

PROYECTO FINAL BASES DE DATOS AVANZADAS
“PRONTOMUEBLE”

Presentado por:

Miguel Ángel Molina Salas - 20162578008

Juan Eduardo Rodríguez Moncada - 20162578138

Presentado a:

Claudia Liliana Hernández García.

UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS
FACULTAD TECNOLOGICA
BASES DE DATOS AVANZADAS
BOGOTA D.C.

2018

1. DEFINICION DEL PROBLEMA

a. Descripción de las funcionalidades:

La aplicación funciona mediante perfiles de usuario, cada uno de los cuales podrá realizar tareas específicas y acceder a ciertas funcionalidades de la aplicación. Los perfiles son los siguientes: administrador y vendedor. El aplicativo está pensado en la gestión de un local físico, es decir, una aplicación de escritorio en vez de un aplicativo web, por lo cual no existe la posibilidad de realizar compras en línea. A continuación, se explica las tareas que cada uno de los perfiles de usuario puede hacer:

Tabla de funcionalidades

Funcionalidad	Descripción	¿Quién puede acceder a ella?
Registro de clientes	La aplicación debe permitir ingresar los datos correspondientes a un nuevo cliente y registrarlo en la base de datos. Al momento de ingresar los datos de un nuevo cliente, la aplicación debe permitir ingresar un primer número de teléfono, que será uno de los dos que podrá tener. Si se desea agregar un segundo número de teléfono, esta acción se podrá realizar desde la aplicación.	Vendedor
Registro de proveedores	La aplicación debe permitir ingresar los datos correspondientes a un nuevo proveedor y registrarlo en la base de datos. Al momento de ingresar los datos de un nuevo proveedor, la aplicación debe permitir ingresar un primer número de teléfono, que será uno de los dos que podrá tener. Si se desea agregar un segundo número de teléfono, esta acción se podrá realizar desde la aplicación. Así mismo, en el momento del registro, se podrá ingresar los datos correspondientes a la persona de contacto de ese proveedor.	Administrador
Registro de vendedores	La aplicación debe permitir ingresar los datos correspondientes a un nuevo vendedor y registrarlo en la base de datos.	Administrador
Registro de muebles	La aplicación debe permitir ingresar los datos correspondientes a un nuevo mueble, así como sus características y registrarlo en la base de datos. Se debe permitir elegir entre las características del mueble registradas en la base de datos, y en el caso de que alguna de estas no exista, se debe registrar automáticamente, como por ejemplo un nuevo color o material.	Administrador
Inactivar proveedores, vendedores y muebles	La aplicación no permite eliminar registros de la base de datos, debido al conflicto que esto genera si el campo de una tabla es llave foránea en otra. En vez de ello, la aplicación hace uso de los estados en las tablas, que pueden ser “activo” o “inactivo”. Mediante la aplicación se puede modificar el estado de los proveedores, vendedores y muebles. En el caso de que en alguna de estas tablas haya registros con el estado “inactivo”, este registro no se mostrará en la lista de registros de la aplicación y si en alguna tabla existe un registro con el estado “activo”, este registro sí se mostrará en la tabla de registros de la aplicación.	Administrador

