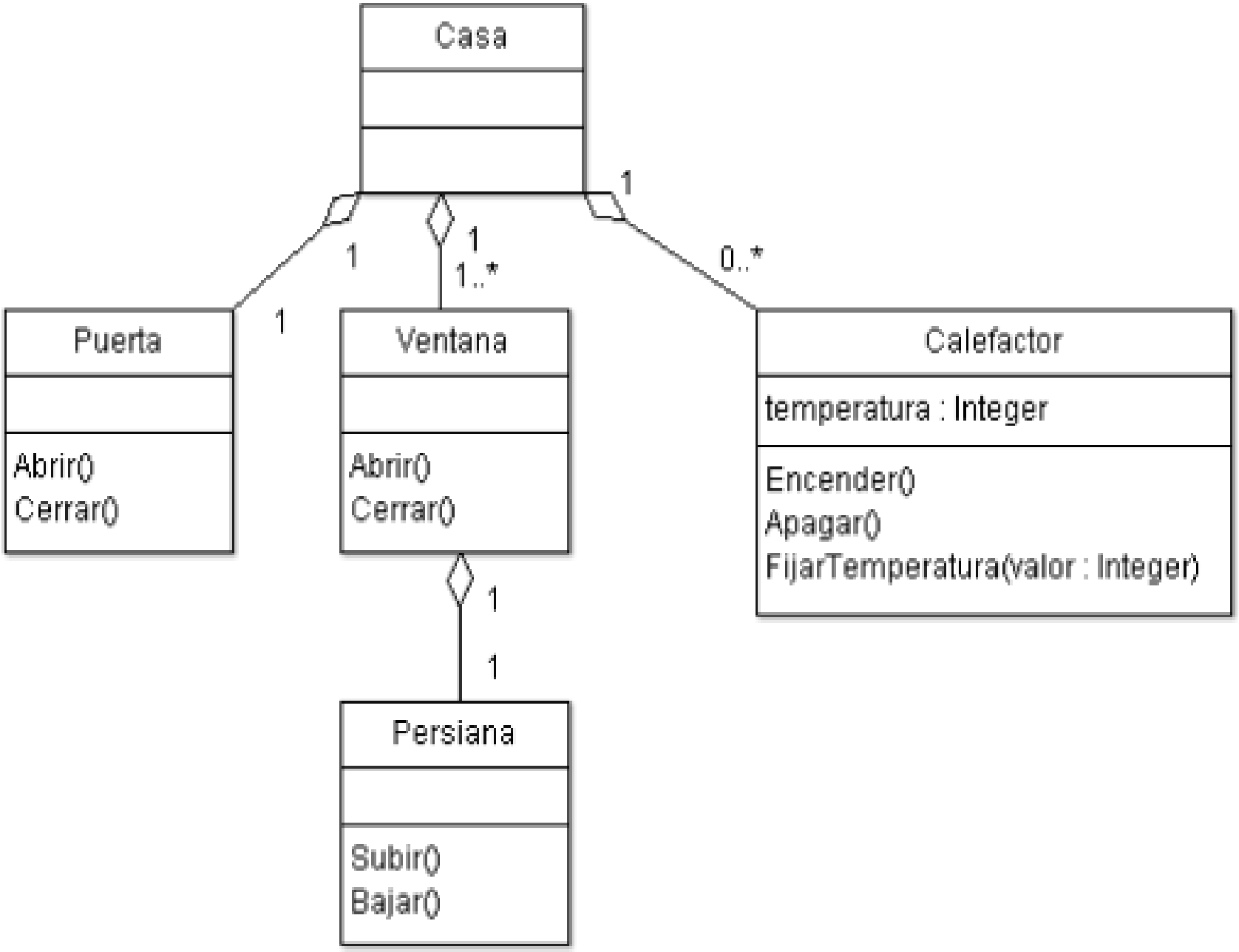
Es decir, representa relaciones en las que un objeto es parte de otro, pero aun así debe tener existencia en sí mismo.


Auto
1
1
1
1
1
1
Motor
Chasis
Carrocería

Tipos de relaciones

A continuación veremos algunos ejemplos combinados.


