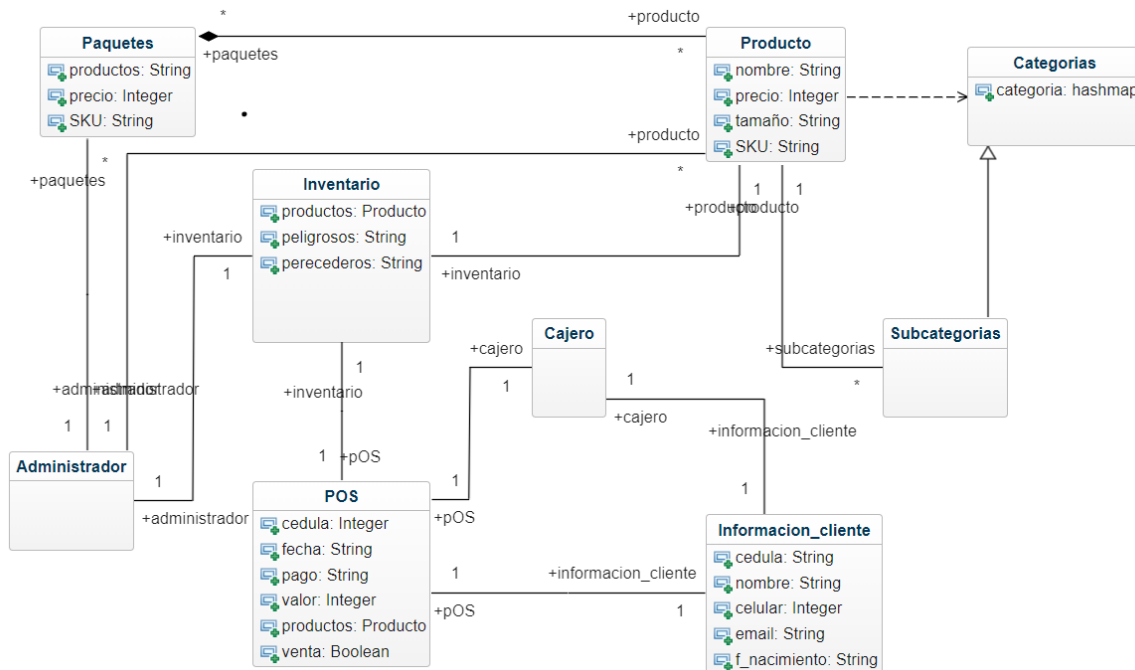


Taller 3

Contexto

Se planea diseñar un sistema que maneje el inventario de una ferretería incluyendo el POS (point-of-sale, el sistema que utilizan los cajeros para registrar las compras, recibir los pagos y actualizar el inventario). La ferretería planea iniciar con un sistema sencillo que será complicado en fases posteriores del proyecto. Dentro del nuevo sistema, la ferretería quiere tener la información básica de todos los productos que haya vendido alguna vez, así no tenga unidades en el momento. Para esto, el administrador de la ferretería podrá registrar, modificar o eliminar la información de cada producto, así como de las categorías a las que pertenezcan. Cada producto en el inventario tendrá un nombre, un precio, una descripción de su tamaño (1 litro, 10 tornillos, 500 gramos, etc.), y algo llamado un SKU (stock keeping unit). Un SKU es un identificador único que tienen los productos y que sirve para diferenciarlos dentro de un inventario de forma más precisa que con su nombre. Los productos de la ferretería pueden venderse de manera individual o en paquetes que pueden combinar varias unidades de productos diferentes, deben tener un SKU, y pueden tener un precio diferente a la suma de los productos que los compongan. La ferretería también manejará en su inventario productos perecederos (por ejemplo, pinturas o pegantes): por ahora no se quiere manejar la fecha exacta de vencimiento de cada ítem, pero sí se quiere que el sistema de inventario sepa cuál es la duración en días esperada para cada producto. Esta duración nunca debería ser menor a 60 días. Otros productos especiales son aquellos que son peligrosos, los cuales deben incluir una descripción de los cuidados que se deben tener para su transporte, manipulación y uso. Adicionalmente, los productos peligrosos podrían tener una restricción para que no puedan venderse a menores de edad y no debería permitirse que un mismo cliente compre en el mismo mes calendario, más de una unidad del mismo producto peligroso. Finalmente, el inventario manejará un árbol de categorías y subcategorías, que podrá ser tan profundo como sea necesario. Cada producto debe pertenecer exactamente a una categoría del árbol y se esperaría que fuera lo más específica posible 3. En el momento de registrar una compra, el cajero de la ferretería consulta los datos del comprador (usando el número de cédula que debería ser único) y, si no lo encuentra, lo registra como un nuevo comprador (con nombre, cédula, fecha de nacimiento, email y número celular). Para cada compra el POS registrará tendrá una fecha, un medio de pago, un valor total y el conjunto de productos que hicieron parte de la compra (el valor total debería corresponder a la suma del precio de los productos en la compra). El POS debería verificar las restricciones asociadas a los productos peligrosos.

