Proyecto Certificador de Desarrollo de SW 1

Tema Nº11:

Modelo de Datos

Indicador de logro Nº11:

Conoce el modelo de datos de un sistema de información para construir el modelo lógico y físico de datos

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº11:**

MODELO DE DATOS

Objeto De La Experiencia

* Elabora el Modelo de Datos de acuerdo a UML
* Elabora el Modelo Lógico y Físico

Concepto

El modelado de datos se utiliza habitualmente en combinación con un sistema de gestión de base de datos. Los datos que se han modelado y preparado para este sistema se pueden identificar de varias maneras, como de acuerdo a lo que representan, o cómo se relacionan con otros datos

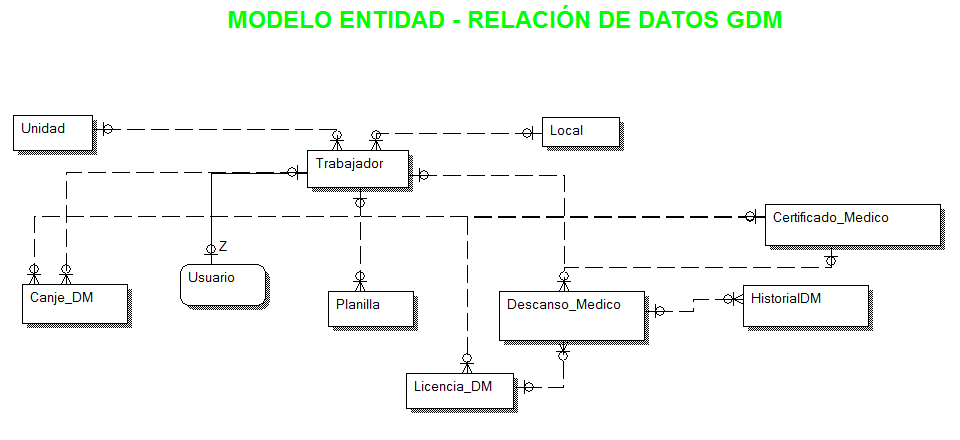
Típicamente un modelo de datos permite describir:

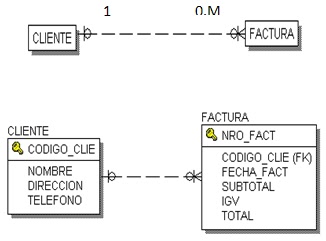
* Las estructuras de datos de la base: El tipo de los datos que hay en la base y la forma en que se relacionan.
* Las restricciones de integridad: Un conjunto de condiciones que deben cumplir los datos para reflejar la realidad deseada.
* Operaciones de manipulación de los datos: típicamente, operaciones de agregado, borrado, modificación y recuperación de los datos de la base.

Modelo Entidad – Relación

Un diagrama Entidad - Relación, también conocido como modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra cómo las "entidades", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema.

Los diagramas ER se componen de entidades, relaciones y atributos. También representan la cardinalidad, que define las relaciones en términos de números





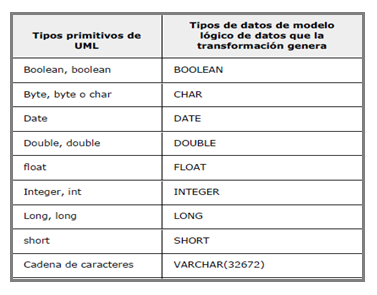
Modelo Lógico

Un modelo lógico de datos es un modelo que no es específico de una base de datos que describe aspectos relacionados con las necesidades de una organización para recopilar datos y las relaciones entre estos aspectos. El modelo lógico de datos es el refinamiento del modelo conceptual. En este modelo no es necesario especificar las llaves primarias y foráneas de las entidades, pues es trabajo que se recomienda realizar en el modelo físico.

Tipos de datos

Existen muchas herramientas que permiten realizar transformaciones de un modelo lógico a partir de un modelo conceptual (con notación UML). La transformación UML a modelo lógico de datos genera tipos de datos de modelo lógico de datos a partir de los tipos primitivos de UML.

