



Universidad Tecnológica Nacional
Facultad Regional Buenos Aires

Bases de datos

Trabajo Práctico: Fábrica de sillones

Estrategia

Número de grupo: 3

Nombre del grupo: LOS_POLLOS_HERMANOS

Integrantes del grupo:

| Apellido | Nombre | Legajo |
|------------|----------|-----------|
| Herzkovich | Agustín | 213.787-2 |
| Piacentini | Nicolás | 214.001-9 |
| Sorrentino | Tadeo | 214.153-0 |
| Maturano | Gabriela | 172.649-3 |

Índice

| | |
|--|-----------|
| Modelo de datos..... | 2 |
| Entidades y sus relaciones..... | 3 |
| (1) Cliente..... | 3 |
| (2) Factura..... | 3 |
| (3) Sucursal..... | 4 |
| (4) Proveedor..... | 5 |
| (5) Ubicacion..... | 5 |
| (6) Compra..... | 6 |
| (7) DetallePedido..... | 6 |
| (8) Sillon..... | 7 |
| (9) Pedido..... | 7 |
| (10) Material..... | 8 |
| (11) Medida..... | 8 |
| (12) Modelo..... | 9 |
| (13) TipoMaterial..... | 9 |
| (14) Tela..... | 10 |
| (15) Madera..... | 10 |
| (16) Relleno..... | 10 |
| (17) DetalleCompra..... | 11 |
| (18) PedidoCancelacion..... | 11 |
| (19) Envio..... | 11 |
| (20) DetalleFactura..... | 12 |
| (21) MaterialPorSillon..... | 12 |
| Inconsistencias..... | 13 |
| Normalización..... | 15 |
| Diseño..... | 16 |

Modelo de datos

Se creó un modelo de datos que organiza y normaliza los datos de la tabla provista por la cátedra. Para esto creamos 19 nuevas tablas con PK y FK para relacionarlas entre ellas. En el siguiente cuadro se detalla cómo se relacionan entre sí. Ejemplo: PK (1) se relaciona con FK (1)

| Entidad | Primary Key | Foreign Key |
|-------------------|---|---|
| Cliente | Cliente_Id (bigint) (1) | Cliente_Ubicacion (bigint) (5) |
| Factura | Factura_Numero (bigint) (2) | Factura_Cliente (bigint) (1) Factura_Sucursal (bigint) (3) |
| DetalleFactura | Detalle_Factura_Numero (bigint) | Detalle_Factura_Factura (bigint) (2) Detalle_Factura_Detalle_Pedido (bigint) (7) |
| Sucursal | Sucursal_Numero (bigint) (3) | Sucursal_Ubicacion (bigint) (5) |
| Proveedor | Proveedor_Id (bigint) (4) | Proveedor_Ubicacion (bigint) (5) |
| Ubicacion | Ubicacion_Id (bigint) (5) | |
| Compra | Compra_Numero (decimal(18,0)) (6) | Compra_Sucursal (bigint) (3) Compra_Proveedor (bigint) (4) |
| DetalleCompra | Detalle_Compra_Codigo (bigint) | Detalle_Compra_Compra (decimal(18,0)) (6) Detalle_Compra_Material (bigint) (10) |
| Envio | Envio_Numero (decimal(18,0)) | Envio_Factura (bigint) (2) |
| DetallePedido | Detalle_Pedido_Numero (bigint) (7) | Detalle_Pedido_Sillon (bigint) (8) Detalle_Pedido_Pedido (decimal(18,0)) (9) |
| Sillon | Sillon_Codigo (bigint) (8) | Sillon_Medida (bigint) (11) Sillon_Modelo (bigint) (12) |
| Pedido | Pedido_Numero (decimal(18,0)) (9) | Pedido_Cliente (bigint) (1) Pedido_Sucursal (bigint) (3) |
| Material | Material_Codigo (bigint) (10) | Material_Tipo (bigint) (13) |
| Relleno | Relleno_Id (bigint) | Relleno_Id (bigint) (13) |
| Tela | Tela_Id (bigint) | Tela_Id (bigint) (13) |
| Madera | Madera_Id (bigint) | Madera_Id (bigint) (13) |
| PedidoCancelacion | Pedido_Cancelacion_Numero (bigint) | Pedido_Cancelacion_Pedido (decimal(18,0)) (9) |
| Medida | Medida_Codigo (bigint) (11) | |
| Modelo | Modelo_Codigo (bigint) (12) | |
| MaterialPorSillon | MaterialPorSillon_Codigo (bigint) | MaterialPorSillon_Material (bigint) (10) MaterialPorSillon_Sillon (bigint) (8) |
| TipoMaterial | TipoMaterial_Codigo (bigint) (13) | |

Entidades y sus relaciones

(1) Cliente

Esta entidad representa al cliente y contiene toda la información del cliente requerida para la realización de un pedido, facturación y gestión de envíos.

