JIBBER

**Sistema de Inventarios**

UNIVERSIDAD DE TENANCINGO

UdeT

MATERIA:

**Análisis y Diseño Orientado a Objetos**

DOCENTE:

**Profesor Roberto Mijares Guerrero**

ALUMNO:

**Diego Flores González**

**QUINTO SEMESTRE**

Índice

[INTRODUCCIÓN 4](#_Toc187950046)

[OBJETIVO 5](#_Toc187950047)

[PROBLEMÁTICA 6](#_Toc187950048)

[JUSTIFICACIÓN 7](#_Toc187950049)

[MARCO TEÓRICO 8](#_Toc187950050)

[ACTIVIDADES Y RESULTADOS 12](#_Toc187950051)

[Diagrama de clases 16](#_Toc187950052)

[Diagrama de clases de usos 18](#_Toc187950054)

[Código 22](#_Toc187950056)

[Formulario de Requisitos del Sistema de Gestión de Inventarios 24](#_Toc187950057)

[CONCLUSIONES Y RECOMENDACIONES 27](#_Toc187950058)

[REFERENCIAS 30](#_Toc187950059)

# INTRODUCCIÓN

El presente trabajo escrito aborda el desarrollo de un sistema de gestión de inventarios, motivado por la creciente necesidad de las pequeñas y medianas empresas (PYMES) de optimizar sus procesos de control y administración de recursos. Como estudiante de la Licenciatura en Ingeniería en Computación, observé que muchos negocios locales enfrentan dificultades para mantener registros precisos de su inventario, lo que ocasiona problemas como sobre existencias o desabastecimientos. Estas situaciones no solo afectan la eficiencia operativa, sino también la rentabilidad y la satisfacción del cliente. Esto me impulsó a investigar y desarrollar una solución tecnológica que atienda esta problemática.

Para abordar este proyecto, utilicé un enfoque metodológico basado en los principios del desarrollo de software ágil, que permite una construcción iterativa y adaptativa del sistema. Los criterios empleados incluyeron la usabilidad, la escalabilidad y la seguridad, siendo este último un aspecto clave dado el manejo de información sensible. En cuanto a los procedimientos, se realizó un análisis de requerimientos a través de entrevistas con pequeños empresarios, seguido de un diseño del sistema utilizando herramientas como Visual Studio y SQL Server para implementar la base de datos. Finalmente, se efectuaron pruebas funcionales para garantizar el correcto funcionamiento del sistema.

El trabajo se estructura en varias secciones. En la justificación, se expone la importancia de mejorar la gestión de inventarios en el contexto actual y los beneficios esperados de implementar un sistema como el desarrollado. En el marco teórico, se describen los conceptos y fundamentos técnicos que sustentan el proyecto, incluyendo temas de bases de datos. En la metodología, se detallan los pasos seguidos para el diseño e implementación del sistema. En los resultados, se presentan las pruebas realizadas y los hallazgos obtenidos. Finalmente, en las conclusiones y recomendaciones, se reflexiona sobre el impacto del proyecto y se sugieren mejoras futuras para optimizar el sistema y adaptarlo a otras necesidades empresariales.

# OBJETIVO

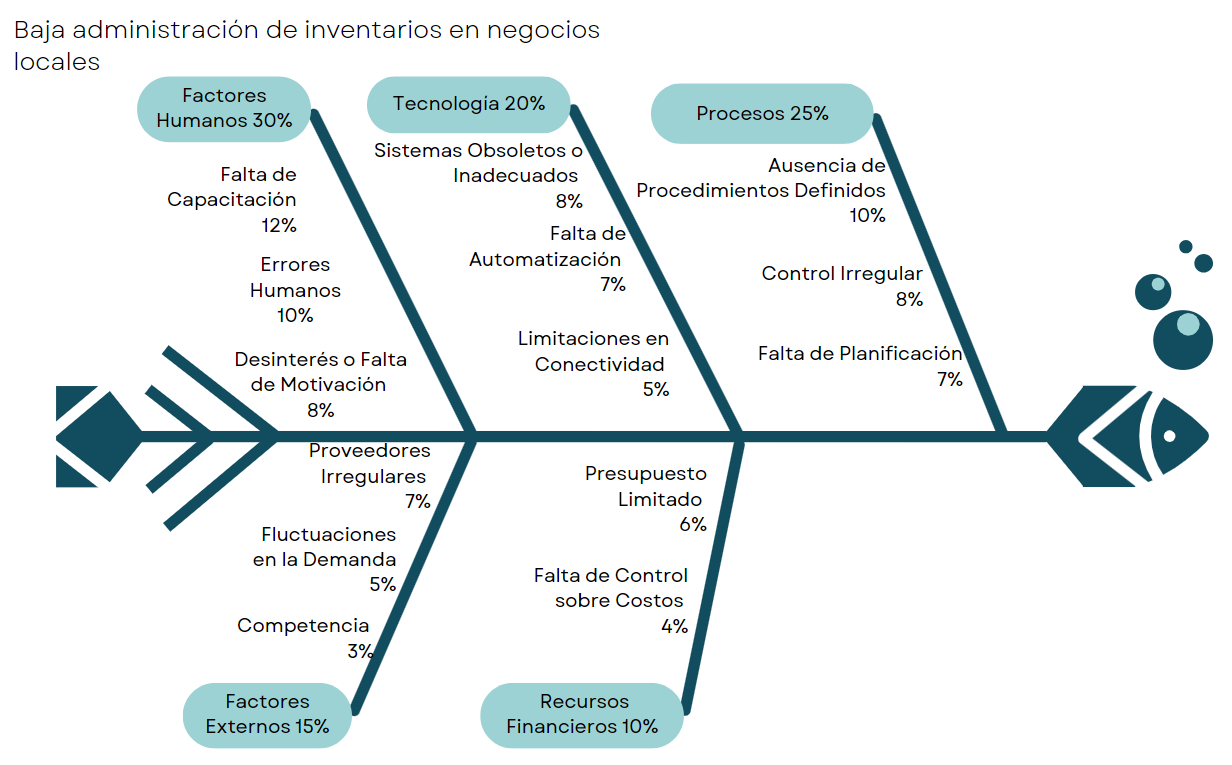
**Objetivo general:** Desarrollar un sistema de gestión de inventarios eficiente, seguro y escalable que permita a las pequeñas y medianas empresas (PYMES) optimizar sus procesos operativos y mejorar su el manejo de sus recursos materiales.

