

## PROYECTO FINAL – BASE DE DATOS E-COMMERCE

**Alumno:** Juan Hernández

**Curso:** SQL

**Base de datos:** ecommerce\_db

### Sumario

1. Introducción
  2. Objetivo
  3. Situación problemática
  4. Modelo de negocio
  5. Diagrama Entidad-Relación (E-R)
  6. Listado de tablas
  7. Scripts de creación de la base de datos (Objetos)
  8. Scripts de inserción de datos
  9. Informes generados
  10. Herramientas y tecnologías utilizadas
- 

### Introducción

El presente proyecto consiste en el diseño e implementación de una base de datos relacional para un sistema de comercio electrónico.

El objetivo principal es modelar un escenario realista de e-commerce contemplando usuarios, productos, pedidos, pagos, promociones, envíos y devoluciones, asegurando integridad referencial y automatizando reglas de negocio desde la base de datos.

La solución fue desarrollada utilizando MySQL 8, incorporando objetos avanzados como vistas, funciones, triggers y procedimientos almacenados. Además, se incluye una tabla de hechos orientada a analítica para facilitar reportes de ventas y comportamiento de clientes.

---

### Objetivo

Crear una base de datos que permita:

- Gestionar usuarios y múltiples direcciones de envío.
- Administrar productos, categorías y promociones.
- Registrar pedidos y sus ítems.
- Controlar stock automáticamente ante operaciones del carrito.
- Gestionar pagos, incluyendo devoluciones.
- Automatizar reglas de negocio críticas (stock, totales, auditoría).
- Generar información para análisis contable, logístico y comercial.

El modelo está pensado para operar como backend de una aplicación de comercio electrónico y también para soportar reportes analíticos.

---

## Situación problemática

En un sistema de e-commerce, la ausencia de una base de datos bien estructurada genera problemas como:

- Ventas con stock insuficiente (sobreventa).
- Totales de pedidos incorrectos por cálculos manuales o inconsistentes.
- Inconsistencias en el ciclo de vida del pedido (sin trazabilidad de estados).
- Pagos o devoluciones no vinculadas correctamente.
- Dificultad para generar reportes confiables.
- Dependencia excesiva de la lógica del backend para validaciones críticas.

Este proyecto busca resolver estas brechas delegando validaciones y automatizaciones al motor de base de datos, reduciendo errores y asegurando consistencia.

---

## Modelo de negocio

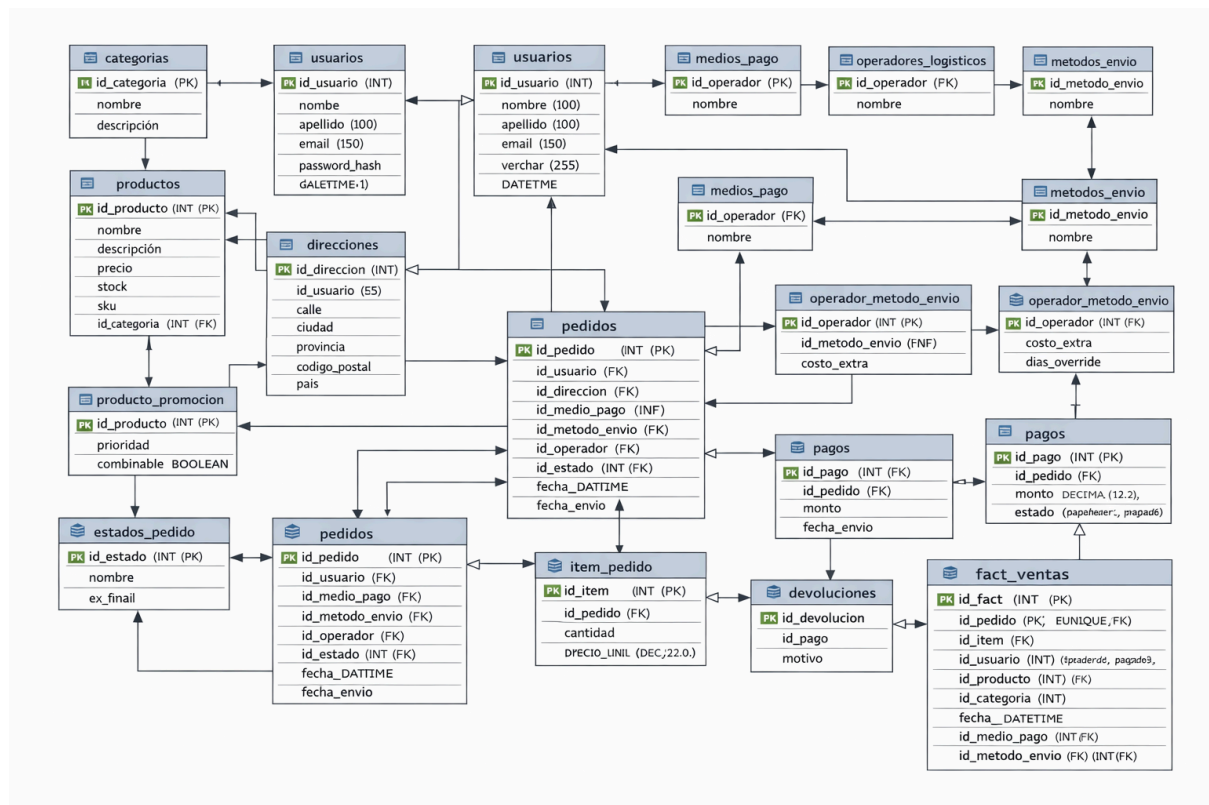
El modelo representa una tienda online que comercializa productos físicos. Los clientes pueden:

- Registrarse y gestionar múltiples direcciones.
- Comprar productos categorizados.
- Acceder a promociones activas.
- Elegir medios de pago y métodos de envío.
- Generar pedidos, confirmarlos, pagarlos y despacharlos mediante operadores logísticos.
- Solicitar devoluciones sobre pagos realizados.

La organización requiere controlar inventario, registrar ventas, auditar estados del pedido, analizar comportamiento de clientes y medir indicadores (ventas por usuario, por categoría, por método de envío, etc.).

---

## Diagrama Entidad-Relación (E-R)



## Listado de tablas

## categorias

| Columna      | Tipo        | Clave  | Descripción                   |
|--------------|-------------|--------|-------------------------------|
| id_categoria | INT         | PK     | Identificador de la categoría |
| nombre       | VARCHAR(80) | UNIQUE | Nombre de la categoría        |
| descripcion  | TEXT        |        | Descripción                   |

### promociones

| Columna      | Tipo          | Clave | Descripción                   |
|--------------|---------------|-------|-------------------------------|
| id_promocion | INT           | PK    | Identificador de promoción    |
| nombre       | VARCHAR(100)  |       | Nombre de la promoción        |
| descripcion  | TEXT          |       | Descripción                   |
| tipo         | ENUM          |       | PORCENTAJE o FIJO             |
| valor        | DECIMAL(10,2) |       | Valor del descuento           |
| fecha_inicio | DATE          |       | Inicio de vigencia (nullable) |
| fecha_fin    | DATE          |       | Fin de vigencia (nullable)    |
| activa       | BOOLEAN       |       | Flag de activación            |

### medios\_pago

| Columna       | Tipo        | Clave  | Descripción             |
|---------------|-------------|--------|-------------------------|
| id_medio_pago | INT         | PK     | Identificador del medio |
| nombre        | VARCHAR(50) | UNIQUE | Nombre del medio        |
| descripcion   | TEXT        |        | Descripción             |

### operadores\_logisticos

| Columna     | Tipo        | Clave  | Descripción                  |
|-------------|-------------|--------|------------------------------|
| id_operador | INT         | PK     | Identificador del operador   |
| nombre      | VARCHAR(80) | UNIQUE | Nombre del operador (correo) |
| descripcion | TEXT        |        | Descripción                  |
| activo      | BOOLEAN     |        | Estado activo                |

### metodos\_envio

| Columna         | Tipo          | Clave  | Descripción              |
|-----------------|---------------|--------|--------------------------|
| id_metodo_envio | INT           | PK     | Identificador del método |
| nombre          | VARCHAR(50)   | UNIQUE | Nombre del método        |
| costo           | DECIMAL(10,2) |        | Costo base de envío      |
| dias_estimados  | INT           |        | Días estimados base      |
| descripcion     | TEXT          |        | Descripción              |

