



UNIVERSIDAD DE LAS AMÉRICAS

FACULTAD DE INGENIERÍAS Y CIENCIAS APLICADAS

CARRERA DE INGENIERÍA DE SOFTWARE

SEMANA 8

Taller colaborativo

Docente:

Bernarda Sandoval

Autores:

Paul Chuquitarco, Adrian Llerena, Francisco Mejía

Año

2024 –2025

Introducción

El programa proporcionado está diseñado para gestionar una tienda, facilitando la administración de productos, inventarios y transacciones. Los sistemas de gestión de tiendas son esenciales en el entorno comercial moderno, ya que automatizan tareas rutinarias, reducen errores humanos y mejoran la eficiencia operativa. Este software permite a los administradores de la tienda llevar un registro preciso de los productos, controlar los niveles de inventario y procesar ventas de manera efectiva.

Los beneficios clave de implementar un programa de gestión de tiendas incluyen:

- **Eficiencia Operacional:** Automatización de procesos de venta, inventario y facturación.
- **Precisión en el Inventario:** Reducción de errores humanos en el seguimiento de inventarios.
- **Mejora en el Servicio al Cliente:** Rápido procesamiento de ventas y accesibilidad a la información del producto.
- **Reportes y Análisis:** Generación de reportes que ayudan en la toma de decisiones estratégicas.

Laudon, KC y Laudon, JP (2020). Sistemas de información gerencial: gestión de la empresa digital. Pearson.

Bowersox, DJ, Closs, DJ y Cooper, MB (2012). Gestión Logística de la Cadena de Suministro. Educación McGraw-Hill.

Somos una tienda:

En nuestra tienda, la gestión eficiente de los productos es esencial para mantener la organización y satisfacer las necesidades de nuestros clientes. Para ello, hemos desarrollado un programa simple pero eficaz que nos permite llevar un control básico de los productos disponibles. Este sistema nos permite ingresar productos, mostrarlos, editarlos y eliminarlos, facilitando así la administración diaria de nuestro inventario.

Beneficios del Sistema de Gestión de Productos

1. Visualización de Productos

Nuestro programa permite una visualización clara y organizada de los productos ingresados. Esto facilita la supervisión del inventario y ayuda a mantener un control constante sobre los productos disponibles.

2. Edición de Productos

La capacidad de editar productos existentes nos permite actualizar información importante como precios, descripciones o cantidades en inventario. Esto asegura que siempre tengamos datos precisos y actualizados.

3. Eliminación de Productos

Eliminar productos que ya no están disponibles o que han sido discontinuados es esencial para mantener nuestro inventario ordenado. Nuestro programa facilita este proceso, permitiéndonos mantener una base de datos limpia y relevante.

4. Interfaz Sencilla

El sistema está diseñado con una interfaz sencilla y fácil de usar, lo que reduce el tiempo de capacitación y facilita su uso diario por parte de todo el personal.

- *Silver, EA, Pyke, DF y Thomas, DJ (2016). Gestión de Inventario y Producción en Cadenas de Suministro. Prensa CRC.*
- *Simchi-Levi, D., Kaminsky, P. y Simchi-Levi, E. (2008). Diseño y gestión de la cadena de suministro: conceptos, estrategias y estudios de casos. Educación McGraw-Hill.*

Formulación del Problema

Nombre del sistema:	Tutienda
Objetivos:	Objetivo General Implementar un sistema de gestión de productos que permita a nuestra tienda

	<p>administrar de manera eficiente y efectiva el inventario, mejorar la precisión en el control de productos y optimizar la experiencia del cliente mediante funcionalidades de visualización, edición y eliminación de productos.</p> <p>Referencia:</p> <ul style="list-style-type: none"> • Laudon, K. C., & Laudon, J. P. (2020). Management Information Systems: Managing the Digital Firm. Pearson. <p>Objetivos Específicos</p> <ol style="list-style-type: none"> 1. Desarrollar una funcionalidad de visualización de productos que proporcione una lista clara y organizada de los productos en inventario, facilitando el monitoreo y la supervisión continua del stock. <p>Referencia:</p> <ul style="list-style-type: none"> ○ Heizer, J., Render, B., & Munson, C. (2016). Operations Management: Sustainability and Supply Chain Management. Pearson. <ol style="list-style-type: none"> 2. Implementar características de edición y eliminación de productos que permitan a los empleados actualizar la información del inventario y eliminar productos
--	--

