

# INFORME TÉCNICO – APLICACIÓN DE CONTROL DE INVENTARIO DE PRODUCTOS FERRETERIA

## Introducción

El presente informe describe el desarrollo e implementación de una aplicación de escritorio diseñada para el control de inventario de productos electrónicos. La solución fue desarrollada utilizando C# y Windows Forms, conectada a una base de datos relacional, e incorpora principios SOLID, patrones de diseño como Singleton y Repository, procedimientos almacenados, y reglas de negocio robustas para garantizar la integridad del sistema.

## Objetivo General

Desarrollar una aplicación de control de inventario que permita registrar productos electrónicos, gestionar proveedores, controlar entradas y salidas de productos, y generar alertas automáticas de stock bajo, con una interfaz amigable y segura para usuarios con distintos roles.

## Funcionalidades Principales del Sistema

Gestión completa de productos (CRUD)

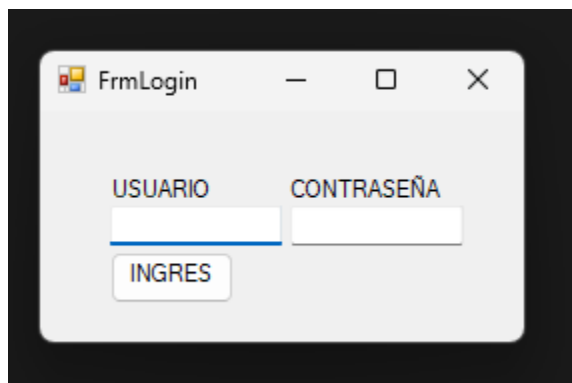
Gestión de proveedores y su historial de compras

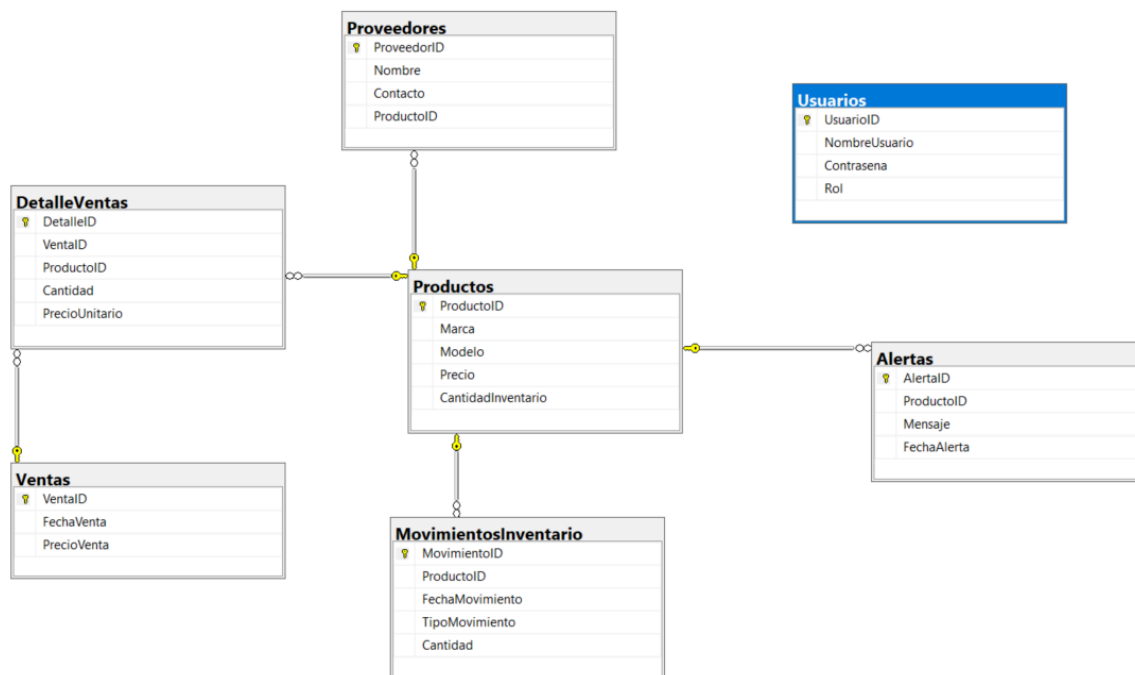
Registro y actualización de movimientos de inventario (entradas y salidas)

Reportes automáticos de ventas, inventario y productos más vendidos

Alertas automáticas por bajo inventario

Control de acceso por roles (administrador y empleado)





Base de Datos y Procedimientos Almacenados  
Se diseñaron cinco procedimientos almacenados:

SP\_RegistrarMovimiento – Registra entradas o salidas de inventario.

