

# **Modelado del Sistema – Gestión de Stock e Inventory (Bazar)**

## **1. Modelo de Datos (Tablas)**

El sistema utiliza 4 tablas principales.

---

### **1.1 Producto**

Atributos:

- **codigo\_producto** (PK)
- nombre
- marca
- precio
- cantidad\_disponible

Relaciones:

- 1 **Producto** → N **DetalleVenta**
- 

### **1.2 Cliente**

Atributos:

- **id\_cliente** (PK)
- nombre
- apellido
- dni

Relaciones:

- 1 **Cliente** → N **Venta**
- 

## 1.3 Venta

Atributos:

- **codigo\_venta** (PK)
- **fecha\_venta**
- **total**
- **id\_cliente** (FK → Cliente)

Relaciones:

- 1 **Venta** → N **DetalleVenta**
  - 1 **Venta** → 1 **Cliente**
- 

## 1.4 DetalleVenta

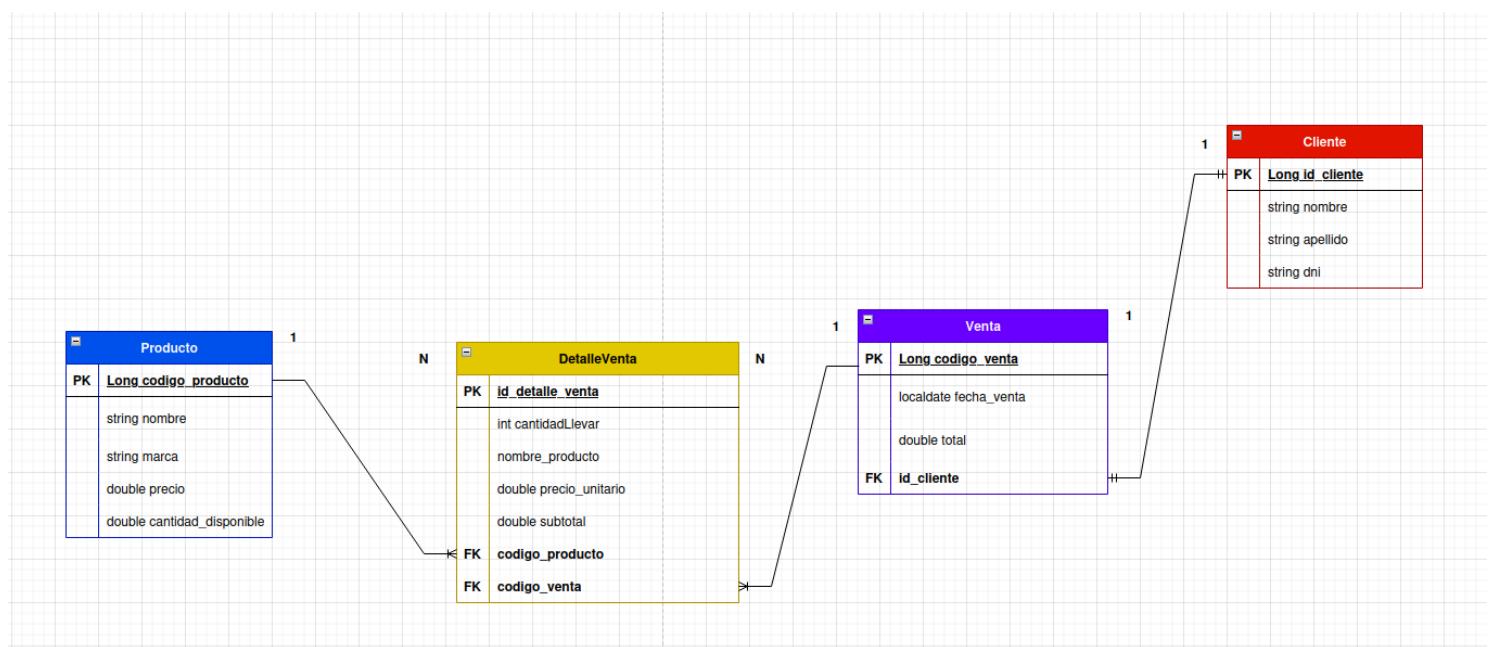
Atributos:

- **id\_detalle\_venta** (PK)
- **cantidadALlevar**
- **nombre\_producto**
- **precio\_unitario**
- **subtotal**
- **codigo\_producto** (FK → Producto)
- **codigo\_venta** (FK → Venta)

Relaciones:

- N DetalleVenta → 1 Producto
  - N DetalleVenta → 1 Venta
- 

## 2. Diagrama Entidad–Relación (Descripción textual)



Descripción:

- Un cliente puede registrar múltiples ventas.
  - Cada venta contiene uno o varios detalles de venta.
  - Cada detalle pertenece a un único producto.
  - Un producto puede aparecer en múltiples detalles de distintas ventas.
- 

## 3. Modelo de Clases (UML descriptivo)

## **Clase Producto**

- códigoProducto: String
- nombre: String
- marca: String
- precio: Double
- cantidadDisponible: Integer

Métodos clave:

- getters/setters
- 

## **Clase Cliente**

- idCliente: Long
- nombre: String
- apellido: String
- dni: String

Métodos clave:

- getters/setters
- 

## **Clase Venta**

- códigoVenta: String
- fechaVenta: LocalDate
- total: Double

- cliente: Cliente
- detalles: List

Métodos clave:

- getters/setters
- 

## Clase DetalleVenta

- idDetalleVenta: Long
- cantidadALlevar: Integer
- nombreProducto: String
- precioUnitario: Double
- subtotal: Double
- producto: Producto
- venta: Venta

Métodos clave:

- calcularSubtotal()
- 

## 4. DTOs Utilizados

### ProductoDTO

- id
- codigo\_venta;
- nombre\_producto;
- costo\_producto;

- cantidad\_llevar\_venta;
- subtotal\_venta;

## **ClienteDTO**

- codigo\_venta
- monto
- fecha\_venta
- cantidadProductos
- cantidadVentas;
- nombre\_cliente;
- apellido\_cliente

## **VentaDTO**

- idCliente
- detalleVentas: List

## **VentaReporteMontoCantidadDTO**

- id
  - monto\_total
  - cantidad\_ventas
- 

## **5. Reglas del Modelo**

- Una venta no puede existir sin cliente asociado.

- Un detalle debe pertenecer a una venta y un producto.
  - El subtotal se calcula en base a (`precioUnitario * cantidadALlavar`).
  - El total de una venta es la suma de los subtotales.
- 

## 6. Consideraciones

- No se permite actualizar/editar una venta.
  - Para evitar inconsistencias, el `nombre_producto` y `precio_unitario` se copian desde Producto al generar el detalle.
  - Se mantiene historial de precios y nombres usado en ventas.
-