

Modelo de Dominio

Introducción

El modelo de dominio sirve para familiarizarse con el vocabulario del tema de la aplicación a desarrollar. Por ejemplo: Si un psiquiatra nos pide un sistema para que dada una patología de un paciente le muestre las distintas combinaciones de fármacos que le puede recetar teniendo en cuenta la compatibilidad con otros fármacos que toma el paciente para otras enfermedades. Como analistas tendremos que averiguar sobre este tema si nunca hicimos un sistema de este tipo, para que en las futuras entrevistas entendamos lo que nos dice el psiquiatra.

El modelo de dominio captura los tipos de clases más importantes en el contexto del sistema. Las clases del dominio representan “cosas” que existen o los eventos que suceden en el entorno en el que trabaja el sistema.

Varias de las clases del dominio pueden obtenerse de una especificación de requerimientos inicial o la entrevista con los expertos del dominio. Las clases del dominio aparecen en formas típicas:

- *Objetos del negocio que representan cosas que se manipulan en el negocio, como pedidos, cuentas y contratos.*
- *Objetos del mundo real y conceptos de los que el sistema debe hacer un seguimiento, misiles y trayectorias.*
- *Sucesos que ocurrirán o han ocurrido, venta, compra, reserva, préstamo.¹*

El modelo del dominio se describe mediante diagramas de UML (diagrama de clases).

Un modelo del dominio muestra (a los modeladores) clases conceptuales significativas en un dominio del problema; es el artefacto más importante que se crea durante el análisis orientado a objetos.

Un modelo del dominio es una representación de las clases conceptuales del mundo real, no de componentes software. No se trata de un conjunto de diagramas que describen clases software, u clases software con responsabilidades.²

Muestra:

- Clases conceptuales.
- Asociaciones entre las clases conceptuales.
- Atributos de las clases conceptuales.

Clases conceptuales

Una clase conceptual es una idea, cosa u objeto. Más formalmente, una clase conceptual podría considerarse en términos de sus símbolos, intención, extensión.

Símbolo: Palabras o imágenes que representan una clase conceptual.

Intención: La definición de una clase conceptual.

Extensión: El conjunto de ejemplos a los que se aplica la clase conceptual.¹

Por ejemplo, considere la clase conceptual para el evento de una transacción de compra. Podría elegir nombrarla como el símbolo venta (si se ve desde el punto de vista de la empresa, no del punto de vista del que realiza la compra). La intención de una venta podría establecer que “representa el hecho de una transacción de una compra, y tiene una fecha y una hora”. La extensión de una venta la forman todos los ejemplos de ventas, en otras palabras, el conjunto de todas las ventas.

Las clases conceptuales tienen solamente atributos. Un atributo es una característica o propiedad de una clase. Por ejemplo, si tuviéramos la clase conceptual venta sus atributos serían número de venta,

1- El proceso unificado de desarrollo de software.

2- UML y Patrones

fecha e importe total de la venta.

No hay que confundir una clase conceptual con una clase software, ya que la primera solo tiene atributos, pertenece al dominio del problema y a la etapa de análisis. La segunda tiene atributos y métodos, pertenece al dominio de la solución y a la etapa del diseño.

Estrategias para indentificar clases conceptuales

Para identificar clases conceptuales existen dos métodos:

- 1- Distinguir de un texto determinado los sustantivos y los verbos. Los sustantivos son posibles clases candidatas o atributos de las mismas. Los verbos son posibles responsabilidades. Las clases candidatas pasaran a ser definitivas cuando de acuerdo al contexto le asignemos los atributos.
- 2- El método número 1 depende de la redacción de cada individuo en particular, por eso el método número 2 es complementario y consiste en la Tabla de categorías de objetos de Larman. Todo objeto va a formar parte de una o más categorías.

Categorías de clases conceptuales según Larman

Objetos tangibles o físicos	Animal, anteojos, libro, camisa
Especificaciones, diseños o descripciones de las cosas	Ingredientes de una receta, composición de un medicamento, detalle nutricional de un producto
Lugares	Aeropuerto, puerto, terminal ferroviaria, supermercado
Transacciones	Compra, reserva, inscripción, alquiler, préstamo
Líneas de transacción	Línea de compra, línea de reserva, línea de inscripción, línea de alquiler, línea de préstamo
Roles	Piloto, supervisor, gerente, colectivo, propietario
Contenedores de otra cosa	Frasco, taza, botella, depósito, gaveta
Cosas en un contenedor	Galletitas, canicas, agua, papeles, prendas de vestir
Otros sistemas externos	Sistemas de autorización, sistemas de registro, sistemas de control
Abstracciones	Políticas, descuento, disponibilidad
Organizaciones	Cadena hotelera, restaurante, sucursal, departamento
Hechos	Compra, pago, reserva, aterrizaje, vuelo
Reglas y políticas	Política de descuento, de reintegro, promociones
Catálogos	Catálogo de piezas, de películas, de libros, de productos varios

1- El proceso unificado de desarrollo de software.