Mostrar reportes mensuales	La aplicación debe permitir determinar cuál vendedor realizó más ventas en el mes, cuales son los clientes nuevos, cuales clientes realizaron las mayores compras y cuáles son los muebles más vendidos.	Administrador
Mostrar factura de venta	La aplicación debe mostrar los datos de cada venta, que son: ¿Quién realizó la venta?, ¿Quién hizo la compra?, ¿Qué artículo se compró?, la cantidad que se compró, la fecha en que se realizó la venta. Se mostrará de manera independiente los valores de los productos comprados, el precio de la instalación (si lo incluye) y por último el total de la venta. La aplicación solo mostrará la factura, mas no se podrá imprimir.	Vendedor
Realizar Venta	La aplicación debe permitir realizar las ventas y registrarlas en la base de datos.	Vendedor
Modificar datos del cliente	<p>La aplicación debe permitir modificar los datos de los clientes, exceptuando su número de identificación que, al momento de abrirse la ventana de modificación, aparecerá inhabilitado, y posteriormente actualizar estos datos en la base de datos. Solamente el administrador puede modificar la casilla del estado del cliente (activo/ inactivo). En el caso de que no recuerde el cliente que desea modificar, la ventana de modificación tiene un campo de búsqueda, la cual se puede realizar de acuerdo a dos cualquier campo de la tabla cliente, es decir, se puede realizar la búsqueda ya sea por “id”, “nombre” o demás campos.</p> <p>Así mismo, se proporciona la opción de agregar un segundo número de teléfono, adicional al ingresado al momento de registrar el cliente. Si se da el caso de querer agregar más números de teléfono, la aplicación mostrará una alerta, avisando de que dicha acción no es permitida.</p>	Vendedor, Administrador
Modificar datos de los muebles	La aplicación permite modificar todos los datos de los muebles, a excepción del “id”. Esto se hace en el caso de que haya habido un error en la digitación de la información, como una medida incorrecta, un color equivocado etc.	Administrador
Ver listado de clientes, proveedores y vendedores activos	La aplicación permite ver el listado de todos los clientes, proveedores y vendedores que siguen activos en la base de datos, es decir, que en su atributo “estado” tengan el valor “activo”. Los que tengan el valor “inactivo” no se mostrarán.	Administrador Vendedor

<p>Modificar datos de los proveedores.</p>	<p>La aplicación debe permitir modificar los datos de los proveedores, exceptuando su número de identificación que, al momento de abrirse la ventana de modificación, aparecerá inhabilitado, y posteriormente actualizar estos datos en la base de datos. Solamente el administrador puede modificar la casilla del estado del proveedor (activo/ inactivo). En el caso de que no recuerde el proveedor que desea modificar, la ventana de modificación tiene un campo de búsqueda, la cual se puede realizar de acuerdo a dos cualquier campo de la tabla proveedor, es decir, se puede realizar la búsqueda ya sea por “id”, “nombre” o demás campos.</p> <p>Así mismo, se proporciona la opción de agregar un segundo número de teléfono, adicional al ingresado al momento de registrar el cliente. Si se da el caso de querer agregar más números de teléfono, la aplicación mostrará una alerta, avisando de que dicha acción no es permitida. Otra opción que permite la aplicación en esta ventana es modificar y/o eliminar los datos de la persona(as) de contacto asociados al proveedor, como lo son sus datos generales y los números de teléfono que tenga asociados. Los números de teléfono si se podrán eliminar de la base de datos.</p>	<p>Administrador</p>
<p>Modificar datos de los vendedores.</p>	<p>La aplicación debe permitir al administrador modificar los datos de los vendedores, exceptuando su número de identificación que, al momento de abrirse la ventana de modificación, aparecerá inhabilitado.</p>	<p>Administrador</p>

b. Descripción del entorno de ejecución:

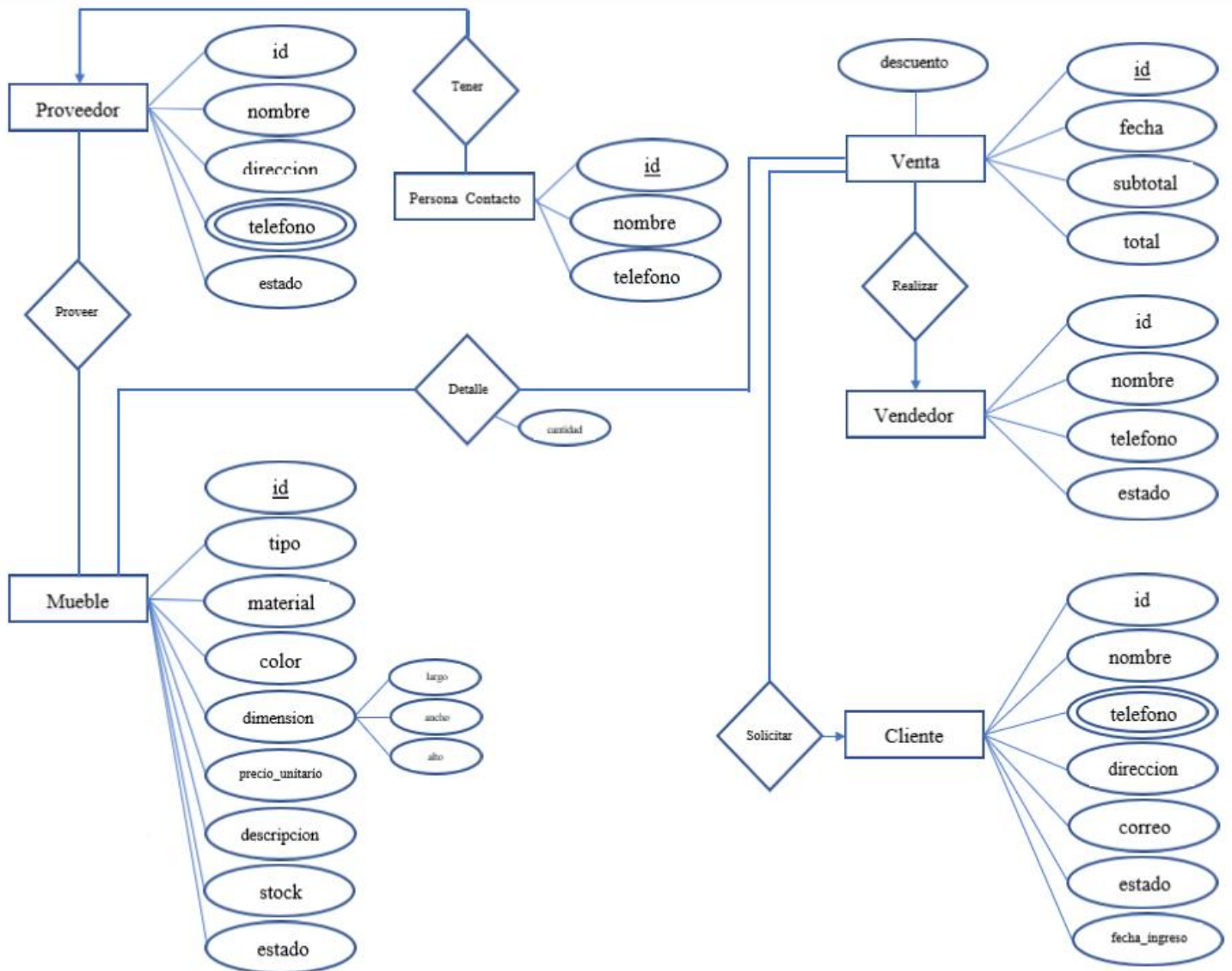
Para la realización del código fuente de la aplicación se utilizará el lenguaje de programación Java. Por otra parte, para la creación y pruebas de la base de datos se utilizará el motor de bases de datos PostgreSQL. Finalmente, la aplicación está pensada para el sistema operativo Windows.

- Sistema Operativo: Windows
- Motor de bases de datos: PostgreSQL
- Lenguaje de programación: Java.

c. Definición de las reglas de negocio:

- Se debe registrar para los proveedores la siguiente información: nombre, la dirección, números de teléfono y la persona de contacto.
- Un proveedor puede tener máximo dos números de teléfono.
- Un proveedor puede tener por lo menos una persona de contacto, de la cual se debe guardar su nombre y teléfono.
- Dos proveedores **no** pueden tener la misma persona de contacto, debido a que cada persona de contacto puede estar asociada a un único proveedor.
- Un proveedor puede proveer varios tipos de mueble a PRONTOMUEBLE, con lo cual un proveedor puede proveer uno o más muebles. Así mismo, un mueble lo puede proveer varios proveedores diferentes.
- Se debe registrar para los muebles la siguiente información: tipo de mueble, material en el cual está hecho, dimensiones (especificadas como largo, ancho y alto), color y precio. Además, se deben registrar la descripción y el stock (atributos adicionales).
- El id de los muebles identifica de manera general a todos aquellos muebles que comparten características en común como el color, material y tipo de mueble, por lo cual existe una cantidad disponible o stock de ese mueble en específico.
- Se debe registrar para los clientes la siguiente información: identificación, nombre, teléfonos de contacto, dirección de correspondencia y correo electrónico.
- Un cliente puede tener máximo dos números de teléfono.
- Se debe registrar para los vendedores la siguiente información: identificación, nombre y número de teléfono.
- Un vendedor puede vender varios tipos de muebles a uno o más clientes.
- Un cliente debe tener mínimo un número de teléfono.
- Un proveedor debe tener mínimo un número de teléfono.
- Al momento de realizar una venta, la aplicación **no** debe permitir ingresar fechas anteriores o posteriores a la fecha en que se está realizando (la aplicación traerá la fecha del sistema).
- La aplicación **no** permite la modificación del id de ninguna tabla, para ello lanza un mensaje de advertencia informando si está seguro de que la información ingresada es la correcta. Esto debido a que es la llave primaria de las tablas, en el caso de proveedores, clientes, vendedores, este id es el número de identificación (cedula, tarjeta de identidad, etc.).
- Un usuario puede acceder a muchas funcionalidades.
- Una misma funcionalidad la pueden tener varios usuarios.
- Un usuario tiene un solo perfil de usuario.

2. Diagrama Entidad - Relación



Aclaraciones sobre el diagrama:

En algunas entidades se agregaron atributos nuevos que no se mencionan en el enunciado del proyecto:

- **Entidad Proveedor:**

- **Atributo “estado”:** Se agrega el atributo “estado” debido a que no se pueden eliminar registros de la tabla porque se puede vulnerar la propiedad de integridad referencial.

- **Entidad Mueble:**

- **Atributo “descripción”:** Se agregó el atributo para que se pueda proporcionar una descripción más detallada de las características de un producto.
- **Atributo “stock”:** Este atributo se agregó para que se pueda conocer la cantidad de referencias con las que se cuenta de determinada referencia de mueble, debido a que el “id” de un mueble no identifica de manera única a un solo mueble, sino que, identifica a un grupo de muebles que compartan características en común.
- **Atributo “estado”:** Se agrega el atributo “estado” debido a que no se pueden eliminar registros de la tabla porque se puede vulnerar la propiedad de integridad referencial. Esto ayuda en el momento que se requiera discontinuar un mueble o no se cuente con existencias

- **Entidad Vendedor:**

- **Atributo “estado”:** Se agrega el atributo “estado” debido a que no se pueden eliminar registros de la tabla porque se puede vulnerar la propiedad de integridad referencial. Esto ayuda en el momento que se requiera cambiar de vendedores.
- **Atributo “fecha_ingreso”:** Este atributo se crea debido a que se requiere almacenar la fecha en la que un cliente ingresa al sistema, con esto más adelante se puede saber si es cliente antiguo o uno nuevo.

- **Entidad Detalle:**

- **Atributo “cantidad”:** Se crea este atributo debido a la necesidad que se tiene de almacenar la cantidad de unidades compradas por un cliente de determinada referencia de mueble.