Creamos la PK *Cliente_Id* para identificar de forma unívoca a la entidad *Cliente* ya que se encontraron casos en donde dos clientes comparten Dni en la tabla Maestra.

- Relación con Pedido: Cliente se relaciona con la entidad Pedido a través de la PK *Cliente_Id*. En esta relación, Un *Pedido* tiene un único *Cliente* obligatorio pero *Cliente* tiene uno o muchos *Pedido*.
- Relación con Factura: Cliente se relaciona con la entidad Factura a través de la PK *Cliente_Id*. En esta relación, una *Factura* tiene un *Cliente* y un *Cliente* puede tener muchas *Factura*.
- Relación con Ubicación: Cliente se relaciona con Ubicación a través de la FK *Cliente_Ubicacion*, donde una *Ubicacion* puede tener un *Cliente* y un *Cliente* tiene una *Ubicacion*.

| Atributo | Tipo de Dato | Tipo de Clave |
|--------------------------|---------------|---------------|
| Cliente_Id | bigint | Primary Key |
| Cliente_Ubicacion | bigint | Foreign Key |
| Cliente_Dni | bigint | |
| Cliente_Nombre | nvarchar(255) | |
| Cliente_Apellido | nvarchar(255) | |
| Cliente_Fecha_Nacimiento | datetime2(6) | |
| Cliente_Mail | nvarchar(255) | |
| Cliente_Telefono | nvarchar(255) | |

(2) Factura

Esta entidad representa la factura que se registra cuando el pedido está completado y el cliente realiza el pago. Todas las facturas registran la información contenida en esta entidad.

- Relación con DetalleFactura: Factura se relaciona con la entidad DetalleFactura a través de la PK *Factura_Numero*, donde una *Factura* tiene al menos un *DetalleFactura* y un *DetalleFactura* tiene una única *Factura*.
- Relación con Envio: Factura se relaciona con la entidad Envio a través de la PK *Factura_Numero*, donde una *Factura* tiene un *Envio* y un *Envio* tiene una *Factura*.

- Relación con Sucursal: Factura se relaciona con la entidad Sucursal a través de la FK Factura_Sucursal, donde: una *Factura* tiene una *Sucursal* y una *Sucursal* puede tener cero o muchas *Factura*.
- Relación con Cliente: Factura se relaciona con un Cliente a través de la FK Factura_Cliente, donde una *Factura* tiene un *Cliente* y un *Cliente* puede tener muchas *Factura*.

| Atributo | Tipo de Dato | Tipo de Clave |
|------------------|---------------|---------------|
| Factura_Numero | bigint | Primary Key |
| Factura_Cliente | bigint | Foreign Key |
| Factura_Sucursal | bigint | Foreign Key |
| Factura_Fecha | datetime2(6) | |
| Factura_Total | decimal(38,2) | |

(3) Sucursal

La fábrica cuenta con varias sucursales. Esta entidad representa a cada una de ellas para poder registrar en cuál de ellas fue realizada cada factura, pedido o compra a proveedor.

- Relación con Factura: Sucursal se relaciona con Factura a través de la PK Sucursal_Numero. En esta relación, una *Factura* tiene una *Sucursal obligatoria* y una *Sucursal* puede tener muchas o ninguna *Factura*.
- Relación con Compra: Sucursal se relaciona con Compra a través de la PK Sucursal_Numero. En esta relación, una *Compra* tiene una *Sucursal* y una *Sucursal* puede tener muchas *Compra* o ninguna.
- Relación con Pedido: Sucursal se relaciona con pedido a través de la PK Sucursal_Numero. En esta relación, un *Pedido* tiene una única *Sucursal* y una *Sucursal* puede relacionarse con muchos *Pedido* o ninguno.
- Relación con Ubicación: Sucursal se relaciona con Ubicacion a través de la FK Sucursal_Ubicacion. En esta relación, una *Ubicacion* puede tener una o ninguna *Sucursal* y una *Sucursal* tiene una única *Ubicacion*.

| Atributo | Tipo de Dato | Tipo de Clave |
|--------------------|---------------|---------------|
| Sucursal_Numero | bigint | Primary Key |
| Sucursal_Ubicacion | bigint | Foreign Key |
| Sucursal_Telefono | nvarchar(255) | |
| Sucursal_Mail | nvarchar(255) | |

(4) Proveedor

Esta entidad representa a cada proveedor al que se le realizan compras de los distintos tipos de materiales: tela, madera y relleno.