### operador\_metodo\_envio

| Columna                 | Tipo          | Clave | Descripción                  |
|-------------------------|---------------|-------|------------------------------|
| id_operador             | INT           | PK/FK | Operador logístico           |
| id_metodo_envio         | INT           | PK/FK | Método de envío              |
| costo_extra             | DECIMAL(10,2) |       | Costo adicional del operador |
| dias_estimados_override | INT           |       | Override de días estimados   |

### usuarios

| Columna        | Tipo         | Clave  | Descripción               |
|----------------|--------------|--------|---------------------------|
| id_usuario     | INT          | PK     | Identificador del usuario |
| nombre         | VARCHAR(100) |        | Nombre                    |
| apellido       | VARCHAR(100) |        | Apellido                  |
| email          | VARCHAR(150) | UNIQUE | Email                     |
| password_hash  | VARCHAR(255) |        | Contraseña hasheada       |
| fecha_creacion | DATETIME     |        | Fecha alta                |

### **direcciones**

| Columna       | Tipo         | Clave | Descripción             |
|---------------|--------------|-------|-------------------------|
| id_direccion  | INT          | PK    | Identificador dirección |
| id_usuario    | INT          | FK    | Usuario dueño           |
| calle         | VARCHAR(150) |       | Calle                   |
| ciudad        | VARCHAR(80)  |       | Ciudad                  |
| provincia     | VARCHAR(80)  |       | Provincia               |
| codigo_postal | VARCHAR(20)  |       | CP                      |
| pais          | VARCHAR(80)  |       | País                    |

### **productos**

| Columna      | Tipo          | Clave  | Descripción                |
|--------------|---------------|--------|----------------------------|
| id_producto  | INT           | PK     | Identificador del producto |
| nombre       | VARCHAR(150)  |        | Nombre                     |
| descripcion  | TEXT          |        | Descripción                |
| precio       | DECIMAL(10,2) |        | Precio base                |
| stock        | INT           |        | Stock disponible           |
| sku          | VARCHAR(50)   | UNIQUE | Código único               |
| id_categoria | INT           | FK     | Categoría                  |

### **producto\_promocion**

| Columna      | Tipo    | Clave | Descripción             |
|--------------|---------|-------|-------------------------|
| id_producto  | INT     | PK/FK | Producto asociado       |
| id_promocion | INT     | PK/FK | Promoción asociada      |
| prioridad    | INT     |       | Prioridad de aplicación |
| combinable   | BOOLEAN |       | Indica si combina       |

**estados\_pedido**

| Columna     | Tipo         | Clave  | Descripción                        |
|-------------|--------------|--------|------------------------------------|
| id_estado   | INT          | PK     | Identificador del estado           |
| nombre      | VARCHAR(30)  | UNIQUE | Nombre (carrito, confirmado, etc.) |
| descripcion | VARCHAR(150) |        | Descripción                        |
| es_final    | BOOLEAN      |        | Marca si es estado final           |

**pedidos**

| Columna         | Tipo          | Clave | Descripción                   |
|-----------------|---------------|-------|-------------------------------|
| id_pedido       | INT           | PK    | Identificador del pedido      |
| id_usuario      | INT           | FK    | Usuario                       |
| id_direccion    | INT           | FK    | Dirección envío               |
| id_medio_pago   | INT           | FK    | Medio de pago                 |
| id_metodo_envio | INT           | FK    | Método envío                  |
| id_operador     | INT           | FK    | Operador logístico (nullable) |
| id_estado       | INT           | FK    | Estado normalizado            |
| total           | DECIMAL(12,2) |       | Total calculado               |
| fecha           | DATETIME      |       | Fecha creación                |
| fecha_envio     | DATETIME      |       | Fecha despacho (nullable)     |

**item\_pedido**

| Columna     | Tipo          | Clave | Descripción              |
|-------------|---------------|-------|--------------------------|
| id_item     | INT           | PK    | Identificador del ítem   |
| id_pedido   | INT           | FK    | Pedido                   |
| id_producto | INT           | FK    | Producto                 |
| cantidad    | INT           |       | Cantidad                 |
| precio_unit | DECIMAL(10,2) |       | Precio al momento compra |

