



Universidad  
Técnica de  
Cotopaxi

UNIVERSIDAD TÉCNICA DE COTOPAXI CIENCIAS DE LA  
INGENIERÍA Y APLICADAS SISTEMAS DE  
INFORMACIÓN  
ANALISIS Y DISEÑO DE SISTEMAS DE INFORMACION.

**TEMA:**

**Desarrollo de un sistema de escritorio de gestión de ventas e inventario para la tienda de ropa "Variedades Norton"**

**INTEGRANTES:**

<b>Daquilema</b>	<b>David</b>
<b>Martínez</b>	<b>Fabian</b>
<b>Quintanilla</b>	<b>Bryan</b>
<b>Velásquez Carolina</b>	

**FECHA: 03 de Julio del 2024.**

**DOCENTE: Ing. Karla Cantuña**

Especificación de Requisitos según el estándar de IEEE 830

**IEEE Std. 830-1998**

15 de Mayo de 2024

Especificación de requisitos de software

Proyecto: Desarrollo de un sistema de escritorio de gestión de ventas e inventario para la tienda de ropa "Variedades Norton"

## **Resumen**

Este documento presenta las especificaciones de los requisitos para el desarrollo sistema de gestión de ventas e inventario para la tienda de ropa "Variedades Norton". Se introducen los requerimientos necesarios, incluyendo su propósito, alcance, acrónimos y abreviaturas que se abordarán a lo largo del documento. El formato utilizado es el de Especificación de Requisitos de Software (ERS) según la última versión del estándar IEEE 830. De acuerdo con IEEE, un buen Documento de Requisitos debe incluir toda la información presentada en dicho estándar, aunque no necesariamente siguiendo estrictamente su organización y formato.

1	Introducción .....	4
1.1	Propósito .....	4
1.2	Ámbito del Sistema .....	4
1.3	Definiciones, Acrónimos y Abreviaturas.....	4
1.4	Referencias.....	5
1.5	Visión General del Documento.....	5
2.	Descripción General .....	6
2.1	Perspectiva del Producto.....	6
2.2	Funciones del Producto .....	6
2.3	Características de los Usuarios .....	8
2.4	Restricciones .....	8
2.5	Suposiciones y Dependencias.....	9
2.6	Requisitos Futuros .....	9
3	Requisitos Específicos .....	9
3.1	Interfaces Externas .....	9
3.2	Funciones .....	9
3.3	Requisitos de Rendimiento.....	14
3.4	Restricciones de Diseño.....	15
3.5	Atributos del Sistema.....	15
3.6	Otros Requisitos .....	15
4.	Apéndices.....	15
5.	Formulario.....	23
5.1	Formulario de almacenes.....	23
5.2	Formulario de entrada y salida de datos.....	26
5.3	Formulario de flujo de datos.....	39
6.	Diccionario de datos generado mediante el software.....	74
7	Documentacion de los diagramas generado con el programa.....	215

# Introducción

## 1.1 Propósito

El presente documento tiene como propósito describir detalladamente los requisitos de software necesarios para el desarrollo del sistema de gestión de ventas e inventario para la tienda de ropa "Variedades Norton". Este documento está dirigido a los desarrolladores del sistema, gerentes de proyecto y partes interesadas de la tienda. Su objetivo es proporcionar una guía completa y clara que facilite la comprensión de los requisitos funcionales y no funcionales, garantizando así que el desarrollo del sistema se alinee con las necesidades comerciales y operativas de "Variedades Norton". Además, el documento pretende servir como referencia durante todo el ciclo de vida del proyecto, asegurando que todas las partes interesadas tengan una visión común y precisa del alcance y los objetivos del sistema propuesto.

## 1.2 Ámbito del Sistema

El sistema, denominado NortonSoft, permitirá la gestión eficiente de ventas, inventario y cuentas de usuario en "Variedades Norton". El sistema realizará las siguientes funciones:

- Consultar precios de productos.
- Registrar ventas y actualizar el stock.
- Gestionar el inventario (añadir, modificar y eliminar productos).
- Gestionar cuentas de usuario (cajeros y administradores).
- Generar recibos para impresión.

No incluirá funciones de contabilidad avanzada ni integración con sistemas de pago en línea.

Beneficios, Objetivos y metas esperados:

- Automatización de tareas como consulta de precios, registro de ventas y gestión de inventario reduciendo tiempo y esfuerzo.
- Mejorar la precisión del inventario y su actualización automática.
- Actualización en tiempo real del stock para decisiones más informadas.
- Diseñar el sistema para que pueda adaptarse a futuras expansiones y nuevas funcionalidades.

## 1.3 Definiciones, Acrónimos y Abreviaturas

- Acrónimos

Nombre	Descripción
ERS	Especificación de Requisitos de Software.
RFQ	Requisitos funcionales.
RNF	Requisitos no funcionales.
PC	Abreviatura de "Personal Computer" en inglés, se refiere a una computadora personal.

- Abreviaturas

Nombre	Descripción
MySQL	Sistema de gestión de bases de datos relacional de código abierto.

- Definiciones

Nombre	Descripción
Stock	Inventario de productos disponibles en una tienda.
Auditorías	Proceso de revisión y verificación de actividades y registros.
Reabastecimiento	Proceso de volver a suministrar productos que se han agotado.
Mantenibilidad	Facilidad con la que se puede actualizar y modificar el sistema.
Integración	Proceso de combinar varios componentes en un sistema unificado.
Control de acceso	Mecanismos para regular quién puede entrar y usar el sistema.
Pruebas de usuario	Evaluaciones realizadas por usuarios finales para verificar la funcionalidad del sistema.
Umbral mínimo	Nivel mínimo predefinido que desencadena una acción.
NortonSoft	Sistema de gestión de ventas e inventario desarrollado para la tienda de ropa "Variedades Norton".
Walmart	Cadena de supermercados de renombre internacional.
Windows Forms	Marco de desarrollo de aplicaciones de escritorio para Windows.
Zara	Cadena de tiendas de moda reconocida a nivel mundial.
Insight	Descubrimiento profundo sobre el comportamiento, las necesidades o las motivaciones de los consumidores.
Insights	Descubrimientos profundos sobre el comportamiento, las necesidades o las motivaciones de los consumidores.

#### 1.4 Referencias

Título de documento	Referencias
Standard IEEE 830-1998	IEEE

#### 1.5 Visión General del Documento

El documento (ERS) para el sistema NortonSoft está organizado en varias secciones. La primera sección proporciona una introducción general que incluye el propósito del documento, el ámbito del sistema, definiciones y acrónimos relevantes, referencias utilizadas y una visión general del contenido. La segunda sección describe el contexto del sistema, cubriendo la perspectiva del producto, las funciones principales que realizará, las características de los usuarios, las restricciones, suposiciones y dependencias. Finalmente, la tercera sección detalla los requisitos específicos del sistema y en el apartado de apéndice, se incluye el cuestionario inicial de requisitos del proyecto.

# Descripción General

## 2.1 Perspectiva del Producto

NortonSoft, el sistema de gestión de ventas e inventario para la tienda de ropa “Variedades Norton”, será un producto independiente diseñado para operar sin necesidad de integración con otros sistemas existentes, ya que la tienda no cuenta con sistemas previos. Su propósito es mejorar la eficiencia de las operaciones de venta y la gestión de inventario, reduciendo errores y facilitando la administración de cuentas de usuario, mediante funciones como consultar precios, registrar ventas y actualizar stock, gestionar inventario y administrar cuentas de usuario. Aunque NortonSoft no incluirá funciones de contabilidad avanzada ni integración con sistemas de pago en línea.

## 2.2 Funciones del Producto

### 2.3 Formularios de caso de uso Normal

Caso de Uso Normal		
<b>Nombre del caso de uso:</b>	<b>Registrar una Venta</b>	
<b>Código de caso de uso:</b>	CU-01	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	<p>El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.</p> <p>Existen productos registrados en el inventario.</p>	
<b>Descripción:</b>	Este caso de uso permite al usuario registrar una venta, seleccionar productos del inventario, ingresar cantidades y generar un recibo.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con productos disponibles y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar productos específicos.
	4	El sistema actualiza la tabla para mostrar solo los

		productos que coinciden con la búsqueda.		
	5	El usuario localiza el producto deseado y hace clic en el botón “Seleccionar” junto al producto.		
	6	El sistema agrega el producto a un carrito de ventas temporal y muestra un formulario para ingresar la cantidad deseada.		
	7	El usuario ingresa la cantidad y hace clic en “Agregar al carrito”.		
	8	El sistema actualiza el carrito de ventas con el producto y la cantidad especificada.		
	9	El usuario repite los pasos 3-8 para agregar más productos al carrito si es necesario.		
	10	El usuario hace clic en “Finalizar Venta”.		
	11	El sistema calcula el total de la venta y muestra un resumen para confirmar.		
	12	El usuario confirma la venta y hace clic en “Generar Recibo”.		
	13	El sistema guarda la venta en la base de datos y genera un recibo en formato PDF.		
	14	El sistema muestra un mensaje de confirmación y la opción de imprimir o guardar el recibo.		
<b>Postcondiciones</b>	La venta queda registrada en el sistema. Se genera y guarda un recibo en formato PDF.			
<b>Caso de Uso Normal</b>				
<b>Nombre del caso de uso:</b> Consultar Ventas				

<b>Código de caso de uso:</b>	CU-02	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	<p>El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.</p> <p>Existen ventas registradas en el sistema.</p>	
<b>Descripción:</b>	Este caso de uso permite al usuario consultar las ventas registradas en el sistema.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar ventas específicas por fecha, número de venta, o cliente.
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.
	5	El usuario selecciona una venta de la tabla para ver los detalles.
	6	El sistema muestra los detalles de la venta seleccionada.
<b>Postcondiciones</b>	El usuario puede ver los detalles de las ventas en el sistema.	

CU-02	<b>Consultar Ventas</b>	
Versión	1.0 (19/07/2024)	
Dependencias	<p>El sistema debe tener acceso a la base de datos de ventas.</p> <p>El usuario debe estar autenticado como Administrador o Vendedor.</p>	
Precondiciones	<p>El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.</p> <p>Existen ventas registradas en el sistema.</p>	
Descripción	Este caso de uso permite al usuario consultar las ventas registradas en el sistema.	
<b>Secuencia Normal</b>	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar ventas específicas por fecha, número de venta, o cliente.
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.
	5	El usuario selecciona una venta de la tabla para ver los detalles.

	6	El sistema muestra los detalles de la venta seleccionada.
Postcondición	El usuario puede ver los detalles de las ventas en el sistema.	
Excepciones	Paso	Acción
	3	Si no se encuentran ventas que coincidan con la búsqueda E1 El sistema muestra un mensaje indicando que no se encontraron ventas.
	5	Si ocurre un error al cargar los detalles de la venta E1 El sistema muestra un mensaje de error y permite al usuario intentar de nuevo.
Comentarios	6	Si los detalles de la venta no se pueden mostrar correctamente E1 El sistema muestra un mensaje de error y permite al usuario intentar de nuevo.
	La interfaz de consulta debe ser intuitiva y permitir una fácil navegación.  Los detalles de la venta deben mostrarse de manera clara y completa.	

<b>Caso de Uso Normal</b>		
<b>Nombre del caso de uso:</b>	<b>Actualizar Datos de Ventas</b>	
<b>Código de caso de uso:</b>	CU-03	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	El usuario (Administrador) ha iniciado sesión en el sistema.  Existen ventas registradas en el sistema.	
<b>Descripción:</b>	Este caso de uso permite al Administrador actualizar los datos de una venta registrada.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El Administrador accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.
	3	El Administrador utiliza el campo de búsqueda para encontrar la venta que desea actualizar.
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.
	5	El sistema muestra un formulario editable con los detalles de la venta seleccionada.
	6	El sistema muestra un formulario editable con los detalles de la venta seleccionada.
	7	El Administrador modifica los detalles de la venta según sea necesario.
	8	El Administrador hace clic en "Guardar Cambios".
	9	El sistema valida los cambios y actualiza la venta en la base de datos.

	10	El sistema muestra una confirmación de la actualización.
<b>Postcondiciones</b>	<p>La interfaz de actualización debe ser clara y permitir una fácil edición de los datos.</p> <p>La validación de datos debe ser rigurosa para evitar errores en la actualización.</p>	

<b>Caso de Uso Normal</b>		
<b>Nombre del caso de uso:</b>	<b>Eliminar Ventas</b>	
<b>Código de caso de uso:</b>	CU-04	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	<p>El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.</p> <p>Existen ventas registradas en el sistema.</p>	
<b>Descripción:</b>	Este caso de uso permite al usuario eliminar ventas registradas en el sistema.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar la venta que desea eliminar.
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.
	5	El usuario selecciona una venta de la tabla y hace clic en "Eliminar".
	6	El sistema muestra un mensaje de confirmación de la eliminación.
	7	El usuario confirma la eliminación.
	8	El sistema elimina la venta de la base de datos y actualiza la tabla de ventas.
	9	El sistema muestra una confirmación de la eliminación.
<b>Postcondiciones</b>	La venta queda eliminada del sistema.	

<b>Caso de Uso Normal</b>		
<b>Nombre del caso de uso:</b>	<b>Agregar Producto</b>	
<b>Código de caso de uso:</b>	CU-05	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	El usuario (Administrador) ha iniciado sesión en el sistema.	
<b>Descripción:</b>	Este caso de uso permite al Administrador agregar nuevos productos al inventario.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El Administrador accede al módulo de inventario desde el menú principal.

	2	El sistema muestra la interfaz del módulo de inventario, que incluye un formulario para agregar nuevos productos.
	3	El Administrador completa el formulario con los datos del nuevo producto, incluyendo nombre, precio, descripción, y cantidad.
	4	El Administrador hace clic en "Guardar".
	5	El sistema valida los datos ingresados.
	6	El sistema guarda el nuevo producto en la base de datos y actualiza la tabla de inventario.
	7	El sistema muestra una confirmación de que el producto ha sido agregado exitosamente.
<b>Postcondiciones</b>	El nuevo producto queda registrado en el inventario.	

<b>Caso de Uso Normal</b>		
<b>Nombre del caso de uso:</b>	<b>Consultar Productos</b>	
<b>Código de caso de uso:</b>	CU-06	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	<p>El usuario (Administrador) ha iniciado sesión en el sistema.</p> <p>Existen productos registrados en el inventario.</p>	
<b>Descripción:</b>	Este caso de uso permite al Administrador consultar productos registrados en el inventario.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El Administrador accede al módulo de inventario desde el menú principal.
	2	El sistema muestra la interfaz del módulo de inventario, que incluye una tabla con productos registrados y un campo de búsqueda.
	3	El Administrador utiliza el campo de búsqueda para encontrar productos específicos por nombre, categoría, u otros criterios.
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.
	5	El Administrador selecciona un producto de la tabla para ver los detalles.
	6	El sistema muestra los detalles del producto seleccionado.
<b>Postcondiciones</b>	El Administrador puede ver los detalles de los productos en el inventario.	

<b>Caso de Uso Normal</b>		
<b>Nombre del caso de uso:</b>	<b>Actualizar Información de Productos</b>	
<b>Código de caso de uso:</b>	CU-07	
<b>Versión:</b>	1.0 (19/07/2024)	

<b>Precondiciones:</b>	El usuario (Administrador) ha iniciado sesión en el sistema. Existen productos registrados en el inventario.	
<b>Descripción:</b>	Este caso de uso permite al Administrador actualizar la información de un producto registrado.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El Administrador accede al módulo de inventario desde el menú principal.
	2	El sistema muestra la interfaz del módulo de inventario, que incluye una tabla con productos registrados y un campo de búsqueda.
	3	El Administrador utiliza el campo de búsqueda para encontrar el producto que desea actualizar.
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.
	5	El Administrador selecciona un producto de la tabla y hace clic en "Actualizar".
	6	El sistema muestra un formulario editable con los detalles del producto seleccionado.
	7	El Administrador modifica los detalles del producto según sea necesario.
	8	El Administrador hace clic en "Guardar Cambios".
	9	El sistema valida los cambios y actualiza la información del producto en la base de datos.
	10	El sistema guarda los cambios en la base de datos.
<b>Postcondiciones</b>	La información del producto queda actualizada en el sistema.	

Caso de Uso Normal		
<b>Nombre del caso de uso:</b>	<b>Eliminar Producto</b>	
<b>Código de caso de uso:</b>	CU-08	
<b>Versión:</b>	1.0 (19/07/2024)	
<b>Precondiciones:</b>	El usuario (Administrador) ha iniciado sesión en el sistema. Existen productos registrados en el inventario.	
<b>Descripción:</b>	Este caso de uso permite al Administrador eliminar productos del inventario.	
<b>Secuencia normal:</b>	Paso	Acción
	1	El Administrador accede al módulo de inventario desde el menú principal.
	2	El sistema muestra la interfaz del módulo de inventario, que incluye una tabla con productos registrados y un campo de búsqueda.
	3	El Administrador utiliza el campo de búsqueda para encontrar el producto que desea eliminar.
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.
	5	El Administrador selecciona un producto de la tabla y hace clic en "Eliminar".

	6	El sistema muestra un mensaje de confirmación de eliminación.
	7	El Administrador confirma la eliminación.
	8	El sistema elimina el producto de la base de datos y actualiza la tabla de inventario.
	9	El sistema muestra una confirmación de que el producto ha sido eliminado exitosamente.
<b>Postcondiciones</b>	El producto queda eliminado del inventario.	

## 2.2 Formularios de caso detallado

CU-01	<b>Registrar una Venta</b>	
Versión	1.0 (19/07/2024)	
Dependencias	El sistema debe tener acceso a la base de datos de ventas e inventario.  El usuario debe estar autenticado como Administrador o Vendedor.	
Precondiciones	El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.  Existen productos registrados en el inventario.	
Descripción	Este caso de uso permite al usuario registrar una venta, seleccionar productos del inventario, ingresar cantidades y generar un recibo.	
Secuencia Normal	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con productos disponibles y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar productos específicos.
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.
	5	El usuario localiza el producto deseado y hace clic en el botón "Seleccionar" junto al producto.
	6	El sistema agrega el producto a un carrito de ventas temporal y muestra un formulario para ingresar la cantidad deseada.
	7	El usuario ingresa la cantidad y hace clic en "Agregar al carrito".
	8	El sistema actualiza el carrito de ventas con el producto y la cantidad especificada.
	9	El usuario repite los pasos 3-8 para agregar más productos al carrito si es necesario.
	10	El usuario hace clic en "Finalizar Venta".
	11	El sistema calcula el total de la venta y muestra un resumen para confirmar.
	12	El usuario confirma la venta y hace clic en "Generar Recibo".
	13	El sistema guarda la venta en la base de datos y genera un recibo en formato PDF.

	14	El sistema muestra un mensaje de confirmación y la opción de imprimir o guardar el recibo.
Postcondición		La venta queda registrada en el sistema.  Se genera y guarda un recibo en formato PDF.
Excepciones	Paso	Acción
	3	Si no se encuentran productos que coincidan con la búsqueda E1 El sistema muestra un mensaje indicando que no se encontraron productos
	6	Si el producto seleccionado no está disponible en el inventario E1 El sistema muestra un mensaje de error indicando que el producto no está disponible.
	11	Si ocurre un error en el cálculo del total de la venta E1 El sistema muestra un mensaje de error y solicita al usuario verificar los detalles de la venta.
Comentarios		Asegúrate de que el carrito de ventas esté claramente visible y accesible para el usuario.  La generación del recibo debe ser rápida y precisa.

CU-02	<b>Consultar Ventas</b>	
Versión	1.0 (19/07/2024)	
Dependencias	El sistema debe tener acceso a la base de datos de ventas.  El usuario debe estar autenticado como Administrador o Vendedor.	
Precondiciones	El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.  Existen ventas registradas en el sistema.	
Descripción	Este caso de uso permite al usuario consultar las ventas registradas en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar ventas específicas por fecha, número de venta, o cliente.
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.
	5	El usuario selecciona una venta de la tabla para ver los detalles.
	6	El sistema muestra los detalles de la venta seleccionada.

Postcondición	El usuario puede ver los detalles de las ventas en el sistema.		
Excepciones	Paso	Acción	
	3	Si no se encuentran ventas que coincidan con la búsqueda	
		E1	El sistema muestra un mensaje indicando que no se encontraron ventas.
	5	Si ocurre un error al cargar los detalles de la venta	
		E1	El sistema muestra un mensaje de error y permite al usuario intentar de nuevo.
	6	Si los detalles de la venta no se pueden mostrar correctamente	
		E1	El sistema muestra un mensaje de error y permite al usuario intentar de nuevo.
Comentarios	<p>La interfaz de consulta debe ser intuitiva y permitir una fácil navegación.</p> <p>Los detalles de la venta deben mostrarse de manera clara y completa.</p>		

CU-03	<b>Actualizar Datos de Ventas</b>		
Versión	1.0 (19/07/2024)		
Dependencias	<p>El sistema debe tener acceso a la base de datos de ventas.</p> <p>El usuario debe estar autenticado como Administrador.</p>		
Precondiciones	<p>El usuario (Administrador) ha iniciado sesión en el sistema.</p> <p>Existen ventas registradas en el sistema.</p>		
Descripción	Este caso de uso permite al Administrador actualizar los datos de una venta registrada.		
Secuencia Normal	Paso	Acción	
	1	El Administrador accede al módulo de ventas desde el menú principal.	
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.	
	3	El Administrador utiliza el campo de búsqueda para encontrar la venta que desea actualizar.	
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.	
	5	El Administrador selecciona una venta de la tabla y hace clic en "Actualizar".	
	6	El sistema muestra un formulario editable con los detalles de la venta seleccionada.	
	7	El Administrador modifica los detalles de la venta según sea necesario.	
	8	El Administrador hace clic en "Guardar Cambios".	
	9	El sistema valida los cambios y actualiza la venta en la base de datos.	
	10	El sistema muestra una confirmación de la actualización.	
Postcondición	La venta queda actualizada en el sistema.		

Excepciones	Paso	Acción
	3	Si no se encuentran ventas que coincidan con la búsqueda E1 El sistema muestra un mensaje indicando que no se encontraron ventas.
	7	Si los cambios realizados son inválidos (por ejemplo, datos incorrectos), E1 El sistema muestra un mensaje de error indicando la causa del problema.
	9	Si ocurre un error al actualizar la venta en la base de datos E1 El sistema muestra un mensaje de error y no guarda los cambios.
Comentarios	La interfaz de actualización debe ser clara y permitir una fácil edición de los datos.  La validación de datos debe ser rigurosa para evitar errores en la actualización.	

CU-04	<b>Eliminar Ventas</b>	
Versión	1.0 (19/07/2024)	
Dependencias	El sistema debe tener acceso a la base de datos de ventas.  El usuario debe estar autenticado como Administrador o Vendedor.	
Precondiciones	El usuario (Administrador o Vendedor) ha iniciado sesión en el sistema.  Existen ventas registradas en el sistema.	
Descripción	Este caso de uso permite al usuario eliminar ventas registradas en el sistema.	
Secuencia Normal	Paso	Acción
	1	El usuario accede al módulo de ventas desde el menú principal.
	2	El sistema muestra la interfaz del módulo de ventas, que incluye una tabla con ventas registradas y un campo de búsqueda.
	3	El usuario utiliza el campo de búsqueda para encontrar la venta que desea eliminar.
	4	El sistema actualiza la tabla para mostrar solo las ventas que coinciden con la búsqueda.
	5	El usuario selecciona una venta de la tabla y hace clic en "Eliminar".
	6	El sistema muestra un mensaje de confirmación de la eliminación.
	7	El usuario confirma la eliminación.
	8	El sistema elimina la venta de la base de datos y actualiza la tabla de ventas.
	9	El sistema muestra una confirmación de la eliminación.
Postcondición	La venta queda eliminada del sistema.	

Excepciones	Paso	Acción
	3	Si no se encuentran ventas que coincidan con la búsqueda E1 El sistema muestra un mensaje indicando que no se encontraron ventas.
	6	Si el sistema no puede mostrar el mensaje de confirmación, E1 El sistema muestra un mensaje de error y permite al usuario intentar de nuevo.
	8	Si ocurre un error al eliminar la venta de la base de datos E1 El sistema muestra un mensaje de error y no elimina el registro.
Comentarios	La eliminación debe ser confirmada para evitar borrados accidentales.  La interfaz debe permitir una visualización clara de las ventas a eliminar.	

CU-05	<b>Agregar Producto</b>	
Versión	1.0 (19/07/2024)	
Dependencias	El sistema debe tener acceso a la base de datos de inventario.  El usuario debe estar autenticado como Administrador.	
Precondiciones	El usuario (Administrador) ha iniciado sesión en el sistema.	
Descripción	Este caso de uso permite al Administrador agregar nuevos productos al inventario.	
Secuencia Normal	Paso	Acción
	1	El Administrador accede al módulo de inventario desde el menú principal.
	2	El sistema muestra la interfaz del módulo de inventario, que incluye un formulario para agregar nuevos productos.
	3	El Administrador completa el formulario con los datos del nuevo producto, incluyendo nombre, precio, descripción, y cantidad.
	4	El Administrador hace clic en "Guardar".
	5	El sistema valida los datos ingresados.
	6	El sistema guarda el nuevo producto en la base de datos y actualiza la tabla de inventario.
	7	El sistema muestra una confirmación de que el producto ha sido agregado exitosamente.
Postcondición	El nuevo producto queda registrado en el inventario.	
Excepciones	Paso	Acción
	3	Si el formulario no se completa correctamente (por ejemplo, falta algún campo obligatorio) E1 El sistema muestra un mensaje indicando los errores en el formulario.
	5	Si los datos ingresados son inválidos (por ejemplo, precio no numérico)

		E1	El sistema muestra un mensaje de error indicando el problema.	
	6	Si ocurre un error al guardar el producto en la base de datos		
Comentarios				El formulario debe ser claro y fácil de usar para evitar errores al ingresar datos.  La validación de datos debe ser rigurosa para mantener la integridad del inventario.

CU-06	<b>Consultar Productos</b>		
Versión	1.0 (19/07/2024)		
Dependencias	El sistema debe tener acceso a la base de datos de inventario.  El usuario debe estar autenticado como Administrador.		
Precondiciones	El usuario (Administrador) ha iniciado sesión en el sistema.  Existen productos registrados en el inventario.		
Descripción	Este caso de uso permite al Administrador consultar productos registrados en el inventario.		
Secuencia Normal	Paso	Acción	
	1	El Administrador accede al módulo de inventario desde el menú principal.	
	2	El sistema muestra la interfaz del módulo de inventario, que incluye una tabla con productos registrados y un campo de búsqueda.	
	3	El Administrador utiliza el campo de búsqueda para encontrar productos específicos por nombre, categoría, u otros criterios.	
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.	
	5	El Administrador selecciona un producto de la tabla para ver los detalles.	
	6	El sistema muestra los detalles del producto seleccionado.	
Postcondición	El Administrador puede ver los detalles de los productos en el inventario.		
Excepciones	Paso	Acción	
	3	Si no se encuentran productos que coincidan con la búsqueda	
		E1	El sistema muestra un mensaje indicando que no se encontraron productos.
	5	Si ocurre un error al cargar los detalles del producto	
		E1	El sistema muestra un mensaje de error y permite al Administrador intentar de nuevo.
	6	Si los detalles del producto no se pueden mostrar correctamente,	

		E1	El sistema muestra un mensaje de error y permite al Administrador intentar de nuevo.	
Comentarios	<p>La interfaz de consulta debe ser intuitiva y permitir una fácil navegación.</p> <p>Los detalles del producto deben mostrarse de manera clara y completa.</p>			

CU-07	<b>Actualizar Información de Productos</b>		
Versión	1.0 (19/07/2024)		
Dependencias	<p>El sistema debe tener acceso a la base de datos de inventario.</p> <p>El usuario debe estar autenticado como Administrador.</p>		
Precondiciones	<p>El usuario (Administrador) ha iniciado sesión en el sistema.</p> <p>Existen productos registrados en el inventario.</p>		
Descripción	Este caso de uso permite al Administrador actualizar la información de un producto registrado.		
Secuencia Normal	Paso	Acción	
	1	El Administrador accede al módulo de inventario desde el menú principal.	
	2	El sistema muestra la interfaz del módulo de inventario, que incluye una tabla con productos registrados y un campo de búsqueda.	
	3	El Administrador utiliza el campo de búsqueda para encontrar el producto que desea actualizar.	
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.	
	5	El Administrador selecciona un producto de la tabla y hace clic en "Actualizar".	
	6	El sistema muestra un formulario editable con los detalles del producto seleccionado.	
	7	El Administrador modifica los detalles del producto según sea necesario.	
	8	El Administrador hace clic en "Guardar Cambios".	
	9	El sistema valida los cambios y actualiza la información del producto en la base de datos.	
	10	El sistema muestra una confirmación de la actualización.	
Postcondición	La información del producto queda actualizada en el sistema.		
Excepciones	Paso	Acción	
	3	Si no se encuentran productos que coincidan con la búsqueda	
		E1	El sistema muestra un mensaje indicando que no se encontraron productos.
	7	Si los cambios realizados son inválidos (por ejemplo, datos incorrectos)	
		E1	El sistema muestra un mensaje de error indicando la causa del problema.

	9	Si ocurre un error al actualizar la información en la base de datos
	E1	El sistema muestra un mensaje de error y no actualiza la información.
Comentarios	La interfaz para actualizar la información del producto debe ser clara y permitir la edición sin errores.	
	La validación de datos debe garantizar que la información actualizada sea precisa.	

CU-07	<b>Eliminar Producto</b>	
Versión	1.0 (19/07/2024)	
Dependencias	El sistema debe tener acceso a la base de datos de inventario. El usuario debe estar autenticado como Administrador.	
Precondiciones	El usuario (Administrador) ha iniciado sesión en el sistema. Existen productos registrados en el inventario.	
Descripción	Este caso de uso permite al Administrador eliminar productos del inventario.	
Secuencia Normal	Paso	Acción
	1	El Administrador accede al módulo de inventario desde el menú principal.
	2	El sistema muestra la interfaz del módulo de inventario, que incluye una tabla con productos registrados y un campo de búsqueda.
	3	El Administrador utiliza el campo de búsqueda para encontrar el producto que desea eliminar.
	4	El sistema actualiza la tabla para mostrar solo los productos que coinciden con la búsqueda.
	5	El Administrador selecciona un producto de la tabla y hace clic en "Eliminar".
	6	El sistema muestra un mensaje de confirmación de eliminación.
	7	El Administrador confirma la eliminación.
	8	El sistema elimina el producto de la base de datos y actualiza la tabla de inventario.
	9	El sistema muestra una confirmación de que el producto ha sido eliminado exitosamente.
Postcondición	El producto queda eliminado del inventario.	
Excepciones	Paso	Acción
	3	Si no se encuentran productos que coincidan con la búsqueda
	E1	El sistema muestra un mensaje indicando que no se encontraron productos.
	5	Si ocurre un error al intentar eliminar el producto (por ejemplo, el producto ya no existe en la base de datos)
	E1	El sistema muestra un mensaje de error indicando que el producto no se puede eliminar.

	8	Sí ocurre un error al eliminar el producto de la base de datos	E1	El sistema muestra un mensaje de error y no elimina el producto.
	9	Si el sistema no puede mostrar la confirmación de eliminación	E1	El sistema muestra un mensaje de error y permite al Administrador intentar de nuevo.
Comentarios	<p>La confirmación de eliminación debe ser clara para evitar la eliminación accidental de productos.</p> <p>La interfaz debe permitir una fácil visualización y selección de productos para eliminar.</p>			

## 2.1 Características de los Usuarios

ACT-0002 Administrador	
Versión	1.0 ( 20/05/2024 )
Autores	<ul style="list-style-type: none"> <li>Bryan Quintanilla</li> <li>Carolina Velásquez</li> <li>David Daquilema</li> <li>Fabian Martinez</li> </ul>
Fuentes	<ul style="list-style-type: none"> <li>Marcia Trujillo</li> <li>Samuel Martinez</li> </ul>
Descripción	Este actor representa a los administradores encargados de tareas complejas como la gestión del inventario y la administración de usuarios en "NortonSoft". Los administradores poseen una experiencia técnica media y necesitan herramientas robustas para manejar sus responsabilidades con eficacia.
Comentarios	<p>Características:</p> <p>Nivel educacional: Educación media.</p> <p>Experiencia: Encargados de tareas complejas como gestión de inventario y administración de usuarios.</p> <p>Experiencia técnica: Experiencia técnica media.</p> <p>Necesidades:</p> <p>Interfaz de usuario intuitiva y herramientas fáciles de usar.</p> <p>Funcionalidades para agregar, modificar y eliminar productos del inventario.</p> <p>Capacidad para gestionar cuentas de usuario, asignar roles y permisos.</p> <p>Acceso a informes detallados y herramientas de auditoría.</p> <p>Ajustes manuales en el inventario y generación de alertas de stock.</p>

ACT-0001 Cajero	
Versión	1.0 ( 18/05/2024 )
Autores	<ul style="list-style-type: none"> <li>Bryan Quintanilla</li> <li>Carolina Velásquez</li> <li>David Daquilema</li> <li>Fabian Martinez</li> </ul>
Fuentes	<ul style="list-style-type: none"> <li>Marcia Trujillo</li> <li>Samuel Martinez</li> </ul>
Descripción	Este actor representa al personal encargado de realizar transacciones de venta en la tienda "Variedades Norton" utilizando el sistema "NortonSoft". Estos individuos tienen la responsabilidad de atender a los clientes, registrar las ventas de productos y mantener un adecuado control del inventario. Los cajeros deben ser capaces de utilizar el sistema de forma ágil y eficiente para garantizar un servicio al cliente óptimo.
Comentarios	<p>Características:</p> <p>Nivel educacional: Educación básica.</p> <p>Experiencia: Amplia experiencia en operaciones de ventas y consultas de precios.</p> <p>Experiencia técnica: Habilidades técnicas básicas.</p> <p>Necesidades:</p> <p>Interfaz de usuario intuitiva y herramientas fáciles de usar.</p> <p>Funcionalidades para agregar, modificar y eliminar productos del inventario.</p> <p>Capacidad para gestionar cuentas de usuario, asignar roles y permisos.</p> <p>Acceso a informes detallados y herramientas de auditoría.</p> <p>Ajustes manuales en el inventario y generación de alertas de stock.</p>

## 2.2 Restricciones

El "NortonSoft" operará dentro de un marco de restricciones y requisitos específicos que asegurarán su adecuado funcionamiento y cumplimiento con los estándares de la tienda "Variedades Norton". Estas restricciones incluyen:

- Políticas de la empresa: al respetar la política de la tienda de no se aceptan devoluciones y de descuentos se restringieron funciones para el módulo de ventas..
- Limitaciones del Hardware: El desarrollo estará sujeto a las capacidades y restricciones del hardware de la PC con Windows 10 que está disponible en la tienda.
- Base de datos: Se utilizará una base de datos MySQL para almacenar la información del "NortonSoft" ya que no se cuenta con presupuesto para una base de datos profesional o versiones de paga.
- Seguridad: Al ser una base de datos gratuito se debe proteger con los datos de forma robusta.
- Operaciones Paralelas: Considerando las operaciones concurrentes de ventas, actualización de inventario. La aplicación deberá ser capaz de ejecutar múltiples tareas simultáneamente de manera eficiente lo que complica un poco el desarrollo del sistema.
- Requisitos Habilidad: El equipo de desarrollo cuenta con cansamientos intermedios en el desarrollo de aplicaciones de escritorio utilizando C# y Windows Forms, así como experiencia en el diseño de interfaces de usuario intuitivas y funcionales.