Si bien *Proveedor_Cuit* no se repite, decidimos crear una Primary Key *Proveedor_Id* para asegurarnos de manejarlos mejor y cerciorarnos de que en un futuro no se repitan.

- Relación con Compra: Proveedor se relaciona con Compra a través de su PK *Proveedor_Id*. En esta relación una *Compra* tiene un *Proveedor* y un *Proveedor* puede tener muchas *Compra*.
- Relación con Ubicación: Proveedor se relaciona con Ubicación a través de su FK *Proveedor_Ubicacion*. Una *Ubicación* puede tener uno o ningún *Proveedor* y un *Proveedor* tiene una *Ubicación*.

| Atributo | Tipo de Dato | Tipo de Clave |
|-----------------------|---------------|---------------|
| Proveedor_Id | bigint | Primary Key |
| Proveedor_Ubicacion | bigint | Foreign Key |
| Proveedor_Cuit | nvarchar(255) | |
| Proveedor_RazonSocial | nvarchar(255) | |
| Proveedor_Telefono | nvarchar(255) | |
| Proveedor_Mail | nvarchar(255) | |

(5) Ubicacion

Esta entidad representa la Ubicación y contiene los atributos Provincia, Localidad y Dirección. Tiene relación con las entidades Proveedor, Cliente y Sucursal ya que en todas se requiere registrar estos datos.

- Relación con Sucursal: Una *Ubicación* puede tener una *Sucursal* y una *Sucursal* tiene una *Ubicación*.
- Relación con Proveedor: Una *Ubicación* puede tener un *Proveedor* y un *Proveedor* tiene una *Ubicación*.
- Relación con Cliente: Una *Ubicación* puede tener un *Cliente* y un *Cliente* tiene una *Ubicación*.

| Atributo | Tipo de Dato | Tipo de Clave |
|---------------------|---------------|---------------|
| Ubicacion_Id | bigint | Primary Key |
| Ubicacion_Provincia | nvarchar(255) | |
| Ubicacion_Localidad | nvarchar(255) | |
| Ubicacion_Direccion | nvarchar(255) | |

(6) Compra

Esta entidad representa cada compra que realiza la fábrica de sillones a sus proveedores. Por cada compra se registra la información contenida en esta entidad.

- Relación con DetalleCompra: Una *Compra* tiene uno o muchos *DetalleCompra* y un *DetalleCompra* tiene una *Compra* obligatoria.
- Relación con Proveedor: Una *Compra* tiene un *Proveedor* y un *Proveedor* tiene una o ninguna *Compra*.
- Relación con Sucursal: Una *Compra* tiene una *Sucursal* y una *Sucursal* tiene muchas o ninguna *Compra*.

| Atributo | Tipo de Dato | Tipo de Clave |
|------------------|---------------|---------------|
| Compra_Numero | decimal(18,0) | Primary Key |
| Compra_Sucursal | bigint | Foreign Key |
| Compra_Proveedor | bigint | Foreign Key |
| Compra_Fecha | datetime2(6) | |
| Compra_Total | decimal(18,2) | |

(7) DetallePedido

Cada pedido puede tener varios sillones distintos, por cada sillón distinto a fabricar, se registra la información contenida en esta entidad.

- Relación con DetalleFactura: Un *DetallePedido* puede tener 0 o 1 *DetalleFactura*. *DetalleFactura* está relacionado sí o sí a un *DetallePedido* que ya existe.
- Relación con Sillon: Un *DetallePedido* tiene un *Sillon* y un *Sillon* puede relacionarse con muchos *DetallePedido*.
- Relación con Pedido: Un *Pedido* tiene uno o muchos *DetallePedido* ya que se realiza por *Sillon* y un *DetallePedido* está asociado a un único *Pedido* obligatorio.

| Atributo | Tipo de Dato | Tipo de Clave |
|-------------------------|---------------|---------------|
| Detalle_Pedido_Numero | bigint | Primary Key |
| Detalle_Pedido_Sillon | bigint | Foreign Key |
| Detalle_Pedido_Pedido | decimal(18,0) | Foreign Key |
| Detalle_Pedido_Cantidad | bigint | |
| Detalle_Pedido_Precio | decimal(18,2) | |
| Detalle_Pedido_Subtotal | decimal(18,2) | |

(8) Sillon

Esta entidad representa a cada sillón, donde el cliente es quien elige las medidas, modelo, y materiales que tendrá el mismo.