SP\_GenerarReporteInventario – Genera reportes de productos en stock bajo o movimientos realizados.

SP\_EnviarAlertaInventario – Envía alertas automáticas al administrador.

SP\_RegistrarVenta – Registra una venta y actualiza el inventario.

SP\_ObtenerHistorialComprasProveedor – Muestra el historial de compras a proveedores.

```

CREATE TABLE Productos (
    ProductoID INT PRIMARY KEY IDENTITY,
    Marca NVARCHAR(50),
    Modelo NVARCHAR(50),
    Precio DECIMAL(10,2),
    CantidadInventario INT
);

CREATE TABLE Proveedores (
    ProveedorID INT PRIMARY KEY IDENTITY,
    Nombre NVARCHAR(100),
    Contacto NVARCHAR(100),
    ProductoID INT FOREIGN KEY REFERENCES Productos(ProductoID)
);

CREATE TABLE MovimientosInventario (
    MovimientoID INT PRIMARY KEY IDENTITY,
    ProductoID INT FOREIGN KEY REFERENCES Productos(ProductoID),
    FechaMovimiento DATETIME,
    TipoMovimiento NVARCHAR(20),
    Cantidad INT
);

CREATE TABLE Ventas (
    VentaID INT PRIMARY KEY IDENTITY,
    ProductoID INT FOREIGN KEY REFERENCES Productos(ProductoID),
    FechaVenta DATETIME,
    Cantidad INT,
    PrecioVenta DECIMAL(10,2)
);

CREATE TABLE Alertas (
    AlertaID INT PRIMARY KEY IDENTITY,
    ProductoID INT FOREIGN KEY REFERENCES Productos(ProductoID),
    Mensaje NVARCHAR(255),
    FechaAlerta DATETIME
);

```

Reglas de Negocio Aplicadas

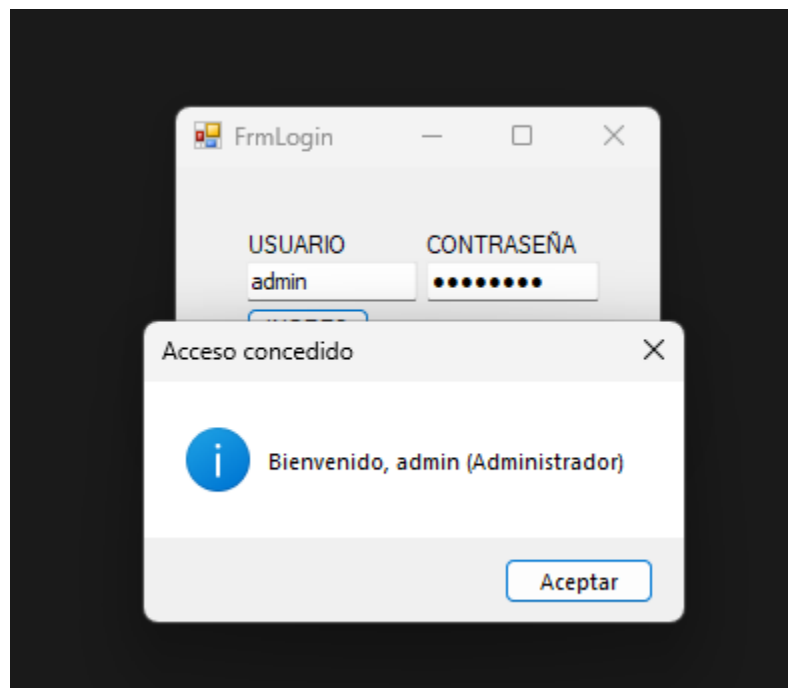
No se puede vender un producto sin stock suficiente.

Los productos con inventario mínimo generan automáticamente una alerta.

