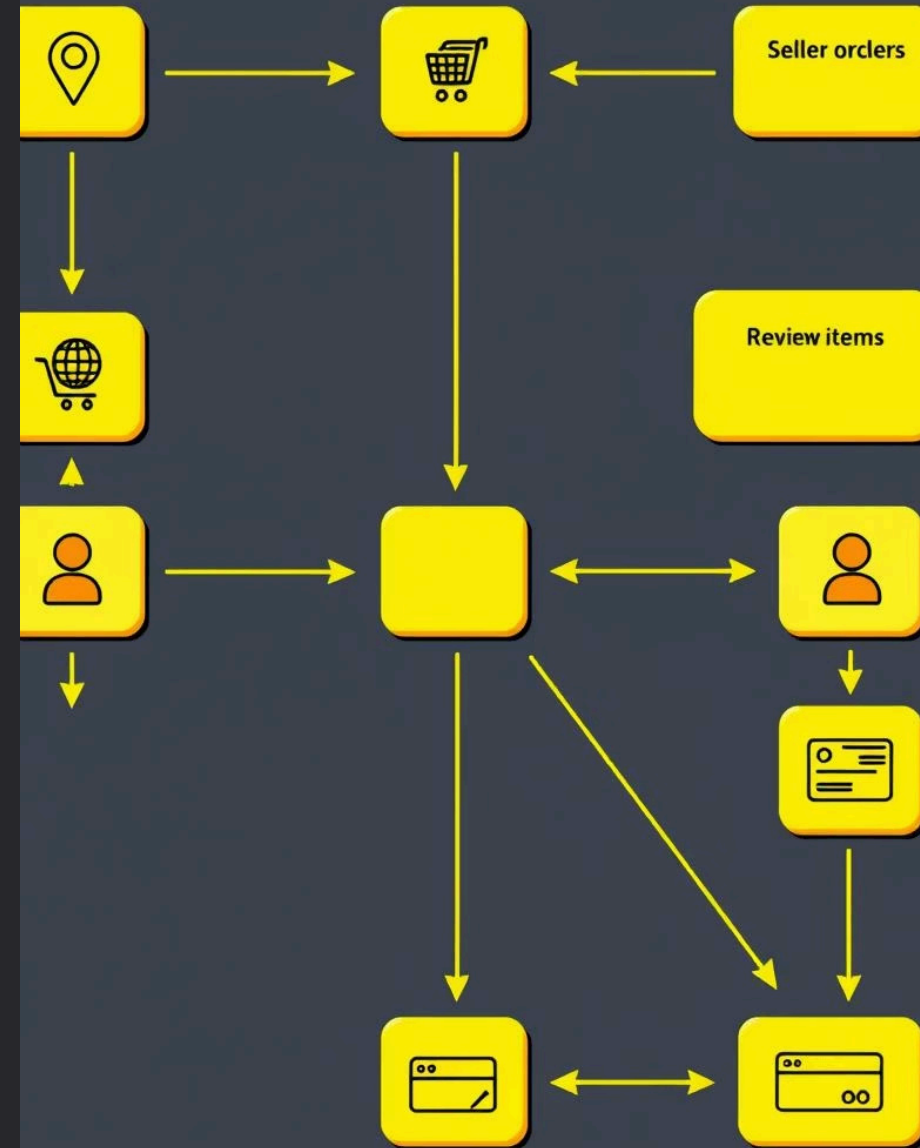


Diseño de Base de Datos Relacional para Fecom Inc.

Este proyecto se centra en el diseño e implementación de una base de datos relacional robusta y eficiente para Fecom Inc., una empresa de e-commerce. La solución se basa en el dataset disponible en Kaggle, con el objetivo de crear un esquema que permita la gestión integral de geolocalizaciones, productos, vendedores, clientes, pedidos, reseñas, pagos e ítems de pedido. La finalidad es proporcionar una base sólida para análisis contables, logísticos y del comportamiento del cliente, mejorando la toma de decisiones y la eficiencia operativa.

 por Luka Lattanzi



Sobre el Dataset

Bienvenido a Fecom Inc.

