

# PROPUESTA PROYECTO

## Desarrollo de Base de Datos

Para Ecommerce B2C de Productos Físicos

**Curso SQL**

**Comisión 53170**

**Alumna:**

Victoria Cabrera

**Profesor:**

Camilo Andres Redondo

**Tutora**

Jennifer Goldfeld



# ÍNDICE

## CONTENIDO DEL INFORME

Introducción	1
Objetivos	1
Situación Problemática	2
Modelo de Negocio	2
Elementos clave del modelo	3
Diagramas entidad relación	4
Modelo de la base de datos	5
Descripción de tablas	6-13
Descripción de funciones	14
Descripción de procedimientos	14
Descripción de triggers	15
Descripción de vistas	15
Tecnologías utilizadas	16

# INTRODUCCIÓN

En el contexto actual del comercio electrónico B2C (Business to Consumer), la presencia online es esencial para el éxito de las empresas minoristas. Con el continuo crecimiento de esta modalidad de comercio, la gestión eficiente de la información se vuelve crucial para ofrecer una experiencia de compra óptima.

Este proyecto se centra en el desarrollo de una base de datos robusta y eficiente, específicamente diseñada para impulsar las operaciones de un ecommerce B2C, mejorando la gestión de datos y elevando la experiencia de compra tanto para los clientes como para la empresa.

## OBJETIVOS

### OPTIMIZACIÓN DE LA GESTIÓN DE DATOS

Implementar una base de datos que permita almacenar y recuperar información de manera rápida y eficiente, garantizando la integridad y seguridad de los datos relacionados con productos físicos.

### MEJORA DE LA EXPERIENCIA DEL CLIENTE

Facilitar la búsqueda de productos, personalizar recomendaciones y agilizar el proceso de compra para mejorar la experiencia del cliente, fomentando la satisfacción y la lealtad.

### EFICIENCIA OPERATIVA

Automatizar procesos internos, como la gestión de inventario, seguimiento de pedidos y análisis de datos específicos para productos físicos, para mejorar la eficiencia operativa y reducir costos.

## SITUACIÓN PROBLEMÁTICA

La falta de una base de datos optimizada puede generar problemas como tiempos de carga prolongados, dificultades en la búsqueda de productos y errores en la gestión de inventario específicos para la venta minorista de productos físicos. Estas deficiencias no solo afectan la experiencia del cliente, sino que también impactan negativamente en la eficiencia interna y la capacidad de la empresa para adaptarse rápidamente a las demandas del mercado minorista.

## MODELO DE NEGOCIO

El proyecto se basa en la implementación de un modelo de negocio centrado en la tecnología de la información para ecommerce B2C. La misma proporcionará a los usuarios la capacidad de explorar productos, comparar precios, realizar pedidos y realizar pagos en línea. El modelo de negocio está pensado para una plataforma de tamaño y complejidad media, ofreciendo una amplia gama de productos.



# ELEMENTOS CLAVE DEL MODELO

## CATÁLOGO DE PRODUCTOS

Una lista completa de productos o servicios disponibles para la venta, con descripciones, imágenes y precios.

## CARRITO DE COMPRAS

Una funcionalidad que permite a los usuarios agregar productos a su "carrito" virtual y luego proceder al proceso de pago.

## PROCESO DE PAGO

La capacidad de realizar transacciones seguras en línea, utilizando métodos de pago como tarjetas de crédito, transferencias bancarias, o plataformas de pago electrónico.

## SEGURIDAD

Medidas de seguridad para proteger la información personal y financiera de los clientes durante el proceso de compra.