### **pagos**

| Columna   | Tipo          | Clave | Descripción                         |
|-----------|---------------|-------|-------------------------------------|
| id_pago   | INT           | PK    | Identificador del pago              |
| id_pedido | INT           | FK    | Pedido                              |
| monto     | DECIMAL(12,2) |       | Monto                               |
| estado    | ENUM          |       | pendiente/pagado/rechazado/devuelto |
| fecha     | DATETIME      |       | Fecha                               |

### **devoluciones**

| Columna       | Tipo     | Clave | Descripción              |
|---------------|----------|-------|--------------------------|
| id_devolucion | INT      | PK    | Identificador devolución |
| id_pago       | INT      | FK    | Pago asociado            |
| motivo        | TEXT     |       | Motivo                   |
| fecha         | DATETIME |       | Fecha devolución         |

### **historial\_estado\_pedido**

| Columna            | Tipo     | Clave | Descripción                 |
|--------------------|----------|-------|-----------------------------|
| id_historial       | INT      | PK    | Identificador del historial |
| id_pedido          | INT      | FK    | Pedido                      |
| id_estado_anterior | INT      | FK    | Estado anterior             |
| id_estado_nuevo    | INT      | FK    | Estado nuevo                |
| fecha              | DATETIME |       | Fecha cambio                |



### fact\_ventas .

| Columna         | Tipo          | Clave     | Descripción            |
|-----------------|---------------|-----------|------------------------|
| id_fact         | INT           | PK        | Identificador hecho    |
| id_pedido       | INT           | FK        | Pedido                 |
| id_item         | INT           | UNIQUE/FK | Ítem asociado          |
| id_usuario      | INT           | FK        | Usuario                |
| id_producto     | INT           | FK        | Producto               |
| id_categoria    | INT           | FK        | Categoría              |
| fecha           | DATETIME      |           | Fecha del pedido       |
| cantidad        | INT           |           | Cantidad               |
| precio_unit     | DECIMAL(10,2) |           | Precio unitario        |
| subtotal        | DECIMAL(12,2) |           | cantidad * precio_unit |
| id_medio_pago   | INT           | FK        | Medio de pago          |
| id_metodo_envio | INT           | FK        | Método envío           |
| id_operador     | INT           | FK        | Operador (nullable)    |

## Scripts de creación de la base de datos

Enlace al [script](#)

### Vistas

Vista 1: vw\_ventas\_por\_usuario

Descripción: consolida usuarios y pedidos para obtener cantidad de pedidos y total gastado por usuario.

Ejemplo: “Juan Pérez – pedidos: 3 – total gastado: \$520.000”

Objetivo: análisis comercial (clientes frecuentes / mayor facturación).

Tablas: usuarios, pedidos.

Vista 2: vw\_productos\_stock\_bajo

Descripción: lista productos con stock < 10.

Ejemplo: "Mouse Gamer – stock: 5"

Objetivo: reposición de inventario y prevención de quiebre de stock.

Tablas: productos.

#### Vista 3: vw\_detalle\_pedidos

Descripción: detalle operativo por pedido con estado normalizado y subtotal por ítem.

Ejemplo: "Pedido 3 – cliente: ana@mail.com – estado: enviado – producto: Notebook i5 – cantidad: 1 – subtotal: \$650.000"

Objetivo: soporte administrativo y atención al cliente.

Tablas: pedidos, usuarios, estados\_pedido, item\_pedido, productos.

#### Vista 4: vw\_promociones\_activas

Descripción: lista promociones vigentes según flag y rango de fechas.

Ejemplo: "Black Friday – 20%"

Objetivo: control de campañas y análisis comercial.

Tablas: promociones.

#### Vista 5: vw\_envios

Descripción: muestra pedido + método de envío + costo + operador + días estimados (función).

Ejemplo: "Pedido 9 – método: Express – operador: correo3 – días estimados: 2"

Objetivo: análisis logístico y SLA de entregas.

Tablas: pedidos, metodos\_envio, operadores\_logisticos, operador\_metodo\_envio (vía función).

