

Diagrama de clases de análisis (Modelo Conceptual de Clases)

Modelo de Clases de análisis (o modelo de datos)

Objetivo

- Los diagramas de clase son el pilar básico del modelado con UML, siendo utilizados tanto para mostrar lo que el sistema puede hacer (análisis), como para mostrar cómo puede ser construido (diseño).
- Durante el análisis de un sistema se pueden realizar Diagramas de clases, a nivel conceptual, sólo con las clases del dominio del problema (sin detalles técnicos), y solo con atributos, sin detallar las operaciones.
 - En este caso el Diagrama de Clases sería aproximadamente equivalente al tradicional Diagrama de Entidad Relación (ERD)

Diagrama de Clases

Identificación de clases de análisis (entidades o datos)

Paso 1: Realizar un análisis sintáctico-gramatical de la documentación existente:

- Utilizar la documentación de los casos de uso.
- Subrayar cada nombre (sustantivo) o cláusula nominal.

Paso 2: A partir de los nombres subrayados, proponer varios objetos potenciales:

- Entidades externas, cosas, ocurrencias, papeles o roles, organizaciones, lugares, estructuras, etc.

Diagrama de Clases

Identificación de clases de análisis

Paso 3: Decidir qué objetos se admiten como objetos del sistema, basándose en:

- **Información retenida:** si la información debe ser "recordada" para que el sistema funcione.
- **Funcionalidad:** debe tener algún tipo de funcionalidad (operaciones) que justifique su existencia.
- **Múltiples atributos:** un objeto con un atributo "seguramente" será atributo y no objetos.
- **Atributos comunes:** todos los objetos de una clase tienen los mismo atributos.
- **Funcionalidad común:** todos los objetos de una clase tienen las mismas operaciones.

Diagrama de Clases

Clases abstractas

- Una Clase abstracta es una clase que no puede existir en la realidad, pero que es útil conceptualmente.
- Las clases abstractas no son instanciables directamente sino en sus descendientes. Una clase abstracta suele ser situada en la jerarquía de clases en una posición que le permita ser un depósito de métodos y atributos para ser compartidos o heredados por las subclases de nivel inferior.

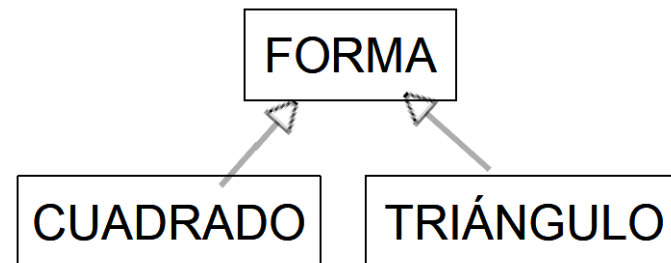


Diagrama de Clases

Identificación de atributos y operaciones

- Los **atributos** se obtienen de la lista de objetos candidatos y de la descripción del sistema:
 - Los objetos descartados por simples serán atributos.
- Las **operaciones** de los objetos se derivan de los verbos que aparecen en la descripción del sistema:
 - Los parámetros de las operaciones se derivan de la información intercambiada por los objetos que interactúan.
 - Las operaciones se establecen en la fase de Diseño.

Diagrama de Clases

Notación de las clases

- El **nombre** de la clase aparece centrado y si la clase es abstracta se representa en cursiva. El **estereotipo**, si se muestra, se sitúa sobre el nombre y entre el símbolo: <<..>>
- **Atributos:** visibilidad nombre : tipo
= valor-inicial { propiedades }
- **Operaciones o Métodos:**
visibilidad nombre (lista-de-parámetros): tipo-devuelto { propiedad }
- **Visibilidad:** publica (+), privada (-) o protegida (#)

