Proyecto Certificador de Desarrollo de SW 1

Tema Nº10:

Convertir el Análisis al Diseño

Indicador de logro Nº10:Desarrolla el Modelo de Análisis y Diseño, de acuerdo a metodología RUP - lenguaje UML

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº10:**

CONVERTIR EL ANÁLISIS AL DISEÑO

Objeto De La Experiencia

* Conoce la disciplina, documenta y explica la actividad del análisis y diseño respecto a la metodología RUP, como también define, los artefactos y los roles más importantes.
* Elabora los diagramas de Diseño.

Marco Teórico

El objetivo del diseño es entender la solución refinando el modelo de análisis con la intención de desarrollar un modelo de diseño que permita una transición sin problemas a la fase de construcción. En el diseño, nos adaptamos al entorno de implementación y despliegue.

Diferencias Entre Análisis Y Diseño

|  |  |
| --- | --- |
| **MODELO DE ANÁLISIS** | **MODELO DE DISEÑO** |
| Es Conceptual. Abstrae el sistema sin el propósito de implementarlo | Es físico. Es el anteproyecto de la implementación |
| Es genérico y válido para muchos diseños | Es específico para la implementación |
| Con Templa tres estereotipos: Boundary, Control y Entity | Diseña muchos estereotipos, dependiendo el lenguaje de programación. |
| Define pocas capas | Puede ser multicapas |

MODELO CONCEPTUAL

Las clases del modelo conceptual se obtienen a partir de los objetos de información que fluyen entre las actividades. Una característica importante que resaltar es que el modelado de los casos de uso del sistema y el modelado conceptual se realizan en paralelo, esto es crucial para obtener casos de uso correctos, puesto que es necesario entender bien el dominio para poder escribir casos de uso que sean realmente útiles.

Importancia del Modelo Conceptual

El Modelo Conceptual Orientado a Objetos beneficiará a dos equipos de trabajo:

Equipo de Desarrolladores

En esta etapa del desarrollo, es conveniente detenerse en la identificación de los conceptos y no tanto en las relaciones entre ellos.

Este modelo incluirá los conceptos y sus relaciones y se describirá mediante un diagrama de clases UML, en el que los conceptos se representan mediante clases (del dominio).

Equipo de Base de Datos

En esta etapa, luego del modelo conceptual, se obtiene el proceso del modelo lógico al diseño físico donde se podrá identificar las tablas relacionales del proyecto de Base de Datos como componente RUP.

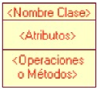
Construcción del Modelo Conceptual

Los pasos que se realizan para la construcción del modelo conceptual son:

1. Identificar clases de entidad con sus atributos
2. Identificar asociaciones y multiplicidades
3. Identificar agregaciones y multiplicidades
4. Definir jerarquías(opcional)
5. Identificar las clases con sus atributos

La Clase es la unidad básica que encapsula toda la información de un objeto (un objeto es una instancia de una clase). A través de ella, podemos modelar el entorno en estudio (una Cuenta Corriente, etc.).

* Los atributos representan las propiedades de la clase que se encuentran en todas las instancias. Definen la estructura de una clase estructura de una clase y de sus objetos.
* Los atributos corresponden a sustantivos y sus valores pueden ser sustantivos o adjetivos.
* Dentro de una clase, los nombres de los atributos deben ser únicos (aunque puede aparecer el mismo nombre de atributo en diferentes clases).
* Los atributos pueden representarse solo mostrando su nombre su tipo e, incluso su valor por defecto
* **Public:** Indica que el atributo será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados
* **Prívate:** indica que el atributo solo será visible desde dentro de la clase (solo sus métodos los pueden acceder).
* **Protected:** Indica que el atributo no será accesible desde fuera de la clase, pero si podrá estar disponible para métodos de la clase además de las subclases que se deriven.