2- UML y Patrones

Como podrá apreciarse esta categorización no es taxativa, dado que una clase conceptual cualquiera puede bien ser incluido en más de una de estas categorías.

Es mejor especificar un exceso un modelo del dominio con muchas clases conceptuales de grano fino que especificar por defecto.

Nombrar y modelar cosas: El cartógrafo.

La estrategia del cartógrafo se aplica tanto a los mapas como al modelo del dominio.

Se hace un modelo del dominio con el espíritu del modo de trabajo de los cartógrafos:

- *Utilice los nombres existentes en el territorio.*
- *Excluya las características irrelevantes.*
- *No añada cosas que no están ahí.*

Un modelo de dominio es un tipo de mapa de conceptos o cosas de un dominio.

Este espíritu destaca el rol analítico de un modelo del dominio, y sugiere lo siguiente:

- *Un cartógrafo utiliza los nombres del territorio (no cambia los nombres de las ciudades en un mapa). Para un modelo del dominio, significa que utilice el vocabulario del dominio al asignar nombres a las clases conceptuales y los atributos. Por ejemplo, si estamos desarrollando un modelo para una biblioteca, nombre al cliente como socio (el termino utilizado por el personal de la biblioteca).*
- *Un cartógrafo elimina las cosas de un mapa si no se consideran relevantes para el propósito del mapa; por ejemplo, la topografía, o la población no es necesario que se muestren. Análogamente, un modelo de dominio podría excluir clases conceptuales del dominio del problema que no son pertinentes para los requerimientos. Por ejemplo, podríamos excluir bolígrafo y bolsa papel de nuestro modelo del dominio (para el conjunto de requerimientos actual) puesto que no tienen un rol relevante obvio.*
- *Un cartógrafo no muestre cosas que no estén ahí, como montañas que no existen. Igualmente, el modelo del dominio debería excluir cosas que no se encuentran en el dominio del problema que se está estudiando.²*

Añadir atributos a las clases conceptuales

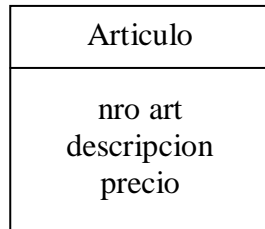
Un atributo es una característica o propiedad de una clase. Los atributos que se le asignan a una clase tienen que ver con el propósito para el cual se realiza.

Perro
raza nombre sexo

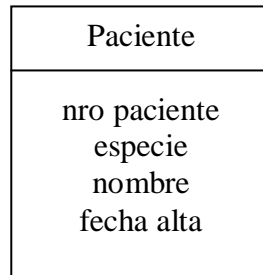
En el contexto venta en un petshop, el perro va a ser un artículo que se pone para la venta igual que comida y ropa para perros, alpiste, etc. En ese caso va a ser otra clase conceptual con distintos atributos.

1- El proceso unificado de desarrollo de software.

2- UML y Patrones



En el contexto de una veterinaria que además de perros atiende gatos, animales exóticos, etc. En ese caso es otra clase conceptual Paciente con otros atributos.



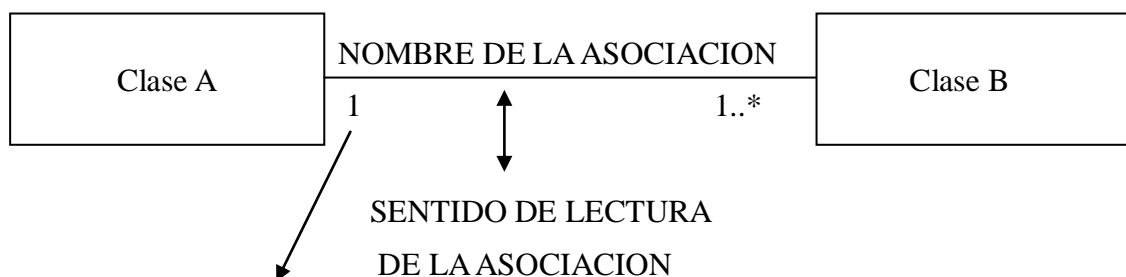
Añadir asociaciones

Una asociación es una relación semántica entre dos o más clases conceptuales que implica conexiones entre sus instancias.

Considere la inclusión de las siguientes asociaciones en un modelo de dominio:

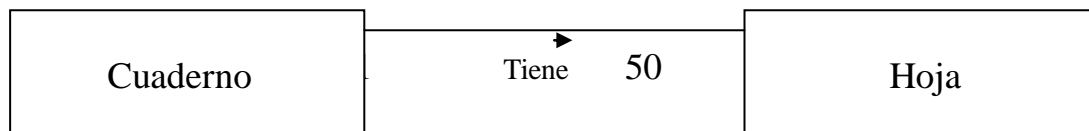
- Asociaciones de las que son necesario conservar el conocimiento de la relación durante algún tiempo (asociaciones “Necesito – conocer”).
- Asociaciones derivadas de la lista de asociaciones comunes.

