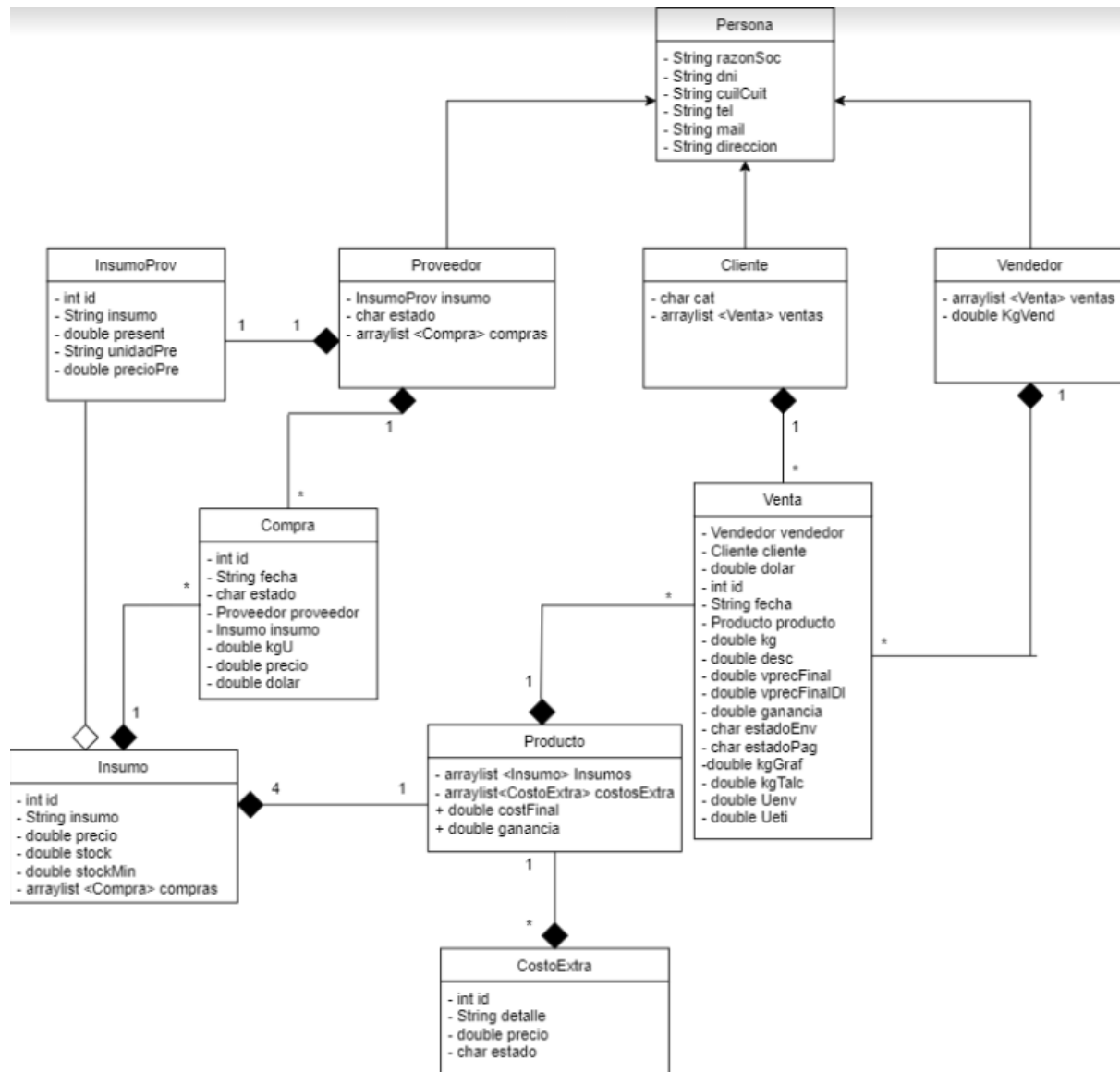


Criterio B - Diseño

Se realiza una entrevista con mi cliente en la cual se obtiene la información necesaria para proceder con el diseño del sistema. (Anexo B). Se realiza el primer boceto (Anexo C)

Estructura de clases:

El proyecto contiene distintas clases en función de objetos, en el siguiente diagrama presentamos las relaciones entre ellas y sus atributos.



Descripción de clases:

Clase Persona: Esta clase cuenta con atributos comunes entre todas las categorías del programa, por lo que podemos llamarla una superclase. Los mismos son, Razon social, DNI, Cuil/cuit, Telefono, Mail y Direccion.

Clase Proveedor: La clase proveedor, como su nombre lo dice, almacena los datos de cada proveedor. La misma cuenta con los mismos atributos que la clase persona,

debido a la herencia y además atributos característicos de la categoría, entre ellos, el respectivo insumo que provee, el estado, si se encuentra activo o no, y una lista con las respectivas compras que se le han realizado. Esta clase se relaciona de 1 a 1 con la clase insumoProv, lo que se debe a que cada proveedor solo provee un insumo (anexo 1). Gracias a esta clase podemos obtener el precio por kg del insumo.

Además, el proveedor se relaciona 1 a muchos con la clase compra, dentro de ella vemos atributos característicos como número de compra, precio, cantidad, entre otros.

Clase InsumoProv: Dentro de esta clase encontramos atributos como el código del insumo, su descripción, su presentación, si viene en de a x cantidad de kg o de unidades, la unidad de esa presentación, sería kg o unidad y el precio de esa presentación. La misma se encuentra relacionada débilmente con la clase insumo a través del código de insumo, esta relación es llamada de composición, la misma permite luego obtener el stock y el costo por kg/unidad de cada insumo.

Clase Compras: En esta clase podemos observar atributos como; el número de pedido, la fecha, el estado de la compra, si está entregada o no, a qué proveedor pertenece, el código del insumo, la cantidad de unidades o kg pedidos, el valor del dólar en el momento de la compra y el precio final de la misma. Esta clase mantiene una relación fuerte con la clase insumo, lo que permite que a través de ello se modifiquen los costos y el stock. Además, está relacionada fuertemente con la clase proveedor e indirectamente con la clase Insumo Prov, ya que luego esto permitirá una comparación de precios y una modificación si es necesario.