## GESTIÓN DE INVENTARIO

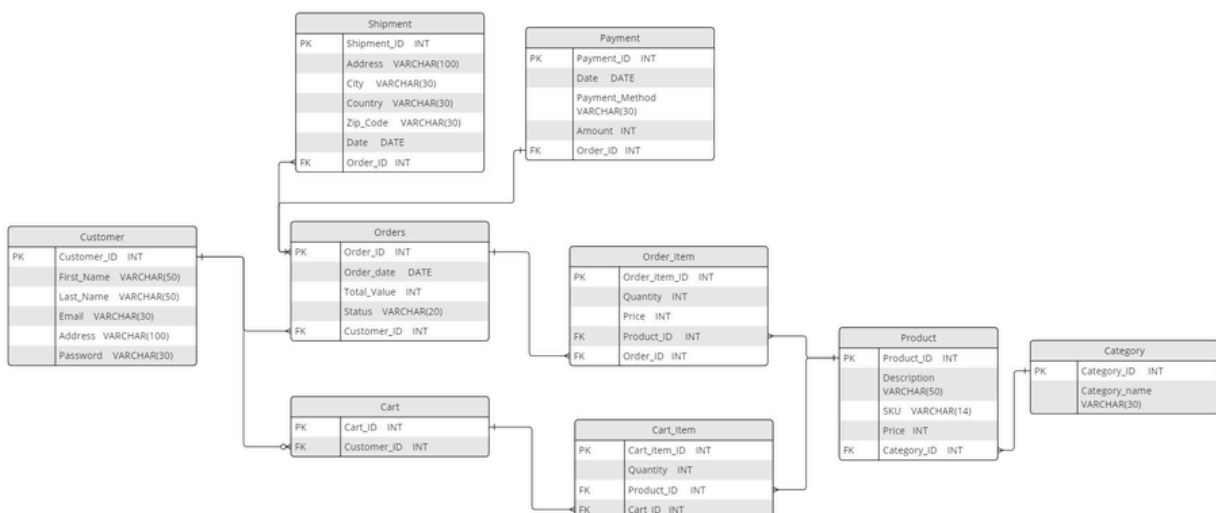
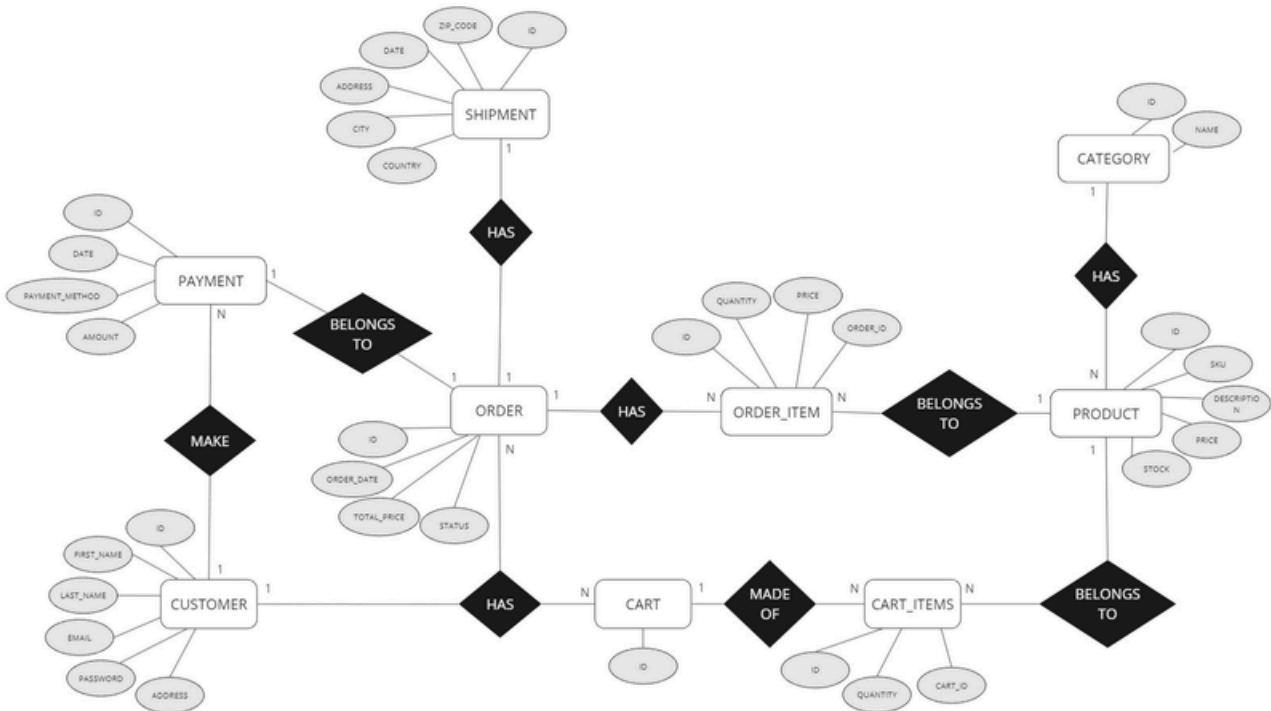
Un sistema para llevar un control eficiente del stock de productos disponibles.

