

Proyecto en C++: Sistema de Gestión de Inventario, Solicitudes de Compra, Clientes en Espera y Registro de Cambios

Enunciado Completo del Proyecto:

Una empresa de tecnología requiere un sistema para gestionar su inventario de productos, atender a sus clientes de manera eficiente y llevar un registro de los cambios realizados en el inventario. El sistema debe permitir:

1. Gestión de Inventario:

- **Registrar nuevos productos:** Almacenar información como nombre del producto, precio y cantidad.
 - **Eliminar productos:** Eliminar productos que ya no están disponibles.
 - **Consultar producto específico:** Buscar y mostrar información de un producto específico.
 - **Listar productos:** Mostrar todos los productos disponibles en orden alfabético.
-

2. Gestión de Solicitudes de Compra:

- **Registrar solicitudes:** Almacenar la información de las solicitudes de compra de los clientes.
 - **Procesar solicitudes:** Atender las solicitudes en el mismo orden en que fueron registradas.
 - **Consultar solicitud en proceso:** Ver la solicitud actual que está siendo atendida.
 - **Listar solicitudes pendientes:** Mostrar todas las solicitudes que están en espera.
-

3. Gestión de Clientes en Espera:

- **Registrar clientes en espera:** Almacenar los clientes que llegan para ser atendidos.
 - **Atender clientes:** Atender a los clientes en el orden en que llegaron.
 - **Consultar lista de espera:** Mostrar la lista de clientes que aún no han sido atendidos.
-

4. Gestión de Registro de Cambios en el Inventario:

Cada vez que se realice una modificación en el inventario (agregar o eliminar productos), el sistema debe registrar ese cambio en un historial, permitiendo deshacer la última modificación realizada en caso de que haya sido un error.

Requisitos para esta gestión:

- El sistema debe almacenar cada cambio realizado en el inventario (como agregar o eliminar un producto).
 - Debe ser posible "deshacer" la última acción realizada, devolviendo el sistema al estado anterior.
 - Solo se podrá deshacer una acción a la vez, y cada vez que se deshaga un cambio, el sistema debe mostrar qué acción fue revertida.
-

Pautas para los Estudiantes:

- **Inventario:** ¿Qué estructura permite almacenar y organizar productos para búsquedas rápidas?
 - **Solicitudes de Compra:** ¿Qué estructura permite atender solicitudes en el mismo orden en que llegan?
 - **Clientes en Espera:** ¿Qué estructura es ideal para manejar una fila de espera?
 - **Registro de Cambios:** ¿Qué estructura permite registrar cada cambio y deshacer la última modificación?
-

Ejemplo del Comportamiento Esperado del Registro de Cambios:

1. Se agrega un producto al inventario: *"Producto A agregado."*
 2. Se elimina un producto del inventario: *"Producto B eliminado."*
 3. Al seleccionar la opción "Deshacer", el sistema revierte la eliminación de *"Producto B"* y lo restaura al inventario.
 4. Si se vuelve a seleccionar "Deshacer", el sistema revierte la acción de agregar el *"Producto A"*.
-

Evaluación:

1. **Análisis y selección de estructuras de datos (40%):** Justificación adecuada de las estructuras elegidas.
 2. **Implementación correcta (40%):** Código funcional, bien estructurado y modular.
 3. **Documentación y claridad (10%):** Comentarios explicativos y claridad en el código.
 4. **Interfaz de usuario (10%):** Menú funcional, intuitivo y organizado.
-