**Objetivos específicos:**

* Realizar un análisis detallado de los requerimientos y necesidades de los negocios locales para asegurar que el sistema cumpla con sus expectativas y demandas.
* Diseñar e implementar una base de datos robusta y funcional que permita el almacenamiento y manejo seguro de la información de inventarios.
* Evaluar y validar el funcionamiento del sistema mediante pruebas funcionales, asegurando su usabilidad y efectividad antes de su implementación final.

# PROBLEMÁTICA

En la zona sur del Estado de México, las pequeñas y medianas empresas enfrentan dificultades en la gestión de inventarios, lo que resulta en desabastecimiento o exceso de stock, errores en el control de productos, y una falta de visibilidad en los niveles de inventario. Además, no cuenta con un sistema eficiente para actualizar el inventario automáticamente tras las ventas, lo que lleva a pérdidas de ventas por falta de disponibilidad de productos y problemas en la relación con proveedores por pedidos mal gestionados. Esta situación será reducidos significativamente, lo que impacta en las ganancias y generación de nuevos insumos e impacta directamente en la satisfacción del cliente y en los costos operativos de la empresa.



# JUSTIFICACIÓN

El desarrollo de un sistema de gestión de inventarios es fundamental para optimizar las operaciones de la empresa, asegurar una administración eficiente del stock y reducir errores humanos en el control de productos. Actualmente, la empresa enfrenta problemas de desabastecimiento y exceso de inventario, lo que genera pérdidas de ventas y costos adicionales de almacenamiento.

Un sistema automatizado permitirá actualizar el inventario en tiempo real, especialmente después de cada venta o recepción de suministros, lo que ayudará a mantener niveles adecuados de stock y a mejorar la precisión en los pedidos a proveedores. Además, proporcionará herramientas para generar reportes, gestionar el historial de productos y analizar datos, facilitando la toma de decisiones y permitiendo identificar productos con mayor rotación o necesidades de abastecimiento.

Un aspecto crucial que aborda este sistema es el fortalecimiento de la seguridad en las ventas, garantizando un control más riguroso sobre las actividades realizadas por los vendedores. Mediante la asignación de usuarios específicos para cada transacción, el sistema registra quién realizó cada operación, permitiendo detectar irregularidades o actos de deshonestidad. Este control fomenta la responsabilidad individual y ayuda a prevenir prácticas indebidas, como el registro incorrecto de ventas o el mal manejo del inventario.

Adicionalmente, al proporcionar reportes detallados y rastreables, el sistema facilita la identificación de discrepancias entre las ventas registradas y los movimientos del inventario. Esto no solo incrementa la transparencia en las operaciones, sino que también genera confianza en la administración y asegura que los recursos de la empresa estén protegidos contra pérdidas derivadas de posibles fraudes internos.

Implementar este sistema también mejorará la relación con los clientes al garantizar una disponibilidad constante de productos y reducir los tiempos de respuesta. A nivel interno, el sistema facilitará el trabajo de los empleados, incrementando la productividad y minimizando el riesgo de errores.

En términos generales, este proyecto contribuirá a una mejor organización, eficiencia y competitividad de la empresa, proporcionando una herramienta esencial para enfrentar las demandas del mercado, optimizar los recursos y reforzar la seguridad en las operaciones comerciales. Esto asegura un entorno más confiable tanto para los empleados como para los clientes, protegiendo los intereses de la empresa y fortaleciendo su sostenibilidad a largo plazo.

# MARCO TEÓRICO

El marco teórico del presente trabajo se basa en los conceptos fundamentales que sustentan el desarrollo de un sistema de gestión de inventarios sencillo pero funcional, empleando herramientas accesibles como Visual Studio y SQL Server. A continuación, se detallan los aspectos principales:

CAPÍTULO 1: INVENTARIOS

1.1. ¿Qué son los inventarios?

Los inventarios son el conjunto de bienes tangibles que una empresa posee para la venta, el uso en la producción o el mantenimiento de operaciones. Incluyen materias primas, productos en proceso y productos terminados.

1.2. ¿Qué es un stock?

El stock es el término utilizado para describir las cantidades disponibles de productos en inventario. Este puede ser:

* Stock de seguridad: Cantidad adicional mantenida para cubrir variaciones en la demanda.
* Stock mínimo: Nivel más bajo necesario para evitar desabastecimientos.
* Stock máximo: Cantidad máxima que puede almacenarse sin incurrir en costos excesivos.

1.3. Tipos de inventarios

* Inventario periódico: Actualizado en intervalos de tiempo regulares.
* Inventario perpetuo: Actualizado en tiempo real tras cada transacción.
* Inventario de seguridad: Destinado a cubrir imprevistos.