	<p>obsoletos o no disponibles, garantizando la precisión y relevancia de los datos almacenados.</p> <p>Referencia:</p> <ul style="list-style-type: none"> ○ Slack, N., & Brandon-Jones, A. (2019). Operations Management. Pearson.
<p>Contexto del problema (Considerar los contextos económico, social y ambiental, dentro del sector productivo en el que funcionaría el sistema de inventarios)</p>	<p>Contexto del Problema</p> <p>Contexto Económico</p> <p>En el contexto económico actual, las tiendas enfrentan una competencia feroz y una presión constante para optimizar sus operaciones y reducir costos. La eficiencia en la gestión de inventarios es crucial para maximizar la rentabilidad y la competitividad en el mercado. Las tiendas que no gestionan adecuadamente su inventario corren el riesgo de incurrir en costos innecesarios, como el sobrestock, que genera costos de almacenamiento, o la falta de stock, que resulta en pérdida de ventas y clientes insatisfechos. Un sistema de gestión de inventarios eficaz permite a las tiendas mantener niveles óptimos de stock, reduciendo costos y mejorando el flujo de caja.</p> <p>Referencia:</p>

- Chopra, S., & Meindl, P. (2016). Supply Chain Management: Strategy, Planning, and Operation. Pearson.

Contexto Social

Desde una perspectiva social, la eficiencia en la gestión de inventarios tiene un impacto directo en la experiencia del cliente y, por ende, en la reputación y lealtad hacia la tienda. Los clientes esperan encontrar los productos que buscan en stock y disponibles para la compra. Un sistema de gestión de inventarios que permite una rápida actualización y visualización de productos garantiza que los empleados puedan brindar información precisa y oportuna a los clientes, mejorando la satisfacción y fomentando la lealtad. Además, un inventario bien gestionado evita situaciones de escasez de productos esenciales, lo que puede ser crítico en momentos de alta demanda.

Referencia:

- Zeithaml, V. A., Bitner, M. J., & Gremler, D. D. (2017). Services Marketing: Integrating Customer Focus Across the Firm. McGraw-Hill Education.

Contexto Ambiental

En el contexto ambiental, una gestión eficiente del inventario contribuye a la sostenibilidad al minimizar el desperdicio de productos. El sobrestock no solo implica costos adicionales sino también un mayor desperdicio de recursos si los productos no se venden antes de su fecha de caducidad o se vuelven obsoletos. Un sistema de inventarios eficaz ayuda a las tiendas a prever la demanda con mayor precisión y a mantener solo la cantidad necesaria de productos, reduciendo el desperdicio y optimizando el uso de recursos. Esto es especialmente importante en sectores como el de alimentos y productos perecederos, donde el manejo adecuado del inventario puede tener un impacto significativo en la reducción del desperdicio alimentario.

Referencia:

- Bowersox, D. J., Closs, D. J., & Cooper, M. B. (2012). Supply Chain Logistics Management. McGraw-Hill Education.

Sector Productivo: Comercio Minorista

En el sector productivo del comercio minorista, la implementación de un sistema de gestión de inventarios es vital para enfrentar los desafíos asociados con

	<p>la alta rotación de productos y la necesidad de adaptarse rápidamente a las demandas del mercado. Las tiendas minoristas deben gestionar una amplia variedad de productos, cada uno con diferentes ciclos de vida y patrones de demanda. Un sistema de gestión de inventarios eficiente permite a las tiendas minoristas optimizar sus niveles de stock, mejorar la disponibilidad de productos y reducir los costos asociados con el manejo de inventarios.</p> <p>Referencia:</p> <ul style="list-style-type: none"> • Heizer, J., Render, B., & Munson, C. (2016). Operations Management: Sustainability and Supply Chain Management. Pearson.
Restricciones (Consideraciones externas que se deben tener para diseñar y desarrollar el proyecto, ejemplo: porcentaje de impuesto, registro sanitario u otros)	<p>Al diseñar y desarrollar nuestro sistema de gestión de inventarios, es crucial considerar varias restricciones externas que pueden influir en la implementación y operación del sistema. Estas restricciones incluyen aspectos económicos, legales, regulatorios, y técnicos que deben ser tomados en cuenta para asegurar que el sistema cumpla con todas las normativas y necesidades del entorno en el que operamos.</p>

