

Trabajo Practico N4

Alumno: Nicolas Montagut

Materia: Laboratorio de Computación II

Curso: 2D

Mercado Sano:

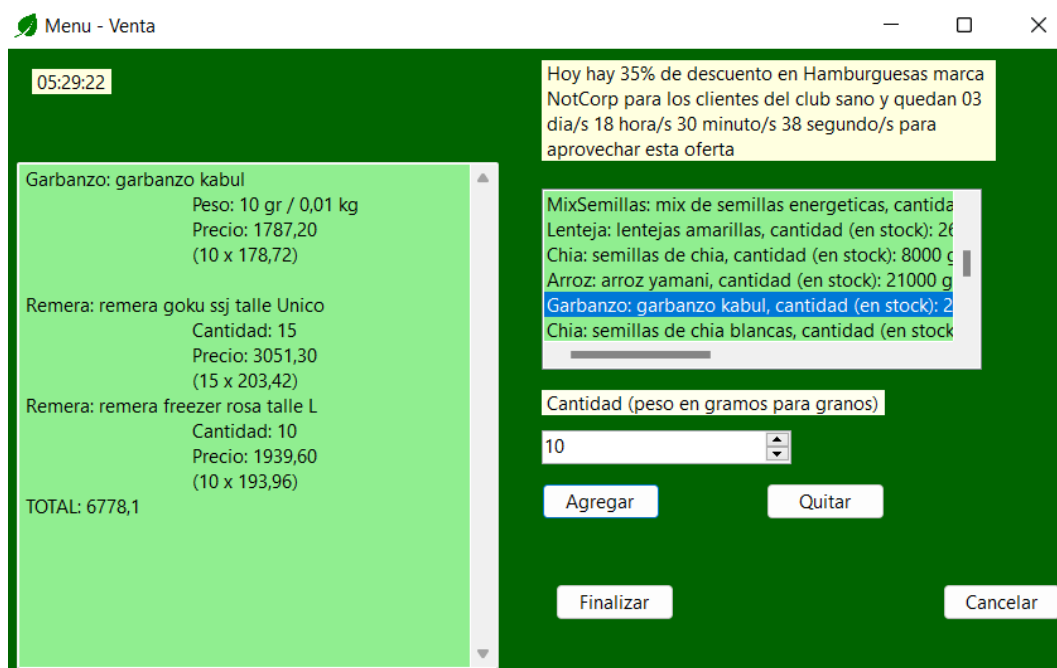
El proyecto es un sistema para la gestión de una tienda y brindar atención al cliente. Al iniciar el programa se presenta el menú principal con cuatro opciones para realizar.



Al clickear administrar stock veremos una lista vacía, ese es nuestro stock que podemos ir llenando y gestionando el stock (en la carpeta del tp4, donde también se encuentra script.sql, está productos.json, lo puede mover a la carpeta ArchivosTPFinal, ahí es donde se guardan los archivos .json y .txt).



Para brindar atención al cliente elegimos en el menú principal la opción de comenzar una nueva venta. Inmediatamente se pregunta si el cliente es miembro del club de descuentos de la tienda, Club Sano. Si lo es entonces le pedimos su ID para corroborar. Una vez corroborado tendremos un menú de ventas donde veremos el descuento actual, una lista con todos los productos en el stock, podremos agregar los productos que el cliente viene a comprar y a la izquierda se irá formando el ticket de la compra. Una vez le demos a finalizar, el ticket se imprimirá y podremos chequearlo accediendo al historial de ventas desde el menú principal o verlo en la carpeta ArchivosTPFinal en el escritorio, en el archivo Historial-De-Ventas-[mes]-[año].txt.



En caso que el cliente comprador no sea miembro del club de descuentos club sano, podemos ofrecerle unirse y luego continuar con su compra.

Por último, un cliente miembro puede solicitar darse de baja o que rectifiquemos información suya. Si solicita la baja se borrará su email de nuestra base de datos y se marcará como inactivo.

TEMAS:

Excepciones: `IdNoEncontradaException`, metodo `BuscarClienteActivoPorId` en `LogicaNegocio`, `NoHayCantidadDelProductoException`, metodo `Agregar` en clases `Elaborado`, `Grano`, `Merchandise`.

Pruebas Unitarias: proyecto `TestUnitarios`, se testea que al agregar un producto al carrito se le reste una cantidad al producto en stock y se sume al producto en el carrito si ya había una cantidad, que al removerlo del carrito se vuelva a sumar al producto en stock, se reste del carrito, se remueve si queda 0 cantidad en el carrito y se lancen excepciones cuando se intente agregar o quitar más de lo posible.

Tipos Genericos: Producto y sus clases derivadas (grano, elaborado, merchandise), interfaz IGuardable y las clases que la implementan (Serializador y HistorialDeVentas)

Interfaces: IGuardable implementada por las clases Serializador y HistorialDeVentas e IStockeable implementada por las clases Elaborado, Grano, Merchandise.

Archivos y serialización: los archivos quedan en la carpeta ArchivosTPFinal en el escritorio, Serializador serializa y deserializa JSON, HistorialDeVentas guarda los tickets de venta de cada mes en un archivo .txt.

SQL y Base de datos: ManejadorBD guarda, lee y modifica los clientes en la base de datos.

Delegados y expresiones lambda: expresiones lambda en Carrito, en los metodos MostrarCarrito y en iniciarTask, delegados en Cliente metodo CambiarInformacion y en LogicaNegocio metodo AgregarCliente.

Hilos: en Carrito y MenuVenta para actualizar lblHora y lblDescuento.

Eventos: en Carrito y MenuVenta para actualizar lblHora y lblDescuento.