

Sistema de Control de Stock para Tienda de Uñas

Alumno: Juan Franklin Lucero

Índice

1. Introducción
2. Objetivos del sistema
3. Diagrama Entidad-Relación (E-R)
4. Explicación del modelo
5. Listado de tablas
6. Vistas
7. Funciones
8. Stored Procedures
9. Triggers
10. Conclusiones

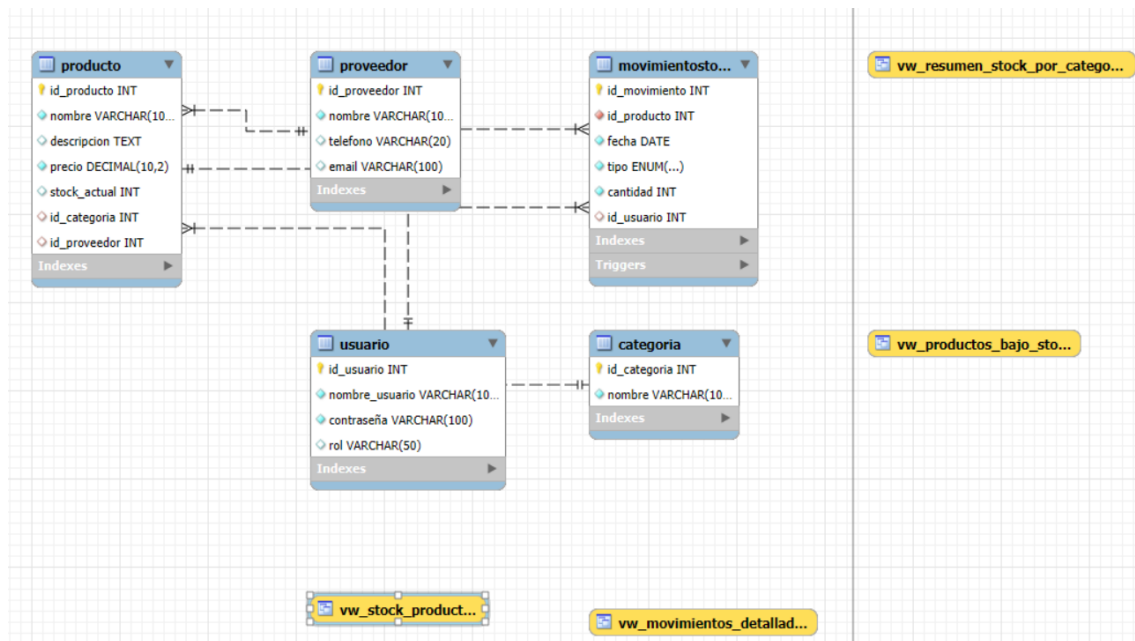
1. Introducción

El presente proyecto tiene como objetivo desarrollar una base de datos relacional para gestionar el stock de una tienda de uñas. El sistema permitirá registrar productos, proveedores, movimientos de inventario y usuarios del sistema, facilitando el control y la trazabilidad de los insumos.

2. Objetivos del sistema

- Controlar el ingreso y egreso de productos.
- Registrar proveedores y categorías de productos.
- Permitir la gestión por parte de distintos usuarios (administrador y empleados).
- Mantener actualizado el stock disponible de cada producto.
- Generar reportes de movimientos y estado de inventario.

3. Diagrama Entidad-Relación (E-R)



4. Explicación del modelo

El modelo se compone de cinco entidades principales:

- **Categoría:** clasifica los productos.
- **Proveedor:** almacena información de proveedores.
- **Producto:** representa los artículos disponibles.
- **Usuario:** identifica a los usuarios del sistema.
- **MovimientoStock:** registra entradas y salidas de productos.

Relaciones

- Un producto pertenece a una categoría (1:N).
- Un producto es provisto por un proveedor (1:N).
- Un movimiento de stock está asociado a un producto (N:1).
- Un movimiento de stock está registrado por un usuario (N:1).

5. Listado de Tablas

A continuación, se detallan las tablas que componen la base de datos ControlStockTienda, con sus campos, abreviaturas, tipos de datos y claves.

- Tabla: categoría

Descripción: Clasifica los productos según su tipo (esmaltes, herramientas, etc.)

