Cuando se habla del enfoque empresarial, se involucran

1. Capa Organizacional
2. Capa de Procedimientos
3. Capa de Datos
4. Capa de Aplicaciones

La capa organizacional, se representa para las empresas a través de los organigramas o estructuras organizaciones.

Los organigramas están compuestos por áreas o dependencias.

Las áreas o dependencias tienen como responsabilidad ejecutar una o varias misiones las cuales estarán debida mente asignadas por la alta gerencia, esto hace pensar, que difícilmente un ingeniero de sistemas, encargado de construir sistemas de información, pueda atreverse a cambiar las misiones.

Las misiones, involucran los escenarios sobre los cuales se deberá hacer el control, para cumplir con sus propósitos, estos escenarios se conocen dentro de la ingeniería de sistemas como clases.

Las áreas o dependencias de una empresa en cuestión Pueden tener a cargo una misión o varias misiones. Las misiones deberán albergar como mínimo dos clases. Y pueden tener las siguientes arquitecturas:

1. Mono misional Simple: Hace referencia, a una sola misión con un mínimo de dos clases, articuladas, en la misma línea de control.
2. Mono misional Extendido: Hace referencia a un modelo de lógica de negocio, compuesto por mas de dos clases, todas articuladas en la misma línea de control.
3. Multi misional Simple: Es un modelo de lógica de negocio, con dos líneas de control, o más, diferentes pero en cada línea de control, aparecen como mínimo 2 clases.
4. Multi misional Extendido: Es un modelo de lógica de negocio, con dos o más líneas de control diferentes y en donde cada línea de control tiene mas de dos clases.

Taller de comprensión

Dibuje un organigrama que tenga el siguiente árbol:

Gerencia

Personal – Ventas (Facturación)

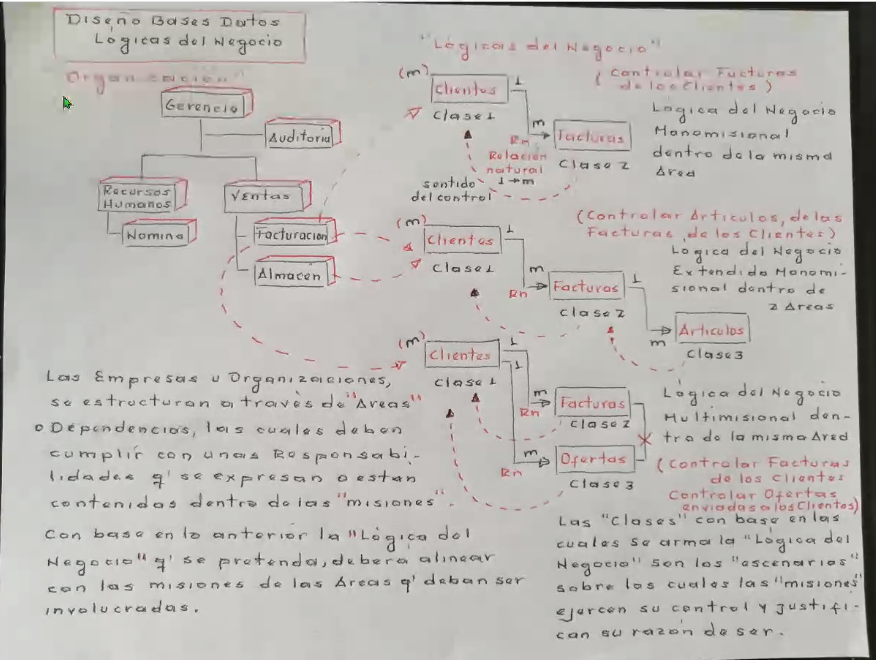
La misión del área de facturación es la de controlar las facturas de los clientes, sin comprometerse con el detalle de los artículos

La mono misionalidad, de la lógica de los negocios siempre declara un mínimo de dos clases, cuando es entendida como simple y solo hará referencia a un área o dependencia.

Modelo Lógica de Negocio Mono Misional Simple

Mono Misional Extendido

Este modelo identifica y vincula mas de dos clases, las cuales podrán ser extraídas de una o más áreas de dependencias,



MULTIMISIONAL SIMPLE

Este modelo, identifica dos o mas líneas de control en donde no existe, por lógica de negocio ningún tipo de relación entre las misiones, definidas por cada línea de control.

Por ser multimisionalidad simple, en cada una de las líneas de control, es decir en cada una de las misiones solo existirán dos clases.

Se advierte que la multimisionalidad, puede hacer referencia a una sola dependencia o área o mas de una dependencia.

MULTIMISIONAL EXTENDIDO

En esta lógica de negocios deberá existir mínimo en una línea de control, es decir en una de las misiones mas de dos clases, significa entonces que de igual manera cada línea de control podrá tener tres clases o más.

Las líneas de control del modelo miltimisional podrán involucrar clases de una sola dependencia, o mas, como sigue:

Se advierte que, para construir el modelo de la lógica de negocios, es importante e impositivo, conocer las misiones, de las áreas o dependencias, involucradas en el sistema de información pretendido.