## 2.3 Suposiciones y Dependencias

- Cambios en la estructura del personal de la tienda, como la adición o eliminación de roles, pueden requerir ajustes en los requisitos relacionados con los permisos de usuario y la gestión de roles.
- Modificaciones en los procedimientos operativos de la tienda, como cambios en el proceso de gestión de inventario, pueden afectar los requisitos de funcionalidad del sistema.
- Alteraciones en la estrategia de precios de la tienda, como descuentos por volumen o promociones temporales, pueden requerir ajustes en los requisitos de procesamiento de ventas y generación de recibos para reflejar los nuevos cálculos de precios y descuentos.
- El crecimiento de la tienda, ya sea en términos de ubicaciones adicionales o aumento de la base de productos, puede requerir cambios en los requisitos de escalabilidad del sistema para manejar un mayor volumen de transacciones y datos de inventario.

## 2.4 Requisitos Futuros

- Integración con sistemas de pago en línea, para ofrecer opciones de pago adicionales a los clientes y facilitar transacciones seguras y convenientes.
- Funcionalidades avanzadas de análisis de ventas y reportes, para proporcionar a la gerencia insights detallados sobre el desempeño de ventas, tendencias del mercado y análisis de inventario.
- Aplicación móvil para la gestión del inventario en tiempo real, permitiendo al personal de la tienda realizar tareas de gestión de inventario desde dispositivos móviles, lo que aumentaría la flexibilidad y la eficiencia operativa.

## Requisitos Específicos

### 3.1 Interfaces Externas

Nuestro sistema no cuenta con esto

### 3.2 Funciones

FRQ-0001	Ver Productos Disponibles
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"><li>• <a href="#">Bryan Quintanilla</a></li><li>• <a href="#">Carolina Velásquez</a></li><li>• <a href="#">David Daquilema</a></li><li>• <a href="#">Fabian Martinez</a></li></ul>
Fuentes	<ul style="list-style-type: none"><li>• <a href="#">Samuel Martinez</a></li></ul>
Dependencias	Ninguno
Descripción	El sistema deberá mostrar una lista con todos los productos disponibles de la tienda.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	Esta funcionalidad permitirá a los usuarios(Administrador/Vendedor) ver todos los productos disponibles para la venta. La lista debe ser clara y fácil de leer, con tiempos de carga mínimos para optimizar la eficiencia del usuario al consultar el inventario.

FRQ-0002	Buscar Productos al Registrar Venta
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"><li>• <a href="#">Bryan Quintanilla</a></li><li>• <a href="#">Carolina Velásquez</a></li><li>• <a href="#">David Daquilema</a></li><li>• <a href="#">Fabian Martinez</a></li></ul>
Fuentes	<ul style="list-style-type: none"><li>• <a href="#">Samuel Martinez</a></li></ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir buscar productos específicos dentro de la tabla de productos al registrar una venta. Esta búsqueda debe ser realizada en la interfaz del formulario de ventas.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	La búsqueda debe ser rápida y permitir aplicar filtros para encontrar productos de manera eficiente mientras se está en el proceso de registrar una venta. Debe facilitar la selección y validación del producto para completar la venta.

FRQ-0003	Ventas y validar stock
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"><li>• <a href="#">Bryan Quintanilla</a></li><li>• <a href="#">Carolina Velásquez</a></li><li>• <a href="#">David Daquilema</a></li><li>• <a href="#">Fabian Martinez</a></li></ul>
Fuentes	<ul style="list-style-type: none"><li>• <a href="#">Marcia Trujillo</a></li><li>• <a href="#">Samuel Martinez</a></li></ul>
Dependencias	Ninguno
Descripción	El sistema deberá poder procesar una venta y no vender cantidades superiores al stock disponible.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	La validación de la cantidad es crucial para evitar errores en el stock y asegurar que los usuarios(Administrador/Vendedor) no vendan más productos de los que hay disponibles. Debe ser precisa y fácil de usar.

FRQ-0004	<b>Carrito de Compras</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir vender más de un producto a la vez.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	El carrito de compras debe permitir a los usuarios agregar varios productos y cantidades, calcular el total de la compra y actualizar el stock en tiempo real. Debe ser intuitivo y mantener la precisión en el cálculo y el manejo del inventario.

FRQ-0005	<b>Recibo</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá generar un recibo al finalizar una venta y guardararlo.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	El recibo debe ser claro y legible. El archivo PNG generado debe almacenarse en una ubicación específica para asegurar un acceso rápido y fácil a los recibos cuando sea necesario.

FRQ-0006	<b>Consultar Ventas</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir consultar las ventas realizadas.
Importancia	importante
Urgencia	hay presión
Estado	en construcción
Estabilidad	media
Comentarios	Esta funcionalidad facilitará la revisión de ventas pasadas. La interfaz de búsqueda debe ser intuitiva para permitir una consulta eficiente y rápida de los registros de ventas.

FRQ-0007	<b>Actualizar Datos de Ventas</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá poder modificar la información de ventas registradas y guardar los cambios.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Las actualizaciones deben reflejarse en la lista y en la base de datos para mantener la coherencia de los datos. Debe ser sencillo modificar y guardar los cambios sin afectar la integridad del registro.

FRQ-0008	<b>Eliminar Registros de Ventas</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir eliminar registros de ventas.
Importancia	importante
Urgencia	hay presión
Estado	en construcción
Estabilidad	baja
Comentarios	Implementar una confirmación para la eliminación de registros para evitar la eliminación accidental de ventas. Esta funcionalidad debe asegurar que los registros se borren de manera segura y definitiva.

FRQ-0009	<b>Ingresar Nuevos Productos</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir agregar nuevos productos al inventario.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	Debe ser fácil y rápido ingresar nuevos productos al inventario, incluyendo todos los datos necesarios como nombre, precio y cantidad en stock. Los datos deben guardarse correctamente en la base de datos al confirmar la entrada.

FRQ-0010	<b>Consultar Productos en Inventario</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir consultar los productos en el inventario mediante una lista con opción de búsqueda.
Importancia	importante
Urgencia	hay presión
Estado	en construcción
Estabilidad	media
Comentarios	Facilitará ver y encontrar productos específicos en el inventario. La búsqueda debe ser clara y eficiente para asegurar una navegación rápida y efectiva a través de los productos.

FRQ-0011	<b>Actualizar Información de Productos</b>
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá mostrar una lista con los productos existentes y permitir su actualización.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	Al actualizar la información de un producto, los cambios deben reflejarse correctamente en la lista y en la base de datos para mantener datos precisos y actualizados.

FRQ-0012	Eliminar Productos del Inventario
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir eliminar productos del inventario de manera rápida.
Importancia	importante
Urgencia	hay presión
Estado	en construcción
Estabilidad	media
Comentarios	Confirmar la eliminación de productos debe actualizar la lista y la base de datos para garantizar que los cambios se reflejen de inmediato y evitar inconsistencias.

FRQ-0013	Acceso al Módulo de Ventas para Vendedores
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir que los vendedores solo puedan gestionar todo lo relacionado con las ventas.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	alta
Comentarios	Configurar los permisos para que los vendedores solo puedan acceder y utilizar el módulo de ventas, evitando el acceso al módulo de inventario para proteger la información sensible.

FRQ-0014	Acceso a Módulo de Ventas e Inventario para Administradores
Versión	1.0 ( 28/07/2024 )
Autores	<ul style="list-style-type: none"> <li><a href="#">Bryan Quintanilla</a></li> <li><a href="#">Carolina Velásquez</a></li> <li><a href="#">David Daquilema</a></li> <li><a href="#">Fabian Martinez</a></li> </ul>
Fuentes	<ul style="list-style-type: none"> <li><a href="#">Marcia Trujillo</a></li> <li><a href="#">Samuel Martinez</a></li> </ul>
Dependencias	Ninguno
Descripción	El sistema deberá permitir que los administradores accedan a ambos módulos: ventas e inventario.
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	media
Comentarios	Los administradores deben tener acceso completo a todas las funciones de ambos módulos para gestionar y supervisar tanto las ventas como el inventario sin restricciones.

### 3.3 Requisitos de Rendimiento

<b>NFR-0002</b>	<b>Disponibilidad del Sistema</b>
<b>Versión</b>	1.0 ( 22/05/2024 )
<b>Autores</b>	<ul style="list-style-type: none"> <li>• <a href="#">Bryan Quintanilla</a></li> <li>• <a href="#">Carolina Velásquez</a></li> <li>• <a href="#">David Daquilema</a></li> <li>• <a href="#">Fabian Martinez</a></li> </ul>
<b>Fuentes</b>	<ul style="list-style-type: none"> <li>• <a href="#">Marcia Trujillo</a></li> <li>• <a href="#">Samuel Martinez</a></li> </ul>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá estar disponible durante las horas de operación de la tienda.
<b>Importancia</b>	vital
<b>Urgencia</b>	inmediatamente
<b>Estado</b>	en construcción
<b>Estabilidad</b>	alta
<b>Comentarios</b>	Garantiza que el sistema esté operativo cuando sea necesario para las actividades comerciales.

<b>NFR-0004</b>	<b>Rendimiento de Operaciones Clave</b>
<b>Versión</b>	1.0 ( 22/05/2024 )
<b>Autores</b>	<ul style="list-style-type: none"> <li>• <a href="#">Bryan Quintanilla</a></li> <li>• <a href="#">Carolina Velásquez</a></li> <li>• <a href="#">David Daquilema</a></li> <li>• <a href="#">Fabian Martinez</a></li> </ul>
<b>Fuentes</b>	<ul style="list-style-type: none"> <li>• <a href="#">Marcia Trujillo</a></li> <li>• <a href="#">Samuel Martinez</a></li> </ul>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá tener operaciones clave, así como las ventas y el registro de inventario, deberán completarse rápidamente.
<b>Importancia</b>	importante
<b>Urgencia</b>	inmediatamente
<b>Estado</b>	en construcción
<b>Estabilidad</b>	media
<b>Comentarios</b>	Mejora la eficiencia y la experiencia del usuario, evitando demoras en la atención al cliente.

### 3.4 Restricciones de Diseño

Nuestro sistema no cuenta con esto

### 3.5 Atributos del Sistema

NFR-0001		Interfaz de Usuario Intuitiva
Versión	1.0 ( 22/05/2024 )	
Autores	<ul style="list-style-type: none"><li><a href="#">Bryan Quintanilla</a></li><li><a href="#">Carolina Velásquez</a></li><li><a href="#">David Daquilema</a></li><li><a href="#">Fabian Martinez</a></li></ul>	
Fuentes	<ul style="list-style-type: none"><li><a href="#">Marcia Trujillo</a></li><li><a href="#">Samuel Martinez</a></li></ul>	
Dependencias	Ninguno	
Descripción	El sistema deberá tener una <i>interfaz de usuario intuitiva y fácil de usar para los cajeros y administradores.</i>	
Importancia	importante	
Urgencia	inmediatamente	
Estado	en construcción	
Estabilidad	media	
Comentarios	Una interfaz amigable reduce el tiempo de capacitación y mejora la eficiencia operativa.	

NFR-0003		Respaldo de Datos
Versión	1.0 ( 22/05/2024 )	
Autores	<ul style="list-style-type: none"><li><a href="#">Bryan Quintanilla</a></li><li><a href="#">Carolina Velásquez</a></li><li><a href="#">David Daquilema</a></li><li><a href="#">Fabian Martinez</a></li></ul>	
Fuentes	<ul style="list-style-type: none"><li><a href="#">Marcia Trujillo</a></li><li><a href="#">Samuel Martinez</a></li></ul>	
Dependencias	Ninguno	
Descripción	El sistema deberá tener <i>mecanismos de respaldo de datos automáticos y periódicos.</i>	
Importancia	importante	
Urgencia	inmediatamente	
Estado	en construcción	
Estabilidad	alta	
Comentarios	Protege contra la pérdida de datos y asegura la continuidad del negocio.	

NFR-0006		Protección de Datos Sensibles
Versión	1.0 ( 22/05/2024 )	
Autores	<ul style="list-style-type: none"><li><a href="#">Bryan Quintanilla</a></li><li><a href="#">Carolina Velásquez</a></li><li><a href="#">David Daquilema</a></li><li><a href="#">Fabian Martinez</a></li></ul>	
Fuentes	<ul style="list-style-type: none"><li><a href="#">Marcia Trujillo</a></li><li><a href="#">Samuel Martinez</a></li></ul>	
Dependencias	Ninguno	
Descripción	El sistema deberá proteger <i>datos sensibles, deberán estar encriptados y protegidos contra accesos no autorizados.</i>	
Importancia	vital	
Urgencia	inmediatamente	
Estado	en construcción	
Estabilidad	alta	
Comentarios	Asegura la confidencialidad y seguridad de la información crítica.	

<b>NFR-0007</b>	<b>Compatibilidad con Equipos de Impresión</b>
<b>Versión</b>	1.0 ( 22/05/2024 )
<b>Autores</b>	<ul style="list-style-type: none"> <li>• <a href="#">Bryan Quintanilla</a></li> <li>• <a href="#">Carolina Velásquez</a></li> <li>• <a href="#">David Daquilema</a></li> <li>• <a href="#">Fabian Martinez</a></li> </ul>
<b>Fuentes</b>	<ul style="list-style-type: none"> <li>• <a href="#">Marcia Trujillo</a></li> <li>• <a href="#">Samuel Martinez</a></li> </ul>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá ser compatible con los equipos estándar de impresión disponibles.
<b>Importancia</b>	quedaría bien
<b>Urgencia</b>	hay presión
<b>Estado</b>	en construcción
<b>Estabilidad</b>	alta
<b>Comentarios</b>	Facilita la integración y uso de los equipos de impresión ya existentes en la tienda.

<b>NFR-0008</b>	<b>Portabilidad del Sistema</b>
<b>Versión</b>	1.0 ( 22/05/2024 )
<b>Autores</b>	<ul style="list-style-type: none"> <li>• <a href="#">Bryan Quintanilla</a></li> <li>• <a href="#">Carolina Velásquez</a></li> <li>• <a href="#">David Daquilema</a></li> <li>• <a href="#">Fabian Martinez</a></li> </ul>
<b>Fuentes</b>	<ul style="list-style-type: none"> <li>• <a href="#">Marcia Trujillo</a></li> <li>• <a href="#">Samuel Martinez</a></li> </ul>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá ser fácil de trasladar e instalar en diferentes ubicaciones.
<b>Importancia</b>	quedaría bien
<b>Urgencia</b>	hay presión
<b>Estado</b>	en construcción
<b>Estabilidad</b>	alta
<b>Comentarios</b>	Permite flexibilidad en la expansión y reubicación del sistema.

<b>NFR-0010</b>	<b>Escalabilidad del Sistema</b>
<b>Versión</b>	1.0 ( 22/05/2024 )
<b>Autores</b>	<ul style="list-style-type: none"> <li>• <a href="#">Bryan Quintanilla</a></li> <li>• <a href="#">Carolina Velásquez</a></li> <li>• <a href="#">David Daquilema</a></li> <li>• <a href="#">Fabian Martinez</a></li> </ul>
<b>Fuentes</b>	<ul style="list-style-type: none"> <li>• <a href="#">Marcia Trujillo</a></li> <li>• <a href="#">Samuel Martinez</a></li> </ul>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá ser escalable para adaptarse a futuros cambios en la cantidad de productos y ventas.
<b>Importancia</b>	vital
<b>Urgencia</b>	inmediatamente
<b>Estado</b>	en construcción
<b>Estabilidad</b>	alta
<b>Comentarios</b>	Permite el crecimiento y expansión del negocio sin necesidad de reemplazar el sistema.

### 3.6 Otros Requisitos

Nuestro sistema no cuenta con esto

## 4 Apéndices

# CUESTIONARIO INICIAL DE REQUISITOS DEL PROYECTO NORMA IEEE830

### **Tipos de requerimientos.**

Según el estándar internacional de Especificación de Requerimientos IEEE830, los documentos de definición y especificación de requerimientos deben contemplar los siguientes aspectos resumidos por [Pfleeger, 2002] como se indica a continuación:

<b>AMBIENTE FISICO</b>	<b>Respuesta</b>
1 ¿Dónde está el equipo que el sistema necesita para funcionar?	1. El equipo está ubicado en la caja de la tienda, en el área de pagos. 2. Actualmente, solo hay una localización. 3. No, es la primera vez que se contara con un cajero, antes se administraba con una libreta donde se guardaba lo obtenido en ventas, pero se ha contratado a una cajero por lo cual ya no se puede llevar el registro de manera manual, a lo que se optó por un sistema que se espera que se espera que ayude a gestionar el proceso de venta y stock de manera eficiente.
2 ¿Existe una localización o varias? ¿Existe tecnología ya implementada a la que	
3 integrarse? Detallar. ¿Hay restricciones ambientales como temperatura, humedad o interferencia	4. No, el clima es normal. 5. Preferiríamos un sistema hecho por expertos, con el sistema escaneo e impresión estándar del mercado.
4 Magnética?	
5 Si no existiera; ¿Se inclina por una tecnología específica de preferencia?	6. Sí, estamos abiertos a propuestas que mejoren la eficiencia.
6 ¿Es posible proponer nuevas tecnologías o mejoras que potencien la solución?	
<b>INTERFACES DE HARDWARE</b>	<b>Respuesta</b>
7 ¿La entrada proviene de uno o más sistemas?	7. No, por eso se espera que este sistema ayude a tener por primera vez un punto de venta.
8 ¿La salida va a uno o más sistemas? ¿Existe una manera preestablecida en que	8. Esperamos que sea principalmente uno, el sistema de facturación e inventario de la tienda.
9 deben formatearse los datos?	9. No, es la primera vez que tendremos este tipo de sistema
<b>USUARIOS Y FACTORES HUMANOS</b>	<b>Respuesta</b>
10 ¿Quién usará el sistema?	10. Principalmente, los cajeros de la tienda y los dueños para ingresar el stock.

- 11 ¿Habrá varios tipos de usuario? Identifique Categorías en función de Áreas. Ej: Admin, Ventas, etc.
- 12 ¿Cuántos Roles o Actores ocupar el Sistema según su análisis?
- 13 ¿Cuál es el nivel de habilidad de cada tipo de usuario?
- 14 ¿Qué clase de entrenamiento posee cada tipo de usuario?
- 15 ¿Cuán fácil le será al usuario comprender y utilizar el sistema?
- 16 ¿Cuán difícil le resultará al usuario hacer uso indebido del sistema?
11. El sistema deberá estar diseñada adecuadamente para ese caso de usuarios ya que está el cajero, administrador.
12. Dos roles principales: Cajero, Administrador de Tienda.
13. Los cajeros tienen habilidades intermedias, y los dueños (Administrador) tiene habilidades básicas.
14. Los cajeros recibirán entrenamiento específico en el uso del sistema, los administradores en la gestión del sistema.
15. Debe ser muy fácil de usar, con una interfaz intuitiva y accesible.
16. Debe ser difícil hacer uso indebido del sistema, con permisos y restricciones claras según el rol del usuario

	<b>FUNCIONALIDAD</b>	<b>Respuesta</b>
17	¿Qué hará el sistema?	17. El sistema debe ser capaz de ingresar productos, llevar contabilidad, gestionar ventas,
18	¿Cuándo lo hará?	18. Cuando se realizan las ventas y cuando toque ingresar el stock
19	¿Cómo funciona el proceso del servicio a mejorar (Modelo de negocio) donde se insertará?	19. No hay servicio a mejorar ya que es nuevo 20. No se cuenta con un modelo de operación profesional debido a que es su primera vez con este tipo de sistemas.
20	¿Existen varios modos de operación?	21. Principalmente el modo de venta estándar, pero se espera en una futura mejora importantes como incluir un modo de entrenamiento y un modo de mantenimiento.
21	¿Cómo y cuándo cambia el sistema?	22. Tal vez por la gran cantidad de productos que se guardara en la base de datos
22	¿Existen restricciones de la velocidad de ejecución, tiempo de respuesta o rendimiento?	23. Información referente a la ropa comprada y su costo y lo que se ha vendido en el mes. 24. Func. 1: Procesamiento de ventas.
23	¿Qué aspectos de la información será administrable en el sistema con respecto a su negocio?	Func. 2: Gestión del inventario. Func. 3: Generación de recibos. Func. 4: Reportes de ventas diarios. Func. 5: Gestión de devoluciones y cambios. Func. 6: Control de acceso y roles de usuarios.
24	<b>Enumere las principales funcionalidades que necesita que el Sistema realice a nivel de atributo:</b>	Func. 1 Func. 2 Func. 3 Func. 4 Func. 5 Func. 6

Func. 7

Func. N

	<b>Tipo de Interfaz de Usuarios Esperada</b>	<b>Respuesta</b>
25	Describa el tipo de apariencia de diseño y características que necesita la interfaz de su sistema a nivel de Usuario	25. Interfaz amigable, intuitiva, con botones grandes y claros, y una navegación sencilla.
26	Señale alguna referencia de un Sistema existente a nivel de Interfaz que se asimile a su estandar esperado	26. Un sistema similar podría ser el de las cajas desupermercados modernos como Walmart o tiendas de ropa como Zara.
	<b>Documentación</b>	<b>Respuesta</b>
27	¿Cuánta documentación se requiere?	27. Se necesita documentación completa para usuarios y administradores.
28	¿Debe estar en línea, en papel o en ambos?	28. Debe estar disponible en ambos formatos.
29	¿A que audiencia está orientado cada tipo de información?	
30	¿Posee información y datos recopilados a nivel documental que se deban trabajar a nivel de formularios y administrar en el sistema?	29. Usuarios finales (cajeros) y administradores. 30. Ninguno
	<b>Datos</b>	<b>Respuesta</b>
31	¿Cuál será el formato de los datos, tanto para la entrada como para la salida?	31. Entrada: teclado. Salida: reportes en formato pdf.
32	¿Cuán a menudo serán recibidos o enviados?	32 Es desconocido al trabajar por primera vez con este tipo de Sistema.
33	¿Cuán exactos deben ser?	33 Deben ser muy exactos, especialmente en la gestión de inventario.
34	¿Con qué grado de precisión deben hacerse los cálculos?	34 Con precisión exacta debe trabajar con el dólar en términos monetarios.
35	¿Cuántos datos fluyen a través del sistema?	35 Solo los datos necesarios para controlar la venta y el stock.
36	¿Debe retenerse algún dato por algún período de tiempo?	36 Sí, los datos de transacciones deben ser retenidos por al menos un año.
37	¿Se debe realizar un poblamiento inicial, parcial o total de los datos a ser utilizados por el sistema?	37 Sí, debe realizarse en el inventario y las configuraciones de usuario
	<b>Recursos</b>	<b>Respuesta</b>
38	¿Qué recursos materiales, personales o de otro tipo se requieren para construir, utilizar y mantener el sistema?	38 Computadoras y personal (Cajero).
39	¿Qué habilidades debe tener un desarrollador de sistemas para desarrolladores?	39 Habilidades en desarrollo de software, bases de datos, integración de hardware y conocimientos de sistemas de punto de venta.
40	¿Cuánto espacio físico será ocupado por el sistema?	40 El espacio físico es limitado a la zona de caja, con un mostrador de aproximadamente 1 metro cuadrado.

41	¿Cuáles son los requerimientos de energía, calefacción o acondicionamiento de aire?	41 Energía eléctrica estándar y un ambiente con aire acondicionado.
42	¿Existe cronograma para un prescripto desarrollo?	42 Sí, idealmente dentro de los próximos tres meses.
43	¿Existe un límite sobre la cantidad de dinero a gastar en el desarrollo o en hardware y software?	43 Sí, el presupuesto es limitado y debe ajustarse a las necesidades básicas sin exceder los \$70.

<b>Seguridad</b>		<b>Respuesta</b>
44	¿Debe controlarse el acceso al sistema o a la información?	
45	¿Cómo se podrán aislar los datos de un usuario de los de otros?	44 Sí, con controles de acceso por usuario.
46	¿Cómo podrán aislarse los programas de usuario de los otros programas y del sistema operativo?	45 Mediante la asignación de roles y permisos.
47	¿Con qué frecuencia deben hacerse copias de respaldo?	46 Uso de cuentas de usuario y restricciones a nivel de sistema operativo.
48	¿Las copias de respaldo deben almacenarse en un lugar diferente?	4 Cada tres meses. 7 Sí, preferiblemente en un servidor externo o 4 en la nube. 8

<b>Aseguramiento de la calidad</b>		<b>Respuesta</b>
49	¿Cuáles son los requerimientos para la confiabilidad, disponibilidad, facilidad de mantenimiento, seguridad y demás atributos de calidad?	
50	¿Cómo deben demostrarse las características del sistema a terceros?	49 Se espera la confiabilidad y disponibilidad, facilidad correcta.
51	¿El sistema debe detectar y aislar defectos?	50 Mediante pruebas de usuario, certificaciones y demostraciones en vivo.
52	¿Cuál es el promedio de tiempo prescripto entre fallas?	
53	¿Existe un tiempo máximo permitido para la recuperación del sistema después de una falla?	51 No es necesario.
54	¿El mantenimiento corregirá los errores, o incluirá también el mejoramiento del sistema?	52 La que se necesite.
55	¿Qué medidas de eficiencia se aplicarán al uso de recursos y al tiempo de respuesta?	53 No.
56	¿Cuán fácil debe ser mover el sistema de una ubicación a otra o de un tipo de computadora a otro?	54 El mantenimiento por el momento. 5 Optimizaciones de software y hardware para garantizar tiempos de respuesta rápidos y uso eficiente de los recursos 5 Debe ser relativamente fácil, con configuraciones mínimas. 6

<b>Otros</b>	<b>Respuesta</b>
--------------	------------------

## 5 Formularios

### 5.1 Formulario de almacenes

ID: D1

Nombre: BD\_Usuario

Alias: Usuarios

Descripción: Almacena la información básica de los usuarios del sistema, incluyendo la hora de ingreso.

Característica del almacén de datos

Tipo de archivo  Computadora  Manual  
Formato de archivo  Base de datos  Indexado  Secuencial  Directos  
Tamaño del registro 200 Tamaño del bloque 8000  
Números de registros Máximos 1000 Promedio 500  
Porcentaje de crecimiento anual 10

Nombre del conjunto de datos: Usuarios

Copia de miembro: BD\_USER

Estructura de datos: ID (PK) + Usuario + Fecha\_Ingreso

Clave principal: ID\_USUARIO

Claves secundarias: No aplica

Comentario: Almacena la información básica de los usuarios del sistema, incluyendo la fecha y hora de ingreso.

ID: D2

Nombre: BD\_Ventas

Alias: Ventas

Descripción: Registra todas las transacciones de venta realizadas.

Característica del almacén de datos

Tipo de archivo  Computadora  Manual  
Formato de archivo  Base de datos  Indexado  Secuencial  Directos  
Tamaño del registro 200 Tamaño del bloque 16000  
Números de registros Máximos 5000 Promedio 1000  
Porcentaje de crecimiento anual 15

Nombre del conjunto de datos: Ventas

Copia de miembro: BD\_VENTASTIENDA

Estructura de datos: ID\_Venta (PK) + Fecha + ID\_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio\_Total

Clave principal: IDVenta

Claves secundarias: No aplica

Comentario: Facilita el seguimiento detallado de las ventas realizadas por el sistema.

ID: D3

Nombre: BD\_Inventario

Alias: Inventario

Descripción: Gestiona el inventario de productos.

Característica del almacén de datos

Tipo de archivo  Computadora  Manual  
Formato de archivo  Base de datos  Indexado  Secuencial  Directos  
Tamaño del registro 200 Tamaño del bloque 8000  
Números de registros Máximos 2000 Promedio 800  
Porcentaje de crecimiento anual 12

Nombre del conjunto de datos: Inventario  
Copia de miembro:BD\_INTARIOTIENDA  
Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible  
Clave principal ID\_PRODUCTO  
Claves secundarias: No aplica  
Comentario: Permite gestionar el inventario de productos disponibles para la venta.

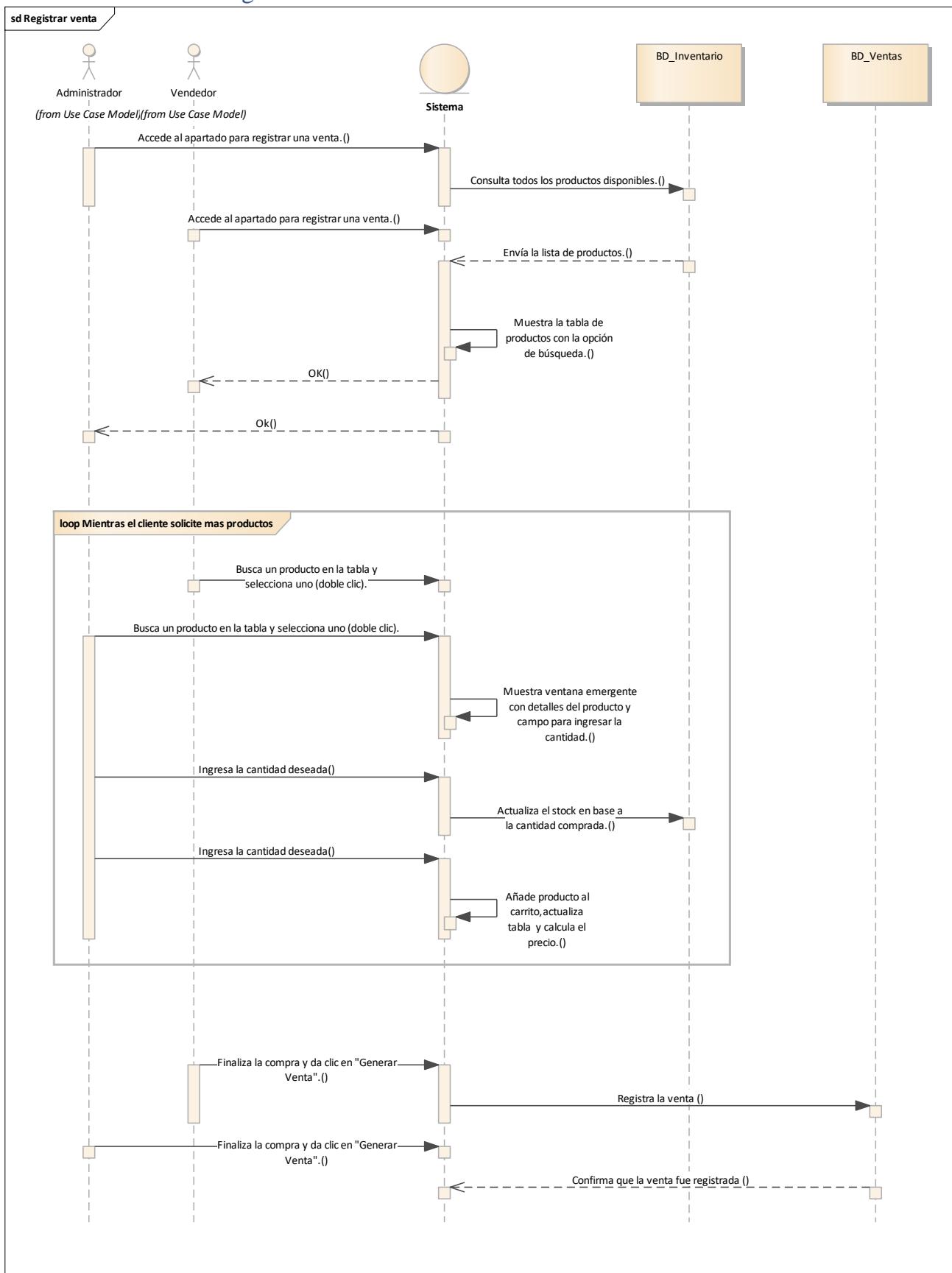
ID: D4  
Nombre: BD\_Recibos  
Alias: Facturas  
Descripción: :Almacena los recibos generados para transacciones.  
Característica del almacén de datos

Tipo de archivo  Computadora  Manual  
Formato de archivo  Base de datos  Indexado  Secuencial  Directos  
Tamaño del registro 200 Tamaño del bloque 2000  
Números de registros Máximos 3000 Promedio 500  
Porcentaje de crecimiento anual 8

