

# **MEMORIA ESCRITA JAVA**

**MascotasUN**

**Presentado por:**

Juan Felipe López Ramírez  
Anna Ospina Bedoya  
Alejandra Uribe Sierra  
Hinara Pastora Sánchez Mata  
Juan José Tobón Zapata

**Grupo 2 - Equipo 1**  
**Práctica 1**

**Profesor:**

Jaime Alberto Guzmán Luna

Martes 25 de Octubre



**Universidad Nacional de Colombia**  
**Programación Orientada a Objetos**  
**Facultad de Minas**  
**2022-2**



## CONTENIDO

<b>I. DESCRIPCIÓN GENERAL DE LA APLICACIÓN</b>	<b>4</b>
Análisis, diseño e implementación	4
• Gestión de los diferentes usuarios	4
• Gestión de los productos de la tienda	4
• Acceso a la facturación	4
• Acceso al manejo del inventario	4
• Generar comisiones para los vendedores	4
• Estadísticas de ventas	4
• Recomendaciones personalizadas al cliente	4
Descripción de las capas del programa	4
1. Capa de persistencia	4
2. Capa lógica	5
3. Capa asociada a la interfaz de usuario (UI)	5
<b>II. DESCRIPCIÓN DEL DISEÑO ESTÁTICO DEL SISTEMA</b>	<b>6</b>
Diagrama de UML	6
Objetos del sistema	6
<b>III. DESCRIPCIÓN DE LA IMPLEMENTACIÓN DE CARACTERÍSTICAS DE PROGRAMACIÓN ORIENTADA A OBJETOS</b>	<b>8</b>
Clase abstracta	8
Método abstracto	9
Interfaz	9
Herencia	9
Ligadura dinámica	10
Atributo de clase	11
Método de clase	11
Constante	12
Encapsulamiento	12
Sobrecarga de constructores	12
Sobrecarga de métodos	13
Manejo de referencias this para desambiguar y this()	13
Enumeración	14
<b>IV. DESCRIPCIÓN DE LAS FUNCIONALIDADES</b>	<b>15</b>
Funcionalidad 1: Venta de productos	15
Funcionalidad 2: Compra de inventario	15
Funcionalidad 3: Mostrar recomendaciones personalizadas	16
Funcionalidad 4: Mostrar estadísticas de la tienda	17
Funcionalidad 5: Cobrar comisión por ventas	17
<b>V. MANUAL DE USUARIO</b>	<b>18</b>
1) Casos de Prueba	18
Vendedores:	18
Usuarios:	19
Tienda:	19



Proveedores:	19
2) Prueba de funcionalidades	20

## I. DESCRIPCIÓN GENERAL DE LA APLICACIÓN

### Análisis, diseño e implementación

MascotasUN es una aplicación pensada para la gestión de facturación e inventario de una tienda de productos para mascotas vista desde un administrador que es el vendedor y al obtener este producto de software tendrá acceso a una aplicación en la cual podrá llevar control de la tienda, en aspectos generales se tiene acceso a características como:

- **Gestión de los diferentes usuarios**

Permite crear, editar y eliminar los diferentes usuarios de tipo persona en los que se contempla vendedores, clientes y proveedores, accediendo a su menú gestión propio

- **Gestión de los productos de la tienda**

Permite crear, editar y eliminar los diferentes productos que se encuentran a disposición para la venta en nuestra tienda.

- **Acceso a la facturación**

Accede a un menú en el cual se puede agregar y eliminar los productos en la factura, mostrar los productos, realizar la venta y generar un envío.

- **Acceso al manejo del inventario**

Se puede acceder a los proveedores para gestionar la compra de inventario, además el programa puede generar informes del stock que está en riesgo.

- **Generar comisiones para los vendedores**

Se pueden pagar comisiones a los vendedores en base a las ventas realizadas.

- **Estadísticas de ventas**

Muestra diferentes estadísticas de las ventas como productos más vendidos, los usuarios con más compras y los vendedores con más ventas.

- **Recomendaciones personalizadas al cliente**

Se recomiendan productos al cliente en base a sus mascotas.

### Descripción de las capas del programa

#### 1. Capa de persistencia

En el paquete baseDatos se tiene:

- La clase serializador que crea los archivos para tener persistencia de los datos creados, que básicamente en cada clase de la capa lógica cada vez que instancia un objeto se añade a una lista llamada como el nombre de la clase en plural y así se tiene la lista completa de todas las instancias creadas de las clases, que se van a almacenar en el directorio temp donde hay un archivo txt con todas las instancias de cada clase de la capa lógica .
- La clase deserializador toma los archivos creados en la clase anterior y los settea al atributo con el nombre de la clase en plural donde se almacenan todas las instancias de las clases de la capa lógica, así logramos mantener la persistencia de nuestro programa



## 2. Capa lógica

En el paquete *gestionAplicacion* se tiene:

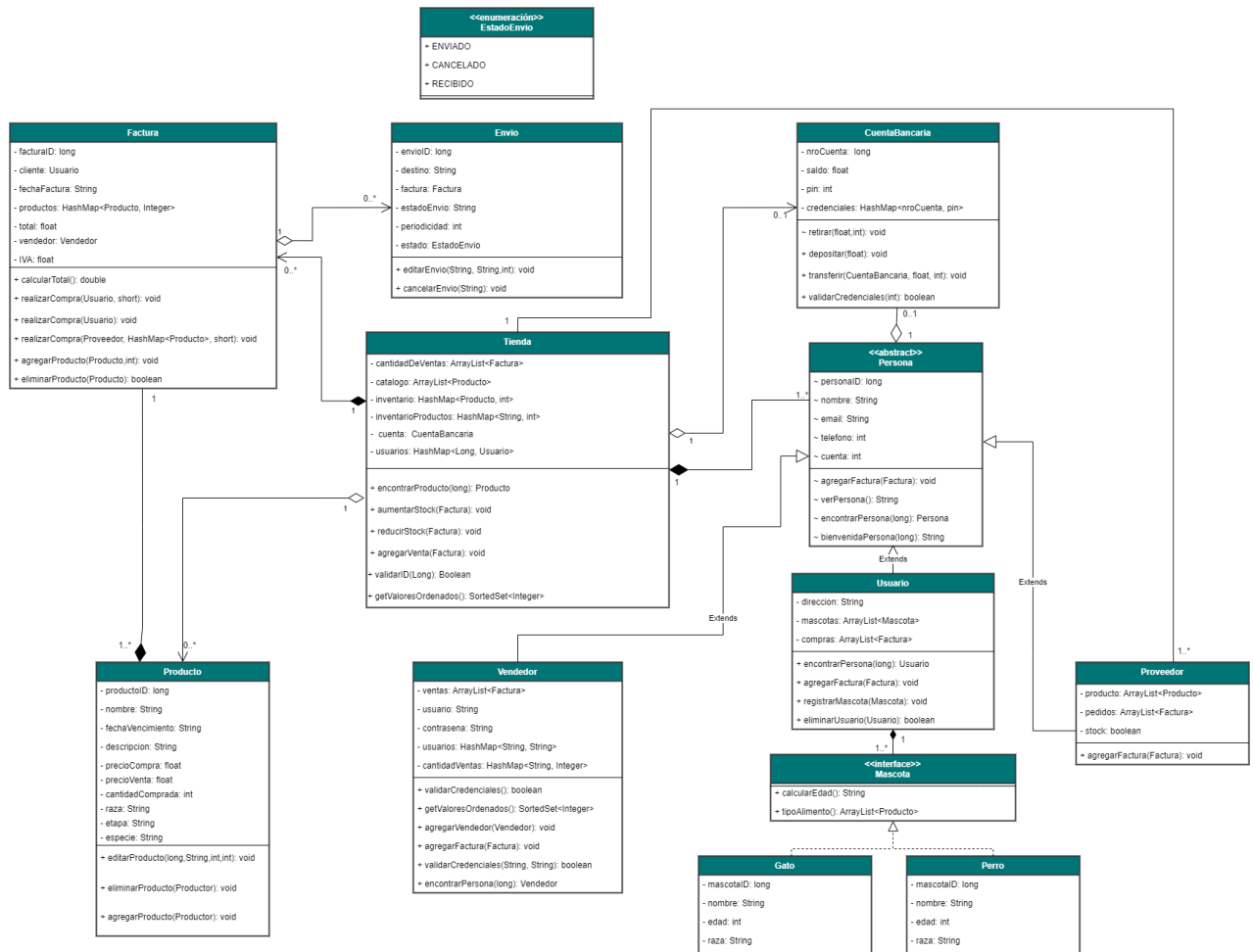
- El paquete *gestionPersonas* donde se encuentran las clases:
  - Vendedor
  - Usuario
  - Proveedor
  - La interfaz Mascota
  - Las clases implementadas de la interfaz mascota: Perro y Gato
  - La clase abstracta Persona
- El paquete *gestionPersonas* donde se encuentran las clases:
  - Factura
  - Producto
  - Envio
  - La clase Tienda que es de donde parte el programa

## 3. Capa asociada a la interfaz de usuario (UI)

En esta capa es donde se crean los menús y es donde se implementa toda la ejecución del programa. Se tiene el paquete *gestion* donde se maneja la gestión de las diferentes clases para poder agregar, modificar y eliminar instancias de las clases hijas de Persona y Producto; se tiene el paquete UI, que se encuentra la *clase Gestion*, y unifica todo lo anteriormente mencionado; luego se tiene el paquete funcionalidad donde están las 5 funcionalidades y una *clase Funcionalidad* donde se unifica el menú de las funcionalidades y para finalizar la *clase Principal* donde se hace el menú de inicio del programa en general.