Una asociación entre una clase conceptual A y una clase conceptual B se indica como una línea entre las mismas, con un nombre, un sentido de lectura de la asociación y una cardinalidad. El sentido de lectura de la asociación es de la clase conceptual A a la clase conceptual B o viceversa. La cardinalidad si el sentido de lectura es de A hacia B significa que una instancia de la clase conceptual A esta relacionada con una o muchas instancias de la clase conceptual B. Este recorrido es puramente abstracto; no se trata de una sentencia sobre conexiones entre entidades software.

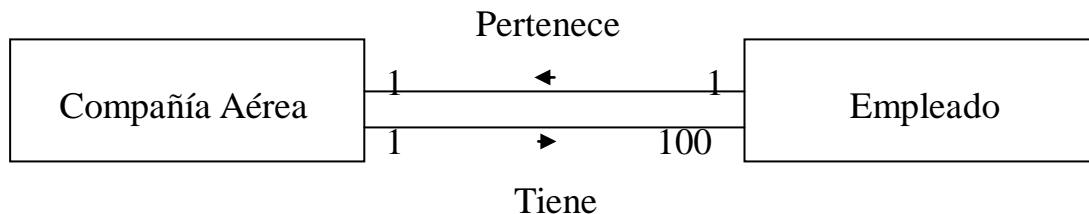


CARDINALIDAD

- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones

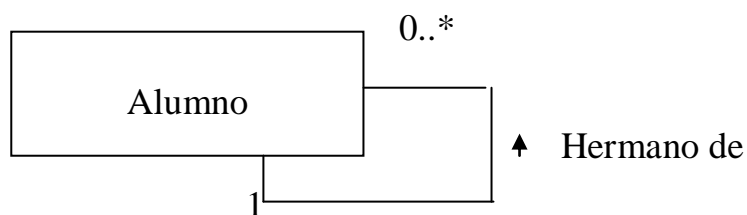


Si el sentido de lectura es de la clase conceptual Cuaderno a la clase conceptual Hoja, la cardinalidad indicaría que una instancia de Cuaderno se relaciona con 50 instancias de Hoja.



Si el sentido de lectura es de la clase conceptual Compañía Aérea a la clase conceptual Empleado la cardinalidad indicaría que una instancia de Compañía Aérea se relaciona con 100 instancias de la clase conceptual Empleado.

Si el sentido de lectura es de la clase conceptual Empleado a la clase conceptual Compañía Aérea indicaría que una instancia de Empleado se relaciona con una sola instancia de Compañía Aérea.



También puede haber relaciones entre instancias de la misma clase. Por ejemplo: Hermano de en este caso una instancia de Alumno puede estar relacionada con una o muchas instancias de de Alumno.

Lista de asociaciones comunes de Larman

Categoría de asociaciones	Ejemplos
A es una parte física de B	Cajón- Registro (o mas concretamente Terminal de puesto de venta), Ala-Avion..
A es una parte lógica de B	Linea de venta – Venta, Ejemplar- Pelicula
A esta contenido físicamente en B	Articulo – Estanteria, Pasajero – Avion
A esta contenido lógicamente en B	Articulo – Catalogo
A es una línea de una transacción o informe de B	Linea de venta – Venta, Linea de orden de compra – Compra

- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones

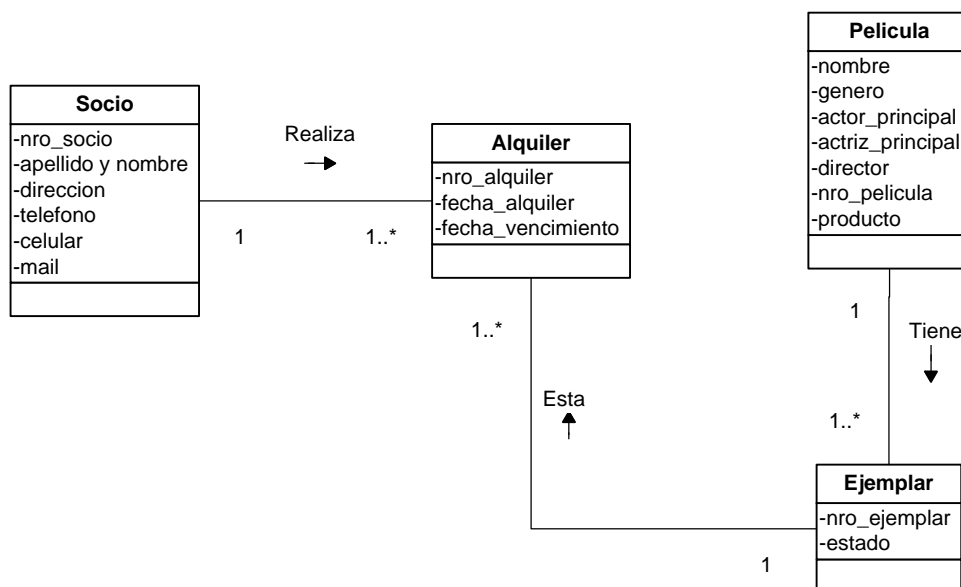
A es miembro de B	Cajero – Tienda, Piloto – Compañía aérea
A esta relacionado con una transacción B	Cliente – Pago, Pasajero – Billeto
A es un evento relacionado con B	Venta – Cliente, Venta – Tienda, Salida – Vuelo
A es una transacción relacionada con otra transacción B	Pago – Venta, Reserva – Cancelacion