Clase Cliente: Esta clase está destinada al almacenamiento de los datos de los clientes que han realizado pedidos. La misma es heredada de la clase personal y además cuenta con atributos como, la categoría a la que pertenece, y sus respectivos pedidos. Además mantiene una relación fuerte de 1 a muchos con la clase ventas, la cual también está relacionada con la clase de producto final y la clase insumo, lo que nos permite modificaciones de stocks constantes.

Clase Ventas: Esta clase además de estar relacionada con la clase cliente también lo está con la clase vendedor, manteniéndolos a ambos como atributos. Por otro lado, también cuenta con atributos característicos de una compra como, número de pedido, fecha del mismo, la cantidad de kg pedidos, si hubo algún descuento, el valor del dólar a la fecha, los estados de pedido, si está pago o no y si está entregado o no, el objeto producto, con el que obtendremos el valor final de la venta en pesos y en dólares, y su respectiva ganancia.

Clase Producto: Dentro de ella encontramos dos listas como atributos, en primer lugar tenemos la lista de insumos, la cual nos permite tener en cuenta los costos de cada uno para el cálculo del costo final, y la lista costos agregados, como por ejemplo, mano de obra, flete, entre otros. Los cuales también influyen en el cálculo del costo final. Por otro lado, esta clase mantiene los atributos de costo final y ganancia, que luego serán utilizados por la clase venta.

Clase Insumo: La clase insumo está relacionada fuertemente con el producto, es una relación de 4 a 1, y contiene el código de insumo, la descripción del mismo, el

precio por kg, el stock físico y el punto de pedido o stock mínimo. Además mantiene una lista de compras, la cual es posible debido a su relación fuerte de 1 a muchos con la clase compra.

Clase Vendedor: La clase vendedor es heredada de la clase persona y además contiene un atributo que almacena la cantidad de kg que él mismo ha vendido y una lista que contiene las ventas que él ha realizado. Por lo tanto está relacionada de 1 a muchos con venta.

Estructura de datos:

Clase Persona		Observaciones
razonSoc	String	Descripción de la empresa o persona
dni	String	Dni de la persona a cargo
cuilCuit	String	Cuil/cuit de la persona o la empresa
tel	String	Teléfono de la persona o de la empresa
mail	String	Mail de la persona o de la empresa
dirección	String	Dirección física de la persona o de la empresa
Getters and Setters:	String/Void	Permiten obtener y modificar los atributos

Clase Proveedor		Observaciones
insumo	InsumoProv	Atributo de tipo InsumoProv que mantiene los datos necesarios del insumo que el mismo provee
estado	char	Si el mismo se encuentra activo, si se le hacen compras habituales, o si solo esta para emergencias o comparación de precios
compras	arraylist Compra	Lista de tipo Compra que almacena todas las compras que se le han hecho a ese proveedor
addCom():	void	Agregar una compra en la lista de compras
verCom():	void	Lista las compras que se le han hecho al proveedor
verProv():	void	Lista los datos del proveedor
modProv():	void	Facilita el hecho de modificar a un proveedor
verHis():	void	Lista solo algunos datos particulares del proveedor
verCompPE():	void	Ver compras pendientes de entrega

modInsumo():	void	Modificar insumo proveedor
verInsumo():	void	Ver datos de insumo proveedor

Aclaraciones:

- El atributo insumo es de tipo InsumoProv debido que la clase mencionada es la que mantiene los atributos que almacenan la información esencial para llevar un control de el precio del insumo por proveedor y de otros datos importantes, la misma mantiene una relación uno a uno con Proveedor, por lo que existe la posibilidad de que haya varios proveedores a los que se le compre el mismo tipo de insumo pero que no pertenezcan al mismo objeto, teniendo así diferentes valores en los atributos de la clase. A diferencia de la clase Insumo, InsumoProv no mantiene atributos que hagan referencia a datos más globales del tipo de insumo.
- El método VerHis() refiere a el muestreo de datos específicos de identificación, como, Razón Social, CUIL/CUIT, telefono e insumo, y datos particulares de las compras al proveedor, por lo que se accedera a cada una de las compras y se mostrarán los kg comprados, a que precio, en que fecha y a que valor de dolar.

Clase InsumoProv		Observaciones
id	int	Código del tipo de insumo
insumo	String	Descripción del insumo

present	double	Presentación del insumo, en que cantidad de unidades o kg es comprado
unidadpre	String	La unidad en la que se encuentra el numero ingresado anteriormente (Kg/U)
preciopre	double	precio por presentación

Aclaraciones:

- El atributo codinsumo es el identificador del tipo de insumo, por lo tanto es fundamental para las relaciones débiles que la clase mantiene con la clase InsumoProv con las clases Insumo y Compra.

Clase Compra		Observaciones
id	int	Número de compra general (Autonumérico)
fecha	String	Fecha de la compra
estado	char	Estado de la compra, si se encuentra entregada (E) o no entregada (N)

proveedor	Proveedor	El proveedor al cual se le ha realizado la compra
insumo	Insumo	código del insumo que fue comprado
kgU	double	Cantidad de kg o unidades que fueron compradas
precio	double	Precio final de la compra realizada
dólar	double	Precio del dólar al momento de la compra
modComp():	void	Modificación/es de la compra realizada
verComp():	void	Listar la compra con sus respectivos datos
verE():	void	Lista datos particulares de la compra, relacionados al estado de la misma
verPH():	void	Lista datos particulares para el historial de proveedores

Aclaraciones:

- Método verE(), el mismo listara datos específicos como el id, la fecha, la razón social y el cuil/cuit del proveedor, el código de insumo y el estado. Lo que generará una visión panorámica de la compra, con el fin de luego realizar diversos procesos.
- Método verPH(), este método listará datos específicos como, el id, la fecha, los kg comprados, el precio, y el dólar. El mismo será utilizado por la clase Proveedor para ver los historiales.