Téngase en cuenta que la “lógica de los negocios” es diferente a “Inteligencia de negocios”. La primera define el escenario, sobre el cual se deberá construir el diseño de las bases de datos, la segunda se relaciona con los diferentes reportes he informes, que la empresa requiere para la analítica de los datos. No obstante, lo anterior es importante tener en cuenta que los atributos o datos, requeridos en la inteligencia de negocios son los que deberán ser asignados por la ley de la autodeterminación “Ley de la autodeterminación” en cada una de las clases del modelo de la lógica de negocio.

**SEGUNDO CAPITULO**

La ley de la autodeterminación (Ley de Armstrog) consiste en asignar de manera (absoluta), a cada una de las clases, del modelo de la lógica de negocios los atributos o datos que les corresponda.

Cuando se habla de la primera forma normal (1FN) esta reclama, de la existencia de un modelo de lógicas de negocios, de la aplicación de la ley de la autodeterminación:

Haciendo referencia al modelo mono misional simple facturas clientes, la ley de la autodeterminación, aplica como sigue:

La asignación absoluta de atributos significa que, en comienzo, los atributos asignados a una determinada clase son de esa clase y no de otras y viceversa; La ley de la autodeterminación nos permite desde el comienzo evitar y eliminar la redundancia de los datos.

Después de haberse sucedido la auto determinación se deberán identificar la primary key de cada una de las clases, la Foreign Key por Defecto (FKD) exista en cada una de ellas; son foreign key por defecto, todos aquellos atributos, que desde la misma aplicación de la ley de la autodeterminación, se comportan como atributos códigos.

Las dependencias funcionales hacen referencia a la relación inversa que existe entre la clase sucesora, y la clase antecesora del modelo de la lógica de negocio (Ri: Relación inversa); las dependencias funcionales son de tipo exclusivo (DFE) y no exclusivo (DFNE).

Una dependencia funcional exclusiva define una relación inversa de 1 a 1.

Una dependencia funcional no exclusiva define una relación inversa de 1 a muchos.

Para el ejemplo de lógica de negocios clientes facturas, existirá por relación inversa una dependencia funcional exclusiva, es decir de 1 a 1 con base a la siguiente reflexión:

1. Existen muchos clientes
2. Cada uno de los muchos clientes, puede tener muchas facturas
3. Cada una de las muchas facturas de un determinado cliente son de ese cliente y no de otros.

A cada una de las clases, que hacen parte del modelo de la lógica de negocios.

Antes de proceder a construir el diseño de la base de datos es importante asegurar “Que la planeación del diseño” a definido de manera correcta el tipo de dependencia funcional que se presenta por relación inversa (Rr) entre cada una de las clases.

Construcción del diseño de la base de datos (Lógica de negocios monomisional simple)

Cuando se ha definido una dependencia funcional exclusiva (dfe) por relación inversa entre las clases, la manera de solucionarla es la siguiente:

El atributo primary key (PK), de la clase antecesora deberá reflejarse (Ley de la Reflexivilidad) en la clase sucesora, en modo conjunto, entidad o tabla como Foreign Key por Proceso (FKP).

Se extrae que una Foreign Key por Proceso es el atributo primary key de una clase antecesora que resuelve una dependencia funcional exclusiva.

**Modelo Notacional de conjuntos**

Emerge como el primer modelo atreves del cual se muestra la construcción del diseño de la base de datos y por tanto comprende, de los siguientes procedimientos de normalización:

Primera Forma normal

Segunda Forma normal

Tercera Forma Normal

La primera Forma normal y segunda Forma normal (1FN 2FN), hacen parte de lo que se llama planeación de diseño de bases de datos.

La tercera forma normal, también y la cuarta y la quinta forma normal se relacionan ya no con la planeación, si no con la ” construcción del diseño de la bases de datos”.

Primera Forma normal 1FN: Se dice que la planeación del diseño de la base de datos se encuentra en primera forma normal cuando existe el modelo, de la lógica de negocios y existen una debida aplicación de la ley de la autodeterminación.

Segunda Forma normal 2FN: Se dice que la planeación del diseño de una base de datos se encuentra en segunda forma normal cuando estando en primera forma normal cumple con los siguientes requisitos.

1. Identificación del atributo PK para cada una de las clases de la lógica de negocios
2. Identificación de los atributos Foreign Key por Defecto FKDcontenidos en cada una de las clases.
3. Definición por relación inversa (Ri) del tipo de dependencia funcional que existen entre las clases (DFE - DFNE).

Tercera Forma normal 3FN: A partir de esta forma normal se habla de la construcción del “diseño de la base de datos”.

Se dice que la construcción del diseño de la base de datos se encuentra en tercera forma normal, cuando estando en 1FN y 2FN, cumple además con solucionar los tipos de dependencias funcionales que existen entre las clases por relación inversa.