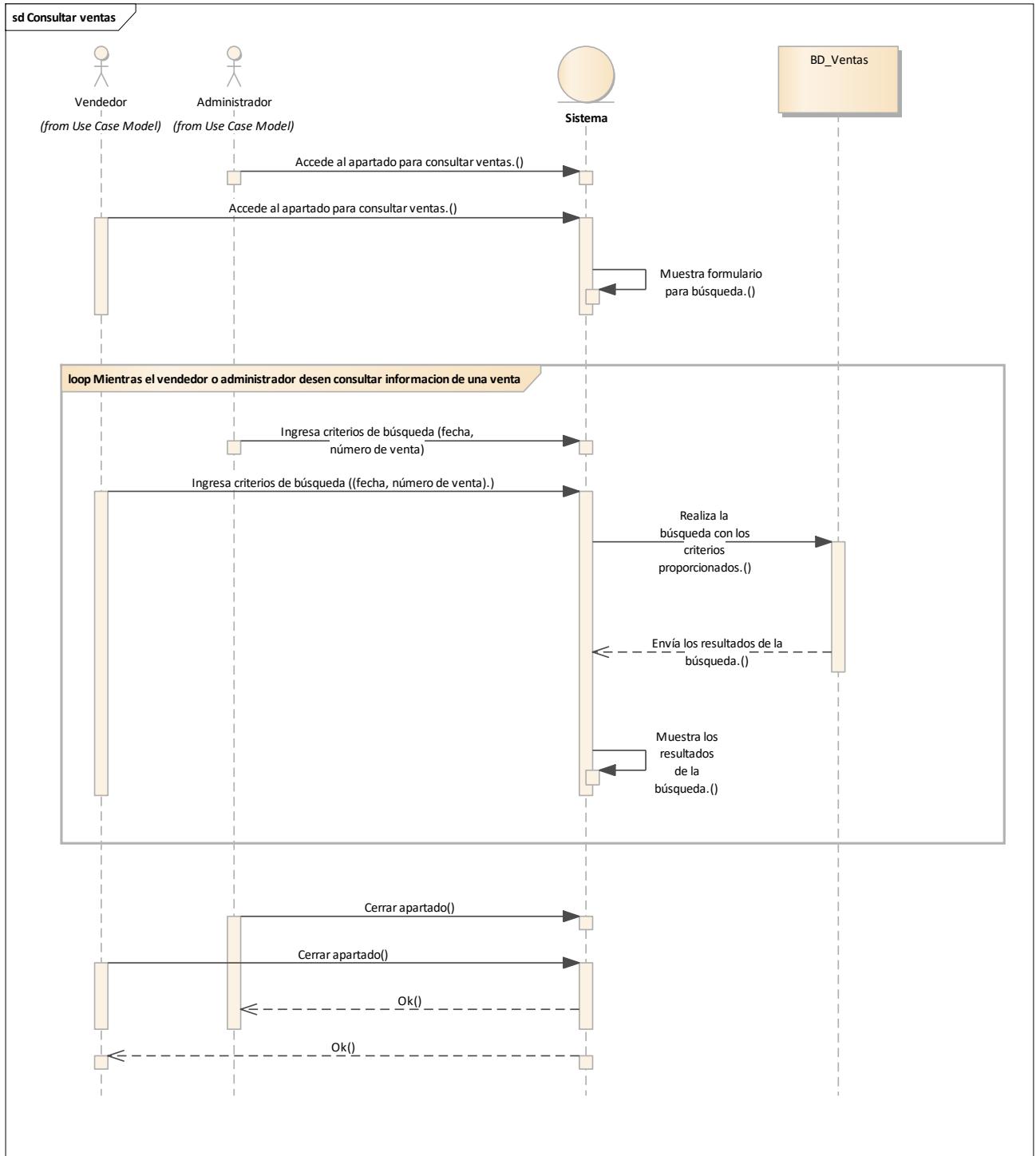
Nombre del conjunto de datos: Recibos  
Copia de miembro:BD\_RECIBOSVENTAS  
Estructura de datos ID\_Recibo (PK) + ID\_Venta (FK hacia Venta) + Direccion\_PDF  
Clave principal ID\_Recibo  
Claves secundarias: No aplica  
Comentario: Guarda la información de la dirección de donde se guardó el recibode la venta en formato pdf