En la siguiente tabla, se muestra la correspondencia entre tipos primitivos de UML y tipos de datos de modelo lógico de datos:

****

Tipos de Relaciones

Los tipos de relaciones que se pueden crear para un modelo lógico son los siguientes:

Relación Identificada

Las relaciones identificadas migran la llave primaria de la entidad padre a la llave primaria de la entidad hija. Su representación es una línea nítida, tal como se muestra en la siguiente figura:

****

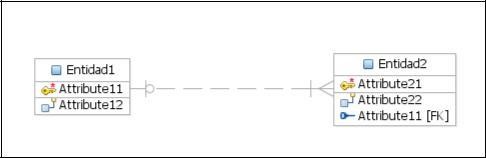
Relación No Identificada

En una relación no identificada la llave primaria de la entidad padre es migrada como un atributo más de la entidad hija, es decir no formará parte de su llave primaria

Además, en este tipo de relación se representa otra característica:

La existencia, quien describe la relación entre un par de entidades desde la perspectiva de la entidad hija. Fundamentalmente haciendo la pregunta, ¿Es el valor de una llave foránea siempre requerida en la entidad hija? Las posibles respuestas son las siguientes:

1. Opcional: Cuando el valor de una llave foránea no es siempre requerido en la entidad hija. Sin embargo, si un valor existe, el valor de la llave foránea debe encontrarse en la llave primaria de la entidad padre. Se representa con una línea entrecortada y en el extremo de la entidad padre aparece una línea con círculo:



*Entidad Padre*

*Entidad Hija*

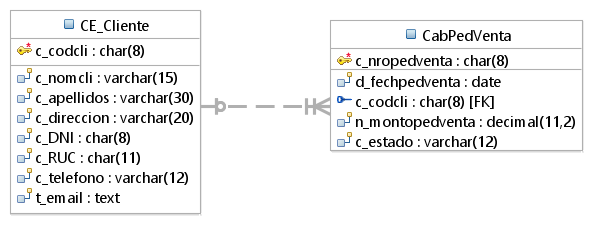
1. Obligatoria. Cuando el valor de una llave foránea debe existir en la entidad hija y el valor de la llave foránea debe encontrarse en la llave primaria de la entidad padre. Se representa con una línea entrecortada y en el extremo de la entidad padre aparece una línea:



*Entidad Padre*

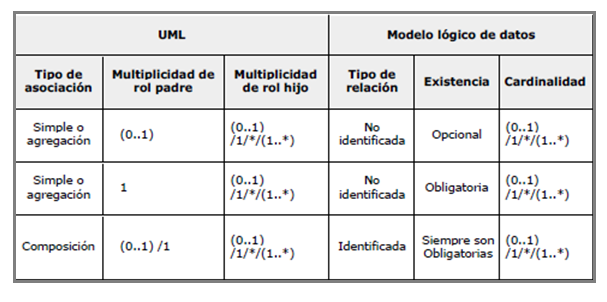
*Entidad Hija*

**Ejemplo de Relación No Identificada**

****

Un proceso de transformación del modelo conceptual (con notación UML) a modelo lógico de datos, genera tipos de relación, existencia y cardinalidad para el modelo lógico de datos de acuerdo a los tipos de asociación y multiplicidad de rol UML.

En la siguiente tabla, se muestran las correlaciones entre un tipo de asociación o multiplicidad de rol UML y el tipo de relación, existencia y cardinalidad de un modelo lógico de datos que se crean en un proceso de transformación.

****

Modelo Físico

El modelo físico es un modelo de datos de bajo nivel. Proporcionan conceptos que describen los detalles de cómo se almacenan los datos en el ordenador.