Creamos una entidad intermedia *MaterialPorSillon* entre las entidades *Sillon* y *Material* que permite modelar correctamente la relación de muchos a muchos entre estas entidades.

- Relación con DetallePedido: Un *DetallePedido* corresponde a un *Sillon* y un *Sillon* puede tener muchos *DetallePedido*.
- Relación con MaterialPorSillon: Un *Sillon* tiene uno o muchos *MaterialPorSillon*. Un *MaterialPorSillon* tiene un *Sillon*.
- Relación con Modelo: Un *Sillon* tiene un *Modelo* y un *Modelo* puede relacionarse con muchos *Sillon*.
- Relación con Medida: Un *Sillon* tiene una *Medida* y una *Medida* puede relacionarse con muchos *Sillon*.

| Atributo | Tipo de Dato | Tipo de Clave |
|---------------|--------------|---------------|
| Sillon_Codigo | bigint | Primary Key |
| Sillon_Medida | bigint | Foreign Key |
| Sillon_Modelo | bigint | Foreign Key |

(9) Pedido

Esta entidad representa a cada pedido que realiza un cliente

- Relación con PedidoCancelacion: Un *Pedido* puede tener un *PedidoCancelacion* o ninguno y un *PedidoCancelacion* está asociado sí o sí a un *Pedido* que ya existe.
- Relación con DetallePedido: Un *Pedido* tiene uno o muchos *DetallePedido* ya que se realiza por *Sillon* y un *DetallePedido* está asociado a un único *Pedido* obligatorio.
- Relación con Cliente: Un *Pedido* tiene un único *Cliente* obligatorio pero *Cliente* tiene uno o muchos *Pedido*.
- Relación con Sucursal: Un *Pedido* tiene una única *Sucursal* y una *Sucursal* puede relacionarse con muchos *Pedido* o ninguno.

| Atributo | Tipo de Dato | Tipo de Clave |
|-----------------|---------------|---------------|
| Pedido_Numero | decimal(18,0) | Primary Key |
| Pedido_Cliente | bigint | Foreign Key |
| Pedido_Sucursal | bigint | Foreign Key |
| Pedido_Fecha | datetime2(6) | |
| Pedido_Total | decimal(18,2) | |
| Pedido_Estado | nvarchar(255) | |

(10) Material

Esta entidad representa a cada material que puede elegir el cliente para su sillón. Entre los materiales se encuentran distintos tipos de telas, maderas y rellenos. Cada uno de los materiales tiene un precio y características.

Creamos una entidad intermedia *MaterialPorSillon* entre las entidades *Sillon* y *Material* que permite modelar correctamente la relación de muchos a muchos entre estas entidades.

- Relación con DetalleCompra: Un *DetalleCompra* tiene un *Material* y un *Material* puede tener muchos *DetalleCompra* o ninguno.
- Relación con MaterialPorSillon: Un *Material* tiene uno o muchos *MaterialPorSillon*, y un *MaterialPorSillon* tiene un único *Material*.
- Relación con TipoMaterial: Un *TipoMaterial* puede tener uno o muchos *Material*. Un *Material* sólo puede tener un *TipoMaterial*.

| Atributo | Tipo de Dato | Tipo de Clave |
|----------------------|---------------|---------------|
| Material_Codigo | bigint | Primary Key |
| Material_Tipo | bigint | Foreign Key |
| Material_Nombre | nvarchar(255) | |
| Material_Descripcion | nvarchar(255) | |
| Material_Precio | decimal(38,2) | |

(11) Medida

Esta entidad representa a cada medida que es elegida por el cliente: ancho, alto y profundidad del sillón. Las medidas se pueden aplicar a cualquier modelo de sillón y cada una tiene un precio.

- Relación con Sillon: Un *Sillon* tiene una *Medida* y una *Medida* puede relacionarse con muchos *Sillon*.

| Atributo | Tipo de Dato | Tipo de Clave |
|--------------------|---------------|---------------|
| Medida_Codigo | bigint | Primary Key |
| Medida_Ancho | decimal(18,2) | |
| Medida_Alto | decimal(18,2) | |
| Medida_Profundidad | decimal(18,2) | |
| Medida_Precio | decimal(18,2) | |

(12) Modelo

Esta entidad representa a cada modelo de sillón. Cada uno de los modelos tiene un precio base.

