

**TECNOLÓGICO NACIONAL DE MÉXICO**

**INSTITUTO TECNOLÓGICO de Nuevo Laredo**

Proyecto final.

**Sistema de cobro y administración para tiendas OXXO.**

Uso de Base de Datos.

**GESTIÓN DE PROYECTOS DE SOFTWARE.**

ING. MARTIN GARIBAY MANRIQUE.

**INGENIERÍA EN SISTEMAS COMPUTACIONALES.**

IVAN ROSALES CANCINO 20100259

JOSE ROBERTO CARRIZALES 21100177

MAURICIO ECHARTEA SILVA 21100191

RAÚL FELIPE GUEVARA MUÑOZ 22100194

CLAVE: SCG1009 GRUPO: H5-06A

**Introducción.**

Este documento describe el desarrollo e implementación de un sistema de cobro y administración de tiendas OXXO con la finalidad de proporcionar una herramienta eficiente para la gestión de sucursales, empleados, proveedores, productos y ventas.

El sistema permitirá que los gerentes administren toda la información relevante, mientras que los empleados podrán realizar ventas de manera rápida y segura.

**Objetivos del plan de calidad.**

**General**

Desarrollar un sistema informático para la administración de empleados y cobro en tiendas OXXO, optimizando la gestión de empleados, sucursales, productos, proveedores y ventas mediante el uso de bases de datos.

**Específicos**

* Implementar una interfaz de usuario intuitiva utilizando C# y Windows Forms.
* Gestionar el acceso a la información según el rol de usuario (gerente o empleado).
* Diseñar y administrar una base de datos en SQL Server para almacenar la información de la tienda.
* Generar reportes de ventas en formato PDF optimizando el tiempo de procesamiento.
* Asegurar la seguridad y el correcto manejo de los datos en el sistema.
* Implementar una función al sistema donde se haga la transferencia de productos entre OXXOS.

**Estándares y normas de calidad.**

**¿CUÁLES EXISTEN?**

**CMMI (Capability Maturity Model Integration)**

Es un modelo de mejora de procesos que proporciona un marco de referencia para la madurez organizacional en el desarrollo de software y otros procesos de negocio.

Objetivo: Mejorar la eficiencia y la calidad de los procesos dentro de una organización, ayudando a gestionar el desarrollo y mantenimiento de productos y servicios.

Niveles de madurez:

* Inicial
* Gestionado
* Definido
* Cuantitativamente Gestionado
* Optimizado

Beneficios: Aumento en la calidad del software, reducción de costos, mejor gestión de proyectos y mayor competitividad.

**MoProSoft (Modelo de Procesos para la Industria de Software en México).**

Es una norma mexicana (NMX-I-059-NYCE-2011) desarrollada para mejorar la calidad de las empresas de software en México, estableciendo un modelo de referencia para la gestión de procesos.

Objetivo: Promover la competitividad de las empresas de desarrollo de software mediante la estandarización de sus procesos.

Estructura: Dividida en tres niveles de organización:

* Alta Dirección
* Gestión de Operación
* Operación

Beneficios: Incremento en la productividad, mejora en la calidad de los productos, facilitación de certificaciones internacionales como CMMI e ISO.

**ISO/IEC 25010 (Evaluación de la Calidad de los Productos de Software)**

ISO/IEC 25010 es una norma internacional que establece un modelo de calidad para los productos de software. Define un conjunto de características que se deben evaluar para medir la calidad de un software.

Las principales características que evalúa son:

1. Funcionalidad: Si el software cumple con los requisitos y proporciona las funciones necesarias. En este caso, las funciones de gestión de ventas, productos y empleados.
2. Fiabilidad: Si el software funciona de manera consistente sin fallos. Es importante asegurarse de que el sistema no se caiga y maneje errores adecuadamente.
3. Usabilidad: Qué tan fácil es para los usuarios interactuar con el sistema. Aquí se consideraría la facilidad de uso para los empleados y gerentes de Oxxo.
4. Eficiencia: Cuán eficiente es el sistema en términos de recursos, como el uso de CPU, memoria, etc. Esto incluye el tiempo de respuesta en las ventas o la gestión de productos.
5. Mantenibilidad: Qué tan fácil es corregir defectos, hacer mejoras o añadir nuevas funciones. Este punto se refiere a la facilidad con la que puedes actualizar y mejorar tu sistema.
6. Portabilidad: Qué tan fácil es transferir el software a otros entornos o plataformas.

Aplicación práctica en el proyecto:

* Asegúrate de realizar pruebas de funcionalidad para confirmar que todas las operaciones del sistema (ventas, gestión de productos, etc.) están correctas.
* Lleva a cabo pruebas de fiabilidad para comprobar que el sistema funciona sin caídas durante las operaciones diarias.
* Fomenta la usabilidad mediante pruebas de usuarios reales para asegurar que la interfaz sea fácil de usar tanto para empleados como para gerentes.
* Evalúa y mejora la eficiencia para garantizar que el sistema pueda manejar múltiples transacciones simultáneamente sin problemas de rendimiento.