## 5.2 Diagrama de secuencia

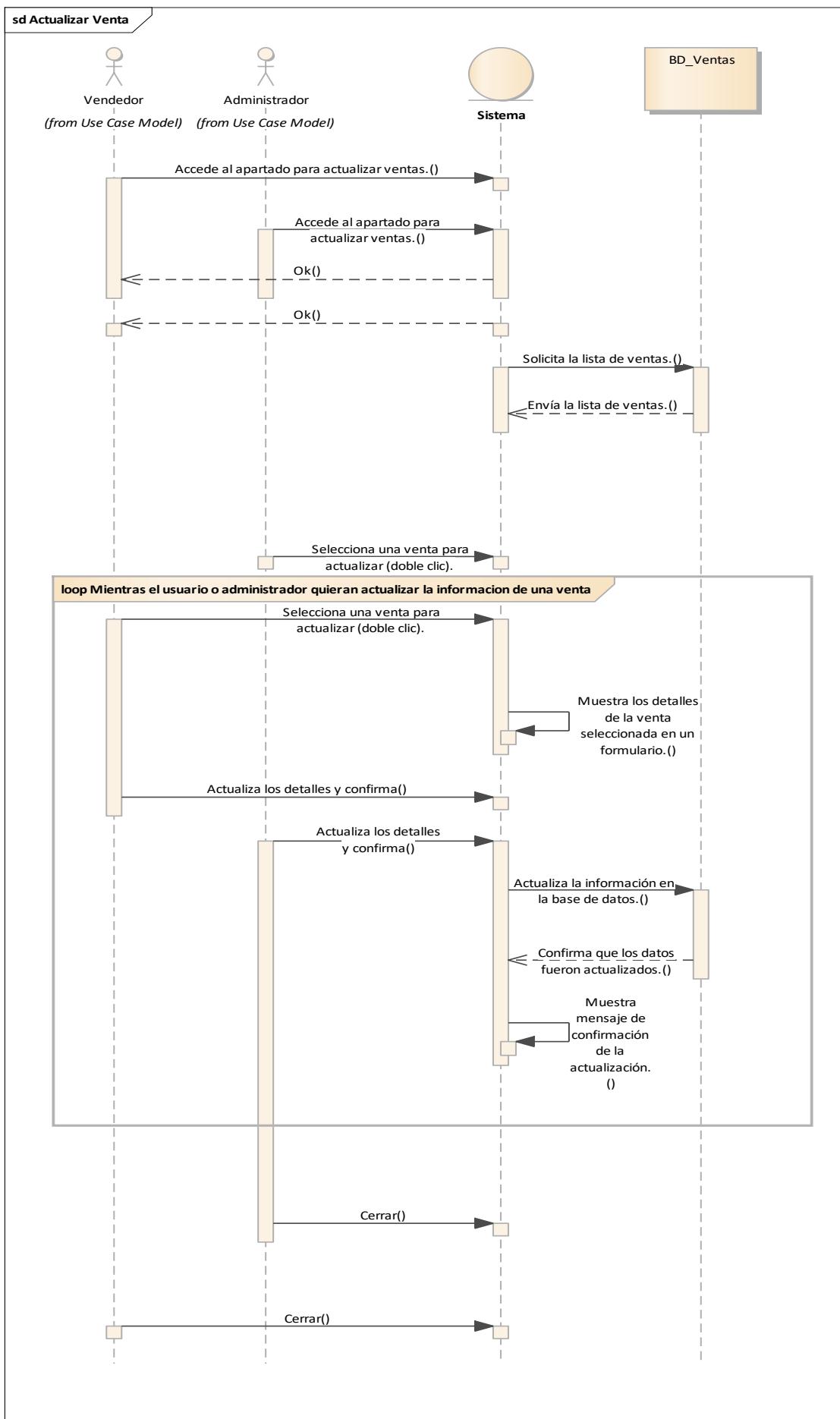
### 5.2.1 Registrar venta



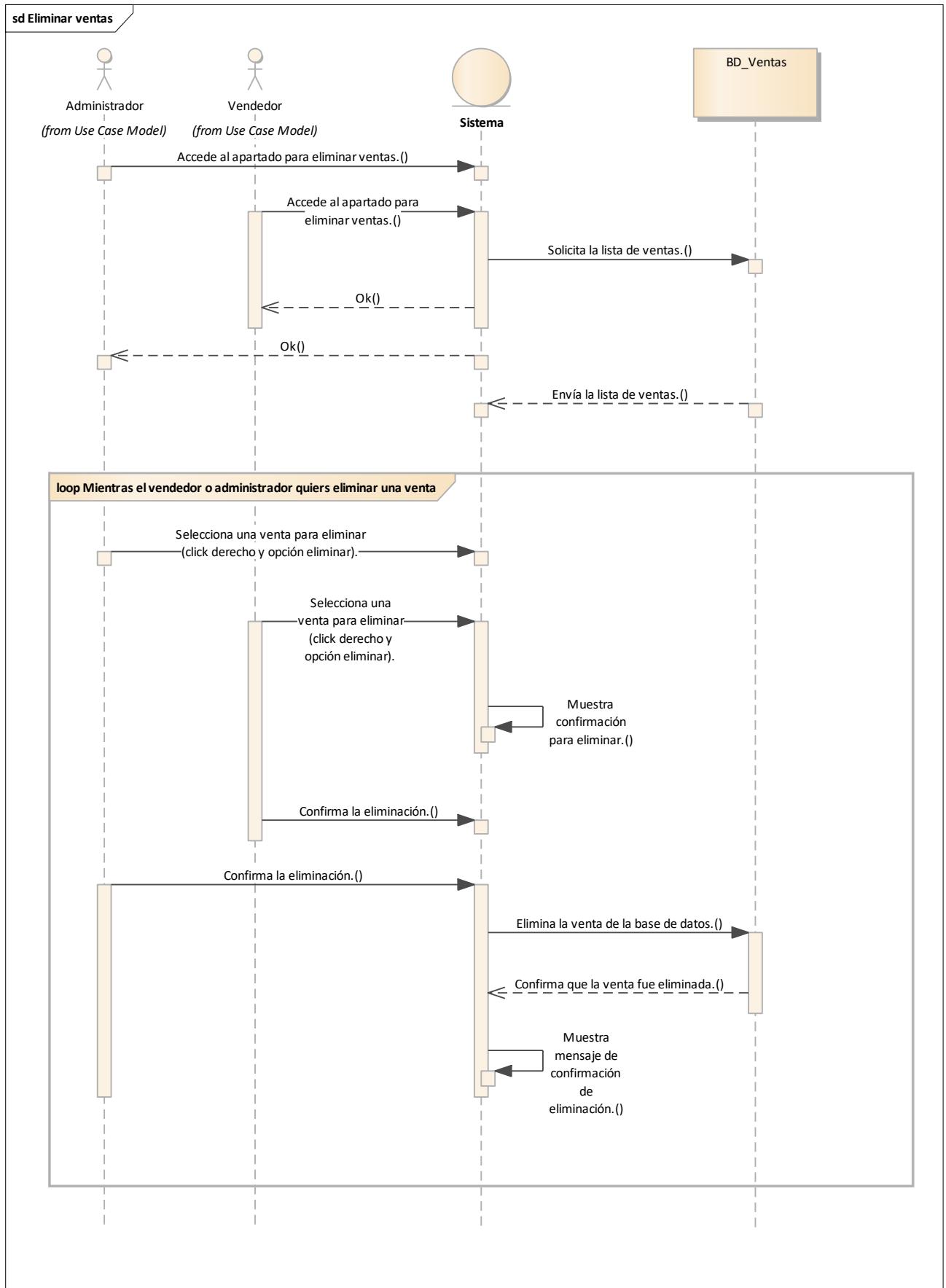
## 5.2.2 Consultar venta



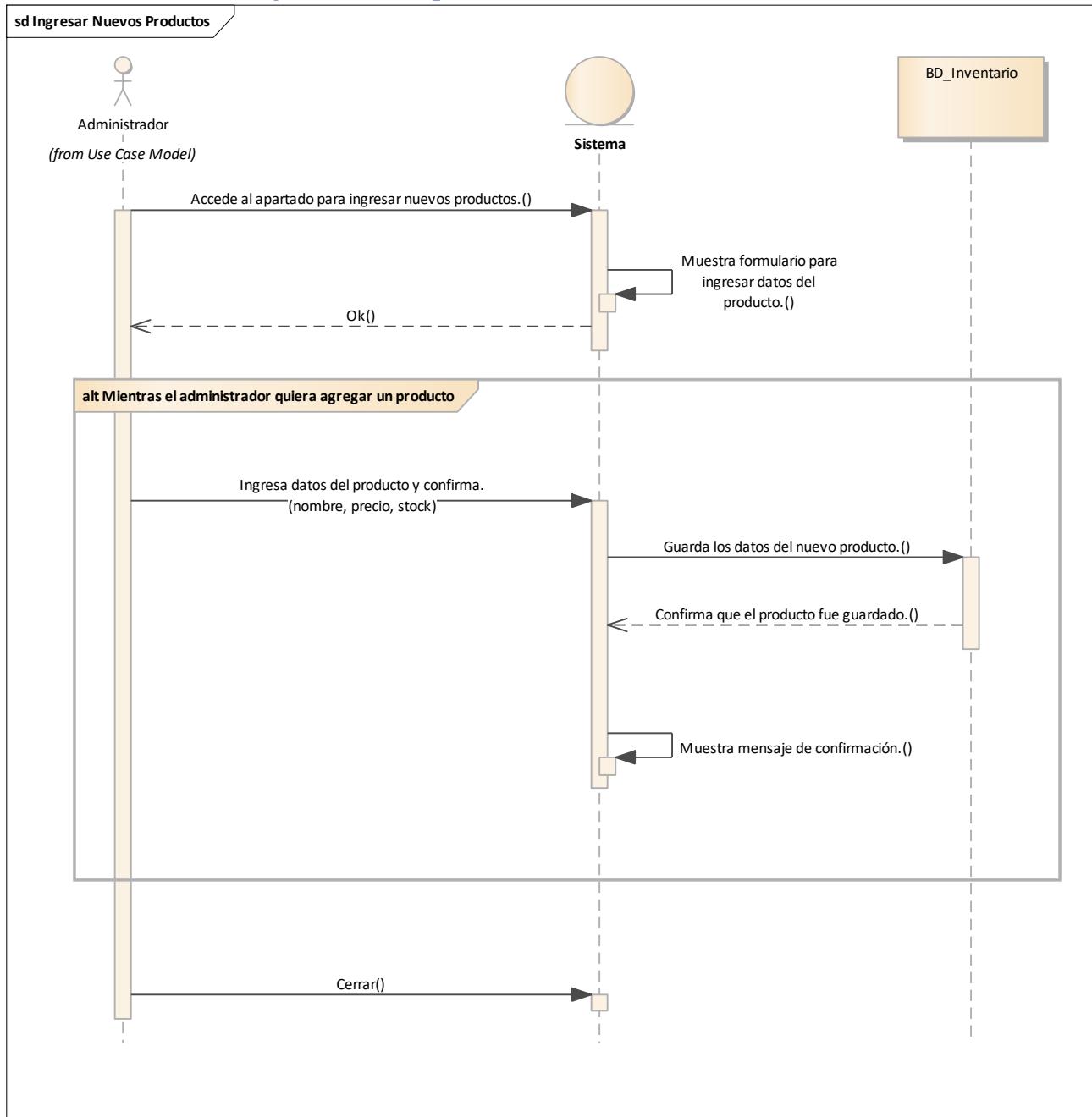
### 5.2.3 Actualizar venta



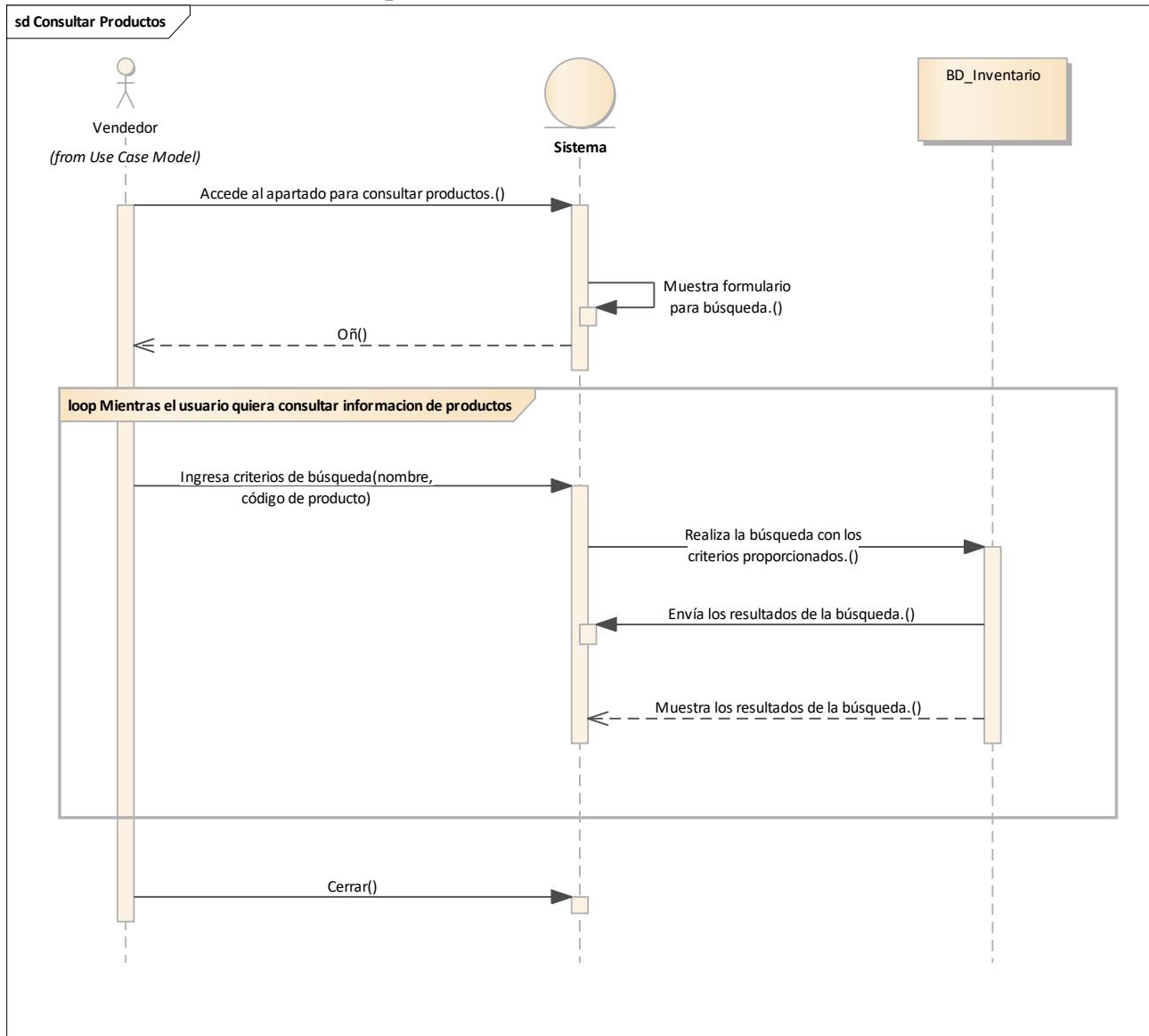
## 5.2.4 Eliminar venta



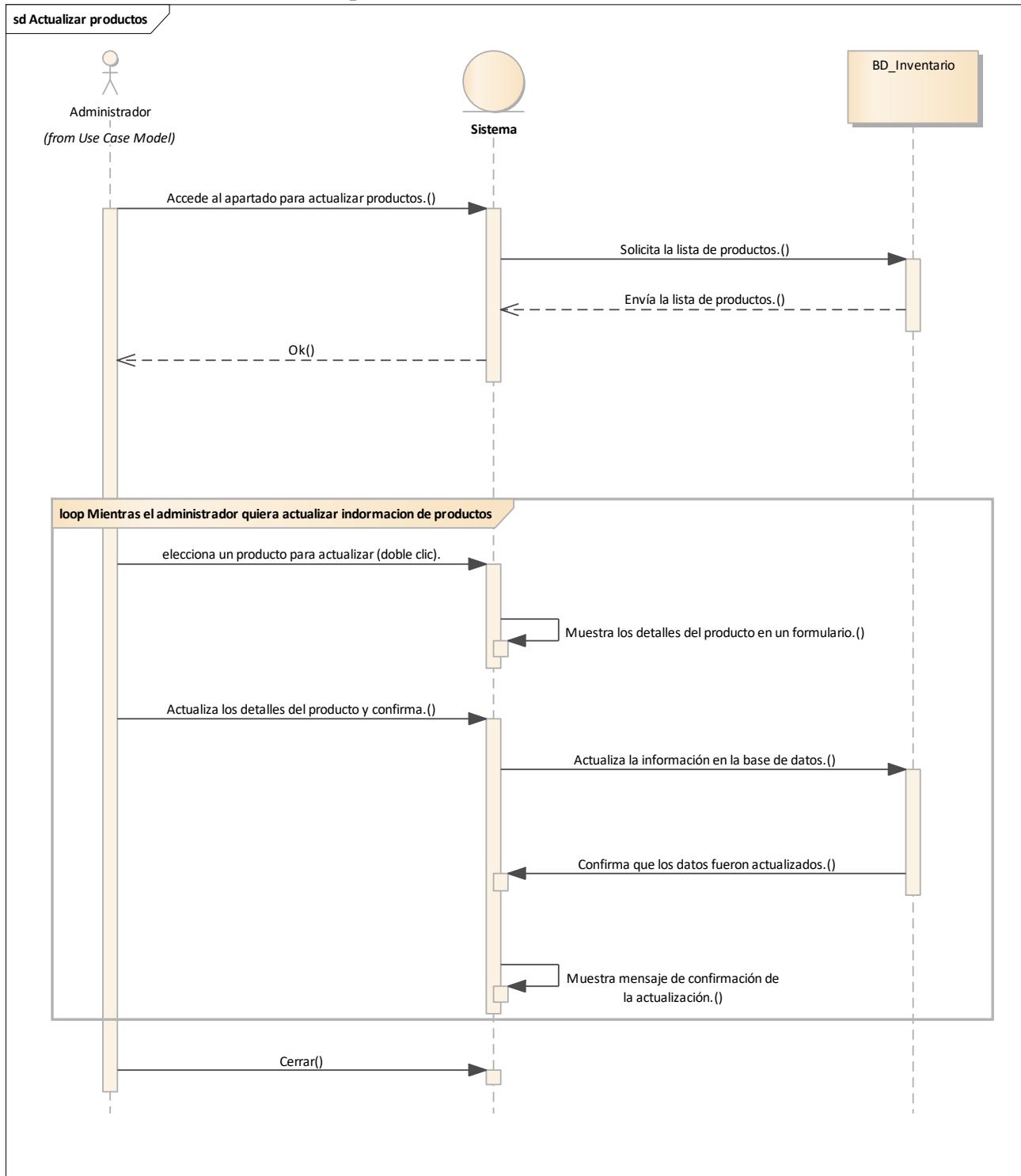
## 5.2.5 Ingresar Nuevo producto



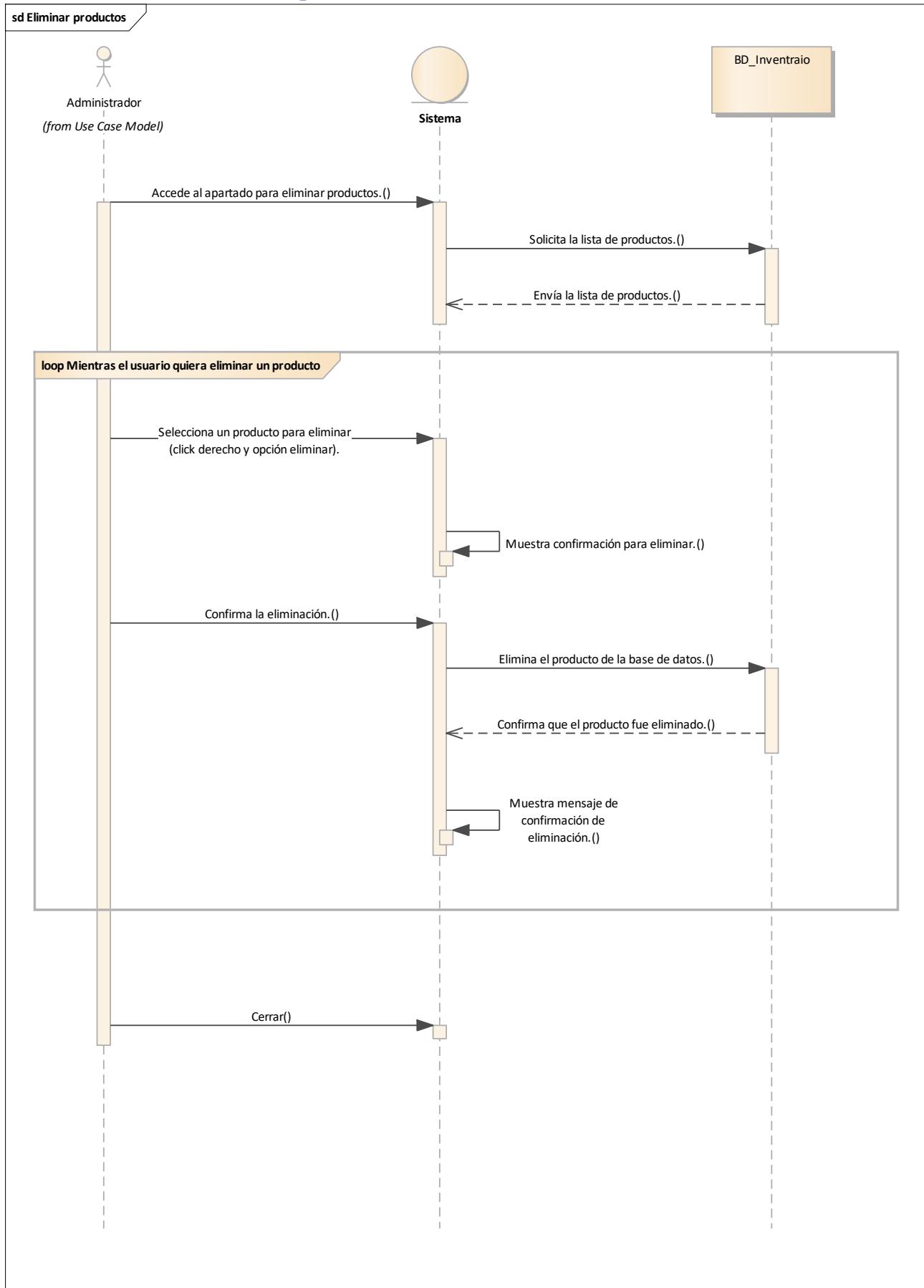
## 5.2.6 Consultar producto



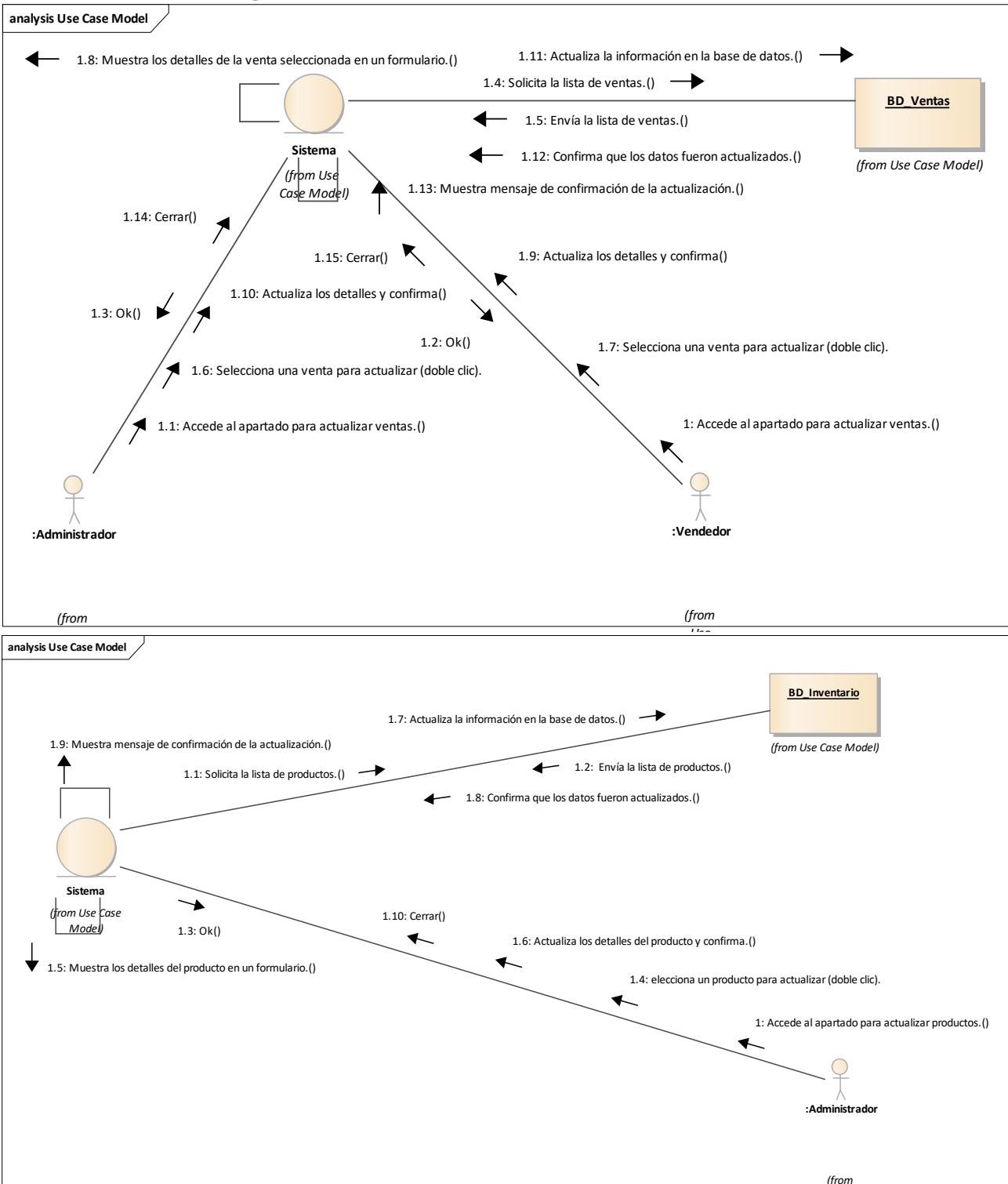
### 5.2.7 Actualizar producto



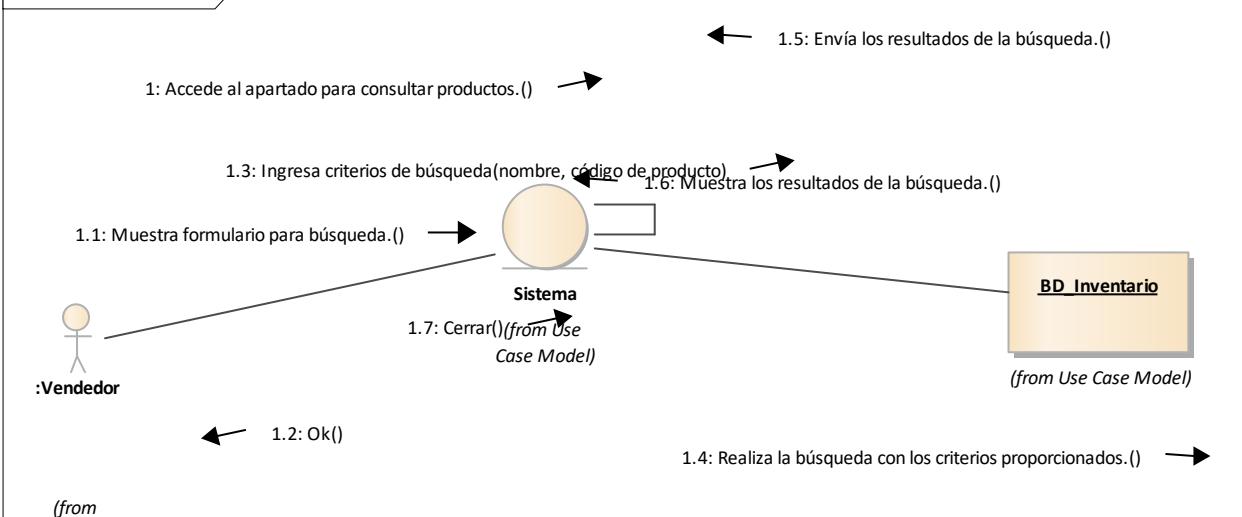
## 5.2.8 Eliminar producto



### 5.3. Diagrama de colaboración

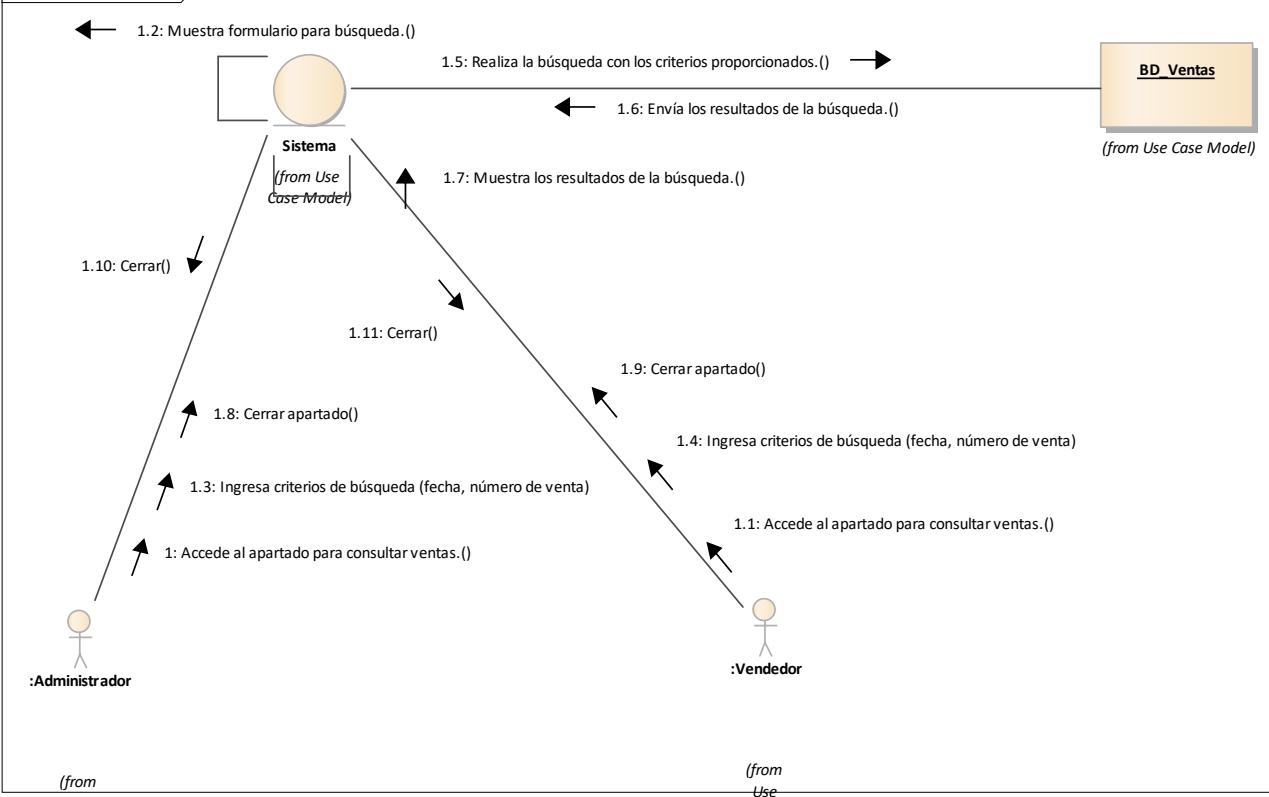


analysis Use Case Model



(from

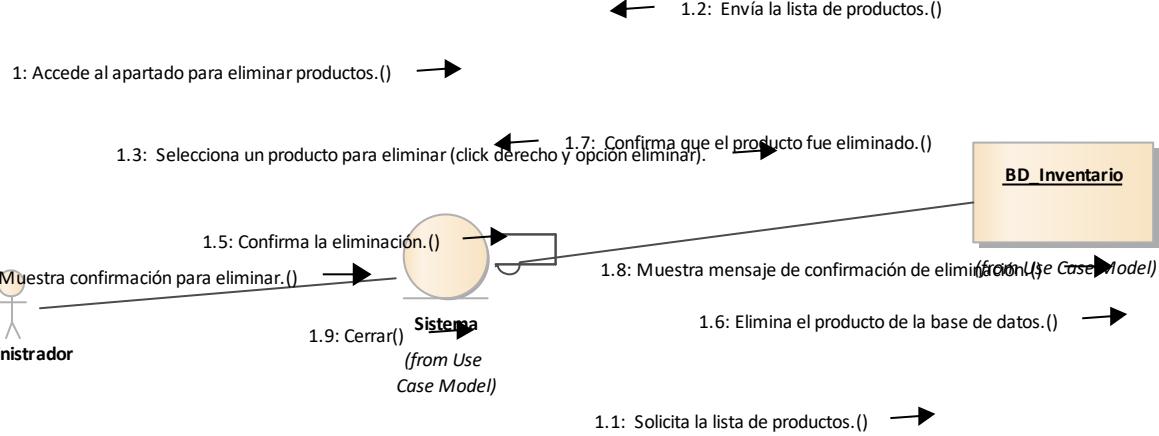
analysis Use Case Model



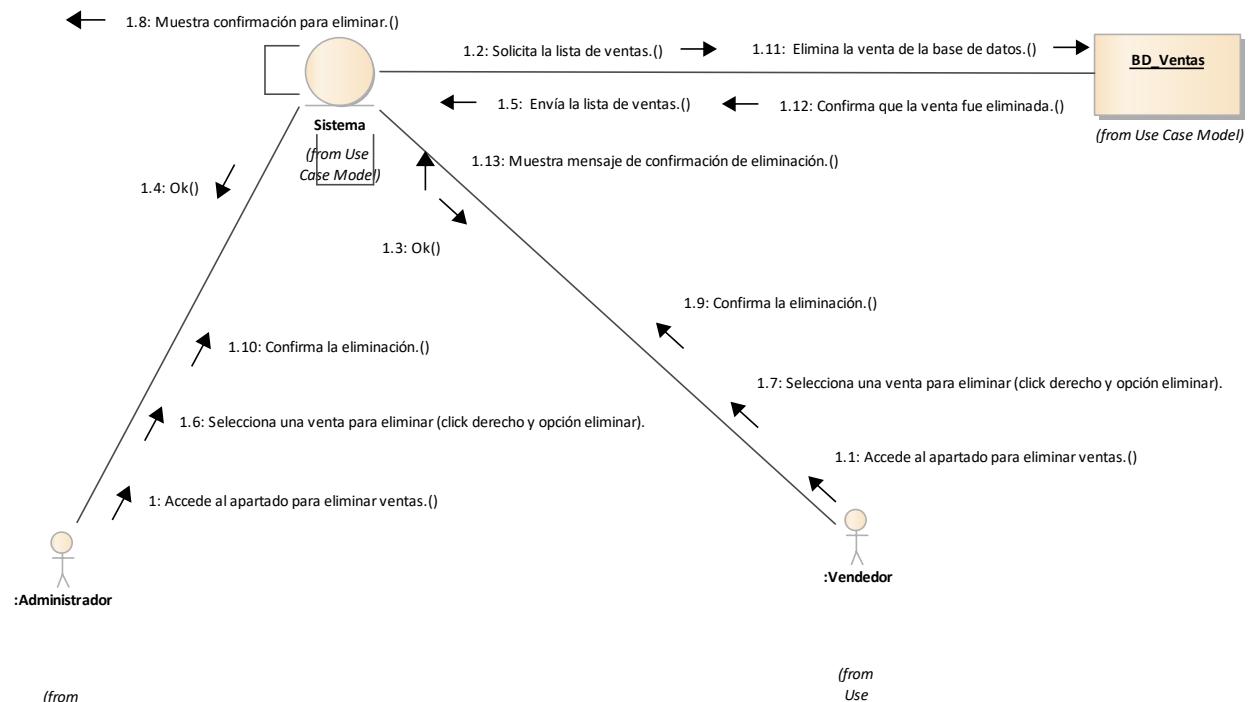
(from

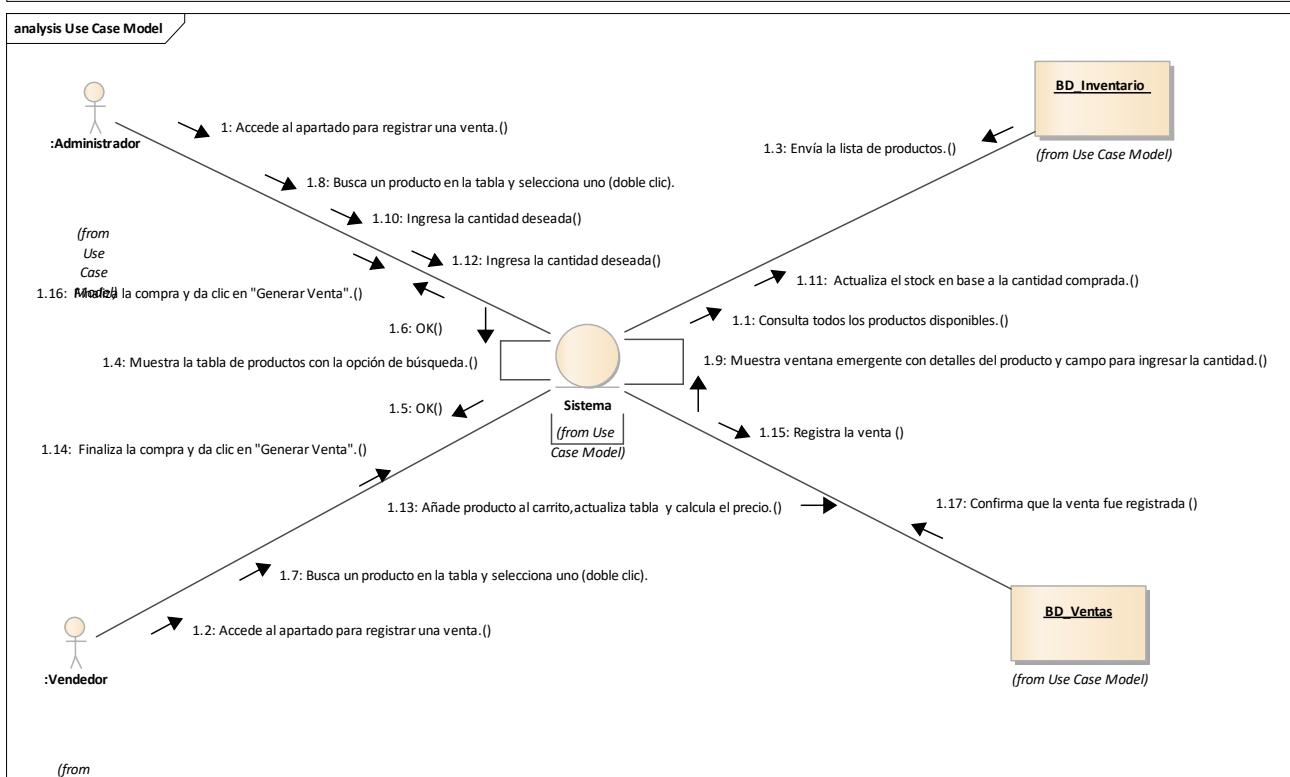
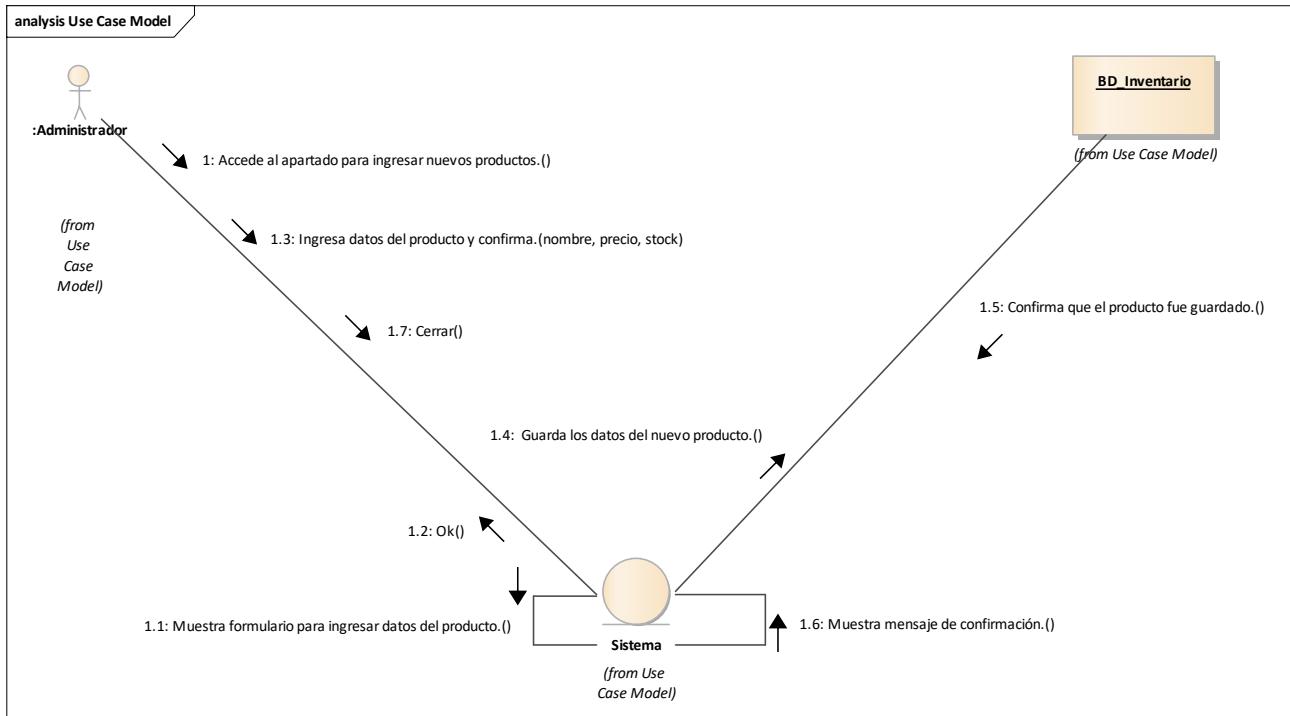
Use

analysis Use Case Model

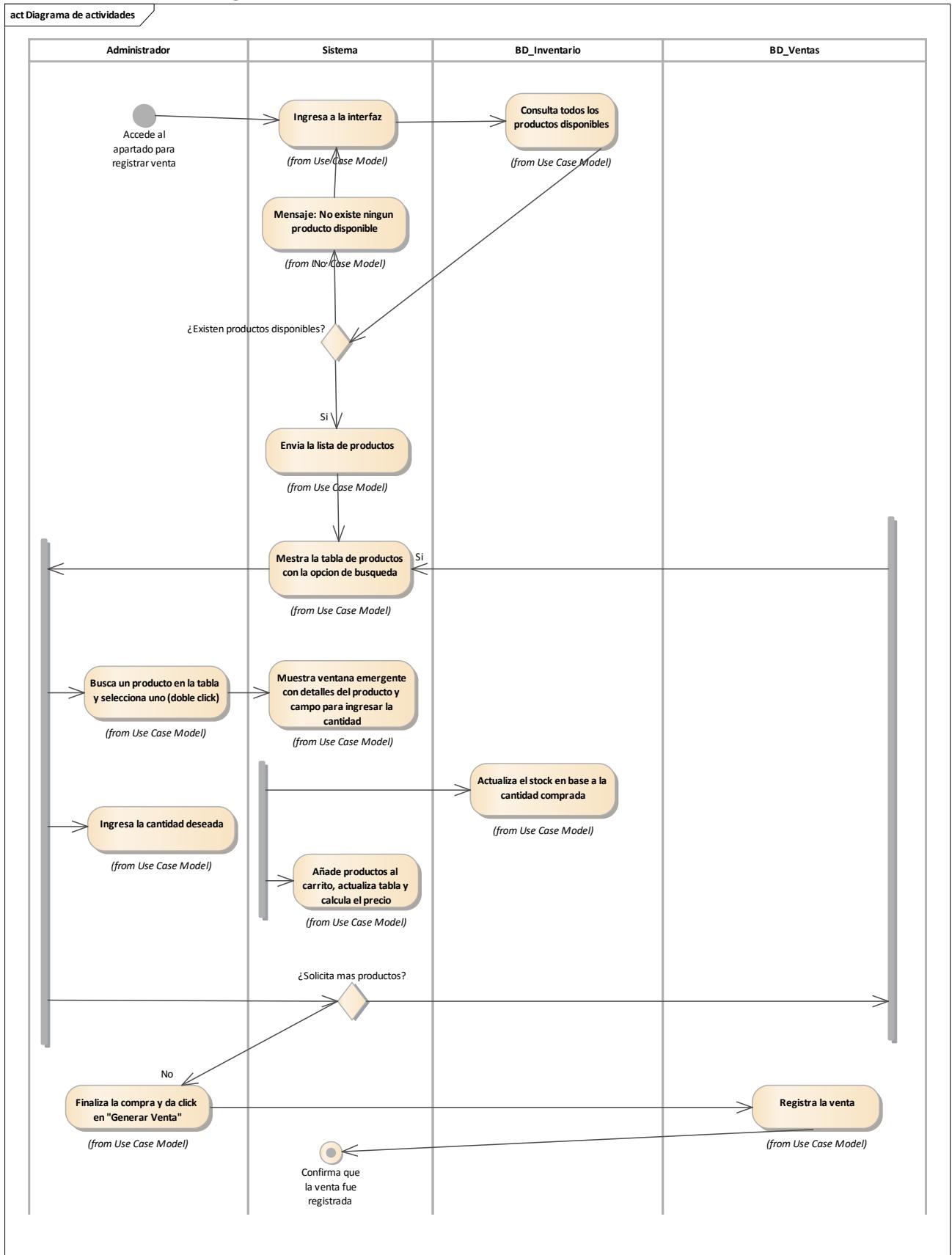


analysis Use Case Model





## 5.4 Diagrama de actividades



## 5.5 Formulario de entrada y salida de datos

Nombre de entrada/Salida: Consulta de precio de productos para venta

Contacto del usuario: Fabian Martínez

Tipo de archivo: Salida

Formato de archivo: Informe

Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Código de producto	10	B	Alfanumérico
Fecha de consulta	10	B	Formato dd/mm/aaaa
Hora de consulta	8	B	Formato hh:mm
Usuario que consulta	20	B	Texto libre

Nombre de entrada/Salida: Detalles de la transacción

Contacto del usuario: Fabian Martínez

Tipo de archivo: Salida

Formato de archivo: Informe

Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	12	B	Alfanumérico
Productos comprados	100	B	Lista de productos
Precio total	10	B	Número decimal
Fecha de transacción	10	B	Formato dd/mm/aaaa
Hora de transacción	8	B	Formato hh:mm
Usuario que transaccionó	20	B	Texto libre

Nombre de entrada/Salida: Detalles de la venta modificada

Contacto del usuario: Fabian Martínez

Tipo de archivo: entrante

Formato de archivo: Informe

Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción			Alfanumérico

	12	B	
Productos vendidos	100	B	Lista de productos
Cantidades modificadas	100	B	Lista de cantidades
Precio total modificado	10	B	Número decimal
Fecha de modificación	10	B	Formato dd/mm/aaaa
Usuario que modificó	20	B	Texto libre

Nombre de entrada/Salida: Criterios de búsqueda de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Entrada Formato de archivo: Pantalla Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID de la venta	10	B	Alfanumérico
Fecha de búsqueda	10	B	Formato dd/mm/aaaa
Usuario que busca	20	B	Texto libre

Nombre de entrada/Salida: Identificación de la venta a eliminar Contacto del usuario: Fabian Martínez Tipo de archivo: Entrada Formato de archivo: Pantalla Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID de la venta	10	B	Alfanumérico
Usuario que elimina	20	B	Texto libre

Nombre de entrada/Salida: Datos Login/Password vendedor Contacto del usuario: Fabian Martínez
--

Tipo de archivo: Entrada Formato de archivo: Pantalla Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Nombre de usuario	20	B	Texto alfanumérico
Contraseña	20	B	Texto oculto

Nombre de entrada/Salida: Confirmación de modificación de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	12	B	Alfanumérico
Estado de confirmación	20	B	Texto libre
Fecha de confirmación	10	B	Formato dd/mm/aaaa
Hora de confirmación	8	B	Formato hh:mm
Usuario que confirma	20	B	Texto libre

Nombre de entrada/Salida: Registro de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	12	B	Alfanumérico
Estado de registro	20	B	Texto libre
Fecha de registro	10	B	Formato dd/mm/aaaa
Hora de registro	8	B	Formato hh:mm
Usuario que registra	20	B	Texto libre

Nombre de entrada/Salida: Confirmación de eliminación de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): ID de la venta			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID de la venta	10	B	Alfanumérico
Estado de eliminación	20	B	Texto libre
Fecha de eliminación	10	B	Formato dd/mm/aaaa
Hora de eliminación	8	B	Formato hh:mm
Usuario que elimina	20	B	Texto libre

Nombre de entrada/Salida: Actualización de inventario después de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Producto vendido	30	B	Texto libre
Cantidad vendida	100	B	Número entero
Fecha de actualización	10	B	Formato dd/mm/aaaa
Hora de actualización	8	B	Formato hh:mm
Usuario que actualiza	20	B	Texto libre

Nombre de entrada/Salida: Actualización de inventario después de modificación de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe			
---	--	--	--

Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Producto modificado	30	B	Texto libre
Cantidad modificada	100	B	Número entero
Fecha de modificación	10	B	Formato dd/mm/aaaa
Hora de modificación	8	B	Formato hh:mm
Usuario que modifica	20	B	Texto libre

Nombre de entrada/Salida: Actualización de inventario después de eliminar venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Producto eliminado	30	B	Texto libre
Cantidad eliminada	100	B	Número entero
Fecha de eliminación	10	B	Formato dd/mm/aaaa
Hora de eliminación	8	B	Formato hh:mm
Usuario que elimina	20	B	Texto libre

Nombre de entrada/Salida: Consulta de producto.
---

Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Salida  
 Formato de archivo: Informe  
 Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Código de producto	10	B	Alfanumérico
Fecha de consulta	10	B	Formato dd/mm/aaaa
Hora de consulta	8	B	Formato hh:mm
Usuario que consulta	20	B	Texto libre

Nombre de entrada/Salida: Acceder al Módulo de Inventario con cuenta del administrador  
 Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Entrada  
 Formato de archivo: Pantalla  
 Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Nombre de usuario	20	B	Texto alfanumérico
Contraseña	20	B	Texto oculto

Nombre de entrada/Salida: Identificación del producto a eliminar  
 Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Entrada  
 Formato de archivo: Pantalla  
 Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
ID del producto	10	B	Alfanumérico
Usuario que elimina	20	B	Texto libre

Nombre de entrada/Salida: Detalles del producto modificado  
 Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Entrada

Formato de archivo: Informe Elemto(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID del producto	10	B	Alfanumérico
Usuario que modifica	20	B	Texto libre

Nombre de entrada/Salida: Detalles del nuevo producto. Contacto del usuario: Fabian Martínez Tipo de archivo: Entrada Formato de archivo: Informe Elemto(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Detalles del producto	Variable	B	Texto libre
Usuario que registra	20	B	Texto libre

Nombre de entrada/Salida: Datos Login/Pasword administrador. Contacto del usuario: Fabian Martínez Tipo de archivo: Entrada Formato de archivo: Pantalla Elemto(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Nombre de usuario	20	B	Texto alfanumérico
Contraseña	20	B	Texto oculto

Nombre de entrada/Salida: Confirmación de agregación de producto Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemto(s) Secuenciador(res): ID del producto			
Nombre del elemento	Longitud	B/D	Criterios de edición

ID del producto	10	B	Alfanumérico
Estado de confirmación	20	B	Texto libre
Fecha de confirmación	10	B	Formato dd/mm/aaaa
Hora de confirmación	8	B	Formato hh:mm
Usuario que confirma	20	B	Texto libre

Nombre de entrada/Salida: Confirmación de modificación de producto Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): ID del producto			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID del producto	10	B	Alfanumérico
Estado de confirmación	20	B	Texto libre
Fecha de confirmación	10	B	Formato dd/mm/aaaa
Hora de confirmación	8	B	Formato hh:mm
Usuario que confirma	20	B	Texto libre

Nombre de entrada/Salida: Confirmación de eliminación de producto Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): = ID del producto			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID del producto	10	B	Alfanumérico
Estado de confirmación	20	B	Texto libre
Fecha de confirmación	10	B	Formato dd/mm/aaaa
Hora de confirmación	8	B	Formato hh:mm
Usuario que confirma	20	B	Texto libre

Nombre de entrada/Salida: Información del productos			
Contacto del usuario: Fabian Martínez			
Tipo de archivo: Salida			
Formato de archivo: Informe			
Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Nombre del producto	30	B	Texto libre
Precio del producto	10	B	Número decimal

Nombre de entrada/Salida: Detalles de la venta			
Contacto del usuario: Fabian Martínez			
Tipo de archivo: Salida			
Formato de archivo: Informe			
Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Productos vendidos	Variable	B	Lista de productos
Cantidades vendidas	Variable	B	Lista de cantidades
Precio total	10	B	Número decimal

Nombre de entrada/Salida: Notificación de registro de venta			
Contacto del usuario: Fabian Martínez			
Tipo de archivo: Salida			
Formato de archivo: Informe			
Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	12	B	Alfanumérico
Estado de notificación	20	B	Texto libre
Fecha de notificación	10	B	Formato dd/mm/aaaa
Hora de notificación	8	B	Formato hh:mm
Usuario que notifica	20	B	Texto libre

Nombre de entrada/Salida: Interfaz del módulo de inventario
---

Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Salida  
 Formato de archivo: Pantalla  
 Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	Variable	B	Texto libre
Detalles del módulo	20	B	Texto libre

Nombre de entrada/Salida: Información del producto  
 Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Salida  
 Formato de archivo: Informe  
 Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Detalles del producto	100	B	Texto libre
Usuario que accede	20	B	Texto libre

Nombre de entrada/Salida: Notificación producto agregado  
 Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Salida  
 Formato de archivo: Informe  
 Elemento(s) Secuenciador(res): ID del producto agregado

Nombre del elemento	Longitud	B/D	Criterios de edición
ID del producto	10	B	Alfanumérico
Estado de notificación	20	B	Texto libre
Fecha de notificación	10	B	Formato dd/mm/aaaa
Hora de notificación	8	B	Formato hh:mm
Usuario que notifica	20	B	Texto libre

Nombre de entrada/Salida: Confirmación de ingreso con usuario administrador  
 Contacto del usuario: Fabian Martínez  
 Tipo de archivo: Salida  
 Formato de archivo: Informe  
 Elemento(s) Secuenciador(res): Ninguno

Nombre del elemento	Longitud	B/D	Criterios de edición
Nombre de usuario	20	B	Texto alfanumérico
Estado de confirmación	20	B	Texto libre
Fecha de confirmación	10	B	Formato dd/mm/aaaa
Hora de confirmación	8	B	Formato hh:mm

Nombre de entrada/Salida: Confirmación de ingreso con usuario vendedor Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID sesión	12	B	Numérico, sin caracteres especiales
Estado de confirmación	20	B	Texto libre
Fecha de confirmación	10	B	Formato dd/mm/aaaa
Hora de confirmación	8	B	Formato hh:mm
Horario que confirma	20	B	Texto Libre

Nombre de entrada/Salida: Recibo de venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	12	B	Numérico, sin caracteres especiales
Estado de recibo	20	B	Texto libre
Fecha de recibo	10	B	Formato dd/mm/aaaa

Hora de recibo	8	B	Formato hh:mm
Cliente destinatario	20	B	Texto libre

Nombre de entrada/Salida: Guardar la dirección del recibo en formato PDF Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Informe Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
Número de transacción	12	B	Numérico, sin caracteres especiales
Estado de recibo	20	B	Texto libre
Fecha de recibo	10	B	Formato dd/mm/aaaa
Hora de recibo	8	B	Formato hh:mm
Cliente destinatario	20	B	Texto libre

Nombre de entrada/Salida: Registrar Hora de Ingreso del Usuario Contacto del usuario: Fabian Martínez Tipo de archivo: Entrada Formato de archivo: Informe Elemento(s) Secuenciador(res): ID			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID	9	B	Numérico, sin caracteres especiales
Usuario	20	B	Texto libre
Hora de ingreso	10	B	Formato HH:MM

Nombre de entrada/Salida: Información del precio del producto para registrar Venta Contacto del usuario: Fabian Martínez Tipo de archivo: Salida Formato de archivo: Pantalla			
--	--	--	--

Elemento(s) Secuenciador(res): Ninguno			
Nombre del elemento	Longitud	B/D	Criterios de edición
ID del producto	12	B	Numérico, sin caracteres especiales
Precio del producto	10	B	Numérico, con dos decimales

### 5.3 Formulario de flujo de datos

ID: FD001

Nombre: Consulta de precio de productos para venta

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

Origen Vendedor	Destino Sistema
--------------------	--------------------

Tipo de flujo de datos

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto	Volumen/Tiempo 100/Dia
--	---------------------------

Comentarios: Facilita al vendedor la obtención rápida de precios para la venta.

ID: FD002

Nombre: Detalles de la transacción

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

Origen Vendedor	Destino Sistema
--------------------	--------------------

Tipo de flujo de datos

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo Detalles de la transacción = ID de la transacción + Productos comprados + Precio total	Volumen/Tiempo 100/Dia
---	---------------------------

Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

Comentarios: Registra todos los detalles necesarios para cada transacción realizada.

ID: FD003

Nombre: Detalles de la venta modificada

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

Origen Vendedor	Destino Sistema
--------------------	--------------------

Tipo de flujo de datos

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado	Volumen/Tiempo 10/Dia
--	--------------------------

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad	
Comentarios: Actualiza la información de ventas modificadas en el sistema.	

ID: FD004	
Nombre: Criterios de búsqueda de venta	
Descripción: El vendedor especifica criterios para buscar ventas en el sistema.	
Origen Vendedor	Destino Sistema
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda	Volumen/Tiempo 20/Dia
Comentarios: Permite al vendedor encontrar rápidamente la información de ventas específicas. Notificación de registro de venta	

ID: FD005	
Nombre: Identificación de la venta a eliminar	
Descripción: El vendedor identifica una venta específica para eliminarla del sistema.	
Origen Vendedor	Destino Sistema
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Identificación de la venta a eliminar = ID de la venta a eliminar	Volumen/Tiempo 5/Dia
Comentarios: Facilita la eliminación de ventas seleccionadas por el vendedor.	

ID: FD006	
Nombre: Datos Login/Password vendedor	
Descripción: Autenticación del vendedor para acceder al sistema.	
Origen Vendedor	Destino Sistema
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Datos Login/Password vendedor = Usuario + Contraseña	Volumen/Tiempo 50/Dia
Comentarios: Permite controlar el acceso al sistema por parte del vendedor.	

ID: FD007	
Nombre: Confirmación de modificación de venta	
Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.	
Origen Vendedor	Destino BD_Ventas
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario	<input checked="" type="checkbox"/> Interno
La estructura de datos que viaja con el flujo Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación	Volumen/Tiempo 10/Dia
Comentarios: Registra la acción de modificación en la base de datos de ventas.	

ID: FD008	
Nombre: Registro de venta	
Descripción: Registro de una nueva venta en la base de datos de ventas.	
Origen Vendedor	Destino BD_Ventas
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario	<input checked="" type="checkbox"/> Interno
La estructura de datos que viaja con el flujo Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago	Volumen/Tiempo 50/Dia
Comentarios: Registra la información completa de cada nueva venta en la base de datos.	

ID: FD009	
Nombre: Confirmación de eliminación de venta	
Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.	
Origen Vendedor	Destino BD_Ventas
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario	<input checked="" type="checkbox"/> Interno
La estructura de datos que viaja con el flujo Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación	Volumen/Tiempo 5/Dia
Comentarios: Registra la acción de eliminación en la base de datos de ventas.	

ID: FD010

Nombre: Actualización de inventario después de venta

Descripción: Actualización del inventario después de que se realiza una venta.

Origen Vendedor	Destino BD_Inventario
--------------------	--------------------------

Tipo de flujo de datos

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo  
Actualización de inventario después de venta = Productos vendidos  
+ Cantidades vendidas

Volumen/Tiempo  
50/Dia

Comentarios: Asegura que el inventario refleje las ventas realizadas.

ID: FD011

Nombre: Actualización de inventario después de modificación de venta

Descripción: Actualización del inventario después de que se modifica una venta.

Origen Vendedor	Destino BD_Inventario
--------------------	--------------------------

Tipo de flujo de datos

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo  
Actualización de inventario después de modificación de venta =  
Productos modificados + Nuevas cantidades

Volumen/Tiempo  
10/Dia

Comentarios: Ajusta el inventario según las modificaciones realizadas en las ventas.

ID: FD012

Nombre: Confirmación de agregación de producto

Descripción: Actualización del inventario después de que se elimina una venta.

Origen Vendedor	Destino BD_Inventario
--------------------	--------------------------

Tipo de flujo de datos

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo  
Actualización de inventario después de eliminar venta = Productos  
eliminados + Cantidades eliminadas

Volumen/Tiempo  
5/Dia

Comentarios: Ajusta el inventario según las ventas eliminadas.

ID: FD013

Nombre: Consulta de producto

Descripción: El administrador consulta la información detallada de un producto.

Origen Administrador	Destino Sistema
-------------------------	--------------------

Tipo de flujo de datos

Archivo  Pantalla Informe Formulario Interno

La estructura de datos que viaja con el flujo Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible	Volumen/Tiempo 20/Dia			
Comentarios: Permite al administrador obtener información precisa sobre productos.				

ID: FD014						
Nombre: Acceder al Módulo de Inventario con cuenta del administrador						
Descripción: Acceso del administrador al módulo de inventario utilizando sus credenciales.						
Origen Administrador						
Destino Sistema						
Tipo de flujo de datos						
<input type="checkbox"/> Archivo	<input checked="" type="checkbox"/> Pantalla	<input type="checkbox"/> Informe	<input type="checkbox"/>	Formulario	<input type="checkbox"/>	Interno
La estructura de datos que viaja con el flujo Acceder al Módulo de Inventario con cuenta del administrador = Usuario + Contraseña del administrador						
Volumen/Tiempo 10/Dia						
Comentarios: Controla el acceso seguro al módulo de inventario por parte del administrador.						

ID: FD015						
Nombre: Identificación del producto a eliminar						
Descripción: El administrador identifica un producto específico para eliminarlo del sistema.						
Origen Administrador						
Destino Sistema						
Tipo de flujo de datos						
<input type="checkbox"/> Archivo	<input checked="" type="checkbox"/> Pantalla	<input type="checkbox"/> Informe	<input type="checkbox"/>	Formulario	<input type="checkbox"/>	Interno
La estructura de datos que viaja con el flujo Identificación del producto a eliminar = ID del producto a eliminar						
Volumen/Tiempo 5/Dia						
Comentarios: Facilita la eliminación de productos seleccionados por el administrador.						

ID: FD016	
Nombre: Detalles del producto modificado	
Descripción: Detalles de un producto que ha sido modificado por el administrador.	
Origen Administrador	
Destino Sistema	

**Tipo de flujo de datos**

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo  
Detalles del producto modificado = ID del producto modificado +  
Nuevos detalles del producto

Volumen/Tiempo  
10/Dia

Comentarios: Actualiza la información del producto de acuerdo con las modificaciones realizadas por el administrador.

**ID: FD017**

Nombre: Detalles del nuevo producto.

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

Origen Administrador	Destino Sistema
-------------------------	--------------------

**Tipo de flujo de datos**

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo  
Detalles del nuevo producto = Nombre del nuevo producto +  
Descripción + Precio + Inventario inicial

Volumen/Tiempo  
5/Dia

Comentarios: Registra la información completa del nuevo producto en el sistema.

**ID: FD018**

Nombre: Datos Login/Password administrador

Descripción: Autenticación del administrador para acceder al sistema.

Origen Administrador	Destino Sistema
-------------------------	--------------------

**Tipo de flujo de datos**

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo  
Datos Login/Pasword administrador = Usuario + Contraseña

Volumen/Tiempo  
10/Dia

Comentarios: Permite controlar el acceso al sistema por parte del administrador.

**ID: FD019**

Nombre: Confirmación de agregación de producto

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

Origen Sistema	Destino BD_inventario
-------------------	--------------------------

**Tipo de flujo de datos**

Archivo  Pantalla  Informe  Formulario  Interno

La estructura de datos que viaja con el flujo

Volumen/Tiempo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales	10/Dia
Comentarios: Registra la acción de agregación en la base de datos de inventario.	

ID: FD020		
Nombre: Confirmación de modificación de producto		
Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.		
Origen Administrador	Destino BD_Inventario	
Tipo de flujo de datos		
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input checked="" type="checkbox"/> Interno		
La estructura de datos que viaja con el flujo Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación	Volumen/Tiempo 10/Dia	
Comentarios: Registra la acción de modificación en la base de datos de inventario.		

ID: FD021		
Nombre: Confirmación de eliminación de producto		
Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.		
Origen Administrador	Destino BD_Inventario	
Tipo de flujo de datos		
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input checked="" type="checkbox"/> Interno		
La estructura de datos que viaja con el flujo Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación	Volumen/Tiempo 10/Dia	
Comentarios: Registra la acción de eliminación en la base de datos de inventario.		

ID: FD022		
Nombre: Información de precio		
Descripción: Información detallada de productos y precio		
Origen Sistema	Destino Vendedor	
Tipo de flujo de datos		
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno		
La estructura de datos que viaja con el flujo Información de precio = Nombre del producto + Precio del producto + Descripción del producto	Volumen/Tiempo 100/Dia	
Comentarios: Proporciona al vendedor cuánto vale el producto		

ID: FD023		
Nombre: Detalles de la venta		

Descripción: Detalles de una venta específica realizada por el vendedor.	
Origen Sistema	Destino Vendedor
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad	
Comentarios: Informa al vendedor sobre los detalles específicos de cada venta realizada.	

ID: FD024	
Nombre: Notificación de registro de venta	
Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.	
Origen Sistema	
Destino Vendedor	
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Notificación de registro de venta = ID de la venta registrada + Fecha de registro	
Comentarios: Informa al vendedor sobre el estado de las ventas realizadas.	