CAPÍTULO 2: BASES DE DATOS

2.1. ¿Qué es una base de datos?

Una base de datos es un sistema organizado que permite almacenar, gestionar y recuperar información de manera eficiente.

2.2. Tablas

Las tablas son estructuras fundamentales en una base de datos que organizan los datos en filas y columnas.

* Fila: Representa un registro único.
* Columna: Representa un atributo o campo de los datos almacenados.

2.3. Relaciones entre tablas

En una base de datos relacional, las tablas están conectadas mediante relaciones, que pueden ser de uno a uno, uno a muchos o muchos a muchos.

2.4. Consultas

Las consultas son solicitudes de información específica a una base de datos mediante un lenguaje como SQL. Ejemplo:

SELECT \* FROM productos WHERE cantidad < 10;

CAPÍTULO 3: SISTEMAS DE VENTAS

3.1. ¿Qué es un sistema de ventas?

Un sistema de ventas es una herramienta tecnológica que permite registrar, procesar y analizar las transacciones comerciales de una empresa.

3.2. Componentes principales

* Interfaz de usuario: Donde los empleados interactúan con el sistema.
* Base de datos: Donde se almacena toda la información sobre productos, ventas y clientes.
* Reportes: Generación de documentos que resumen el estado de las operaciones.

3.3. Funcionalidades básicas

* Registro de productos vendidos.
* Emisión de tickets o facturas.
* Actualización automática del inventario tras cada venta.

CAPÍTULO 4: SEGURIDAD EN LAS VENTAS

4.1. Importancia del control de usuarios

El control de usuarios garantiza que cada transacción esté vinculada a una persona específica, lo que permite rastrear irregularidades y prevenir fraudes.

4.2. Medidas de seguridad

* Registro de usuarios: Cada empleado tiene un inicio de sesión único.
* Historial de transacciones: Se registra quién realizó cada operación.
* Restricción de permisos: Solo usuarios autorizados pueden realizar ciertas acciones, como eliminar productos o modificar datos.

4.3. Auditorías internas

Las auditorías consisten en revisar periódicamente las operaciones registradas para detectar errores o posibles actos de deshonestidad.

CAPÍTULO 5: METODOLOGÍA

5.1. Análisis de requerimientos

Se recopilaron las necesidades del sistema a través de entrevistas con el cliente, identificando las funcionalidades esenciales, como:

* Registro de productos y ventas.
* Generación de reportes.
* Actualización automática del inventario.

5.2. Diseño del sistema

Se utilizaron herramientas como Visual Studio para la interfaz gráfica y SQL Server para la base de datos. El diseño incluyó:

* Diagramas de tablas para modelar la estructura de la base de datos.
* Diagramas de flujo para representar los procesos principales.

5.3. Implementación

El sistema se desarrolló en módulos, permitiendo la incorporación progresiva de funcionalidades.

5.4. Pruebas y validación

Se realizaron pruebas funcionales para garantizar que el sistema respondiera adecuadamente en escenarios reales, incluyendo:

* Registro y consulta de productos.
* Generación de reportes y tickets.
* Actualización en tiempo real del inventario.

# ACTIVIDADES Y RESULTADOS

**Actividades y Resultados**

Para el desarrollo del sistema, se llevaron a cabo las siguientes actividades:

1. Análisis de requerimientos: Se identificaron las necesidades específicas de los usuarios a través de entrevistas.
2. Diseño del sistema: Se elaboraron diagramas de clases y casos de uso que definieron la estructura y funcionalidades del sistema.
3. Implementación: Se desarrolló el sistema utilizando Visual Studio y SQL Server para la gestión de datos.
4. Pruebas funcionales: Se validaron las funcionalidades implementadas, asegurando su correcto desempeño en diferentes escenarios.

Los resultados incluyen un sistema funcional que automatiza la gestión de inventarios, mejora la precisión en el control de stock y facilita la generación de reportes para la toma de decisiones empresariales.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| NO | Nombre | Descripción | Categoría | | |  |
| C | MC | FO | Pij |
| C1 | Falta de capacitación | Los empleados no reciben suficiente formación en el manejo de inventarios | x |  |  | 20% |
| C1.1 | Ausenca de programas de formación | No existen programas regulares para formar a los empleados en gestión de inventarios | x |  |  | 10% |
| C1.2 | No hay presupuesto para capacitaciones | No se destinan a recursos económicos para la formación | x |  |  | 10% |
| C2 | Mala gestión de inventarios | Los empleados no gestionan correctamente los inventarios debido a procesos deficientes |  | x |  | 30% |
| C2.1 | Errores Humanos | Distracciones o falta de conocimiento técnico llevan a errores de registro |  | x |  | 10% |
| C2.2 | Desinterés o falta de motivación | Los empleados no están motivados para llevar un control adecuado del inventario |  | x |  | 10% |
| C2.3 | Falta de procedimientos claros | No existen políticas o procedimientos claramente definidos para gestionar el inventario | x |  |  | 10% |
| C3 | Sistemas Obsoletos o Inadecuados | Las herramientas tecnológicas utilizadas no son suficientes para el control optimo de inventarios | x |  |  | 15% |
| C3.1 | Software desactualizado | El sistema utilizado no es moderno y presenta muchas limitaciones | x |  |  | 10% |
| C3.2 | Falta de automatización | No se utilizan herramientas automáticas para gestionar el inventario | x |  |  | 5% |
| C4 | Falta de información Oportuna | Los datos sobre el inventario no llegan a tiempo, afectando decisiones |  |  | x | 10% |
| C4.1 | Limitaciones en la Conectividad | Mala conectividad a internet que afecta el uso de sistemas digitales |  |  | x | 10% |
| C4.2 | Control Irregular | No se realizan controles de inventario con la frecuencia requerida |  |  | x | 10% |
| C4.3 | Falta de Planificación | No se planifica correctamente la entrada y salida de productos |  |  | x | 5% |
| C5 | Problemas con Proveedores | Las fallas de los proveedores afectan la disponibilidad de inventario |  |  | x | 10% |
| C5.1 | Proveedores irregulares | Los proveedores no entregan productos a tiempo |  |  | x | 5% |
| C5.2 | Fluctuaciones en la Demanda | Cambios en la demanda que no son previstos afectan al inventario |  |  | x | 5% |