**PCI DSS (Payment Card Industry Data Security Standard)**

PCI DSS es un conjunto de estándares de seguridad globales diseñados para proteger la información de tarjetas de crédito. Se aplica a todas las organizaciones que almacenan, procesan o transmiten información de tarjetas de pago.

Características que evalua:

* Seguridad de los datos de pago: Si el sistema manejará pagos con tarjeta, debes asegurarte de que todos los datos de las tarjetas de crédito estén protegidos. Esto incluye el cifrado de la información sensible y el cumplimiento de los requisitos de almacenamiento seguro.
* Autenticación y control de acceso: Solo las personas autorizadas deben poder acceder a la información sensible de las transacciones.
* Monitoreo y auditoría: Debes implementar un sistema de monitoreo para detectar actividades sospechosas relacionadas con el manejo de datos de tarjetas de crédito.

Aplicación práctica en el proyecto:

* Si el sistema va a incluir pagos con tarjeta, deberías implementar cifrado de datos al almacenar o procesar información de tarjetas.
* Se asegurará de que las transacciones sean seguras utilizando métodos como tokenización.
* Realiza auditorías de seguridad y revisa regularmente el cumplimiento de las normas PCI DSS en todas las transacciones de pago.
* ISO/IEC 25010: Medir y evaluar las características de calidad del software como funcionalidad, fiabilidad, usabilidad y eficiencia. Nos aseguraremos de que el sistema cumpla con los requisitos funcionales y de calidad.
* PCI DSS: Si el sistema incluye pagos con tarjeta de crédito, se implementara medidas de seguridad estrictas para proteger los datos de pago.

**ISO/IEC 25010 (Evaluación de la Calidad de los Productos de Software)**

Aplicable: Sí.

Razón:

* Calidad del software: Este estándar establece un marco claro para evaluar y mejorar la calidad del software, lo cual es crucial para un sistema de gestión de ventas y administración de productos. Debes asegurarte de que el sistema sea funcional, confiable, fácil de usar, eficiente y seguro.
* Evaluación de calidad en diferentes áreas: ISO/IEC 25010 permite evaluar la calidad del software en diferentes aspectos, como funcionalidad, fiabilidad, usabilidad y eficiencia, lo cual es esencial para garantizar un buen desempeño en el sistema de ventas y manejo de inventarios.

Cómo aplicarlo:

* Realizar pruebas de funcionalidad para asegurar que el sistema cumpla con los requisitos del cliente.
* Evaluar la fiabilidad, usabilidad y rendimiento del sistema.
* Implementar un proceso de control de calidad durante todo el ciclo de vida del software.

**PCI DSS (Payment Card Industry Data Security Standard)**

Aplicable: Si

Razón:

* Si el sistema maneja pagos con tarjeta de crédito: Si el sistema de cobro incluye transacciones con tarjetas de crédito (por ejemplo, si se permite pagar con tarjeta en las tiendas Oxxo), entonces PCI DSS es totalmente aplicable. Esta norma proporciona pautas estrictas sobre cómo manejar los datos de tarjetas de crédito de forma segura.
* Si no maneja pagos con tarjeta: Si solo se manejan transacciones en efectivo o no se procesan pagos directamente con tarjeta de crédito, entonces PCI DSS no sería relevante para el proyecto. El sistema aún puede necesitar consideraciones de seguridad, pero no será necesario implementar PCI DSS en este caso.

Cómo aplicarlo:

* Implementar medidas de seguridad como el cifrado de datos para proteger la información de las tarjetas de crédito.
* Asegurarse de que los datos de pago se manejen según los estándares de seguridad de PCI DSS, protegiendo las tarjetas y otros datos sensibles.
* Auditar el sistema regularmente para verificar que se cumpla con las normativas de seguridad.

**Requisitos del sistema.**

**Requisitos Funcionales**

* Inicio de sesión con roles (Gerente y Empleado).
* Registro y administración de empleados.
* Administración de sucursales.
* Gestión de proveedores.
* Control y registro de productos.
* Realización de ventas.
* Generación de reportes de ventas en PDF.

**Requisitos No Funcionales**

* Seguridad en el acceso a datos mediante roles.
* Interfaz amigable y sencilla.
* Conexión con base de datos SQL Server.
* Uso de C# y Windows Forms para la interfaz gráfica.

**Modelado de base de datos.**

La base de datos consta de 7 tablas principales:

* Empleados (ID, Nombre, Apellido, Cargo, Fecha de Ingreso, SucursalID, etc.)
* Sucursales (ID, Nombre, Dirección, Teléfono, etc.)
* Proveedores (ID, Nombre, Contacto, Teléfono, Email, etc.)
* Productos (ID, Nombre, Categoría, Precio, ProveedorID, Stock, etc.)
* Ventas (ID, EmpleadoID, SucursalID, Fecha, Total, etc.)
* VentaProductos (ID, VentaID, ProductoID, Cantidad, Subtotal, etc.)
* Disponibilidad (ID, (ID, ProductoID, SucursalID, Cantidad)