

Este documento presenta una base de datos relacional para un sistema de gestión de ventas, compras y stock. El objetivo principal es modelar la información necesaria para una empresa que comercializa productos, integrando áreas como ventas, logística, análisis comercial y abastecimiento.

La base de datos está diseñada para ser escalable, permitiendo incorporar nuevos módulos en el future, como múltiples depósitos/sucursales, reglas avanzadas de precio, tracking de envíos y reportes analíticos

### **OBJETIVO:**

El objetivo es que una organización pueda centralizar toda esta información en un sistema único y evitar inconsistencias, duplicación de datos o errores manuales.

Implementando el siguiente programa del ciclo comercial

- Gestión de clientes
- Registro de ventas
- Control de detalle de venta (producto–precio–cantidad)
- Gestión de vendedores y zonas
- Métodos de pago
- Envíos
- Gestión de proveedores
- Compras y detalle de compra
- Control de productos
- Stock mínimo y alertas de reposición
- Movimientos de stock (entrada/salida)
- Listas de precios y porcentajes de ganancia
- Depósitos de almacenamiento
- Clasificación por categorías

### **SITUACIÓN PROBLEMÁTICA:**

Muchas pequeñas y medianas empresas utilizan para gestionar sus procesos muchas hojas de cálculo separadas sin relación alguna, controles deficientes de stock, precios y dificultades para determinar el momento de comprar su mercadería, todo esto conlleva a generar pérdidas económicas y a no poder obtener reportes confiables para poder tomar buenas decisiones que ayuden al crecimiento del negocio.

### **MODELO DE NEGOCIO:**

La organización representada en este proyecto es una empresa dedicada a la comercialización de productos físicos, la cual opera tanto en canales presenciales como online. Su funcionamiento requiere la interacción coordinada de múltiples áreas: ventas, compras, logística, reposición y

análisis. La empresa adquiere productos a distintos proveedores y almacena su inventario en uno o varios depósitos propios, desde donde se gestionan posteriormente las entregas a clientes. El proceso comercial involucra vendedores asignados por zonas geográficas, que atienden a los clientes y concretan las ventas mediante distintos métodos de pago. Algunas ventas requieren además un servicio de envío, el cual también queda registrado en el sistema para garantizar la trazabilidad.

La administración del inventario es un aspecto crítico para el negocio. Cada producto pertenece a una categoría y posee un stock mínimo definido, lo que permite identificar cuándo es necesario reponerlo. Las compras realizadas a los proveedores incrementan el inventario, mientras que las ventas generan salidas de stock, quedando todo registrado a través de movimientos detallados. La empresa establece sus precios mediante listas que combinan el costo de adquisición con porcentajes de ganancia asociados a categorías o productos específicos. De esta forma puede definir estrategias comerciales, mantener márgenes adecuados y actualizar precios de manera ordenada.

### **LISTADO DE TABLAS:**

A continuación se describen las tablas del modelo, las claves principales, relaciones principales y notas de diseño. Este documento sirve guía la ejecución del script SQL incluido en el repositorio.

Cliente: cliente\_id PK, apellido, nombre, genero, direccion, email, fecha\_nacimiento, fecha\_alta, estado

Metodo\_pago: metodo\_pago\_id PK, tipo

Zona\_venta: zona\_id PK, sector

Proveedor: proveedor\_id PK, razon\_social, direccion, pais

Deposito: deposito\_id PK, direccion, tipo

Categoria: categoria\_id PK, tipo\_producto

Porcentaje\_gcia: porcentaje\_id PK, porcentaje, categoria\_id FK → categoria(categoria\_id)

Producto: producto\_id PK, nombre, stock\_minimo, categoria\_id FK → categoria(categoria\_id), reponer

Lista\_precio: lista\_id PK, producto\_id FK → producto(producto\_id), costo\_unitario, porcentaje\_id FK → porcentaje\_gcia(porcentaje\_id), precio\_unitario,

Vendedor: vendedor\_id PK, nombre, apellido, ranking, comision, zona\_id FK → zona\_venta(zona\_id)

Envio: envio\_id PK, costo\_envio, fecha\_envio

Venta: venta\_id PK, cliente\_id FK → cliente(cliente\_id), fecha\_venta, metodo\_pago\_id FK → metodo\_pago(metodo\_pago\_id), total, estado\_pago, vendedor\_id FK → vendedor(vendedor\_id), envio\_id FK → envio(envio\_id)

Detalle\_venta: detallev\_id PK, venta\_id FK → venta(venta\_id), producto\_id FK → producto(producto\_id), cantidad, precio\_unitario, subtotal

Compra: compra\_id PK, proveedor\_id FK → proveedor(proveedor\_id), fecha\_compra, total, estado\_pago

Detalle\_compra: detallec\_id PK, compra\_id FK → compra(compra\_id), producto\_id FK → producto(producto\_id), cantidad, costo\_unitario, subtotal