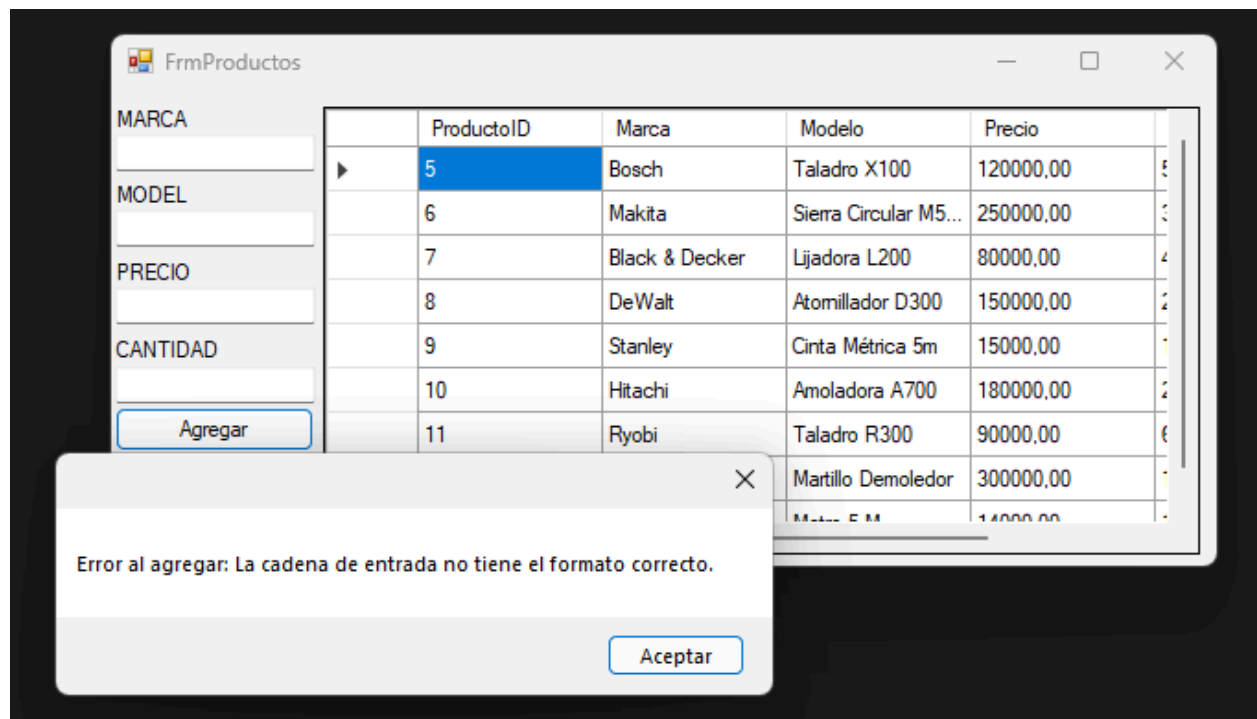
Solo los administradores pueden autorizar devoluciones y ver el historial completo.

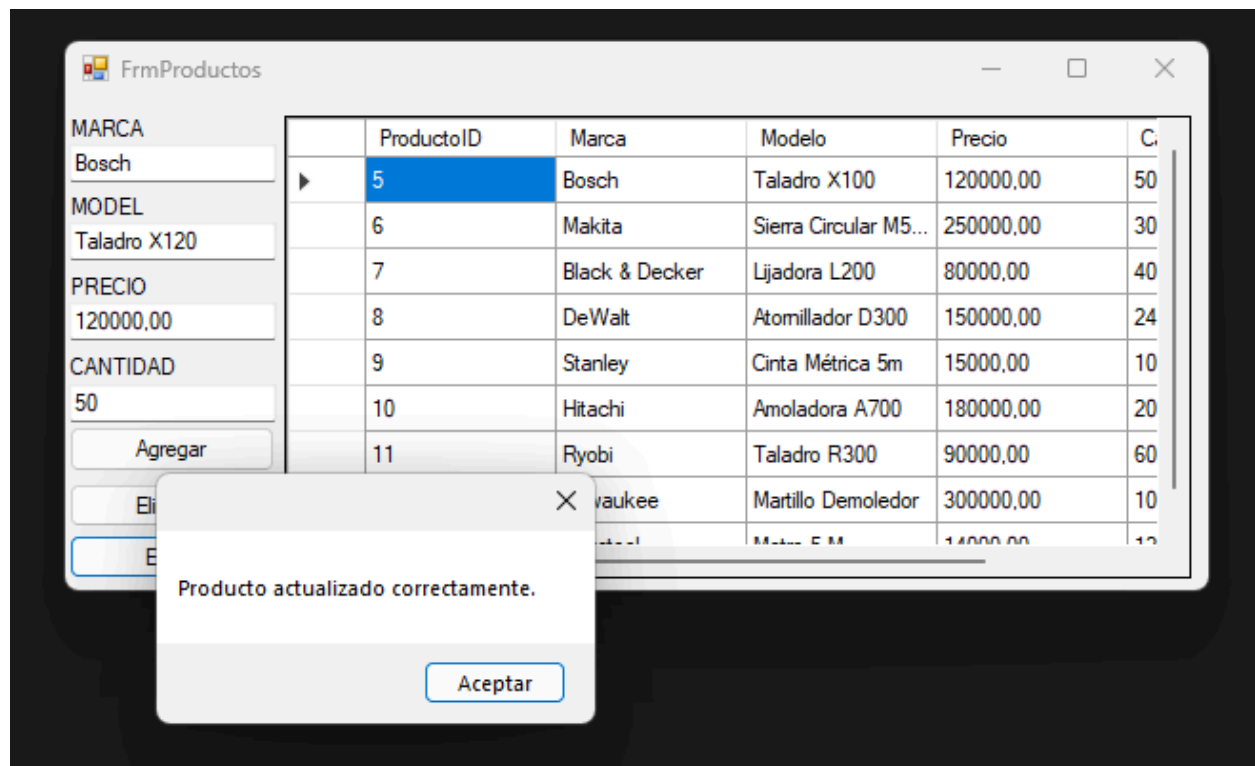
Se lleva un registro histórico de todos los movimientos.