ID: FD025	
Nombre: Interfaz del módulo de inventario	
Descripción: Interfaz que muestra el estado actualizado del inventario al administrador	
Origen Sistema	
Destino Administrador	
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input checked="" type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Interfaz del módulo de inventario = Listado de productos + Inventario disponible de cada producto	
Comentarios: Proporciona al administrador una vista completa y actualizada del inventario.	

ID: FD026	
Nombre: Información del producto	
Descripción: Información detallada de un producto específico para el administrador.	
Origen Sistema	
Destino Administrador	
Tipo de flujo de datos	

<input type="checkbox"/> Archivo	<input type="checkbox"/> Pantalla	<input type="checkbox"/> Informe	<input checked="" type="checkbox"/>	Formulario	<input type="checkbox"/>	I interno
La estructura de datos que viaja con el flujo Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible	Volumen/Tiempo 20/Dia					
Comentarios: Proporciona al administrador detalles precisos de productos específicos.						

ID: FD027						
Nombre: Notificación producto agregado						
Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.						
Origen Sistema	Destino Administrador					
Tipo de flujo de datos						
<input type="checkbox"/> Archivo	<input checked="" type="checkbox"/> Pantalla	<input type="checkbox"/> Informe	<input type="checkbox"/>	Formulario	<input type="checkbox"/>	I interno
La estructura de datos que viaja con el flujo Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación	Volumen/Tiempo 5/Dia					
Comentarios: Informa al administrador sobre el éxito de la acción de agregar productos.						

ID: FD028						
Nombre: Confirmación de ingreso con usuario administrador						
Descripción: Mensaje de confirmación del acceso exitoso al sistema por parte de un usuario administrador.						
Origen Sistema	Destino Administrador					
Tipo de flujo de datos						
<input type="checkbox"/> Archivo	<input type="checkbox"/> Pantalla	<input type="checkbox"/> Informe	<input type="checkbox"/>	Formulario	<input checked="" type="checkbox"/>	I interno
La estructura de datos que viaja con el flujo Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso	Volumen/Tiempo 10/Dia					
Comentarios: Notifica al Administrador el acceso exitoso al sistema.						

ID: FD029					
Nombre: Confirmación de ingreso con usuario vendedor					
Descripción: Mensaje de confirmación del acceso exitoso al sistema por parte de un usuario vendedor.					
Origen Sistema	Destino Vendedor				
Tipo de flujo de datos					
Archivo	Pantalla	Informe	Formulario	<input checked="" type="checkbox"/>	I interno

La estructura de datos que viaja con el flujo Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso	Volumen/Tiempo 50/Dia
Comentarios: Notifica al vendedor el acceso exitoso al sistema.	

ID: FD030	
Nombre: Recibo de venta	
Descripción: Recibo detallado de una venta para el cliente.	
Origen Sistema	Destino Clientes
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input checked="" type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración	Volumen/Tiempo 50/Dia
Comentarios: Proporciona al cliente un recibo detallado de la compra realizada.	

ID: FD031	
Nombre: Guardar la dirección del recibo en formato PDF	
Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.	
Origen Sistema	Destino BD_Recibo
Tipo de flujo de datos	
<input checked="" type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF	Volumen/Tiempo 50/Dia
Comentarios: Proporciona al cliente un recibo detallado de la compra realizada.	

ID: FD032	
Nombre: Registrar Hora de Ingreso del Usuario	
Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.	
Origen Sistema	Destino BD_Usuario
Tipo de flujo de datos	
<input type="checkbox"/> Archivo <input type="checkbox"/> Pantalla <input type="checkbox"/> Informe <input type="checkbox"/> Formulario <input checked="" type="checkbox"/> Interno	
La estructura de datos que viaja con el flujo Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso	Volumen/Tiempo 50/Dia
Comentarios: Registra la hora exacta en la que cada usuario accede al sistema.	

ID: FD033	
Nombre: Información del precio del producto para registrar Venta	
Descripción: Transferencia de información específica del producto (incluyendo el precio) desde la base de datos al proceso de registro de venta para completar la transacción.	
Origen Sistema –Consultar Producto	Destino Sistema –Registrar venta

Tipo de flujo de datos					
<input type="checkbox"/>	<input type="checkbox"/>	Archivo	<input type="checkbox"/>	Pantalla	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	Informe	<input type="checkbox"/>	Formulario	<input checked="" type="checkbox"/> Interno
La estructura de datos que viaja con el flujo Información del precio del producto para Registrar Venta = ID del producto + Precio del producto				Volumen/Tiempo 50/Dia	
Comentarios: Asegura que la información crucial del precio del producto esté disponible y sea precisa durante la transacción de venta.					

6 Diccionario de datos generado mediante el software

# Model Report

## Business Process Model

Version • Proposed



Date/Time Generated: 4/7/2024  
Author: Carolina



## Table of Contents

Business Process Model	6
Business Process Model diagram	6
Administrador	6
BD_Inventario	9
BD_Inventario	10
BD_Recibo	11
BD_Usuario	11
BD_Ventas	12
Clientes	13
Vendedor	13
3.- Gestionar Inventario	17
Proceso inventario diagram	20
Gestionar Inventario diagram	20
1.- Iniciar Sesión como Administrador	21
2.- Acceder al Módulo de Inventario	21
2..- Gestión del producto	21
3.- Agregar Nuevo Producto	22
3.1.- Acceder al Módulo de Inventario	22
3.1.- Gestión de productos	22
3.- Gestión de productos diagram	26
3.- Gestión de productos diagram	26
3.2.1.- Agregar Producto	27
3.2.2.- Consultar Producto	28
3.2.3.-Modificar Producto	29

3.2.4.-Eliminar Producto	29
Administrador	30
BD_Administrador	31
BD_Inventario	32
3.1.- Iniciar Sesión como Administrador	32
Administrador	33
Administrador	35
BD_Inventario	35
BD_Inventario	36
BD_resivo	36
BD_Usuario	37
BD_Usuario	37
Cliente	37
Proceso Venta	38
Proceso Venta diagram	40
Gestionar Ventas diagram	40
1.- Iniciar sesión como vendedor	41
2.- Consultar precio	41
2.1.- Consultar precio	42
2.1.- Iniciar sesión como vendedor	42
2.2- Gestionar Ventas	43
3- Gestionar Ventas diagram	46
Gestinar Ventas diagram	47
2.2.1.- Registrar Venta	48
2.2.2.- Consultar Venta	49
2.2.3.- Modificar Venta	50
2.2.4.- Eliminar Venta	51
BD_Inventario	52

BD_Ventas	52
DataStore1	53
Vendedor	53
2.3.- Generar Recibo	55
3.- Registrar Venta	56
4.- Generar Recibo	56
ImageAsset	56
BD_Inventario	56
BD_Inventario	57
BD_Recibo	57
BD_Resivo	58
BD_Usuarios	58
BD_Usuarios	58
BD_Ventas	58
BD_Ventas	59
Cliente	59
Cliente	60
DataStore2	60
Vendedor	60
Vendedor	60
Sistema de caja NortonSoft	63
Precesosos diagram	69
1.- Gestionar inicio de cuenta	70
1.- Gestionar inicio de cuenta diagram	71
1.- Gestionar inicio de cuenta diagram	72
1.1.-Verificar tipo de cuenta	72
Process1	74
BD_Usuarios	74

Vendedor	74
1.- Iniciar Sesión como Administrador	75
1.- Iniciar sesión como vendedor	75
2.- Acceder al Módulo de Inventario	76
2.- Consultar precio	76
2.- Gestionar ventas	76
3.- Agregar Nuevo Producto	80
3.- Registrar Venta	80
4.- Generar Recibo	81
a	81
Administrador	81
Adminitrador	82
b	84
BD_Inventario	84
BD_Inventario	85
BD_Inventario	85
BD_Recibo	86
BD_Resivo	87
BD_Usuario	87
BD_Usuario	88
BD_Usuarios	88
BD_Ventas	88
BD_Ventas	89
Cliente	90
Cliente	90
Vendedor	90
Vendedor	91

## Business Process Model

*Package in package 'Model'*

Business Process Model

Version Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## Business Process Model diagram

*Data Flow diagram in package 'Business Process Model'*

Business Process Model

Version 1.0

ASUS created on 8/6/2024. Last modified 2/7/2024

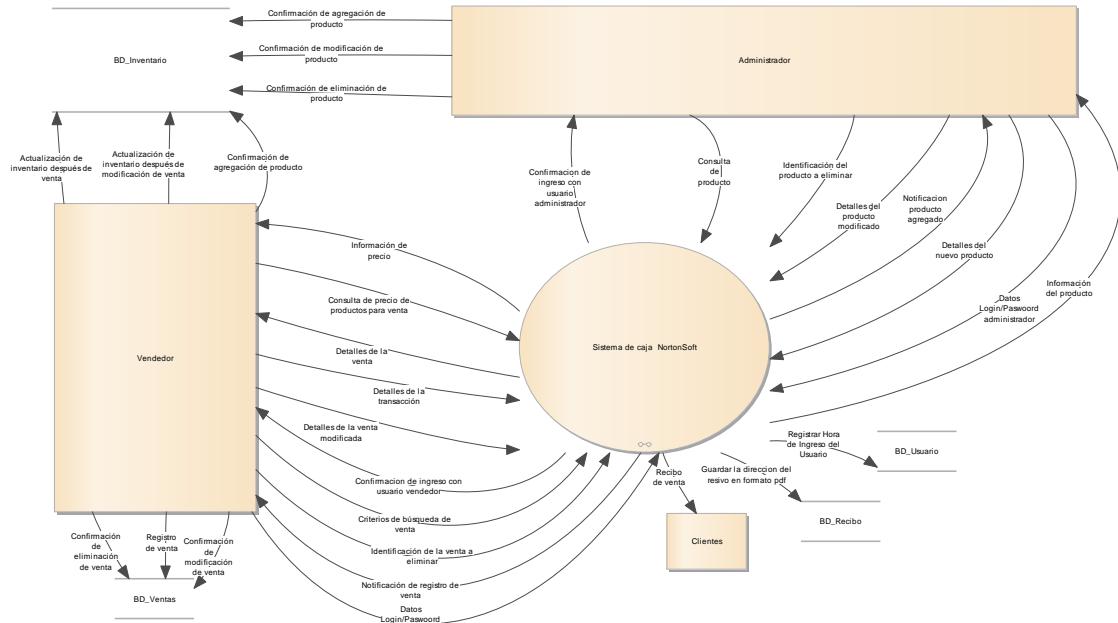


Figure 1: Business Process Model

## Administrador

*External «DFD\_External» in package 'Business Process Model'*

El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que

pueda surgir.

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

 **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto   Source -> Destination

From: Administrador : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

 **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto   Source -> Destination

From: Administrador : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de producto    Source -> Destination

From: Administrador : External, Public

To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord administrador    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Autenticación del administrador para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword administrador = Usuario + Contraseña

↗ **DataFlow** «DFD\_DataFlow» Consulta de producto    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario administrador   Source -> Destination

From: Sistema de caja NortonSoft : Process, Public

To: Administrador : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario administrador.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso

↗ **DataFlow** «DFD\_DataFlow» Información del producto   Source -> Destination

From: Sistema de caja NortonSoft : Process, Public

To: Administrador : External, Public

Descripción: Información detallada de un producto específico para el administrador.

a estructura de datos que viaja con el flujo

Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

↗ **DataFlow** «DFD\_DataFlow» Notificacion producto agregado   Source -> Destination

From: Sistema de caja NortonSoft : Process, Public

To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

## BD\_Inventario

*DataStore «DFD\_DataStore» in package 'Business Process Model'*

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de venta    Source -> Destination

From:    Vendedor : External, Public  
To:       BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de venta = Productos vendidos + Cantidad vendidas

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto    Source -> Destination

From:    Administrador : External, Public  
To:       BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

↗ **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto    Source -> Destination

From:    Administrador : External, Public  
To:       BD\_Inventario : DataStore, Public

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source ->

Destination

From: Vendedor : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de producto    Source -> Destination

From: Administrador : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

## BD\_Inventario

*DataStore «DFD\_DataStore» in package 'Business Process Model'*

BD\_Inventario

Version 1.0 Phase 1.0 Proposed  
J2D created on 12/6/2024. Last modified 12/6/2024

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Producto agregado   Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: BD\_Inventario : DataStore, Public

## BD\_Recibo

DataStore «DFD\_DataStore» in package 'Business Process Model'

Descripción: :Almacena los recibos generados para transacciones.

Estructura de datos ID\_Recibo (PK) + ID\_Venta (FK hacia Venta) + Direccion\_PDF

## BD\_Recibo

Version 1.0 Phase 1.0 Proposed  
Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Guardar la direccion del resivo en formato pdf   Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: BD\_Recibo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo

Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

## BD\_Usuario

DataStore «DFD\_DataStore» in package 'Business Process Model'

Descripción: Almacena la información básica de los usuarios del sistema, incluyendo la hora de ingreso.

Estructura de datos: ID (PK) + Usuario + Fecha\_Ingreso

BD\_Usuario

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

### CONNECTORS

 **DataFlow** «DFD\_DataFlow» Registrar Hora de Ingreso del Usuario    Source -> Destination  
From: Sistema de caja NortonSoft : Process, Public  
To: BD\_Usuario : DataStore, Public

Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.

La estructura de datos que viaja con el flujo :  
Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso

## BD\_Ventas

DataStore «DFD\_DataStore» in package 'Business Process Model'

Descripción: Registra todas las transacciones de venta realizadas.

Estructura de datos: ID\_Venta (PK) + Fecha + ID\_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio\_Total

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registro de venta    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

### ↗ DataFlow «DFD\_DataFlow» Confirmación de eliminación de venta    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

### ↗ DataFlow «DFD\_DataFlow» Confirmación de modificación de venta    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

## Clients

*External «DFD\_External» in package 'Business Process Model'*

El cliente, siendo el usuario final, interactúa directamente con el vendedor para llevar a cabo compras en la tienda. Aunque no tiene un contacto directo con el sistema

de gestión de ventas e inventario, sus acciones y necesidades influyen en cómo el vendedor utiliza dicho sistema. Entre las interacciones del cliente se incluyen: seleccionar productos para su compra y efectuar el pago en la caja, solicitar información detallada sobre productos específicos, recibir un recibo impreso al completar una transacción y, en caso de devoluciones, solicitar al vendedor el procesamiento correspondiente, lo que puede implicar ajustes en el inventario.

## Clientes

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 4/7/2024

### CONNECTORS

 **DataFlow** «DFD\_DataFlow» Recibo de venta   Source -> Destination

From: Sistema de caja NortonSoft : Process, Public

To: Clientes : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

## Vendedor

*External «DFD\_External» in package 'Business Process Model'*

El vendedor, también conocido como cajero, despliega un papel crucial como usuario principal del sistema de gestión de ventas e inventario. En el transcurso de su jornada laboral, interactúa directamente con la plataforma para llevar a cabo diversas funciones. Entre las responsabilidades del vendedor se encuentran: registrar ventas mediante el procesamiento de transacciones, consultar información sobre productos para ayudar a los clientes, generar recibos para las ventas realizadas, y actualizar el inventario disminuyendo las existencias de productos vendidos. Además, con la autorización del administrador, el vendedor puede acceder al módulo de inventario para añadir nuevos productos a la base de datos.

## Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

Alias Cajero

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de venta    Source -> Destination

From: Vendedor : External, Public

To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de venta = Productos vendidos + Cantidad vendidas

↗ **DataFlow** «DFD\_DataFlow» Registro de venta    Source -> Destination

From: Vendedor : External, Public

To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad vendida + Precio total + Forma de pago

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord vendedor    Source -> Destination

From: Vendedor : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Autenticación del vendedor para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword vendedor = Usuario + Contraseña

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Confirmación de agregación de producto    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

### ↗ DataFlow «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source ->

Destination

From: Vendedor : External, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

### ↗ DataFlow «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

La estructura de datos que viaja con el flujo

Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la transacción    Source -> Destination

From: Vendedor : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

La estructura de datos que viaja con el flujo

Detalles de la transacción = ID de la transacción + Productos comprados + Precio total  
Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado  
Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

### ↗ DataFlow «DFD\_DataFlow» Confirmación de eliminación de venta    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

### ↗ DataFlow «DFD\_DataFlow» Confirmación de modificación de venta    Source -> Destination

From: Vendedor : External, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Información de precio    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Información detallada de un producto específico para el administrador.

La estructura de datos que viaja con el flujo

Información de precio = Nombre del producto + Precio del producto + Descripción del producto

### ↗ DataFlow «DFD\_DataFlow» Notificación de registro de venta    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario vendedor    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public

To: Vendedor : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario vendedor.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso

## 3.- Gestionar Inventory

Process «DFD\_Process» in package 'Business Process Model'

Este proceso abarca todas las acciones relacionadas con la gestión de productos en el inventario, desde la adición de nuevos productos hasta la consulta, modificación y eliminación de productos existentes. Se asegura de que el inventario esté siempre actualizado y preciso.

### 3.- Gestionar Inventory

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 4/7/2024

## ELEMENTS OWNED BY 3.- Gestionar Inventory

■ 1.- Iniciar Sesión como Administrador : Process «DFD\_Process»

■ 2.- Acceder al Módulo de Inventory : Process «DFD\_Process»

■ 2.- Gestión del producto : Process «DFD\_Process»

■ 3.- Agregar Nuevo Producto : Process «DFD\_Process»

<b>ELEMENTS OWNED BY 3.- Gestionar Inventario</b>
■ 3.1.- Acceder al Módulo de Inventario : Process «DFD_Process»
■ 3.1.- Gestiar productos : Process «DFD_Process»
■ 3.1.- Iniciar Sesión como Administrador : Process «DFD_Process»
■ Administrador : External «DFD_External»
El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que pueda surgir.
■ Administrador : External «DFD_External»
■ BD_Inventario : DataStore «DFD_DataStore»
Descripción: Gestiona el inventario de productos.  Estructura de datos ID_Inventario (PK) + ID_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad_Disponible

ELEMENTS OWNED BY 3.- Gestionar Inventario	
■	BD_Inventario : DataStore «DFD_DataStore»
■	BD_resivo : DataStore «DFD_DataStore»
■	BD_Usuario : DataStore «DFD_DataStore»
■	BD_Usuario : DataStore «DFD_DataStore»
■	Cliente : External «DFD_External»

CONNECTORS	
↗	<b>DataFlow</b> «DFD_DataFlow» Confirmación de eliminación de producto    Source -> Destination From: 3.- Gestionar Inventario : Process, Public To: BD_Inventario : DataStore, Public
	Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.  La estructura de datos que viaja con el flujo Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación
↗	<b>DataFlow</b> «DFD_DataFlow» Confirmación de agregación de producto    Source -> Destination From: 3.- Gestionar Inventario : Process, Public To: BD_Inventario : DataStore, Public
	Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.  La estructura de datos que viaja con el flujo Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Información del producto    Source -> Destination

From: 3.- Gestionar inventario : Process, Public  
To: Administrador : External, Public

↗ **DataFlow** «DFD\_DataFlow» Notificación producto agregado    Source -> Destination

From: 3.- Gestionar inventario : Process, Public  
To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto    Source -> Destination

From: 3.- Gestionar inventario : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public  
To: 3.- Gestionar inventario : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

↗ **DataFlow** «DFD\_DataFlow» Consulta de producto    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## Proceso inventario diagram

*Data Flow diagram in package 'Business Process Model'*

Proceso inventario

Version 1.0

J2D created on 12/6/2024. Last modified 12/6/2024

Figure 2: Proceso inventario

Gestionar Inventario diagram

*Data Flow diagram in package 'Business Process Model'*

Gestionar Inventario

Version 1.0

J2D created on 12/6/2024. Last modified 4/7/2024

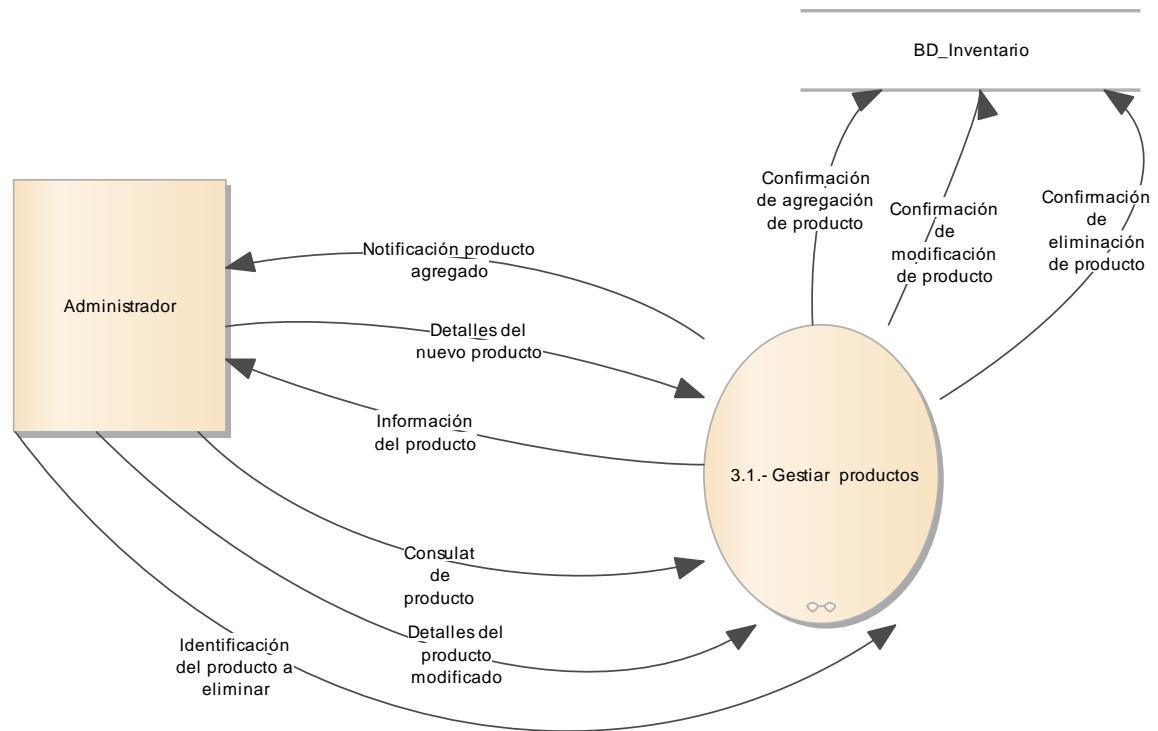


Figure 3: Gestionar Inventario

### 1.- Iniciar Sesión como Administrador

Process «DFD\_Process» owned by '3.- Gestionar Inventario', in package 'Business Process Model'

1.- Iniciar Sesión como Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## 2.- Acceder al Módulo de inventario

*Process «DFD\_Process» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

2.- Acceder al Módulo de inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## 2.- Gestión del producto

*Process «DFD\_Process» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

2.- Gestión del producto

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 12/6/2024

## 3.- Agregar Nuevo Producto

*Process «DFD\_Process» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

3.- Agregar Nuevo Producto

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

### 3.1.- Acceder al Módulo de inventario

*Process «DFD\_Process» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

3.1.- Acceder al Módulo de inventario

Version 1.0 Phase 1.0 Proposed

CONNECTORS
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Interfaz del módulo de inventario    Source -&gt; Destination</p> <p>From: 3.1.- Acceder al Módulo de Inventario : Process, Public To: Administrador : External, Public</p> <p>Descripción: Interfaz que muestra el estado actualizado del inventario al administrador</p> <p>La estructura de datos que viaja con el flujo Interfaz del módulo de inventario = Listado de productos + Inventario disponible de cada producto</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Acceder al Módulo de Inventario con cuenta del administrador    Source -&gt; Destination</p> <p>From: Administrador : External, Public To: 3.1.- Acceder al Módulo de Inventario : Process, Public</p> <p>Descripción: Acceso del administrador al módulo de inventario utilizando sus credenciales.</p> <p>La estructura de datos que viaja con el flujo Acceder al Módulo de Inventario con cuenta del administrador = Usuario + Contraseña del administrador</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Acceder al Módulo de Inventario con cuenta del administrador    Source -&gt; Destination</p> <p>From: BD_Usuario : DataStore, Public To: 3.1.- Acceder al Módulo de Inventario : Process, Public</p>

### 3.1.- Gestiar productos

Process «DFD\_Process» owned by '3.- Gestionar Inventario', in package 'Business Process Model'

3.1.- Gestiar productos

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 2/7/2024

## ELEMENTS OWNED BY 3.1.- Gestiar productos

### ■ 3.2.1.- Agregar Producto : Process «DFD\_Process»

Permite al administrador agregar nuevos productos al inventario. Incluye la entrada de datos del producto como nombre, descripción, precio y cantidad. Asegura que los nuevos productos se registren correctamente en la base de datos de inventario.

### ■ 3.2.2.- Consultar Producto : Process «DFD\_Process»

Permite al usuario buscar y visualizar detalles de productos existentes en el inventario. Proporciona información detallada sobre los productos para facilitar la gestión del inventario y la toma de decisiones.

### ■ 3.2.3.-Modificar Producto : Process «DFD\_Process»

Permite al administrador modificar detalles de productos ya registrados en el inventario, como nombre, descripción, precio y cantidad. Asegura que cualquier cambio en los productos se refleje correctamente en la base de datos de inventario.

### ■ 3.2.4.-Eliminar Producto : Process «DFD\_Process»

Permite al administrador eliminar productos del inventario. Este proceso actualiza la base de datos de inventario para reflejar la eliminación de los productos y asegura que el inventario esté siempre actualizado.

## ELEMENTS OWNED BY 3.1.- Gestiar productos

### ■ Administrador : External «DFD\_External»

El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que pueda surgir.

### ■ BD\_Administrador : DataStore «DFD\_DataStore»

### ■ BD\_Inventario : DataStore «DFD\_DataStore»

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Confirmación de agregación de producto    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

### ↗ DataFlow «DFD\_DataFlow» Confirmación de modificación de producto    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

### ↗ DataFlow «DFD\_DataFlow» Información del producto    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: Administrador : External, Public

Descripción: Información detallada de un producto específico para el administrador.

a estructura de datos que viaja con el flujo

Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Notificación producto agregado    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

### ↗ DataFlow «DFD\_DataFlow» Confirmación de eliminación de producto    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

### ↗ DataFlow «DFD\_DataFlow» Guardar la dirección del resivo en formato pdf    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_resivo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo

Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Recibo de venta    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: Cliente : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

### ↗ DataFlow «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public  
To: 3.1.- Gestiar productos : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

### ↗ DataFlow «DFD\_DataFlow» Consulta de producto    Source -> Destination

From: Administrador : External, Public  
To: 3.1.- Gestiar productos : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Gestiar productos : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Gestiar productos : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

## 3.- Gestión de productos diagram

*Data Flow diagram in package 'Business Process Model'*

3.- Gestión de productos

Version 1.0

J2D created on 12/6/2024. Last modified 12/6/2024

Figure 4: 3.- Gestión de productos

### 3.- Gestión de productos diagram

Data Flow diagram in package 'Business Process Model'

3.- Gestión de productos

Version 1.0

J2D created on 12/6/2024. Last modified 1/7/2024

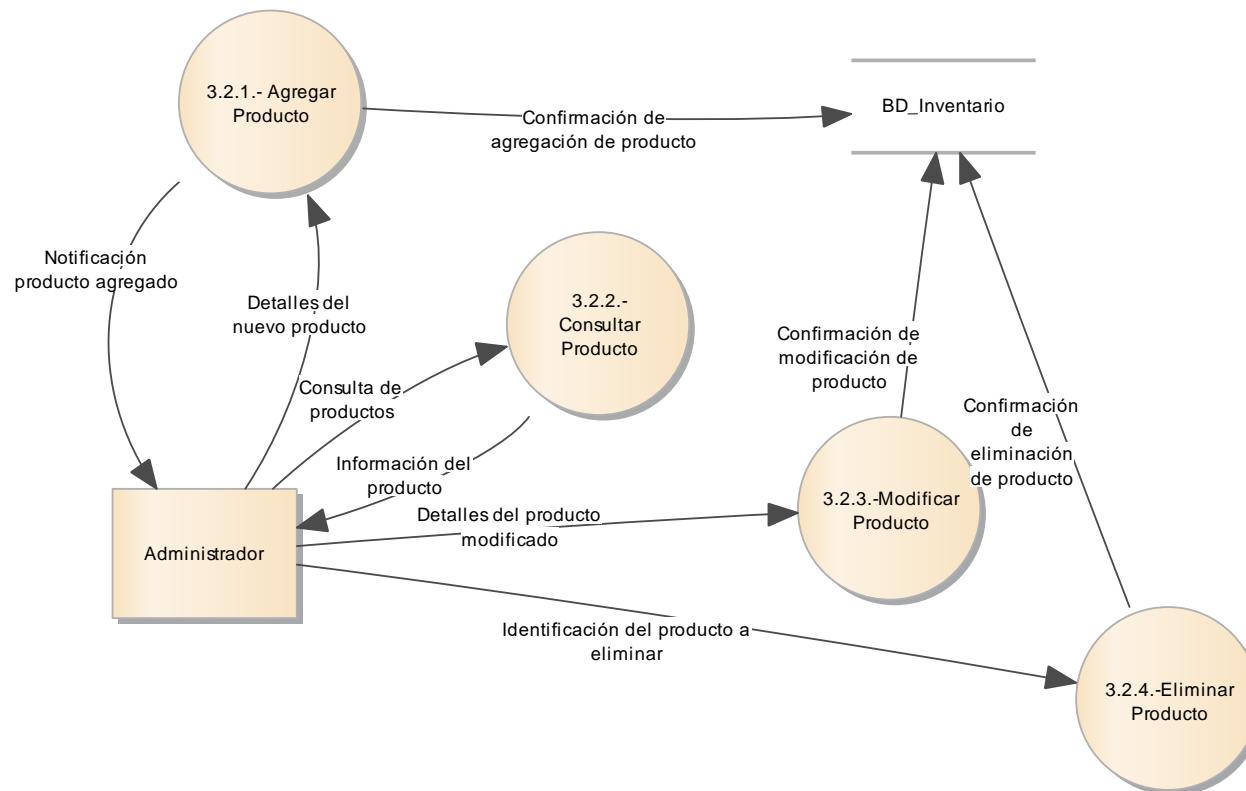


Figure 5: 3.- Gestión de productos

### **3.2.1.- Agregar Producto**

*Process «DFD\_Process» owned by '3.1.- Gestiar productos', in package 'Business Process Model'*

Permite al administrador agregar nuevos productos al inventario. Incluye la entrada de datos del producto como nombre, descripción, precio y cantidad. Asegura que los nuevos productos se registren correctamente en la base de datos de inventario.

#### 3.2.1.- Agregar Producto

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 4/7/2024

CONNECTORS
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Confirmación de agregación de producto    Source -&gt; Destination</p> <p>From: 3.2.1.- Agregar Producto : Process, Public To: BD_Inventario : DataStore, Public</p> <p>Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.</p> <p>La estructura de datos que viaja con el flujo Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Notificación producto agregado    Source -&gt; Destination</p> <p>From: 3.2.1.- Agregar Producto : Process, Public To: Administrador : External, Public</p> <p>Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.</p> <p>La estructura de datos que viaja con el flujo Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación</p>

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Administrador : External, Public

To: 3.2.1.- Agregar Producto : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

### 3.2.2.- Consultar Producto

Process «DFD\_Process» owned by '3.1.- Gestiar productos', in package 'Business Process Model'

Permite al usuario buscar y visualizar detalles de productos existentes en el inventario. Proporciona información detallada sobre los productos para facilitar la gestión del inventario y la toma de decisiones.

3.2.2.- Consultar Producto

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Información del producto    Source -> Destination

From: 3.2.2.- Consultar Producto : Process, Public

To: Administrador : External, Public

Descripción: Información detallada de un producto específico para el administrador.

a estructura de datos que viaja con el flujo

Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Consulta de productos    Source -> Destination

From: Administrador : External, Public

To: 3.2.2.- Consultar Producto : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

### 3.2.3.-Modificar Producto

Process «DFD\_Process» owned by '3.1.- Gestiar productos', in package 'Business Process Model'

Permite al administrador modificar detalles de productos ya registrados en el inventario, como nombre, descripción, precio y cantidad. Asegura que cualquier cambio en los productos se refleje correctamente en la base de datos de inventario.

3.2.3.-Modificar Producto

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto    Source -> Destination

From: 3.2.3.-Modificar Producto : Process, Public

To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public

To: 3.2.3.-Modificar Producto : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

### 3.2.4.-Eliminar Producto

Process «DFD\_Process» owned by '3.1.- Gestiar productos', in package 'Business Process Model'

Permite al administrador eliminar productos del inventario. Este proceso actualiza la base de datos de inventario para reflejar la eliminación de los productos y asegura que el inventario esté siempre actualizado.

#### 3.2.4.-Eliminar Producto

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de producto    Source -> Destination

From: 3.2.4.-Eliminar Producto : Process, Public

To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Administrador : External, Public

To: 3.2.4.-Eliminar Producto : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

## Administrador

External «DFD\_External» owned by '3.1.- Gestiar productos', in package 'Business Process Model'

El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que pueda surgir.

Administrator

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Administrador : External, Public

To: 3.2.1.- Agregar Producto : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Administrador : External, Public

To: 3.2.4.-Eliminar Producto : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

↗ **DataFlow** «DFD\_DataFlow» Consulta de productos    Source -> Destination

From: Administrador : External, Public

To: 3.2.2.- Consultar Producto : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public

To: 3.2.3.-Modificar Producto : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

↗ **DataFlow** «DFD\_DataFlow» Notificación producto agregado    Source -> Destination

From: 3.2.1.- Agregar Producto : Process, Public

To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

↗ **DataFlow** «DFD\_DataFlow» Información del producto    Source -> Destination

From: 3.2.2.- Consultar Producto : Process, Public

To: Administrador : External, Public

Descripción: Información detallada de un producto específico para el administrador.

a estructura de datos que viaja con el flujo

Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## BD\_Administrador

*DataStore «DFD\_DataStore» owned by '3.1.- Gestiar productos', in package 'Business Process Model'*

BD\_Administrador

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 12/6/2024

## **BD\_Inventario**

DataStore «DFD\_DataStore» owned by '3.1.- Gestiar productos', in package 'Business Process Model'

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 2/7/2024

### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de producto    Source -> Destination

From:    3.2.4.-Eliminar Producto : Process, Public

To:       BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

↗ **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto    Source -> Destination

From:    3.2.1.- Agregar Producto : Process, Public

To:       BD\_Inventario : DataStore, Public

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto    Source -> Destination

From: 3.2.3.-Modificar Producto : Process, Public

To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

### 3.1.- Iniciar Sesión como Administrador

*Process «DFD\_Process» owned by '3.- Gestionar Inventario', in package 'Business Process Model'*

3.1.- Iniciar Sesión como Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 19/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de inicio de sesión como administrador    Source -> Destination

From: 3.1.- Iniciar Sesión como Administrador : Process, Public

To: BD\_Usuario : DataStore, Public

↗ **DataFlow** «DFD\_DataFlow» Credenciales de administrador (nombre de usuario y contraseña)    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Iniciar Sesión como Administrador : Process, Public

## Administrador

External «DFD\_External» owned by '3.- Gestionar inventario', in package 'Business Process Model'

El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que pueda surgir.