- **Manejo de Usuarios:**

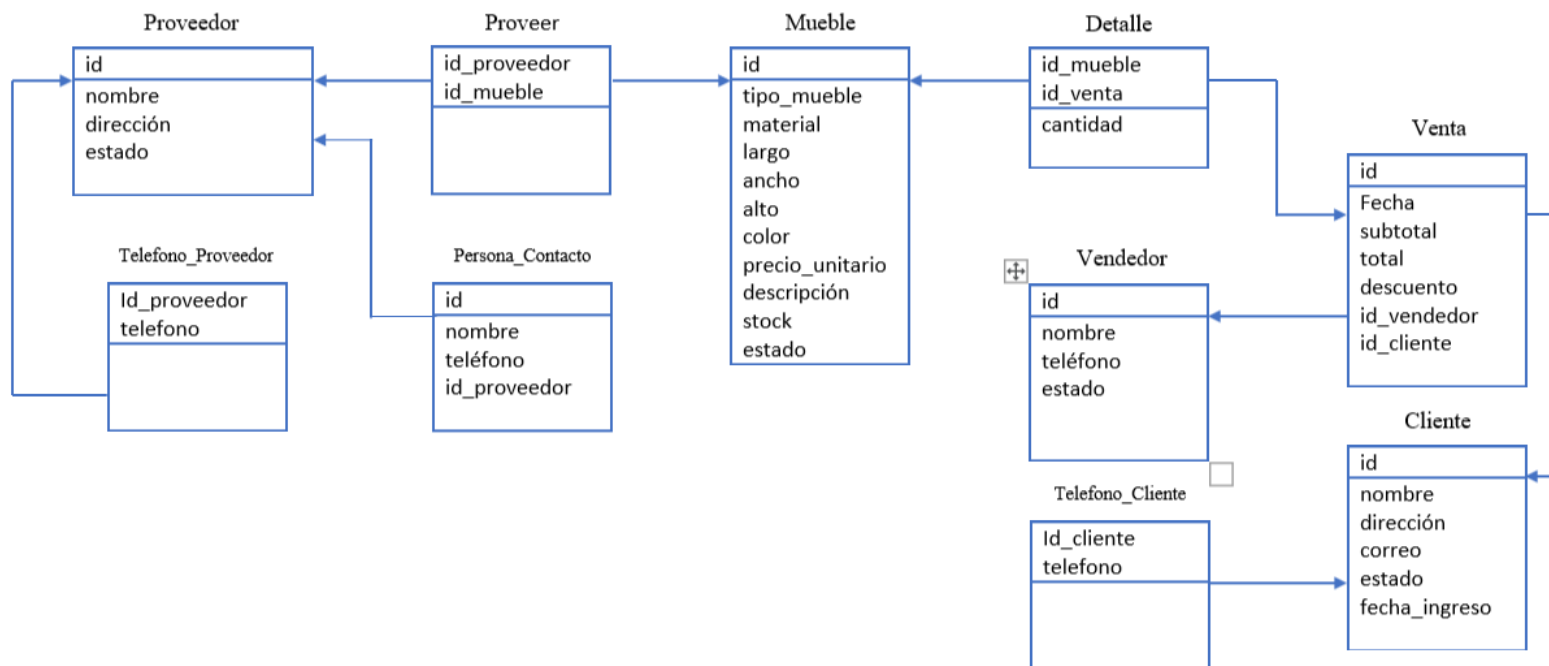
Solo existe un administrador y un vendedor registrados en la base de datos, de tal forma que el usuario “Admin” representa al administrador y el usuario “Vendedor” representa a todos los vendedores. Es por esta razón que las entidades están aisladas. Si en cambio un usuario tuviera que representar a una persona en especifica con sus datos personales, las entidades no estarían aisladas.

Por otro lado, la entidad “Funcionalidad” hace referencia a las funcionalidades o acciones que puede realizar el usuario, por ejemplo “Modificar datos del cliente”. Cada usuario podrá tener o acceder a ciertas funcionalidades, por ejemplo, el administrador podrá “ver reportes mensuales”, cosa que el vendedor no podrá. Cada funcionalidad se identifica con un numero único.

