DOCUMENTACION PROYECTO TIENDA

JOHN ALEJANDRO OBANDO GIL

PROFESOR: JULIO CESAR CHAVARRO

UNIVERSIDAD TECNOLOGICA DE PEREIRA SEDE CUBA

INGENIERIA EN SISTEMAS Y COMPUTACION

SEMESTRE V

PROGRAMACION IV

PEREIRA-RISARALDA

20/03/2019

# Modelo programa Tienda

**Identificación del problema**

En una tienda se genera una gran cantidad de información, de la cual se quiere crear un registro y tener un control de ventas, pero hacerlo de una forma manual es un trabajo dispendioso, que se crece a medida que la tienda aumenta de producción, ventas, tamaño o aumentan los empleados. El administrador de la tienda debe ser el encargado de organizar y mantener actualizados los registros de los productos, empleados, usuarios y bonos de descuento. El empleado por otra parte debe registrar cada compra que haga un cliente o al cliente mismo.

**Descripción del programa**

El programa se encarga de llevar los registros de ventas de una tienda, en donde se cuenta con una interfaz para iniciar sesión, con un nombre de usuario y contraseña únicos dependiendo si es administrador o empleado, si el usuario del software inicia como administrador contara con acceso a agregar empleados, consultar sus estados o los datos de sus empleados, agregar y editar productos, y crear bonos de descuento para sus clientes, en cambio si el usuario inicia sesión como empleado podrá registrar las compras de los clientes, aplicar los bonos de descuento a los productos y registrar a los clientes, por último el cliente puede acumular puntos para recibir los bonos.

**Paquetes del programa**

Clases paquete *tienda*:

Interfaz gráfica paquete *ventanas*:

**Jerarquía – Herencia**

**Modelo Clases Paquete tienda**

Clases Usuarios – Heredan de clase Persona:

|  |
| --- |
| **Clase Persona** |
| Atributos:  -String código  -long id  -String nombre  -String apellidos  -long celular  Métodos:  -Getters and setters |

|  |  |  |
| --- | --- | --- |
| **Clase Administrador** | **Clase Empleado** | **Clase Cliente** |
| Atributos:  -Super()  Métodos:  - Guardar datos  -Consultar datos  -Modificar datos | Atributos:  -Super()  -int numeroVentas  -Boolean activo  Métodos:  -Getters and setters  -Guardar datos  -Consultar datos  -Modificar datos  -Modificar activo | Atributos:  -Super()  -int puntos  Métodos:  -Guardar datos  -Consultar datos  -Modificar datos  -Modificar puntos |

Clase Bono y Productos:

|  |  |
| --- | --- |
| **Clase Bono** | **Clase Producto** |
| Atributos:  - String codigo  - String descuento  Metodos:  -Getters and setters  -Guardar datos  -Consultar datos  -Modificar datos | Atributos:  - String codigo  - String nombre  - String precio  Metodos:  -Getters and setters  -Guardar datos  -Consultar datos  -Modificar datos |

**Modelo Interfaz gráfica paquete *ventanas***

|  |  |  |
| --- | --- | --- |
| **Nombre y Descripción** | **Componentes** | **Función componentes** |
| Class Tienda  Es la interfaz para el inicio de usuario  Inicia sesión como:  -Adminitrador  -empleado | - JButton  - JButton  - JLabel  - JLabel  - JLabel  - JPasswordField  - JTextField | -Botón salir  -Botón iniciar sesión  -Fondo  -Imprime usuario  -Imprime contraseña  -lee la contraseña  -lee el usuario |

|  |  |  |
| --- | --- | --- |
| **Nombre y Descripción** | **Componentes** | **Función componentes** |
| Class Admin  Es la interfaz para el manejo de los empleados, productos y bonos de la tienda | - JButton  - JButton  - JButton  - JButton  - JButton  - JButton  - JButton  - JButton  - JButton  - JLabel | - Botón agregar empleado  - Botón estado empleado  - Botón consultar empleado  - Botón listar empleados  - Botón agregar producto  - Botón editar producto  - Botón crear bono  - Botón listar bonos  - Botón regresar  - Fondo |

