

Documentación del Proyecto

Punto LAN

Introducción:

Punto LAN es un emprendimiento dedicado a la venta de productos tecnológicos y accesorios. Para mejorar la gestión de los productos, ventas y clientes, se ha desarrollado una base de datos que automatiza los procesos administrativos y comerciales, asegurando una mejor experiencia para clientes y gestores del negocio.

Objetivo:

El objetivo de esta base de datos es gestionar eficientemente la información de Punto LAN, cubriendo aspectos clave como el control de inventario, ventas, gestión de clientes, información contable y logística. Esto permitirá mejorar la toma de decisiones mediante el análisis de datos en tiempo real.

Situación Problemática:

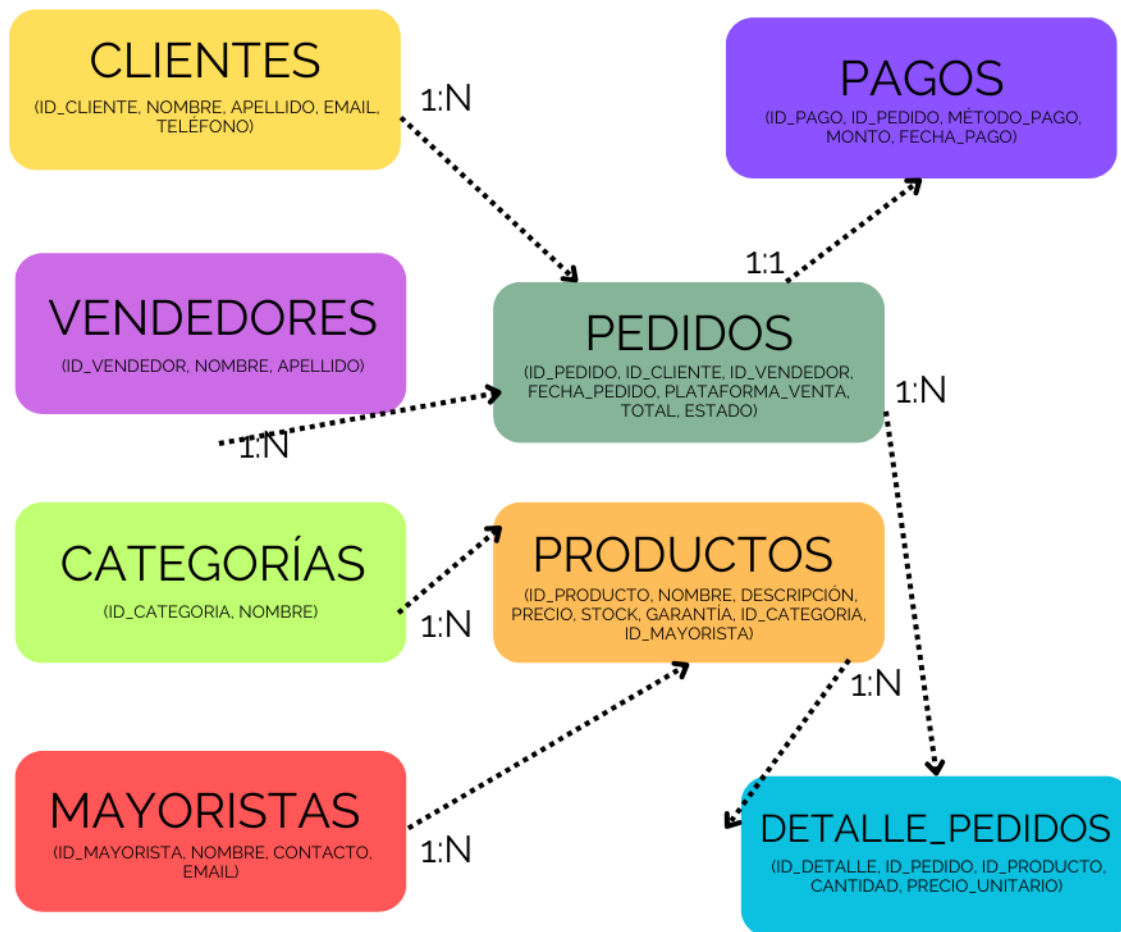
Punto LAN enfrenta dificultades en el manejo de stock, seguimiento de ventas y control de garantías. La falta de un sistema centralizado provoca errores en los registros de productos y pedidos, afectando la eficiencia operativa. La implementación de esta base de datos permitirá solucionar estos problemas optimizando la gestión de la información.