La tabla es un **análisis categorizado de problemas en la gestión de inventarios**. Este tipo de tabla suele utilizarse para identificar y clasificar las causas principales de problemas operativos en un sistema, como parte de un diagnóstico o análisis de procesos.

**Explicación de los elementos de la tabla:**

1. **NO**  
   Es un identificador numérico que asigna un número único a cada causa principal o subcausa. Ejemplo: C1, C1.1, C2, etc.
2. **Nombre**  
   Describe el problema o la causa principal identificada. Ejemplo: *Falta de capacitación*, *Mala gestión de inventarios*.
3. **Descripción**  
   Proporciona un detalle más específico del problema identificado. Ejemplo: *Los empleados no reciben suficiente formación en el manejo de inventarios*.
4. **Categoría**  
   Clasifica las causas según su tipo. Las abreviaturas pueden tener significados específicos, como:
   * **C**: Causas relacionadas con el **Conocimiento** o la **Capacitación**.
   * **MC**: Causas relacionadas con la **Mala Comunicación**.
   * **FO**: Causas vinculadas con **Factores Organizacionales**.
   * **Pij**: Causas relacionadas con **Proveedores o Infraestructura Jurídica**.
5. **Subcausas**  
   Las filas con identificadores como C1.1, C2.1, etc., corresponden a subcausas o aspectos más específicos dentro de una causa principal. Ejemplo:
   * *C1: Falta de capacitación* (causa principal).
   * *C1.1: Ausencia de programas de formación* (subcausa).
   * *C1.2: No hay presupuesto para capacitaciones* (subcausa).
6. **Porcentaje (%)**  
   Indica la proporción de importancia o impacto que cada causa tiene en el problema general. Por ejemplo:
   * *C1 tiene un impacto del 20%*.
   * *C1.1 y C1.2 contribuyen con un 10% cada una al impacto total de C1*.

**Propósito de la tabla:**

* Identificar los problemas clave que afectan la gestión de inventarios.
* Clasificar las causas y subcausas para entender las áreas específicas que necesitan mejoras.
* Priorizar las acciones correctivas basándose en la relevancia de cada problema (representado por los porcentajes).

## Diagrama de clases

Un **diagrama de clases** es un modelo gráfico utilizado en el diseño de sistemas orientados a objetos para representar la estructura estática del sistema. Este diagrama muestra las clases, sus atributos, métodos y las relaciones entre ellas. Es una de las herramientas clave en el lenguaje UML (Unified Modeling Language).

**Propósito:**

El diagrama de clases se utiliza para:

1. **Definir la estructura del sistema:** Describe las entidades principales (clases) y cómo interactúan entre sí.
2. **Modelar relaciones:** Representa asociaciones, agregaciones, composiciones y herencias entre clases.
3. **Planificar la base del código:** Actúa como un esquema para los programadores al implementar las clases en el sistema.
4. **Comunicar ideas:** Facilita la comprensión del diseño entre desarrolladores, analistas y otros interesados.

**Elementos clave:**

* **Clases:** Representadas como rectángulos divididos en tres secciones: nombre, atributos y métodos.
* **Relaciones:**
  + **Asociación:** Representa una conexión entre clases (línea simple).
  + **Herencia:** Una clase deriva de otra (línea con un triángulo).
  + **Composición:** Una clase está compuesta de otras partes (línea con un rombo relleno).
  + **Agregación:** Una relación más débil que la composición (línea con un rombo vacío).

En esta imagen se representa el diagrama de clases de nuestro sistema de ventas.

## 

# Diagrama de clases de usos

Un **diagrama de casos de uso** es una representación gráfica que describe cómo los usuarios (actores) interactúan con un sistema para cumplir con sus objetivos. Se enfoca en **qué** hace el sistema, destacando sus funcionalidades principales desde la perspectiva del usuario.

**Propósito:**

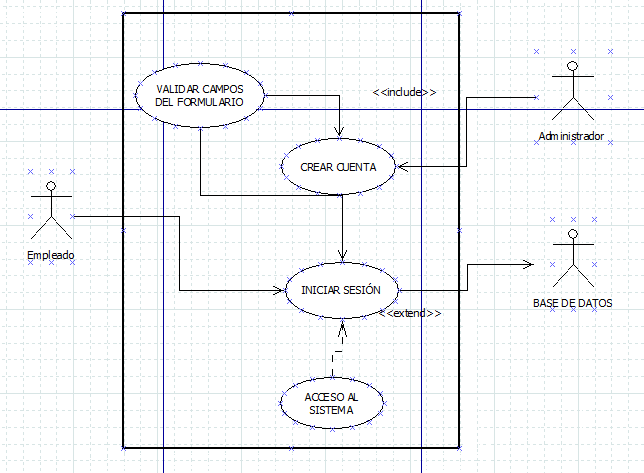
Los diagramas de casos de uso son esenciales para:

1. **Identificar los requerimientos del sistema:** Ayudan a documentar las funcionalidades que debe ofrecer el sistema para satisfacer las necesidades de los usuarios.
2. **Entender la interacción usuario-sistema:** Muestran cómo los usuarios interactúan con las funcionalidades del sistema.
3. **Comunicar ideas:** Sirven como una herramienta visual que facilita la comunicación entre desarrolladores, analistas y clientes.
4. **Definir los límites del sistema:** Especifican qué funcionalidades están dentro o fuera del alcance del proyecto.

