MANUAL BASE DE DATOS INVENTORY

ABEL AUDINO PANTOJA RODRIGUEZ

Presentado a Brayan Arcos

INSTITUTO TECNOLOGICO DEL PUTUMAYO

DESARROLLO DE BASE DE DATOS

MOCOA-PUTUMAYO

2024

RESUMEN EJECUTIVO

Este documento presenta el diseño de la base de datos para el sistema Inventory, enfocado en optimizar la gestión integral de inventarios, el procesamiento de órdenes de compra y la interacción con proveedores y clientes a través de la relación con users y people. La estructura se ha desarrollado bajo estrictas prácticas de normalización para asegurar la integridad referencial y evitar redundancias en el almacenamiento de datos. Con un modelo libre de duplicidades entre people y users, y una relación de roles de usuario simplificada, se espera lograr un rendimiento superior y una administración más eficiente del inventario y las operaciones comerciales.

La solución SQL implementada garantiza tanto la integridad como la escalabilidad del sistema, permitiendo una expansión sin problemas a medida que aumentan las transacciones y usuarios. La base de datos se organiza alrededor de seis tablas principales: users, people, products, purchaseOrders, orderDetails e historyProductPrices. Estas tablas representan las entidades clave para la gestión de usuarios, proveedores, productos y el ciclo completo de órdenes de compra, proporcionando una base sólida y flexible para las necesidades actuales y futuras del sistema.

Se espera que este diseño impulse mejoras en el rendimiento, minimice la redundancia de datos y proporcione una administración eficaz de los inventarios y relaciones con proveedores, con una estructura que facilita el escalamiento en línea con el crecimiento del negocio.

INTRODUCCIÓN

El desarrollo de este proceso lo encamino en el ámbito de aprendizaje teniendo como objetivo principal el desarrollo de la base de datos Inventory para proporcionar una estructura robusta para la gestión eficiente de usuarios, productos, proveedores y órdenes de compra en una organización. El diseño anterior incluía ciertas redundancias y complejidades que podrían dificultar la escalabilidad y el mantenimiento a largo plazo. Este documento detalla las mejoras implementadas para cumplir con las reglas de normalización y optimizar el uso de los recursos, manteniendo la integridad de los datos y evitando duplicidades.

Propósito del manual: En este documento se describe la estructura y funcionamiento de la base de datos Inventory, que se utiliza para gestionar inventarios, pedidos, productos, pagos y relaciones con clientes.

Alcance: Esta base de datos es adecuada para gestionar operaciones relacionadas con productos, ventas, facturas, y pagos, así como los datos de usuarios y ubicaciones asociadas.

Audiencia objetivo: Este manual está dirigido a administradores de bases de datos, desarrolladores y analistas responsables de la operación y mantenimiento de Inventory.

ESTRUCTURA DE LA BASE DE DATOS

A continuación, se describen las tablas principales y sus relaciones:

Tabla **addresses**

Descripción: Almacena la información de direcciones asociadas a personas y ubicaciones.

Columnas:

**id**: Identificador único de la dirección (clave primaria).

**street**: Calle de la dirección (no nulo).

**city**: Ciudad de la dirección (no nulo).

**state**: Estado o región de la dirección.

**zipCode**: Código postal.

**createdAt, updatedA**t: Tiempos de creación y actualización de la entrada.

Tabla **identificationType**

Descripción: Define los tipos de identificación para personas (por ejemplo, “CC”, “CE”).

Columnas:

**id**: Identificador único del tipo de identificación (clave primaria).

**type**: Tipo de identificación (p.ej., “CC”, “CE”).

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **roles**

Descripción: Define los roles que pueden tener los usuarios (p.ej., administrador, cliente).

Columnas:

**id**: Identificador único del rol (clave primaria).

**roleName**: Nombre del rol.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **users**

Descripción: Almacena la información de los usuarios registrados.

Columnas:

**id**: Identificador único del usuario (clave primaria).

**userName**: Nombre del usuario (no nulo).

**password**: Contraseña (no nulo).

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **userRoles**

Descripción: Relaciona usuarios con roles, permitiendo asignar múltiples roles a cada usuario.

Columnas:

**id**: Identificador único de la relación (clave primaria).

**idUser**: Referencia al identificador en users.

**idRole**: Referencia al identificador en roles.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **people**

Descripción: Almacena la información de personas, asociadas a usuarios y direcciones.

Columnas:

**id**: Identificador único de la persona (clave primaria).

**idUser**: Referencia al usuario en users.

**idType**: Referencia al tipo de identificación en identificationType.

**identificationNumber**: Número de identificación de la persona.

**firstName, middleName, lastName**: Nombre y apellidos de la persona.

**email, phone:** Información de contacto.

**idAddress**: Referencia a la dirección en addresses.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **orderStatus**

Descripción: Define los posibles estados de las órdenes de compra (p.ej., pendiente, completado).