Una dependencia funcional exclusiva siempre dará origen a una FKP.

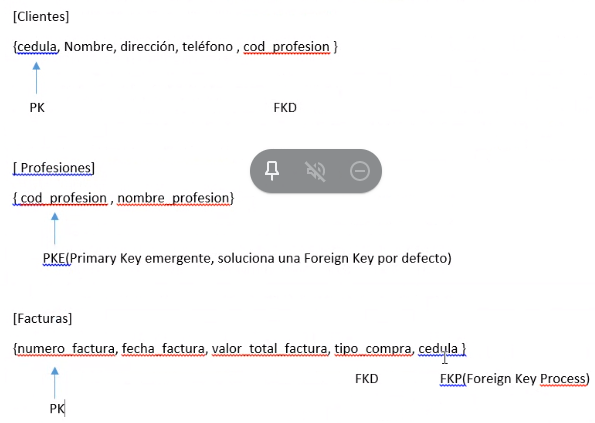
Hacen también parte de la tercera forma normal, solucionar los atributos FKD que se encuentran dentro de cada una de las clases.

Cuarta forma normal 4FN: La construcción del diseño de la base de datos, estará en 4FN, cuando cumpliendo con estar, en 1FN, 2FN y 3FN resuelve parcialmente una dependencia funcional no exclusiva armando una llave compuesta con la PK de la clase antecesora y la PK de la clase sucesora, asociando a esta llave compuesta los atributos de la clase sucesora que dependan estrictamente de las dos partes de la llave.

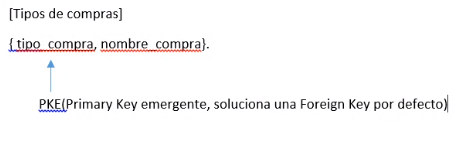
Quinta Forma normal 5FN: Se dice que la construcción del diseño de la base de datos se encuentra en 5FN cuando estando en 1FN, 2FN, 3FN y 4FN, se cumple con:

Identificar dentro de la clase sucesora parcialmente resuelta que atributos dependen estrictamente ya no de la llave compuesta, si no de la segunda parte de la llave.

Retomando el modelo de la lógica de negocios clientes – facturas, que hace referencia a una lógica monomisional simple y que tiene definido por relación inversa (Ri) una dependencia funcional exclusiva, la construcción del diseño a través del modelo notacional de conjuntos es el siguiente:



Este conjunto ha solucionado la Dependencia Funcional Exclusiva



Nota: téngase en cuenta que la metodología para construir los conjuntos del diseño de la base de datos indica que deberá hacerse a partir de la clase superior del modelo de la lógica de negocios

MODELO CONCEPTUAL (MODELO ENTIDAD RELACION DE PETER CHAN)

Para construir un modelo conceptual es haber construido previamente el modelo notacional de conjuntos.

La metodología para construir este modelo es la siguiente:

Inicie a la izquierda, con el conjunto en modo entidad que soluciono la ultima clase del modelo de la lógica de negocios.

Segundo: ubique a la derecha de la entidad de la izquierda los conjuntos en modo entidad que contienen las PK de la FK que se encuentran en la entidad de la izquierda.

Tercero: Repita el paso dos hasta haber involucrado todos los conjuntos que hayan sido previstos.

Cuarto: Relacione las entidades de derecha a izquierda, manteniendo siempre el 1 a la derecha y el muchos a la izquierda.

Quinto: En adelante deberá ser leído el modelo de derecha a izquierda

Sexto: Así como deberá ser leído el modelo, deberá ser llevado al motor.

Nota: Evite hasta donde sea posible, el cruce entre las relaciones.

Las únicas aristas de contacto entre las entidades del modelo serán:

1. La arista derecha de la entidad de la izquierda
2. La arista izquierda de la entidad de la derecha

Modelo relacional:

Este modelo se construye teniendo en cuenta la misma metodología utilizada en el modelo conceptual o entidad relación, teniendo en cuenta que en lugar de utilizarse entidad se utilizaran tablas.

Modelo DDL (Motor):

jjlondonop@udistrital.edu.co

create database facturacion;

use facturacion;

create table profesiones

(

codigo\_de\_profesion int(2),

nombre\_de\_profesion varchar(20),

primary key (codigo\_de\_profesion)

);

create table clientes

(

cedula int(8),

nombre varchar(30),

direccion varchar(30),

telefono int(10),

codigo\_de\_profesion int(2),

primary key (cedula),

foreign key (codigo\_de\_profesion) references profesiones (codigo\_de\_profesion)

);

create table tiposCompras

(

codigo\_de\_compra int(2),

nombre\_tipo\_compra varchar(20),

primary key (codigo\_de\_compra)

);

create table facturas

(

numero\_factura int(4),

fecha date,

valor\_total int,

cedula int(8),

codigo\_de\_compra int(2),

primary key (numero\_factura),

foreign key (cedula) references clientes(cedula),

foreign key (codigo\_de\_compra) references tiposCompras (codigo\_de\_compra)

);