ElementoHistorico
Fisica
-DNI
Juridica
-CIF
*
1
Persona
-email
-dirección
-teléfono
Animal
-Tipo
-Nombre
-Edad
Histórico
-RefHistorico
1
*
1
1
1
*
Veterinario
Diagnóstico
-Fecha
-Descripción
Personal
-Nombre
-Apellidos
-FechaContratacion
Auxiliar
*
1
1
1
ElementoFactura
-Elemento
-Precio
-Cantidad
Factura
-RefFactura
*
1


Casa
1
1
1
1..*
0..*
Puerta
Abrir()
Cerrar()
Ventana
Abrir()
Cerrar()
Calefactor
temperatura : Integer
Encender()
Apagar()
FijarTemperatura(valor : Integer)
1
1
Persiana
Subir()
Bajar()

Herramientas UML

Herramientas UML: opciones online

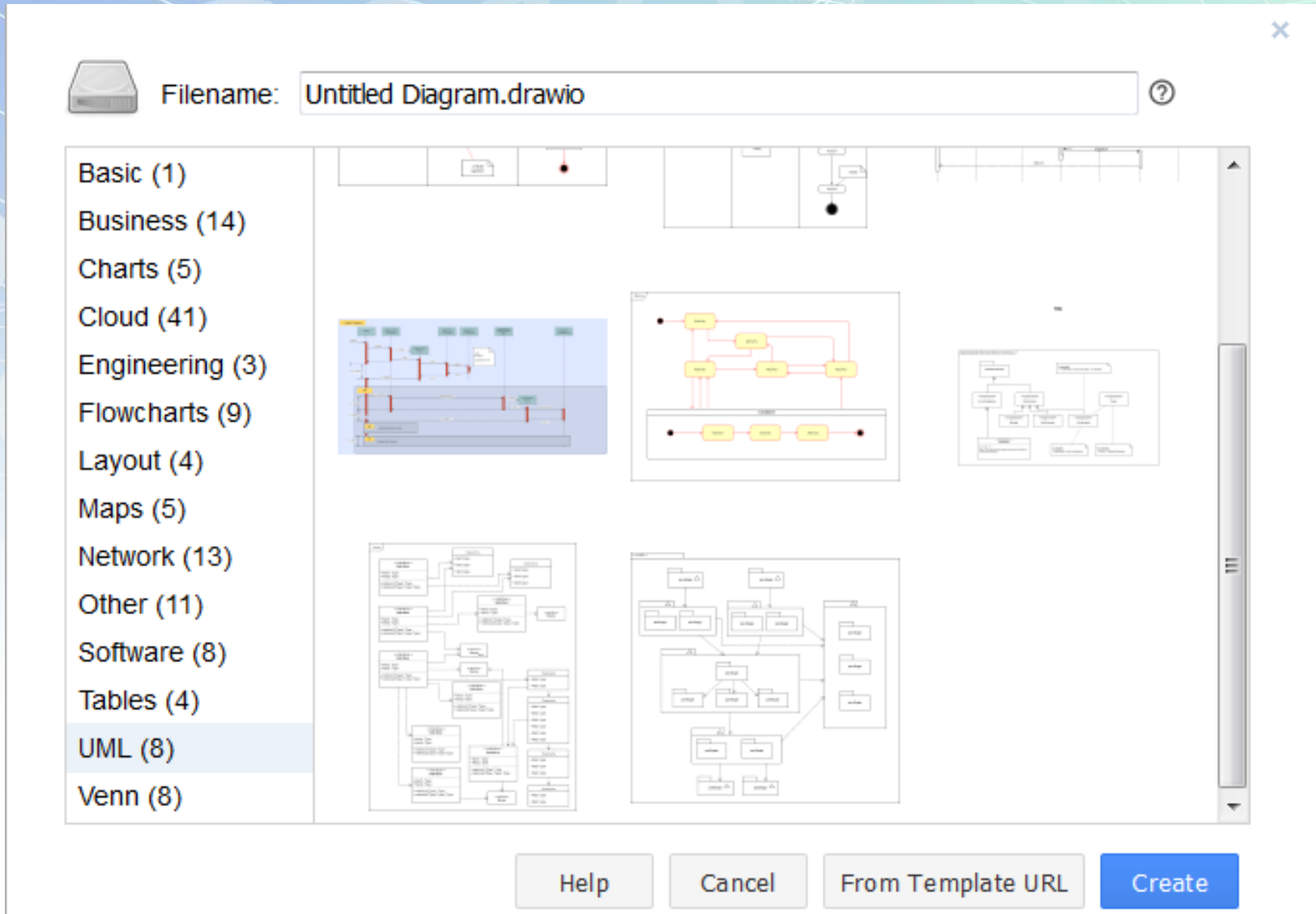
Draw

https://www.draw.io/