Columnas:

**id**: Identificador único del estado (clave primaria).

**name**: Nombre del estado.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **productCategories**

Descripción: Almacena las categorías de productos.

Columnas:

**id**: Identificador único de la categoría (clave primaria).

**categoryName**: Nombre de la categoría.

**description**: Descripción de la categoría.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **products**

Descripción: Almacena la información de productos disponibles en el inventario.

Columnas:

**id**: Identificador único del producto (clave primaria).

**name**: Nombre del producto.

**description**: Descripción detallada del producto.

**price**: Precio actual del producto.

**idCategory**: Referencia a la categoría en productCategories.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **purchaseOrders**

Descripción: Registra las órdenes de compra generadas por la empresa.

Columnas:

**id**: Identificador único de la orden de compra (clave primaria).

**idSupplier**: Referencia al proveedor en users.

**idStatus**: Estado de la orden (referencia en orderStatus).

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **orderDetails**

Descripción: Almacena los detalles de cada producto en una orden de compra.

Columnas:

**id**: Identificador único del detalle (clave primaria).

**idPurchaseOrder**: Referencia a la orden de compra en purchaseOrders.

**idProduct**: Referencia al producto en products.

**quantity**: Cantidad del producto.

**price**: Precio unitario al momento de la compra.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **historyProductPrices**

Descripción: Guarda el historial de precios de productos.

Columnas:

**id**: Identificador único del registro de precio (clave primaria).

**idProduct**: Referencia al producto en products.

**price**: Precio del producto en un momento específico.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **invoices**

Descripción: Almacena las facturas generadas para los clientes.

Columnas:

**id**: Identificador único de la factura (clave primaria).

**idUser**: Referencia al usuario que generó la factura en users.

**total**: Total de la factura.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **invoicesDetails**

Descripción: Contiene los detalles de los productos en cada factura.

Columnas:

**id**: Identificador único del detalle (clave primaria).

**idInvoice**: Referencia a la factura en invoices.

**idProduct**: Referencia al producto en products.

**quantity**: Cantidad de producto facturada.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **paymentMethods**

Descripción: Define los métodos de pago (p.ej., tarjeta, efectivo).

Columnas:

**id**: Identificador único del método de pago (clave primaria).

**methodName**: Nombre del método de pago.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **locations**

Descripción: Almacena la información de las ubicaciones físicas de almacenamiento.

Columnas:

**id**: Identificador único de la ubicación (clave primaria).

**locationName**: Nombre de la ubicación.

**idAddress**: Referencia a addresses.

**phone, email**: Información de contacto.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **paymentGateways**

Descripción: Define los gateways de pago disponibles para cada método de pago.

Columnas:

**id**: Identificador único del gateway de pago (clave primaria).

**idPaymentMethod**: Referencia a paymentMethods.

**gateway**: Nombre del gateway.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **paymentGatewaysLocation**

Descripción: Asocia un gateway de pago con una ubicación.

Columnas:

**id**: Identificador único de la relación (clave primaria).

**idLocation**: Referencia a locations.

**idPaymentGateway**: Referencia a paymentGateways.

**createdAt**, **updatedAt**: Fechas de creación y actualización.

Tabla **payments**

Descripción: Registra los pagos realizados para facturas.

Columnas:

**id**: Identificador único del pago (clave primaria).

**idInvoice**: Referencia a la factura en invoices.

**amount**: Monto del pago.

**idPaymentMethod**: Método de pago.

**paymentDate**: Fecha del pago.

**status**: Estado del pago (exitoso, pendiente, fallido).

**idPaymentGatewaysLocation**: Gateway de pago y ubicación asociados.

**createdAt**, **updatedAt**: Fechas de creación y actualización.

Tabla **parameters**

Descripción: Define parámetros adicionales para configuraciones del sistema.

Columnas:

**id**: Identificador único del parámetro (clave primaria).

**code**, **description**: Código y descripción del parámetro.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **paymentGatewaysLocationParameter**

Descripción: Define parámetros específicos para cada gateway de pago y ubicación.

Columnas:

**id**: Identificador único (clave primaria).

**idParameter**: Referencia a parameters.

**idPaymentGatewayLocation**: Gateway y ubicación asociados.

**value**: Valor del parámetro.

**createdAt, updatedAt**: Fechas de creación y actualización.

Tabla **locationProduct**

Descripción: Asocia productos con ubicaciones de almacenamiento.