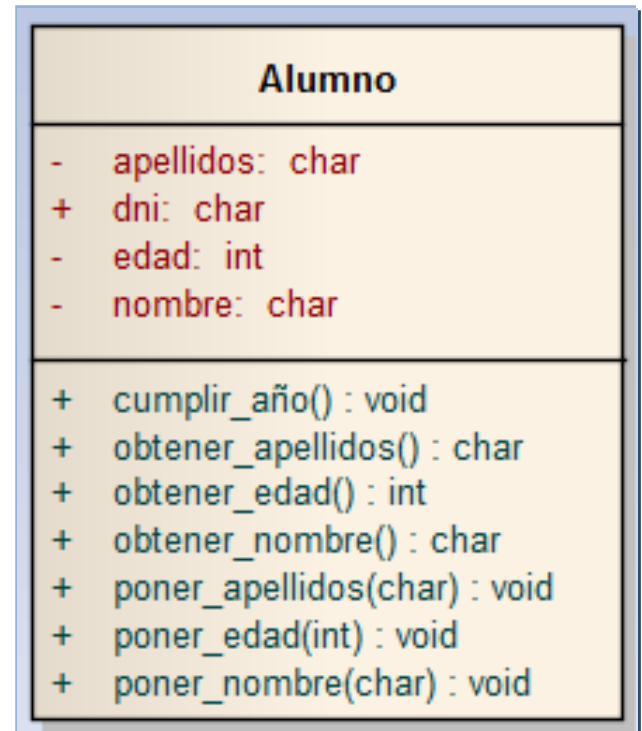


Diagrama de Clases

Identificación de relaciones

Paso 1: Las relaciones pueden derivarse examinando la interacción entre objetos:

- Si dos objetos interactúan ("hablan") debe haber un camino para la comunicación.
- Una interacción implica relación de dependencia entre las clases.

Paso 2: El resto de relaciones se obtienen analizando la estructura de la información del sistema:

- Expresiones como: "es", "tiene", "consta de", en la descripción del sistema, sugieren la existencia de relaciones entre objetos.

Diagrama de Clases

Relaciones de asociación

- Una relación de asociación se representa como una línea continua entre las clases asociadas que representa el sentido de la navegación. Se representan, también, la multiplicidad o cardinalidad (min..max) y el nombre de la asociación.

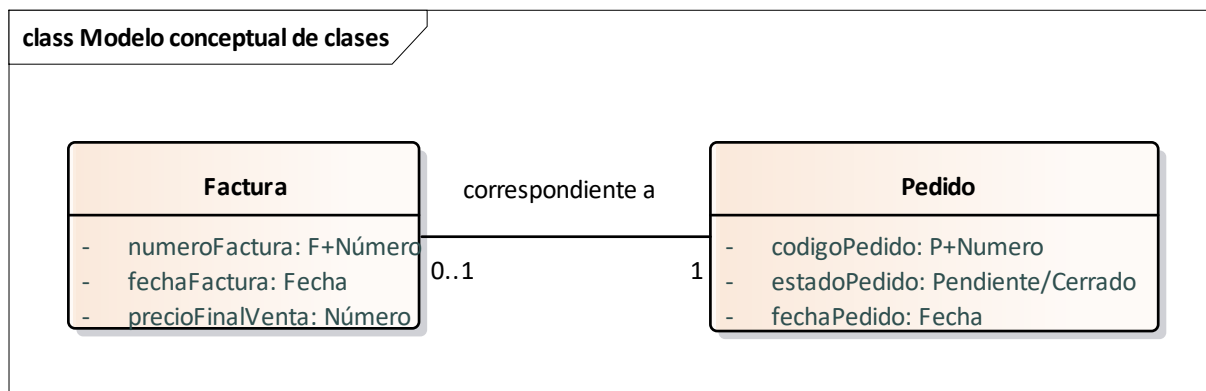


Diagrama de Clases

Relación con clase de asociación

- Entre relaciones de asociación con multiplicidad máxima * de suele utilizar una clase para modelar los datos o atributos asociados a cada instancia de la relación.