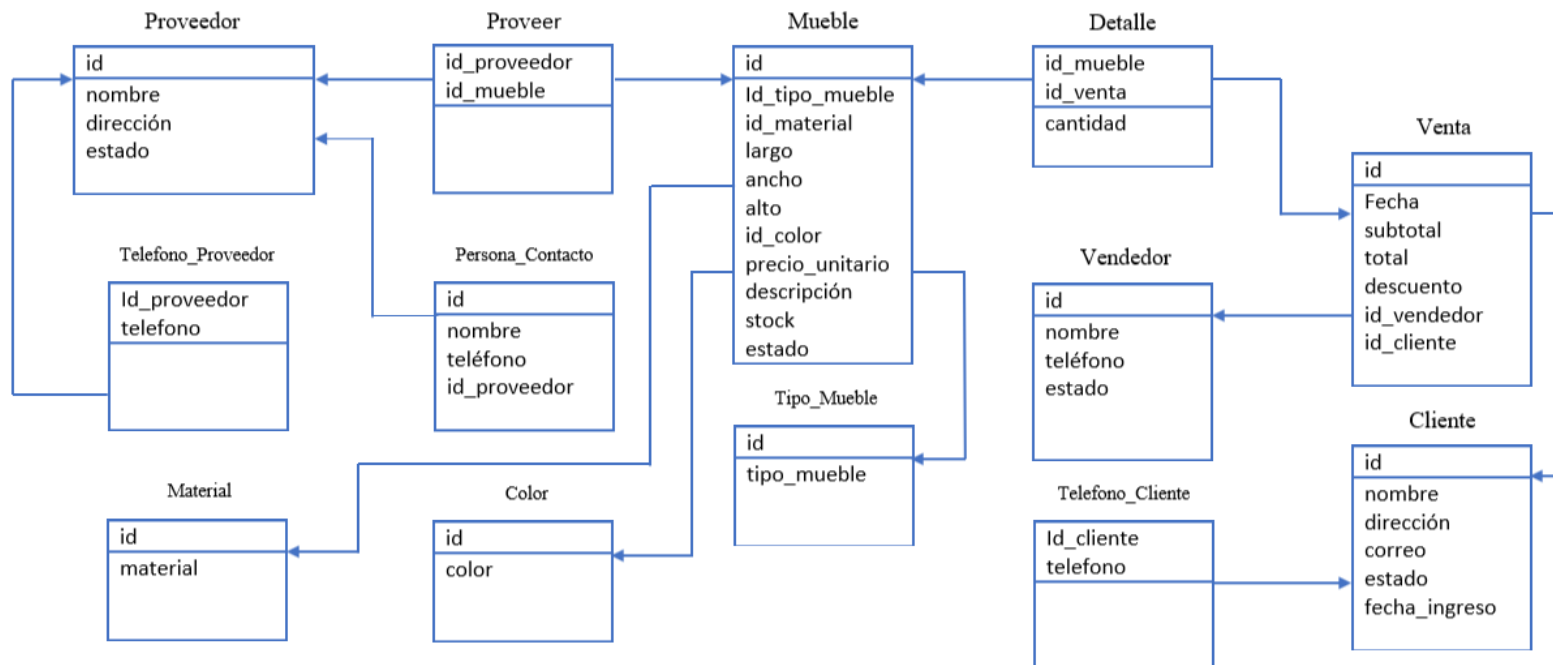
Dependiendo de las funcionalidades de cada usuario, la aplicación habilitará o no ciertas partes de la interfaz, bloqueando o activando ciertas opciones del menú según corresponda.

3. Diagrama Relacional

3.1. Diagrama Relacional (No Normalizado)



3.2. Diagrama Relacional (Normalizado)



Diferencias entre el modelo Relacional Normalizado y sin Normalizar

En el proceso de normalización, se crearon tres nuevas tablas. Son las siguientes:

- **Color:** Reemplaza al atributo “color” de la tabla “Mueble” en el modelo Relacional no normalizado.
- **Material:** Reemplaza al atributo “material” de la tabla “Mueble” en el modelo Relacional no normalizado.
- **Tipo_Mueble:** Reemplaza al atributo “tipo_mueble” de la tabla “Mueble” en el modelo Relacional no normalizado.

