

# SISTEMA DE VENTAS

## 1. Introducción

El presente proyecto de programación orientado a objetos se basa en el análisis y diseño de un sistema de ventas de un negocio dedicado a la venta de licores.

El proyecto se desarrolló con el propósito de automatizar tareas, como por ejemplo el control de los stocks o facturar de manera eficiente, lo que conlleva a mejorar la gestión de la información para la toma de decisiones del negocio.

El objetivo principal del proyecto es automatizar el control del inventario y la facturación del negocio.

## 2. Motivación y Problema

Motivación: La idea del proyecto surge al observar la gran demanda que existe en este tipo de productos que no se limita a un solo sector y/u ocasión en específico, pues evidente la venta masiva de bebidas alcohólicas para personas mayores de 18 años para su posterior consumo en distintos modelos de eventos o situaciones tales como: cumpleaños, bodas, bautizos, reuniones familiares, etc.

Problema: Como todo negocio con productos de alta demanda, se necesita llevar un control de lo que se vende y lo que no, de los gastos para así evitar desperdicios o despilfarros, conocimiento del estado financiero de la empresa, etc. Además de esto se requiere, para la facilidad de las transacciones, un sistema que permita pagar lo adquirido en la modalidad que desee (efectivo o tarjeta), así como la emisión de boletas o facturas. Esto y más es lo que se buscará cubrir con la aplicación que desarrollaremos en el presente proyecto, esto se explicará mejor en la lista de requerimientos.

## 3. Requerimientos

IN SCOPE REQUIREMENTS			
Business Function / Category	Type	Requirement	Priority
venta de bebidas alcohólicas/ industria de bebidas	(Functional, Technical, Operational, Transitional)		(Core, Essential, Desired)
Para el vendedor	Funcional	El programa permitirá ingresar los datos del consumidor.	Nucleo
Interfaz grafica	Funcional	El sistema permitirá ingresar los datos de los productos .	Nucleo
Para el producto	Funcional	El sistema permitirá leer códigos opcional al nombre del producto.	Nucleo

Interfaz grafica	Funcional	Las unidades en almacen y ventas admitirán numeros de hasta 5 cifras	Necesario
Interfaz grafica	Funcional	Los valores unitarios del precio y total de ventas deberan admitir 2 valores decimales como máximo	Necesario
<u>Para el producto</u>	Funcional	base de datos( producto, marca, unidades, precio)	Necesario
Para el producto	Funcional	El sistema permitira eliminar los productos vendidos luego de la transaccion	Nucleo
Para el producto	Funcional	El sistema actualizara la disponibilidad de los productos constantemente	Necesario
Interfaz grafica	Funcional	El sistema permitira ver el precio y la disponibilidad del producto	Nucleo
Para el vendedor	Funcional	El programa hara las respectivas operaciones para obtener el monto total a pagar de los productos por el comprador	Nucleo
Para el administrador	Funcional	Al confirmarse la venta, el programa guardara la informacion para poder verificarla	Necesario
Para el cliente	Funcional	El programa brindara una boleta donde se indique el nombre del vendedor, el nombre del consumidor, el precio, la fecha, el codigo del producto, el dinero recibido, el vuelto y la cantidad de productos vendidos.	Necesario
Para el administrador	Funcional	El sistema brindará el día y la hora de la venta del producto	Necesario
Para el vendedor	Seguridad	El programa pedira un usuario y contraseña para iniciar	Necesario
Interfaz grafica	Uso	El programa contara con una interfaz grafica intuitiva para el vendedor	Opcional
<u>Para el administrdor</u>	Seguridad	El sistema guardara en la memoria la cantidad de dinero ingresado durante 24 horas	Opcional
Para el administrador	Seguridad	El programa guardara los datos de la venta(codigo del producto, nombre del consumidor, fecha y hora) en caso de algun reclamo	Opcional
Para el administrdor	Uso	Los vendedores no pueden modificar los datos del sistema	Necesario
Interfaz grafica	Uso	Si existe promoción de algun producto, el programa permitira ingresar un codigo para poder realizar la respectiva venta	Opcional
Para el producto	Uso	El sistema permitirá aumentar el stock del establecimiento	Necesario
Para el administrdor	Seguridad	Los administradores pueden modificar pero no eliminar datos del sistema	Opcional

Para el administrador	Uso	Todo lo hecho por un usuario debe estar registrado y debe verse en todos los usuarios	Opcional
Para el vendedor	Seguridad	El inicio de sesión del usuario no debe pasar de los 5 minutos	Opcional
Para el usuario	Seguridad	En caso de algún error, el sistema especificará y enviará al usuario los detalles del error	Opcional
Para el usuario	Uso	El sistema no debe presentar fallas al momento de la venta	Necesario

## 4. Procedimientos

### 4.1.1. Etapas de desarrollo

Etapa 1. Planeación y evaluación del proyecto.

Se elaboró un plan de proyecto que permita identificar aspectos relacionados con tiempo, recursos humanos, calidad, etc.

