

Programación Avanzada



Taller N°3

Asignatura: Programación Avanzada

Profesor: Tomás Reimann (tomas.reimann@ce.ucn.cl)

Ayudantes: Rodrigo Aguilera, Cristian Jiménez, Diego Cortés

Integrantes:

- Vicente Espinoza Lazo, 21.551.776-4, vicente.espinoza02@alumnos.ucn.cl
- Benjamín Nuñez Cepeda, 21.228.231-6, benjamin.nunez05@alumnos.ucn.cl

Facultad de Ingeniería y Ciencias Geológicas – Universidad Católica del Norte

1. Introducción

El presente proyecto corresponde al desarrollo de un **sistema informático para la cadena de minimarkets “Distribución Segura”**, que busca digitalizar los procesos de **gestión de empleados, productos y ventas** en su primera sucursal ubicada en **Antofagasta**.

El sistema tiene como propósito **automatizar tareas que antes se realizaban de forma manual**, tales como la administración de inventario, el registro de personal y la gestión de ventas con incentivos.

Mediante un enfoque de **Programación Orientada a Objetos (POO)**, se implementan mecanismos de **herencia e interfaces** para modelar de manera eficiente los distintos tipos de empleados y productos, garantizando una arquitectura flexible, extensible y de fácil mantenimiento.

Roles del sistema

- **Administrador:** posee acceso completo al sistema. Puede registrar nuevos empleados, listar información general y consultar todas las ventas.
- **Cajero:** realiza ventas aplicando un **descuento aleatorio** a una categoría de productos (Alimentos o Bebidas) por sesión, y recibe un **incentivo del 5 %** del total vendido.
- **Reponedor:** gestiona el **stock**, reponiendo o eliminando productos según su categoría asignada y registrando la cantidad total repuesta durante su jornada.

Archivos manejados

El sistema inicializa su información a partir de dos archivos de texto:

- **empleados.txt:** contiene los datos de cada empleado (tipo, RUT, contraseña, nombre, remuneración y campos específicos).
- **productos.txt:** contiene los datos de cada producto (tipo, código, nombre, precio, marca, cantidad, proveedor y campos adicionales según el subtipo).

2. Diagrama de Clases

-> El siguiente diagrama de clases representa la estructura general del sistema desarrollado para el Taller 3.

En él se modelan las **principales entidades** del minimarket “Distribución Segura”, destacando las relaciones de **herencia** e **interfaces** que permiten implementar un diseño flexible y modular.

El diagrama incluye tres jerarquías principales:

- Empleados: conformados por las clases Administrador, Cajero y Reponedor, que heredan de la clase base Empleado. Cada tipo posee comportamientos específicos según su rol.
- Productos: estructurados a partir de la clase Producto y sus subtipos (Bebida y Alimento), que se dividen a su vez en clases más específicas como BebidaAlcoholica, BebidaCaliente, AlimentoPerecible, entre otras.
- Ventas y gestión del sistema: el sistema incluye las clases Venta, DetalleVenta, Sesion y DescuentoSesion, las cuales permiten controlar las transacciones, los descuentos aleatorios aplicados al iniciar la sesión de un cajero y los incentivos por venta.

Este diseño permite representar de forma clara cómo interactúan los distintos elementos del sistema y facilita la posterior implementación en **Java**, aplicando los principios de la **Programación Orientada a Objetos (POO)**, tales como podrían ser la **herencia**, la **abstracción**, entre otros.

Horas dedicadas

- Vicente Espinoza Lazo: 3 horas dedicadas(Informe, Diagrama de clases).

- Benjamin Nuñez Cepeda: 5 horas dedicadas (Informe, Contratos, Diagrama de clases).

Conclusiones:

La primera etapa del **Taller 3** permitió establecer las bases conceptuales y estructurales del sistema.

El diseño orientado a objetos, junto con el uso de **herencia e interfaces**, facilita la separación de responsabilidades entre los distintos tipos de empleados y productos.

La definición de contratos claros garantiza la correcta interacción entre las clases y los usuarios dentro de los parámetros establecidos, mientras que el **diagrama de clases** servirá como guía para la implementación del programa en **Java** durante la entrega final.