## II. DESCRIPCIÓN DEL DISEÑO ESTÁTICO DEL SISTEMA

### Diagrama de UML



Para visualizarlo mejor entrar a este link:

<https://drive.google.com/file/d/1G69AKOee4nIR3klb9OshgPKL0oZXrm71/view?usp=sharing>

### Objetos del sistema

El sistema MascotasUN está compuesto por 11 clases en donde se incluye una clase abstracta y una interfaz, además posee dos paquetes en donde se divide la gestión de los aspectos relacionados con inventario, facturación, productos y ventas (paquete *gestionVentas*) y *gestionPersonas* en donde se centraliza el manejo de las clases que inciden dentro de la ejecución del programa. Estas son sus relaciones:

- En el paquete *gestionVentas* se encuentran las clases:

- Factura*

Se encarga de realizar las facturas de nuestra tienda, obteniendo información de distintas clases para su correcta ejecución. La clase factura tiene relaciones con las clases:

- Envio*: Relación de agregación en donde una factura puede agregar un envío, pero si no lo hace no se altera su funcionalidad.
    - Tienda*: Relación de composición con la clase tienda, en donde es necesaria para poder generar instancias de la clase factura.



- **Producto:** La clase Factura debe tener al menos un producto para ejecutarse (relación de composición).
- **Producto**  
Esta clase contiene las características de los productos que tenemos en nuestra tienda de mascotas. Tiene relación con:
  - **Tienda:** El producto puede estar o no instanciado dentro de la clase tienda ya que poseen una relación de agregación.
  - **Factura:** Productos debe estar al menos una o más veces instanciado para que se ejecute la clase tienda.
- **Envío**  
Se encarga de realizar los envíos de nuestra tienda, obteniendo información de distintas clases para su correcta ejecución. La clase envío tiene relaciones con las clases:
  - **Factura:** El envío debe de tener asociada una factura.
- **Tienda**  
La clase Tienda que es de donde parte el programa, se encarga de la gestión de todas las clases de la tienda y está relacionada con:
  - **Producto:** Una tienda puede agregar o no productos y esto no afecta su funcionamiento.
  - **Factura:** Una tienda puede tener o no facturas y tiene una relación de asociación.
  - **Proveedor:** La tienda puede tener uno o más proveedores.
  - **CuentaBancaria:** La tienda puede o no tener cuentas bancarias, tienen relación de agregación.
  - **Persona:** Una tienda está compuesta por personas y puede tener una o más de ellas.
- En el paquete *gestionPersonas* se encuentran las clases:
  - **Persona**  
La clase abstracta persona se encarga de brindar atributos y métodos a sus hijos y estos se utilizan para la gestión principal de nuestra tienda de mascotas.
    - **Vendedor:** Relación de herencia, Persona es padre de Vendedor.
    - **Usuario:** Relación de herencia, Persona es padre de Usuario.
    - **Proveedor:** Relación de herencia, Persona es padre de Proveedor.
    - **Tienda:** La clase Tienda debe tener al menos una persona para existir (relación de composición).
    - **CuentaBancaria:** Relación de agregación, una Persona puede o no tener una CuentaBancaria.
  - **Usuario**  
La clase Usuario es hija de la clase Persona, esta hace referencia a los compradores de nuestra tienda.
    - **Persona:** Relación de herencia, Usuario es hijo de Persona.
    - **Mascota:** Relación de agregación, un Usuario debe de tener una o más mascotas.
  - **Vendedor**  
El Vendedor es un hijo de la clase Persona y este da referencia a las características y funcionalidades que tiene un vendedor.
    - **Persona:** Relación de herencia, Vendedor es hijo de Persona.
  - **Proveedor**

El Proveedor es un hijo de la clase Persona y este da referencia a las características y funcionalidades que tiene un proveedor.

- *Persona*: Relación de herencia, Usuario es hijo de Persona.
- *Tienda*: Un proveedor solo hace parte de una tienda, pero la tienda tiene uno o más proveedores.

- *Perro*

Un Perro implementa la interfaz mascota y da atributos y métodos a una mascota de tipo perro

- *Mascota*: Relación de herencia, Perro es hijo de Mascota.

- *Gato*

Un Gato implementa la interfaz mascota y da atributos y métodos a una mascota de tipo gato.

- *Mascota*: Relación de herencia, Gato es hijo de Mascota.

- *Mascota*

Es una interfaz que da bases para métodos que se aplicarán en los tipos de mascotas perros y gatos

- *Usuario*: Relación de composición, un Usuario debe de tener una o más mascotas.

- *Cuenta bancaria*

La clase cuenta bancaria nos permite dar atributos y métodos que serán implementados en las clases tienda, persona y sus hijos. Esta es su relación:

- *Persona*: Relación de agregación, una persona puede o no tener una cuenta bancaria.
- *Tienda*: Relación de agregación, una tienda puede o no tener una cuenta bancaria.

### III. DESCRIPCIÓN DE LA IMPLEMENTACIÓN DE CARACTERÍSTICAS DE PROGRAMACIÓN ORIENTADA A OBJETOS

#### Clase abstracta

Nuestra clase abstracta se encuentra en el paquete *gestionPersonas* es la clase *Persona* la cual implementa el tipo de dato Persona que hereda a las clases *Vendedor*, *Proveedor* y *Usuario*, así generalizamos las personas y tenemos atributos generales para nuestras personas y darle sentido a *gestionPersona*. Esta es la clase abstracta escogida:

```
9  9 usages 3 inheritors ▲ Alejandra Uribe +4
10 public abstract class Persona {
11
12     //Atributos (protected para utilizarlos dentro del paquete)
13     4 usages
14     protected static ArrayList<Persona> personas = new ArrayList<Persona>();
15     6 usages
16     protected long personaId;
17     6 usages
18     protected String nombre;
19     6 usages
20     protected String email;
21     6 usages
22     protected int telefono;
23     protected CuentaBancaria cuenta;
24
25     //Constructor
26     1 usage ▲ Anna Ospina Bedoya
27     public Persona(long personaId, String nombre, String email, int telefono){
28         this.personaId=personaId;
29         this.nombre=nombre;
```



## Método abstracto

En nuestra clase abstracta se crea el método abstracto agregar factura, el cual se implementa en nuestras clases hijas para permitir que se añadan en: usuario agregar las facturas de las compras, en vendedor agregar las facturas de venta, y en proveedor agregar las facturas de venta de inventarios.

```
33      //Métodos
34
35      /* Método abstracto que se hereda a las clases Usuario, Proveedor y Vendedor,
36      * que recibe como parámetro un objeto tipo Factura y retorna un void */
37      11 usages 3 implementations Alejandra Uribe
      protected abstract void agregarFactura(Factura f); // Método abstracto
```

## Interfaz

Se implementa dentro del paquete gestionPersonas la interfaz Mascota que es implementada en las las clases Gato y Perro para unificarse en el tipo de dato mascota.

```
9  public interface Mascota {
10      // Métodos
11
12      /* Este metodo dependiendo de la clase que lo implemente retornara una lista de los productos recomendados*/
13      2 usages 2 implementations jtobonz
      ArrayList<Producto> tipoAlimento();
14
15      /* Dependiendo de la clase que lo implemente retornara en que tipo de etapa de desarrollo se encuentra la masc
16      2 usages 2 implementations jtobonz
      String calcularEdad();
17  }
```

## Herencia

De la clase abstracta Persona descrita anteriormente se hereda a las clases Usuario, Vendedor y Proveedor con la intención introducir el tipo de dato persona implicado en estas 3 clases y hacer atributos y métodos comunes.

- Herencia en proveedor

```
12  public class Proveedor extends Persona implements Serializable {
13
14      //Serializador Proveedores
15      private static final long serialVersionUID = 1L;
16      4 usages
17      private static ArrayList<Proveedor> proveedores = new ArrayList<>();
18
19      //Atributos
20      4 usages
21      private ArrayList<Producto> producto = new ArrayList<>();
22      3 usages
23      private ArrayList<Factura> pedidos = new ArrayList<>();
24      4 usages
25      private boolean stock; //Pueden haber proveedores con y si stock
```

- Herencia en usuario

```

13 public class Usuario extends Persona implements Serializable {
14
15     //Serializador Usuarios
16     private static final long serialVersionUID = 1L;
17     5 usages
18     private static ArrayList<Usuario> usuarios = new ArrayList<>();
19
20     // Atributos
21     4 usages
22     private String direccion;
23     5 usages
24     private ArrayList<Mascota> mascotas = new ArrayList<>();
25     4 usages
26     private ArrayList<Factura> compras = new ArrayList<>();

```

- Herencia en vendedor

```

18 public class Vendedor extends Persona implements Serializable {
19
20     //Serializador vendedor
21     private static final long serialVersionUID = 1L;
22     1 usage
23     private static final double comision = 0.05;
24     5 usages
25     private static ArrayList<Vendedor> vendedores = new ArrayList<>();
26
27     // Atributos
28     5 usages
29     private ArrayList<Factura> ventas=new ArrayList<>();