|  |  |  |
| --- | --- | --- |
| **Nombre y Descripción** | **Componentes** | **Función componentes** |
| Class Compra  Es la interfaz para el manejo de las compras aplicación de bonos y botón de registro de clientes de la tienda | - JButton  - JButton  - JButton  - JButton  - JButton  - JButton  - JButton  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JTextField  - JTextField | - Botón buscar producto  - Botón buscar bono  - Botón comprar  - Botón finalizar  - Botón puntos  - Botón registrar  - Botón salir  - Imprime producto  - Muestra nombre del producto  - Imprime código  - Imprime Precio  - Muestra precio del producto  - Imprime total acumulado  - Muestra suma total de la compra  - Imprime descuento  - Fondo  - lee código del producto  - lee código de bono descuento |

|  |  |  |
| --- | --- | --- |
| **Nombre y Descripción** | **Componentes** | **Función componentes** |
| Class RegistroCliente  Es la interfaz para el registro de los clientes de la tienda | - JButton  - JButton  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JLabel  - JTextField  - JTextField  - JTextField  - JTextField | - Botón registrar cliente  - Botón salir  - Imprime documento  - Imprime nombre  - Imprime apellidos  - Imprime celular  - Imprime titulo  - Fondo  - lee documento  - lee nombre  - lee apellidos  - lee celular |

**Funcionamiento del programa**

El siguiente es un esquema explicativo solo de los bonos, los métodos y funciones que contienen la clase bono y funcionamiento en las interfaces y el resto del programa, y es solo para esta clase ya que de la misma forma son utilizados los métodos de la clase bono como agregar, consultar, listar y modificar en el resto de las clases del proyecto como administrador, empleado, cliente y se volvería muy largo el modelo.

Clase Bono:

- Los bonos solo pueden ser creados y listados por el administrador.

- Los bonos solo pueden ser aplicados por el empleado.

**Archivos de registro**

Base de datos para el almacenamiento de la información.

**Criticas**

El programa cuenta con una gran variedad de funciones que son necesarias para llevar registros en una tienda, como lo son la interfaz para el administrador el cual es el único autorizado para agregar nuevos empleados al registro, ingresar nuevos productos y más. Por otro lado está la interfaz para los registros de las ventas, de la cual se encargan los empleados, su función es buscar los productos, registrar las compras, aplicar los bonos de descuento y registrar nuevos clientes.

La estructura del programa tiene muy buenas bases, el programa incorpora el uso de registros para guardar la información más relevante de una tienda todo de una manera organizada, además el código en sí mismo está muy bien formado, está bastante simplificado y es bastante legible y entendible. Hace un buen uso de la programación orientada aplicada a objetos, definiendo claramente los atributos y métodos de cada clase y utilizando conceptos como la herencia de clases.

El programa podría tener mayor eficiencia, ser más versátil y ser más intuitivo con algunos de los siguientes elementos:

- Tener una interfaz para crear nuevas cuentas de administrador en vez de que el programa venga con un usuario y contraseña predeterminado para el administrador, ya que tal vez se releve ese puesto o se pierda su contraseña, y se tengan que crear nuevos usuarios.

- Mejorar las interfaces haciéndolas más intuitivas y tal vez con un menú donde reúna las opciones y las divida según su tipo. En la interfaz compras por ejemplo se podrían agregar nuevas funciones como agregar cuantos productos se quiere de un mismo tipo, también se podría agregar un buscador por nombre del producto en vez de buscar por el código y que se vallan listando todos los productos del mismo tipo pero con diferente marca.

- Otro punto importante es el registro de los clientes ya que en un principio no queda claro como interactuar con ellos en el programa ni como se almacena la información, basado en esto, otra mejora podría ser manejar datos estadisticos de ventas, como cuál es el producto que más se vende y que tipo de usuario es el que más compra en la tienda, para con esta información mejorar las ventas en general.

**Funcionalidad a mejorar**

Realizar algunos cambios en la interfaz de compras para que el empleado pueda interactuar con el programa más eficazmente o crear una interfaz para nuevas cuentas de administrador en vez de que el programa venga con un usuario y contraseña predeterminado para el administrador, ya que tal vez se releve ese puesto o se pierda su contraseña, y se tengan que crear nuevos usuarios.