

Estudiante: Anyair Pabón

Título: Aberturas Centro Pilar

Descripción: empresa dedicada a la fabricación y distribución de aberturas (rejas y aluminio) y puertas de marcas (PAVIR, OBLAK, BARMETAL, NEXO). El modelo de negocio que se quiere implementar, refleja el flujo de compra, control de stock, venta y envío de mercadería.

Nodos: Proveedor, Distribuidor, Productos, Facturación, Gestión_envío

Introducción

Aberturas Centro Pilar es una pequeña empresa que tiene como objetivo la distribución mayorista de puertas y portones de marca, así como la distribución de puertas, portones, ventanas y rejas de producción propia.

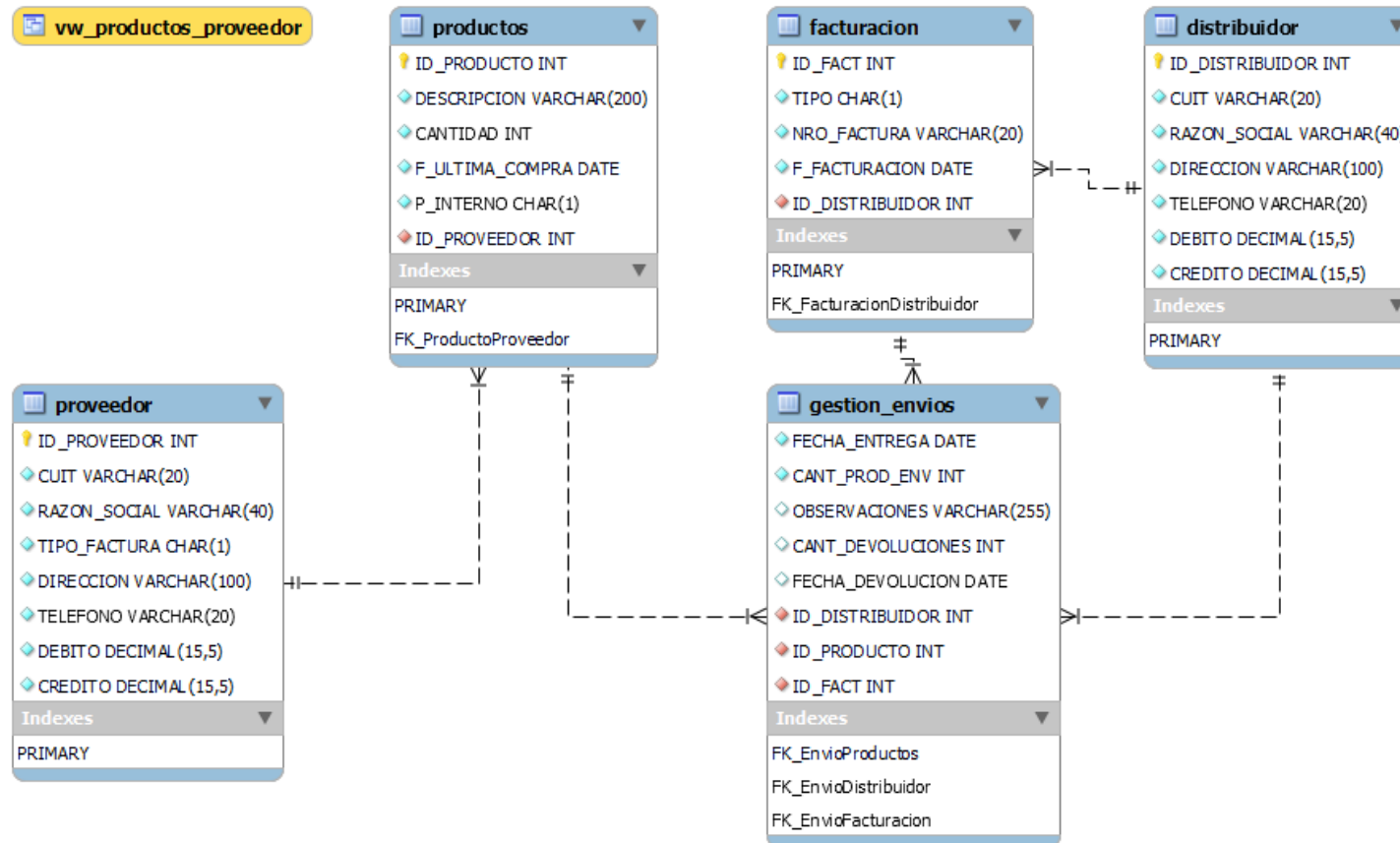
Con base en ello, se propone una base de datos relacional que le permita tener a la empresa un control puntual sobre sus movimientos de compra y entrega, teniendo como base madre una tabla transaccional denominada GESTION_ENVIOS. El modelo de negocio que ello desencadena, se compone en total de cinco tablas, que abordaremos a continuación.