Ejemplos

1 –

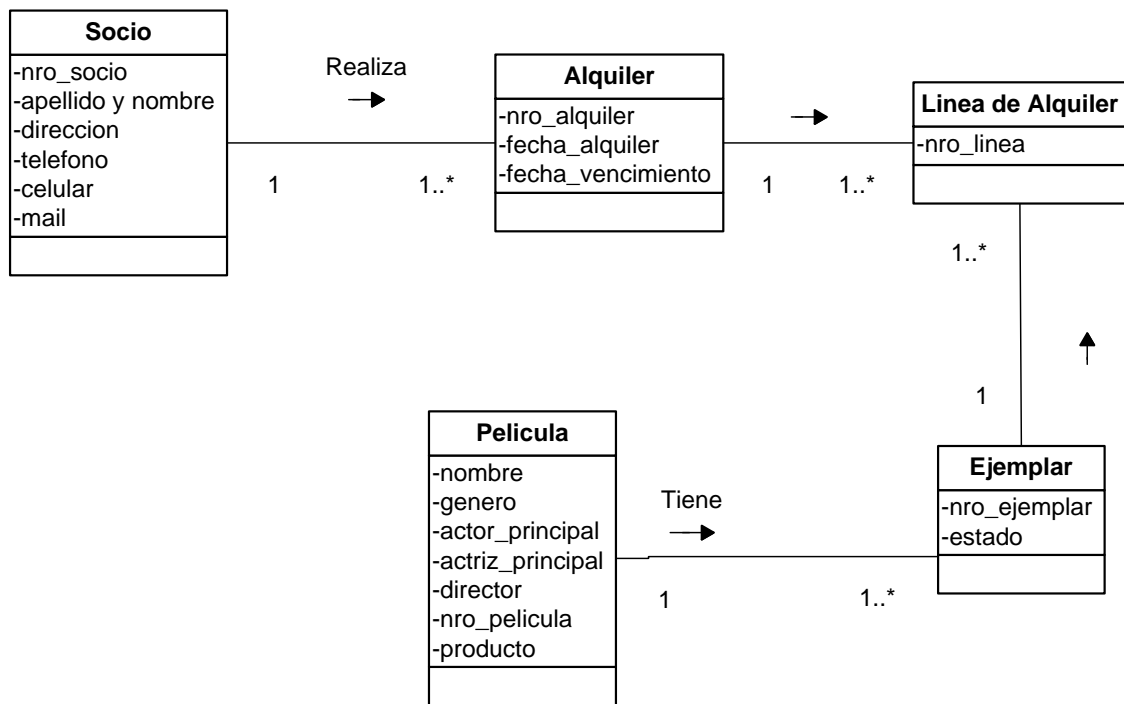
- A) En un videoclub cuando una persona quiere alquilar una película se tiene que hacer primero socio (dando sus datos personales: dni, apellido y nombre, domicilio, teléfono, celular, mail) luego le indicara al empleado de alquileres la película que desea. Se debe tener en cuenta si hay ejemplares disponibles.

Nota: se alquila una sola película.



- B) En un videoclub cuando una persona quiere alquilar una película se tiene que hacer primero socio (dando sus datos personales: dni, apellido y nombre, domicilio, teléfono, celular, mail) luego le indicara al empleado de alquileres la película que desea. Se debe tener en cuenta si hay ejemplares disponibles.

Nota: se alquila más de una película.



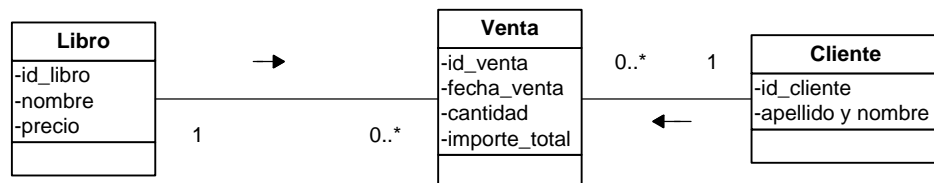
- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones

2 –

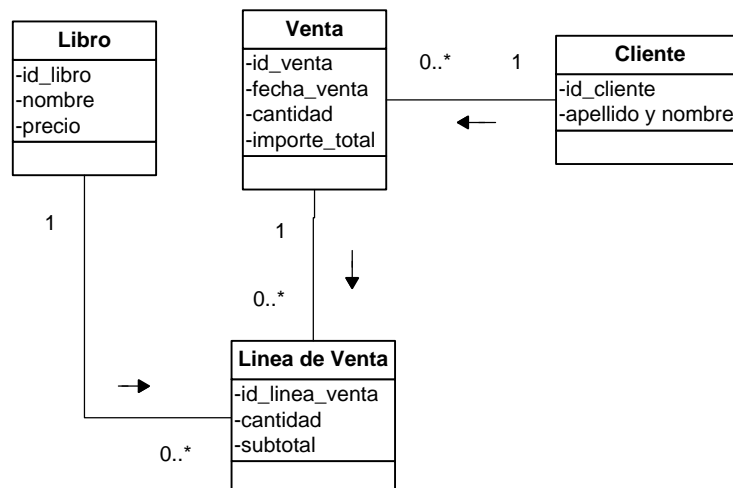
a – La librería NADIR necesita un sistema informático para registrar sus ventas.



b – La librería NADIR necesita un sistema informático para registrar la venta a sus clientes.