1. Normativas Fiscales y Tributarias

Porcentaje de Impuesto sobre Ventas (IVA): El sistema debe ser capaz de manejar diferentes tasas de impuestos aplicables a los productos vendidos. Esto incluye calcular y aplicar el Impuesto al Valor Agregado (IVA) u otros impuestos específicos de la región en todas las transacciones.

Referencia:

- Gobierno de España (2021). Ley del Impuesto sobre el Valor Añadido.

Declaración de Impuestos:

El sistema debe facilitar la generación de reportes de ventas e impuestos que puedan ser usados para la declaración fiscal. Esto incluye un registro preciso de todas las ventas realizadas, mostrando los impuestos recaudados.

Referencia:

- Internal Revenue Service (IRS) (2021). Publication 334: Tax Guide for Small Business.

2. Regulaciones Sanitarias

- Registro Sanitario: Para productos que requieren regulación sanitaria

	<p>(como alimentos, medicinas, y productos cosméticos), el sistema debe permitir el seguimiento de los números de lote y las fechas de vencimiento, asegurando que todos los productos cumplan con los requisitos de registro sanitario.</p> <p>Referencia:</p> <ul style="list-style-type: none"> ○ Administración de Alimentos y Medicamentos de EE.UU. (FDA) (2021). Guidance for Industry: Product Tracing Requirements. <p>3. Consideraciones Ambientales</p> <p>Gestión de Residuos: El sistema debe incluir funcionalidades para minimizar el desperdicio, como alertas para productos próximos a vencer y estrategias de rotación de stock (FIFO - First In, First Out). Además, debe ser compatible con prácticas de sostenibilidad, como la reducción de residuos y el reciclaje.</p> <p>Referencia:</p> <ul style="list-style-type: none"> ○ Unión Europea (2020). Directiva 2008/98/EC sobre residuos.
--	---

4. Normativas de Protección al Consumidor

Etiquetado y Transparencia: El sistema debe asegurar que toda la información de los productos (precios, descripción, origen, etc.) esté disponible y sea fácilmente accesible para los consumidores, cumpliendo con las leyes de protección al consumidor que requieren transparencia y veracidad en la información.

Referencia:

- Comisión Federal de Comercio de EE.UU. (FTC) (2021). Guías para la Publicidad y el Etiquetado de Productos.

5. Seguridad y Protección de Datos

Cumplimiento del GDPR: Si nuestra tienda opera en o tiene clientes de la Unión Europea, el sistema debe cumplir con el Reglamento General de Protección de Datos (GDPR), asegurando que la información personal de los clientes esté protegida y manejada de acuerdo con las regulaciones de privacidad.

Referencia:

- Parlamento Europeo y Consejo de la Unión Europea (2016).

	<p>Reglamento (UE) 2016/679 (Reglamento General de Protección de Datos - GDPR).</p> <p>Seguridad de la Información: El sistema debe implementar medidas robustas de seguridad para proteger los datos contra accesos no autorizados, brechas de seguridad y ciberataques.</p> <p>Referencia:</p> <ul style="list-style-type: none"> ○ Instituto Nacional de Estándares y Tecnología de EE.UU. (NIST) (2020). Framework for Improving Critical Infrastructure Cybersecurity. <p>6. Restricciones Técnicas</p> <p>Compatibilidad e Integración: El sistema debe ser compatible con las infraestructuras tecnológicas existentes en la tienda, incluyendo hardware (computadoras, escáneres de código de barras, etc.) y software (sistemas de punto de venta, ERP, etc.). También debe ser escalable para adaptarse al crecimiento futuro de la tienda.</p> <p>Referencia:</p> <ul style="list-style-type: none"> ○ Sommerville, I. (2016). Software Engineering. Pearson.
--	---