El atributo *Modelo_Codigo_Numero* en el modelo que diseñamos se corresponde con el atributo *Sillon_Modelo* de la tabla maestra.

- Relación con Sillon: Un *Sillon* tiene un *Modelo* y un *Modelo* puede relacionarse con muchos *Sillon*.

| Atributo | Tipo de Dato | Tipo de Clave |
|----------------------|---------------|---------------|
| Modelo_Codigo | bigint | Primary Key |
| Modelo_Codigo_Numero | nvarchar(255) | |
| Modelo_Descripcion | nvarchar(255) | |
| Modelo_Precio_Base | decimal(18,2) | |

(13) TipoMaterial

Esta entidad se relaciona con *Material* y con cada uno de los tipos de materiales (*Tela*, *Madera*, *Relleno*) que contienen los atributos específicos de cada tipo de material.

El atributo de esta entidad llamado *TipoMaterial_Tipo* se corresponde con el atributo *Material_Tipo* de la tabla maestra.

- Relación con Material: Un *TipoMaterial* puede tener muchos *Material*. Un *Material* sólo puede tener un *TipoMaterial*.
- Relación con Tela: Un *TipoMaterial* tiene una *Tela* y una *Tela* tiene un *TipoMaterial*.
- Relación con Madera: Un *TipoMaterial* tiene una *Madera* y una *Madera* tiene una *TipoMaterial*.
- Relación con Relleno: Un *TipoMaterial* tiene un *Relleno* y un *Relleno* tiene un *TipoMaterial*.

| Atributo | Tipo de Dato | Tipo de Clave |
|---------------------|---------------|---------------|
| TipoMaterial_Codigo | bigint | Primary Key |
| TipoMaterial_Tipo | nvarchar(255) | |

(14) Tela

Esta entidad representa a cada una de las telas con la que se fabrican los sillones y contiene el color y la textura de las mismas.

- Relación con TipoMaterial: Un *TipoMaterial* tiene una *Tela* y una *Tela* tiene un *TipoMaterial*.

| Atributo | Tipo de Dato | Tipo de Clave |
|--------------|---------------|--------------------------|
| Tela_Id | bigint | Primary Key, Foreign Key |
| Tela_Color | nvarchar(255) | |
| Tela_Textura | nvarchar(255) | |

(15) Madera

Esta entidad representa a cada una de las maderas con la que se fabrican los sillones y contiene el color y la dureza de las mismas.

- Relación con TipoMaterial: Un *TipoMaterial* tiene una *Madera* y una *Madera* tiene un *TipoMaterial*.

| Atributo | Tipo de Dato | Tipo de Clave |
|---------------|---------------|--------------------------|
| Madera_Id | bigint | Primary Key, Foreign Key |
| Madera_Color | nvarchar(255) | |
| Madera_Dureza | nvarchar(255) | |

(16) Relleno

Esta entidad representa a cada una de los rellenos con los que se fabrican los sillones y contiene información sobre la densidad de los mismos.

- Relación con TipoMaterial: Un *TipoMaterial* tiene un *Relleno* y un *Relleno* tiene un *TipoMaterial*.

| Atributo | Tipo de Dato | Tipo de Clave |
|------------------|---------------|--------------------------|
| Relleno_Id | bigint | Primary Key, Foreign Key |
| Relleno_Densidad | decimal(38,2) | |

(17) DetalleCompra

Esta entidad representa el detalle de compra de cada material que se compra al proveedor. Es una entidad intermedia entre las entidades Compra y Material para poder representar la relación de muchos a muchos entre ellas.

| Atributo | Tipo de Dato | Tipo de Clave |
|-------------------------|---------------|---------------|
| Detalle_Compra_Codigo | bigint | Primary Key |
| Detalle_Compra_Compra | decimal(18,0) | Foreign Key |
| Detalle_Compra_Material | bigint | Foreign Key |
| Detalle_Compra_Cantidad | decimal(18,0) | |
| Detalle_Compra_Precio | decimal(18,2) | |
| Detalle_Compra_Subtotal | decimal(18,2) | |

(18) PedidoCancelacion

Esta entidad representa la información de cada pedido cancelado.

- Relación con Pedido: un Pedido tiene uno o ningún PedidoCancelacion, y un PedidoCancelacion tiene un único Pedido.

| Atributo | Tipo de Dato | Tipo de Clave |
|---------------------------|---------------|---------------|
| Pedido_Cancelacion_Numero | bigint | Primary Key |
| Pedido_Cancelacion_Pedido | decimal(18,0) | Foreign Key |
| Pedido_Cancelacion_Fecha | datetime2(6) | |
| Pedido_Cancelacion_Motivo | varchar(255) | |

(19) Envio

Esta entidad contiene la información de envío relacionada a cada venta.