4. Diccionario de datos

4.1 Descripción de tablas y columnas

Tabla	Proveedor			
Descripción	Esta tabla almacena la información de los proveedores. Un proveedor hace referencia a una empresa, mas no a una persona en particular.			
Columnas				
Nombre	Tipo	PK	FK	Descripción
id	Integer	X		Código único de cada proveedor
nombre	Varchar(50)			Almacena la razón social del proveedor.
direccion	Varchar(25)			Almacena la dirección del proveedor.
estado	Varchar(15)			Almacena el estado en el que se encuentra el proveedor (activo o inactivo)

Tabla	Telefono_Proveedor			
Descripción	Esta tabla almacena los números de teléfono asociados a los proveedores.			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id_proveedor	Integer	X	X	Código único de teléfono cada proveedor
telefono	Varchar(50)	X		Almacena el número de teléfono del proveedor

Tabla	Persona_contacto			
Descripción	Esta tabla almacena la información de contacto de la persona encargada de proveer suministros			
Columnas				
Nombre	Tipo	PK	FK	Descripción
id	Integer	X		Código único de cada persona de contacto
nombre	Varchar(50)			Almacena el nombre del contacto
telefono	Varchar(15)			Almacena el número de teléfono del contacto
Id_proveedor	integer		X	Hace referencia al id del proveedor al que esta asociada la persona de contacto.

Tabla	material			
Descripción	Esta tabla almacena el material de elaboración de cada mueble			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	Integer	X		Numero único de cada material
Material	Varchar(20)			Almacena el nombre del material

Tabla	Color			
Descripción	Esta tabla almacena los colores que pueden tener los muebles			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	integer	X		Numero único de cada color
Color	Varchar(15)			Nombre del color

Tabla	Tipo_mueble			
Descripción	Esta tabla almacena los tipos de muebles que se tienen			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	Integer	X		Numero único de cada tipo de mueble
Tipo_mueble	Varchar(35)			Almacena el nombre del tipo de mueble

Tabla	Mueble			
Descripción	Esta tabla almacena los muebles que se tienen registrados para la venta			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	Integer	X		Numero único que referencia a un conjunto de muebles que tienen las mismas características
Id_tipo_mueble	Integer		X	Referencia el tipo de mueble al que pertenece
Id_material	Integer		X	Referencia al material del cual está compuesto el mueble
largo	Integer			Almacena la dimensión del mueble
ancho	integer			Almacena la dimensión del mueble
alto	integer			Almacena la dimensión del mueble
id_color	integer		X	Referencia del color del mueble
precio_unitario	integer			Almacena el costo del mueble por unidad
descripcion	varchar(100)			Almacena una breve descripción del articulo
stock	integer			Número actual de existencias del mueble
estado	Varchar(15)			Hace regencia a la disponibilidad del mueble (agotado o disponible)

Tabla	Proveer			
Descripción	Esta tabla almacena los muebles que provee cada proveedor			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id_proveedor	Integer	X	X	Referencia del proveedor
Id_mueble	Integer	X	X	Referencia del mueble

Tabla	Vendedor			
Descripción	Esta tabla almacena los datos de los vendedores			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	Integer	X		Numero único de cada vendedor
Nombre	Varchar(50)			Almacena el nombre del vendedor
Telefono	Varchar(15)			Almacena el número de teléfono
Estado	Varchar(15)			“activo” si el vendedor aún está trabajando en la empresa o “inactivo” en caso contrario

Tabla	Cliente			
Descripción	Almacena los datos de los clientes			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	Integer	X		Numero único de cada cliente
Nombre	Varchar(50)			Almacena el nombre del cliente
Dirección	Varchar(25)			Almacena la dirección del cliente
Correo	Varchar(50)			Almacena el correo del cliente
estado	varchar(15)			"Activo" o "Inactivo"
fecha_ingreso	date			Almacena la fecha de ingreso al sistema