	<p>Mantenimiento y Soporte: El sistema debe incluir planes de mantenimiento y soporte continuo para asegurar su funcionamiento óptimo y la rápida resolución de cualquier problema técnico que pueda surgir.</p> <p>Referencia:</p> <ul style="list-style-type: none"> ○ Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach. McGraw-Hill Education.
<p>Limitaciones (Consideraciones internas que se deben tener para diseñar y desarrollar el proyecto, ejemplo: lenguaje de programación, tipos de almacenamiento u otros)</p>	<p>1. Lenguaje de Programación</p> <p>C :</p> <p>El sistema actual está desarrollado en C, un lenguaje de programación eficiente y de bajo nivel. La elección de C permite un control preciso sobre la memoria y el rendimiento del sistema, lo que es crucial para aplicaciones que requieren alta eficiencia. Sin embargo, C puede ser más complejo y propenso a errores en comparación con lenguajes de alto nivel.</p> <p>Para futuras expansiones o integraciones, considerar el uso de C++ podría ofrecer ventajas adicionales como programación orientada a objetos, que facilita el mantenimiento y escalabilidad del código.</p> <p>Referencia:</p>

	<ul style="list-style-type: none"> • Kernighan, B. W., & Ritchie, D. M. (1988). The C Programming Language. Prentice Hall. • Stroustrup, B. (2013). The C++ Programming Language. Addison-Wesley. <p>2. Tipo de Almacenamiento</p> <p>Memoria Volátil (RAM):</p> <p>Actualmente, los productos se almacenan en matrices en memoria volátil. Esto es adecuado para una aplicación pequeña, pero limita la cantidad de datos que se pueden manejar simultáneamente y se pierden todos los datos si el sistema se reinicia.</p> <p>Base de Datos Relacional:</p> <p>Para una solución más robusta y escalable, es recomendable utilizar una base de datos relacional (como MySQL o PostgreSQL) que permita el almacenamiento persistente de datos, consultas eficientes y la integridad referencial.</p> <p>Referencia:</p> <ul style="list-style-type: none"> • Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). Database System Concepts. McGraw-Hill.
--	---

3. Interfaz de Usuario

Consola de Comandos:

La interfaz actual basada en consola es simple y suficiente para operaciones básicas. Sin embargo, para mejorar la experiencia del usuario, se podría desarrollar una interfaz gráfica (GUI) utilizando herramientas como Qt o GTK.

Referencia:

- Summerfield, M. (2007). Rapid GUI Programming with Python and Qt. Prentice Hall.

4. Capacidad de Escalabilidad

Escalabilidad Limitada:

El sistema actual está diseñado para manejar un número limitado de productos (hasta 1000). Para asegurar que el sistema pueda crecer con la tienda, es necesario considerar soluciones que permitan la escalabilidad horizontal (distribuir la carga entre múltiples servidores) y vertical (aumentar la capacidad del servidor).

Referencia:

- Bondi, A. B. (2000). Characteristics of Scalability and Their Impact on Performance. Proceedings of the 2nd

International Workshop on
Software and Performance.

5. Integración con Otros Sistemas

Integración Limitada:

El sistema actual no está diseñado para integrarse fácilmente con otros sistemas como el punto de venta (POS) o sistemas de gestión de relaciones con clientes (CRM). Es crucial diseñar una arquitectura que permita futuras integraciones mediante APIs o middleware.

Referencia:

- Hohpe, G., & Woolf, B. (2004). Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions. Addison-Wesley.

6. Recursos Humanos y Capacitación

Capacitación del Personal:

El personal debe ser capacitado para utilizar y mantener el sistema. Esto implica desarrollar manuales de usuario y ofrecer sesiones de formación. Además, se debe asegurar que el personal tenga conocimientos básicos de informática y uso de sistemas de gestión.

Referencia:

	<ul style="list-style-type: none"> Sommerville, I. (2016). Software Engineering. Pearson. <p>7. Mantenimiento y Actualización del Sistema</p> <p>Plan de Mantenimiento:</p> <p>Es esencial establecer un plan de mantenimiento regular para asegurar que el sistema funcione correctamente, incluyendo la corrección de errores, la optimización del rendimiento y la actualización de funcionalidades.</p> <p>Referencia:</p> <ul style="list-style-type: none"> Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach. McGraw-Hill Education.
--	--

- Identificación de variables entrada, salida y procesos asociados.