- Relación con Factura: una Factura tiene una único Envio, al igual que cada Envio tiene una única Factura.

| Atributo | Tipo de Dato | Tipo de Clave |
|------------------------|---------------|---------------|
| Envio_Numero | decimal(18,0) | Primary Key |
| Envio_Factura | bigint | Foreign Key |
| Envio_Fecha_Programada | datetime2(6) | |
| Envio_Fecha_Entrega | datetime2(6) | |
| Envio_Importe_Traslado | decimal(18,2) | |
| Envio_Importe_Subida | decimal(18,2) | |

(20) DetalleFactura

Esta entidad representa al detalle de la factura, conteniendo información sobre los ítems de la factura. Relacionándose tanto con la Factura como con el DetallePedido al que están asociados a los ítems facturados.

- Relación con Factura: una Factura puede tener uno o muchos DetalleFactura, un DetalleFactura puede pertenecer sólo una Factura.
- Relación con DetallePedido: un DetallePedido puede tener cero o un DetalleFactura. Un DetalleFactura puede tener un único DetallePedido.

| Atributo | Tipo de Dato | Tipo de Clave |
|-------------------------------|---------------|---------------|
| Detalle_Factura_Numero | bigint | Primary Key |
| Detalle_Factura_Factura | bigint | Foreign Key |
| Detalle_Factura_DetallePedido | bigint | Foreign Key |
| Detalle_Factura_Cantidad | decimal(18,0) | |
| Detalle_Factura_Precio | decimal(18,2) | |
| Detalle_Factura_Subtotal | decimal(18,2) | |

(21) MaterialPorSillon

Creamos una entidad intermedia *MaterialPorSillon* entre las entidades *Sillon* y *Material* que permite modelar correctamente la relación de muchos a muchos entre estas entidades.

- Relación con Sillon: Un Sillon puede tener uno o muchos MaterialPorSillon. Un MaterialPorSillon puede estar únicamente en un Sillon.
- Relación con Material: Un Material puede estar en uno o muchos MaterialPorSillon. Un MaterialPorSillon puede corresponderse únicamente con un Material.

| Atributo | Tipo de Dato | Tipo de Clave |
|----------------------------|--------------|---------------|
| MaterialPorSillon_Codigo | bigint | Primary Key |
| MaterialPorSillon_Material | bigint | Foreign Key |
| MaterialPorSillon_Sillon | bigint | Foreign Key |

Aclaración: Todas las PK que creamos nosotros (es decir, no estaban en la Tabla Maestra) las hicimos de tipo *bigint* para poder definirlos como *Autoincrement*.

Inconsistencias

- Se encontraron emails, nombres de calles, provincias, localidades y nombres de clientes, proveedores y sucursales que contienen espacios, símbolos especiales, punto y coma y tildes. Estos errores sintácticos tendrán un impacto en la base de datos ya que, por ejemplo, al hacer consultas, en el caso de querer seleccionar a los clientes de “Santiago del Estero”, habrá complicaciones al no estar escrito de esa forma.

Santia; Del Estero

Avenida Crisólo; Laralde N° 7479

- Todos los pedidos y facturas están realizados en fechas futuras, entre 2026 y 2028. Esto podría tener un impacto en la base de datos al no estar cronológicamente ordenado. Cuando lleguen esas fechas, se tendrán nuevamente pedidos y facturas con la misma fecha en la que ya se facturó o generó un pedido. Podrá originar problemas al buscarlas o intentar diferenciar cuáles corresponden a la fecha correcta.

| Results | | Messages |
|---------|----------------------------|----------------------------|
| | factura_fecha | pedido_fecha |
| 1 | 2026-01-11 08:00:00.000000 | 2026-01-01 08:00:00.000000 |
| 95873 | 2028-07-24 20:00:00.000000 | 2028-06-29 20:00:00.000000 |

- Hay clientes que tienen el mismo DNI que otra persona, esto puede impactar en la base de datos ya que, al querer buscar un cliente por DNI, se tendrán múltiples resultados, no sabiendo cuál es el correcto.

| | cliente_Dni | cliente_nombre | cliente_apellido |
|---|-------------|----------------|------------------|
| 1 | 2718249 | LAYA | Lucero |
| 2 | 2718249 | PIERRE | Álvarez |
| 3 | 16085930 | CLODOMIRO | Silva |
| 4 | 16085930 | JEHIEL | Rojas |
| 5 | 21173566 | FLABIAN | Para |
| 6 | 21173566 | PERCIVAL | Riquelme |