* Para los Identificadores, en el momento de incluir atributos en la descripción de una clase se debe distinguir entre los atributos que reflejan las características de los objetos en el mundo real y los identificadores que son utilizados por razones de implementación.
* Guías prácticas para la definición de Clases y Atributos
* Las clases poseen información descriptiva; los atributos, no.
* Los atributos multivaluados deben ser clasificados como clases.
* Convertir, en una clase, a un atributo que tenga una relación muchos a uno con otra clase.
* Asociar atributos a las clases que ellos describen más directamente. Los atributos deben ser inherentes a la clase.
* Evitar los identificadores compuestos en la medida que sea posible.
* Existen algunas categorías de clases que podríamos utilizar para identificarlas correctamente.

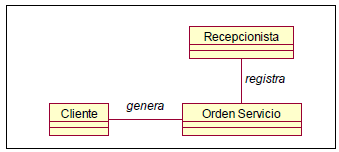
|  |  |
| --- | --- |
| **CATEGORÍA** | **EJEMPLO** |
| Tangibles o físicos | Edificio, Producto |
| Especificaciones o descripciones | Especificación Producto,  Descripción Vuelo |
| Lugares | Tienda, Aula, Laboratorio |
| Transacciones | Venta, Pago, Reserva |
| Líneas o detalle de transacción | LineaVenta, DetalleReserva |

1. Identificar asociaciones y multiplicidades

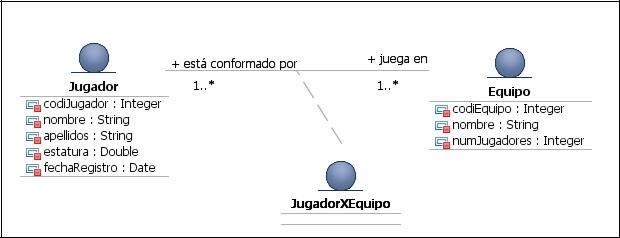
* La asociación es una relación entre clases que indica una conexión significativa.
* Está representada como una línea entre clases con nombre. La asociación es inherentemente bidireccional.
* Es convencional leer la asociación de izquierda a derecha o de arriba hacia abajo.
* Pueden existir múltiples asociaciones entre dos clases.
* Las asociaciones pueden ser binarias, ternarias, o de mayor grado.

TIPOS DE ASOCIACIONES

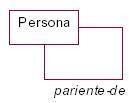
Asociación Binaria: Asociación entre dos clases.



Asociación de clase: Asociación entre dos clases que contiene otra entidad. Generalmente, este tipo de asociación se utiliza para representar una relación de muchos a muchos entre dos clases.

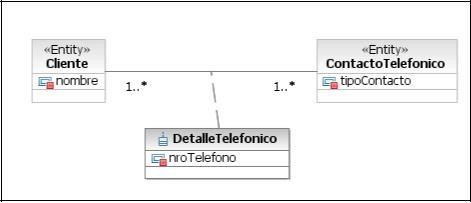


Asociación Reflexiva: Se da en la misma clase.



Clase Asociativa

* Se da cuando uno o más atributos están relacionados con la asociación.
* Las instancias de la clase asociativa dependen del tiempo de vida de la asociación.
* Se da en una asociación de muchos a muchos entre dos clases y existe información asociada con la propia relación de asociación.

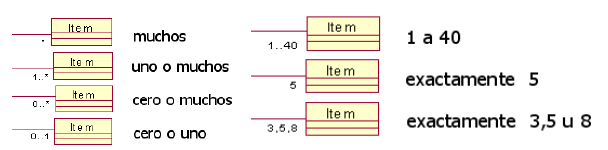


Roles

* Dada una asociación entre dos entidades, decimos que cada entidad representa un rol en dicha asociación.
* Muchas veces, según el punto de vista de cada entidad, es posible nombrar a la asociación de manera diferente.