**Elementos clave:**

1. **Actores:** Representan a los usuarios del sistema o a otros sistemas externos que interactúan con el sistema principal. Pueden ser personas (usuarios finales) o entidades externas (como un sistema de pago). Se representan como figuras de un "muñeco palo".
2. **Casos de Uso:** Representan las funcionalidades o procesos del sistema que los actores pueden ejecutar. Se dibujan como elipses con un nombre descriptivo dentro.
3. **Sistema:** Un rectángulo que delimita el alcance del sistema. Los casos de uso se ubican dentro del rectángulo, mientras que los actores están fuera.
4. **Relaciones:**
   * **Asociación:** Conexión entre un actor y un caso de uso.
   * **Inclusión:** Un caso de uso incluye la funcionalidad de otro.
   * **Extensión:** Un caso de uso extiende el comportamiento de otro, pero no es necesario ejecutarlo siempre.

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de uso | INICIAR SESIÓN DE EMPLEADO | | |
| ID | 1 | FECHA | 10/25/2024 |
| Autor | ING. DIEGO FLORES GONZÁLEZ | | |
| Descripción | El empleado debe iniciar sesión para tener acceso al sistema y así realizar diferentes operaciones como la celebración de una venta. | | |
| Actores | El empleado debe iniciar sesión en el sistema.  El administrador de sistema debe dar de alta al empleado para que tenga acceso adecuado. | | |
| Pre-Condiciones | El administrador del sistema debe dar de alta al empleado y otorgar los permisos necesarios. | | |
| Post-Condiciones | El empleado debe estar registrado en el sistema con un número de empleado y contraseña asignados. | | |
| Fllujo Principal | 1. El administrador debe dar de alta al empleado ingresando los datos del número de empleado y su contraseña. 2. El administrador debe verificar que el empleado esté registrado en la base de datos. 3. El sistema debe mostrar la pantalla principal de inicio de sesión. 4. El empleado debe ingresar su número de empleado y contraseña. 5. El sistema debe verificar que los datos y la fecha ingresados sean correctos. 6. El sistema debe mostrar una pantalla de bienvenida con el número de empleado, la fecha, y un catálogo de productos. | | |
| Flujo Alternativo | El sistema puede iniciar en modo administrador bajo la supervisión de un administrador  Verificar que el empleado haya ingresado correctamente su número y contraseña. En ocasiones el teclado puede estar en mayúsculas. El administrador también debe verificar la conexión de red de la base de datos. | | |