Clase Cliente		Observaciones
cat	char	La categoría del cliente, si es distribuidor o particular
ventas	arraylist Venta	Lista de tipo Venta que almacena todas las ventas que ha realizado ese cliente
AddVent):	void	Agregar una venta a la lista de ventas
ModCli():	void	Modificar los datos de el cliente
VerCli():	void	Listar los datos de cliente
VerVent():	void	Listar las ventas realizadas a ese cliente

Clase Venta		Observaciones
vendedor	Vendedor	El vendedor que ha concretado la venta
dólar	double	valor del dólar a la fecha del pedido

cliente	Cliente	El cliente que la ha realizado
id	int	El número de venta general (Autonumérico)
fecha	String	La fecha en la que se ha realizado la venta
producto	Producto	El producto que se ha vendido que proviene de la clase producto
kg	double	La cantidad de kg que se han vendidos de producto final
desc	double	El porcentaje de descuento que se realizado, en el caso de no haber se ingresara 0
vprecFinal	double	Donde se guarda el precio final en pesos de la compra luego de ser calculado en la clase Producto
vprecFinalDI	double	Donde se guarda el precio final en dólares de la compra luego de ser calculado en la clase Producto
Ganancia	double	Donde se guarda la ganancia calculada en la clase Producto.
estadoEnv	boolean	El estado de la venta, si se ha enviado(TRUE) y si no se ha enviado (FALSE)
estadoPag	boolean	El estado de la venta, si se ha abonado (TRUE) y si no se ha abonado (FALSE)
KgGraf	double	Almacenan el stock que utilizo esa compra de cada uno de los insumos
KgTalc	double	
Uenv	double	
Ueti	double	
modVen():	void	Modificar la venta
verVen():	void	Listar los datos de la venta

verH():	void	Listar datos seleccionados
stocks():	double []	Calcula el stock utilizado de cada insumo para generar la cantidad de KG de la compra.

Aclaraciones:

- El método ver() mostrará datos particulares de cada venta como, la cantidad de kg que fueron vendidos, el precio de la venta, si hubo descuentos, cuales, y la ganancia de esa venta.

El método stocks(): permitirá calcular el stock utilizado en esa venta para luego poder restarle al stock actual de cada insumo.

Clase Producto		Observaciones
insumo	arraylist Insumo	Lista con los objetos de clase insumo con todos sus atributos
costosExtra	arraylist costoExtra	Lista con los costos extra para utilizar en el cálculo del costo final
costFinal	double	atributo que almacena el costo final de cierta cantidad de KG de producto al realizar una venta
ganancia	double	Atributo donde se almacenará la cantidad de pesos de ganancia que se calculan para cierta cantidad de KG, se utilizará al realizar una venta
verIns():	void	Muestra todos los insumos
modIns():	void	Modificar un insumo particular
verSE ():	void	Muestra todos los costos extra

modCE();	void	Modifica un costo extra particular
AgCE():	void	Agregar un costo extra para ser utilizado en las ventas
veriE():	boolean	verifica la existencia o la no existencia de el nuevo costo extra agregado
calCosIna():	double	Calcula el costo final de los insumos necesarios para generar un KG de producto
calCostF():	double	Calcula el costo final teniendo en cuenta el costo de los insumos y los costos extra (Dentro de ellos la ganancia)
calc Costos des():	double []	Calcula el costo final con la cantidad de KG del pedido teniendo en cuenta lo obtenido de el calCos(), teniendo en cuenta que se han agregado descuentos.
calcCostos():	double []	Calcula el costo final con la cantidad de KG del pedido teniendo en cuenta lo obtenido de el calCos(), teniendo en cuenta que NO se han agregado descuentos.

Aclaraciones:

- La clase producto mantiene 2 listas en las cuales almacena todos los datos necesarios para calcular el costo final del producto.
- Lista Insumo, la misma contiene los diferentes insumos y sus datos. Los que son necesarios para la producción del producto final que es solo uno.
- Lista costProducto, esta lista contiene los diferentes costos extra que se tienen en cuenta en el cálculo del costo final, los mismo en un principio serán, mano de obra, valor del flete, monotributo y contaduría, flete/es de distribución y porcentaje de ganancia. Aun así, se le podrán agregar nuevos costos mediante el método AgCE().
- Los métodos que devuelve información serán utilizados por diferentes métodos para mostrar y guardar esa información en diferentes fuentes de datos.

Clase Insumo		Observaciones
id	int	Código del tipo de insumo
insumo	String	Descripción del insumo
precio	double	Precio del insumo por kg o unidad
stock	double	El stock físico que hay actualmente del insumo
stockMin	double	El punto de pedido ingresado por mi cliente del insumo
compras	arraylist Compra	Lista de tipo Compra que almacena todas las compras que se le han hecho a ese insumo.
verIns():	void	Lista los datos del insumo
modIns():	void	Modificar insumo
verStock():	void	Lista el stock del insumo y los datos principales del mismo
actStock():	void	Modifica el stock del insumo
addCom():	void	Agregar una compra en la lista de compras

verHis():	void	Ver datos puntuales para el historial de compras
actPrecio():	void	Modifica el precio del insumo

Aclaraciones:

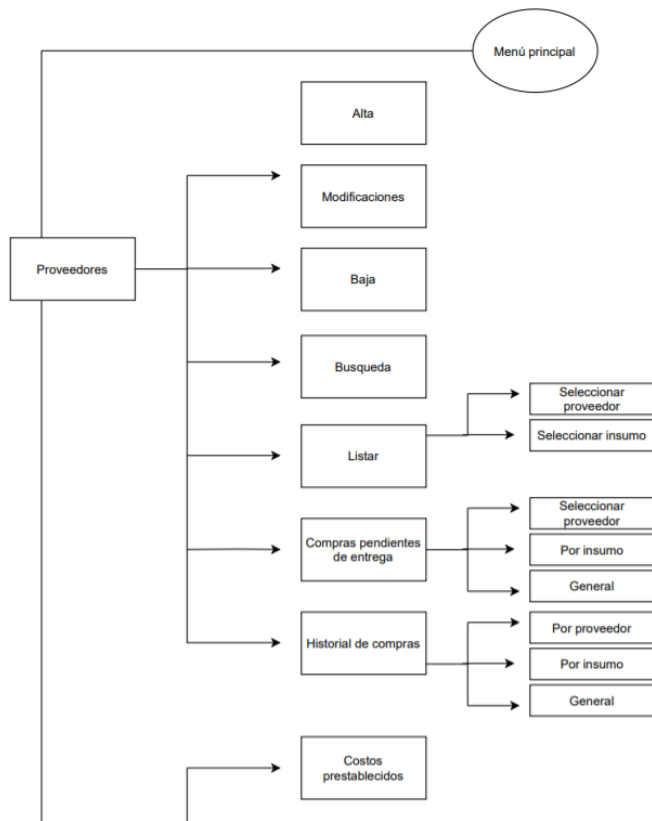
- Esta clase consta de datos esenciales para el insumo en general, en la misma no se le da ninguna relevancia a quien ha proveído el insumo, ni datos relacionados a ello.
- Como he mencionado anteriormente se mantiene una relación débil mediante codinsumo con la clase Compra e InsumoProv.
- El atributo precio será cargado por mi cliente.

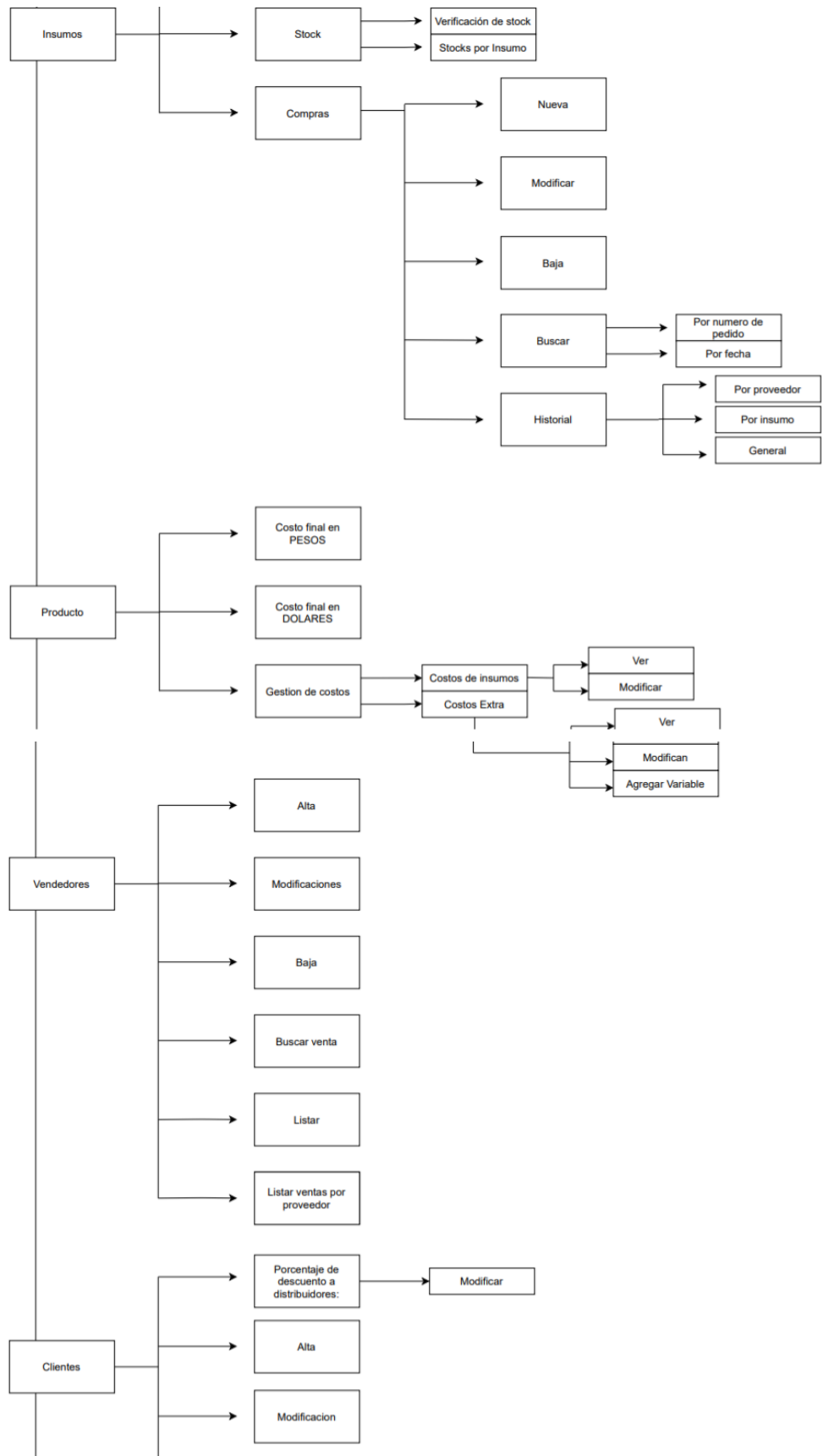
Clase Vendedor		Observaciones
ventas	arraylist Venta	Lista con todas las ventas que él ha concretado
kgVend	double	Cantidad de kg vendidos de producto final hasta el día de la fecha
AddVen():	void	Permite añadir una venta a la lista de ventas del vendedor
VerVend():	void	Lista los datos del vendedor
ModVend():	void	Permite modificar los datos del vendedor
VerVent():	void	Lista las ventas realizadas por el vendedor