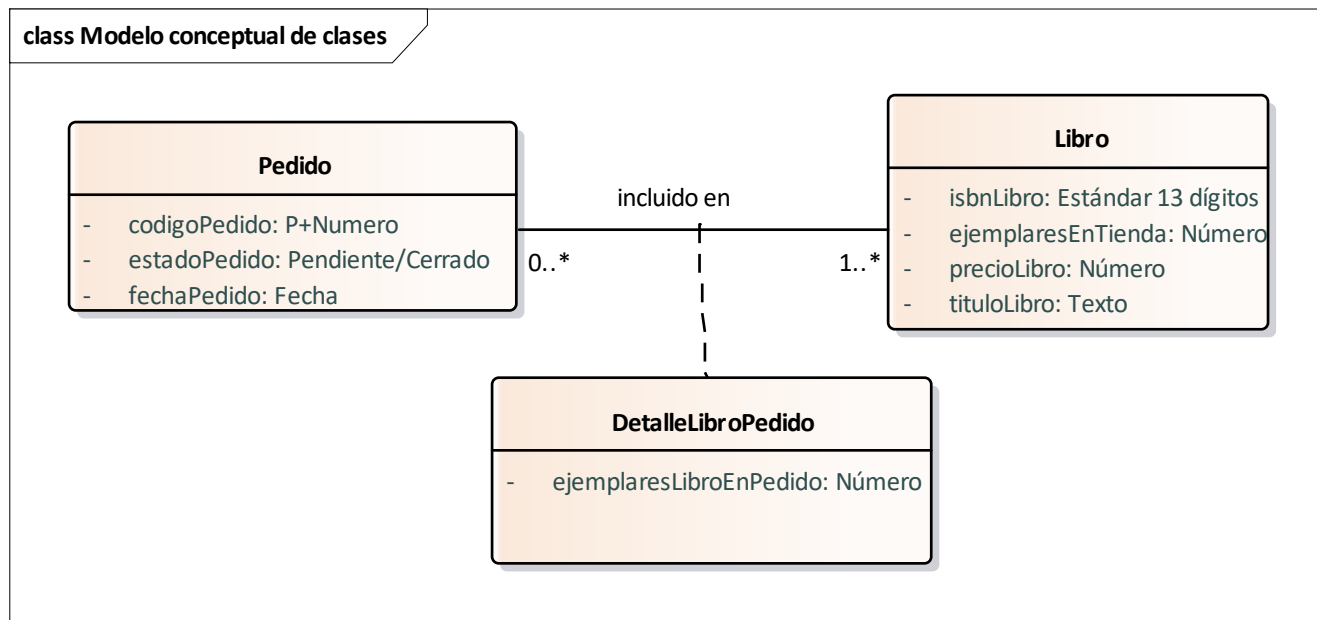


Diagrama de Clases

Relaciones de agregación y composición

- **Agregación:** Se representa con un rombo hueco en la clase cuya instancia es una agregación de las instancias de la otra.
- **Composición:** Se representa con un rombo lleno en la clase cuya instancia contiene las instancias de la otra clase.

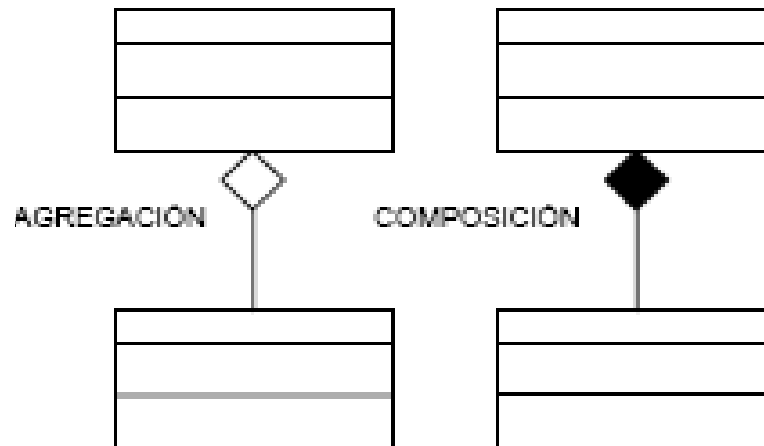


Diagrama de Clases

Relaciones de herencia y dependencia

- **Dependencia:** Una línea discontinua con una flecha apuntando a la clase cliente.
- **Herencia:** Esta relación se representa como una línea continua con una flecha hueca en el extremo que apunta a la superclase.