Columnas:

**id**: Identificador único de la relación (clave primaria).

**idProduct**: Referencia a products.

**idLocation**: Referencia a locations.

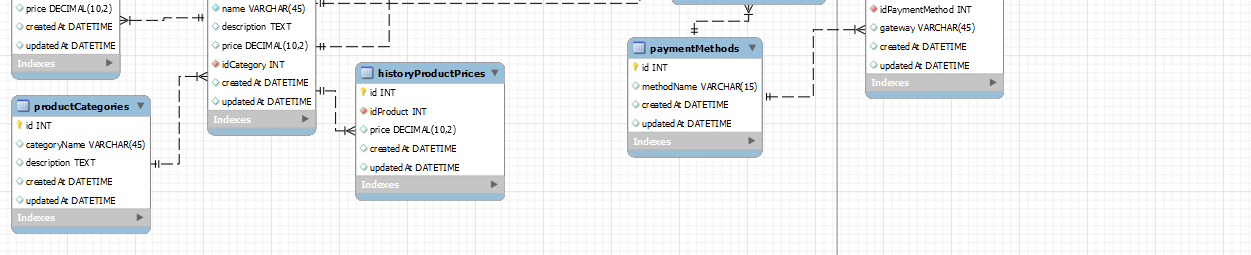
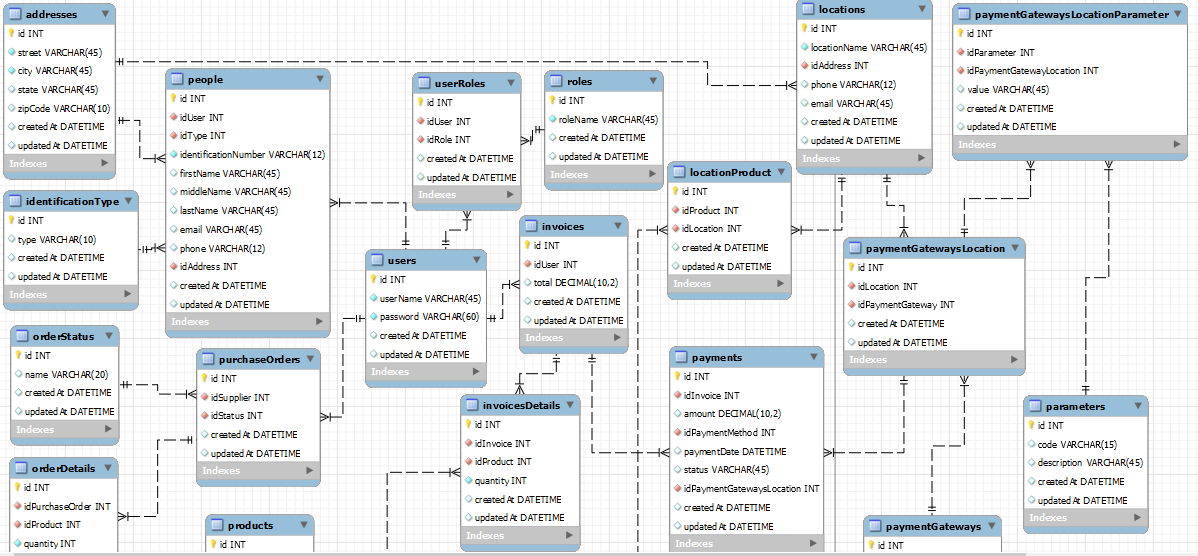
**createdAt**, **updatedAt**: Fechas de creación y actualización.

DESCRIPCIÓN GENERAL DE LA BASE DE DATOS

**Visión General del Sistema**: Inventory está diseñada para gestionar información de productos, usuarios, roles, pedidos y pagos. Es útil para aplicaciones de gestión de inventarios y facturación.

**Arquitectura General**: La base de datos contiene tablas que organizan la información en módulos como usuarios, productos, pedidos, pagos, y roles. Utiliza claves foráneas para relacionar datos entre tablas.

**Diagrama de Entidad-Relación (ERD):**



**Requisitos de Software y Hardware**: Requiere MySQL (versión recomendada 5.7 o superior), un servidor compatible, y espacio suficiente en disco para manejar el almacenamiento de inventario y datos de transacciones.

INSTRUCCIONES DE CONFIGURACIÓN

Instalación y Configuración: Crear la base de datos con CREATE DATABASE Inventory3 y ejecutar las instrucciones SQL.

Seguridad: Configurar permisos de usuario en MySQL para restringir el acceso.

Copia de Seguridad: Utilizar mysqldump para respaldar la base de datos periódicamente.

GUÍA DE OPERACIÓN

Consultas Comunes:

Obtener productos por categoría:

SELECT \* FROM products WHERE idCategory = 1;

Buscar usuarios por rol

SELECT u.userName, r.roleName FROM users u

JOIN userRoles ur ON u.id = ur.idUser

JOIN roles r ON ur.idRole = r.id;

PROCEDIMIENTOS DE ACTUALIZACIÓN Y MIGRACIÓN

**Actualización**: Al modificar estructuras de tablas, asegúrate de actualizar cualquier índice o clave foránea afectada.

**Migración de Datos**: Ejecutar en un entorno de pruebas antes de la producción.

SOLUCIÓN DE PROBLEMAS COMUNES

**Error en claves foráneas**: Revisar que los datos referenciados existan antes de insertarlos.

**Problemas de rendimiento**: Usar índices en campos clave como idUser en userRoles o idCategory en products.