Fecom Inc. es una empresa ficticia de mercado de comercio electrónico con sede en Berlín, Alemania. Entre 2022 y 2024, registró 99.441 pedidos de 102.727 clientes únicos y realizó un seguimiento de todas las transacciones comerciales de 3.095 vendedores.

Este conjunto de datos contiene datos comerciales de 338 ciudades de 28 países e incluye diversos detalles sobre clientes y pedidos, lo que permite realizar análisis de CRM, de oportunidades de venta o de marketing. Los datos incluyen fechas de pedido, fechas de entrega, valores del carrito, gastos de envío, reseñas y valoraciones de los pedidos, métodos de pago y mucho más.

Dado que el mercado no se limita a un producto específico, cuenta con 32.951 productos distintos en 72 categorías diferentes.

Situación Problemática

Durante el desarrollo del proyecto con el dataset de Fecom Inc. extraído de Kaggle, se identificó una clara necesidad de estructurar la información dentro de una base de datos relacional. El objetivo principal fue implementar una solución que permitiera realizar un seguimiento completo del funcionamiento de un sistema de e-commerce, abarcando tanto el comportamiento de los clientes como de los vendedores, así como también el movimiento de productos y ventas.

Uno de los primeros desafíos fue la inconsistencia y la falta de datos en varias tablas del conjunto original. Se encontraron documentos incompletos, campos ausentes y diferencias de formato que dificultaban la carga directa en SQL. Para lograr una integración exitosa, fue necesario realizar un proceso exhaustivo de limpieza, transformación y normalización de los datos. Esto incluyó operaciones como:

- 1 Conversión de formatos de fecha y hora
- 2 Unificación de claves entre tablas
- 3 Relleno o eliminación de valores faltantes
- 4 Segmentación de campos complejos como direcciones o descripciones

La falta de una estructura previa también representaba una limitación al momento de responder preguntas fundamentales del negocio, como por ejemplo:

- 1 ¿Dónde se concentran los clientes geográficamente?
- 2 ¿Qué productos se venden más y en qué zonas?
- 3 ¿Cómo es el rendimiento de cada vendedor?
- 4 ¿Qué patrones hay en las compras por tipo de producto o categoría?

La implementación de una base de datos resolvió estas brechas permitiendo:

- 1 Trazabilidad completa de las ventas, desde el cliente hasta el producto vendido
- 2 Geolocalización de clientes y vendedores para realizar análisis espaciales
- 3 Seguimiento de productos, tanto en stock como en circulación
- 4 Generación de reportes más robustos para la toma de decisiones

Gracias a este modelo, ahora es posible explorar el comportamiento del sistema de e-commerce de manera organizada, ágil y escalable.

Modelo de Negocio

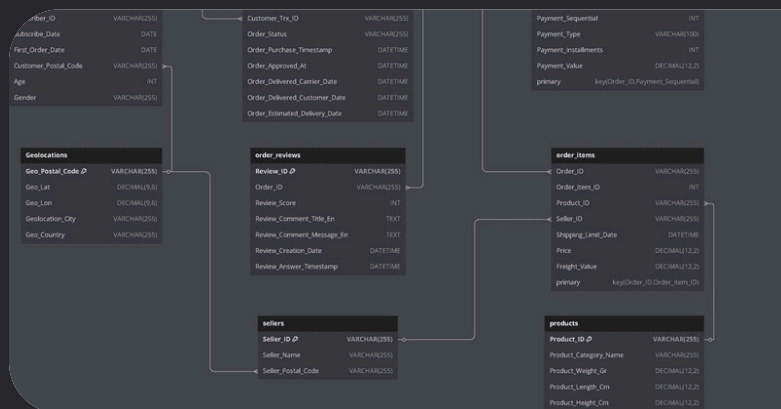
La base de datos implementada está diseñada para modelar y analizar cada aspecto del proceso de compra y venta dentro del sistema, con el objetivo de obtener información clave para la toma de decisiones.