Problemática

Dado que la compra de Centro Pilar es mayorista y su distribución también, debe tener un control de lo que tiene en stock, de las entregas según facturas que realiza, y además las devoluciones por daño que recibe.

Simultáneamente, debe tener un control contable sobre los débitos y créditos que la empresa tiene con sus proveedores, así como con sus distribuidores, de tal forma que pueda tener a la mano información importante para la toma de decisiones.

Diagrama Entidad-Relación



Descripción de tablas

Proveedor		
int	ID_PROVEEDOR	PK
varchar(20)	CUIT	
varchar(40)	NOMBRE	
char(1)	TIPO_FACTURA	
varchar(100)	DIRECCION	
varchar(20)	TELEFONO	
decimal(15,5)	DEBITO	
decimal(15,5)	CREDITO	
<p>Tabla transaccional en donde se tienen los distintos proveedores con los que cuenta la empresa. Se registran datos generales, el saldo deudor y a favor con la misma, y el tipo de factura que nos emite.</p>		

Distribuidor		
int	ID_DISTRIBUIDOR	PK
varchar(20)	CUIT	
varchar(40)	NOMBRE	
varchar(100)	DIRECCION	
varchar(20)	TELEFONO	
decimal(15,5)	DEBITO	
decimal(15,5)	CREDITO	
<p>Tabla transaccional. Se registran los distintos distribuidores de la empresa, sus datos generales, y el saldo deudor y a favor que ellos tienen.</p>		

Productos		
int	ID_PRODUCTO	PK
varchar(100)	DESCRIPCION	
int	CANTIDAD	
Date	F_ULTIMA_COMPRA	
char(1)	P_INTERNO	
int	ID_PROVEEDOR	FK
<p>Tabla transaccional. Registra todos los productos con los que cuenta la empresa, su descripción, la cantidad actual en stock, la fecha en que se realizó la última compra del producto, si es un producto producido por nosotros o no (P_INTERNO), y el proveedor al que se le compró.</p>		

Facturacion		
int	ID_FACT	PK
char(1)	TIPO	
varchar(20)	NRO_FACTURA	
Date	F_FACTURACION	
int	ID_DISTRIBUIDOR	FK
<p>Tabla transaccional. Se registran todas las ventas realizadas por la empresa, qué tipo de factura se emitió, número y fecha de factura, y a cuál distribuidor se le vendió.</p>		

GESTION_ENVIOS		
Date	FECHA_ENTREGA	
int	CANT_PROD_ENV	
varchar(255)	OBSERVACIONES	is null
int	CANT_DEVOLUCIONES	is null
date	FECHA_DEVOLUCION	is null
varchar(10)	EST_DEVOLUCION	is null *
int	ID_DISTRIBUIDOR	FK
int	ID_PRODUCTO	FK
int	ID_FACT	FK
<p>Tabla transaccional. Se registran todos los envíos realizados por la empresa. No cuenta con una Primary key debido a que en un mismo envío, se les puede llevar a varios distribuidores distintos productos. Se organiza por producto empaquetado, especificando la cantidad a enviar (CANT_PROD_ENV), la fecha de entrega, el distribuidor a entregar y la facturación correspondiente. Los campos de observaciones y cantidad de devoluciones son retro-activos post-entrega, especificando la fecha de la devolución del producto y el estado para resolver y reentregar.</p>		