Multiplicidad

* Restringe el número de objetos de una clase que se pueden implicar en una relación determinada en cualquier momento en el tiempo. La frase “en cualquier momento en el tiempo” es vital para entender las multiplicidades.
* Define cuántas instancias de la clase A pueden estar asociadas con una instancia de la clase B.
* La multiplicidad presenta las relaciones con valores de datos de acuerdo al detalle siguiente:

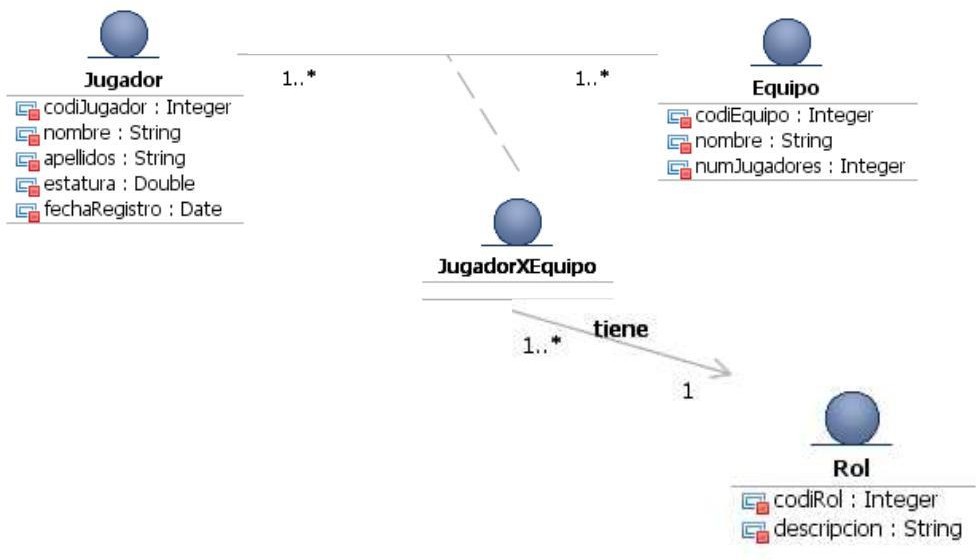


**En el siguiente ejemplo, se representa las siguientes relaciones:**

- Un jugador “juega en” muchos equipos

- Un equipo “está conformado por” varios jugadores.

- Cada jugador, dependiendo del equipo en que se encuentre tendrá un rol diferente.



+juega en

+está conformado por

•

•

•

•

•

•

1. Identificar las agregaciones y multiplicidades

* La agregación indica una relación de “un todo conformado por partes” • Existen 2 tipos de agregaciones:

Agregación Débil o Compartida

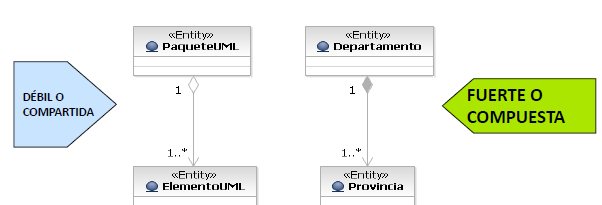
Agregación Fuerte o compuesta

• La agregación representa

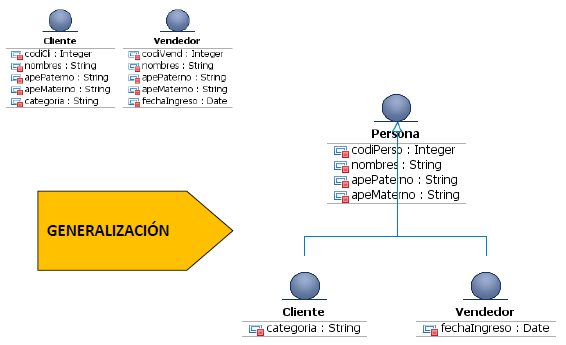
• En UML se proporciona una escasa caracterización de la agregación.