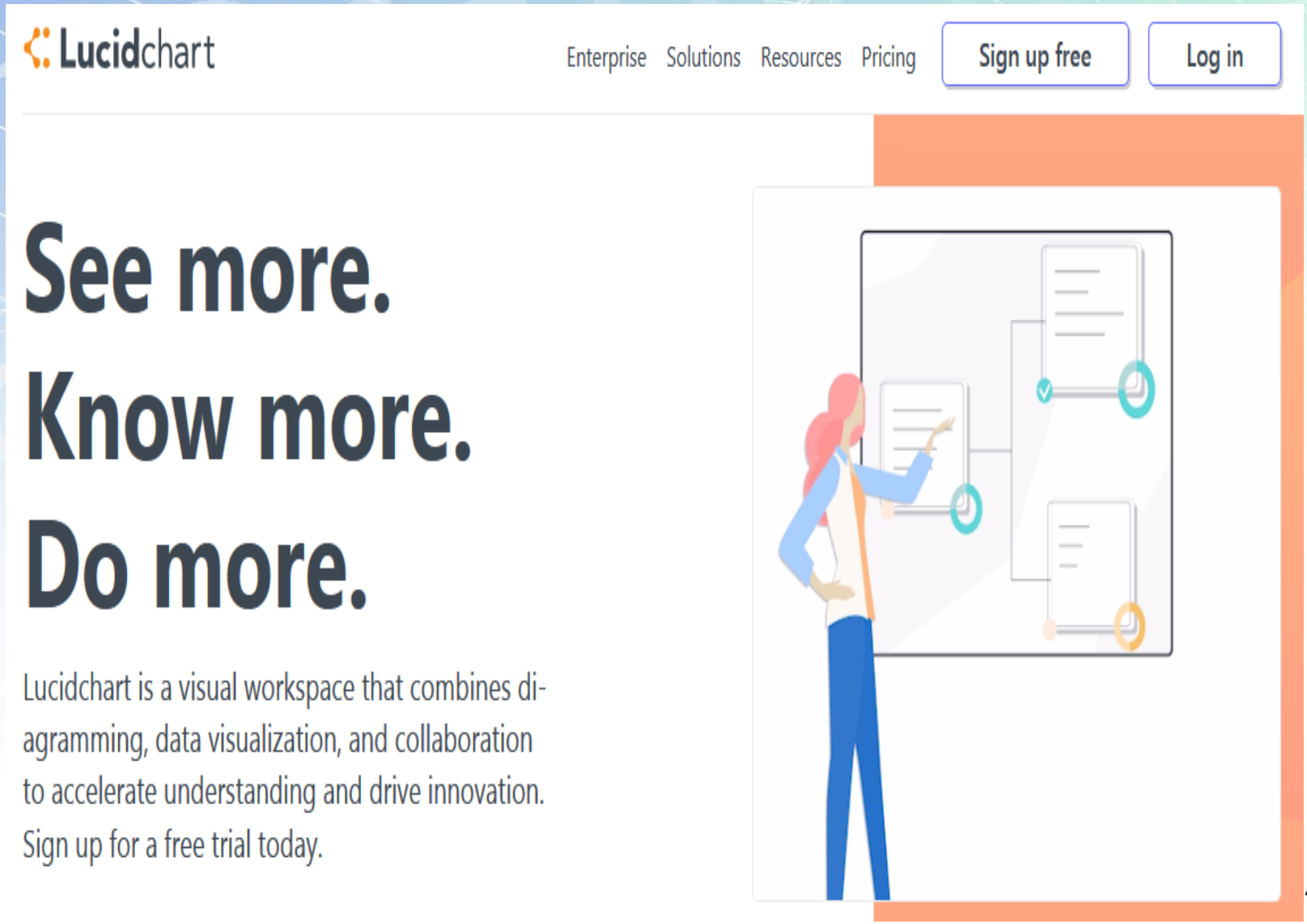
LucidChart

https://www.lucidchart.com/

Herramientas UML: opciones online



Herramientas UML: opciones online



Herramientas UML: Opciones de escritorio

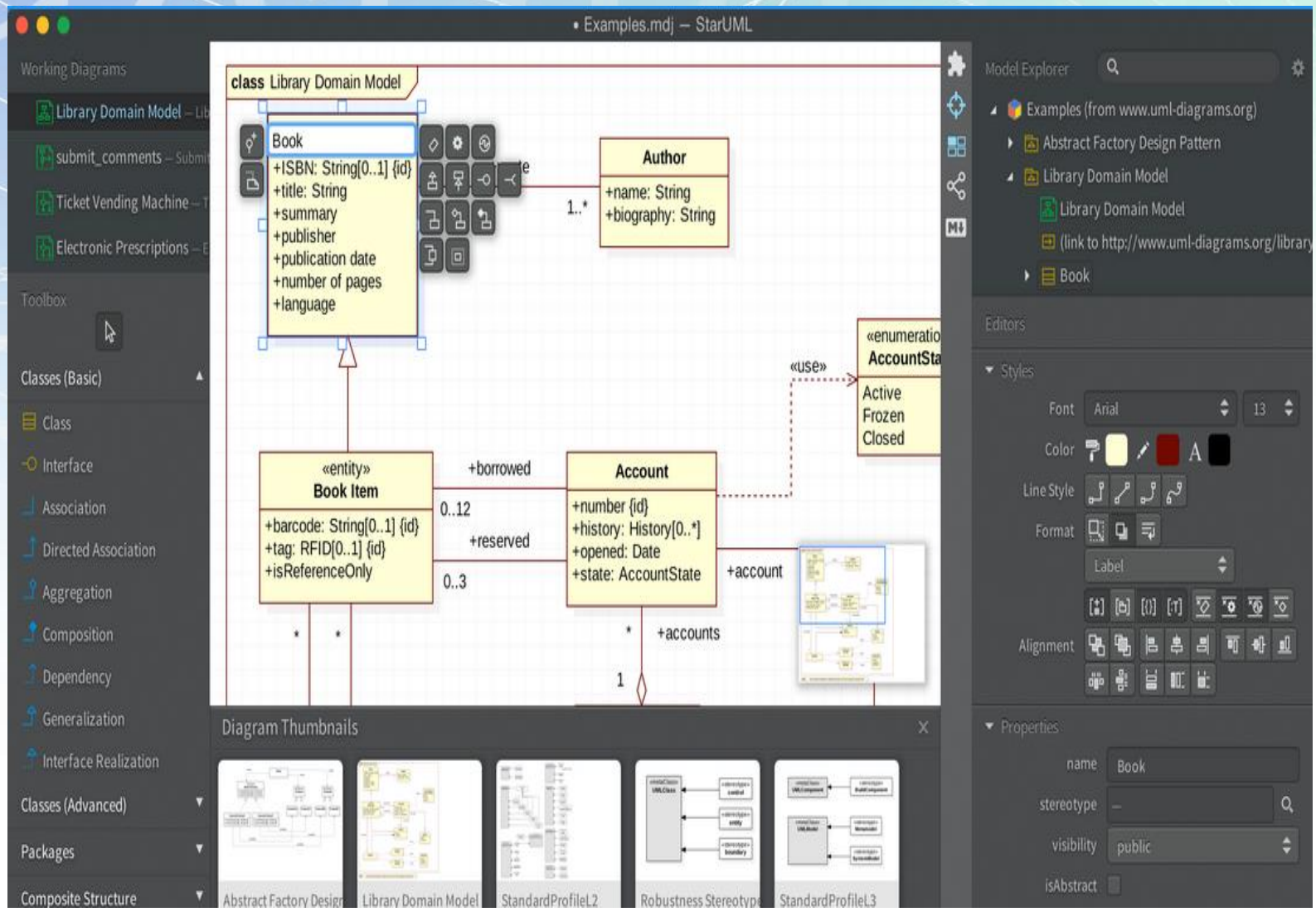
StarUML

http://staruml.io/

Enterprise Architect

https://sparxsystems.com/products/ea/downloads.html

Herramientas UML: Opciones de escritorio



Herramientas UML: Opciones de escritorio

Fuentes

- Deitel, P. J., & Deitel, H. M. (2008). Java: como programar. Pearson Educación.
- DiagramasUML. (2019). DiagramasUML: Diagrama de clases. Recuperado de https://diagramasuml.com/diagrama-de-clases/
- Fowler, M., & Scott, K. (1999). UML gota a gota. Pearson Educación.
- Junta de Andalucia. (2016). Guía para la redacción de casos de uso: Marco de Desarrollo de la Junta de Andalucía. Recuperado de http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416

Fuentes

- Karla Cevallos. (2015). UML: Casos de Uso. Recuperado de https://ingsotfwarekarlacevallos.wordpress.com/2015/06/04/uml-casos-de-uso/
- Schmuller, J. (2000). Aprendiendo UML en 24 horas. Pearson educación.
- Source Making. (2019). Source Making: Design Patterns and Refactoring. Recuperado de https://sourcemaking.com/es/uml/modeling-it-systems

¿PREGUNTAS?

MUCHAS GRACIAS POR SU ATENCIÓN.