

Taller POO en C# - Inventario simple

Integrantes

Dairo Andres Mosquera Blandon

Harrison Alonso Arroyave Gaviria

Caso de negocio

Problema

En pequeños negocios no se tiene control organizado del inventario. En algunos se utilizan hojas de Excel, archivos de texto u hojas de papel. Esto genera pérdidas por productos agotados, sobreabastecimiento, falta de registro de productos perecederos, entre otros problemas conocidos.

Usuarios

Operador: Encargado de registrar y actualizar inventario.

Supervisor: Revisa reportes y estado de inventario.

Valor

La aplicación permitirá tener un inventario actualizado, fácil de consultar y administrar, reduciendo la probabilidad de pérdidas, mejorando la gestión y garantizando un mejor cumplimiento de los principios de seguridad de la información que son la disponibilidad, integridad y confidencialidad.

Historias de Usuario

HU1 – Registrar producto

Como *Operador* quiero registrar un nuevo producto en el inventario para tenerlo disponible para su gestión.

Criterios de aceptación:

- *Given* que no existe un producto con el mismo código, *When* ingreso sus datos, *Then* queda guardado en el inventario.
- *Given* que existe un producto con el mismo código, *When* ingreso los datos del producto; *Then* se muestra un mensaje indicando que ya existe un producto registrado con ese código.
- *Given* que no se ingresó ningún valor, *When* el sistema solicita ingresar el valor para un dato requerido, *Then* se muestra un mensaje indicando que el dato es requerido.

HU2 – Consultar productos

Como *Operador* quiero listar todos los productos registrados para tener una vista general del inventario.

Criterios de aceptación:

- *Given* que existen productos, *When* selecciono la opción de listar, *Then* se muestra el código, nombre, cantidad y precio de cada producto.
- *Given* que no existen productos, *When* selecciono la opción de listar, *Then* se muestra un mensaje indicando que no se encontraron productos.

HU3 – Actualizar stock

Como *Operador* quiero modificar la cantidad de un producto para mantener actualizado el inventario.

Criterios de aceptación:

- *Given* que el producto existe, *When* ingreso una nueva cantidad, *Then* se actualiza correctamente sin permitir valores negativos.
- *Given* que el producto no existe, *When* ingreso el código del producto, *Then* se muestra un mensaje indicando que el producto no existe.

HU4 – Eliminar producto

Como *Operador* quiero eliminar un producto del inventario para mantener la base de datos depurada.

Criterios de aceptación:

- *Given* que el producto existe, *When* lo selecciono para eliminar, *Then* se borra del inventario.
- *Given* que el producto no existe, *When* ingreso el código del producto, *Then* se muestra un mensaje indicando que el producto no existe.

HU5 – Reporte de stock bajo

Como *Supervisor* quiero ver un reporte de productos con stock bajo para poder reabastecerlos a tiempo.

Criterios de aceptación:

- *Given* que existen productos con stock menor a un umbral, *When* consulto el reporte, *Then* se muestra el nombre, tipo y cantidad de cada producto.
- *Given* que no existen productos con stock menor a un umbral, *When* consulto el reporte, *Then* se muestra un mensaje indicando que no se encontraron productos con stock menor al umbral.

HU6 – Buscar producto por nombre

Como *Operador* quiero buscar productos por nombre para encontrarlos rápidamente.

Criterios de aceptación:

- *Given* que existen productos en el inventario, *When* ingreso un texto de búsqueda, *Then* se muestra únicamente los productos que coinciden.
- *Given* que no hay coincidencias, *When* realizo la búsqueda, *Then* se muestra un mensaje indicando que no se encontraron resultados.

HU7 – Exportar productos a CSV

Como *Supervisor* quiero exportar los productos a CSV.

Criterios de aceptación:

- *Given* que existen productos en el inventario, *When* selecciono la opción de exportar a CSV, *Then* se guardan los datos de los productos en un archivo CSV en la ruta especificada.
- *Given* que no existen productos en el inventario, *When* selecciono la opción de exportar a CSV, *Then* se muestra un mensaje indicando que no se encontraron productos para exportar.
- *Given* que no se ingresó ningún valor, *When* el sistema solicita la ruta destino, *Then* se muestra un mensaje indicando que la ruta destino es requerida.