Variables de entrada

Variables de entrada	Descripción
Nombre del producto	Nombre del producto a ingresar
Cantidad de productos	Cantidad de productos a ingresar
Precio	Precio del producto
Opción del Menú	Selección de la acción a realizar en el menú

Variables de Salida

Variables de Salida	Descripción
Listado de productos	Información concatenada a los productos
Errores/Alertas	Mensajes de error o alerta del sistema

Procesos

Procesos	Descripción
Ingreso de productos	Captura y almacenamiento de datos de nuevos productos
Mostrar productos	Visualización de productos en inventario
Editar producto	Modificación de productos existentes
Eliminar producto	Eliminación de productos del inventario

Planteamiento del problema

Método 1: Sistema de Gestión de Inventarios con Base de Datos Relacional

Descripción: Este método implica la creación de un sistema de gestión de inventarios que utiliza una base de datos relacional para almacenar y gestionar la información de los productos. Las bases de datos relacionales, como MySQL, PostgreSQL o SQLite, son ideales para manejar grandes volúmenes de datos y proporcionar capacidades robustas de consulta y manipulación de datos.

Ventajas:

1. **Escalabilidad:** Las bases de datos relacionales pueden manejar grandes volúmenes de datos, lo que permite la expansión futura de la tienda.
2. **Integridad de Datos:** Las bases de datos relacionales utilizan restricciones y relaciones entre tablas para mantener la integridad de los datos, reduciendo errores y duplicaciones.
3. **Consultas Eficientes:** El uso de SQL permite realizar consultas complejas y obtener información específica rápidamente.
4. **Seguridad:** Las bases de datos relacionales ofrecen características de seguridad avanzadas, como control de acceso y cifrado.
5. **Backup y Recuperación:** Las bases de datos relacionales permiten realizar copias de seguridad automáticas y restaurar datos en caso de fallos.

Desventajas:

1. **Complejidad Inicial:** Configurar y diseñar una base de datos relacional puede ser complejo y requerir conocimientos técnicos específicos.
2. **Costo:** Dependiendo de la solución de base de datos elegida, puede haber costos asociados con licencias, mantenimiento y soporte.
3. **Requerimientos de Hardware:** Las bases de datos relacionales pueden necesitar servidores y hardware más robustos para funcionar eficientemente.

Método 2: Sistema de Gestión de Inventarios Basado en Archivos

Descripción: Este método implica la creación de un sistema de gestión de inventarios que utiliza archivos de texto o archivos CSV para almacenar y gestionar la información de los productos. Los archivos de texto pueden ser manipulados fácilmente mediante programación en lenguajes como C o Python.

Ventajas:

1. **Simplicidad:** El manejo de archivos de texto es simple y no requiere la instalación y configuración de un servidor de base de datos.
2. **Costo Bajo:** No hay costos asociados con licencias de software de bases de datos, lo que hace que sea una solución más económica.
3. **Portabilidad:** Los archivos de texto son fácilmente portables y pueden ser copiados o transferidos sin complicaciones.

Desventajas:

1. **Escalabilidad Limitada:** Manejar grandes volúmenes de datos con archivos de texto puede ser ineficiente y lento.
2. **Integridad de Datos:** No hay mecanismos integrados para asegurar la integridad de los datos, lo que puede resultar en errores y duplicaciones.
3. **Consultas Ineficientes:** Realizar búsquedas y consultas en archivos de texto es menos eficiente y más lento que en bases de datos relacionales.
4. **Seguridad:** Los archivos de texto carecen de características avanzadas de seguridad, lo que los hace más vulnerables a accesos no autorizados.
5. **Backup y Recuperación:** La gestión de copias de seguridad y recuperación de datos es manual y menos eficiente en comparación con bases de datos relacionales.