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 4/7/2024

### CONNECTORS

 **DataFlow** «DFD\_DataFlow» Acceder al Módulo de Inventario con cuenta del administrador      Source ->  
Destination

From: Administrador : External, Public  
To: 3.1.- Acceder al Módulo de Inventario : Process, Public

Descripción: Acceso del administrador al módulo de inventario utilizando sus credenciales.

La estructura de datos que viaja con el flujo  
Acceder al Módulo de Inventario con cuenta del administrador = Usuario + Contraseña del administrador

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Gestiar productos : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

↗ **DataFlow** «DFD\_DataFlow» Consulta de producto    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Gestiar productos : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Gestiar productos : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

↗ **DataFlow** «DFD\_DataFlow» Credenciales de administrador (nombre de usuario y contraseña)    Source ->

Destination

From: Administrador : External, Public

To: 3.1.- Iniciar Sesión como Administrador : Process, Public

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Administrador : External, Public

To: 3.1.- Gestiar productos : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

↗ **DataFlow** «DFD\_DataFlow» Información del producto    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public

To: Administrador : External, Public

Descripción: Información detallada de un producto específico para el administrador.

a estructura de datos que viaja con el flujo

Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

↗ **DataFlow** «DFD\_DataFlow» Notificación producto agregado    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public

To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Interfaz del módulo de inventario    Source -> Destination

From: 3.1.- Acceder al Módulo de Inventario : Process, Public

To: Administrador : External, Public

Descripción: Interfaz que muestra el estado actualizado del inventario al administrador

La estructura de datos que viaja con el flujo

Interfaz del módulo de inventario = Listado de productos + Inventario disponible de cada producto

## Administrador

*External «DFD\_External» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## BD\_Inventario

*DataStore «DFD\_DataStore» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto   Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto   Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de producto   Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

## BD\_Inventario

*DataStore «DFD\_DataStore» owned by '3.- Gestionar Inventario', in package 'Business Process Model'*

BD\_Inventario

Version 1.0 Phase 1.0 Proposed  
ASUS created on 12/6/2024. Last modified 8/6/2024

### BD\_resivo

*DataStore «DFD\_DataStore» owned by '3.- Gestionar Inventario', in package 'Business Process Model'*

#### BD\_resivo

Version 1.0 Phase 1.0 Proposed  
ASUS created on 21/6/2024. Last modified 21/6/2024

#### CONNECTORS

 **DataFlow** «DFD\_DataFlow» Guardar la direccion del resivo en formato pdf   Source -> Destination  
From: 3.1.- Gestiar productos : Process, Public  
To: BD\_resivo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo  
Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

### BD\_Usuario

*DataStore «DFD\_DataStore» owned by '3.- Gestionar Inventario', in package 'Business Process Model'*

#### BD\_Usuario

Version 1.0 Phase 1.0 Proposed  
ASUS created on 12/6/2024. Last modified 8/6/2024

### BD\_Usuario

*DataStore «DFD\_DataStore» owned by '3.- Gestionar Inventario', in package 'Business Process Model'*

BD\_Usuario

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Acceder al Módulo de inventario con cuenta del administrador    Source -> Destination

From: BD\_Usuario : DataStore, Public  
To: 3.1.- Acceder al Módulo de inventario : Process, Public

↗ **DataFlow** «DFD\_DataFlow» Confirmación de inicio de sesión como administrador    Source -> Destination

From: 3.1.- Iniciar Sesión como Administrador : Process, Public  
To: BD\_Usuario : DataStore, Public

## Cliente

*External «DFD\_External» owned by '3.- Gestionar inventario', in package 'Business Process Model'*

Cliente

Version 1.0 Phase 1.0 Proposed

ASUS created on 21/6/2024. Last modified 21/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Recibo de venta    Source -> Destination

From: 3.1.- Gestiar productos : Process, Public  
To: Cliente : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

## Proceso Venta

Process «DFD\_Process» in package 'Business Process Model'

fdsvdsv

Proceso Venta

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 12/6/2024

ELEMENTS OWNED BY Proceso Venta	
■	1.- Iniciar sesión como vendedor : Process «DFD_Process»
■	2.- Consultar precio : Process «DFD_Process»
■	2.1.- Consultar precio : Process «DFD_Process»  Este proceso permite al usuario, ya sea un vendedor o administrador, consultar el precio de los productos disponibles en el inventario. Proporciona información detallada sobre los productos, incluyendo el precio, para facilitar la venta y gestión de inventario.
■	2.1.- Iniciar sesión como vendedor : Process «DFD_Process»
■	2.2- Gestionar Ventas : Process «DFD_Process»  Este proceso abarca todas las acciones relacionadas con la gestión de ventas, desde el registro de nuevas ventas hasta la modificación y eliminación de ventas existentes. Se asegura de que todas las transacciones sean precisas y estén actualizadas en el sistema.

ELEMENTS OWNED BY Proceso Venta	
■ 2.3.- Generar Recibo : Process «DFD_Process»	Este proceso genera y emite un recibo detallado para el cliente después de completar una venta. Incluye la creación del recibo en formato PDF, que se almacena en la base de datos de recibos para referencia futura.
■ 3.- Registrar Venta : Process «DFD_Process»	
■ 4.- Generar Recibo : Process «DFD_Process»	
■ ImageAsset : Artifact «Image»	
■ BD_Inventario : DataStore «DFD_DataStore»	<p>Descripción: Gestiona el inventario de productos.</p> <p>Estructura de datos ID_Inventario (PK) + ID_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad_Disponible</p>
■ BD_Inventario : DataStore «DFD_DataStore»	
■ BD_Recibo : DataStore «DFD_DataStore»	<p>Descripción: Almacena los recibos generados para transacciones.</p> <p>Estructura de datos ID_Recibo (PK) + ID_Venta (FK hacia Venta) + Direccion_PDF</p>
■ BD_Resivo : DataStore «DFD_DataStore»	

ELEMENTS OWNED BY Proceso Venta	
■ BD_Usuarios : DataStore «DFD_DataStore»	
■ BD_Usuarios : DataStore «DFD_DataStore»	
■ BD_Ventas : DataStore «DFD_DataStore»	<p>Descripción: Registra todas las transacciones de venta realizadas.</p> <p>Estructura de datos: ID_Venta (PK) + Fecha + ID_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio_Total</p>
■ BD_Ventas : DataStore «DFD_DataStore»	
■ Cliente : External «DFD_External»	
■ Cliente : External «DFD_External»	<p>El cliente, siendo el usuario final, interactúa directamente con el vendedor para llevar a cabo compras en la tienda. Aunque no tiene un contacto directo con el sistema de gestión de ventas e inventario, sus acciones y necesidades influyen en cómo el vendedor utiliza dicho sistema. Entre las interacciones del cliente se incluyen: seleccionar productos para su compra y efectuar el pago en la caja, solicitar información detallada sobre productos específicos, recibir un recibo impreso al completar una transacción y, en caso de devoluciones, solicitar al vendedor el procesamiento correspondiente, lo que puede implicar ajustes en el inventario.</p>
■ DataStore2 : DataStore «DFD_DataStore»	
■ Vendedor : External «DFD_External»	

## ELEMENTS OWNED BY Proceso Venta

### ■ Vendedor : External <DFD\_External>

El vendedor, también conocido como cajero, despliega un papel crucial como usuario principal del sistema de gestión de ventas e inventario. En el transcurso de su jornada laboral, interactúa directamente con la plataforma para llevar a cabo diversas funciones. Entre las responsabilidades del vendedor se encuentran: registrar ventas mediante el procesamiento de transacciones, consultar información sobre productos para ayudar a los clientes, generar recibos para las ventas realizadas, y actualizar el inventario disminuyendo las existencias de productos vendidos. Además, con la autorización del administrador, el vendedor puede acceder al módulo de inventario para añadir nuevos productos a la base de datos.

## Proceso Venta diagram

*Data Flow diagram in package 'Business Process Model'*

Proceso Venta

Version 1.0

J2D created on 12/6/2024. Last modified 12/6/2024

Figure 6: Proceso Venta

## Gestionar Ventas diagram

*Data Flow diagram in package 'Business Process Model'*

Gestionar Ventas

Version 1.0

J2D created on 12/6/2024. Last modified 2/7/2024

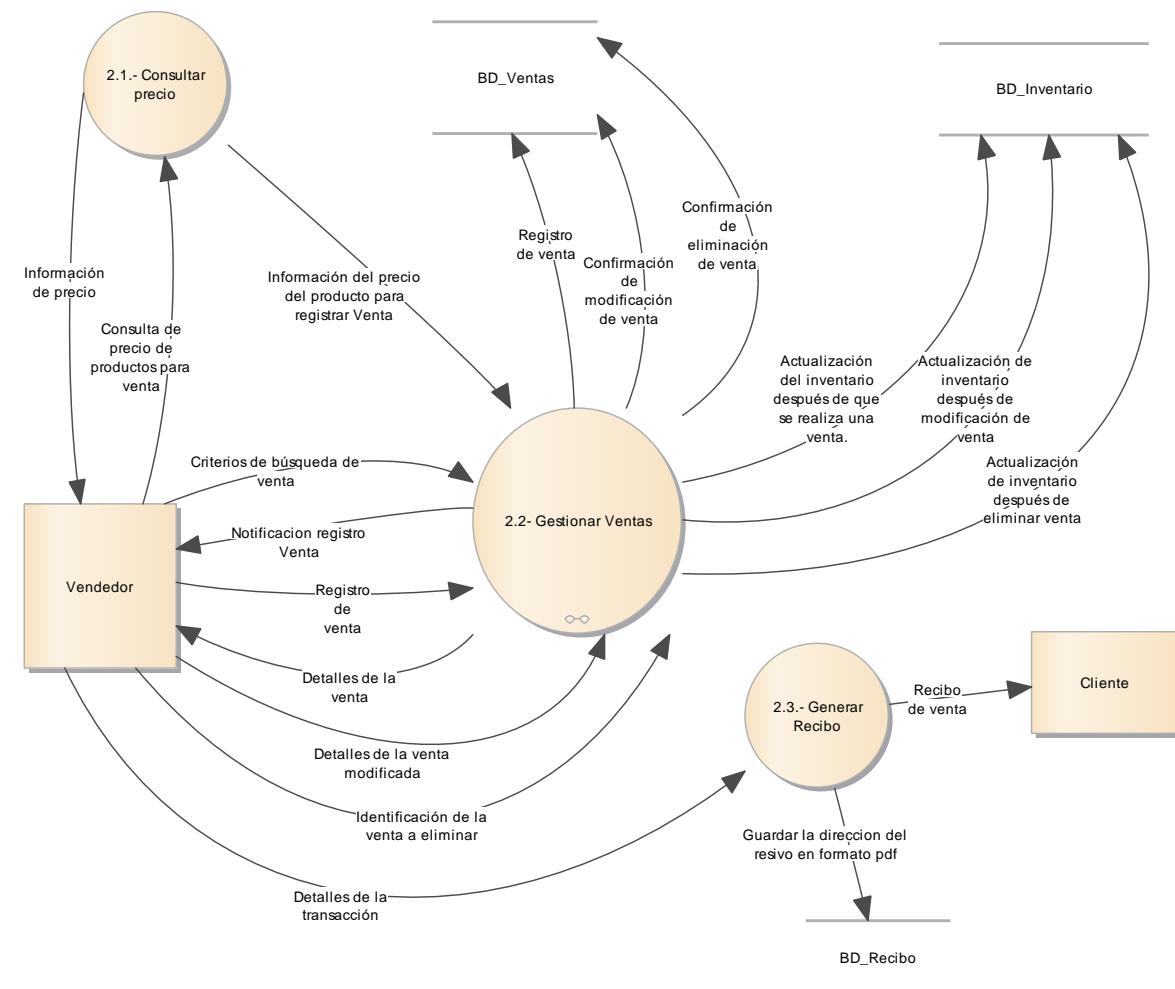


Figure 7: Gestionar Ventas

## **1.- Iniciar sesión como vendedor**

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

1.- Iniciar sesión como vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## **2.- Consultar precio**

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

2.- Consultar precio

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

### **2.1.- Consultar precio**

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

Este proceso permite al usuario, ya sea un vendedor o administrador, consultar el precio de los productos disponibles en el inventario. Proporciona información detallada sobre los productos, incluyendo el precio, para facilitar la venta y gestión de inventario.

2.1.- Consultar precio

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Información del precio del producto para registrar Venta    Source -> Destination

From: 2.1.- Consultar precio : Process, Public  
To: 2.2- Gestionar Ventas : Process, Public

Descripción: Transferencia de información específica del producto (incluyendo el precio) desde la base de datos al proceso de registro de venta para completar la transacción.

La estructura de datos que viaja con el flujo

Información del precio del producto para Registrar Venta = ID del producto + Precio del producto

↗ **DataFlow** «DFD\_DataFlow» Información de precio    Source -> Destination

From: 2.1.- Consultar precio : Process, Public  
To: Vendedor : External, Public

Descripción: Información detallada de productos y precio

Información de precio = Nombre del producto + Precio del producto + Descripción del producto

↗ **DataFlow** «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public  
To: 2.1.- Consultar precio : Process, Public

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

la estructura de datos que viaja con el flujo

Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto

### 2.1.- Iniciar sesión como vendedor

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

2.1.- Iniciar sesión como vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 19/6/2024

CONNECTORS
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Source -&gt; Destination From: 2.1.- Iniciar sesión como vendedor : Process, Public To: BD_Usuarios : DataStore, Public</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Credenciales de inicio de sesión (nombre de usuario y contraseña) Source -&gt; Destination From: Vendedor : External, Public To: 2.1.- Iniciar sesión como vendedor : Process, Public</p>

## 2.2- Gestionar Ventas

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

Este proceso abarca todas las acciones relacionadas con la gestión de ventas, desde el registro de nuevas ventas hasta la modificación y eliminación de ventas existentes. Se asegura de que todas las transacciones sean precisas y estén actualizadas en el sistema.

### 2.2- Gestionar Ventas

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 4/7/2024

ELEMENTS OWNED BY 2.2- Gestionar Ventas
<p>█ 2.2.1.- Registrar Venta : Process «DFD_Process»  Permite al vendedor registrar una nueva venta en el sistema. Este proceso incluye la entrada de datos del cliente, los productos vendidos, cantidades y el precio total. Asegura que la venta se registre correctamente en la base de datos de ventas y se actualice el inventario.</p>

ELEMENTS OWNED BY 2.2- Gestionar Ventas	
■ 2.2.2.- Consultar Venta : Process «DFD_Process»	Permite al usuario buscar y visualizar detalles de ventas registradas previamente. Se pueden usar varios criterios de búsqueda como ID de la venta, fecha, o datos del cliente para acceder a la información requerida.
■ 2.2.3.- Modificar Venta : Process «DFD_Process»	Permite al vendedor modificar detalles de una venta ya registrada, como productos vendidos, cantidades y precios. Asegura que cualquier cambio en la venta se refleje correctamente en la base de datos de ventas y en el inventario.
■ 2.2.4.- Eliminar Venta : Process «DFD_Process»	Permite al vendedor eliminar una venta previamente registrada del sistema. Este proceso actualiza tanto la base de datos de ventas como el inventario para reflejar la eliminación de la venta.
■ BD_Inventario : DataStore «DFD_DataStore»	<p>Descripción: Gestiona el inventario de productos.</p> <p>Estructura de datos ID_Inventario (PK) + ID_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad_Disponible</p>
■ BD_Ventas : DataStore «DFD_DataStore»	<p>Descripción: Registra todas las transacciones de venta realizadas.</p> <p>Estructura de datos: ID_Venta (PK) + Fecha + ID_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio_Total</p>
■ DataStore1 : DataStore «DFD_DataStore»	

## ELEMENTS OWNED BY 2.2- Gestionar Ventas

■ Vendedor : External <DFD\_External>

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de eliminar venta    Source -> Destination  
From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo  
Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de venta    Source -> Destination  
From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo  
Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

↗ **DataFlow** «DFD\_DataFlow» Registro de venta    Source -> Destination  
From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo  
Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidades + Precio total + Forma de pago

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización del inventario después de que se realiza una venta.    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de venta = Productos vendidos + Cantidades vendidas

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de venta    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Notificacion registro Venta    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public  
To: 2.2- Gestionar Ventas : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registro de venta    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Información del precio del producto para registrar Venta    Source -> Destination

From: 2.1.- Consultar precio : Process, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: Transferencia de información específica del producto (incluyendo el precio) desde la base de datos al proceso de registro de venta para completar la transacción.

La estructura de datos que viaja con el flujo

Información del precio del producto para Registrar Venta = ID del producto + Precio del producto

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

## 3- Gestionar Ventas diagram

*Data Flow diagram in package 'Business Process Model'*

3- Gestionar Ventas

Version 1.0

ASUS created on 15/6/2024. Last modified 15/6/2024

Figure 8: 3- Gestionar Ventas

## Gestinar Ventas diagram

*Data Flow diagram in package 'Business Process Model'*

Gestinar Ventas

Version 1.0

ASUS created on 15/6/2024. Last modified 2/7/2024

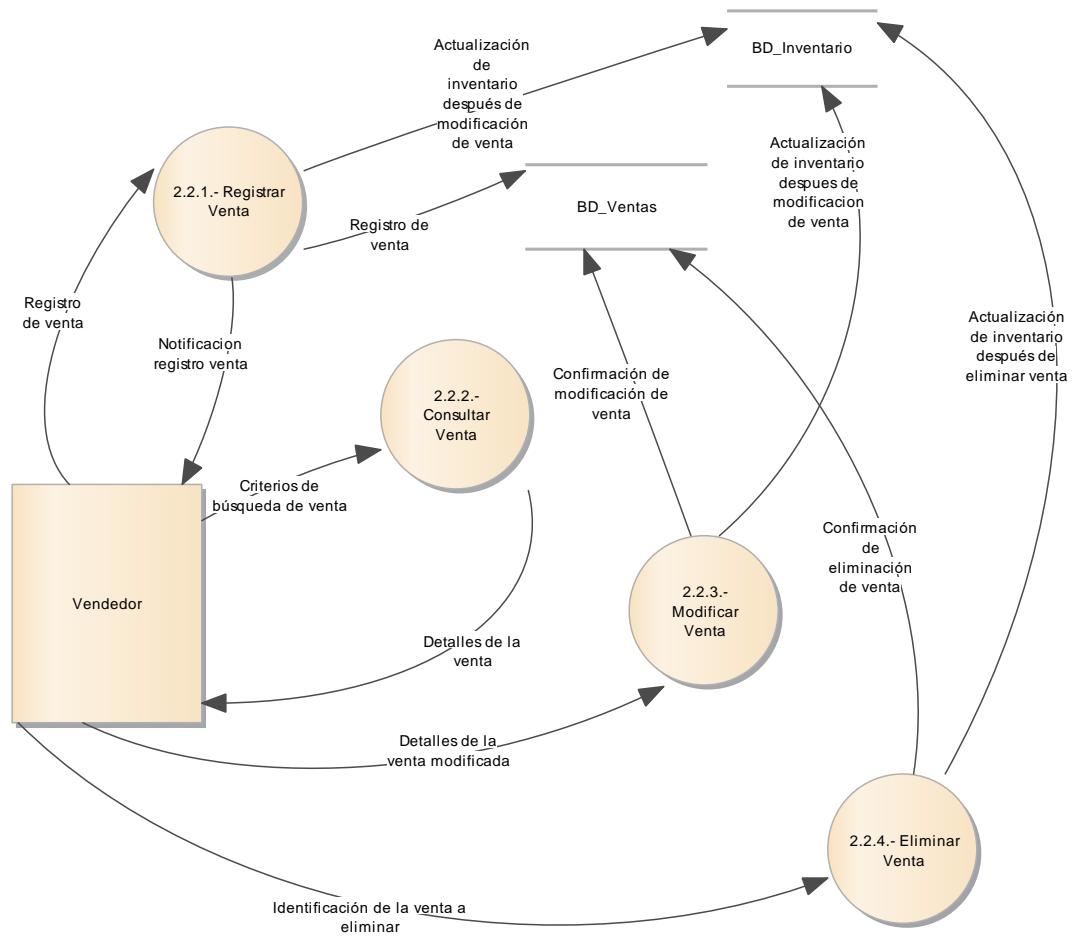


Figure 9: Gestinar Ventas

## 2.2.1.- Registrar Venta

Process «DFD\_Process» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'

Permite al vendedor registrar una nueva venta en el sistema. Este proceso incluye la entrada de datos del cliente, los productos vendidos, cantidades y el precio total. Asegura que la venta se registre correctamente en la base de datos de ventas y se actualice el inventario.

### 2.2.1.- Registrar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 4/7/2024

#### CONNECTORS

 **DataFlow** «DFD\_DataFlow» Notificacion registro venta    Source -> Destination

From: 2.2.1.- Registrar Venta : Process, Public  
To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

 **DataFlow** «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source -> Destination

From: 2.2.1.- Registrar Venta : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registro de venta   Source -> Destination

From: 2.2.1.- Registrar Venta : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

### ↗ DataFlow «DFD\_DataFlow» Registro de venta   Source -> Destination

From: Vendedor : External, Public  
To: 2.2.1.- Registrar Venta : Process, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

## 2.2.2.- Consultar Venta

Process «DFD\_Process» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'

Permite al usuario buscar y visualizar detalles de ventas registradas previamente. Se pueden usar varios criterios de búsqueda como ID de la venta, fecha, o datos del cliente para acceder a la información requerida.

2.2.2.- Consultar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 4/7/2024

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: 2.2.2.- Consultar Venta : Process, Public  
To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public  
To: 2.2.2.- Consultar Venta : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

## 2.2.3.- Modificar Venta

Process «DFD\_Process» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'

Permite al vendedor modificar detalles de una venta ya registrada, como productos vendidos, cantidades y precios. Asegura que cualquier cambio en la venta se refleje correctamente en la base de datos de ventas y en el inventario.

### 2.2.3.- Modificar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 4/7/2024

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Confirmación de modificación de venta    Source -> Destination

From: 2.2.3.- Modificar Venta : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

### ↗ DataFlow «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source -> Destination

From: 2.2.3.- Modificar Venta : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de venta = Productos vendidos + Cantidad vendida

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public  
To: 2.2.3.- Modificar Venta : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad vendida + Precio total actualizado

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

## 2.2.4.- Eliminar Venta

Process «DFD\_Process» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'

permite al vendedor eliminar una venta previamente registrada del sistema. Este proceso actualiza tanto la base de datos de ventas como el inventario para reflejar la eliminación de la venta.

#### 2.2.4.- Eliminar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 4/7/2024

#### CONNECTORS

##### DataFlow «DFD\_DataFlow» Actualización de inventario después de eliminar venta   Source -> Destination

From: 2.2.4.- Eliminar Venta : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

##### DataFlow «DFD\_DataFlow» Confirmación de eliminación de venta   Source -> Destination

From: 2.2.4.- Eliminar Venta : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

##### DataFlow «DFD\_DataFlow» Identificación de la venta a eliminar   Source -> Destination

From: Vendedor : External, Public  
To: 2.2.4.- Eliminar Venta : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

## **BD\_Inventario**

*DataStore «DFD\_DataStore» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'*

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 2/7/2024

CONNECTORS
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Actualización de inventario después de eliminar venta    Source -&gt; Destination From: 2.2.4.- Eliminar Venta : Process, Public To: BD_Inventario : DataStore, Public</p> <p>Descripción: Actualización del inventario después de que se elimina una venta.</p> <p>La estructura de datos que viaja con el flujo Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Actualización de inventario despues de modificacion de venta    Source -&gt; Destination From: 2.2.3.- Modificar Venta : Process, Public To: BD_Inventario : DataStore, Public</p> <p>Descripción: Actualización del inventario después de que se realiza una venta.</p> <p>La estructura de datos que viaja con el flujo Actualización de inventario después de venta = Productos vendidos + Cantidades vendidas</p>

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source -> Destination

From: 2.2.1.- Registrar Venta : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

## BD\_Ventas

*DataStore «DFD\_DataStore» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'*

Descripción: Registra todas las transacciones de venta realizadas.

Estructura de datos: ID\_Venta (PK) + Fecha + ID\_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio\_Total

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de venta    Source -> Destination

From: 2.2.3.- Modificar Venta : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de venta    Source -> Destination

From: 2.2.4.- Eliminar Venta : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

↗ **DataFlow** «DFD\_DataFlow» Registro de venta    Source -> Destination

From: 2.2.1.- Registrar Venta : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

## DataStore1

*DataStore «DFD\_DataStore» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'*

DataStore1

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 15/6/2024

## Vendedor

*External «DFD\_External» owned by '2.2- Gestionar Ventas', in package 'Business Process Model'*

Vendedor

Version 1.0 Phase 1.0 Proposed

CONNECTORS
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Criterios de búsqueda de venta    Source -&gt; Destination</p> <p>From: Vendedor : External, Public To: 2.2.2.- Consultar Venta : Process, Public</p> <p>Descripción: El vendedor especifica criterios para buscar ventas en el sistema.</p> <p>La estructura de datos que viaja con el flujo Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Registro de venta    Source -&gt; Destination</p> <p>From: Vendedor : External, Public To: 2.2.1.- Registrar Venta : Process, Public</p> <p>Descripción: Registro de una nueva venta en la base de datos de ventas.</p> <p>La estructura de datos que viaja con el flujo Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidades + Precio total + Forma de pago</p>
<p>↗ <b>DataFlow</b> «DFD_DataFlow» Detalles de la venta modificada    Source -&gt; Destination</p> <p>From: Vendedor : External, Public To: 2.2.3.- Modificar Venta : Process, Public</p> <p>Descripción: Detalles de una venta que ha sido modificada por el vendedor.</p> <p>La estructura de datos que viaja con el flujo Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidades + Precio total actualizado Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad</p>

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public

To: 2.2.4.- Eliminar Venta : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

↗ **DataFlow** «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: 2.2.2.- Consultar Venta : Process, Public

To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

↗ **DataFlow** «DFD\_DataFlow» Notificacion registro venta    Source -> Destination

From: 2.2.1.- Registrar Venta : Process, Public

To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

## 2.3.- Generar Recibo

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

Este proceso genera y emite un recibo detallado para el cliente después de completar una venta. Incluye la creación del recibo en formato PDF, que se almacena en la

base de datos de recibos para referencia futura.

### 2.3.- Generar Recibo

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 4/7/2024

#### CONNECTORS

##### ↗ DataFlow «DFD\_DataFlow» Recibo de venta    Source -> Destination

From: 2.3.- Generar Recibo : Process, Public  
To: Cliente : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

##### ↗ DataFlow «DFD\_DataFlow» Guardar la dirección del resivo en formato pdf    Source -> Destination

From: 2.3.- Generar Recibo : Process, Public  
To: BD\_Recibo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo

Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

##### ↗ DataFlow «DFD\_DataFlow» Detalles de la transacción    Source -> Destination

From: Vendedor : External, Public  
To: 2.3.- Generar Recibo : Process, Public

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

La estructura de datos que viaja con el flujo

Detalles de la transacción = ID de la transacción + Productos comprados + Precio total

Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### **3.- Registrar Venta**

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

3.- Registrar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

### **4.- Generar Recibo**

*Process «DFD\_Process» owned by 'Proceso Venta', in package 'Business Process Model'*

4.- Generar Recibo

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

### **ImageAsset**

*Artifact «Image» owned by 'Proceso Venta', in package 'Business Process Model'*

ImageAsset

Version 1.0 Phase 1.0 Proposed

J2D created on 12/6/2024. Last modified 12/6/2024

### **BD\_Inventario**

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

BD\_Inventario

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de eliminar venta    Source -> Destination  
From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo  
Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

↗ **DataFlow** «DFD\_DataFlow» Actualización del inventario después de que se realiza una venta.    Source -> Destination  
From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo  
Actualización de inventario después de venta = Productos vendidos + Cantidades vendidas

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de modificación de venta    Source -> Destination  
From: 2.2- Gestionar Ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo  
Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

## BD\_Inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## BD\_Recibo

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

Descripción: :Almacena los recibos generados para transacciones.

Estructura de datos ID\_Recibo (PK) + ID\_Venta (FK hacia Venta) + Direccion\_PDF

## BD\_Recibo

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 2/7/2024

### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Guardar la direccion del resivo en formato pdf    Source -> Destination  
From: 2.3.- Generar Recibo : Process, Public  
To: BD\_Recibo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo

Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

## BD\_Resivo

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

## BD\_Resivo

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## BD\_Usuarios

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

## BD\_Usuarios

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

CONNECTORS
<p>↗ <b>DataFlow</b> «DFD_DataFlow»    Source -&gt; Destination From: 2.1.- Iniciar sesión como vendedor : Process, Public To: BD_Usuarios : DataStore, Public</p>

## BD\_Usuarios

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

## BD\_Usuarios

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## BD\_Ventas

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

Descripción: Registra todas las transacciones de venta realizadas.

Estructura de datos: ID\_Venta (PK) + Fecha + ID\_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio\_Total

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 2/7/2024

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Confirmación de modificación de venta   Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

### ↗ DataFlow «DFD\_DataFlow» Registro de venta   Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

### ↗ DataFlow «DFD\_DataFlow» Confirmación de eliminación de venta   Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

## **BD\_Ventas**

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## **Cliente**

*External «DFD\_External» owned by 'Proceso Venta', in package 'Business Process Model'*

Cliente

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## **Cliente**

*External «DFD\_External» owned by 'Proceso Venta', in package 'Business Process Model'*

El cliente, siendo el usuario final, interactúa directamente con el vendedor para llevar a cabo compras en la tienda. Aunque no tiene un contacto directo con el sistema de gestión de ventas e inventario, sus acciones y necesidades influyen en cómo el vendedor utiliza dicho sistema. Entre las interacciones del cliente se incluyen: seleccionar productos para su compra y efectuar el pago en la caja, solicitar información detallada sobre productos específicos, recibir un recibo impreso al completar una transacción y, en caso de devoluciones, solicitar al vendedor el procesamiento correspondiente, lo que puede implicar ajustes en el inventario.

Cliente

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Recibo de venta   Source -> Destination

From: 2.3.- Generar Recibo : Process, Public

To: Cliente : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

## DataStore2

*DataStore «DFD\_DataStore» owned by 'Proceso Venta', in package 'Business Process Model'*

DataStore2

Version 1.0 Phase 1.0 Proposed

ASUS created on 15/6/2024. Last modified 15/6/2024

## Vendedor

*External «DFD\_External» owned by 'Proceso Venta', in package 'Business Process Model'*

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 8/6/2024

## Vendedor

*External «DFD\_External» owned by 'Proceso Venta', in package 'Business Process Model'*

El vendedor, también conocido como cajero, despliega un papel crucial como usuario principal del sistema de gestión de ventas e inventario. En el transcurso de su jornada laboral, interactúa directamente con la plataforma para llevar a cabo diversas funciones. Entre las responsabilidades del vendedor se encuentran: registrar ventas mediante el procesamiento de transacciones, consultar información sobre productos para ayudar a los clientes, generar recibos para las ventas realizadas, y actualizar el

inventario disminuyendo las existencias de productos vendidos. Además, con la autorización del administrador, el vendedor puede acceder al módulo de inventario para añadir nuevos productos a la base de datos.

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 12/6/2024. Last modified 4/7/2024

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

### ↗ DataFlow «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public

To: 2.1.- Consultar precio : Process, Public

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

La estructura de datos que viaja con el flujo

Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto

### ↗ DataFlow «DFD\_DataFlow» Detalles de la transacción    Source -> Destination

From: Vendedor : External, Public

To: 2.3.- Generar Recibo : Process, Public

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

La estructura de datos que viaja con el flujo

Detalles de la transacción = ID de la transacción + Productos comprados + Precio total

Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registro de venta    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public

To: 2.2- Gestionar Ventas : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

### ↗ DataFlow «DFD\_DataFlow» Credenciales de inicio de sesión (nombre de usuario y contraseña)    Source -> Destination

From: Vendedor : External, Public

To: 2.1.- Iniciar sesión como vendedor : Process, Public

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Notificacion registro Venta    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public

To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

↗ **DataFlow** «DFD\_DataFlow» Información de precio    Source -> Destination

From: 2.1.- Consultar precio : Process, Public

To: Vendedor : External, Public

Descripción: Información detallada de productos y precio

Información de precio = Nombre del producto + Precio del producto + Descripción del producto

↗ **DataFlow** «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: 2.2- Gestionar Ventas : Process, Public

To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

Sistema de caja NortonSoft

*Process «DFD\_Process» in package 'Business Process Model'*

Sistema encargado de ingresar ventas,actualizar inventario y agrgar nuevos productos al inventario

Sistema de caja NortonSoft

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 11/6/2024

ELEMENTS OWNED BY Sistema de caja NortonSoft	
■ 1.- Gestionar inicio de cuenta : Process «DFD_Process»	Los vendedores y administradores interactúan con el sistema de en el modulo de gestionar inicio de cuentas para gestionar usuarios y permisos.
■ 1.- Iniciar Sesión como Administrador : Process «DFD_Process»	
■ 1.- Iniciar sesión como vendedor : Process «DFD_Process»	
■ 2.- Acceder al Módulo de inventario : Process «DFD_Process»	
■ 2.- Consultar precio : Process «DFD_Process»	
■ 2.- Gestionar ventas : Process «DFD_Process»	fdsvdsv
■ 3.- Agregar Nuevo Producto : Process «DFD_Process»	
■ 3.- Registrar Venta : Process «DFD_Process»	
■ 4.- Generar Recibo : Process «DFD_Process»	

ELEMENTS OWNED BY Sistema de caja NortonSoft	
■ a : Process «DFD_Process»	
■ Administrador : External «DFD_External»	
■ Administrador : External «DFD_External»	
<p>El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que pueda surgir.</p>	
■ b : External «DFD_External»	
■ BD_Inventario : DataStore «DFD_DataStore»	
■ BD_Inventario : DataStore «DFD_DataStore»	

ELEMENTS OWNED BY Sistema de caja NortonSoft	
■ BD_Inventario : DataStore «DFD_DataStore»	<p>Descripción: Gestiona el inventario de productos.</p> <p>Estructura de datos ID_Inventario (PK) + ID_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad_Disponible</p>
■ BD_Recibo : DataStore «DFD_DataStore»	<p>Descripción: Almacena los recibos generados para transacciones.</p> <p>Estructura de datos ID_Recibo (PK) + ID_Venta (FK hacia Venta) + Direccion_PDF</p>
■ BD_Resivo : DataStore «DFD_DataStore»	
■ BD_Usuario : DataStore «DFD_DataStore»	
■ BD_Usuario : DataStore «DFD_DataStore»	<p>Descripción: Almacena la información básica de los usuarios del sistema, incluyendo la hora de ingreso.</p> <p>Estructura de datos: ID (PK) + Usuario + Fecha_Ingreso</p>
■ BD_Usuarios : DataStore «DFD_DataStore»	
■ BD_Ventas : DataStore «DFD_DataStore»	

## ELEMENTS OWNED BY Sistema de caja NortonSoft

### ■ BD\_Ventas : DataStore «DFD\_DataStore»

Descripción: Registra todas las transacciones de venta realizadas.