Requerimientos

Reglas de negocio (RN)

- RN-01: Cada producto debe tener un código único.
- RN-02: La cantidad mínima de un producto es 0 (no se permiten negativos).
- RN-03: Un producto puede ser Perecedero (con fecha de vencimiento) o No Perecedero (con categoría).
- RN-04: Se considera stock bajo cuando la cantidad < 5.

Requerimientos funcionales (RF)

- RF-01: CRUD de productos.
- RF-02: Diferenciar productos perecederos y no perecederos.
- RF-03: Validar stock (no negativos).
- RF-04: Validar datos requeridos.
- RF-05: Reporte de productos con stock bajo.
- RF-06: Menú de consola con opciones de gestión.
- RF-07: Búsqueda de producto por nombre.
- RF-08: Exportación a CSV.

Requerimientos no funcionales (RNF)

- RNF-01: La aplicación responde en < 1 s con hasta 100 registros.
- RNF-02: Mensajes de error claros al usuario.
- RNF-03: Código limpio y organizado cumpliendo principios SOLID.
- RNF-04: Portabilidad: debe correr en Windows y Linux (.NET 8).

- RNF-05: Testabilidad: las clases deben estar separadas para poder ser testeadas de forma unitaria.
- RNF-06: Compatible con múltiples métodos de persistencia, como JSON, en memoria, entre otros.
- RNF-07: Exportación de datos eficiente con un límite de hasta 1000 registros.

Diseño POO

La **encapsulación** se evidencia en el control de acceso a los datos mediante campos privados como *_quantity* para almacenar stock de cada producto y *_repository* para almacenar la instancia del repositorio que se usará en cada servicio. Las propiedades son públicas con setters privados, garantizando que el estado interno de los objetos solo pueda modificarse a través de métodos controlados como *IncreaseStock* y *DecreaseStock* para actualizar el stock de los productos realizando las validaciones correspondientes. La **herencia** se implementa a través de una jerarquía donde *BaseEntity* proporciona funcionalidad común de las entidades, *Product* actúa como clase abstracta base de productos, y *PerishableProduct* y *NonPerishableProduct* extienden esta funcionalidad con características específicas.

El **polimorfismo** se manifiesta mediante métodos virtuales como *GetDisplayInfo*, *GetNameAndStock* y *GetCSVLine* que son sobrescritos en las clases derivadas para proporcionar comportamientos específicos según el tipo de producto, permitiendo que el mismo código trate diferentes tipos de productos de manera uniforme. Finalmente, la **abstracción** se logra a través de interfaces como *IRepository<T>*, *IInventoryService*, *IReportService* y *IExportService* que ocultan la complejidad de implementación y permiten intercambiar implementaciones, como el caso de los repositorios de JSON y en memoria (*JsonRepository* e *InMemoryRepository*) sin afectar el código cliente.

Plan de pruebas

HU1 - Registrar Producto

Caso de prueba	Datos de entrada	Resultado esperado
TC1 - Registro exitoso producto perecedero	ID: "1", Nombre: "Leche", Descripción: "Leche Colanta", Precio: 3500, Cantidad: 100, Fecha: 2025-09-15	Producto registrado exitosamente
TC2 - Registro exitoso producto no perecedero	ID: "2", Nombre: "Arroz", Descripción: "Arroz Diana", Precio: 4000, Cantidad: 50, Categoría: "Granos"	Producto registrado exitosamente
TC3 - Producto duplicado	ID: "1" (ya existe), Nombre: "Pan", Descripción: "Pan Bimbo", Precio: 2000, Cantidad: 20	Excepción: "Ya existe un recurso con Id 1"
TC004 - Nombre requerido	ID: "3", Nombre: "", Descripción: "Descripción", Precio: 1000, Cantidad: 10	Excepción: "Nombre requerido"
TC005 - Descripción	ID: "4", Nombre: "Producto",	Excepción: "Descripción"

requerida	Descripción: "", Precio: 1000, Cantidad: 10	requerida"
TC006 - Precio negativo	ID: "5", Nombre: "Producto", Descripción: "Desc", Precio: -100, Cantidad: 10	Excepción: "El precio no puede ser negativo"
TC007 - Cantidad negativa	ID: "6", Nombre: "Producto", Descripción: "Desc", Precio: 1000, Cantidad: -5	Excepción: "La cantidad no puede ser negativa"
TC008 - Fecha expiración pasada	ID: "7", Nombre: "Yogurt", Descripción: "Yogurt", Precio: 2000, Cantidad: 10, Fecha: 2020-01-01	Excepción: "La fecha de expiración debe ser futura"