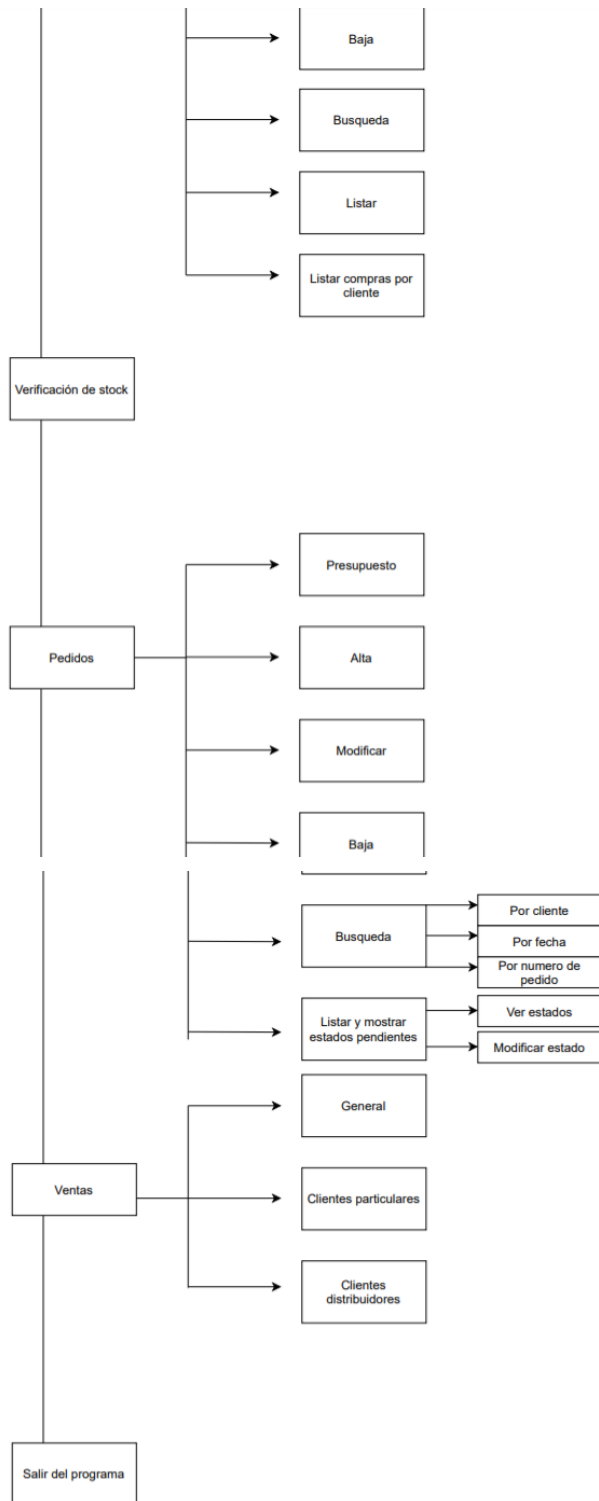
Diagrama general de procesos y menús:

Se realizan numerosos cambios y se conversa con mi clienta para nuevas mejoras.

(Anexo D)







Prototipo de pantallas en forma de tabla:

MENÚ PRINCIPAL	
1.	Proveedores
2.	Insumos
3.	Producto
4.	Vendedores
5.	Clientes
6.	Verificación de stock
7.	Pedidos
8.	Ventas
*.	Guardar
0.	Salir del sistema

MENÚ DE PROVEEDORES	
1-	Alta
2-	Modificar
3-	Baja
4-	Búsqueda
5-	Listar
6-	Listar compras pendientes de entrega
7-	Historial
0-	Salir del menú de proveedores
Ingrese la opción:	

1.	Alta
-	Cuil/Cuit:
-	El proveedor ya existe/Sigue ingresando datos
-	Razón Social
-	Dni
-	Teléfono/Celular
-	Mail
-	Dirección
-	Ingrese estado [A o D]
-	CARGA DE INSUMO (nuevo menú)

Carga de insumo
Grafito: 1 Talco industrial: 2 Envases: 3 Etiquetas: 4 - Ingrese ID: - Ingrese descripción: - Kg o unidades de presentación: - Unidad [KG si es en esa unidad o U si es en enteros] : - Precio por presentación:

2. Modificar
- Cuil/cuit del proveedor a modificar: ----- 0- Salir 1- Datos personales del proveedor: 2- Datos de insumo: Ingrese la opción:

1. Modificar datos personales del proveedor
0- Salir 1- Razón social: 2- Dni: 3- Cuil/cuit: 4- Teléfono: ... 5- Mail: 6- Dirección: 7- Estado: Ingrese la opción:

2. Modifique datos del insumo
0- Salir 1- Descripción Insumo: 2- Presentación del Insumo: 3- Unidad de la presentación del Insumo: 4- Precio de la presentación del Insumo:

Ingrese la opción:

3. Baja

| Cuil/cuit:

4. Búsqueda

| Cuil/cuit:

0- Salir

1- Ver datos personales

2- Ver compras pendientes de entrega

3- Ver historial

| Ingrese la opción:

1.

| Cuil/cuit:

5. Listar

0- Salir

1- Por proveedor:

2- Por insumo:

| Ingrese la opción:

6. Listar compras pendientes de entrega

0-Salir

1- Por proveedor

2- Por insumo

3-General

| Ingrese la opción:

7. Historial

0-Salir

1- Por proveedor

2- Por insumo

3-General

| Ingrese la opción:

1. Insumos

1- Costos preestablecidos

2- Stock

3- Compras

1. Costos preestablecidos

| --- Por unidad ó kg --- |

|1|GRAFITO: 0.0

|2|TALCO INDUSTRIAL: 0.0

|3|ENVASES: 0.0

|4|ETIQUETAS: 0.0

2. Stock

|0| Salir

|1| Verificación de stock para nuevo pedido

|2| Stocks por insumo

| Ingrese opcion:

1. Verificación de stock para nuevo pedido
Cantidad de KG a producir:

2. Stocks por insumo
<p> --- Por unidad ó kg --- </p> <p> --- En el caso de querer modificar algún stock mínimo ingrese el número correspondiente --- </p> <p> 1 GRAFITO: 0.0</p> <p> 2 TALCO INDUSTRIAL: 0.0</p> <p> 3 ENVASES: 0.0</p> <p> 4 ETIQUETAS: 0.0</p> <p> Ingrese opcion, si no desea realizar ningún cambio ingrese '0'</p>