- La mayoría de los clientes tienen un DNI que no corresponde con su fecha de nacimiento, esto puede impactar en la base de datos ya que, al querer buscar un cliente por DNI, el DNI del cliente puede no corresponder con el DNI que está en la base de datos.

| Cliente_FechaNacimiento | cliente_dni |
|----------------------------|-------------|
| 1994-10-17 20:01:59.190000 | 1316729 |
| 1994-10-17 20:01:59.190000 | 1316729 |

- Encontramos casos en los que *Detalle_Compra_Precio * Detalle_Compra_Cantidad* no coincide completamente con *Detalle_Compra_SubTotal*, sino que hay una leve diferencia en el resultado del cálculo en comparación con los datos cargados en la Tabla Maestra. Aquí mostramos el subtotal ya cargado en comparación con el calculado y a continuación la diferencia.

| Results | | Messages | | | | |
|---------|----------------------|-------------------------|-------------------------|---------------------|------------|--|
| | Detalle_compraPrecio | detalle_compra_cantidad | detalle_compra_subtotal | Resultado_Calculado | Diferencia | |
| 1 | 1054.28 | 2515 | 2651524.37 | 2651514.20 | 10.17 | |
| 2 | 1054.28 | 2511 | 2647307.24 | 2647297.08 | 10.16 | |
| 3 | 1054.28 | 2466 | 2599864.46 | 2599854.48 | 9.98 | |
| 4 | 1054.28 | 2465 | 2598810.17 | 2598800.20 | 9.97 | |
| 5 | 1054.28 | 2454 | 2587213.05 | 2587203.12 | 9.93 | |
| 6 | 1054.28 | 2442 | 2574561.64 | 2574551.76 | 9.88 | |
| 7 | 1054.28 | 2437 | 2569290.22 | 2569280.36 | 9.86 | |
| 8 | 1054.28 | 2417 | 2548204.54 | 2548194.76 | 9.78 | |
| 9 | 1054.28 | 2405 | 2535553.13 | 2535543.40 | 9.73 | |
| 10 | 1054.28 | 2394 | 2523956.00 | 2523946.32 | 9.68 | |
| 11 | 1054.28 | 2385 | 2514467.45 | 2514457.80 | 9.65 | |
| 12 | 1054.28 | 2385 | 2514467.45 | 2514457.80 | 9.65 | |
| 13 | 1054.28 | 2354 | 2481784.64 | 2481775.12 | 9.52 | |
| 14 | 1054.28 | 2348 | 2475458.94 | 2475449.44 | 9.50 | |
| 15 | 1054.28 | 2273 | 2396387.64 | 2396378.44 | 9.20 | |
| 16 | 1054.28 | 2246 | 2367921.97 | 2367912.88 | 9.09 | |
| 17 | 1054.28 | 2229 | 2349999.14 | 2349990.12 | 9.02 | |
| 18 | 1054.28 | 2201 | 2320479.18 | 2320470.28 | 8.90 | |
| 19 | 1054.28 | 2185 | 2303610.64 | 2303601.80 | 8.84 | |
| 20 | 1054.28 | 2169 | 2286742.09 | 2286733.32 | 8.77 | |
| 21 | 1054.28 | 2130 | 2245625.02 | 2245616.40 | 8.62 | |
| 22 | 1054.28 | 2083 | 2196073.67 | 2196065.24 | 8.43 | |
| 23 | 1054.28 | 2069 | 2181313.69 | 2181305.32 | 8.37 | |
| 24 | 1250.01 | 2659 | 3323784.91 | 3323776.59 | 8.32 | |
| 25 | 1054.28 | 2055 | 2166553.71 | 2166545.40 | 8.31 | |
| 26 | 1054.28 | 2049 | 2160228.01 | 2160219.72 | 8.29 | |
| 27 | 1054.28 | 2017 | 2126490.92 | 2126482.76 | 8.16 | |
| 28 | 1054.28 | 2015 | 2124382.35 | 2124374.20 | 8.15 | |
| 29 | 1250.01 | 2592 | 3240034.03 | 3240025.92 | 8.11 | |
| 30 | 1054.28 | 1990 | 2098025.25 | 2098017.20 | 8.05 | |
| 31 | 1250.01 | 2564 | 3205033.66 | 3205025.64 | 8.02 | |
| 32 | 1250.01 | 2547 | 3183783.44 | 3183775.47 | 7.97 | |
| 33 | 1054.28 | 1950 | 2055853.89 | 2055846.00 | 7.89 | |

Esto también ocurre con los demás detalles, pero aquí lo ejemplificamos con Detalle_Compra. Si bien la diferencia no es muy significativa, lo aclaramos por si acaso.