```

## Ligadura dinámica

1. La primera ligadura dinámica se realizó en la Clase persona dentro del método verPersona() que recorre la lista de personas y las imprime retornando un string, se resuelve utilizando el método toString() más específico entre el Usuario, Proveedor o Vendedor en tiempo de ejecución.

```

45 /*Método que recorre la lista de personas y las imprime, retornando un String.
46 * Se resuelve el metodo toString() más específico de Usuario, Proveedor o Vendedor
47 * por medio de la ligadura dinámica en tiempo de ejecución*/
48 Alejandro Uribe
49 public static String verPersona() {
50     String resultado_nombre = "";
51
52     for (int i=0; i < personas.size(); i++) {
53         resultado_nombre += (i+1) + ". " + personas.get(i) + "\n";
54     }
55     return resultado_nombre;

```

2. La segunda ligadura dinámica se realizó en la Clase persona dentro del método bienvenidaPersona(long ID) que recibe como parámetro un id y devuelve un String con el respectivo saludo, esto se resuelve con ligadura dinámica dentro del método encontrarPersona(long ID) y busca el más específico entre el Usuario, Proveedor o Vendedor en tiempo de ejecución.

```
public String bienvenidaPersona(long ID) {  
    String bienvenida = "Bienvenido, " + encontrarPersona(ID).getNombre();  
  
    return bienvenida;  
}  
  
public static Persona encontrarPersona(long personaId){  
    for(Persona p: personas){  
        if(p.getPersonaId()== personaId){  
            return p;  
        }  
    }  
    return null;  
}
```

### Atributo de clase

En la clase tienda se tiene el atributo de clase inventario de tipo hashmap que nos permite almacenar en la clave de este el producto y en el valor del mismo la cantidad de este producto disponible.

```
23 // Atributos  
24 private static ArrayList<Factura> cantidadDeVentas = new ArrayList<>();  
25 private static ArrayList<Producto> catalogo = new ArrayList<>();  
26 private static HashMap<Producto, Integer> inventario = new HashMap<>();  
27 private static HashMap<String, Integer> inventarioProductos = new HashMap<>();  
28 private static CuentaBancaria cuenta;  
29 private static HashMap<Long, Usuario> usuarios = new HashMap<>();
```

### Método de clase

En la clase tienda se tiene el método agregar stock, que recibe como parámetro la factura de compra de inventario que se realizó al proveedor y lo que hace, es actualizar el atributo de clase inventario en base a la compra de inventario realizada.

```
// Método que aumenta la cantidad de productos de un producto de la tienda  
1 usage  Felipe López +1  
public static void aumentarStock(Factura factura){  
    for (Map.Entry<Producto, Integer> fact : factura.getProductos().entrySet()) {  
        Producto k = fact.getKey();  
        Integer v = fact.getValue();  
        Integer ch = inventario.get(k);  
        inventario.put(k, ch + v);  
    }  
}
```

## Constante

En la clase factura se tiene un atributo constante de tipo float llamado IVA, que nos permite tener el valor del IVA para poder agregarlo a la hora de realizar las ventas.

```
24     private static float total = 0.0f;
      3 usages
25     private Vendedor vendedor;
      3 usages
26     private final float IVA = 1.16f;
```

## Encapsulamiento

Los atributos de la mayoría de las clases son privados, estos se acceden mediante los getters y setters los cuales son públicos, para la herencia se utiliza el protected en la clase Persona y el default se hace uso en el método retirar que se encuentra en la clase CuentaBancaria.

```
11     public class CuentaBancaria implements Serializable{
12
13         //Atributo para el serializador
14         private static final long serialVersionUID = 1L;
      4 usages
15         private static ArrayList<CuentaBancaria> cuentasBancarias = new ArrayList<>();
16
17         //Atributos
18         private long nroCuenta;
      4 usages
19         private float saldo;
      3 usages
20         private int pin;
      3 usages
21         static private HashMap<Long,Integer> credenciales = new HashMap<>();
22
```

## Sobrecarga de constructores

En la clase Usuario hay un constructor con sus parámetros completos y uno en el que no se encuentra el parámetro cuentaBancaria para así tener la opción de crear nuevos Usuarios con y sin cuenta bancaria

```
25     public Usuario(long personaId, String nombre, String email, int telefono, CuentaBancaria cuenta, String direcc
26         super(personaId, nombre, email, telefono, cuenta);
27         this.direccion = direccion;
28         this.mascotas = mascotas;
29         usuarios.add(this);
30         Tienda.setUsuarios(personaId, usuario: this);
31     }
32     // Sobrecarga de constructor, cuando el usuario no posee cuenta bancaria
      Anna Ospina Bedoya +1
33     public Usuario(long personaId, String nombre, String email, int telefono, String direccion, ArrayList<Mascota>
34         this(personaId, nombre, email, telefono, cuenta: null, direccion, mascotas);
35         usuarios.add(this);
36     }
```

## Sobrecarga de métodos

En la clase Factura se encuentra dicha sobrecarga con el método realizarCompra, este se sobrecarga 3 veces ya que hay uno para realizar la compra de la tienda al proveedor, otro para realizar una compra cuando el usuario utiliza como método de pago con tarjeta y otro para cuando el usuario utiliza tarjeta como método de pago.

```

69 @ public void realizarCompra(Usuario cliente, short pswd){
70     //Primero se calcula el total de la compra con este for
71     Tienda.reducirStock( factura: this);
72     Tienda.getCuenta().transferir(cliente.getCuenta(),total,pswd);
73     cliente.agregarFactura( f: this);
74     Tienda.agregarVenta( f: this);
75 }
76
77 //Metodo sobrecargado que realiza la compra, cuando el usuario utiliza como método de pago efectivo
78 1 usage Felipe López +1
79 public void realizarCompra(Usuario cliente){
80     //Primero se calcula el total de la compra con este for
81     for (Map.Entry<Producto, Integer> entry : productos.entrySet()) {
82         Producto k = entry.getKey();
83         Integer v = entry.getValue();
84         total += k.getPrecioVenta()*v;
85     }
86     // se reduce el stock de la tienda, se resta el saldo al cliente y se le agrega a la tienda
87     total = total*IVA;
88     Tienda.reducirStock( factura: this);
89     Tienda.getCuenta().depositar(total);
90     cliente.agregarFactura( f: this);
91     Tienda.agregarVenta( f: this);
92 }

```

## Manejo de referencias this para desambiguar y this()

- **this():** Este es usado en la sobrecarga de constructores en las siguientes clases:
1. Usuario, esto debido a que al crear un usuario este puede o no tener cuenta bancaria

```

25 public Usuario(long personaId, String nombre, String email, int telefono, CuentaBancaria cuenta, String direcc
26     super(personaId, nombre, email, telefono, cuenta);
27     this.direccion = direccion;
28     this.mascotas = mascotas;
29     usuarios.add(this);
30     Tienda.setUsuarios(personaId, usuario: this);
31 }
32 // Sobrecarga de constructor, cuando el usuario no posee cuenta bancaria
33 1 Anna Ospina Bedoya +1
34 public Usuario(long personaId, String nombre, String email, int telefono, String direccion, ArrayList<Mascota>
35     this(personaId, nombre, email, telefono, cuenta: null, direccion, mascotas);
36     usuarios.add(this);
37 }

```

2. CuentaBancaria, ya que es uno para el vendedor que no necesita pin y otro para el usuario que sí lo necesita

```
public CuentaBancaria(long nroCuenta, float saldo, int pin) {
    this(nroCuenta,saldo);
    this.pin = pin;
    credenciales.put(nroCuenta, pin);
    cuentasBancarias.add(this);
}

//Sobrecarga de constructores que se utiliza para la clase vendedores
Anna Ospina Bedoya +1
public CuentaBancaria(long nroCuenta, float saldo){
    this.nroCuenta= nroCuenta;
    this.saldo=saldo;
    cuentasBancarias.add(this);
}
```

- **this**

1. El this para desambiguar se utiliza para el constructor de la clase Persona para acceder directamente al atributo del objeto.

```
24 //Constructor
1 usage Anna Ospina Bedoya
25 public Persona(long personaId, String nombre, String email, int telefono){
26     this.personaId=personaId;
27     this.nombre=nombre;
28     this.email=email;
29     this.telefono=telefono;
30 }
```

2. Se utiliza para el constructor de la clase Cuenta Bancaria para acceder directamente al atributo del objeto.

```
public CuentaBancaria(long nroCuenta, float saldo, int pin) {
    this(nroCuenta,saldo);
    this.pin = pin;
    credenciales.put(nroCuenta, pin);
    cuentasBancarias.add(this);
}

//Sobrecarga de constructores que se utiliza para la clase vendedores
Anna Ospina Bedoya +1
public CuentaBancaria(long nroCuenta, float saldo){
    this.nroCuenta= nroCuenta;
    this.saldo=saldo;
    cuentasBancarias.add(this);
}
```

## Enumeración

En la clase envío se implementa una enumeración llamada EstadoEnvío la cual contiene el estado del envío, este se hace uso en la funcionalidad RealizarCompra, al realizar una compra y requerir un envío el estado del envío pasa a ser ENVIADO.

```
20 public enum EstadoEnvío{
21     ENVIADO, CANCELADO, RECIBIDO;
22 }
```