Estructura de datos: ID\_Venta (PK) + Fecha + ID\_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio\_Total

### ■ Cliente : External «DFD\_External»

### ■ Cliente : External «DFD\_External»

El cliente, siendo el usuario final, interactúa directamente con el vendedor para llevar a cabo compras en la tienda. Aunque no tiene un contacto directo con el sistema de gestión de ventas e inventario, sus acciones y necesidades influyen en cómo el vendedor utiliza dicho sistema. Entre las interacciones del cliente se incluyen: seleccionar productos para su compra y efectuar el pago en la caja, solicitar información detallada sobre productos específicos, recibir un recibo impreso al completar una transacción y, en caso de devoluciones, solicitar al vendedor el procesamiento correspondiente, lo que puede implicar ajustes en el inventario.

### ■ Vendedor : External «DFD\_External»

### ■ Vendedor : External «DFD\_External»

El vendedor, también conocido como cajero, despliega un papel crucial como usuario principal del sistema de gestión de ventas e inventario. En el transcurso de su jornada laboral, interactúa directamente con la plataforma para llevar a cabo diversas funciones. Entre las responsabilidades del vendedor se encuentran: registrar ventas mediante el procesamiento de transacciones, consultar información sobre productos para ayudar a los clientes, generar recibos para las ventas realizadas, y actualizar el inventario disminuyendo las existencias de productos vendidos. Además, con la autorización del administrador, el vendedor puede acceder al módulo de inventario para añadir nuevos productos a la base de datos.

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Confirmacion de ingreso con usuario administrador    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Administrador : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario administrador.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso

### ↗ DataFlow «DFD\_DataFlow» Información del producto    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Administrador : External, Public

Descripción: Información detallada de un producto específico para el administrador.

a estructura de datos que viaja con el flujo

Información del producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Recibo de venta    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Clientes : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

### ↗ DataFlow «DFD\_DataFlow» Producto agregado    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: BD\_Inventario : DataStore, Public

### ↗ DataFlow «DFD\_DataFlow» Información de precio    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Información detallada de un producto específico para el administrador.

La estructura de datos que viaja con el flujo

Información de precio = Nombre del producto + Precio del producto + Descripción del producto

### ↗ DataFlow «DFD\_DataFlow» Notificación de registro de venta    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registrar Hora de Ingreso del Usuario    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: BD\_Usuario : DataStore, Public

Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.

La estructura de datos que viaja con el flujo :

Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso

### ↗ DataFlow «DFD\_DataFlow» Confirmacion de ingreso con usuario vendedor    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Vendedor : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario vendedor.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso

### ↗ DataFlow «DFD\_DataFlow» Guardar la direccion del resivo en formato pdf    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: BD\_Recibo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo

Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Notificacion producto agregado    Source -> Destination

From: Sistema de caja NortonSoft : Process, Public  
To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

### ↗ DataFlow «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Administrador : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

### ↗ DataFlow «DFD\_DataFlow» Datos Login/Paswoord vendedor    Source -> Destination

From: Vendedor : External, Public  
To: Sistema de caja NortonSoft : Process, Public

Descripción: Autenticación del vendedor para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword vendedor = Usuario + Contraseña

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

La estructura de datos que viaja con el flujo

Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto

### ↗ DataFlow «DFD\_DataFlow» Detalles de la transacción    Source -> Destination

From: Vendedor : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

La estructura de datos que viaja con el flujo

Detalles de la transacción = ID de la transacción + Productos comprados + Precio total

Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado

Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

↗ **DataFlow** «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + Inventario inicial

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord administrador    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: Autenticación del administrador para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword administrador = Usuario + Contraseña

↗ **DataFlow** «DFD\_DataFlow» Consulta de producto    Source -> Destination

From: Administrador : External, Public

To: Sistema de caja NortonSoft : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

## Precesos diagram

*Data Flow diagram in package 'Business Process Model'*

Precesos

Version 1.0

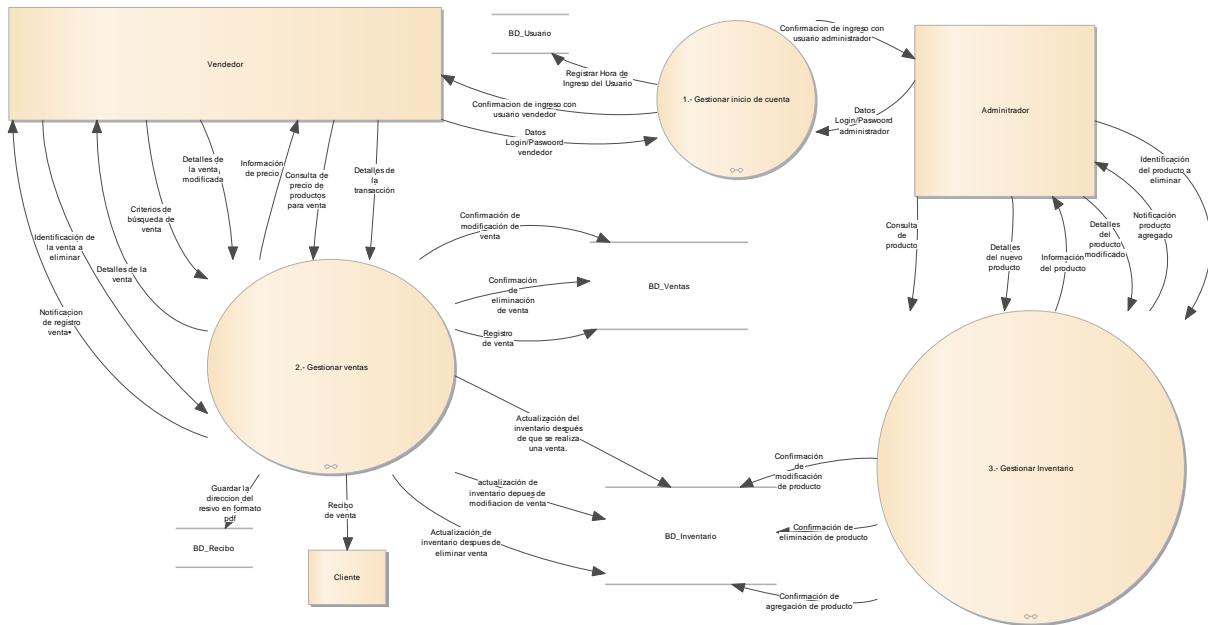


Figure 10: Precesos

### 1.- Gestionar inicio de cuenta

Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'

Los vendedores y administradores interactúan con el sistema de en el modulo de gestionar inicio de cuentas para gestionar usuarios y permisos.

#### 1.- Gestionar inicio de cuenta

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## ELEMENTS OWNED BY 1.- Gestionar inicio de cuenta

### █ 1.1.-Verificar tipo de cuenta : Process «DFD\_Process»

Verificación de las credenciales del usuario (nombre de usuario y contraseña) para garantizar que la persona que intenta acceder es quien dice ser.

### █ Process1 : Process «DFD\_Process»

### █ BD\_Usuarios : DataStore «DFD\_DataStore»

### █ Vendedor : External «DFD\_External»

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registrar Hora de Ingreso del Usuario    Source -> Destination

From: 1.- Gestionar inicio de cuenta : Process, Public  
To: BD\_Usuario : DataStore, Public

Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.

La estructura de datos que viaja con el flujo :

Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso

### ↗ DataFlow «DFD\_DataFlow» Confirmacion de ingreso con usuario vendedor    Source -> Destination

From: 1.- Gestionar inicio de cuenta : Process, Public  
To: Vendedor : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario vendedor.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario administrador   Source -> Destination

From: 1.- Gestionar inicio de cuenta : Process, Public

To: Administrador : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario administrador.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord vendedor   Source -> Destination

From: Vendedor : External, Public

To: 1.- Gestionar inicio de cuenta : Process, Public

Descripción: Autenticación del vendedor para acceder al sistema y solo modulos autorizados

La estructura de datos que viaja con el flujo

Datos Login/Pasword vendedor = Usuario + Contraseña

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord administrador   Source -> Destination

From: Administrador : External, Public

To: 1.- Gestionar inicio de cuenta : Process, Public

Descripción: Autenticación del administrador para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword administrador = Usuario + Contraseña

## 1.- Gestionar inicio de cuenta diagram

*Data Flow diagram in package 'Business Process Model'*

1.- Gestionar inicio de cuenta

Version 1.0

Fabian created on 19/6/2024. Last modified 19/6/2024



Figure 11: 1.- Gestionar inicio de cuenta

### 1.- Gestionar inicio de cuenta diagram

*Data Flow diagram in package 'Business Process Model'*

1.- Gestionar inicio de cuenta

Version 1.0

Fabian created on 19/6/2024. Last modified 2/7/2024

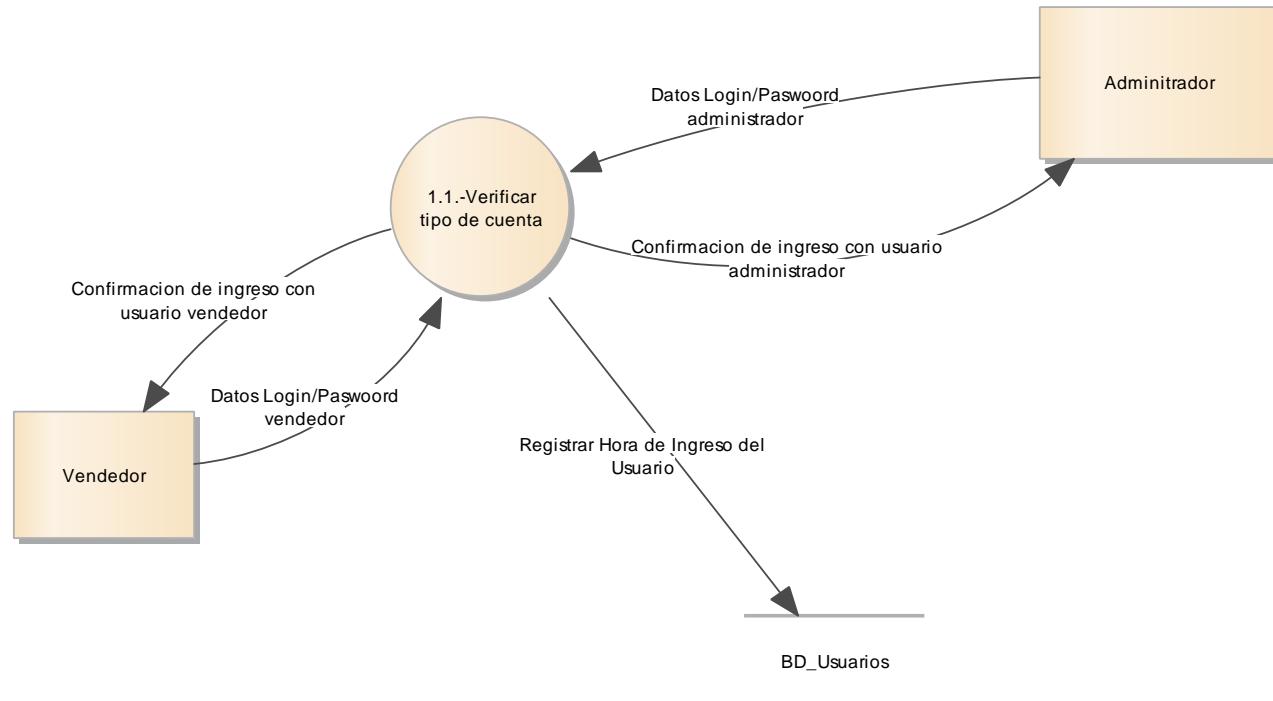


Figure 12: 1.- Gestionar inicio de cuenta

### 1.1.-Verificar tipo de cuenta

*Process «DFD\_Process» owned by '1.- Gestionar inicio de cuenta', in package 'Business Process Model'*

Verificación de las credenciales del usuario (nombre de usuario y contraseña) para garantizar que la persona que intenta acceder es quien dice ser.

1.1.-Verificar tipo de cuenta

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Confirmacion de ingreso con usuario vendedor    Source -> Destination

From: 1.1.-Verificar tipo de cuenta : Process, Public  
To: Vendedor : External, Public

Descripción: Mensaje de confirmación del acceso exitoso al sistema por parte de un usuario vendedor.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso

### ↗ DataFlow «DFD\_DataFlow» Registrar Hora de Ingreso del Usuario    Source -> Destination

From: 1.1.-Verificar tipo de cuenta : Process, Public  
To: BD\_Usuarios : DataStore, Public

Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.

La estructura de datos que viaja con el flujo :

Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso

### ↗ DataFlow «DFD\_DataFlow» Confirmacion de ingreso con usuario administrador    Source -> Destination

From: 1.1.-Verificar tipo de cuenta : Process, Public  
To: Adminitrador : External, Public

Descripción: Mensaje de confirmación del acceso exitoso al sistema por parte de un usuario administrador.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord administrador    Source -> Destination

From:    Administrador : External, Public

To:        1.1.-Verificar tipo de cuenta : Process, Public

Descripción: Autenticación del administrador para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword administrador = Usuario + Contraseña

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord vendedor    Source -> Destination

From:    Vendedor : External, Public

To:        1.1.-Verificar tipo de cuenta : Process, Public

Descripción: Autenticación del vendedor para acceder al sistema y solo modulos autorizados

La estructura de datos que viaja con el flujo

Datos Login/Pasword vendedor = Usuario + Contraseña

## Process1

*Process «DFD\_Process» owned by '1.- Gestionar inicio de cuenta', in package 'Business Process Model'*

Process1

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 19/6/2024

## BD\_Usuarios

*DataStore «DFD\_DataStore» owned by '1.- Gestionar inicio de cuenta', in package 'Business Process Model'*

## BD\_Usuarios

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 19/6/2024

### CONNECTORS

#### ↗ DataFlow «DFD\_DataFlow» Registrar Hora de Ingreso del Usuario    Source -> Destination

From: 1.1.-Verificar tipo de cuenta : Process, Public

To: BD\_Usuarios : DataStore, Public

Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.

La estructura de datos que viaja con el flujo :

Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso

## Vendedor

External «DFD\_External» owned by '1.- Gestionar inicio de cuenta', in package 'Business Process Model'

## Vendedor

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

### CONNECTORS

#### ↗ DataFlow «DFD\_DataFlow» Datos Login/Paswoord vendedor    Source -> Destination

From: Vendedor : External, Public

To: 1.1.-Verificar tipo de cuenta : Process, Public

Descripción: Autenticación del vendedor para acceder al sistema y solo modulos autorizados

La estructura de datos que viaja con el flujo

Datos Login/Pasword vendedor = Usuario + Contraseña

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario vendedor   Source -> Destination

From: 1.1.-Verificar tipo de cuenta : Process, Public

To: Vendedor : External, Public

Descripción: Mensaje de confirmación del acceso exitoso al sistema por parte de un usuario vendedor.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso

## 1.- Iniciar Sesión como Administrador

*Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

1.- Iniciar Sesión como Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de inicio de sesión como administrador   Source -> Destination

From: 1.- Iniciar Sesión como Administrador : Process, Public

To: BD\_Usuario : DataStore, Public

↗ **DataFlow** «DFD\_DataFlow» Credenciales de administrador (nombre de usuario y contraseña)   Source ->

Destination

From: Administrador : External, Public

To: 1.- Iniciar Sesión como Administrador : Process, Public

## 1.- Iniciar sesión como vendedor

*Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

1.- Iniciar sesión como vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Source -> Destination

From: 1.- Iniciar sesión como vendedor : Process, Public  
To: BD\_Usuarios : DataStore, Public

↗ **DataFlow** «DFD\_DataFlow» Credenciales de inicio de sesión (nombre de usuario y contraseña) Source -> Destination

From: Vendedor : External, Public  
To: 1.- Iniciar sesión como vendedor : Process, Public

#### 2.- Acceder al Módulo de inventario

*Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

2.- Acceder al Módulo de inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Interfaz del módulo de inventario Source -> Destination

From: 2.- Acceder al Módulo de inventario : Process, Public  
To: Administrador : External, Public

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Acceder al Módulo de inventario con cuenta del administrador    Source -> Destination  
From: BD\_Usuario : DataStore, Public  
To: 2.- Acceder al Módulo de inventario : Process, Public

## 2.- Consultar precio

*Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

2.- Consultar precio

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Información de productos (nombre, precio)    Source -> Destination  
From: 2.- Consultar precio : Process, Public  
To: Vendedor : External, Public

↗ **DataFlow** «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination  
From: Vendedor : External, Public  
To: 2.- Consultar precio : Process, Public

## 2.- Gestionar ventas

*Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

fdsvdsv

2.- Gestionar ventas

Version 1.0 Phase 1.0 Proposed

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles de la venta    Source -> Destination  
From: 2.- Gestionar ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo  
Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total  
Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

↗ **DataFlow** «DFD\_DataFlow» Notificacion de registro venta•    Source -> Destination  
From: 2.- Gestionar ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo  
Notificación de registro de venta = ID de la venta registrada + Fecha de registro

↗ **DataFlow** «DFD\_DataFlow» Guardar la direccion del resivo en formato pdf    Source -> Destination  
From: 2.- Gestionar ventas : Process, Public  
To: BD\_Recibo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo  
Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» actualización de inventario después de modificación de venta      Source ->

Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

↗ **DataFlow** «DFD\_DataFlow» Actualización del inventario después de que se realiza una venta.      Source ->

Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de venta = Productos vendidos + Cantidades vendidas

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de venta      Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de eliminar venta   Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

↗ **DataFlow** «DFD\_DataFlow» Recibo de venta   Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: Cliente : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidades + Precio total + Detalles adicionales según configuración

↗ **DataFlow** «DFD\_DataFlow» Información de precio   Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: Vendedor : External, Public

Descripción: Información detallada de productos y precio

Información de precio = Nombre del producto + Precio del producto + Descripción del producto

↗ **DataFlow** «DFD\_DataFlow» Confirmación de eliminación de venta   Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Registro de venta    Source -> Destination

From: 2.- Gestionar ventas : Process, Public  
To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

### ↗ DataFlow «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

### ↗ DataFlow «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

La estructura de datos que viaja con el flujo

Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la transacción    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

La estructura de datos que viaja con el flujo

Detalles de la transacción = ID de la transacción + Productos comprados + Precio total  
Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado  
Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

## 3.- Agregar Nuevo Producto

Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'

### 3.- Agregar Nuevo Producto

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto   Source -> Destination

From: 3.- Agregar Nuevo Producto : Process, Public

To: BD\_Inventario : DataStore, Public

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto (nombre, descripción, precio, cantidad)   Source -> Destination

From: Administrador : External, Public

To: 3.- Agregar Nuevo Producto : Process, Public

### 3.- Registrar Venta

Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'

#### 3.- Registrar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» actualizar inventario   Source -> Destination

From: 3.- Registrar Venta : Process, Public

To: BD\_Inventario : DataStore, Public

↗ **DataFlow** «DFD\_DataFlow» Confirmación de venta   Source -> Destination

From: 3.- Registrar Venta : Process, Public

To: BD\_Ventas : DataStore, Public

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles de la venta (productos vendidos, cantidades, precio total)   Source -> Destination  
From: Vendedor : External, Public  
To: 3.- Registrar Venta : Process, Public

## 4.- Generar Recibo

Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'

### 4.- Generar Recibo

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Guardar la direccion del resivo en formato pdf   Source -> Destination  
From: 4.- Generar Recibo : Process, Public  
To: BD\_Resivo : DataStore, Public

↗ **DataFlow** «DFD\_DataFlow» Recibo impreso   Source -> Destination  
From: 4.- Generar Recibo : Process, Public  
To: Cliente : External, Public

↗ **DataFlow** «DFD\_DataFlow» Detalles de la transacción (productos comprados, precio total)   Source -> Destination  
From: Vendedor : External, Public  
To: 4.- Generar Recibo : Process, Public

*Process «DFD\_Process» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

a

Version 1.0 Phase 1.0 Proposed

ASUS created on 21/6/2024. Last modified 21/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Source -> Destination

From: a : Process, Public  
To: b : External, Public

## Administrador

*External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Credenciales de administrador (nombre de usuario y contraseña) Source -> Destination

From: Administrador : External, Public  
To: 1.- Iniciar Sesión como Administrador : Process, Public

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto (nombre, descripción, precio, cantidad) Source -> Destination

From: Administrador : External, Public  
To: 3.- Agregar Nuevo Producto : Process, Public

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Interfaz del módulo de inventario    Source -> Destination  
From: 2.- Acceder al Módulo de Inventario : Process, Public  
To: Administrador : External, Public

### Administrador

*External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

El administrador asume la responsabilidad de la gestión integral del inventario y la administración del sistema. Este usuario posee privilegios adicionales que le capacitan para llevar a cabo tareas más avanzadas en comparación con el vendedor. Sus responsabilidades abarcan: la gestión de usuarios, que implica crear, modificar y eliminar cuentas de usuarios en el sistema; el acceso a reportes y estadísticas para revisar y analizar el desempeño de ventas y la situación del inventario; la adición de nuevos productos al inventario, detallando información como nombre, descripción, precio y cantidad; la actualización de datos de productos existentes, como precios y descripciones; y la vigilancia constante del sistema para garantizar su correcto funcionamiento, así como resolver cualquier inconveniente técnico o administrativo que pueda surgir.

Administrador

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Detalles del producto modificado    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: Detalles de un producto que ha sido modificado por el administrador.

La estructura de datos que viaja con el flujo

Detalles del producto modificado = ID del producto modificado + Nuevos detalles del producto

↗ **DataFlow** «DFD\_DataFlow» Detalles del nuevo producto    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: Detalles completos de un nuevo producto que el administrador desea agregar al sistema.

La estructura de datos que viaja con el flujo

Detalles del nuevo producto = Nombre del nuevo producto + Descripción + Precio + inventario inicial

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord administrador    Source -> Destination

From: Adminitrador : External, Public

To: 1.- Gestionar inicio de cuenta : Process, Public

Descripción: Autenticación del administrador para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword administrador = Usuario + Contraseña

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación del producto a eliminar    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: El administrador identifica un producto específico para eliminarlo del sistema.

La estructura de datos que viaja con el flujo

Identificación del producto a eliminar = ID del producto a eliminar

↗ **DataFlow** «DFD\_DataFlow» Consulta de producto    Source -> Destination

From: Adminitrador : External, Public

To: 3.- Gestionar inventario : Process, Public

Descripción: El administrador consulta la información detallada de un producto.

La estructura de datos que viaja con el flujo

Consulta de producto = ID del producto + Nombre del producto + Descripción + Precio + Inventario disponible

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord administrador    Source -> Destination

From: Adminitrador : External, Public

To: 1.1.-Verificar tipo de cuenta : Process, Public

Descripción: Autenticación del administrador para acceder al sistema.

La estructura de datos que viaja con el flujo

Datos Login/Pasword administrador = Usuario + Contraseña

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario administrador   Source -> Destination

From: 1.1.-Verificar tipo de cuenta : Process, Public

To: Administrador : External, Public

Descripción: Mensaje de confirmación del acceso exitoso al sistema por parte de un usuario administrador.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso

↗ **DataFlow** «DFD\_DataFlow» Información del producto   Source -> Destination

From: 3.- Gestionar Inventario : Process, Public

To: Administrador : External, Public

↗ **DataFlow** «DFD\_DataFlow» Notificación producto agregado   Source -> Destination

From: 3.- Gestionar Inventario : Process, Public

To: Administrador : External, Public

Descripción: Notificación al administrador sobre la adición exitosa de un nuevo producto.

La estructura de datos que viaja con el flujo

Notificación producto agregado = ID del producto agregado + Nombre del producto + Fecha de agregación

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario administrador   Source -> Destination

From: 1.- Gestionar inicio de cuenta : Process, Public

To: Administrador : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario administrador.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario administrador = Usuario administrador + Fecha y hora de acceso

b

*External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

b

Version 1.0 Phase 1.0 Proposed

ASUS created on 21/6/2024. Last modified 21/6/2024

#### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Source -> Destination

From: a : Process, Public  
To: b : External, Public

### BD\_Inventario

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmación de agregación de producto Source -> Destination

From: 3.- Agregar Nuevo Producto : Process, Public  
To: BD\_Inventario : DataStore, Public

### BD\_Inventario

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

CONNECTORS	
 <b>DataFlow</b>	«DFD_DataFlow» actualizar inventario      Source -> Destination
From:	3.- Registrar Venta : Process, Public
To:	BD_Inventario : DataStore, Public

### BD\_Inventario

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

Descripción: Gestiona el inventario de productos.

Estructura de datos ID\_Inventario (PK) + ID\_Producto (FK hacia Producto) + Nombre + Precio + descripcion + Cantidad\_Disponible

### BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

CONNECTORS	
 <b>DataFlow</b>	«DFD_DataFlow» actualización de inventario después de modificación de venta      Source -> Destination
From:	2.- Gestionar ventas : Process, Public
To:	BD_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se modifica una venta.

La estructura de datos que viaja con el flujo  
Actualización de inventario después de modificación de venta = Productos modificados + Nuevas cantidades

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Confirmación de eliminación de producto    Source -> Destination

From: 3.- Gestionar inventario : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la eliminación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de producto = ID del producto eliminado + Fecha de eliminación

### ↗ DataFlow «DFD\_DataFlow» Actualización del inventario después de que se realiza una venta.    Source ->

Destination

From: 2.- Gestionar ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se realiza una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de venta = Productos vendidos + Cantidad vendidas

### ↗ DataFlow «DFD\_DataFlow» Confirmación de agregación de producto    Source -> Destination

From: 3.- Gestionar inventario : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la agregación de un nuevo producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de agregación de producto = ID del producto agregado + Fecha de agregación + Detalles adicionales

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Actualización de inventario después de eliminar venta   Source -> Destination

From: 2.- Gestionar ventas : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Actualización del inventario después de que se elimina una venta.

La estructura de datos que viaja con el flujo

Actualización de inventario después de eliminar venta = Productos eliminados + Cantidades eliminadas

↗ **DataFlow** «DFD\_DataFlow» Confirmación de modificación de producto   Source -> Destination

From: 3.- Gestionar Inventario : Process, Public  
To: BD\_Inventario : DataStore, Public

Descripción: Confirmación de la modificación de un producto en la base de datos de inventario.

La estructura de datos que viaja con el flujo

Confirmación de modificación de producto = ID del producto modificado + Fecha de modificación + Detalles adicionales de la modificación

## BD\_Recibo

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

Descripción: :Almacena los recibos generados para transacciones.

Estructura de datos ID\_Recibo (PK) + ID\_Venta (FK hacia Venta) + Direccion\_PDF

## BD\_Recibo

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Guardar la dirección del resivo en formato pdf   Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Recibo : DataStore, Public

Descripción: Almacenamiento de la dirección del recibo en formato PDF en la base de datos.

La estructura de datos que viaja con el flujo

Guardar la dirección del recibo en formato PDF = Productos comprados + Cantidad + Precio total + Fecha y hora de la venta + Ubicación del archivo PDF + Nombre del archivo PDF

## BD\_Resivo

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

BD\_Resivo

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Guardar la dirección del resivo en formato pdf   Source -> Destination

From: 4.- Generar Recibo : Process, Public

To: BD\_Resivo : DataStore, Public

## BD\_Usuario

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

BD\_Usuario

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Acceder al Módulo de inventario con cuenta del administrador    Source -> Destination

From: BD\_Usuario : DataStore, Public  
To: 2.- Acceder al Módulo de inventario : Process, Public

↗ **DataFlow** «DFD\_DataFlow» Confirmación de inicio de sesión como administrador    Source -> Destination

From: 1.- Iniciar Sesión como Administrador : Process, Public  
To: BD\_Usuario : DataStore, Public

## BD\_Usuario

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

Descripción: Almacena la información básica de los usuarios del sistema, incluyendo la hora de ingreso.

Estructura de datos: ID (PK) + Usuario + Fecha\_Ingreso

## BD\_Usuario

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Registrar Hora de Ingreso del Usuario    Source -> Destination

From: 1.- Gestionar inicio de cuenta : Process, Public  
To: BD\_Usuario : DataStore, Public

Descripción: Almacenamiento de la hora de ingreso del usuario en la base de datos al momento de autenticarse en el sistema.

La estructura de datos que viaja con el flujo :  
Registrar Hora de Ingreso del Usuario = Id+Usuario + Hora de ingreso

## BD\_Usuarios

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

### BD\_Usuarios

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

##### ↗ DataFlow «DFD\_DataFlow» Source -> Destination

From: 1.- Iniciar sesión como vendedor : Process, Public  
To: BD\_Usuarios : DataStore, Public

## BD\_Ventas

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

### BD\_Ventas

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

#### CONNECTORS

##### ↗ DataFlow «DFD\_DataFlow» Confirmación de venta Source -> Destination

From: 3.- Registrar Venta : Process, Public  
To: BD\_Ventas : DataStore, Public

## BD\_Ventas

*DataStore «DFD\_DataStore» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

Descripción: Registra todas las transacciones de venta realizadas.

Estructura de datos: ID\_Venta (PK) + Fecha + ID\_Vendedor (FK hacia Usuario) + Producto + precio + Cantidad + Precio\_Total

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 2/7/2024

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Confirmación de modificación de venta    Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la modificación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de modificación de venta = ID de la venta modificada + Fecha de modificación + otros detalles de la modificación

↗ DataFlow «DFD\_DataFlow» Confirmación de eliminación de venta    Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Confirmación de la eliminación de una venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Confirmación de eliminación de venta = ID de la venta eliminada + Fecha de eliminación + Razón de la eliminación

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Registro de venta    Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: BD\_Ventas : DataStore, Public

Descripción: Registro de una nueva venta en la base de datos de ventas.

La estructura de datos que viaja con el flujo

Registro de venta = ID de la venta + Fecha de la venta + Productos vendidos + Cantidad + Precio total + Forma de pago

## Cliente

*External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

Cliente

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

## CONNECTORS

↗ DataFlow «DFD\_DataFlow» Recibo impreso    Source -> Destination

From: 4.- Generar Recibo : Process, Public

To: Cliente : External, Public

## Cliente

*External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

El cliente, siendo el usuario final, interactúa directamente con el vendedor para llevar a cabo compras en la tienda. Aunque no tiene un contacto directo con el sistema de gestión de ventas e inventario, sus acciones y necesidades influyen en cómo el vendedor utiliza dicho sistema. Entre las interacciones del cliente se incluyen: seleccionar productos para su compra y efectuar el pago en la caja, solicitar información detallada sobre productos específicos, recibir un recibo impreso al completar una transacción y, en caso de devoluciones, solicitar al vendedor el procesamiento correspondiente, lo que puede implicar ajustes en el inventario.

## Cliente

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 4/7/2024

### CONNECTORS

#### ↗ DataFlow «DFD\_DataFlow» Recibo de venta    Source -> Destination

From: 2.- Gestionar ventas : Process, Public

To: Cliente : External, Public

Descripción: Recibo detallado de una venta para el cliente

La estructura de datos que viaja con el flujo

Recibo de venta = Productos comprados + Cantidad + Precio total + Detalles adicionales según configuración

## Vendedor

External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'

## Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 8/6/2024. Last modified 8/6/2024

### CONNECTORS

#### ↗ DataFlow «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public

To: 2.- Consultar precio : Process, Public

#### ↗ DataFlow «DFD\_DataFlow» Credenciales de inicio de sesión (nombre de usuario y contraseña)    Source ->

Destination

From: Vendedor : External, Public

To: 1.- Iniciar sesión como vendedor : Process, Public

## CONNECTORS

 <b>DataFlow</b> «DFD_DataFlow» Detalles de la transacción (productos comprados, precio total)      Source -> Destination From: Vendedor : External, Public To: 4.- Generar Recibo : Process, Public
 <b>DataFlow</b> «DFD_DataFlow» Detalles de la venta (productos vendidos, cantidades, precio total)      Source -> Destination From: Vendedor : External, Public To: 3.- Registrar Venta : Process, Public
 <b>DataFlow</b> «DFD_DataFlow» Información de productos (nombre, precio)      Source -> Destination From: 2.- Consultar precio : Process, Public To: Vendedor : External, Public

## Vendedor

*External «DFD\_External» owned by 'Sistema de caja NortonSoft', in package 'Business Process Model'*

El vendedor, también conocido como cajero, despliega un papel crucial como usuario principal del sistema de gestión de ventas e inventario. En el transcurso de su jornada laboral, interactúa directamente con la plataforma para llevar a cabo diversas funciones. Entre las responsabilidades del vendedor se encuentran: registrar ventas mediante el procesamiento de transacciones, consultar información sobre productos para ayudar a los clientes, generar recibos para las ventas realizadas, y actualizar el inventario disminuyendo las existencias de productos vendidos. Además, con la autorización del administrador, el vendedor puede acceder al módulo de inventario para añadir nuevos productos a la base de datos.

Vendedor

Version 1.0 Phase 1.0 Proposed

Fabian created on 19/6/2024. Last modified 4/7/2024

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Identificación de la venta a eliminar    Source -> Destination

From: Vendedor : External, Public

To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor identifica una venta específica para eliminarla del sistema.

La estructura de datos que viaja con el flujo

Identificación de la venta a eliminar = ID de la venta a eliminar

↗ **DataFlow** «DFD\_DataFlow» Consulta de precio de productos para venta    Source -> Destination

From: Vendedor : External, Public

To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor consulta el precio de los productos disponibles para realizar una venta.

La estructura de datos que viaja con el flujo

Consulta de precio de productos para venta = ID del producto + Nombre del producto + Precio del producto

↗ **DataFlow** «DFD\_DataFlow» Datos Login/Paswoord vendedor    Source -> Destination

From: Vendedor : External, Public

To: 1.- Gestionar inicio de cuenta : Process, Public

Descripción: Autenticación del vendedor para acceder al sistema y solo modulos autorizados

La estructura de datos que viaja con el flujo

Datos Login/Pasword vendedor = Usuario + Contraseña

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la transacción    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor envía los detalles de los productos comprados y el precio total de la venta.