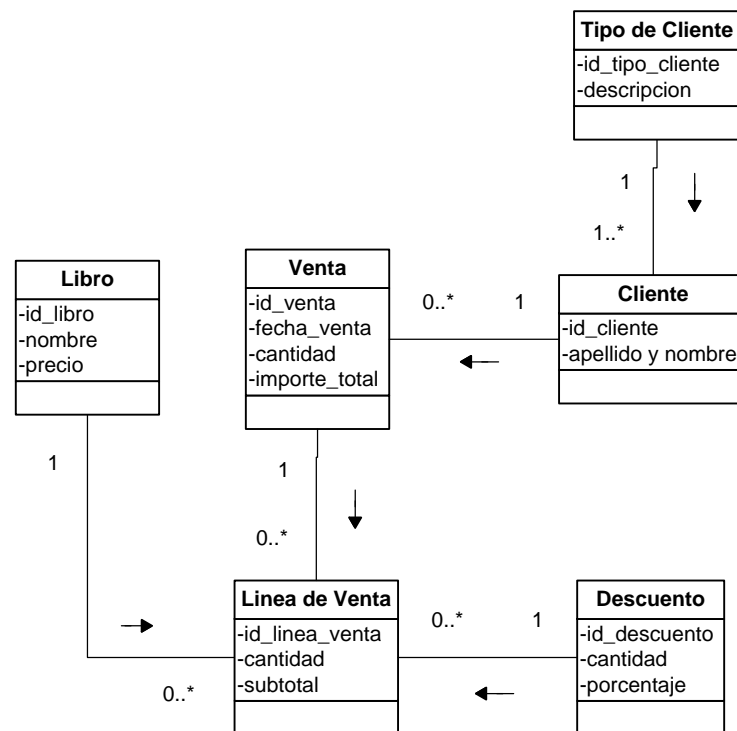
c – La librería NADIR necesita un sistema informático para registrar la venta a sus clientes. Si el cliente compra más de un libro se le aplica un descuento del 10%.



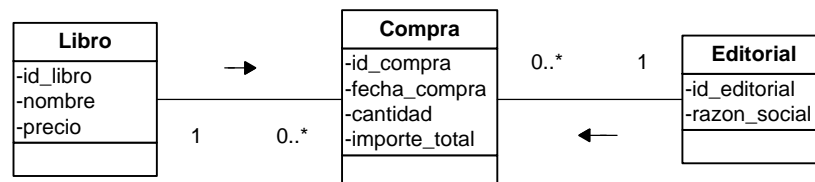
1- El proceso unificado de desarrollo de software.

2- UML y Patrones

d – La librería NADIR necesita un sistema informático para registrar la venta a sus clientes. Si el cliente es revendedor, el descuento será en función de la cantidad de libros que compre.

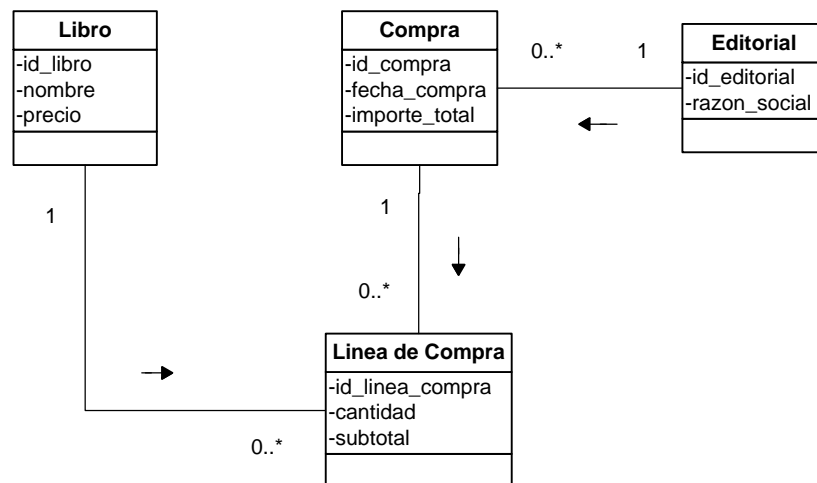


e – La librería NADIR necesita un ampliar su sistema informático agregando la compra de libros a proveedores (proveedor = editorial). Cada libro es provisto por una única editorial, los pedidos son por título. Cada orden de compra está compuesta por un único libro.