Etapa 2. Recopilación de requisitos

Se realizaron preguntas entre los autores para recopilar los requisitos del sistema.

Etapa 3. Análisis de requerimientos y diseño del sistema.

Etapa 4. Desarrollo del software.

El sistema cuenta con los siguientes módulos:

Archivo: Se registra al cliente y su número de tarjeta.

Venta: Se realiza las operaciones del negocio y te permite visualizar las mismas.

Almacén: Permite mostrar los productos del negocio, con sus respectivos códigos, precios y cantidad, previamente agregados.

Etapa 5. Prueba del software.

## 5. Arquitectura del sistema

- Diagrama de componentes

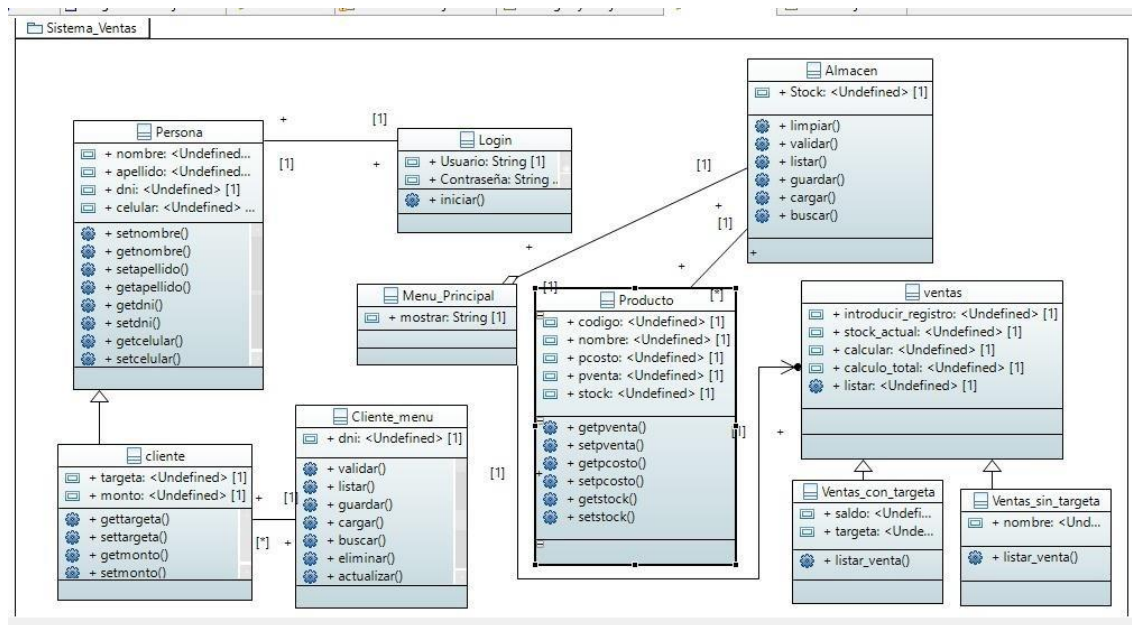
El sistema cuenta con módulos claramente definidos los cuales acceden al modelo de datos único de la solución. Los módulos son los siguientes:

Modulo Archivo

Modulo Venta

Modulo Almacén

- Diagrama de clases



El diagrama de clases permite presentar claramente el enfoque de la solución. Se buscó crear una clase que permita ingresar Usuario y Contraseña. Esta clase sería Login estará asociada con la clase Persona y Menu\_Principal y estas últimas contienen los métodos o llamados adecuados a otras clases para generar las acciones.

La clase Persona tiene Atributos como: nombre, apellido, Dni, celular; y Métodos como: setnombre, getnombre, getdni, setdni, etc. Dichos atributos y métodos serán heredados por la clase Cliente que a su vez tendrá sus propios métodos y atributos.

La clase Almacén está agregada en la clase Menu\_principal, como se podrá visualizar en la interfaz gráfica.

La clase Producto está asociada a la clase Almacén, en esta se podrá registrar los distintos licores con su respectivo código, nombre, precio de costo, precio de venta, stock.

La clase Venta presentará dos clases hijas (Ventas\_con\_tarjeta y Ventas\_sin\_tarjeta), dichas clases heredarán los atributos y métodos de la clase Venta.

## 6. Herramientas utilizadas

Se usará el diagrama de clases para identificar los tipos de relaciones entre las clases (herencia, dependencia, agregación, asociación, etc.), identificando en cada clase los atributos y métodos correspondientes.

La base de datos se crea con el método Serializable.

Clases: Login, menú principal, persona, cliente, cliente\_menú, almacén, ventas, producto, venta con tarjeta, venta sin tarjeta.

Métodos usados: String.valueOf(), Integer.parseInt(), Double.parseDouble, get, set, for.

Java.Swing: Para implementar la interfaz gráfica. (Jframe, Jpanel, Jbutton, JLabel, Jtable, JTextField, JPasswordField, etc)