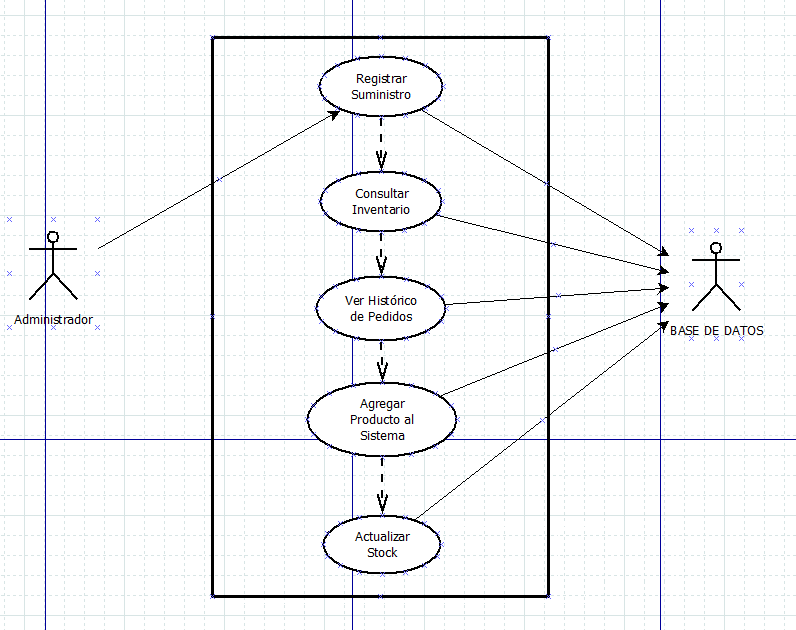


## 

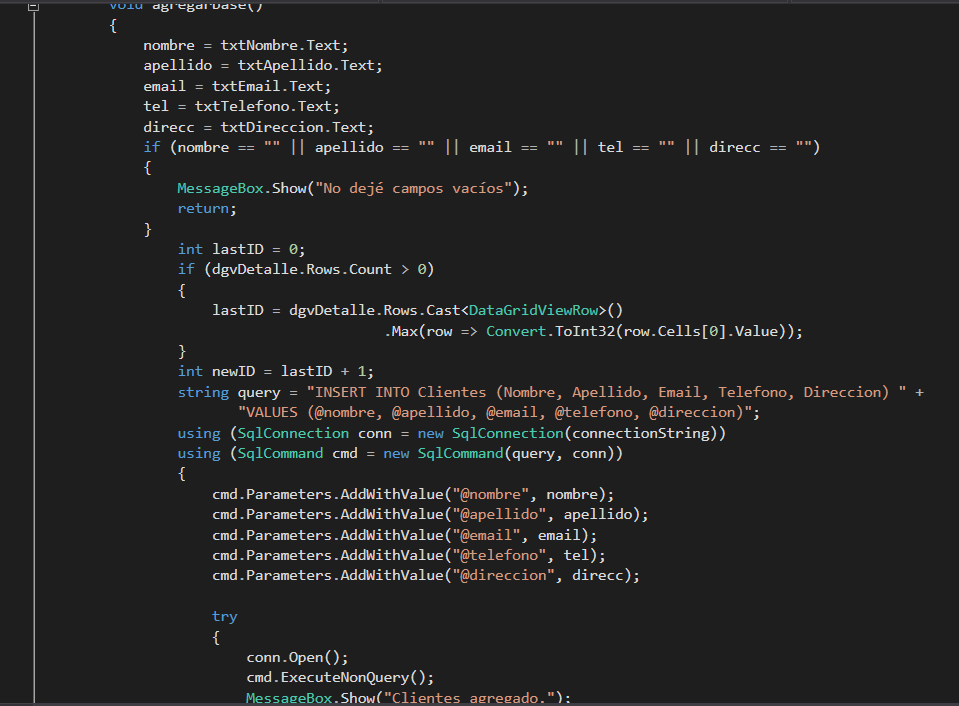
## 

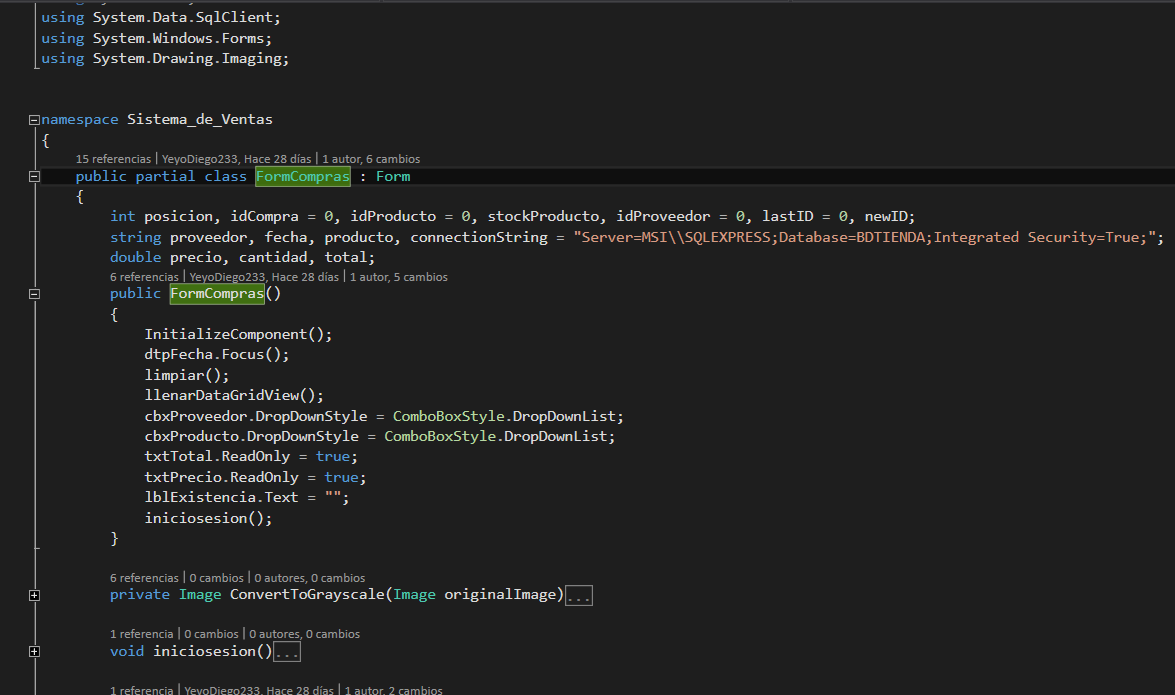
|  |  |  |  |
| --- | --- | --- | --- |
| Caso de uso | REALIZAR PEDIDO | | |
| ID | 2 | FECHA | 10/25/2024 |
| Autor | ING DIEGO FLORES GONZÁLEZ | | |
| Descripción | El empleado accede al sistema, busca el producto a vender en la lista de productos y procesa el pedido ingresando la información solicitada, como la cantidad. Una vez completada la operación, se procesa el pago, se genera una factura y el stock se actualiza automáticamente con la cantidad vendida. | | |
| Actores | - El empleado  - El sistema de base de datos para buscar productos, generar la factura y actualizar el stock | | |
| Pre-Condiciones | El sistema debe verificar que haya suficiente cantidad del producto disponible antes de procesar la venta. En caso de no haber suficiente stock, el sistema debe notificarlo antes de proceder. | | |
| Post-Condiciones | Tras completar el pedido y generar la factura, el empleado procede a cerrar el sistema. | | |
| Flujo Principal | 1. El empleado busca el producto en el sistema  2. Se verifica que el stock es suficiente  3. El empleado procesa el pago  4. Se genera la factura automáticamente  5. El sistema actualiza el stock | | |
| Flujo Alternativo | Si el sistema está en modo administrador, se puede realizar el pedido sin restricciones. En caso de no tener acceso al sistema, el pedido se puede realizar de manera manual en una hoja, sin generar factura automáticamente. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de uso | REGISTRAR SUMINISTRO | | |
| ID | 3 | FECHA | 10/25/2024 |
| Autor | ING DIEGO FLORES GONZÁLEZ | | |
| Fecha | Cuando el dueño haya solicitado un reabastecimiento de los productos | | |
| Descripción | El administrador accede al sistema en modo administrador, consulta el inventario y verifica qué productos hacen falta. Luego, revisa el histórico de pedidos realizados. Una vez que el proveedor ha entregado el suministro, el administrador ingresa los nuevos productos al sistema, y el stock se actualiza automáticamente. | | |
| Actores | - Administrador  - Sistema de base de datos que proporciona la información de inventario y pedidos | | |
| Pre-Condiciones | El sistema debe tener un inventario base previamente registrado y productos disponibles. Además, el administrador debe haber recibido los nuevos productos para proceder a la actualización del stock. | | |
| Post-Condiciones | Tras actualizar el stock con los nuevos productos, el administrador debe cerrar el sistema. | | |
| Flujo Principal | 1. El administrador consulta el inventario en el sistema  2. Verifica qué productos hacen falta  3. Revisa el histórico de pedidos para evaluar el rendimiento del producto  4. Agrega los nuevos productos entregados por el proveedor  5. El sistema actualiza el stock automáticamente | | |
| Flujo Alternativo | No se especifica en esta tabla un flujo alternativo | | |