Modelo de Negocio:

Punto LAN opera como un comercio electrónico de productos tecnológicos. A través de distintas plataformas de venta (Instagram, Facebook y tienda web), se generan pedidos que son gestionados por vendedores. Los productos provienen de distintos mayoristas y cuentan con garantía, asegurando la calidad del servicio.

Diagrama Entidad-Relación (DER):

A continuación, se presenta el diagrama E-R que representa la estructura de la base de datos. Este diagrama muestra las entidades de la base de datos y sus relaciones.



Listado de Tablas:

Las entidades de la base de datos incluyen: Categorías, Mayoristas, Productos, Clientes, Vendedores, Pedidos, Detalle de Pedidos y Pagos. Cada tabla contiene claves primarias y foráneas para mantener la integridad referencial.

Campo	Tipo de Dato	Restricciones
id_categoria	INT	PRIMARY KEY
nombre	VARCHAR(50)	NOT NULL

Campo	Tipo de Dato	Restricciones
id_mayorista	INT	PRIMARY KEY
nombre	VARCHAR(100)	NOT NULL
contacto	VARCHAR(50)	
email	VARCHAR(100)	UNIQUE

Campo	Tipo de Dato	Restricciones
id_producto	INT	PRIMARY KEY
nombre	VARCHAR(100)	NOT NULL
descripcion	TEXT	
precio	DECIMAL(10,2)	NOT NULL
stock	INT	NOT NULL
garantia	VARCHAR(50)	
id_categoria	INT	FOREIGN KEY
id_mayorista	INT	FOREIGN KEY

Campo	Tipo de Dato	Restricciones
id_cliente	INT	PRIMARY KEY
nombre	VARCHAR(50)	NOT NULL
apellido	VARCHAR(50)	NOT NULL
email	VARCHAR(100)	UNIQUE NOT NULL
telefono	VARCHAR(15)	

Campo	Tipo de Dato	Restricciones
id_vendedor	INT	PRIMARY KEY
nombre	VARCHAR(50)	NOT NULL
apellido	VARCHAR(50)	NOT NULL

Campo	Tipo de Dato	Restricciones
id_pedido	INT	PRIMARY KEY
id_cliente	INT	FOREIGN KEY
id_vendedor	INT	FOREIGN KEY
fecha_pedido	DATETIME	DEFAULT CURRENT_TIMESTAMP
plataforma_venta	ENUM('Instagram', 'Facebook', 'Web')	NOT NULL
total	DECIMAL(10,2)	NOT NULL
estado	ENUM('Pendiente', 'Pagado', 'Cancelado')	DEFAULT 'Pendiente'

Campo	Tipo de Dato	Restricciones
id_detalle	INT	PRIMARY KEY
id_pedido	INT	FOREIGN KEY
id_producto	INT	FOREIGN KEY
cantidad	INT	NOT NULL
precio_unitario	DECIMAL(10,2)	NOT NULL

Campo	Tipo de Dato	Restricciones
id_pago	INT	PRIMARY KEY
id_pedido	INT	FOREIGN KEY
metodo_pago	ENUM('Efectivo', 'Tarjeta', 'Transferencia')	NOT NULL
monto	DECIMAL(10,2)	NOT NULL
fecha_pago	DATETIME	DEFAULT CURRENT_TIMESTAMP

Descripción de las tablas:

Categorías:

Contiene información sobre las categorías de productos (id_categoria, nombre).