HU2 - Consultar Productos

Caso de prueba	Datos de entrada	Resultado esperado
TC009 - Listar productos existentes	Inventario con 2 productos	Muestra información de ambos productos
TC010 - Inventario vacío	Inventario sin productos	Mensaje: "No se encontraron productos."
TC011 - Formato producto perecedero	Producto perecedero en inventario	Muestra: "ID, Nombre, Precio, Cantidad, Expira en: fecha"
TC012 - Formato producto no perecedero	Producto no perecedero en inventario	Muestra: "ID, Nombre, Precio, Cantidad, Categoría"

HU3 - Actualizar Stock

Caso de prueba	Datos de entrada	Resultado esperado
TC013 - Aumentar stock existente	ID: "1" (existe), Nueva cantidad: 150	Stock actualizado de 100 a 150
TC014 - Disminuir stock existente	ID: "1" (existe), Nueva cantidad: 80	Stock actualizado de 100 a 80
TC015 - Stock igual (sin cambio)	ID: "1" (existe), Nueva cantidad: 100	Stock permanece en 100
TC016 - Producto inexistente	ID: "999" (no existe), Nueva cantidad: 50	Excepción: "Producto con ID 999 no encontrado"
TC017 - Cantidad negativa	ID: "1" (existe), Nueva cantidad: -10	Stock se establece en 0 (validación interna)

HU4 - Eliminar Producto

Caso de prueba	Datos de entrada	Resultado esperado
TC018 - Eliminar producto existente	ID: "1" (existe en inventario)	Producto eliminado del inventario
TC019 - Eliminar producto inexistente	ID: "999" (no existe)	Excepción: "Recurso con ID 999 no encontrado"
TC020 - ID vacío	ID: "" (cadena vacía)	Excepción: "ID requerido"

TC021 - ID nulo	ID: null	Excepción: "ID requerido"
-----------------	----------	---------------------------

HU5 - Reporte de Stock Bajo

Caso de prueba	Datos de entrada	Resultado esperado
TC022 - Productos con stock bajo	Umbral: 5, Productos: Mantequilla(3), Detergente(10)	Reporte muestra solo "Mantequilla (Perecedero) (Cantidad: 3)"
TC023 - Sin productos con stock bajo	Umbral: 2, Todos los productos > 2	Mensaje: "No hay productos con stock menor a 2"
TC024 - Umbral personalizado	Umbral: 15, Varios productos	Muestra productos con cantidad < 15
TC025 - Inventario vacío	Umbral: 5, Sin productos	Mensaje: "No hay productos con stock menor a 5"

HU6 - Buscar Producto por Nombre

Caso de prueba	Datos de entrada	Resultado esperado
TC026 - Búsqueda con coincidencias	Texto: "Pan", Productos: "Pan", "Panela"	Muestra ambos productos que contienen "Pan"
TC027 - Búsqueda sin coincidencias	Texto: "Chocolate", Sin productos con "Chocolate"	Mensaje: "No se encontraron productos que coincidan con la búsqueda."
TC028 - Búsqueda case-insensitive	Texto: "pan" (minúscula), Producto: "Pan"	Encuentra el producto "Pan"
TC029 - Búsqueda parcial	Texto: "Le", Producto: "Leche"	Encuentra el producto "Leche"
TC030 - Búsqueda cadena vacía	Texto: "", Varios productos	Muestra todos los productos

HU7 – Exportar productos a CSV

Caso de prueba	Datos de entrada	Resultado esperado
TC031 - Exportar CSV con productos	Productos en inventario, Ruta: "C:\temp\productos.csv"	Archivo CSV creado con encabezados y datos
TC032 - Exportar CSV sin productos	Inventario vacío	Mensaje: "No se encontraron productos para exportar."
TC033 - Ruta CSV inválida	Ruta: "" (vacía)	Excepción: "Ruta de archivo requerida"