El paso de un modelo lógico a uno físico requiere un profundo entendimiento del manejador de bases de datos que se desea emplear, incluyendo características como:

* Conocimiento a fondo de los tipos de objetos (elementos) soportados
* Detalles acerca del indexamiento, integridad referencial, restricciones, tipos de datos, etc
* Detalles y variaciones de las versiones
* Parámetros de configuración
* Data Definition Language (DDL)
* El paso de convertir el modelo lógico de datos a tablas hace que las entidades pasen a ser tablas (más las derivadas de las relaciones) y los atributos se convierten en las columnas de dichas tablas.

Las relaciones entre las tablas del modelo físico son las mismas que se describieron en el modelo lógico:

* Relación Identificada
* Relación No Identificada Opcional
* Relación No Identificada Obligatoria

Campos

1. Tipos de Datos

Los tipos de datos son según los tipos de datos disponibles en el sistema manteador de base de datos (SMBD). En la mayoría de casos, se considera lo siguiente:

* Número de dígitos en números enteros
* La precisión de los flotantes
* Cadenas de caracteres de longitud fija. Ejemplo: CHAR (8)
* Cadenas de caracteres de longitud variable. Ejemplo: VARCHAR (20)
* Para archivos binarios (imágenes) se utiliza el tipo BLOB (Binary large objects)
* Para cadenas de longitudes grandes se utiliza el tipo CLOB (Character large objects)

1. Llaves primarias

Representa el identificador de una tabla y está formada por uno o más campos de la tabla.

Algunos SMBD poseen la capacidad de "autoincrement" o "identity property" con la cual pueden automáticamente manipular algún atributo para generar llaves primarias de forma incremental. Pero es importante verificar:

* Cómo se manejan internamente
* Si se pueden reiniciar
* Si se permite especificar algún valor inicial

1. Orden de los campos o columnas

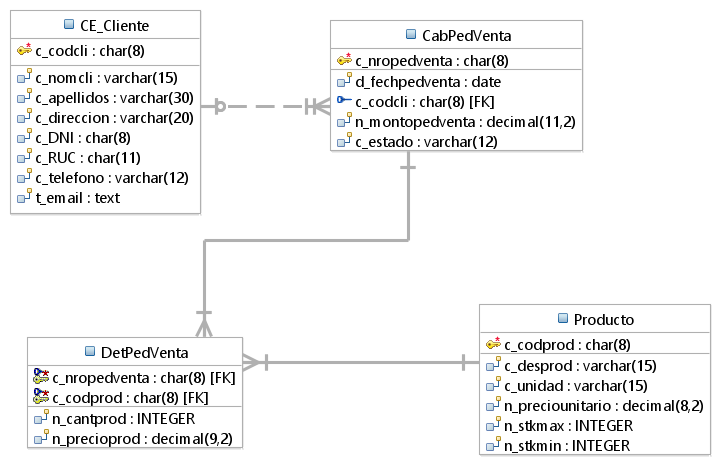
Por lo general la secuencia es la siguiente

* Columnas de longitud fija que no se actualizan frecuentemente.
* Aquellas que nunca se actualizan y por lo general, tendrán longitud variable.
* Las que se actualizan frecuentemente.

1. Integridad Referencial

* En la medida de lo posible, indicar las columnas que brindan o sirven de vínculo entre 2 tablas.
* El administrador de base de datos puede hacerse cargo de esto, pero es mejor que el SMBD lo haga.
* No se recomienda en ambientes de desarrollo.

**Ejemplo: Modelo Físico**



Índices

Un índice es un atajo desde un campo llave hacia la localización real de los datos. Es el punto clave de la optimización de velocidad de toda base de datos.

Si se busca algún registro entonces, se realiza un escaneo de la tabla completa lo cual es demasiado costoso, por eso es recomendable usar índices en:

* Llaves primarias
* Llaves foráneas
* Índices de acceso
* Ordenamiento

No olvidar que el uso de un índice implica:

* Overhead, debido a la actualización de los mismos
* Espacio adicional en disco
* Procesos batch de muchos datos pueden volverse demasiados lentos.
* Manipulación de archivos adicionales por el sistema operativo

MIGRACIÓN DE DATOS

Descripción

Cada uno de los elementos del modelo lógico se tiene que transformar en un elemento del modelo físico. En algunos casos la transformación es directa porque el concepto se soporta igual en ambos modelos, pero otras veces no existe esta correspondencia, por lo que es necesario buscar una transformación que conserve lo mejor posible la semántica, teniendo en cuenta los aspectos de eficiencia que sean necesarios en cada caso

Transformación de Entidades

Una entidad se transforma en una tabla.

