

Informe de Proyecto: Sistema de gestión en C# WP

Nombre: Lucas Nahuel Cabanillas

Fecha: 26 de febrero de 2025

Institución: Centro Universitario de la Innovación (CUDI)

Este proyecto implementa un sistema de gestión con varios sistemas de carga, actualización de datos y administración de personal, productos y precios. La aplicación está desarrollada en C# usando WPF y conecta con una base de datos para almacenar los respectivos datos ingresados.

1. Introducción

Objetivo del proyecto: El objetivo del proyecto es desarrollar un sistema que permita a los usuarios gestionar todo tipo de información mediante un sistema de gestión de carga de datos, gestionar productos, precios, stock, personal, realizar pedidos y ventas. Además, se permite la gestión de información detallada.

Tecnologías utilizadas:

Lenguaje de programación: C#

Framework: WPF

Base de datos: MySQL

Entorno de desarrollo: Visual Studio

Entorno de desarrollo: La aplicación fue desarrollada en Visual Studio 2022, utilizando el framework .NET 5 para WPF.

2. Metodología

Desarrollo:

La aplicación fue desarrollada utilizando el patrón de diseño MVVM (Model-View-ViewModel), lo que facilita la separación de responsabilidades y mejora la escalabilidad y mantenibilidad del código.

Modelo: Contiene la lógica y las clases que representan las entidades de la base de datos, como ProductosYPrecios, Pedidos, DetallePedidos, etc.

Vista: Las interfaces de usuario fueron creadas usando XAML.

VistaModelo: Gestiona la interacción entre el modelo y la vista, ofreciendo comandos y propiedades que la vista puede enlazar.

Proceso de implementación:

El desarrollo comenzó con la creación de la estructura básica de la interfaz de usuario (pantallas de login, gestión de pedidos, etc.). Luego, se integró la funcionalidad de autenticación mediante un sistema de login, con validación de usuarios y manejo de contraseñas encriptadas. Posteriormente, se implementaron las pantallas para gestionar productos, pedidos y ventas.

Bases de datos:

Se utiliza MySQL para almacenar la información de los pedidos, ventas, personal, productos. Las tablas principales son:

productosyprecios: Contiene información sobre los productos y precios.

pedidos: Almacena información sobre cada pedido, como el cliente y la fecha de entrega.

detalle_pedidos: Almacena los productos asociados a cada pedido, con su cantidad y precio.

ventas: Almacena información sobre cada venta, como el horario que se realizó la venta, peso, total y cantidad.

detalle_ventas: Almacena los productos asociados a cada venta, con su producto, cantidad, precio_unitario, subtotal y peso.

stock: Almacena todos los productos que están a la venta al igual la cantidad disponible de este y su precio unitario

register_login: Almacena información personal de cada empleado, delicada al público conteniendo un método de encriptado para proteger dicha información como dni, nombre, apellido, fechaDeNacimiento, telefono, email, domicilio, localidad, provincia, contraseña, fechaDeContratacion, tipoDeContrato, rango.

3. Descripción del Programa

Vistas (XAML):

La aplicación tiene diversas vistas que permiten al usuario interactuar con el sistema. Algunas de las vistas más importantes son:

1. Login.xaml: Permite la autenticación de los usuarios.

2. ProductosYPrecios.xaml: Muestra y gestiona la lista de productos y sus precios.
3. StockPage.xaml: Gestiona el inventario disponible.
4. Usuarios.xaml: Permite la administración de los usuarios del sistema.
5. Ventas.xaml: Muestra las ventas realizadas y sus detalles.
6. Pedidos.xaml: Permite la creación y modificación de pedidos.

Fragmentos de código clave

Conexión.cs (Conexión a la base de datos):

```
public MySqlConnection conexion()
{
    string server = "127.0.0.1";
    string port = "3306";
    string database = "bd_proyecto_final22";
    string uid = "root";
    string password = "root";

    string cadenaConexion = $"server={server}; port={port};
database={database}; uid={uid}; password={password};";

    try
    {
        MySqlConnection conexionBD = new
        MySqlConnection(cadenaConexion);
        return conexionBD;
    }
    catch (Exception ex)
    {
        Console.WriteLine("Error de conexion." + ex.Message);
        return null;
    }
}
```

4. Características Técnicas

Lenguaje y Frameworks

La aplicación fue desarrollada en C# utilizando el framework WPF para crear interfaces de usuario ricas y reactivas.

Manejo de errores

La aplicación maneja errores comunes como intentos de login fallidos, errores de conexión a la base de datos y excepciones en la entrada de datos. Se emplea el uso de bloques try-catch para capturar excepciones y mostrar mensajes de error adecuados.

Seguridad

Las contraseñas de los usuarios se almacenan de forma encriptada con codificación UTF-8. Al momento de iniciar sesión, la contraseña ingresada por

el usuario se cifra y se compara con la versión cifrada almacenada en la base de datos.

5. Conclusiones

Resultados obtenidos:

El sistema funciona correctamente, permitiendo a los usuarios iniciar sesión y gestionar pedidos de manera eficiente. La aplicación conecta correctamente con la base de datos y gestiona los productos y pedidos según lo esperado.

Dificultades encontradas:

Un desafío fue la implementación de la encriptación de contraseñas y la gestión de sesiones de usuario, lo cual requirió un manejo adecuado de los datos en memoria.

Uno de los mayores problemas fue la creación de los pdf en cual no pude encontrar solución, pero sigo en búsqueda de poder finalmente solucionarlo.

Mejoras futuras

- Stock de insumos para producción.
- Manejo de entrada y salidas de turno.
- Creación de pdf de pedidos para repartos.

6. Referencias

- Documentación de Microsoft: <https://docs.microsoft.com/es-es/dotnet/>
- Tutoriales sobre AES UTF-8 y encriptación: <https://www.youtube.com/watch?v=9PSxKHxfkS0>