Control de acceso según el rol del usuario.



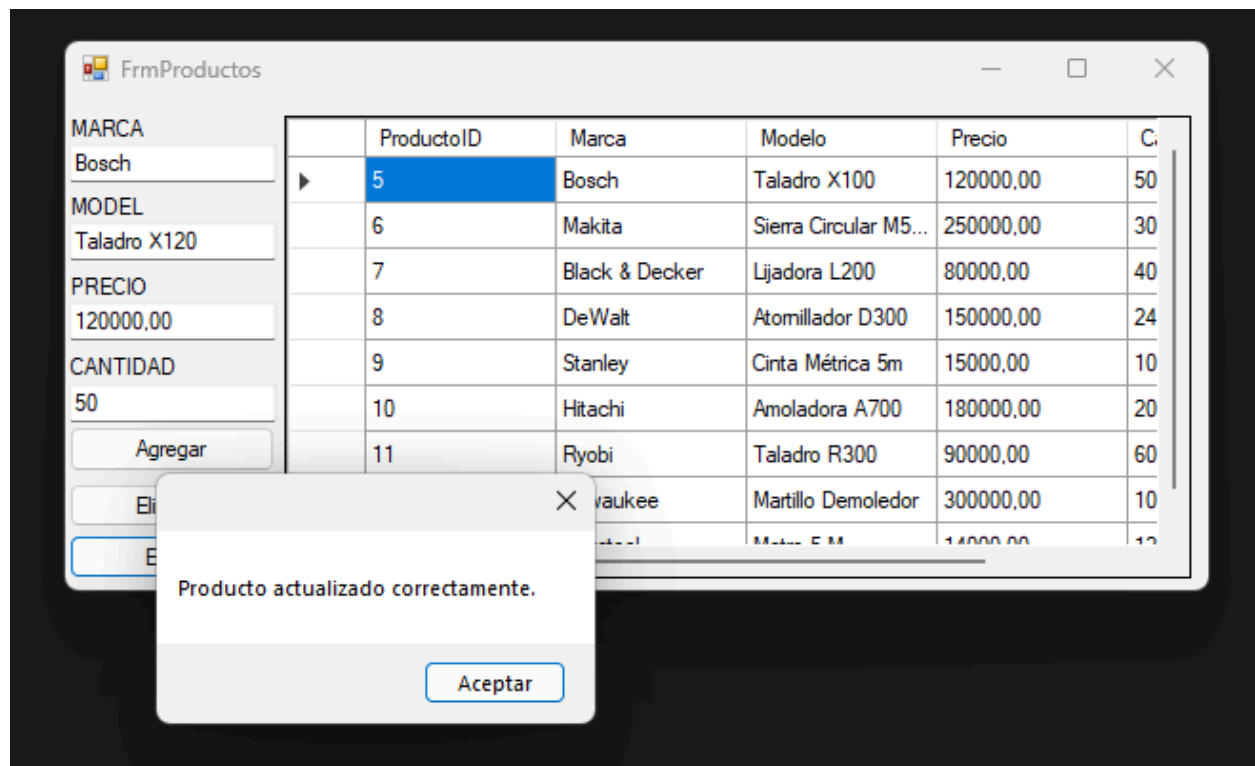
## 7. Capturas de Funcionalidades





### Validaciones y Manejo de Excepciones

La aplicación valida los campos obligatorios y captura errores durante la ejecución de consultas o actualizaciones. Se utilizan bloques try-catch y validaciones previas para asegurar estabilidad.



### Conclusiones del Equipo

El proyecto nos permitió aplicar conceptos de programación orientada a objetos, arquitectura en capas, y conexión a base de datos real.

Implementar roles nos ayudó a comprender la importancia de la seguridad y el acceso restringido.

La lógica de inventario automatizada y las alertas mejoraron la usabilidad y proyección del sistema.

Esta experiencia fortalece nuestras competencias como desarrolladores de software, preparándonos para entornos reales de trabajo.