### Funciones

#### Función 1: fn\_total\_pedido(id\_pedido)

Descripción: suma cantidad \* precio\_unit de item\_pedido para el pedido indicado.

Objetivo: centralizar el cálculo de totales.

Tablas: item\_pedido.

#### Función 2: fn\_total\_pedido\_con\_envio(id\_pedido)

Descripción: total del pedido + costo del método de envío.

Objetivo: obtener el monto final a pagar incluyendo logística.

Tablas: item\_pedido, pedidos, metodos\_envio.

#### Función 3: fn\_stock\_disponible(id\_producto)

Descripción: devuelve el stock actual del producto.

Objetivo: reutilización y validación consistente en triggers/procesos.

Tablas: productos.

#### Función 4: fn\_promocion\_activa(id\_promocion)

Descripción: valida si una promoción está activa por flag + fechas.  
Objetivo: evitar aplicar promociones fuera de vigencia.  
Tablas: promociones.

**Función 5: fn\_producto\_en\_promocion(id\_producto)**

Descripción: devuelve TRUE/FALSE si el producto posee una promo activa.  
Objetivo: facilitar lógica comercial y reportes.  
Tablas: producto\_promocion, promociones.

**Función 6: fn\_envio\_estimado\_pedido(id\_pedido)**

Descripción: devuelve días estimados considerando método de envío y posible override del operador.  
Objetivo: estimación logística en reportes y vistas.  
Tablas: pedidos, metodos\_envio, operador\_metodo\_envio.

### **Stored Procedures**

**SP 1: sp\_crear\_pedido(id\_usuario, id\_direccion, id\_medio\_pago, id\_metodo\_envio)**

Descripción: crea un pedido en estado “carrito”.  
Objetivo: estandarizar creación del pedido y asegurar estado inicial válido.  
Tablas: pedidos, estados\_pedido.

**SP 2: sp\_agregar\_item\_pedido(id\_pedido, id\_producto, cantidad)**

Descripción: agrega un ítem tomando precio vigente del producto.  
Objetivo: estandarizar carga y disparar triggers (stock/total/hechos).  
Tablas: productos, item\_pedido (+ triggers en productos, pedidos, fact\_ventas).

**SP 3: sp\_quitar\_item\_pedido(id\_pedido, id\_producto)**

Descripción: elimina un ítem del pedido.  
Objetivo: permitir edición del carrito y recomputar total (por trigger).  
Tablas: item\_pedido (+ trigger que devuelve stock y recalcula total).

**SP 4: sp\_confirmar\_pedido(id\_pedido)**

Descripción: cambia el estado del pedido a “confirmado”.  
Objetivo: formalizar la transición del flujo de compra.  
Tablas: pedidos, estados\_pedido.

**SP 5: sp\_enviar\_pedido(id\_pedido, id\_operador)**

Descripción: asigna operador logístico, fecha de envío y cambia estado a “enviado”.  
Objetivo: soportar gestión logística y trazabilidad del despacho.  
Tablas: pedidos, operadores\_logisticos, estados\_pedido.

**SP 6: sp\_registrar\_devolucion(id\_pago, motivo)**

Descripción: inserta devolución y marca el pago como “devuelto”.  
Objetivo: estandarizar devoluciones y mantener consistencia.  
Tablas: devoluciones, pagos.

### **Triggers**

Trigger 1: trg\_control\_stock

Descripción: antes de insertar un ítem valida stock y descuenta inventario.  
Objetivo: evitar sobreventa y automatizar inventario.  
Evento: BEFORE INSERT en item\_pedido.  
Tablas afectadas: productos.

Trigger 2: trg\_actualizar\_total\_ai

Descripción: luego de insertar ítem recalcula total del pedido con fn\_total\_pedido.  
Objetivo: asegurar consistencia del total.  
Evento: AFTER INSERT en item\_pedido.  
Tablas afectadas: pedidos.

Trigger 3: trg\_historial\_estado

Descripción: audita cambios de estado del pedido registrando estado anterior/nuevo.  
Objetivo: trazabilidad y auditoría del ciclo de vida.  
Evento: AFTER UPDATE en pedidos.  
Tablas afectadas: historial\_estado\_pedido.