Modelo de Dominio



Reglas:

El precio de los paquetes puede ser diferente de la suma total de los productos que lo componen.

Los productos peligrosos podrían tener una restricción para que no puedan venderse a menores de edad.

No debería permitirse que un mismo cliente compre en el mismo mes calendario, más de una unidad del mismo producto peligroso.

Cada producto debe pertenecer exactamente a una categoría del árbol y se esperaría que fuera lo más específica posible.

En el momento de registrar una compra, el cajero de la ferretería consulta los datos del comprador (usando el número de cédula que debería ser único) y, si no lo encuentra, lo registra como un nuevo comprador.

El POS debería verificar las restricciones asociadas a los productos peligrosos.

Descripción:

Administrador: la clase administrador se encargará de darle las funciones correspondientes al administrador de la ferretería cuanto ingrese en el sistema, este puede modificar paquetes y productos y revisar inventario.

Cajero: la clase cajero se encargará de darle las funciones correspondientes al cajero de la ferretería cuanto ingrese en el sistema, este utiliza la clase POS y la Informaicion_cliente.

Información_cliente: la clase Información_cliente guarda la información de un nuevo cliente, siendo esta: cedula, nombre, celular, email, fecha de nacimiento, siendo cada una de estas un atributo.

POS: la clase POS permite registrar una venta usando sus atributos para guardar la información de esta. La cedula del cliente, la fecha de venta, el medio de pago, el valor a pagar, los productos a vender, y si se puede vender (booleano) teniendo en cuenta las restricciones que existen.

Categorías: la clase Categorías permite guardar productos de una categoría, y siempre debe de haber un producto en una categoría creando una relación de dependencia.

Subcategorías: la clase Subcategorías permite guardar productos de una determinada subcategoría, esta funcionará similar a la clase Categorías por lo que hereda de esta.

Producto: la clase Producto permite guardar la información de los productos que la ferretería maneja, teniendo como atributos nombre, precio, tamaño, SKU, que es la información respectiva de los productos.


Inventario: la clase Inventario permite medir cuantos productos quedan en la ferretería y la información de estos sobre si son peligrosos o perecederos y la respectiva información de estos.

Paquetes: la clase Paquete no puede existir si no tiene un producto al menos, y tiene como atributos los productos del paquete correspondiente, el precio del paquete y el SKU.

Requerimientos Funcionales


USUARIO ADMINISTRADOR

1. Crear paquetes de productos

	Historia de Usuario
	Como administrador de la ferretería me gustaría poder crear paquetes de productos y modificarlos.


El sistema debe permitir que el administrador pueda crear paquetes de venta de productos de la ferretería.

2. Modificar productos

	Historia de Usuario
	Como administrador de la ferretería me gustaría poder registrar, modificar o eliminar productos del lugar.


El sistema debe permitir que el administrador de la ferretería pueda modificar la información de los productos ofrecidos.

3. Modificar categorías y subcategorías

	Historia de Usuario
	Como administrador de la ferretería me gustaría poder modificar las categorías y subcategorías a las que pertenece determinado producto.

El sistema debe permitir que el administrador de la ferretería pueda modificar las categorías en que se dividen los productos.


4. Revisar inventario

	Historia de Usuario
	Como administrador de la ferretería me gustaría poder revisar el inventario actual de la tienda.

El sistema debe permitir que el administrador de la ferretería pueda revisar el estado actual del inventario de la tienda.

USUARIO CAJERO


5. Consultar los datos del comprador

	Historia de Usuario
	Como cajero de la ferretería me gustaría poder consultar la información de un cliente que ya ha sido registrado en la tienda

El sistema debe permitir que el cajero de la ferretería pueda revisar la información de un cliente.


6. Registrar los datos del nuevo comprador

	Historia de Usuario
--	---------------------

	<p>Como cajero de la ferretería me gustaría poder agregar a un nuevo comprador al sistema.</p>
---	--

El sistema debe permitir que el cajero de la ferretería pueda agregar información de un nuevo cliente de la tienda.

7. Registrar una compra

	Historia de Usuario
	<p>Como cajero de la ferretería me gustaría poder registrar los datos de una venta a realizar.</p>

El sistema debe permitir que el cajero de la ferretería pueda guardar la información de una compra realizada, así como actualizar los datos del nuevo inventario.

8.

Requerimientos No Funcionales

El sistema debe correr de forma rápida, con la mayor eficiencia posible.

Restricciones del Proyecto

El sistema debe de contar con una interfaz de consola que permita al cajero y al administrador realizar sus respectivas funciones.

No se debe pasar un límite de tiempo en la carga de datos e información, dado que afectara la eficiencia.

Es necesario guardar la información de los clientes y sus compras en archivos externos.