Mayoristas:

Almacena datos sobre los mayoristas (id_mayorista, nombre, contacto, email).

Productos:

Incluye los productos vendidos (id_producto, nombre, descripción, precio, stock, garantía, id_categoria, id_mayorista).

Clientes:

Registra a los clientes (id_cliente, nombre, apellido, email, teléfono).

Vendedores:

Datos de los vendedores (id_vendedor, nombre, apellido).

Pedidos:

Información sobre los pedidos (id_pedido, id_cliente, id_vendedor, fecha_pedido, plataforma_venta, total, estado).

Detalle de Pedidos:

Detalle de los productos en cada pedido (id_detalle, id_pedido, id_producto, cantidad, precio_unitario).

Pagos:

Información sobre los pagos realizados (id_pago, id_pedido, metodo_pago, monto, fecha_pago).

Script SQL de Creación:

El siguiente script SQL crea la base de datos y sus tablas, estableciendo las relaciones necesarias.

```
CREATE DATABASE punto_lan; USE punto_lan;
```

```
CREATE TABLE Categorías ( id_categoria INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(50) NOT NULL );
```

```
CREATE TABLE Mayoristas ( id_mayorista INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(100) NOT NULL, contacto VARCHAR(50), email VARCHAR(100) UNIQUE );
```

```
CREATE TABLE Productos ( id_producto INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(100) NOT NULL, descripcion TEXT, precio DECIMAL(10,2) NOT NULL, stock INT NOT NULL, garantia VARCHAR(50), id_categoria INT, id_mayorista INT, FOREIGN KEY (id_categoria) REFERENCES Categorias(id_categoria) ON DELETE SET NULL, FOREIGN KEY (id_mayorista) REFERENCES Mayoristas(id_mayorista) ON DELETE SET NULL );
```

```
CREATE TABLE Clientes ( id_cliente INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(50) NOT NULL, apellido VARCHAR(50) NOT NULL, email VARCHAR(100) UNIQUE NOT NULL, telefono VARCHAR(15) );
```

```
CREATE TABLE Vendedores ( id_vendedor INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(50) NOT NULL, apellido VARCHAR(50) NOT NULL );
```

```
CREATE TABLE Pedidos ( id_pedido INT AUTO_INCREMENT PRIMARY KEY, id_cliente INT NOT NULL, id_vendedor INT NOT NULL, fecha_pedido DATETIME DEFAULT CURRENT_TIMESTAMP, plataforma_venta ENUM('Instagram', 'Facebook', 'Web') NOT NULL, total DECIMAL(10,2) NOT NULL, estado ENUM('Pendiente', 'Pagado', 'Cancelado') NOT NULL DEFAULT 'Pendiente', FOREIGN KEY (id_cliente) REFERENCES Clientes(id_cliente) ON DELETE CASCADE, FOREIGN KEY (id_vendedor) REFERENCES Vendedores(id_vendedor) ON DELETE SET NULL );
```

```
CREATE TABLE Detalle_Pedidos ( id_detalle INT AUTO_INCREMENT PRIMARY KEY, id_pedido INT NOT NULL, id_producto INT NOT NULL, cantidad INT NOT NULL, precio_unitario DECIMAL(10,2) NOT NULL, FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido) ON DELETE CASCADE, FOREIGN KEY (id_producto) REFERENCES Productos(id_producto) ON DELETE CASCADE );
```

```
CREATE TABLE Pagos ( id_pago INT AUTO_INCREMENT PRIMARY KEY, id_pedido INT NOT NULL, metodo_pago ENUM('Efectivo', 'Tarjeta', 'Transferencia') NOT NULL, monto DECIMAL(10,2) NOT NULL, fecha_pago DATETIME DEFAULT CURRENT_TIMESTAMP, FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido) ON DELETE CASCADE );
```