• Puede ser caracterizada con precisión determinando las relaciones de comportamiento y

estructura que existen entre el objeto



1. Generalización: Consiste en identificar todos aquellos atributos iguales de un conjunto de entidades para formar una entidad global con dichos atributos semejantes, dicha entidad global quedará a un nivel más alto al de las entidades origen.



1. La especialización es el resultado de tomar un subconjunto de entidades de alto nivel para formar un conjunto de entidades de más bajo nivel.

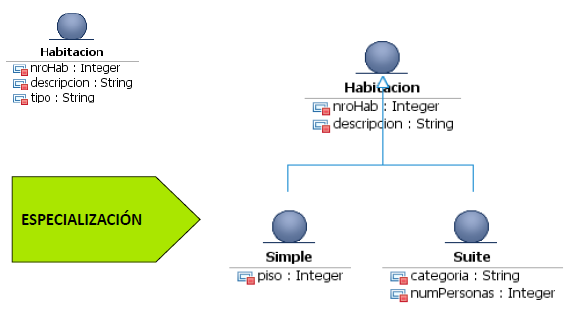
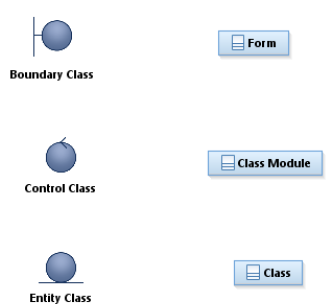


DIAGRAMA DE CLASES DE DISEÑO

Al igual que el diagrama de análisis muestra las interrelaciones entre sus elementos, Equivalencia de los estereotipos:

Análisis y Diseño



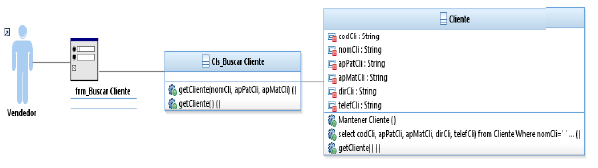
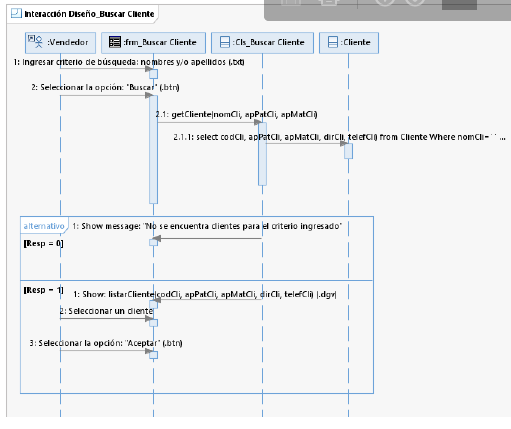
**EJEMPLO**

Diagrama De Secuencia De Diseño

Describe el comportamiento de los casos de uso en función del tiempo. Su uso respecto a UML no cambia. En este diagrama se escriben las líneas de tiempo de los actores y componentes implicados. Los actores y componentes se envían mensajes entre ellos describiendo el comportamiento del sistema.

**EJEMPLO.**



**Actividad:**

Ingresa a la plataforma virtual, luego desarrolla la siguiente actividad propuesta:

1. **CUESTIONARIO TÉCNICO**
2. Tomando como ejemplo la Guiía\_09, Del Diagrama de Diseño, elabore el Diagrama del Modelo de Diseño de su proyecto en grupo a desarrollar. Deberá utilizar metodología RUP y además IBM Rational Software Architect Designer V9.6
3. Diseñe los prototipos de los formularios de su proyecto utilizando Ms Visual Studio. Net, C#.
4. Elaborar el Diagrama de Secuencias de Diseño de su proyecto en grupo a desarrollar. Deberá utilizar metodología RUP y además IBM Rational Software Architect Designer V9.6
5. **CONCLUSIONES DE LA EXPERIENCIA**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_