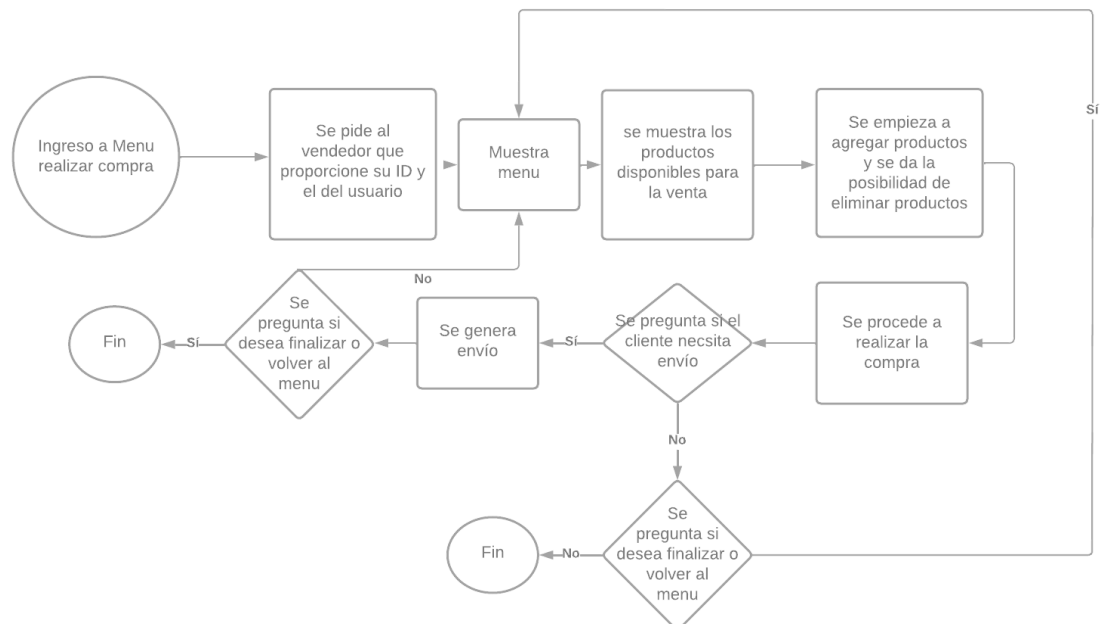
#### IV. DESCRIPCIÓN DE LAS FUNCIONALIDADES

##### Funcionalidad 1: Venta de productos

Esta funcionalidad está pensada para permitir la gestión de facturación, por tanto cuenta con los métodos: agregar producto, eliminar producto, realizar venta, realizar envío y permite al vendedor mostrar los productos disponibles para la venta.

Dentro de la funcionalidad intervienen las clases: Vendedor para asociar la factura a este, usuario para asociar la factura a este y si es requerido al realizar la compra descontar el saldo de la cuenta bancaria, la clase factura donde se realiza todo el proceso de la venta descrito anteriormente, la clase envío para si el usuario lo solicita se genere el envío y finalmente la clase tienda para obtener de sí el inventario de los productos y actualizar este cuando la compra se realice, además se añade la factura a las ventas.

##### Secuencia de la funcionalidad



##### Ejecución funcionalidad



```
Indique su eleccion : 2
Que haremos hoy?
1. Realizar una venta
2. Comprar inventario
3. Cobrar una comision
4. Mostrar recomendaciones personalizadas con base a mascota
5. Mostrar estadísticas de ventas
6. Regresar
```

```
Indique su eleccion : 1
Para iniciar la venta por favor indique el ID a continuacion:
46785214
Hola Juan Pablo Arcila puede proceder con la compra
```

```
Para crear la factura ingrese el id del cliente
1093186721
Se va a crear una factura a nombre de: Manuela Arboleda
Ingresa la fecha en formato: dd/mm/yyyy
```

```
11/07/2003
Realizar una compra
1. Mostrar productos disponibles
2. Agregar productos a la compra
3. Regresar
```

```
Indique su eleccion : 1
```

```
~ Cuido Adulto Extratiene 4 productos disponibles y tiene un costo de 15000.0, ID: 1003
~ Cuido Cachorro Smalltiene 2 productos disponibles y tiene un costo de 16000.0, ID: 1012
~ Whiskas para felino cachorrotiene 1 productos disponibles y tiene un costo de 15000.0, ID: 1015
~ Comida humeda Adulto Maxtiene 3 productos disponibles y tiene un costo de 5000.0, ID: 1002
~ Croquetas Adulto Minitiene 2 productos disponibles y tiene un costo de 13000.0, ID: 1007
~ Cuido Adulto Smalltiene 4 productos disponibles y tiene un costo de 16000.0, ID: 1009
~ Comida enlatada felino adultotiene 3 productos disponibles y tiene un costo de 9000.0, ID: 1017
~ Croquetas Cachorro Minitiene 5 productos disponibles y tiene un costo de 13000.0, ID: 1010
~ Comida enlatada felino cachorrotiene 2 productos disponibles y tiene un costo de 9000.0, ID: 1014
~ Comida humeda Cachorro Juniotiene 3 productos disponibles y tiene un costo de 6000.0, ID: 1011
~ Croquetas Adulto Plustiene 10 productos disponibles y tiene un costo de 12000.0, ID: 1001
~ Comida humeda Cachorro Maxtiene 2 productos disponibles y tiene un costo de 5000.0, ID: 1005
~ Purina catchow adultotiene 2 productos disponibles y tiene un costo de 10000.0, ID: 1016
~ Comida humeda Adulto Juniotiene 10 productos disponibles y tiene un costo de 6000.0, ID: 1008
~ Purina catchow cachorrotiene 7 productos disponibles y tiene un costo de 10000.0, ID: 1013
~ Whiskas para felino adultotiene 4 productos disponibles y tiene un costo de 15000.0, ID: 1018
~ Croquetas Cachorro Plustiene 4 productos disponibles y tiene un costo de 12000.0, ID: 1004
~ Cuido Cachorro Extratiene 3 productos disponibles y tiene un costo de 15000.0, ID: 1006
```

```
Su accion ha sido finalizada presione 1 para volver al menu
```

```
1
```

```
Realizar una compra
1. Mostrar productos disponibles
2. Agregar productos a la compra
3. Regresar
```

```
Indique su eleccion : 2
```



```
Agregar a la compra
1. Agregar un nuevo producto
2. Eliminar un producto
3. Mostrar resumen de factura
4. Finalizar compra
5. Configurar envio de compra
6. volver al menu compra
Indique su eleccion : 1
Ingrese el ID del producto que desea agregar:
1010
Ingrese la cantidad del producto que desea agregar
0
El producto se ha agregado con exito a la compra
Desea agregar mas productos?
1 para si, 0 en caso contrario
1
Ingrese el ID del producto que desea agregar:
1012
Ingrese la cantidad del producto que desea agregar
7
Lo sentimos no hay stock suficiente para esta venta
Desea cambiar la cantidad? 1 para si 0 para no
1
1
El producto se ha agregado con exito a la compra

1 para si, 0 en caso contrario
0
Su accion ha sido finalizada presione 1 para volver al menu
1
Agregar a la compra
1. Agregar un nuevo producto
2. Eliminar un producto
3. Mostrar resumen de factura
4. Finalizar compra
5. Configurar envio de compra
6. volver al menu compra
Indique su eleccion : 3
Resumen de la factura:
~ Factura #1
~ Fecha = 11/07/2003
~ Cliente = Manuela Arboleda
~ Cantidad a pagar = 93960.0

1) Cuido Cachorro Small 1 unidades, y tiene un precio unitario de: 16000.0
2) Croquetas Cachorro Mini 5 unidades, y tiene un precio unitario de: 13000.0
Su accion ha sido finalizada presione 1 para volver al menu
1
Agregar a la compra
1. Agregar un nuevo producto
2. Eliminar un producto
```

```
Agregar a la compra
1. Agregar un nuevo producto
2. Eliminar un producto
3. Mostrar resumen de factura
4. Finalizar compra
5. Configurar envio de compra
6. volver al menu compra
Indique su eleccion : 4
Est a punto de finalizar la compra, est seguro que desea terminar la venta?
ingrese 1 para finalizar la compra y cero para continuar con la compra
|
Resumen de compra
~ Factura #1
~ Fecha = 11/07/2003
~ Cliente = Manuela Arboleda
~ Cantidad a pagar = 202953.6

El usuario desea pagar en efectivo?
ingrese 1 si el pago es en efectivo y 0 si es con tarjeta
|
el total de la compra fue: 329386.16
Su accion ha sido finalizada presione 1 para volver al menu
|
Agregar a la compra
1. Agregar un nuevo producto
```

```
3. Mostrar resumen de factura
4. Finalizar compra
5. Configurar envio de compra
6. volver al menu compra
Indique su eleccion : 5
La compra con id 1 fue generada

Desea que se realice un envio de esta?

Digite 1 si el cliente lo requiere o 0 si no
1
Ingrese el destino de su compra:
oli
Su envio fue creado exitosamente
Gracias por preferirnos
Su accion ha sido finalizada presione 1 para volver al menu
6
Agregar a la compra
1. Agregar un nuevo producto
2. Eliminar un producto
3. Mostrar resumen de factura
4. Finalizar compra
5. Configurar envio de compra
6. volver al menu compra
Indique su eleccion : 6
Por favor seleccione una respuesta correcta
```