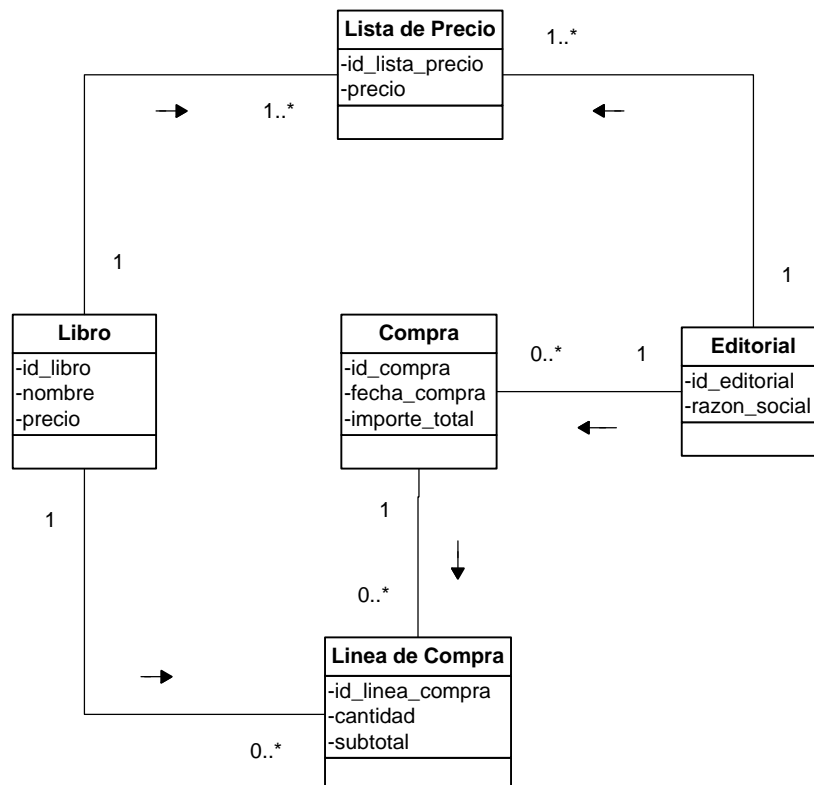


- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones

f – La librería NADIR necesita un ampliar su sistema informático agregando la compra de libros a proveedores (proveedor = editorial). Cada orden de compra está compuesta por más de un libro.



g – La librería NADIR necesita un ampliar su sistema informático agregando la compra de libros a proveedores (proveedor = editorial). Cada orden de compra está compuesta por más de un libro y un titulo puede ser provisto por más de una editorial.

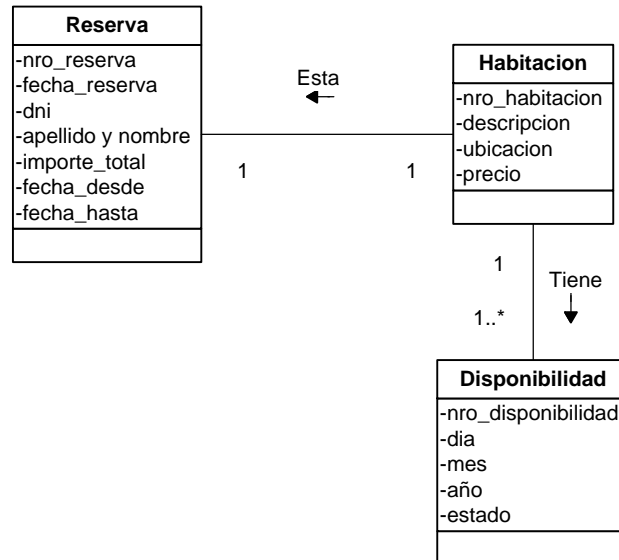


- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones

3 -

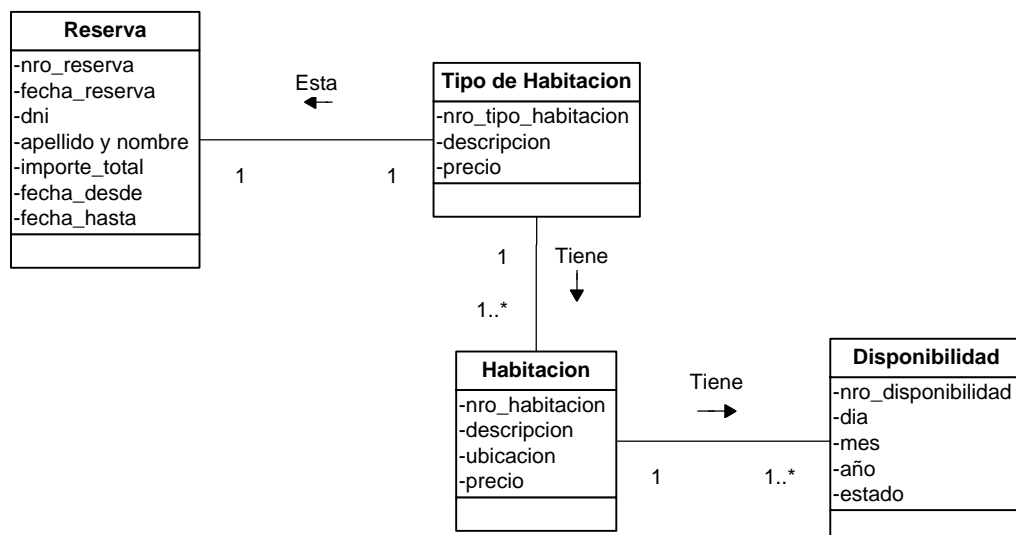
- a. El hotel Hostal del Sol tiene un solo tipo de habitación. Cuando una persona quiere reservar una habitación debe hacerlo telefónicamente indicando el periodo para el cual quiere reservar.