Normalización

- Consideramos la opción de normalizar las direcciones a través de la creación de una entidad *Direccion* que tuviera los atributos *nombre_calle* y *altura_calle* para separar el nombre de la calle de la altura en las direcciones, pero al no saber si la calle iba a estar siempre separada del número por un “ N° ”, decidimos dejarlo como está en la tabla maestra.

| | cliente_Direccion | nombre_calle_normalizado | altura_calle_normalizado |
|----|----------------------------------|--------------------------|--------------------------|
| 1 | Valdenegro N° 9489 | Valdenegro | 9489 |
| 2 | Valdenegro N° 9489 | Valdenegro | 9489 |
| 3 | Valdenegro N° 9489 | Valdenegro | 9489 |
| 4 | Valdenegro N° 9489 | Valdenegro | 9489 |
| 5 | Valdenegro N° 9489 | Valdenegro | 9489 |
| 6 | Avenida Figueroa Alcorta N° 6197 | Avenida Figueroa Alcorta | 6197 |
| 7 | Avenida Figueroa Alcorta N° 6197 | Avenida Figueroa Alcorta | 6197 |
| 8 | Avenida Figueroa Alcorta N° 6197 | Avenida Figueroa Alcorta | 6197 |
| 9 | Avenida Figueroa Alcorta N° 6197 | Avenida Figueroa Alcorta | 6197 |
| 10 | Avenida Brasil N° 7374 | Avenida Brasil | 7374 |
| 11 | Avenida Brasil N° 7374 | Avenida Brasil | 7374 |

- Redundancias: hay campos que tienen el mismo nombre que el atributo que representa donde se podría almacenar únicamente el valor, liberando así el espacio en memoria que ocuparía el nombre del campo almacenado como varchar en cada fila.

| | proveedor_razonSocial |
|----|-----------------------|
| 1 | Razon Social N°7 |
| 2 | Razon Social N°7 |
| 17 | Razon Social N°5 |
| 18 | Razon Social N°5 |
| 19 | Razon Social N°4 |

Lo mismo ocurre en *sillon_modelo_codigo*, *sillon_modelo* y *sillon_modelo_descripcion*, que repiten el nombre del atributo en cada registro

| sillon_modelo_codigo | sillon_modelo | sillon_modelo_Descripcion |
|----------------------|-------------------|-------------------------------|
| 435448 | Modelo N°: 435448 | Descripcion Modelo N°: 435448 |
| 612514 | Modelo N°: 612514 | Descripcion Modelo N°: 612514 |
| 612514 | Modelo N°: 612514 | Descripcion Modelo N°: 612514 |
| 474559 | Modelo N°: 474559 | Descripcion Modelo N°: 474559 |

Diseño

- Decidimos separar la entidad Modelo de la entidad Medida para evitar redundancias en los atributos del sillón. Mientras que Modelo tiene la descripción del diseño del sillón, Medida almacena las dimensiones físicas (alto, ancho y profundidad), para evitar que se repitan los mismos datos en varios modelos. Esto también permite reutilizar medidas, y mantiene independientes las modificaciones sobre diseño estético y estructura física. Además, esto fue conveniente ya que cada medida está previamente cargada en el sistema, con un precio propio, al igual que los modelos.
- Creamos las entidades DetallePedido, DetalleFactura y DetalleCompra para descomponer estructuras repetitivas dentro de Pedido, Factura y Compra respectivamente. Esto permite modelar correctamente las relaciones de uno a muchos y facilita el cálculo de precios y subtotales.
- Creamos la entidad Material para representar de forma genérica los insumos utilizados dentro de los detalles de compras. De esta manera almacenamos los atributos comunes como *material_tipo*, *material_nombre*, *material_descripcion* y *material_precio* y evitamos repetirlos en múltiples lugares.
- Consideramos la posibilidad de no incluir los subtotales en las tablas ya que son valores calculables. Los conservamos porque forman parte de los datos de la tabla Maestra.
- Creamos las entidades Relleno, Tela y Madera que son los tipos de materiales disponibles. Al reutilizar el identificador de material como clave primaria logramos una jerarquía donde Relleno, Tela y Madera son “subtipos” de Material.