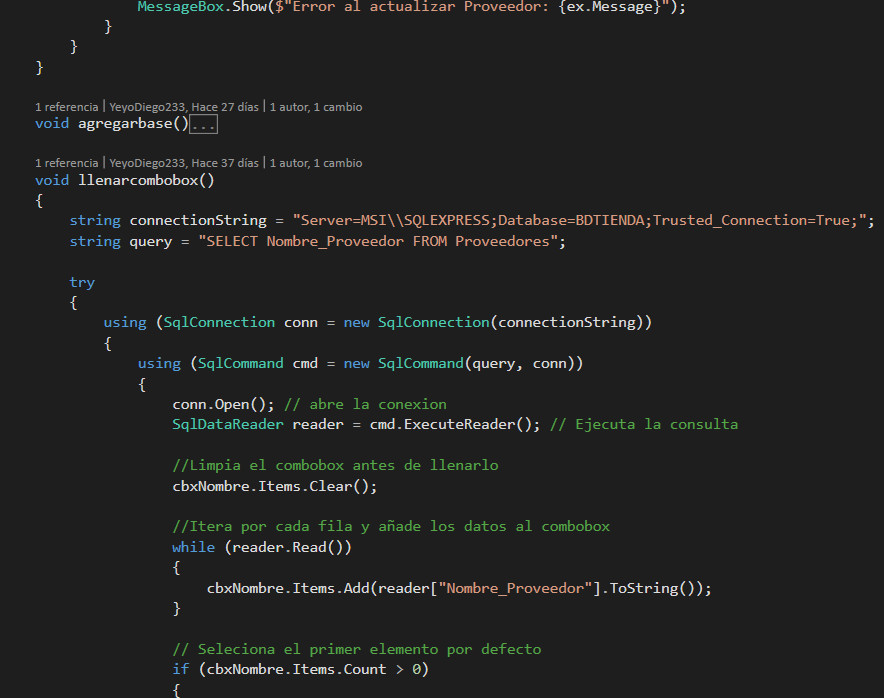


## Código









## Formulario de Requisitos del Sistema de Gestión de Inventarios

*Propósito: Recopilar información clave para garantizar que el sistema desarrollado cumpla con las necesidades del cliente.*

**1. ¿Cuál es el principal objetivo que desea lograr con el sistema de gestión de inventarios?**

* Mantener un control preciso del inventario.
* Reducir errores manuales en los registros.
* Generar reportes sobre el estado del inventario.
* Automatizar procesos de actualización tras ventas o ingresos de productos.
* Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**2. ¿Qué funcionalidades considera indispensables en el sistema?**

Seleccione las que apliquen:

* Registro de nuevos productos.
* Consulta en tiempo real del inventario.
* Generación de alertas cuando el inventario esté bajo.
* Reportes de ventas y niveles de inventario.
* Registro de ingresos y salidas de productos.
* Actualización automática del inventario tras una venta.
* Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**3. ¿Cuántos usuarios utilizarán el sistema?**

* Solo una persona (uso personal).
* De 2 a 5 personas (pequeño equipo).
* Más de 5 personas.

**4. ¿Qué nivel de conocimiento técnico tienen los usuarios que utilizarán el sistema?**

* Básico (conocimientos mínimos de computación).
* Intermedio (manejo básico de software).
* Avanzado (conocimiento técnico sobre sistemas de información).

**5. ¿Cómo espera que el sistema sea accesible?**

* Instalación en una computadora local.
* Acceso desde múltiples dispositivos en red local.
* Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**6. ¿Qué información necesita que el sistema registre?**

* Nombre y descripción de los productos.
* Categorías de productos.
* Cantidades disponibles.
* Precios de venta y costo.
* Fecha de última actualización.
* Historial de transacciones.
* Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**7. ¿Qué reportes le gustaría que el sistema genere automáticamente?**

* Niveles de inventario por producto.
* Productos más vendidos.
* Productos con menor rotación.
* Fechas y volúmenes de reabastecimiento.
* Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**8. ¿Desea que el sistema incluya alertas automáticas?**

* Sí, cuando los niveles de inventario sean bajos.
* Sí, para productos que no se hayan vendido en un tiempo específico.
* No, no es necesario.

**9. ¿Qué tan importante es la generación de reportes visuales (gráficos o tablas)?**

* Muy importante, para análisis rápido.
* Moderadamente importante, solo en casos necesarios.
* No es importante, prefiero reportes simples en texto.