Trigger 4: trg\_devolver\_stock\_ad

Descripción: al borrar un ítem devuelve stock y actualiza total.  
Objetivo: permitir edición de carrito sin inconsistencias.  
Evento: AFTER DELETE en item\_pedido.  
Tablas afectadas: productos, pedidos.

Trigger 5: trg\_fact\_ventas\_ai

Descripción: al insertar ítem inserta snapshot en fact\_ventas (tabla de hechos).  
Objetivo: habilitar analítica y reporting sin recalcular joins complejos.  
Evento: AFTER INSERT en item\_pedido.  
Tablas afectadas: fact\_ventas.

---

## Scripts de inserción de datos

Enlace al [script](#)

El Script 2 incluye inserción de datos de prueba para **todas** las tablas del esquema:

- estados del pedido
- operadores logísticos y métodos de envío
- usuarios y direcciones
- categorías, productos y promociones
- pedidos, ítems, pagos y devoluciones

Además, ejecuta procedimientos almacenados y consultas sobre vistas para mostrar resultados en pantalla y validar el correcto funcionamiento de la base.

---

## Informes generados

A partir de las vistas y la tabla de hechos se pueden generar informes como:

- Ventas totales por usuario (vw\_ventas\_por\_usuario)
- Productos con stock bajo (vw\_productos\_stock\_bajo)
- Detalle operativo de pedidos (vw\_detalle\_pedidos)
- Promociones activas (vw\_promociones\_activas)
- Pedidos con información logística y días estimados (vw\_envios)
- Análisis de ventas por categoría, método de envío o medio de pago (usando fact\_ventas)

Estos informes permiten análisis comercial, logístico y contable.

---

## Herramientas y tecnologías utilizadas

Listado de herramientas y tecnologías:

- MySQL Workbench: Herramienta empleada para la creación y administración de la base de datos, ejecución de scripts SQL, pruebas de consultas, gestión de usuarios y generación del diagrama entidad-relación.
- SQL: Lenguaje principal utilizado para la definición de estructuras de datos, creación de objetos de base de datos, inserción de registros y generación de consultas analíticas.
- Visual Studio Code: Editor de texto utilizado para la edición de los scripts SQL y la redacción de la documentación técnica del proyecto, facilitando la organización y control del código fuente.
- draw.io: Herramienta utilizada para diseñar el Diagrama Entidad-Relación (DER), permitiendo representar gráficamente las entidades, atributos y relaciones de la base de datos.
- Microsoft Excel: Utilizados para visualizar, analizar y validar los datos insertados en la base de datos, así como para generar tablas de apoyo e informes preliminares.

- GitHub / Google Drive : Plataforma utilizada para almacenar, versionar y compartir los scripts SQL, diagramas y documentación del proyecto.
- 

## Líneas futuras de evolución del sistema

Detalle sobre futuras líneas de ampliación de la base de datos:

- Gestión avanzada de promociones: Incorporar reglas más complejas de descuentos, cupones por usuario, promociones acumulables y campañas por temporadas.
- Módulo de facturación: Agregar tablas y procesos para la generación de comprobantes fiscales, facturas electrónicas y reportes impositivos.
- Seguimiento logístico detallado: Implementar el tracking de envíos con estados intermedios (en preparación, en tránsito, entregado), integración con APIs de operadores logísticos y notificaciones automáticas al cliente.
- Gestión de devoluciones ampliada: Incorporar estados de devolución, control de reintegros parciales y validaciones automáticas según políticas comerciales.
- Análisis avanzado de ventas: Integrar la base de datos con herramientas de Business Intelligence como Power BI o Tableau para generar dashboards de ventas, comportamiento de clientes y rentabilidad por categoría.
- Escalabilidad del sistema: Adaptar el modelo para soportar múltiples tiendas, sucursales o marketplaces desde una única base centralizada.
- Portal web para clientes: Desarrollar una interfaz web donde los usuarios puedan consultar pedidos, estados de envío, historial de compras y promociones activas.
- Aplicación móvil: Crear una aplicación para que clientes y operadores accedan a la información desde dispositivos móviles, mejorando la experiencia de usuario.
- Automatización de notificaciones: Implementar el envío automático de correos electrónicos o mensajes para confirmar pedidos, informar despachos, promociones vigentes y devoluciones procesadas.