Elección del Mejor Método: Sistema de Gestión de Inventarios con Base de Datos Relacional

Razones para Elegir el Método 1:

1. **Escalabilidad y Futuro Crecimiento:** A medida que nuestra tienda crezca, necesitaremos manejar un mayor volumen de datos. Una base de datos relacional puede escalar eficientemente para satisfacer estas necesidades.
2. **Integridad y Seguridad de Datos:** Mantener la integridad de los datos es crucial para evitar errores que puedan afectar la operación de la tienda. Las bases de datos relacionales ofrecen mecanismos robustos para asegurar la integridad y la seguridad de los datos.
3. **Consultas y Análisis Eficientes:** La capacidad de realizar consultas complejas y obtener información detallada de manera rápida y eficiente es esencial para la toma de decisiones estratégicas. }
4. **Backup y Recuperación Automatizados:** La posibilidad de realizar copias de seguridad automáticas y restaurar datos rápidamente en caso de fallos asegura la continuidad del negocio.

Diagrama de flujo

Imágenes de las partes más importantes del código explicadas.

```

#include <stdio.h>
#include <string.h>
#include "funciones.h"
#include "lecturas.h"

#define MaxProducto 1000
#define MaxNombre 60
#define MaxDatos 1000

int main(void) {
    int cantidadProduct;
    char Productos[MaxProducto][MaxDatos];
    int opcion;
    int indice;

    printf("          Bienvenidos a TuTienda\n");
    printf("Cantidad de productos a ingresar (máximo %d): ", MaxProducto);
    scanf("%d", &cantidadProduct);
    getchar();

    leerProducto(cantidadProduct, Productos);

    do {
        printf("\nMenu:\n");
        printf("1. Mostrar productos\n");
        printf("2. Editar producto\n");
        printf("3. Eliminar producto\n");
        printf("4. Salir\n\n");
    } while (opcion != 4);
}

```

El main es la parte más importante de un programa por varias razones fundamentales. En primer lugar, sirve como el punto de entrada del programa, siendo la primera función que se ejecuta cuando el programa se inicia. Desde aquí, se coordina toda la ejecución del programa, interactuando con el usuario al presentar mensajes, solicitar entrada y gestionar acciones. Además, el main es responsable de dirigir la ejecución de otras funciones y módulos del programa, organizando el flujo de operaciones y decidiendo qué funciones se llaman y en qué orden. También alberga la lógica principal del programa, incluyendo cálculos, manipulación de datos, toma de decisiones y cualquier operación central. Además, controla el flujo del programa, decidiendo qué operaciones realizar basadas en condiciones específicas y cuándo finalizar la ejecución del programa. Por último, el main a menudo se encarga de gestionar errores y excepciones, detectando problemas, informando al usuario y tomando medidas correctivas. En resumen, el `main` es la columna vertebral del programa, coordinando su funcionamiento y asegurando que cumpla su propósito de manera efectiva.

Imágenes de la ejecución de cada sección del programa explicadas.

```
Bienvenidos a TuTienda
Cantidad de productos a ingresar (máximo 1000): 3
Escribe el nombre del producto 1: papa
Ingresa el precio del producto 1: 1.2
Ingresa la cantidad del producto 1: 54
Escribe el nombre del producto 2: aceite
Ingresa el precio del producto 2: 3.4
Ingresa la cantidad del producto 2: 39
Escribe el nombre del producto 3: atun
Ingresa el precio del producto 3: 2.3
Ingresa la cantidad del producto 3: 34
```

En esta primera sección se están leyendo los productos, específicamente tres en este caso.

```
Menu:
1. Mostrar productos
2. Editar producto
3. Eliminar producto
4. Salir

Seleccione una opción: 1
Producto 1:
Nombre: papa
Precio: 1.20
Cantidad: 54

Producto 2:
Nombre: aceite
Precio: 3.40
Cantidad: 39

Producto 3:
Nombre: atun
Precio: 2.30
Cantidad: 34
```

En esta segunda sección se exhiben los productos introducidos antes.

```
Menu:
1. Mostrar productos
2. Editar producto
3. Eliminar producto
4. Salir

Seleccione una opción: 2
Ingrese el producto a editar (1-3): 1
Escribe el nuevo nombre del producto 1: azucar
Ingrese el nuevo precio del producto 1: 3.4
Ingrese la nueva cantidad del producto 1: 87

Menu:
1. Mostrar productos
2. Editar producto
3. Eliminar producto
4. Salir

Seleccione una opción: 1
Producto 1:
Nombre: azucar
Precio: 3.40
Cantidad: 87

Producto 2:
Nombre: aceite
Precio: 3.40
Cantidad: 39

Producto 3:
Nombre: atun
Precio: 2.30
Cantidad: 34
```