## LOGÍSTICA Y ENVÍO

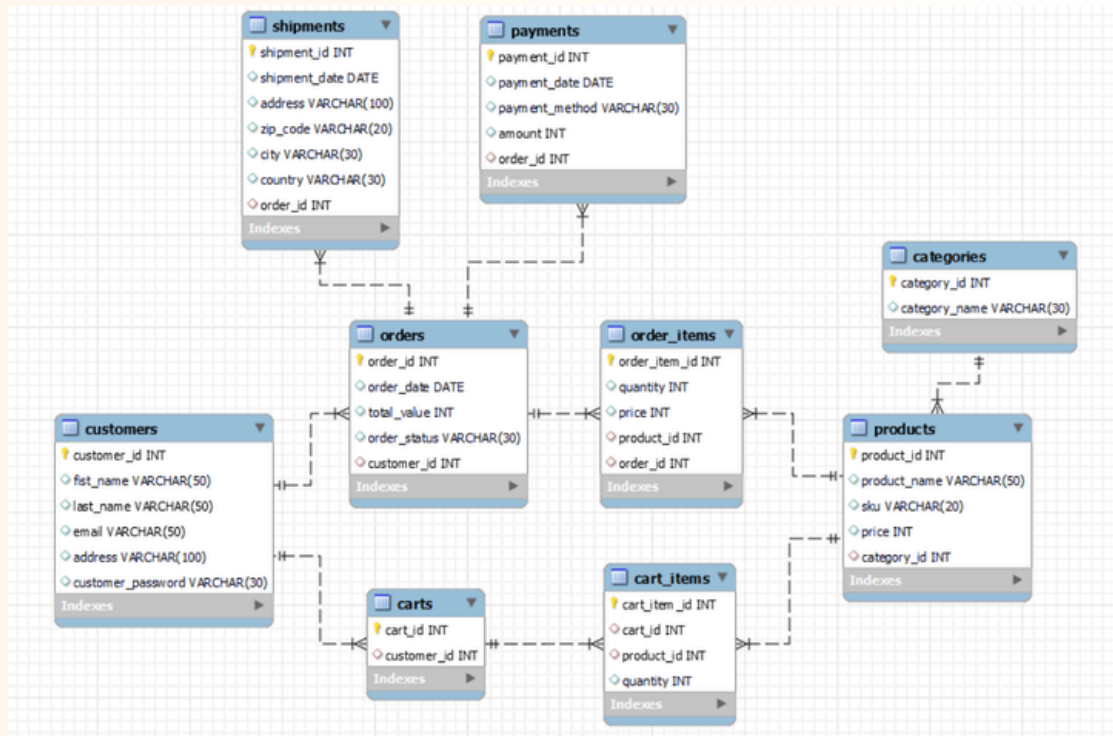
Coordinación de la entrega de productos al cliente.



# DIAGRAMAS ENTIDAD RELACIÓN



# MODELO DE LA BASE DE DATOS EN MYSQL



# DESCRIPCIÓN DE TABLAS

## CUSTOMERS

Esta tabla tiene por objetivo almacenar toda la información relacionada a los clientes.