Vistas

- vw_productos_proveedor: Muestra el nombre del producto, la cantidad y el nombre de su proveedor.
- vw_productos_devueltos: Muestra las devoluciones que se han recibido, por parte de un distribuidor, reflejando el débito y crédito que dicho distribuidor posee.
- vw_productos_re_entregador: Muestra el nombre de los productos re entregados, las fechas y las observaciones.
- vw_saldos_proveedor: Muestra el débito y el crédito que se tiene con los proveedores, y la diferencia.
- vw_saldos_distribuidor: Muestra el débito y el crédito que tienen los distribuidores, y la diferencia.

Funciones

- valores_repetidos: Encuentra las facturas que se repiten en la tabla de gestión de envíos, es decir: los envíos que han sido devueltos y re-entregado. Recibe por parámetros el número que indica las coincidencias. Por ejemplo, si se coloca 1, buscas todas las facturas que se repitan más de una vez.
- fn_get_credito_distribuidor: suma el crédito que tiene un distribuidor particular, o bien todos los distribuidores, depende de si se ingresa parámetro. El parámetro es la razón social del distribuidor.
- fn_get_descripcion: Obtiene el estado de una entrega a partir de su facturación. Tiene que ser una factura que no se repita.

Procedimientos almacenados

- sp_insert_and_delete: inserta y elimina de la tabla proveedor. Recibe por parámetro todos los campos de la tabla proveedor (CUIT, RAZON_SOCIAL, TIPO_FACTURA, DIRECCION, TELEFONO, DEBITO, CREDITO), y un campo FIELD_TO_DELETE1 que indica cuál valor se va a eliminar.
- order_statement: obtiene y ordena una tabla según el nombre de la tabla, el campo por el que se ordenara, y el tipo de orden pasados por parámetro.

Triggers

- tg_insert_insertar_en_gestion_envios: se dispara tras insertar un dato en la tabla gestion_envios. Se crea un registro en la tabla GESTION_ENVIOS_LOG

- `tg_delete_eliminar_en_gestion_envios`: se dispara tras eliminar un dato de la tabla `gestion_envios`. Se crea un registro en la tabla `GESTION_ENVIOS_LOG`
- `tg_update_actualizar_proveedor`: se ejecuta tras actualizar la tabla proveedor. Se inserta en la tabla `PROVEEDOR_LOG`.
- `tg_insert_insertar_proveedor`: se ejecuta tras insertar en la tabla proveedor. Se inserta en la tabla `PROVEEDOR_LOG`.

Tecnologías utilizadas

Las herramientas utilizadas para generar el conjunto de tablas y relaciones, así como la presentación, se compone de:

- MySQL – Workbench.
- Excel.
- Word.

Conclusiones

En síntesis, podemos destacar que el uso de una base de datos relacional permite a la empresa tener un mayor control sobre los productos que maneja, cuáles son los más vendidos, un control sobre las devoluciones que recibe y por qué, así como la satisfacción final del cliente.

El manejo de una base de datos, en conjunto con sus funciones, vistas, triggers y procedimientos almacenados, permite automatizar toda la carga de flujo que la empresa acarrea por su naturaleza.

Es importante señalar además que las vistas y las funciones son muy importantes al momento de querer relevar información puntual o relacionada acerca de la dinámica y compras/ventas, saldos a favor y en contra que tiene la empresa. Ello permite tener una visualización de su comportamiento que posiblemente permee las decisiones de negocio.