3. Compras
<p>1- Nueva</p> <p>2- Modificar</p> <p>3- Eliminar</p> <p>4- Buscar</p> <p>5- Historial</p>

1. Nueva
<p> Cuit/Cuit del proveedor:</p> <p> KG o Unidades encargadas:</p> <p> Precio total:</p> <p> Dólar a la fecha:</p> <p> Estado [E de entregado, N de no entregado]:</p> <p> Fecha: 20/09/21[Si quiere modificar la fecha ingrese 'M', de lo contrario de ENTER]:</p>

2. Modificar
<p> Id de compra:</p> <p>Compra número, a proveedor, Por insumo</p> <p>0- Salir</p> <p>1- Fecha:</p>

2- Estado:

3- Cantidad de KG pedidos:

4- Precio final en pesos:

5- Precio dolar a fecha:

| Ingrese la opción:

3. Eliminar

| Id de compra:

|Id:

|Fecha:

|Estado:

|Proveedor:

|Id insumo:

|Kg pedidos:

|Precio final:

|Precio del dólar asociado a la compra:

| Ingrese E si desea proseguir con la eliminación, de lo contrario de ENTER:

4. Buscar

|0|Salir

|1|Por ID de compra

|2|Por fecha

| Ingrese opcion:

1/2. Por ID de compra/Por fecha

| Id de compra: /

| Día [en formato dd]:

| Mes [en formato mm]:

| Año [en formato aaaa]:

|Id:

|Fecha:

|Estado:

|Proveedor:
|Id insumo:
|Kg pedidos:
|Precio final:
|Precio del dólar asociado a la compra:

5. Historial

|0|Salir
|1|Por proveedor seleccionado
|2|Por Insumo
|3|General
| Ingrese opcion:

1/2. Por proveedor seleccionado/ por insumo

| Cuil/cuit del proveedor:

2. Producto

1- Ver costo final en PESOS
2- Ver costo final en DÓLARES
3- Gestión de costos

1/2. Ver costo final en PESOS/DÓLARES

|valor:

3. Gestión de costos

0- Salir

1- Costos de insumos

2- Costos extra

| Ingrese la opción:

1. Costos de insumo

0- Salir

1- Ver

2- Modificar

| Ingrese la opción:

1. Ver

|Id: 1

|Insumo: Grafito

|Precio: 0.0

|Stock: 0.0

|Id: 2

|Insumo: Talco industrial

|Precio: 0.0

|Stock: 0.0

|Id: 3

|Insumo: Envases

|Precio: 0.0

|Stock: 0.0

|Id: 4

|Insumo: Etiquetas

|Precio: 0.0

|Stock: 0.0

2. Modificar
* Muestra Insumos. Ingrese el id del insumo a modificar:

2. Costos Extra
0- Salir 1- Ver == Costos de Insumo 2- Modificar == Costos de Insumo 3- Agregar variable Ingrese la opción:

3. Agregar variable
Recuerde que el ID: 0 pertenece a ganancia, y el valor ingresado se utilizara como porcentaje * Muestra Costos Extra Luego de ver los costos ya existentes desea seguir agregando el ingresado?, Ingresos por sí, de lo contrario de ENTER

4. Vendedores
1- Alta 2- Modificar 3- Baja 4- Búsqueda 5- Listar 6- Listar ventas por vendedor

1. Alta
Cuil/Cuit: Razón Social: DNI: Teléfono/Celular: Mail: Dirección:

2. Modificar
Cuil/cuit del vendedor a modificar: ----- 0- Salir 1- Razón social: 2- Dni: 3- Cuil/cuit: 4- Teléfono: 5- Mail: 6- Dirección:

3. Baja
Cuil/cuit del vendedor a modificar:

4. Búsqueda
Cuil/cuit de vendedor a buscar:

5. Listar
Razón social: Cuil/Cuit: Teléfono: Mail: Dirección: Kg vendidos: Ingrese S si desea ver la lista de ventas a el vendedor,

de lo contrario de ENTER

6. Listar ventas por vendedor

| Cuil/cuit del vendedor:

5. Clientes

1- % de descuento clientes distribuidores

2- Cargar

3- Modificar

4- Baja

5- Búsqueda

6- Listar

7- Listar compras por cliente

1. % de descuento clientes distribuidores

|Descuento distribuidores: 0.0

| Ingrese M si desea modificar, de lo contrario de ENTER:

2. Alta

| Cuil/Cuit:

| Razón Social:

| DNI:

| Teléfono/Celular:

| Mail:

| Dirección:

| Categoría[P particular, D distribuidor]:

3. Modificar

| Cuil/cuit del cliente a modificar:

0- Salir

1- Razón social:

2- Dni:

3- Cuil/cuit:

4- Teléfono:

5- Mail:

6- Dirección:

7- Categoría:

| Ingrese la opción:

4. Baja

| Cuil/cuit del cliente a dar de baja:

|Razón social:

|Cuil/Cuit:

|Teléfono:

|Mail:

|Dirección:

|Categoría:

| Ingrese S si desea ver la lista de ventas a el cliente, de lo contrario de ENTER

| Luego de ver el cliente seleccionado, Ingrese 'E' si desea eliminarlo, de lo contrario de ENTER y el mismo seguirá en los registros.