Nota: puede reservar solo una habitación y la misma tiene un precio fijo todo el año. Puede o no haber disponibilidad de la misma para el periodo seleccionado.



- b. El hotel Hostal del Sol tiene distintos tipos de habitación (doble estándar, doble superior, triple o cuádruple todas con vista al mar o al bosque), cada una con diferentes precios durante todo el año. Cuando una persona quiere reservar una habitación debe hacerlo telefónicamente indicando el periodo para el cual quiere reservar y el tipo de habitación que desea.

Nota: puede reservar solo una habitación y la misma tiene un precio fijo todo el año. Puede o no haber disponibilidad de la misma para el periodo seleccionado.

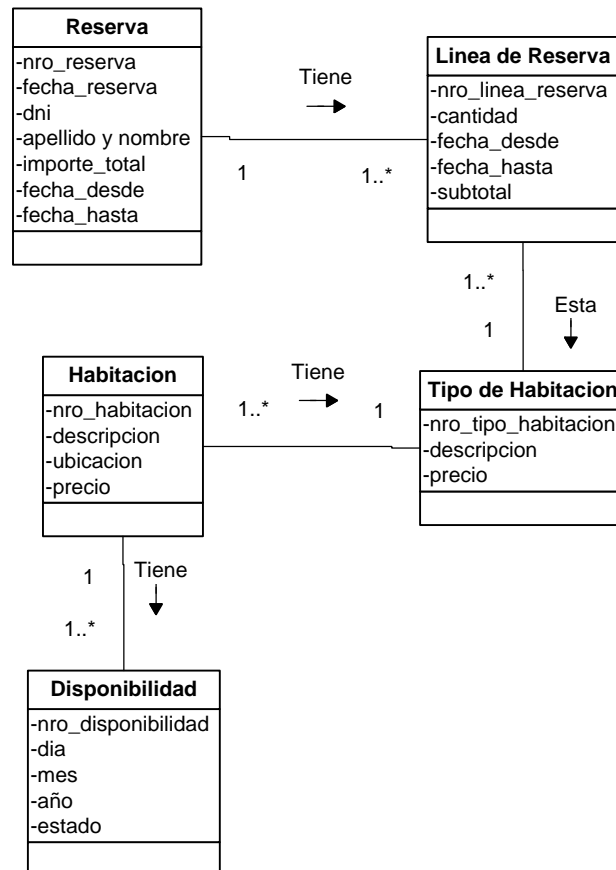


1- El proceso unificado de desarrollo de software.

2- UML y Patrones

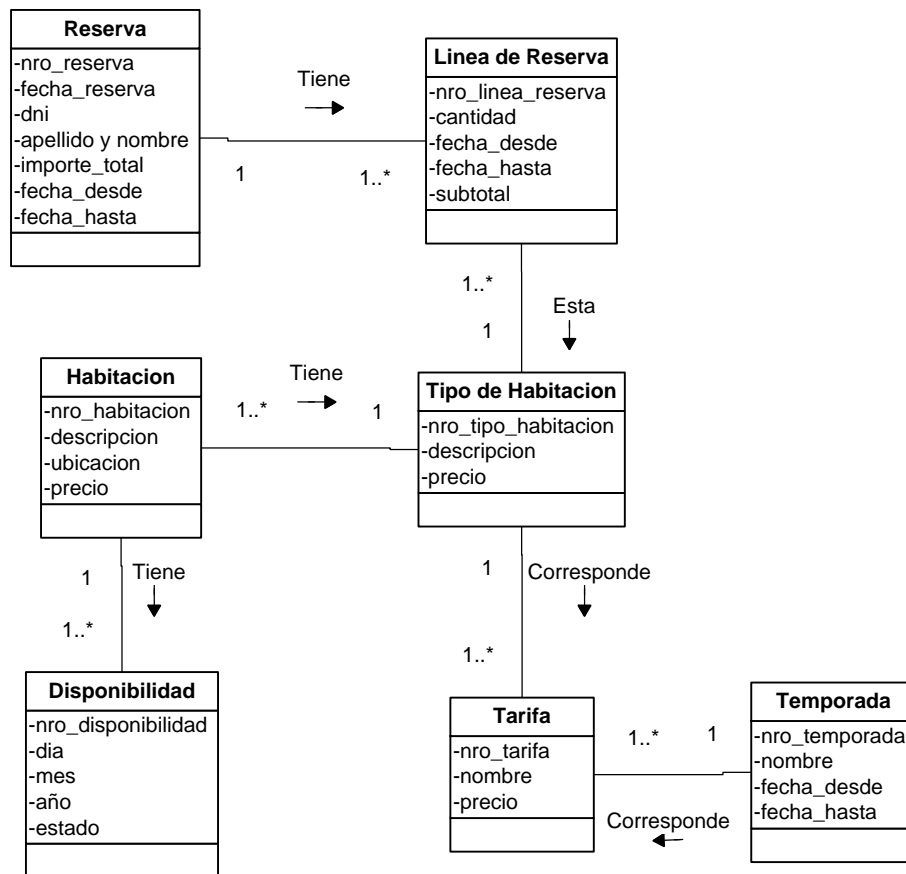
- c. El hotel Hostal del Sol tiene distintos tipos de habitación (doble estándar, doble superior, triple o cuádruple todas con vista al mar o al bosque), cada una con diferentes precios durante todo el año. Cuando una persona quiere reservar una habitación debe hacerlo telefónicamente indicando el periodo para el cual quiere reservar y el tipo de habitación que desea.