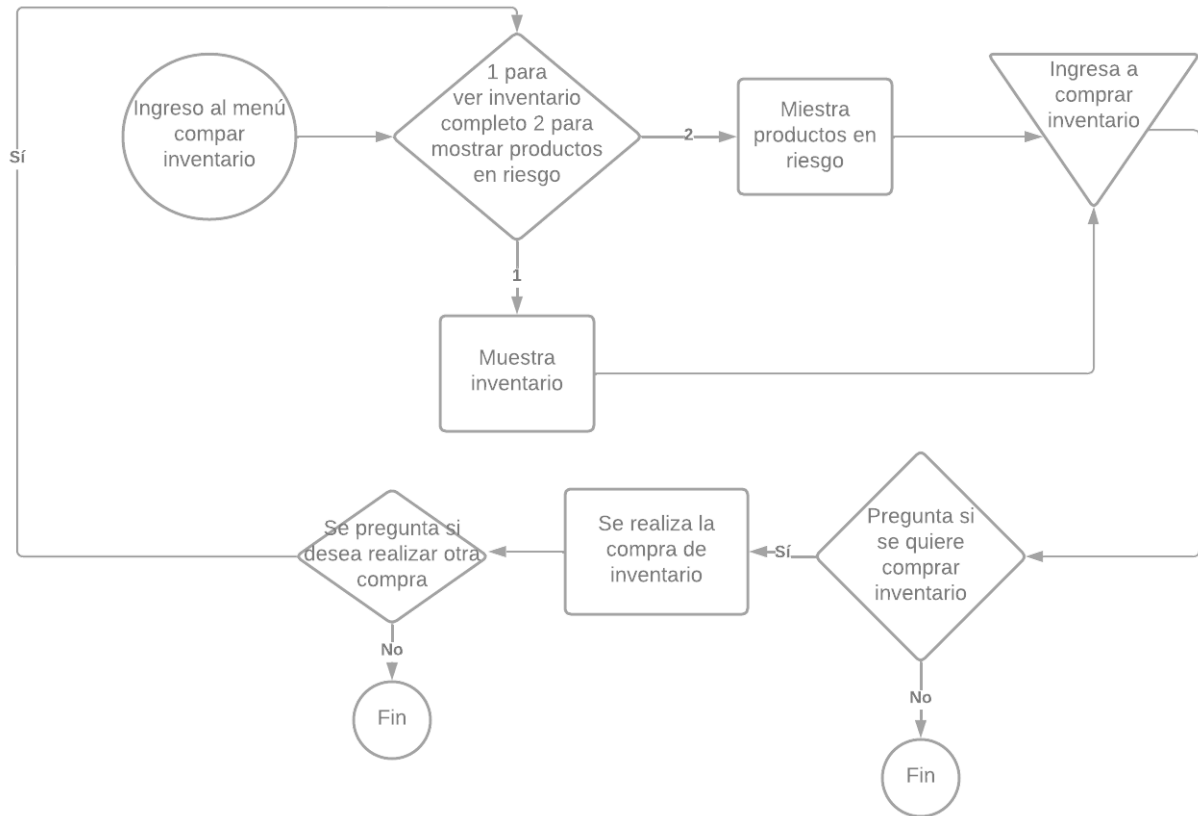
```
Indique su eleccion : 6
Por favor seleccione una respuesta correcta
Realizar una compra
1. Mostrar productos disponibles
2. Agregar productos a la compra
3. Regresar
Indique su eleccion : |
```

## Funcionalidad 2: Compra de inventario

Esta funcionalidad consiste en que el vendedor puede realizar una compra de inventario, esta se compone de los métodos compraProducto que pide el ID del producto la cantidad y realiza la compra,,

compralInventario el cual muestra una lista de proveedores, le pide al vendedor el proveedor al cual se desea comprar y el producto, imprimirOrdenado muestra los productos ordenados, imprimirCatalogo muestra el inventario completo e imprimirRiesgo muestra los productos a punto de acabarse. Se relaciona con las clases Producto, Proveedor y Tienda.

### Secuencia de la funcionalidad





¿Quieres ver el inventario completo o los productos en riesgo de agotarse?

1. Ver inventario completo
2. Ver productos en riesgo de agotarse

Indique su eleccion : 1

Este es su inventario actual

Producto: Whiskas para felino cachorro. Cantidad: 1  
Producto: Comida humeda Cachorro Max. Cantidad: 2  
Producto: Purina catchow adulto. Cantidad: 2  
Producto: Cuido Cachorro Small. Cantidad: 2  
Producto: Croquetas Adulto Mini. Cantidad: 2  
Producto: Comida enlatada felino cachorro. Cantidad: 2  
Producto: Cuido Cachorro Extra. Cantidad: 3  
Producto: Comida enlatada felino adulto. Cantidad: 3  
Producto: Comida humeda Adulto Max. Cantidad: 3  
Producto: Comida humeda Cachorro Junior. Cantidad: 3  
Producto: Cuido Adulto Extra. Cantidad: 4  
Producto: Cuido Adulto Small. Cantidad: 4  
Producto: Croquetas Cachorro Plus. Cantidad: 4  
Producto: Whiskas para felino adulto. Cantidad: 4  
Producto: Croquetas Cachorro Mini. Cantidad: 5  
Producto: Purina catchow cachorro. Cantidad: 7  
Producto: Croquetas Adulto Plus. Cantidad: 10  
Producto: Comida humeda Adulto Junior. Cantidad: 10

¿Desea comprar inventario?

1. Sí
2. No
3. Volver

```
Indique su eleccion : 1
¿A cuál proveedor desea comprar?
ID: 43825896. Nombre: Pedro Casas
ID: 43825897. Nombre: Armando Marin
ID: 58415. Nombre: Sol lunar
Indique el ID del proveedor a comprar: 58415
¿Qué producto desea comprar?
ID: 1001. Producto: Croquetas Adulto Plus
ID: 1002. Producto: Comida humeda Adulto Max
ID: 1003. Producto: Cuido Adulto Extra
ID: 1004. Producto: Croquetas Cachorro Plus
ID: 1006. Producto: Cuido Cachorro Extra
ID: 1007. Producto: Croquetas Adulto Mini
ID: 1008. Producto: Comida humeda Adulto Junior
ID: 1009. Producto: Cuido Adulto Small
ID: 1010. Producto: Croquetas Cachorro Mini
ID: 1012. Producto: Cuido Cachorro Small
ID: 1013. Producto: Purina catchow cachorro
ID: 1014. Producto: Comida enlatada felino cachorro
ID: 1015. Producto: Whiskas para felino cachorro
ID: 1016. Producto: Purina catchow adulto
ID: 1018. Producto: Whiskas para felino adulto
Indique el ID del producto a comprar: 1001
¿Cuántos desea comprar?
Indique su eleccion : 2
¿Desea comprar más productos?
Indique su eleccion :
  1. Sí
  2. No
Indique su eleccion :
2
;Su compra ha sido realizada con éxito!
Saldo actual Tienda: 1.0877857E7
```

```
Indique su eleccion :
2
;Su compra ha sido realizada con éxito!
Saldo actual Tienda: 1.0877857E7

Saldo actual Proveedor: 11600.0
```

**Eligiendo la 2 opcion, para mostrar productos críticos**



```
¿Quieres ver el inventario completo o los productos en riesgo de agotarse?
    1. Ver inventario completo
    2. Ver productos en riesgo de agotarse
Indique su eleccion : 2
Estos son los productos en riesgo de agotarse
Producto: Whiskas para felino cachorro. Cantidad: 1
Producto: Comida humeda Cachorro Max. Cantidad: 2
Producto: Purina catchow adulto. Cantidad: 2
Producto: Cuido Cachorro Small. Cantidad: 2
Producto: Croquetas Adulto Mini. Cantidad: 2
Producto: Comida enlatada felino cachorro. Cantidad: 2
¿Desea comprar inventario?
    1. Sí
    2. No
    3. Volver
Indique su eleccion : 1
¿A cuál proveedor desea comprar?
ID: 43825896. Nombre: Pedro Casas
ID: 43825897. Nombre: Armando Marin
ID: 58415. Nombre: Sol lunar
Indique el ID del proveedor a comprar: 43825896
¿Qué producto desea comprar?
ID: 1001. Producto: Croquetas Adulto Plus
ID: 1002. Producto: Comida humeda Adulto Max
ID: 1003. Producto: Cuido Adulto Extra
ID: 1004. Producto: Croquetas Cachorro Plus
ID: 1006. Producto: Cuido Cachorro Extra
ID: 1007. Producto: Croquetas Adulto Mini
ID: 1008. Producto: Comida humeda Adulto Junior
ID: 1009. Producto: Cuido Adulto Small
ID: 1010. Producto: Croquetas Cachorro Mini
ID: 1012. Producto: Cuido Cachorro Small
```

**Comprando dos productos**

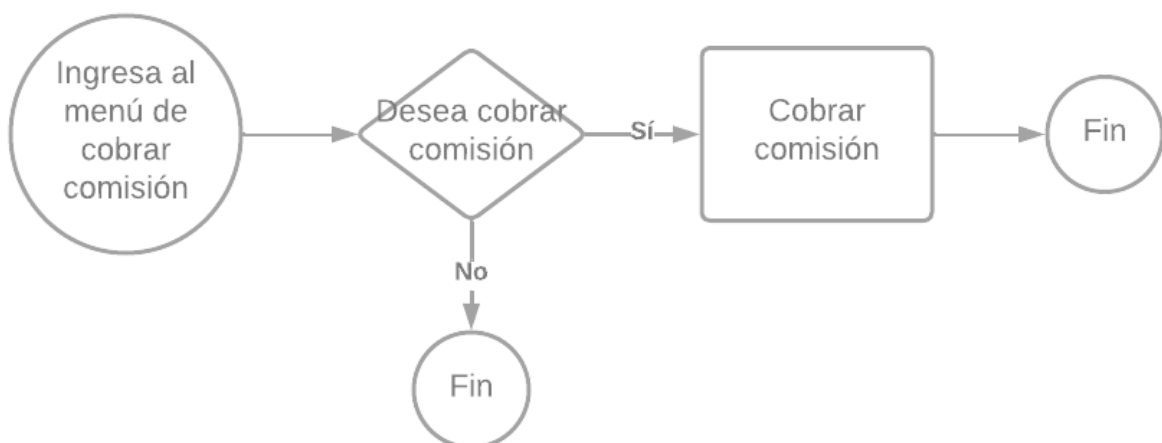
```
Indique el ID del producto a comprar: 1018
¿Cuántos desea comprar?
Indique su eleccion : 2
¿Desea comprar más productos?
Indique su eleccion :
1. Sí
2. No
Indique su eleccion :
1
Indique el ID del producto a comprar: 1001
¿Cuántos desea comprar?
Indique su eleccion : 3
¿Desea comprar más productos?
Indique su eleccion :
1. Sí
2. No
Indique su eleccion :
2
¡Su compra ha sido realizada con éxito!
Saldo actual Tienda: 1.0834937E7

Saldo actual Proveedor: 2.0901888E7
```