5. Buscar

| Cuil/cuit de cliente a buscar:

0- Salir

1- Ver datos personales

2- Ver compras del cliente

| Ingrese la opción:

1. Ver datos personales
Razón social: Cuil/Cuit: Teléfono: Mail: Dirección: Categoría: Ingrese S si desea ver la lista de ventas a el cliente, de lo contrario de ENTER

6. Listar
Razón social: Cuil/Cuit: Teléfono: Mail: Dirección: Categoría: Ingrese S si desea ver la lista de ventas a el cliente, de lo contrario de ENTER

7. Listar compras por cliente
Ingrese el cuil/cuit del cliente:

6. Verificación de stock
Cantidad de KG a producir:

7. Pedidos
1- Presupuestos 2- Nuevo 3- Modificación 4- Baja 5- Búsqueda 6- Listar y mostrar estados pendientes

1. Presupuestos
Kg a producir:
Dólar con el que quiere calcular:
Categoría [D por distribuidor, P por particular]:

Precio final en pesos: 0.0
Precio final en dólares: 0.0
Ingrese E para enviar el pedido, de lo contrario de ENTER:

2. Nuevo
Cuil/Cuit del vendedor:
Cuil/Cuit del Cliente:
Valor del dólar:
Cantidad de kg pedidos:
Fecha: 20/09/21[Si quiere modificar la fecha ingrese 'M', de lo contrario de ENTER]:
¿Desea agregar algún descuento? [S por si, ENTER por NO]

Costo en pesos: 0.0
Costo en Dólares: 0.0

3. Modificación de pedido
Id de Pedido:
Venta numero, de Cliente
0- Salir
1- Fecha:
2- Vendedor
3- Estado del envío:
4- Estado de pago
Ingrese la opción:

4. Baja
Id de Pedido/Venta: -----
Id:
Fecha:
Cliente:
Vendedor:
Kg vendidos:
Descuentos:
Precio final pesos :
Precio final dolares :
Estado del envío:
Estado del pago:
Ingrese E si desea proseguir con la eliminación, de lo contrario de ENTER:

5. Búsqueda
0- Salir
1- Por ID de pedido
2- Por fecha
3- Por cliente
Ingrese opcion:

1/2. Por ID de compra/Por fecha
Id de compra: /
Día [en formato dd]:
Mes [en formato mm]:
Año [en formato aaaa]: -----
Id:
Fecha:
Cliente:
Vendedor:
Kg vendidos:

|Descuentos:

|Precio final pesos :

|Precio final dolares :

|Estado del envío:

|Estado del pago:

1. Por Cliente

| Cuil/cuit del cliente:

|Razón social:

|Cuil/Cuit:

|Teléfono:

|Mail:

|Dirección:

|Categoría:

| Ingrese S si desea ver la lista de ventas a el cliente, de lo contrario de ENTER

6. Listar y mostrar estados pendientes

1- Ver pedidos con estados en 'pendiente'

2- Modificar los estados de los pedidos

| Ingrese opcion:

1. Ver pedidos con estados en 'pendiente'

|Id:

|Fecha:

|Cliente:

|Vendedor:

|Kg vendidos:

|Descuentos:

|Precio final pesos :

|Precio final dolares :

|Estado del envío:

|Estado del pago:

--

2. Modificar los estados de los pedidos
Número de pedido a modificar: -----
Id:
Fecha:
Cliente:
Vendedor:
Kg vendidos:
Descuentos:
Precio final pesos :
Precio final dolares :
Estado del envío:
Estado del pago:

¿Desea modificar el estado del ENVÍO? ['S' por si, ENTER por no]

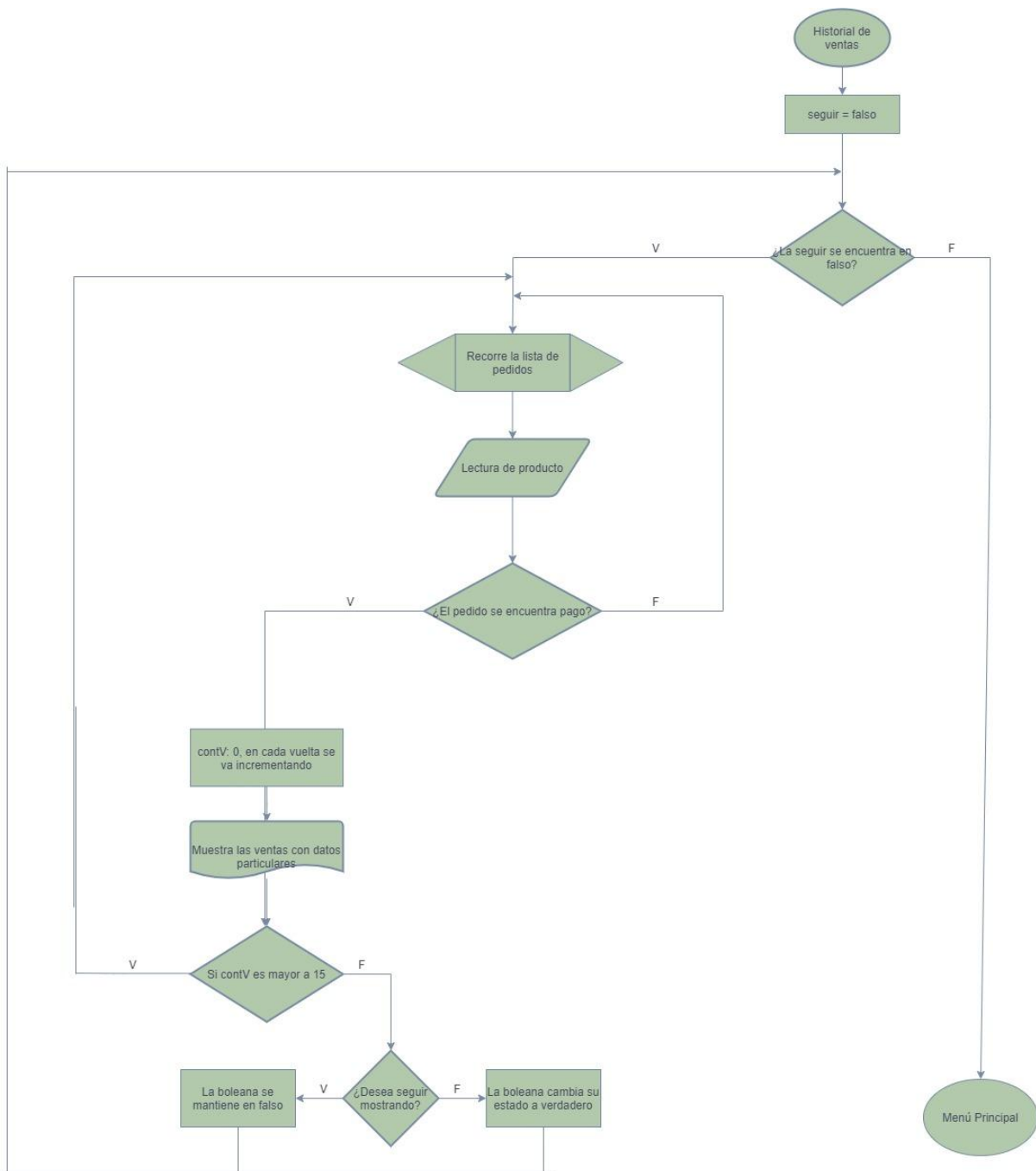
¿Desea modificar el estado del PAGO? ['S' por si, ENTER por no]