**10. ¿Cuáles serían las principales mejoras que espera del sistema respecto a los métodos actuales?**

* Mayor precisión en los registros.
* Reducción de tiempos de administración.
* Mejor organización de los datos.
* Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Respuestas Simuladas:**

1. Mantener un control preciso del inventario.
2. Registro de nuevos productos, consulta en tiempo real, y reportes de ventas.
3. De 2 a 5 personas.
4. Básico.
5. Instalación en una computadora local.
6. Nombre de productos, cantidades disponibles, precio.
7. Niveles de inventario y productos más vendidos.
8. No, no es necesario.
9. Moderadamente importante.
10. Mejor organización de los datos y reducción de tiempos de administración.

# CONCLUSIONES Y RECOMENDACIONES

El desarrollo de este sistema de gestión de inventarios ha permitido evidenciar la importancia de implementar soluciones tecnológicas incluso en escenarios con recursos limitados. A continuación, se detallan las principales conclusiones del trabajo:

**Conclusiones**

* **Impacto en la Organización:**

El sistema desarrollado, aunque sencillo, cumple con los objetivos planteados al proporcionar una herramienta funcional para registrar, actualizar y consultar el estado del inventario. Esto demuestra que incluso pequeñas implementaciones tecnológicas pueden generar cambios significativos en la eficiencia operativa de una empresa.

* **Reducción de Errores Humanos:**

La automatización básica en la actualización de inventarios tras cada transacción ha reducido la posibilidad de errores comunes, como el registro manual incorrecto o la omisión de datos. Esto también ha mejorado la precisión de la información utilizada para la toma de decisiones.

* **Facilidad de Uso:**

La interfaz gráfica diseñada en Visual Studio ha resultado ser intuitiva y accesible, permitiendo a los usuarios realizar operaciones básicas sin necesidad de formación técnica avanzada. Esto refuerza la importancia de priorizar la usabilidad en sistemas destinados a pequeñas empresas.

* **Flexibilidad y Escalabilidad:**

Aunque el sistema fue diseñado con funcionalidades básicas, su estructura permite que se amplíe en el futuro para incluir características adicionales, como la generación de reportes avanzados, alertas automáticas o la integración con otros sistemas de ventas.

* **Limitaciones Técnicas:**

Se identificaron algunas limitaciones inherentes a las herramientas utilizadas, como la dependencia de una configuración manual inicial para la conexión con SQL Server y la falta de funcionalidades avanzadas como la predicción de demanda. Sin embargo, estas limitaciones no afectan el desempeño del sistema en su estado actual.

**Recomendaciones**

* **Ampliar Funcionalidades:**

Para maximizar el impacto del sistema, se sugiere incorporar nuevas funcionalidades, como la capacidad de generar gráficos de ventas, realizar análisis predictivo de demanda y gestionar múltiples ubicaciones de inventario. Esto puede lograrse mediante la integración de herramientas adicionales en SQL Server o bibliotecas especializadas en Visual Studio.

* **Capacitación para los Usuarios:**

Aunque el sistema es sencillo, una breve capacitación para los usuarios finales garantizará que puedan utilizar todas las funcionalidades disponibles y aprovechar al máximo la herramienta. Además, se recomienda proporcionar un manual de usuario detallado.

* **Automatización de Tareas Adicionales:**

Se puede mejorar la eficiencia del sistema implementando funciones como el envío de alertas por correo electrónico cuando los niveles de inventario sean bajos o la generación automática de órdenes de compra basadas en parámetros predefinidos.

* **Mantenimiento y Mejora Continua:**

Es esencial realizar un seguimiento constante del desempeño del sistema y recopilar retroalimentación de los usuarios para identificar áreas de mejora. Esto incluye actualizar la base de datos y el código fuente según las necesidades cambiantes del negocio.

* **Preparación para Escalabilidad:**

Si el sistema se implementa en un entorno empresarial más amplio, será necesario preparar la infraestructura para manejar un mayor volumen de datos y transacciones. Esto podría incluir la migración a servidores más robustos o la implementación de un sistema basado en la nube.

# REFERENCIAS

1. Microsoft. (n.d.). *SQL Server Documentation*. Recuperado de <https://learn.microsoft.com/sql>  
   Guía oficial sobre la instalación, configuración y uso de SQL Server para la gestión de bases de datos relacionales.
2. Microsoft. (n.d.). *Visual Studio Documentation*. Recuperado de <https://learn.microsoft.com/visualstudio>  
   Documentación oficial que detalla el uso de Visual Studio para desarrollar aplicaciones de escritorio con conexión a bases de datos.
3. Sommerville, I. (2011). *Ingeniería de software* (9.ª ed.). Pearson.  
   Texto clásico que aborda las metodologías de desarrollo de software, incluyendo enfoques ágiles.
4. Pressman, R. S. (2014). *Ingeniería del software: Un enfoque práctico* (7.ª ed.). McGraw-Hill.  
   Referencia útil para el diseño de sistemas y el modelado orientado a objetos.
5. Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The Unified Modeling Language User Guide* (2.ª ed.). Addison-Wesley.  
   Libro que explica el modelado orientado a objetos y la creación de diagramas de clases y casos de uso.
6. Vega, R. (2020). *Gestión de inventarios en pequeñas empresas: Una guía práctica*. Editorial Independiente.  
   Recurso que aborda las estrategias básicas para la administración de inventarios en negocios pequeños y medianos.
7. Fowler, M. (2004). *UML Distilled: A Brief Guide to the Standard Object Modeling Language* (3.ª ed.). Addison-Wesley.  
   Introducción al modelado UML para diseñar sistemas de software.