### Funcionalidad 3: Cobrar comisión por ventas

Esta funcionalidad consiste en que el vendedor puede cobrar su comisión obtenida por las ventas que realizó, en esta a partir de las ventas realizadas se calcula las ganancias totales obtenidas por las mismas, a este se le calcula un 5% que serán las comisiones del vendedor. Esta funcionalidad utiliza el método `calcularTotalVentas` el cual hace la suma de todas las ventas. Se relaciona con las clases `Vendedor`, `Factura` y `CuentaBancaria`.

#### Secuencia de la funcionalidad



```
Indique su eleccion : 3
¿Desea cobrar su comisión por las ventas realizadas?

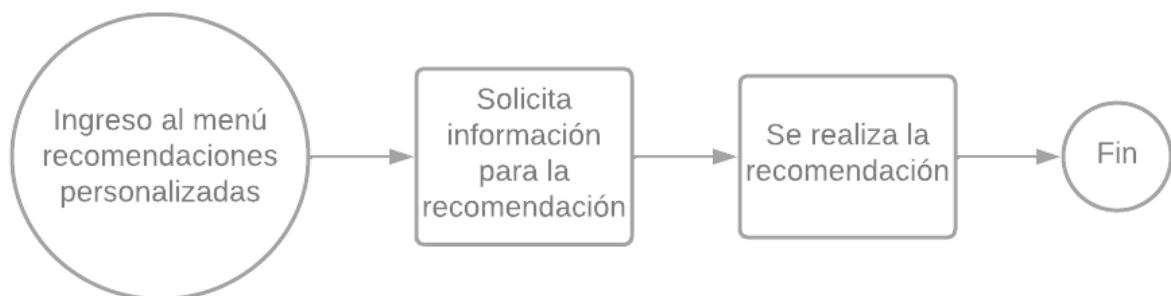
    1. Si
    2. No, Salir

Indique su eleccion : 1
Ingrese su cedula:
13205481
Tus ventas totales fueron: 367029.5625
Por lo que tus comisiones son: 18351.478125
Tu dinero ha sido transferido exitosamente
Vuelva pronto!
```

#### Funcionalidad 4: Mostrar recomendaciones personalizadas

Esta funcionalidad consiste en que el vendedor, puede generar una lista de productos que sean adecuados para alguna mascota específica del usuario que está atendiendo. Esta se compone de los métodos: SeleccionarMascota, que muestra el código de cada mascota del usuario, y sirve para poder ingresar este código y retornar la mascota elegida. RecomendaciónPerro y RecomendacionGato estos muestran el nombre de la mascota (Perro o Gato), y genera la lista de Productos adecuados para la mascota, basado en sus características (Raza y Edad para perros, y Edad para Gatos). El método Funcionalidad, es el que permite la implementación de los otros métodos, e ingresar los datos del usuario, aquí primero se solicita ingresar el usuario, posteriormente muestra el ID de cada mascota, con su respectivo nombre del usuario, luego se ejecuta el método para seleccionar la mascota, y el método que retorna la lista de Productos para esta mascota. Finalmente se muestra la información de cada Producto que cumple con las características.

#### Secuencia de la funcionalidad





```
Ingrese el ID de usuario
Indique su eleccion : 1093186721
Para cual de sus mascotas desea la recomendacion
100 Dante
123 Jackie
103 Lupita
Ingrese el ID de su mascota
ID : 100
La mascota seleccionada es: Dante
Cachorro
Grande
Producto 1
~ Codigo = 1004
~ Producto = Croquetas Cachorro Plus
~ fechaDeVencimiento =18/05/2023
~ Descripcion = Comida en croquetas para perro cachorro de raza grande
~ Precio =12000.0
Producto 2
~ Codigo = 1006
~ Producto = Cuido Cachorro Extra
~ fechaDeVencimiento =17/04/2023
~ Descripcion = Comida diaria para perro cachorro de raza grande
~ Precio =15000.0
```

#### Funcionalidad 5: Mostrar estadísticas de la tienda

Esta funcionalidad consiste en que el vendedor puede observar las estadísticas de la tienda, entre ellas el número de ventas por vendedor ordenadas de menor a mayor, los productos más vendidos y los clientes destacados con su respectiva cantidad de compras. Esta funcionalidad utiliza los métodos `calcularProductoMasVendido` el cual calcula el producto más vendido sumando la cantidad de veces que se compra en cada una de las facturas de la tienda, `imprimirProductos` el cual imprime los productos de forma ordenada, lista que obtiene una lista para ordenar los clientes de mayor a menor, `imprimirClientes2` imprime los clientes ordenados y `getValoresOrdenados` obtiene la forma de ordenar los productos. Se relaciona con las clases `Vendedor`, `Producto`, `Usuario`, `Tienda` y `Factura`.

```
¿Qué estadísticas generales desea observar?

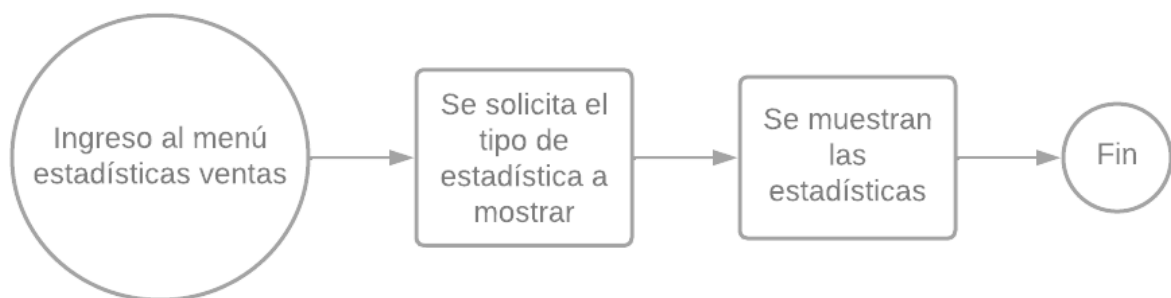
1. Número de ventas por vendedor
2. Productos más vendidos
3. Clientes destacados
4. Salir

Indique su eleccion : 1
Estas son las ventas por vendedor
Vendedor: Juan Pablo Arcila. Cantidad: 2
Vendedor: Roberto Barrientos. Cantidad: 3
Vendedor: Laura Vanegas. Cantidad: 4
```

```
Indique su eleccion : 2
Estos son los productos más vendidos
Producto: Whiskas para felino cachorro. Cantidad: 1
Producto: Comida humeda Cachorro Max. Cantidad: 2
Producto: Purina catchow adulto. Cantidad: 2
Producto: Cuido Cachorro Small. Cantidad: 2
Producto: Croquetas Adulto Mini. Cantidad: 2
Producto: Croquetas Adulto Plus. Cantidad: 2
Producto: Comida enlatada felino cachorro. Cantidad: 2
Producto: Cuido Cachorro Extra. Cantidad: 3
Producto: Comida enlatada felino adulto. Cantidad: 3
Producto: Whiskas para felino adulto. Cantidad: 3
Producto: Comida humeda Adulto Max. Cantidad: 3
Producto: Comida humeda Cachorro Junior. Cantidad: 3
Producto: Cuido Adulto Extra. Cantidad: 4
Producto: Cuido Adulto Small. Cantidad: 4
Producto: Croquetas Cachorro Plus. Cantidad: 4
Producto: Comida humeda Adulto Junior. Cantidad: 4
Producto: Croquetas Cachorro Mini. Cantidad: 5
Producto: Purina catchow cachorro. Cantidad: 7
```

```
Indique su eleccion : 3
Estos son los clientes destacados
Cliente: Manuela Sanchez. Cantidad: 3
Cliente: Manuela Arboleda. Cantidad: 2
Cliente: Valentina Higuiter. Cantidad: 2
Cliente: Jorge Sierra. Cantidad: 2
```

### Secuencia de la funcionalidad



## V. MANUAL DE USUARIO

### 1) Casos de Prueba

Para comprobar el funcionamiento correcto de las funcionalidades, se tienen los siguientes objetos de prueba:



## Vendedores

### Vendedor 1:

ID: 46785214  
Nombre: Juan Pablo Arcila  
Usuario: Juan Pablo Arcila  
Contraseña: holi12  
Cuenta Bancaria:  
Número de cuenta: 1085645245  
Saldo: 986543.3  
Pin: 4328

### Vendedor 2:

ID: 13205481  
Nombre: Roberto Barrientos  
Usuario: Roberto Barrientos  
Contraseña: adios13  
Cuenta Bancaria:  
Número de cuenta: 10793452  
Saldo: 123344.3  
Pin: 3342