Nombre completo	Tipo de dato	Clave
ID de categoría	INT	Primaria
Nombre	VARCHAR(100)	

- Tabla: Proveedor

Descripción: Registra los proveedores que abastecen productos a la tienda.

Nombre completo	Tipo de dato	Clave
ID de proveedor	INT	Primaria
Nombre	VARCHAR(100)	
Telefono	VARCHAR(20)	
Correo electronico	VARCHAR(100)	

- Tabla: Producto

Descripción: Contiene los productos disponibles en la tienda, con su categoría y proveedor.

Nombre completo	Tipo de dato	Clave
ID de producto	INT	Primaria
Nombre del producto	VARCHAR(100)	
Descripción	TEXT	
Precio unitario	DECIMAL(10,2)	
Stock disponible	INT	
ID de categoría	INT	Foránea
ID de proveedor	INT	Foránea

- Tabla: Usuario

Descripción: Usuarios del sistema con distintos roles (administrador o empleado).

Nombre completo	Tipo de dato	Clave
ID de usuario	INT	Primaria
Nombre de usuario	VARCHAR(100)	
Contraseña de acceso	VARCHAR(100)	
Rol del usuario	VARCHAR(50)	

- Tabla: Movimiento Stock

Descripción: Registra cada entrada o salida de productos del inventario.

Nombre completo	Tipo de dato	Clave
ID de movimiento	INT	Primaria
ID de producto	INT	Foránea
Fecha del movimiento	DATE	
Tipo de movimiento	ENUM('entrada','salida')	
Cantidad movida	INT	
ID de usuario	INT	Foránea

6. Vistas

- **vw_stock_productos**
 - **Objetivo:** mostrar el stock actual de cada producto con su categoría y proveedor.
 - **Tablas:** Producto, Categoría, Proveedor.
- **vw_movimientos_detallados**
 - **Objetivo:** mostrar el historial de movimientos con detalle de producto y usuario.
 - **Tablas:** MovimientoStock, Producto, Usuario.
- **vw_resumen_stock_por_categoria**

- **Objetivo:** obtener un resumen de stock total y cantidad de productos por categoría.
- **Tablas:** Categoría, Producto.
- **vw_productos_bajo_stock**
- **Objetivo:** identificar productos con stock menor a 10 unidades.
- **Tablas:** Producto.

7. Funciones

- **fn_precio_con_descuento(precio, descuento)**
 - **Objetivo:** calcular precio final aplicando un descuento.
 - **Datos:** campos de Producto.
- **fn_valor_stock(precio, stock)**
 - **Objetivo:** calcular el valor total del stock por producto.
 - **Datos:** campos de Producto.

8. Stored Procedures

- **sp_registrar_entrada(p_id_producto, p_cantidad, p_id_usuario)**
 - **Objetivo:** registrar una entrada de stock.
 - **Tablas:** MovimientoStock.
- **sp_registrar_salida(p_id_producto, p_cantidad, p_id_usuario)**
 - **Objetivo:** registrar una salida de stock (validación de stock a través del trigger).
 - **Tablas:** MovimientoStock.
- **sp_eliminar_producto(p_id_producto)**
 - **Objetivo:** eliminar un producto (y sus movimientos si la FK está en cascada).
 - **Tablas:** Producto, MovimientoStock.
- **sp_reponer_stock_por_proveedor(p_id_proveedor, p_cantidad, p_id_usuario)**

- **Objetivo:** insertar entradas masivas para todos los productos de un proveedor.
- **Tablas:** Producto, MovimientoStock.

9. Triggers

- **trg_actualizar_stock**
 - **Acción:** AFTER INSERT ON MovimientoStock.
 - **Objetivo:** actualizar automáticamente el stock de productos.
 - **Tablas:** MovimientoStock, Producto.
- **trg_evitar_stock_negativo**
- **Acción:** BEFORE INSERT ON MovimientoStock.
- **Objetivo:** evitar que se registren salidas con stock insuficiente.
- **Tablas:** MovimientoStock, Producto.

10. Conclusiones

El sistema de control de stock implementado permite:

- Mantener la integridad de los datos mediante triggers y restricciones.
- Automatizar procesos de actualización de stock.
- Generar reportes útiles a través de vistas.
- Ampliar la funcionalidad con funciones y procedimientos almacenados.