Movstock: movimiento\_id PK, producto\_id FK → producto(producto\_id), deposito\_id FK → deposito(deposito\_id), tipo\_mov, cantidad, fecha, compra\_id FK → compra(compra\_id), venta\_id FK → venta(venta\_id)

### **Algunas Relaciones**

Cliente 1:N Venta

Venta 1:N Detalle\_venta y 1:N Movstock

Producto 1:N Detalle\_venta y 1:N Compras y 1:N Movstock y 1:N Lista de precios

Proveedor 1:N Compra

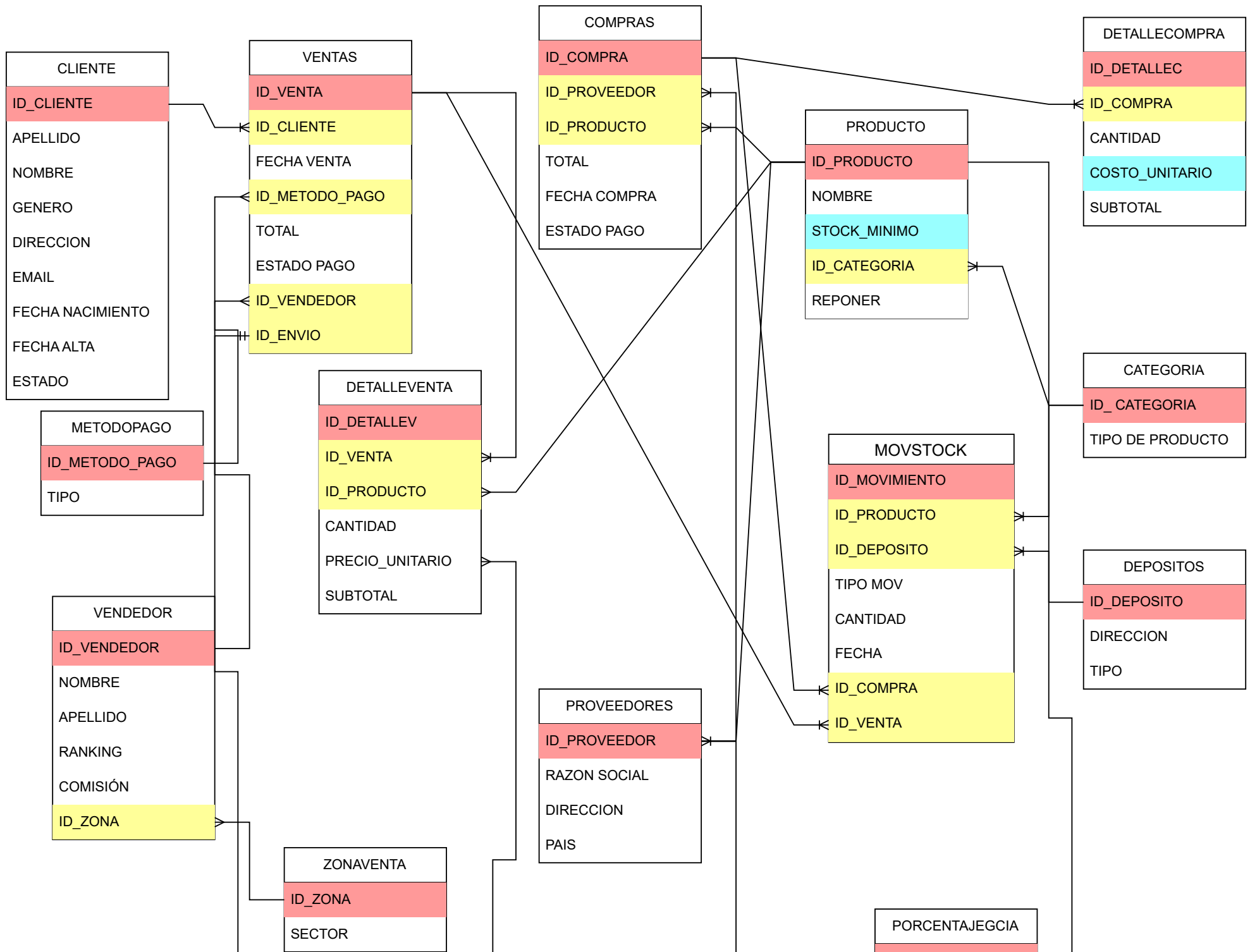
Compra 1:N Detalle\_compra 1:N Movstock

Categoria 1:N Producto y 1:N Porcentaje\_gcia

Envío 1: 1 Ventas

### **Script en SQL de creación de la base de datos y tablas**

[https://github.com/giselaocantos/PROYECTO-SISTEMA-VENTAS/blob/main/schema\\_sistema\\_ventas%20\(GISELAOCANTOS\).sql](https://github.com/giselaocantos/PROYECTO-SISTEMA-VENTAS/blob/main/schema_sistema_ventas%20(GISELAOCANTOS).sql)



ENVIO
ID_ENVIO
COSTO ENVIO
FECHA ENVIO