Nota: puede reservar más una habitación, del mismo o diferente tipo. Puede o no haber disponibilidad de la/las misma/s para el periodo seleccionado.



- d. El hotel Hostal del Sol tiene distintos tipos de habitación (doble estándar, doble superior, triple o cuádruple todas con vista al mar o al bosque). La tarifa del tipo de habitación depende de la temporada (baja, media o alta). Cuando una persona quiere reservar una habitación debe hacerlo telefónicamente indicando el periodo para el cual quiere reservar y el tipo de habitación que desea.

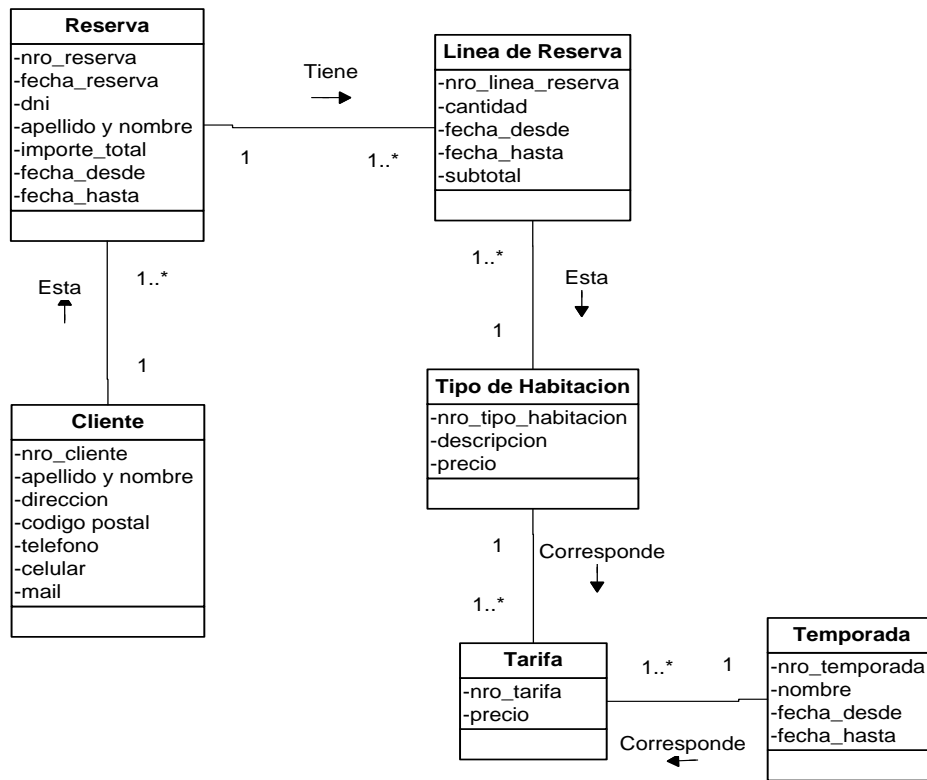
Nota: puede reservar más una habitación, del mismo o diferente tipo. Puede o no haber disponibilidad de la/las misma/s para el periodo seleccionado.



- e. Una cadena hotelera tiene posee varios hoteles cada uno con distintos tipos de habitación. La tarifa del tipo de habitación depende de la temporada (baja, media o alta). Cuando una persona quiere reservar una habitación debe vía web indicando el hotel, el periodo y el tipo de habitación para el cual quiere reservar.

Nota: puede reservas solo en un hotel, más una habitación, del mismo o diferente tipo. Puede o no haber disponibilidad de la/las misma/s para el periodo seleccionado.

- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones



- 1- El proceso unificado de desarrollo de software.
- 2- UML y Patrones

Bibliografia

El proceso unificado de desarrollo de software –

Autores: Ivar Jacobson, Grady Booch, James Rumbaugh

Editorial: Addison Wesley

1era. Edicion – Año: 1999

UML y Patrones

Autor: Craig Larman

Editorial: Prentice Hall

2da. Edicion – Año: 2003

1- El proceso unificado de desarrollo de software.

2- UML y Patrones