### Vendedor 3:

ID: 119258647  
Nombre: Laura Vanegas  
Usuario: Laura Vanegas  
Contraseña: lau1245  
Cuenta Bancaria:  
Número de cuenta: 108956543  
Saldo: 341234.3  
Pin: 1733

## Usuarios

### Usuario 1:

ID: 1093186721  
Nombre: Manuela Arboleda  
Cuenta bancaria del usuario1:  
Número de cuenta: 108952347  
Saldo: 2023456.1  
Pin: 1852

### Usuario 2:

ID: 35164789  
Nombre: Valentina Higueta  
Cuenta bancaria del usuario2:  
Número de cuenta: 278782778  
Saldo: 800.5  
Pin: 8728

Este usuario se hace el supuesto que no tiene dinero

### Usuario 3:

ID: 25787278  
Nombre: Jorge Sierra  
Cuenta bancaria del usuario3:



Número de cuenta: 387787222  
Saldo: 500456.7  
Pin: 7885

Usuario 4:  
ID: 1027278278  
Nombre: Manuela Sanchez

Este usuario se hace el supuesto que no tiene cuenta bancaria

### **Tienda**

Cuenta bancaria de la tienda:  
Número de cuenta: 1077774562  
Saldo: 10889456.6  
Pin: 1023

### **Proveedores**

Proveedor 1:  
ID: 43825896  
Nombre: Pedro Casas  
Cuenta bancaria del proveedor1:  
Número de cuenta: 1089523479  
Saldo: 20858967.8  
Pin: 5782

Proveedor 2:  
ID: 43825897  
Nombre: Armando Marin  
Cuenta bancaria del proveedor2:  
Número de cuenta: 1872854124  
Saldo: 48248582.8  
Pin: 5784

Proveedor 3:  
ID: 58415  
Nombre: Sol lunar  
Cuenta bancaria del proveedor3:  
Número de cuenta: 1234567887  
Saldo: 0  
Pin: 1875

## **2) Prueba de funcionalidades**

### **Inicio:**

Cuando se inicia el programa, se imprime el siguiente mensaje:

```
Bienvenido a MascotasUN

Donde encontraras todo para tus peludos

Que desea hacer?

1. Iniciar sesion
0. Salir
```

Al darle de la opción 1, pide un usuario y una contraseña. Se puede iniciar sesión con el usuario y contraseña de cualquiera de los vendedores.

```
Ingresa el usuario:
Roberto Barrientos
Ingresa la contraseña:
adios13
Bienvenido, Roberto Barrientos
Que quieres hacer?
  1. Gestionar la aplicacion
  2. Navegar por las funcionalidades
  3. Salir
Indique su eleccion :
```

Validado el inicio de sesión, se despliega un menú que le pregunta al vendedor qué desea hacer, si gestionar la aplicación o navegar por las funcionalidades.

Gestionar la aplicación incluye funcionalidades CRUD que hacen parte de la lógica del negocio, desplegando las siguientes opciones:

```
Bienvenido, en este lugar puedes gestionar todo lo relacionado con tus companeros, usuarios, proveedores y productos de tu tienda
Que deseas hacer?
  1. Gestionar Productos
  2. Gestionar Proveedores
  3. Gestionar Usuarios
  4. Gestionar Companeros
  5. Regresar
```

Donde para cada tipo de gestión despliega un menú preguntando si desea ver, agregar, eliminar o editar el objeto correspondiente

```
Estas a punto de gestionar tus productos
Que deseas hacer?
  1. Ver productos
  2. Agregar nuevo producto
  3. Eliminar producto
  4. Editar producto existente
  5. Regresar
```

```
Estas a punto de gestionar tus proveedores
Que deseas hacer?
  1. Ver proveedores
  2. Agregar nuevo proveedor
  3. Eliminar proveedor
  4. Regresar
```

```
Estas a punto de gestionar tus usuarios
Que desees hacer?
1. Ver usuarios
2. Agregar nuevo usuario
3. Eliminar usuario
4. Regresar
```

```
Estas a punto de acceder a la gestion de tus companeros de trabajo
Que desees hacer?
1. Ver vendedor
2. Agregar nuevo vendedor
3. Regresar
```

El objetivo de este menú es modularizar los métodos que manipulan las instancias principales de nuestro programa, asignando por medio de los setters y getters correspondientes los atributos del objeto. A continuación, se presentará cómo estos objetos interactúan con las funcionalidades del programa.

Cuando estamos en el menú principal, y seleccionamos la opción de Navegar por las funcionalidades, se despliega el siguiente menú:

```
Que haremos hoy?
1. Realizar una venta
2. Comprar inventario
3. Cobrar una comision
4. Mostrar recomendaciones personalizadas con base a mascota
5. Mostrar estadisticas de ventas
6. Regresar
Indique su eleccion :
```

### Funcionalidad 1:

Al indicar 1, pide el ID del vendedor para iniciar la venta.

```
Para iniciar la venta por favor indique el ID a continuacion:
13205481
Hola Roberto Barrientos puede proceder con la compra

Para crear la factura ingrese el id del cliente
```

Se pide el ID del cliente para crear la venta. Este tiene que ser un cliente registrado. Luego, pide ingresar la fecha actual y se despliega otro menú, donde se puede mostrar los productos disponibles, con su respectivo costo y ID del producto, y agregar el producto a la compra.

```
Para crear la factura ingrese el id del cliente
1093186721
Se va a crear una factura a nombre de: Manuela Arboleda
Ingrese la fecha en formado: dd/mm/yyyy
25/10/2022
Realizar una compra
1. Mostrar productos disponibles
2. Agregar productos a la compra
3. Regresar
```

Al mostrar el menú se elige lo siguiente:

```
~ Comida humeda Adulto Juniortiene 10 productos disponibles y tiene un costo de 6000.0, ID: 1008
~ Croquetas Cachorro Minitiene 5 productos disponibles y tiene un costo de 13000.0, ID: 1010
~ Cuido Adulto Smalltiene 4 productos disponibles y tiene un costo de 16000.0, ID: 1009
~ Purina catchow cachorrotiene 7 productos disponibles y tiene un costo de 10000.0, ID: 1013
~ Croquetas Adulto Minitiene 2 productos disponibles y tiene un costo de 13000.0, ID: 1007
~ Comida enlatada felino adultotiene 3 productos disponibles y tiene un costo de 9000.0, ID: 1017
~ Comida humeda Cachorro Juniortiene 3 productos disponibles y tiene un costo de 6000.0, ID: 1011
~ Whiskas para felino adultotiene 4 productos disponibles y tiene un costo de 15000.0, ID: 1018
~ Cuido Cachorro Extratiene 3 productos disponibles y tiene un costo de 15000.0, ID: 1006
~ Comida enlatada felino cachorrotiene 2 productos disponibles y tiene un costo de 9000.0, ID: 1014
~ Croquetas Cachorro Plustiene 4 productos disponibles y tiene un costo de 12000.0, ID: 1004
~ Whiskas para felino cachorrotiene 1 productos disponibles y tiene un costo de 15000.0, ID: 1015
~ Comida humeda Adulto Maxtiene 3 productos disponibles y tiene un costo de 5000.0, ID: 1002
~ Purina catchow adultotiene 2 productos disponibles y tiene un costo de 10000.0, ID: 1016
~ Cuido Adulto Extratiene 4 productos disponibles y tiene un costo de 15000.0, ID: 1003
~ Comida humeda Cachorro Maxtiene 2 productos disponibles y tiene un costo de 5000.0, ID: 1005
~ Croquetas Adulto Plustiene 10 productos disponibles y tiene un costo de 12000.0, ID: 1001
~ Cuido Cachorro Smalltiene 2 productos disponibles y tiene un costo de 16000.0, ID: 1012
Su accion ha sido finalizada presione 1 para volver al menu
```

Al elegir agregar productos a la compra, se despliega el siguiente menú:

```
Agregar a la compra
1. Agregar un nuevo producto
2. Eliminar un producto
3. Mostrar resumen de factura
4. Finalizar compra
5. Configurar envio de compra
6. volver al menu compra
Indique su eleccion :
```

Al agregar un nuevo producto, pide el ID del producto y la cantidad de productos a comprar de dicho producto. Luego, pregunta si se desea agregar un nuevo producto o si regresar a la compra.

```
Ingrese el ID del producto que desea agregar:
1001
Ingrese la cantidad del producto que desea agregar
1
El producto se ha agregado con exito a la compra
Desea agregar mas productos?
1 para si, 0 en caso contrario
1
Ingrese el ID del producto que desea agregar:
1002
Ingrese la cantidad del producto que desea agregar
2
El producto se ha agregado con exito a la compra
Desea agregar mas productos?
1 para si, 0 en caso contrario
0
Su accion ha sido finalizada presione 1 para volver al menu
```

Al elegir la opción de eliminar una compra, pide ingresar el ID del producto a eliminar

```
Ingrese el id del producto que desea eliminar
1002
El producto fue borrado con exito
Desea eliminar mas productos?
1 para si, 0 en caso contrario
0
Su accion ha sido finalizada presione 1 para volver al menu
```



Al elegir la opción de mostrar resumen factura, se imprime lo siguiente:

```
Indique su eleccion : 3
Resumen de la factura:
~ Factura #1
~ Fecha = 25/10/2022
~ Cliente = Manuela Arboleda
~ Cantidad a pagar = 13920.0

1) Croquetas Adulto Plus 1 unidades, y tiene un precio unitario de: 12000.0
Su accion ha sido finalizada presione 1 para volver al menu
```