8. Ventas
1- General 0
2- Clientes Distribuidores
3- Clientes Particulares

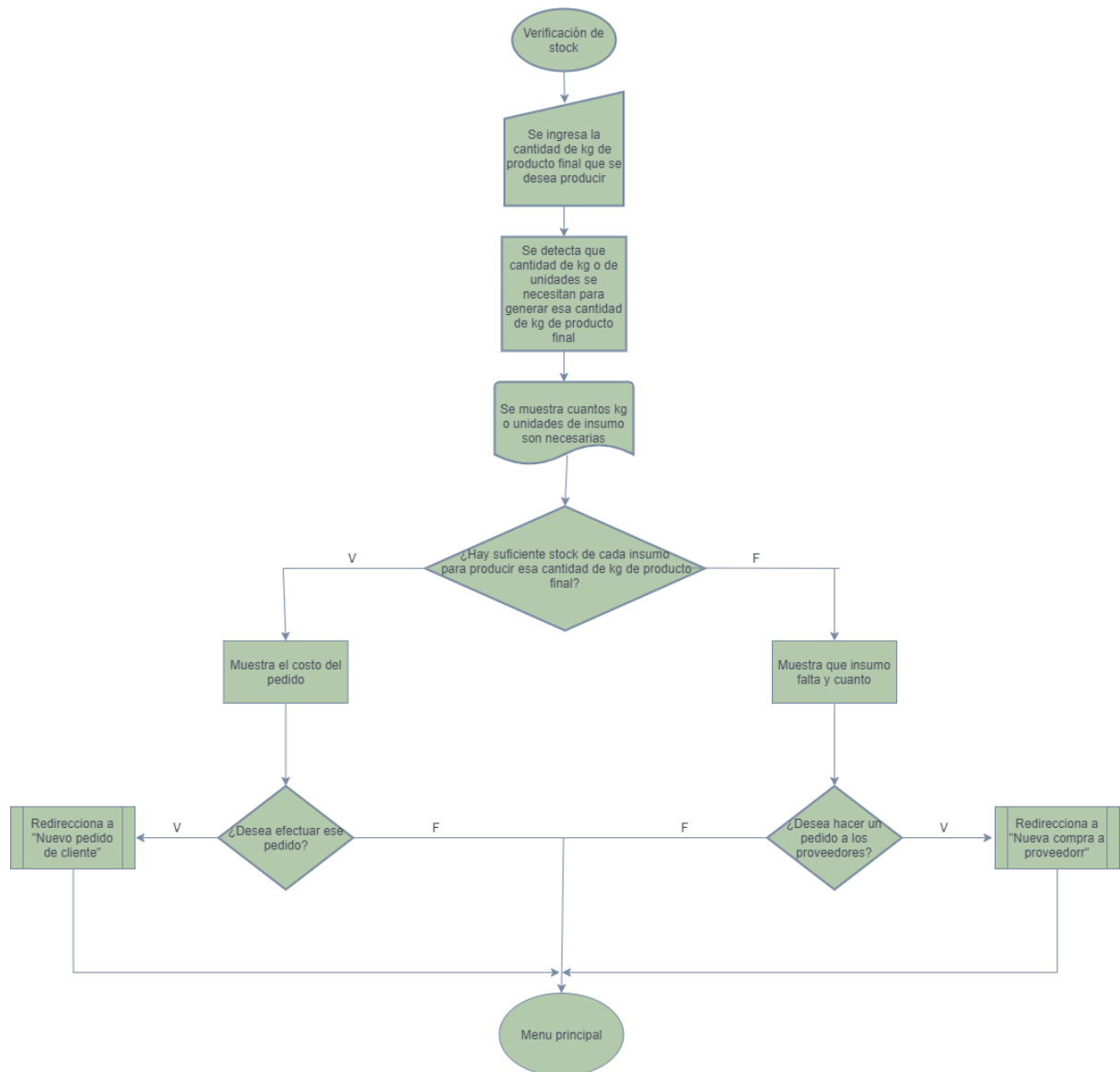
}

Procesos del programa:

Historial de ventas



Verificación de stock



Plan de pruebas:

Aspecto a evaluar	Prueba a realizar
Ingreso y creación de datos preexistentes	Evaluar a través del uso del sistema
Ingreso de nuevos datos	Evaluar a través del uso del sistema y el muestreo
Búsqueda y lectura de datos	Realizar una búsqueda de datos existentes y no existentes para corroborar su funcionamiento
Modificación de datos	Modificar datos y observar si los datos han cambiado a través de una búsqueda
Eliminación de datos	Eliminar datos y luego efectuar una búsqueda de los mismos
Listar datos	Realizar el procedimiento y observar si los mismos se listan correctamente
Listar datos parciales	Realizar el procedimiento y observar si se listan los datos determinados
Funcionamiento de los menús	Verificar el funcionamiento de todos los menues a través de su uso
Métodos de agregación	Listar los datos y observar si se encuentran dentro de la lista
Escritura y lectura de archivos	Verificar a través de la salida y entrada al programa, observar si los datos siguen manteniéndose.