Columna	Tipo de dato	Tipo de clave	Descripción
customer_id	INT	PK	*AUTO-INCREMENTAL id del cliente
first_name	VARCHAR(50)		nombre del cliente
last_name	VARCHAR(50)		apellido del cliente
email	VARCHAR(50)		*NOT NULL email del cliente
address	VARCHAR(100)		dirección del cliente
customer_password	VARCHAR(30)		contraseña del cliente



# DESCRIPCIÓN DE TABLAS

## PRODUCTS

En esta tabla se almacenan los productos pertenecientes al ecommerce

Columna	Tipo de dato	Tipo de clave	Descripción
product_id	INT	PK	*AUTO-INCREMENTAL id del producto
product_name	VARCHAR(50)		nombre del producto
sku	VARCHAR(14)		número de referencia único
price	DECIMAL		precio del producto
category_id	INT	FK	categoría a la que pertenece el producto

## CATEGORIES

La tabla contiene la información relacionada a las categorías a las que puede pertenecer un producto

Columna	Tipo de dato	Tipo de clave	Descripción
category_id	INT	PK	*AUTO-INCREMENTAL id de la categoría
category_name	VARCHAR(30)		nombre de la categoría

# DESCRIPCIÓN DE TABLAS

## CARTS

La tabla contiene los carritos de compras relacionados a los clientes

Columna	Tipo de dato	Tipo de clave	Descripción
cart_id	INT	PK	*AUTO-INCREMENTAL id del carrito
customer_id	INT	FK	id del cliente dueño del carrito

## CART\_ITEMS

Esta tabla tiene como finalidad guardar los items pertenecientes a los carritos de compras. Se crea un cart\_item por producto agregado a un carrito, y se relaciona con un carrito de compras y un producto.

Columna	Tipo de dato	Tipo de clave	Descripción
cart_item_id	INT	PK	*AUTO-INCREMENTAL id del item del carrito
cart_id	INT	FK	id del carrito al que pertenece el item
product_id	INT	FK	id del producto agregado al carrito
quantity	INT		cantidad de unidades del producto agregada al carrito

# DESCRIPCIÓN DE TABLAS

## ORDERS

Esta tabla guarda la información de cada orden que se genera, relacionandola con un cliente

Columna	Tipo de dato	Tipo de clave	Descripción
order_id	INT	PK	*AUTO-INCREMENTAL id de la orden
order_date	DATE		fecha en la que se generó la orden
total_value	DECIMAL		valor total de la orden
order_status	VARCHAR(30)		Estado de la orden
customer_id	INT	FK	id del cliente dueño de la orden

# DESCRIPCIÓN DE TABLAS

## ORDER\_ITEMS

Guarda información de cada item que pertenece a una orden, y lo relaciona a un producto existente.

Columna	Tipo de dato	Tipo de clave	Descripción
order_item_id	INT	PK	*AUTO-INCREMENTAL id del item de la orden
quantity	INT		*NOT NULL cantidad de unidades del producto
price	DECIMAL		*NOT NULL precio x unidad del producto
product_id	INT	FK	*NOT NULL id del producto ordenado
order_id	INT	FK	*NOT NULL id de la orden a la que pertenece el item

# DESCRIPCIÓN DE TABLAS

## SHIPMENTS

Tabla que almacena información sobre cada envío, y la relaciona con una orden en particular.

Columna	Tipo de dato	Tipo de clave	Descripción
shipment_id	INT	PK	*AUTO-INCREMENTAL id del envío
shipment_date	DATE		Fecha en la que se realizó el envío
address	VARCHAR(100)		Dirección de envío
zip_code	VARCHAR(20)		Código postal
city	VARCHAR(30)		Ciudad
country	VARCHAR(30)		País
order_id	INT	FK	id de la orden a la que pertenece el envío

# DESCRIPCIÓN DE TABLAS

## PAYMENTS

La finalidad de esta tabla es guardar la información sobre pagos, relacionandola con una orden

Columna	Tipo de dato	Tipo de clave	Descripción
payment_id	INT	PK	*AUTO-INCREMENTAL id del pago
payment_date	DATE		Fecha en la que se realizó el pago
payment_method	VARCHAR(30)		Método de pago con el que se realizó el pago
amount	DECIMAL		Valor total pagado
order_id	INT	FK	Id de la orden pagada