Tabla	Telefono_cliente			
Descripción	Esta tabla almacena los números de teléfono de los clientes			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id_cliente	Integer	X	X	Referencia del cliente
telefono	Varchar(15)	X		Almacena el número de teléfono

Tabla	Venta			
Descripción	Esta tabla almacena los datos de cada venta realizada			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id	integer	x		Número único de venta
Fecha	date			Fecha en que se realiza la venta
subtotal	integer			Precio parcial de la venta (cantidad del producto a vender multiplicado por su precio unitario)
id_vendedor	integer		x	Referencia del vendedor que realiza la venta
id_cliente	integer		x	Referencia del cliente que realiza una compra
descuento	integer			Almacena el valor del descuento en la compra. Se aplica un descuento general del 10% a todas las ventas.
total_pagar	float			Almacena el total a pagar, restando el descuento y agregando el precio de instalación (si lo incluye)

Tabla	Detalle			
Descripción	Esta tabla almacena los muebles que se vendieron en cada venta.			
Columnas				
Nombre	Tipo	PK	FK	Descripción
Id_mueble	integer	X	X	Referencia del mueble vendido
Id_venta	Integer	X	X	Referencia de la venta realizada
cantidad	integer			Referencia la cantidad vendida de un mueble específico

4.2 Descripción de procedimientos y triggers

Trigger	descuento
Descripción	<p>Este trigger toma el valor de la compra y calcula el descuento del 10% que reciben los clientes por sus compras. Es decir, toma el dato que encuentra en el atributo “subtotal” en la tabla venta y le descuenta el 10%, el resultado de la operación se almacena en el atributo “descuento” de la tabla venta.</p> <p>Luego se calcula el total a pagar, utilizando nuevamente el atributo “subtotal” y restando “descuento”, el resultado de la operación se almacena en “total_pagar” de la misma tabla.</p> <p>Para finalizar, el trigger se asocia al evento de inserción en la tabla venta, es decir, siempre que se inserte una tupla en la tabla, se ejecutara esta instrucción automáticamente.</p>
Script	<pre>create or replace function descuento() returns trigger as \$descuento\$ begin NEW.descuento := NEW.valor * 0.10; NEW.total_pagar := NEW.valor - NEW.descuento; RETURN NEW; end \$descuento\$ LANGUAGE plpgsql; CREATE TRIGGER tr_descuento before insert ON venta for each row EXECUTE PROCEDURE descuento();</pre>

Procedimiento almacenado	consultar_telefono_prov
Descripción	Este procedimiento consulta y devuelve la cantidad de números de teléfono asociados a un proveedor.
Script	<pre>CREATE OR REPLACE FUNCTION consultar_telefono_prov(int) RETURNS bigint AS 'SELECT COUNT(telefono) FROM TELEFONO_PROVEEDOR where id_proveedor=\$1' LANGUAGE SQL;</pre>

Procedimiento almacenado	consultar_telefono_cliente
Descripción	Este procedimiento consulta y devuelve la cantidad de números de teléfono asociados a un cliente
Script	<pre>CREATE OR REPLACE FUNCTION consultar_telefono_cliente(int) RETURNS bigint AS 'SELECT COUNT(telefono) FROM TELEFONO_CLIENTE where id_cliente=\$1' LANGUAGE SQL;</pre>

4.3 Descripción de vistas

Vista	vista_venta_mueble
Descripción	Esta vista restringe los datos que puede visualizar el vendedor sobre un mueble en específico al momento de realizar una venta. Los campos que puede visualizar son id, descripción, precio_unitario, stock y estado.
Script	<pre>CREATE VIEW vista_venta_mueble AS SELECT id, descripcion, precio_unitario,stock, estado FROM mueble;</pre>