Transformación de atributos de entidades

Cada atributo se transforma en una columna de la tabla en la que se transformó́ la entidad a la que pertenece. El identificador único se convierte en clave primaria.

Si existen restricciones asociadas a los atributos, estás pueden recogerse con algunas cláusulas del lenguaje lógico, que se convertirán en disparadores cuando estos sean soportados por el sistema gestor de base de datos.

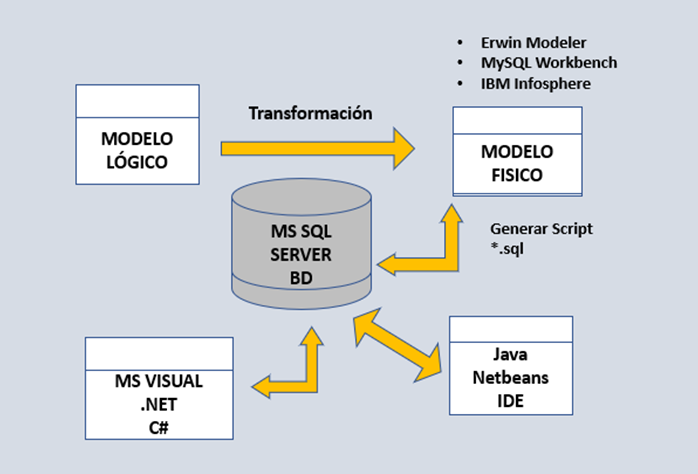
Transformación de relaciones

Según el tipo de correspondencia

Para el proyecto solución de escritorio, el modelo lógico y físico se puede diseñar en Erwin Modeler o Mysql Workbench, otro software libre, luego se deberá crear la Base de Datos en MS SQL SERVER, después se migrará el Modelo Físico a la Base de Datos creada en el motor de Base de Datos del MSQL SERVER.

También es posible generar el script de la base de datos y luego ir al Administrador de Base de Datos y ejecutar el Script, actualizando las tablas.

MIGRACIÓN DEL MODELO FÍSICO AL MOTOR DE BASE DE DATOS MS SQL SERVER

****

Caso Práctico Nro. 3

La empresa INFOTEL BUSINESS, es una empresa que se dedica a la comercialización y distribución de productos informáticos.

Proceso De Ventas

Un Cliente solicita un pedido de venta a un Vendedor. El Vendedor antes de Generar el Pedido de Venta, Consulta el Stock de Productos y el Estado del Cliente.

Si todo está correcto, el Vendedor Genera el Pedido de Venta. Luego el Cajero Consulta el Pedido de Venta para Generar el Documento de Venta previa Consulta de la Forma de Pago y del Tipo de Documento de Venta.

El Tipo de Documento de Venta puede ser Factura de Venta o Boleta de Venta (Ticket electrónico). La Forma de Pago, puede ser Contado o Crédito. Después el Cajero Genera el Documento de Venta.

Se pide lo siguiente:

1.0 Identificar las clases del modelo del proceso de ventas

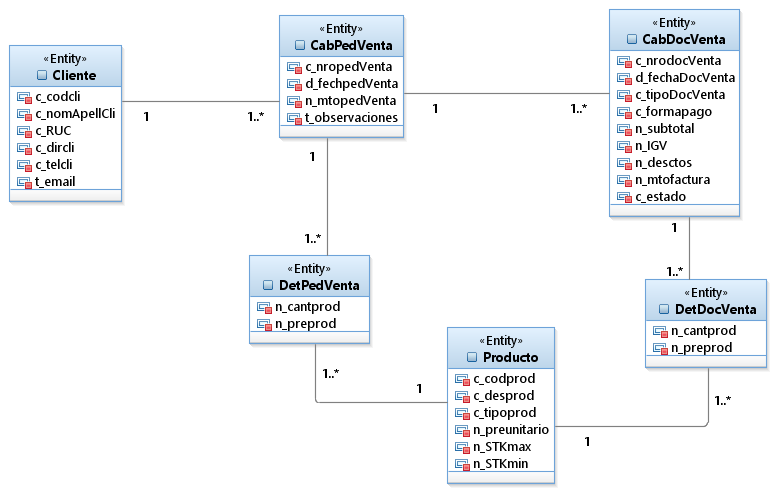
2.0 Elaborar el Modelo Lógico y Modelo Físico de Datos del Procesos de Ventas.

Solución

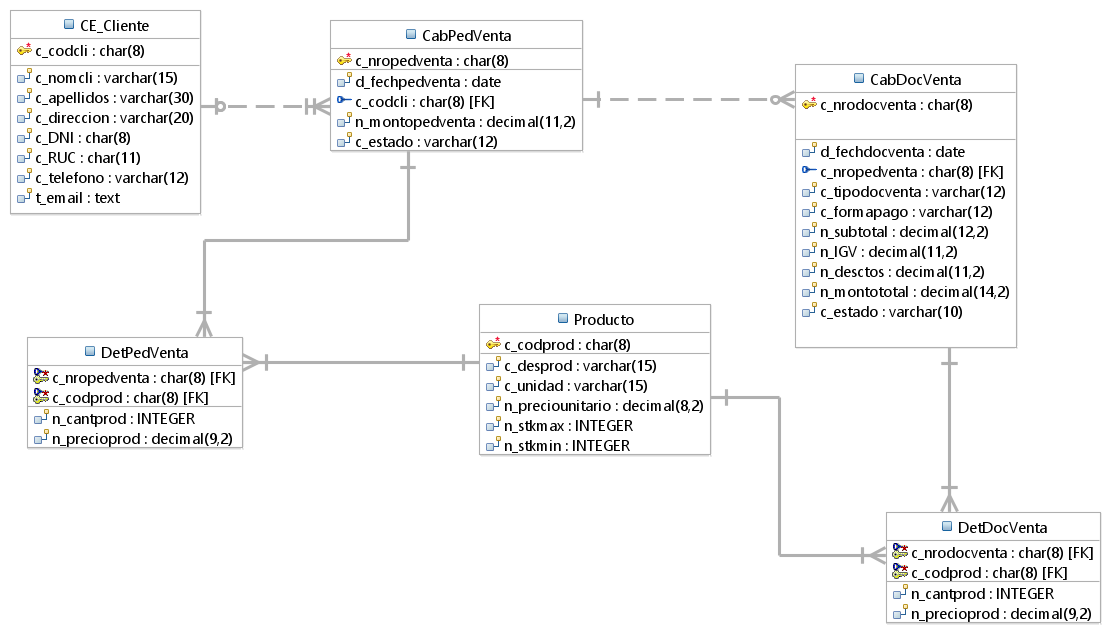
Identificación de Clases: Las clases deberán estar normalizadas en sus diferentes formas normales

Cliente. CabpedVenta. DetPedVta. CabDocVenta. DetDocVentas

Modelo Lógico



Modelo Físico



**Actividad:**

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

1. **CUESTIONARIO TÉCNICO**
2. Tomando como ejemplo la Guiía\_10, Identifique las clases del Modelo de Datos de su proyecto en grupo a desarrollar, completamente normalizadas. Utilizar metodología RUP y además Erwin Modeler o Mysql Workbench
3. Elabore el Diagrama del Modelo Lógico de Datos de su proyecto en grupo a desarrollar. Deberá utilizar metodología RUP y además Erwin Modeler o Mysql Workbench, otro software libre.
4. Elaborar el Diagrama del Modelo Físico de Datos. Deberá utilizar metodología RUP y además Erwin Modeler o Mysql Workbench, otro software libre.
5. **CONCLUSIONES DE LA EXPERIENCIA**

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