# DESCRIPCIÓN DE TABLAS

## LOG\_AUDITORIA

Para registrar movimientos realizados en tablas con información sensible

Columna	Tipo de dato	Tipo de clave	Descripción
ID_LOG	INT	PK	*AUTO-INCREMENTAL id del registro
PK_ID	INT		*NOT NULL id (PK) perteneciente al registro que fue modificado/insertado/e eliminado
CAMPONUEVO_CAMP OANTERIOR	VARCHAR(255)		Registra los valores que sufrieron la modificación. Insert: campo nuevo Update: campo nuevo + campo anterior Delete: campo anterior
NOMBRE_DE_ACCION	VARCHAR(10)		Acción que se ejecutó: INSERT/UPDATE/DELETE
NOMBRE_TABLA	VARCHAR(50)		Tabla que sufrió la modificación
USUARIO	VARCHAR(100)		Usuario que ejecutó la consulta
FECHA_UPD_INS_DEL	TIMESTAMP		Fecha de la modificación

## DESCRIPCIÓN DE FUNCIONES

### **TOTAL\_SPENT\_BY\_CUSTOMER**

Calcula el total gastado por un cliente en todas sus órdenes.

Requiere 1 parámetro de entrada: el `customer_id`

Retorna la cantidad gastada por el cliente como un entero

### **COUNT\_ORDERS\_COMPLETED\_BETWEEN\_DATES**

Cuenta la cantidad de órdenes completadas en un rango de fechas.

Requiere 2 parámetros de entrada: `start_date` y `end_date`

Retorna la cantidad de órdenes como un entero

## DESCRIPCIÓN DE PROCEDIMIENTOS ALMACENADOS

### **GET\_PRODUCTS\_IN\_CART**

Procedimiento para obtener todos los productos en un carrito de compras.

Este procedimiento toma el ID del carrito como entrada y devuelve todos los productos incluidos en ese carrito

### **GET\_ORDER\_STATUS\_AND\_SHIPMENT\_DATE**

Procedimiento para obtener el estado de una orden y la fecha de envío.

Toma el ID de la orden como entrada y devuelve el estado actual de la orden y la fecha de envío si está disponible.

## DESCRIPCIÓN DE TRIGGERS

### ■ TRG\_LOG\_AUDITORIA\_1

Se dispara luego de insertar un nuevo registro en la tabla Customers y almacena datos clave en la tabla de Auditoría: número de customer, email, y responsable de la modificación, además de la fecha y hora de la misma, entre otros.

### ■ TRG\_LOG\_AUDITORIA\_2

Se dispara luego de modificar ordenes y crea un registro en la tabla de Auditoría, almacenando número de orden modificada, valores totales nuevos y antiguos, fecha y hora, usuario responsable de la modificación, entre otros.

## DESCRIPCIÓN DE VISTAS

### ■ VW\_LOG\_AUDITORIA

Vista para acceder a la información almacenada en la tabla Auditoría. La misma esta ordenada por fecha de forma descendente (últimos movimientos se ven primero)

### ■ VW\_CUSTOMER\_CART\_DETAILS

Vista de detalles de los clientes y sus carritos de compras asociados

### ■ VW\_ORDER\_STATUS\_DETAILS

Vista de detalles de las ordenes, su estado e información del envío asociado a cada una.

### ■ VW\_ORDER\_PAYMENT\_DETAILS

Vista de detalles de las ordenes, su estado e información del pago asociado a cada una.

### ■ VW\_SALES\_BY\_CATEGORY

Vista de resumen de ventas por categoría.

# TECNOLOGÍAS Y HERRAMIENTAS UTILIZADAS

Para el desarrollo del proyecto se utilizaron las siguientes tecnologías:

**Miro:** Creación de Diagrama Entidad Relación

**ChatGPT:** Consultas puntuales técnicas

**MySQL Workbench:** Describir la base de datos, definir su estructura y crear sus objetos.

**GitHub:** Exponer la información del proyecto y su código.

**Canva:** Desarrollo de informe.