En esta tercera sección, observamos el proceso de edición del primer producto, el cual cambia de "papa" a "azúcar".

```
Menu:
1. Mostrar productos
2. Editar producto
3. Eliminar producto
4. Salir

Seleccione una opción: 3
Ingrese el producto a eliminar (1-3): 3

Menu:
1. Mostrar productos
2. Editar producto
3. Eliminar producto
4. Salir

Seleccione una opción: 1
Producto 1:
Nombre: azucar
Precio: 3.40
Cantidad: 87

Producto 2:
Nombre: aceite
Precio: 3.40
Cantidad: 39

Menu:
1. Mostrar productos
2. Editar producto
3. Eliminar producto
4. Salir
```

En esta cuarta sección, se muestra cómo eliminar el producto número tres del inventario.

Conclusiones

La implementación de un sistema de gestión de inventarios eficiente es esencial para mejorar la operación y competitividad de nuestra tienda. Tras evaluar diferentes métodos, hemos concluido que un sistema basado en una base de datos relacional es la opción más adecuada debido a sus ventajas en términos de escalabilidad, integridad de datos, eficiencia en consultas y seguridad.

1. **Escalabilidad:** Las bases de datos relacionales permiten manejar un volumen creciente de datos a medida que nuestra tienda expande sus operaciones y su inventario.

2. **Integridad de Datos:** Los mecanismos de integridad referencial y las restricciones en las bases de datos relacionales minimizan errores y duplicaciones, asegurando que los datos sean precisos y confiables.
3. **Eficiencia en Consultas:** Las capacidades de consulta avanzadas de SQL permiten realizar análisis detallados y obtener información crítica rápidamente, lo que es esencial para la toma de decisiones estratégicas.
4. **Seguridad:** Las características de seguridad avanzadas, como control de acceso y cifrado, protegen los datos sensibles contra accesos no autorizados y brechas de seguridad.
5. **Backup y Recuperación:** La posibilidad de realizar copias de seguridad automáticas y restaurar datos rápidamente en caso de fallos asegura la continuidad del negocio.

Referencias:

- Laudon, K. C., & Laudon, J. P. (2020). *Management Information Systems: Managing the Digital Firm*. Pearson.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). *Database System Concepts*. McGraw-Hill.
- Heizer, J., Render, B., & Munson, C. (2016). *Operations Management: Sustainability and Supply Chain Management*. Pearson.

Recomendaciones

Para maximizar los beneficios del sistema de gestión de inventarios basado en una base de datos relacional, se deben considerar las siguientes recomendaciones:

1. Capacitación del Personal:

Proporcionar capacitación adecuada a los empleados sobre el uso del sistema y la gestión de bases de datos para asegurar que todos los usuarios puedan operar el sistema de manera eficiente.

Referencia: Sommerville, I. (2016). *Software Engineering*. Pearson.

2. Mantenimiento Regular:

Establecer un plan de mantenimiento regular para garantizar que el sistema funcione correctamente, incluyendo la actualización del software, la optimización del rendimiento y la corrección de errores.

Referencia: Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education.

3. Seguridad de Datos:

Implementar medidas de seguridad robustas, como cifrado de datos, autenticación de usuarios y control de acceso, para proteger la información sensible contra accesos no autorizados y ciberataques.

Referencia: Instituto Nacional de Estándares y Tecnología de EE.UU. (NIST) (2020). *Framework for Improving Critical Infrastructure Cybersecurity*.

4. **Monitorización y Análisis:**

Utilizar herramientas de monitorización y análisis para revisar continuamente el rendimiento del sistema y realizar ajustes necesarios para mejorar la eficiencia y la precisión del inventario.

Referencia: Davenport, T. H., & Harris, J. G. (2007). *Competing on Analytics: The New Science of Winning*. Harvard Business Review Press.

5. **Evaluación y Mejora Continua:**

Realizar evaluaciones periódicas del sistema para identificar áreas de mejora y adaptarse a los cambios en las necesidades del negocio y del mercado.

Referencia: Turban, E., Sharda, R., & Delen, D. (2020). *Business Intelligence, Analytics, and Data Science: A Managerial Perspective*. Pearson.