Al elegir la opción de configurar envío venta, pide indicar la dirección de envío

```
Desea que se realice un envio de esta?

Digite 1 si el cliente lo requiere o 0 si no
1
Ingrese el destino de su compra:
Medellin
Su envio fue creado exitosamente
Gracias por preferirnos
Su accion ha sido finalizada presione 1 para volver al menu
```

Finalmente, al elegir Finalizar compra, se guarda la compra, y se pregunta si se desea pagar en efectivo o con tarjeta. Si se elige efectivo, inmediatamente imprime la factura y se completa la compra. Si se elige con tarjeta, pide el pin de la tarjeta del cliente, y si es correcta se imprime la factura y se completa la compra.

```
Está a punto de finalizar la compra, está seguro que desea terminar la venta?
ingrese 1 para finalizar la compra y cero para continuar con la compra
1
Resumen de compra
~ Factura #1
~ Fecha = 25/10/2022
~ Cliente = Manuela Arboleda
~ Cantidad a pagar = 30067.2

El usuario desea pagar en efectivo?
ingrese 1 si el pago es en efectivo y 0 si es con tarjeta
0
Solicite al cliente la contraseña de su cuenta bancaria
1852
el total de la compra fue: 30067.2
Su accion ha sido finalizada presione 1 para volver al menu
```

**Funcionalidad 2:**





Al indicar 1, se muestra el inventario completo y luego se selecciona si se desea comprar o no inventario

```
¿Quieres ver el inventario completo o los productos en riesgo de agotarse?
  1. Ver inventario completo
  2. Ver productos en riesgo de agotarse
Indique su eleccion : 1
Este es su inventario actual
Producto: Whiskas para felino cachorro. Cantidad: 1
Producto: Comida humeda Cachorro Max. Cantidad: 2
Producto: Purina catchow adulto. Cantidad: 2
Producto: Cuido Cachorro Small. Cantidad: 2
Producto: Croquetas Adulto Mini. Cantidad: 2
Producto: Comida enlatada felino cachorro. Cantidad: 2
Producto: Cuido Cachorro Extra. Cantidad: 3
Producto: Comida enlatada felino adulto. Cantidad: 3
Producto: Comida humeda Adulto Max. Cantidad: 3
Producto: Comida humeda Cachorro Junior. Cantidad: 3
Producto: Cuido Adulto Extra. Cantidad: 4
Producto: Cuido Adulto Small. Cantidad: 4
Producto: Croquetas Cachorro Plus. Cantidad: 4
Producto: Whiskas para felino adulto. Cantidad: 4
Producto: Croquetas Cachorro Mini. Cantidad: 5
Producto: Purina catchow cachorro. Cantidad: 7
Producto: Croquetas Adulto Plus. Cantidad: 10
Producto: Comida humeda Adulto Junior. Cantidad: 10
¿Desea comprar inventario?
  1. Sí
  2. No
  3. Volver
```

Al indicar 2, se muestran los productos en riesgo de agotarse y luego se selecciona si se desea comprar o no inventario

```
¿Quieres ver el inventario completo o los productos en riesgo de agotarse?
  1. Ver inventario completo
  2. Ver productos en riesgo de agotarse
Indique su eleccion : 2
Estos son los productos en riesgo de agotarse
Producto: Whiskas para felino cachorro. Cantidad: 1
Producto: Comida humeda Cachorro Max. Cantidad: 2
Producto: Purina catchow adulto. Cantidad: 2
Producto: Cuido Cachorro Small. Cantidad: 2
Producto: Croquetas Adulto Mini. Cantidad: 2
Producto: Comida enlatada felino cachorro. Cantidad: 2
¿Desea comprar inventario?
  1. Sí
  2. No
  3. Volver
```

Al indicar 1 (Si), se muestra una lista de productos con su respectivo ID y luego se pregunta el ID y la cantidad que se desea comprar y se pregunta si se desea comprar más productos

```
Indique su eleccion : 1
¿A cuál proveedor desea comprar?
ID: 43825896. Nombre: Pedro Casas
ID: 43825897. Nombre: Armando Marin
ID: 58415. Nombre: Sol lunar
Indique el ID del proveedor a comprar: 58415
¿Qué producto desea comprar?
ID: 1001. Producto: Croquetas Adulto Plus
ID: 1002. Producto: Comida humeda Adulto Max
ID: 1003. Producto: Cuido Adulto Extra
ID: 1004. Producto: Croquetas Cachorro Plus
ID: 1006. Producto: Cuido Cachorro Extra
ID: 1007. Producto: Croquetas Adulto Mini
ID: 1008. Producto: Comida humeda Adulto Junior
ID: 1009. Producto: Cuido Adulto Small
ID: 1010. Producto: Croquetas Cachorro Mini
ID: 1012. Producto: Cuido Cachorro Small
ID: 1013. Producto: Purina catchow cachorro
ID: 1014. Producto: Comida enlatada felino cachorro
ID: 1015. Producto: Whiskas para felino cachorro
ID: 1016. Producto: Purina catchow adulto
ID: 1018. Producto: Whiskas para felino adulto
Indique el ID del producto a comprar: 1001
¿Cuántos desea comprar?
Indique su eleccion : 2
¿Desea comprar más productos?
Indique su eleccion :
1. Sí
2. No
Indique su eleccion :
2
¿Su compra ha sido realizada con éxito!
Saldo actual Tienda: 1.0877857E7
```

**Al indicar 2, (No) se devuelve al menu principal**

```
¿Desea comprar inventario?
1. Sí
2. No
3. Volver
Indique su eleccion : 2
Que quieres hacer?
1. Gestionar la aplicaciòn
2. Navegar por las funcionalidades
3. Salir
```

### Funcionalidad 3:

Al indicar 3, se ingresa a la funcionalidad de cobrar comisión por ventas. En este caso simplemente se ingresa el ID del vendedor al cual se le asignará la comisión, la cual se calcula según su número de ventas y cantidad vendida

```
Indique su eleccion : 3
¿Desea cobrar su comisión por las ventas realizadas?

    1. Si
    2. No, Salir

Indique su eleccion : 1
Ingrese su cedula:
13205481
Tus ventas totales fueron: 367029.5625
Por lo que tus comisiones son: 18351.478125
Tu dinero ha sido transferido exitosamente
Vuelva pronto!
```

#### Funcionalidad 4:

Al indicar 4, se pide el ID del cliente registrado al cual se le quiere realizar las recomendaciones personalizadas de productos para sus mascotas. Se despliega la lista de mascotas registradas de dicho cliente, y se pide que ingrese el ID de la mascota a la cual se le quieren recomendar los productos. Finalmente, se imprimen los productos recomendados.

```
Ingrese el ID de usuario
Indique su eleccion : 1093186721
Para cual de sus mascotas desea la recomendacion
100 Dante
123 Jackie
103 Lupita
Ingrese el ID de su mascota
ID : 100
La mascota seleccionada es: Dante
Cachorro
Grande
Producto 1
~Codigo = 1004
~Producto = Croquetas Cachorro Plus
~fechaDeVencimiento =18/05/2023
~Descripcion = Comida en croquetas para perro cachorro de raza grande
~Precio =12000.0
Producto 2
~Codigo = 1006
~Producto = Cuido Cachorro Extra
~fechaDeVencimiento =17/04/2023
~Descripcion = Comida diaria para perro cachorro de raza grande
~Precio =15000.0
```

#### Funcionalidad 5:

Al indicar 5, muestra las estadísticas principales relacionadas con la tienda, como el número de ventas por vendedor, los productos más vendidos y los clientes con más compras

¿Qué estadísticas generales desea observar?

1. Número de ventas por vendedor
2. Productos más vendidos
3. Clientes destacados
4. Salir

Indique su eleccion : 1

Estas son las ventas por vendedor

Vendedor: Juan Pablo Arcila. Cantidad: 2

Vendedor: Roberto Barrientos. Cantidad: 3

Vendedor: Laura Vanegas. Cantidad: 4

Indique su eleccion : 2

Estos son los productos más vendidos

Producto: Whiskas para felino cachorro. Cantidad: 1

Producto: Comida humeda Cachorro Max. Cantidad: 2

Producto: Purina catchow adulto. Cantidad: 2

Producto: Cuido Cachorro Small. Cantidad: 2

Producto: Croquetas Adulto Mini. Cantidad: 2

Producto: Croquetas Adulto Plus. Cantidad: 2

Producto: Comida enlatada felino cachorro. Cantidad: 2

Producto: Cuido Cachorro Extra. Cantidad: 3

Producto: Comida enlatada felino adulto. Cantidad: 3

Producto: Whiskas para felino adulto. Cantidad: 3

Producto: Comida humeda Adulto Max. Cantidad: 3

Producto: Comida humeda Cachorro Junior. Cantidad: 3

Producto: Cuido Adulto Extra. Cantidad: 4

Producto: Cuido Adulto Small. Cantidad: 4

Producto: Croquetas Cachorro Plus. Cantidad: 4

Producto: Comida humeda Adulto Junior. Cantidad: 4

Producto: Croquetas Cachorro Mini. Cantidad: 5

Producto: Purina catchow cachorro. Cantidad: 7

Indique su eleccion : 3

Estos son los clientes destacados

Cliente: Manuela Sanchez. Cantidad: 3

Cliente: Manuela Arboleda. Cantidad: 2

Cliente: Valentina Higueta. Cantidad: 2

Cliente: Jorge Sierra. Cantidad: 2