Estructura y funcionamiento de la base de datos

El diseño relacional implementado se compone de múltiples tablas que representan las distintas entidades involucradas en el proceso comercial. A continuación, se describen los componentes principales:

- 1 customers**
Registra información personal y demográfica de los clientes, incluyendo edad, género y código postal. Esta tabla permite hacer segmentaciones y análisis de comportamiento de compra según perfiles.
- 2 geolocations**
Contiene datos de localización como latitud, longitud, ciudad y país, asociados al código postal. Esta relación facilita los análisis geográficos para determinar zonas de mayor demanda o problemas logísticos.
- 3 sellers**
Almacena información de los vendedores, vinculándolos también con su código postal. Esto permite estudiar su distribución geográfica y evaluar su desempeño en función del volumen de ventas o tiempos de entrega.
- 4 products**
Define los productos disponibles, su categoría y dimensiones físicas. Esta tabla es esencial para hacer análisis de stock, tendencias de consumo y optimización de envíos.
- 5 orders**
Central en el modelo, esta tabla enlaza cada compra con el cliente que la realizó y contiene las fechas clave del proceso (compra, aprobación, entrega, estimación). Permite estudiar tiempos logísticos y tasas de conversión.
- 6 order_items**
Detalla los productos incluidos en cada orden, junto con el vendedor correspondiente, el precio y los costos de envío. Es clave para estudiar el comportamiento de ventas por ítem y proveedor.
- 7 order_reviews**
Registra la retroalimentación de los clientes en forma de calificaciones y comentarios. Aporta valor al análisis de la experiencia de usuario y calidad del servicio.
- 8 order_payments**
Contiene los detalles de pago, tipo, valor e incluso cuotas, lo cual permite observar tendencias en métodos de pago y ticket promedio.

Diagrama Entidad Relación



PDF-файл

Untitled.pdf

279.4 KB



Descripción de Tablas y Atributos

Tabla	Atributos	Tipo
customers	<ul style="list-style-type: none">customer_id (Primary Key)first_namelast_nameemailgenderagepostal_code (Foreign Key referencing geolocations.postal_code)	<ul style="list-style-type: none">IntegerVarcharVarcharVarcharVarcharIntegerVarchar
geolocations	<ul style="list-style-type: none">postal_code (Primary Key)citystatecountrylatitudelongitude	<ul style="list-style-type: none">VarcharVarcharVarcharVarcharFloatFloat
sellers	<ul style="list-style-type: none">seller_id (Primary Key)nameemailphonepostal_code (Foreign Key referencing geolocations.postal_code)	<ul style="list-style-type: none">IntegerVarcharVarcharVarcharVarchar
products	<ul style="list-style-type: none">product_id (Primary Key)namecategorylengthwidthheightweight	<ul style="list-style-type: none">IntegerVarcharVarcharFloatFloatFloatFloat
orders	<ul style="list-style-type: none">order_id (Primary Key)customer_id (Foreign Key referencing customers.customer_id)order_dateapproved_datedelivered_dateestimated_delivery_date	<ul style="list-style-type: none">IntegerIntegerTimestampTimestampTimestampTimestamp
order_items	<ul style="list-style-type: none">order_item_id (Primary Key)order_id (Foreign Key referencing orders.order_id)product_id (Foreign Key referencing products.product_id)seller_id (Foreign Key referencing sellers.seller_id)quantitypriceshipping_cost	<ul style="list-style-type: none">IntegerIntegerIntegerIntegerDecimalDecimal
order_reviews	<ul style="list-style-type: none">review_id (Primary Key)order_id (Foreign Key referencing orders.order_id)ratingreview_text	<ul style="list-style-type: none">IntegerIntegerIntegerText
order_payments	<ul style="list-style-type: none">payment_id (Primary Key)order_id (Foreign Key referencing orders.order_id)payment_typepayment_amountinstallments	<ul style="list-style-type: none">IntegerIntegerVarcharDecimalInteger