La estructura de datos que viaja con el flujo

Detalles de la transacción = ID de la transacción + Productos comprados + Precio total  
Productos comprados = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta modificada    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: Detalles de una venta que ha sido modificada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta modificada = ID de la venta modificada + Productos vendidos + Cantidad + Precio total actualizado  
Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Criterios de búsqueda de venta    Source -> Destination

From: Vendedor : External, Public  
To: 2.- Gestionar ventas : Process, Public

Descripción: El vendedor especifica criterios para buscar ventas en el sistema.

La estructura de datos que viaja con el flujo

Criterios de búsqueda de venta = ID de la venta + Fecha de la venta + Otros criterios de búsqueda

## CONNECTORS

### ↗ DataFlow «DFD\_DataFlow» Detalles de la venta    Source -> Destination

From: 2.- Gestionar ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Detalles de una venta específica realizada por el vendedor.

La estructura de datos que viaja con el flujo

Detalles de la venta = ID de la venta + Productos vendidos + Cantidad + Precio total  
Productos vendidos = Número del producto + Nombre del producto + Cantidad + Precio por unidad

### ↗ DataFlow «DFD\_DataFlow» Notificacion de registro venta•    Source -> Destination

From: 2.- Gestionar ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Notificación al vendedor sobre el registro exitoso de una nueva venta.

La estructura de datos que viaja con el flujo

Notificación de registro de venta = ID de la venta registrada + Fecha de registro

### ↗ DataFlow «DFD\_DataFlow» Información de precio    Source -> Destination

From: 2.- Gestionar ventas : Process, Public  
To: Vendedor : External, Public

Descripción: Información detallada de productos y precio

Información de precio = Nombre del producto + Precio del producto + Descripción del producto

## CONNECTORS

↗ **DataFlow** «DFD\_DataFlow» Confirmacion de ingreso con usuario vendedor    Source -> Destination

From: 1.- Gestionar inicio de cuenta : Process, Public

To: Vendedor : External, Public

Descripción: Confirmación del acceso exitoso al sistema por parte de un usuario vendedor.

La estructura de datos que viaja con el flujo

Confirmación de ingreso con usuario vendedor = Usuario vendedor + Fecha y hora de acceso

7 Documentacion de los diagramas generado con el programa

# Model Report

Version •



Date/Time Generated: 6/8/2024 19:42:10

Author: BryaN

EA Repository : C:\Users\BryaN\AppData\Local\Microsoft\Windows\INetCache\IE\AY5NXFWE\Diagramas\_CSCACOC[1].EAP

CREATED  
WITH



## Table of Contents

Model.....	14
Use Case Model .....	14
SNorton diagram .....	14
Sistema de gestion de ventas e inventarios Nortsoft.....	14
Actors.....	15
Actors diagram .....	15
User .....	15
Diagrama de Objetos .....	16
Diagrama de Objetos diagram .....	16
Agrega mercancia a la tienda.....	16
Agregar un pantalon .....	17
Cambiar la cantidad de pantalones que se llevara .....	18
Cambiar un nuevo pantalon .....	19
Eliminar el Vestido.....	20
Eliminar venta que no se completo.....	20
Ingresa el Usuario .....	20
Realizar la venta .....	21
Rebicion cuanto se vendio en el dia.....	22
Rebicion de los articulos .....	22
Revisar la venta de hace un momento .....	23
consulta por las camisas .....	23
Diagrama de actividades.....	25
Diagrama de actividades diagram.....	25
Mensaje: No existe ningun producto disponible .....	26

Administrador.....	27
Busca un producto en la tabla y selecciona uno (doble click).....	27
Finaliza la compra y da click en "Generar Venta" .....	28
Ingresa la cantidad deseada .....	28
Accede al apartado para registrar venta .....	28
BD_Inventario .....	28
Actualiza el stock en base a la cantidad comprada .....	29
Consulta todos los productos disponibles .....	29
ObjectNode1 .....	29
BD_Ventas .....	30
Registra la venta.....	30
Sistema .....	30
Añade productos al carrito, actualiza tabla y calcula el precio .....	31
Envia la lista de productos.....	31
Ingresa a la interfaz .....	31
Muestra la tabla de productos con la opcion de busqueda .....	32
Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad	32
¿Existen productos disponibles?.....	33
Confirma que la venta fue registrada .....	33
¿Solicita mas productos? .....	33
Busca un producto en la tabla y selecciona uno (doble click).....	33
Finaliza la compra y da click en "Generar Venta" .....	34
Ingresa la cantidad deseada .....	34
Accede al apartado para registrar venta .....	34
Añade productos al carrito, actualiza tabla y calcula el precio .....	35
Envia la lista de productos.....	35
Ingresa a la interfaz .....	35
Muestra la tabla de productos con la opcion de busqueda .....	36
Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad	36
¿Existen productos disponibles?.....	36

Confirma que la venta fue registrada .....	37
Actualiza el stock en base a la cantidad comprada .....	37
Consulta todos los productos disponibles .....	37
ObjectNode1 .....	38
Registra la venta .....	38
Synchronization.....	38
Synchronization.....	39
Synchronization.....	39
Diagrama de componentes.....	40
Diagrama de componentes diagram.....	40
Admin.....	40
Base de Datos inventario.....	41
Base de Datos Ventas .....	41
Login .....	41
Menu de Modulos.....	42
Modulo inventario.....	42
Modulo Venta.....	43
Usuario .....	43
Admin.....	44
Generacion de reportes diagram .....	44
Base de Datos inventario .....	45
Base de Datos inventario diagram .....	45
Base de Datos Ventas .....	46
Base de Datos Ventas diagram .....	46
Usuario .....	47
Usuario diagram .....	47
Login .....	48
Generacion de reportes diagram .....	48
Autenticacion de Usuarios.....	48

Comprobacion de Usuarios .....	49
Validar si es Admin.....	49
Validar si es Usuario .....	50
Menu de Modulos.....	51
Menu de Modulos diagram.....	51
Modulo inventario.....	51
Modulo venta .....	51
Modulo inventario.....	53
Inventario diagram .....	53
Actualizar producto .....	53
Agregar Producto .....	54
Consultar Producto.....	54
Eliminar Producto .....	54
Modulo Venta.....	55
Venta diagram .....	55
Actualizar Ventas .....	55
Agregar una Venta.....	56
Consultar Ventas .....	56
Eliminar Ventas .....	56
Generar reporte de stock.....	56
diagrama de clases.....	57
diagrama de clases diagram .....	57
Actualizar productos.....	58
Actualizar ventas .....	58
Agregar Ventas.....	59
Agregar productos .....	60
Consultar productos.....	61
Consultar ventas .....	62
Eliminar productos .....	62
Eliminar venta .....	62

Login .....	63
Menu Modulos .....	64
Modulo Ventas .....	65
Modulo inventario .....	66
diagrama de colaboracion .....	68
Actualizar Productos diagram.....	68
Object.....	68
BD_Inventario .....	69
Sistema.....	70
Actualizar Ventas diagram .....	71
Object.....	72
Object.....	73
BD_Ventas .....	73
Sistema.....	74
Consultar Productos diagram.....	76
Object.....	77
BD_Inventario .....	78
Sistema.....	78
Consultar Ventas diagram .....	79
Object.....	80
Object.....	81
BD_Ventas .....	81
Sistema.....	82
Eliminar Productos diagram .....	84
Object.....	84
BD_Inventario .....	85
Sistema.....	86

Eliminar Ventas diagram .....	87
Object.....	88
Object.....	89
BD_Ventas .....	89
Sistema.....	90
Ingresar nuevos Productos diagram.....	92
Object.....	92
BD_Inventario .....	93
Sistema.....	94
Registrar Ventas diagram .....	95
Object.....	96
Object.....	97
BD__Inventario .....	97
BD_Ventas .....	98
Sistema.....	98
Casos de uso.....	102
Actualizar Información de Productos diagram .....	102
Actualizar Ventas diagram .....	102
Aregar venta diagram .....	103
Consultar Productos diagram.....	103
Consultar Ventas diagram .....	104
Eliminar Productos diagram .....	104
Eliminar Venta diagram.....	105
Ingresra nuevo producto diagram .....	105
Actors .....	106
Actors diagram .....	106
User .....	106
Primary Use Cases.....	108

Primary Use Cases diagram.....	108
User.....	108
System Boundary .....	109
Note.....	109
Use Case1.....	109
Use Case1 diagram.....	110
User.....	110
Object1.....	111
Use Case2.....	111
Actor1.....	111
Actualizar .....	111
Administrador.....	112
Administrador.....	112
Administrador.....	112
Administrador.....	113
Usuario .....	113
Vendedor.....	114
Vendedor.....	114
Vendedor.....	114
Seleccionar Producto.....	115
Acceder a Eliminar Ventas .....	115
Acceder a al modulo de Consultar Ventas .....	115
Acceder a l modulo Actualizar Ventas .....	116
Acceder al Módulo de Inventario .....	116
Acceder al Módulo de Inventario .....	116
Acceder al apartado de actualizar .....	116
Acceder al apartado de eliminar productos.....	117

Actualizar Información de Productos .....	117
Actualizar Venta.....	118
Agregar al Carrito.....	119
Agregar venta.....	119
Buscar Producto .....	120
Buscar Producto .....	121
Buscar Producto .....	121
Buscar Venta .....	121
Buscar Venta .....	122
Buscar Venta .....	122
Buscar producto solicitado .....	122
Cobrar al Cliente .....	122
Confirmar Cambios .....	123
Confirmar Cambios: .....	123
Confirmar Eliminación .....	123
Confirmar Eliminación .....	124
Confirmar Ingreso .....	124
Consultar Productos .....	124
Consultar Ventas .....	125
Eliminar Productos .....	126
Eliminar Venta .....	126
Eliminar Venta: .....	127
Entregar Recibo al Cliente.....	127
Generar Venta .....	128
Guardar Cambios.....	128
Guardar Producto .....	128
Ingresar Cantidad .....	129

Ingresar Datos del Producto .....	129
Ingresar nuevo producto .....	129
Modificar Información .....	130
Modificar Información .....	130
Repetir para Otros Productos.....	131
Seleccionar Producto .....	131
Seleccionar Producto.....	131
Seleccionar Producto.....	131
Seleccionar Venta.....	132
Seleccionar Venta.....	132
Seleccionar Venta.....	132
Ver Detalles de Venta.....	133
Ver Detalles del Producto.....	133
Object.....	133
Object.....	133
Object.....	134
Diagrama de secuencias .....	135
Actualizar Venta diagram.....	135
Vendedor.....	136
Administrador .....	138
Administrador.....	141
Actualizar productos diagram.....	145
Administrador.....	146
Consultar Productos diagram.....	150
Vendedor.....	150
Administrador .....	152
Consultar ventas diagram .....	156
Vendedor.....	157

Administrador .....	159
Administrador.....	162
Eliminar productos diagram .....	166
Administrador.....	166
Eliminar ventas diagram.....	170
Vendedor.....	171
Administrador .....	173
Administrador.....	176
Ingresar Nuevos Productos diagram .....	180
Administrador.....	180
Registrar venta diagram.....	184
Vendedor.....	185
Administrador .....	187
Administrador.....	190
Domain Objects.....	194
Domain Objects diagram .....	194
Note.....	195
Class1 .....	195
Class2 .....	195
Class3 .....	196
Actor1.....	196
Vendedor.....	196
BD_Inventario .....	196
BD_Inventario .....	197
BD_Inventario .....	197
BD_Inventario .....	198
BD_Inventraio .....	198
BD_Ventas .....	199
BD_Ventas .....	199

BD_Ventas .....	200
BD_Ventas .....	200
Mientras el administrador quiera actualizar indormacion de productos .....	201
Mientras el administrador quiera agregar un producto .....	201
Mientras el cliente solicite mas productos .....	201
Mientras el usuario o administrador quieran actualizar la informacion de una venta	201
Mientras el usuario quiera consultar informacion de productos.....	201
Mientras el usuario quiera eliminar un producto .....	202
Mientras el vendedor o administrador desen consultar informacion de una venta....	202
Mientras el vendedor o administrador quiers eliminar una venta.....	202
Object1 .....	202
Object2 .....	202
Object3 .....	202
Sistema .....	203
Sistema .....	204
Sistema .....	205
Sistema .....	206
Sistema .....	207
Sistema .....	208
Sistema .....	209
Sistema .....	210
Vendedor.....	212
Administrador.....	214
UseCase.....	217
Gestionar inio de cuenta .....	217
Gestionar inventario .....	217
Gestionar inventario diagram .....	218

Gestionar venta.....	218
Gestionar venta diagram.....	219
Administrador.....	220
Actualizar venta.....	220
Agregar venta .....	221
Bucar precio del producto por ID .....	221
Consultar venta.....	222
Eliminar venta .....	222
Generar recibo .....	222
Gestion venta.....	223
Verficar que el id de laventa exista.....	223
Object.....	224
Object.....	225
Object.....	225
Object.....	226
Object.....	227
Object.....	227
Object.....	228
Object.....	229
Object.....	229
Object.....	230
Object.....	231
Object.....	231
BD__Inventario.....	232
BD_Inventario.....	233
BD_Inventario.....	233
BD_Inventario.....	234

BD_Inventario.....	234
BD_Ventas.....	235
BD_Ventas.....	236
BD_Ventas.....	236
BD_Ventas.....	237
Object1 .....	238
Object2 .....	238
Sistema.....	238
Sistema.....	241
Sistema.....	242
Sistema.....	244
Sistema.....	246
Sistema.....	248
Sistema.....	249
Sistema.....	250
Mensaje: No existe ningun producto disponible.....	253
Administrador .....	253
Busca un producto en la tabla y selecciona uno (doble click).....	253
Finaliza la compra y da click en "Generar Venta" .....	254
Ingresa la cantidad deseada .....	254
Accede al apartado para registrar venta .....	254
Agregar Venta .....	255
BD_Inventario.....	255
Actualiza el stock en base a la cantidad comprada .....	255
Consulta todos los productos disponibles .....	255
ObjectNode1 .....	256
BD_Ventas.....	256
Registra la venta .....	256

Sistema .....	256
Añade productos al carrito, actualiza tabla y calcula el precio .....	257
Envia la lista de productos.....	257
Ingresa a la interfaz .....	258
Muestra la tabla de productos con la opcion de busqueda .....	258
Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad .....	259
¿Existen productos disponibles?.....	259
Confirma que la venta fue registrada .....	259
Sistema .....	259
Sistema .....	260
¿Solicita mas productos? .....	260

# Model

Package in package "

Model

Version Phase 1.0 Proposed

BryaN created on 6/8/2024. Last modified 6/8/2024

## Use Case Model

Package in package 'Model'

Use Case Model

Version Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## SNorton diagram

Use Case diagram in package 'Use Case Model'

SNorton

Version 1.0

ASUS created on 11/7/2024. Last modified 19/7/2024

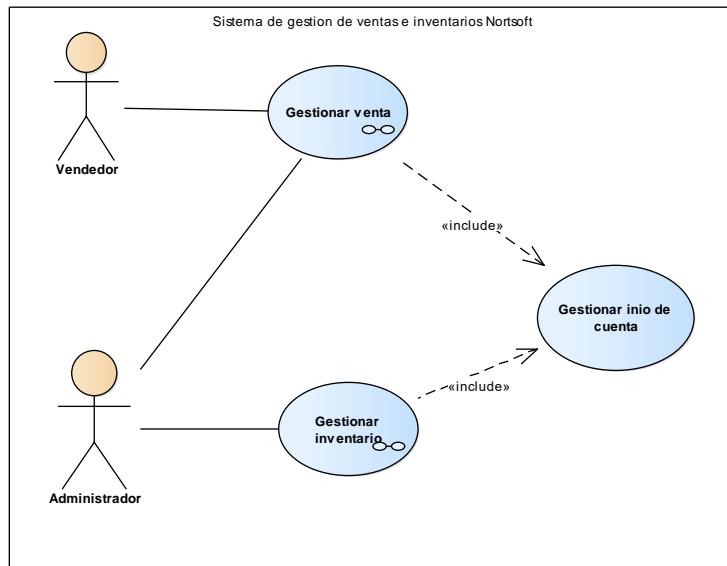


Figure 1: SNorton

## Sistema de gestión de ventas e inventarios Nortsoft

Boundary in package 'Use Case Model'

Sistema de gestión de ventas e inventarios Nortsoft

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

Extends

## Actors

Package in package 'Use Case Model'

Actors  
Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 11/7/2024

### Actors diagram

Use Case diagram in package 'Actors'

Actors  
Version 1.0  
BryaN created on 11/7/2024. Last modified 11/7/2024

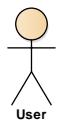


Figure 2: Actors

## User

Actor in package 'Actors'

User  
Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 11/7/2024

## Diagrama de Objetos

Package in package 'Use Case Model'

Diagrama de Objetos  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 5/8/2024. Last modified 5/8/2024

### Diagrama de Objetos diagram

Object diagram in package 'Diagrama de Objetos'

Diagrama de Objetos  
Version 1.0  
Fabian created on 5/8/2024. Last modified 6/8/2024

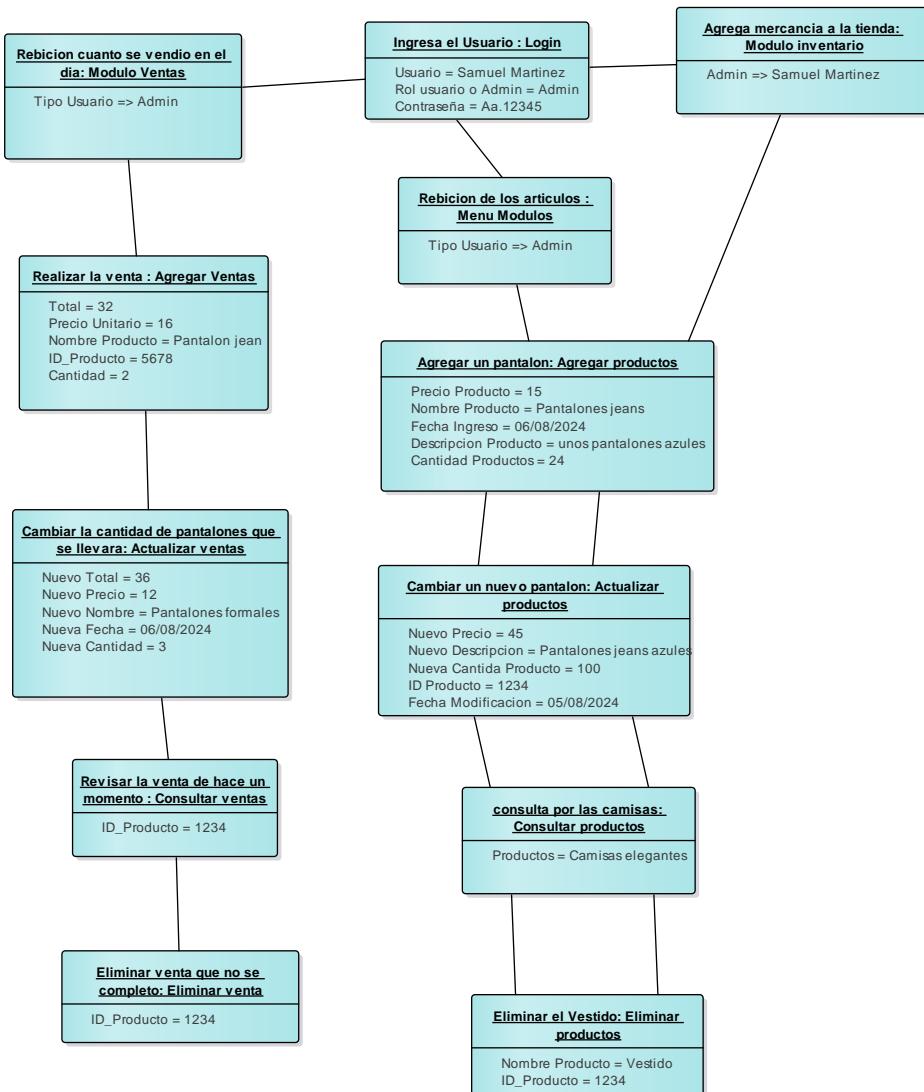


Figure 3: Diagrama de Objetos

### Agrega mercancia a la tienda

Object in package 'Diagrama de Objetos'

Agrega mercancia a la tienda  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 6/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
Association (direction: Unspecified)	Source: Public (Object) Agrega mercancia a la tienda	Target: Public (Object) Agregar un pantalon	
Association (direction: Unspecified)	Source: Public (Object) Ingresa el Usuario	Target: Public (Object) Agrega mercancia a la tienda	

RUN STATES	OPERATOR	VALUE	NOTES
Admin	=>	Samuel Martinez	

**Agregar un pantalon**  
Object in package 'Diagrama de Objetos'

Agregar un pantalon  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 6/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
Association (direction: Unspecified)	Source: Public (Object) Agregar un pantalon	Target: Public (Object) Cambiar un nuevo pantalon	
Association (direction: Unspecified)	Source: Public (Object) Agregar un pantalon	Target: Public (Object) Cambiar un nuevo pantalon	
Association (direction: Unspecified)	Source: Public (Object) Agrega mercancia a la tienda	Target: Public (Object) Agregar un pantalon	
Association (direction: Unspecified)	Source: Public (Object) Rebicion de los articulos	Target: Public (Object) Agregar un pantalon	

RUN STATES	OPERATOR	VALUE	NOTES
Cantidad	=	24	

Productos			
🏃 Descripcion Producto	=	unos pantalones azules	
🏃 Fecha Ingreso	=	06/08/2024	
🏃 Nombre Producto	=	Pantalones jeans	
🏃 Precio Producto	=	15	

### Cambiar la cantidad de pantalones que se llevara

Object in package 'Diagrama de Objetos'

Cambiar la cantidad de pantalones que se llevara

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
✍ Association (direction: Unspecified)	Source: Public (Object) Cambiar la cantidad de pantalones que se llevara	Target: Public (Object) Revisar la venta de hace un momento	
✍ Association (direction: Unspecified)	Source: Public (Object) Realizar la venta	Target: Public (Object) Cambiar la cantidad de pantalones que se llevara	

RUN STATES	OPERATOR	VALUE	NOTES
🏃 Nueva Cantidad	=	3	
🏃 Nueva Fecha	=	06/08/2024	
🏃 Nuevo Nombre	=	Pantalones formales	
🏃 Nuevo Precio	=	12	
🏃 Nuevo Total	=	36	

## Cambiar un nuevo pantalon

Object in package 'Diagrama de Objetos'

Cambiar un nuevo pantalon

Version 1.0 Phase 1.0 Proposed

Fabian created on 5/8/2024. Last modified 6/8/2024

### ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Cambiar un nuevo pantalon

Target: Public (Object) consulta por las camisas

✓ Association (direction: Unspecified)

Source: Public (Object) Cambiar un nuevo pantalon

Target: Public (Object) consulta por las camisas

✓ Association (direction: Unspecified)

Source: Public (Object) Agregar un pantalon

Target: Public (Object) Cambiar un nuevo pantalon

✓ Association (direction: Unspecified)

Source: Public (Object) Agregar un pantalon

Target: Public (Object) Cambiar un nuevo pantalon

RUN STATES	OPERATOR	VALUE	NOTES
🏃 Fecha Modificacion	=	05/08/2024	
🏃 ID Producto	=	1234	
🏃 Nueva Cantida Producto	=	100	
🏃 Nuevo Descripcion	=	Pantalones jeans azules	
🏃 Nuevo Precio	=	45	

## Eliminar el Vestido

Object in package 'Diagrama de Objetos'

Eliminar el Vestido

Version 1.0 Phase 1.0 Proposed  
Fabian created on 5/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
 Association (direction: Unspecified)			
Source: Public (Object) consulta por las camisas		Target: Public (Object) Eliminar el Vestido	
 Association (direction: Unspecified)			
Source: Public (Object) consulta por las camisas		Target: Public (Object) Eliminar el Vestido	

RUN STATES	OPERATOR	VALUE	NOTES
 ID_Producto	=	1234	
 Nombre Producto	=	Vestido	

**Eliminar venta que no se completo**  
Object in package 'Diagrama de Objetos'

Eliminar venta que no se completo  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 6/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
 Association (direction: Unspecified)			
Source: Public (Object) Revisar la venta de hace un momento		Target: Public (Object) Eliminar venta que no se completo	

RUN STATES	OPERATOR	VALUE	NOTES
 ID_Producto	=	1234	

**Ingresar el Usuario**  
Object in package 'Diagrama de Objetos'

Ingresar el Usuario  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 5/8/2024. Last modified 5/8/2024

ASSOCIATIONS			
 Association (direction: Unspecified)			

## ASSOCIATIONS

Source: Public (Object) Ingresa el Usuario

Target: Public (Object) Agrega mercancia a la tienda

Association (direction: Unspecified)

Source: Public (Object) Ingresa el Usuario

Target: Public (Object) Rebicion de los articulos

Association (direction: Unspecified)

Source: Public (Object) Ingresa el Usuario

Target: Public (Object) Rebicion cuanto se vendio en el dia

RUN STATES	OPERATOR	VALUE	NOTES
Contraseña	=	Aa.12345	
Rol usuario o Admin	=	Admin	
Usuario	=	Samuel Martinez	

## Realizar la venta

Object in package 'Diagrama de Objetos'

Realizar la venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Object) Realizar la venta

Target: Public (Object) Cambiar la cantidad de pantalones que se llevara

Association (direction: Unspecified)

Source: Public (Object) Rebicion cuanto se vendio en el dia

Target: Public (Object) Realizar la venta

RUN STATES	OPERATOR	VALUE	NOTES
Cantidad	=	2	
ID_Producto	=	5678	

↖ Nombre Producto	=	Pantalon jean	
↖ Precio Unitario	=	16	
↖ Total	=	32	

### Rebicion cuanto se vendio en el dia

Object in package 'Diagrama de Objetos'

Rebicion cuanto se vendio en el dia

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
↖ Association (direction: Unspecified)			
Source: Public (Object) Rebicion cuanto se vendio en el dia		Target: Public (Object) Realizar la venta	

↖ Association (direction: Unspecified)			
Source: Public (Object) Ingresa el Usuario		Target: Public (Object) Rebicion cuanto se vendio en el dia	

RUN STATES	OPERATOR	VALUE	NOTES
↖ Tipo Usuario	=>	Admin	

### Rebicion de los articulos

Object in package 'Diagrama de Objetos'

Rebicion de los articulos

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

ASSOCIATIONS			
↖ Association (direction: Unspecified)			
Source: Public (Object) Rebicion de los articulos		Target: Public (Object) Agregar un pantalon	

↖ Association (direction: Unspecified)			
Source: Public (Object) Ingresa el Usuario		Target: Public (Object) Rebicion de los articulos	

## ASSOCIATIONS

RUN STATES	OPERATOR	VALUE	NOTES
 Tipo Usuario	=>	Admin	

### Revisar la venta de hace un momento

Object in package 'Diagrama de Objetos'

Revisar la venta de hace un momento

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## ASSOCIATIONS

-  Association (direction: Unspecified)

Source: Public (Object) Revisar la venta de hace un momento

Target: Public (Object) Eliminar venta que no se completo

-  Association (direction: Unspecified)

Source: Public (Object) Cambiar la cantidad de pantalones que se llevara

Target: Public (Object) Revisar la venta de hace un momento

RUN STATES	OPERATOR	VALUE	NOTES
 ID_Producto	=	1234	

### consulta por las camisas

Object in package 'Diagrama de Objetos'

consulta por las camisas

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## ASSOCIATIONS

-  Association (direction: Unspecified)

Source: Public (Object) consulta por las camisas

Target: Public (Object) Eliminar el Vestido

-  Association (direction: Unspecified)

Source: Public (Object) consulta por las camisas

Target: Public (Object) Eliminar el Vestido

## ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Object) Cambiar un nuevo pantalon

Target: Public (Object) consulta por las camisas

 Association (direction: Unspecified)

Source: Public (Object) Cambiar un nuevo pantalon

Target: Public (Object) consulta por las camisas

RUN STATES	OPERATOR	VALUE	NOTES
 Productos	=	Camisas elegantes	

**Diagrama de actividades**  
Package in package 'Use Case Model'

Diagrama de actividades  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 4/8/2024. Last modified 4/8/2024

**Diagrama de actividades diagram**  
Activity diagram in package 'Diagrama de actividades'

Diagrama de actividades  
Version 1.0  
Fabian created on 3/8/2024. Last modified 3/8/2024

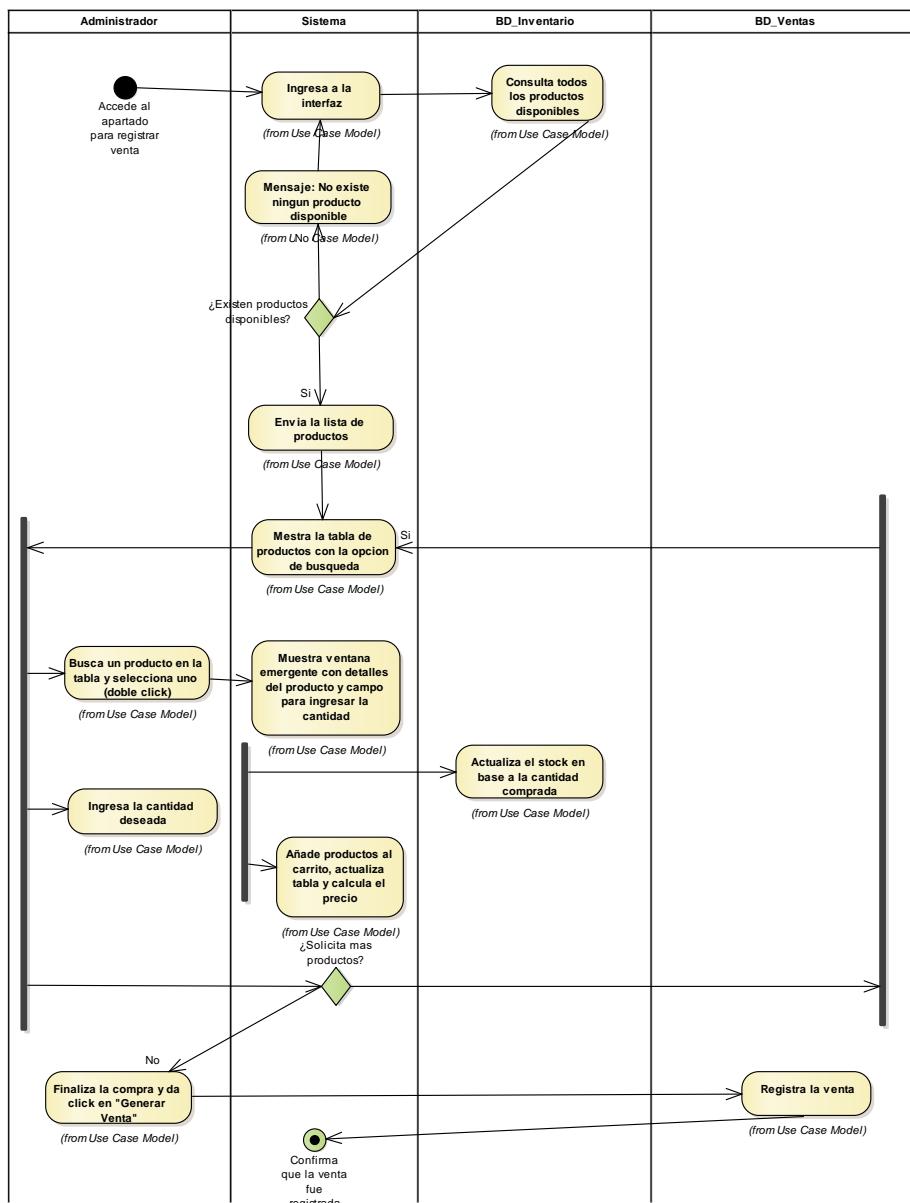


Figure 4: Diagrama de actividades

## Mensaje: No existe ningun producto disponible

Activity in package 'Use Case Model'

Mensaje: No existe ningun producto disponible

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Mensaje: No existe ningun producto disponible to Ingres a la interfaz

### INCOMING BEHAVIORAL RELATIONSHIPS

Name: No

⇒ ControlFlow from ¿Existen productos disponibles? to Mensaje: No existe ningun producto disponible

## Administrador

ActivityPartition in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### ELEMENTS OWNED BY Administrador

█ Busca un producto en la tabla y selecciona uno (doble click) : Activity

█ Finaliza la compra y da click en "Generar Venta" : Activity

█ Ingresa la cantidad deseada : Activity

█ Accede al apartado para registrar venta : ActivityInitial

## Busca un producto en la tabla y selecciona uno (doble click)

Activity owned by 'Administrador', in package 'Use Case Model'

Busca un producto en la tabla y selecciona uno (doble click)

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Busca un producto en la tabla y selecciona uno (doble click) to Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Busca un producto en la tabla y selecciona uno (doble click)

## Finaliza la compra y da click en "Generar Venta"

Activity owned by 'Administrador', in package 'Use Case Model'

Finaliza la compra y da click en "Generar Venta"

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Finaliza la compra y da click en "Generar Venta" to Registra la venta

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: No

⇒ ControlFlow from ¿Solicita mas productos? to Finaliza la compra y da click en "Generar Venta"

## Ingresá la cantidad deseada

Activity owned by 'Administrador', in package 'Use Case Model'

Ingresá la cantidad deseada

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Ingresá la cantidad deseada

## Accede al apartado para registrar venta

ActivityInitial owned by 'Administrador', in package 'Use Case Model'

## OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Accede al apartado para registrar venta to Ingresá a la interfaz

## BD\_Inventario

ActivityPartition in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

#### ELEMENTS OWNED BY BD\_Inventario

▣ Actualiza el stock en base a la cantidad comprada : Activity

▣ Consulta todos los productos disponibles : Activity

### Actualiza el stock en base a la cantidad comprada

Activity owned by 'BD\_Inventario', in package 'Use Case Model'

Actualiza el stock en base a la cantidad comprada

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Actualiza el stock en base a la cantidad comprada

### Consulta todos los productos disponibles

Activity owned by 'BD\_Inventario', in package 'Use Case Model'

Consulta todos los productos disponibles

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

#### STRUCTURAL PART OF Consulta todos los productos disponibles

✿ ObjectNode1 : ActionPin

#### OUTGOING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Consulta todos los productos disponibles to ¿Existen productos disponibles?

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Ingresa a la interfaz to Consulta todos los productos disponibles

### ObjectNode1

ActionPin owned by 'Consulta todos los productos disponibles', in package 'Use Case Model'

## **BD\_Ventas**

ActivityPartition in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### **ELEMENTS OWNED BY BD\_Ventas**

- Registra la venta : Activity

## **Registra la venta**

Activity owned by 'BD\_Ventas', in package 'Use Case Model'

Registra la venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### **OUTGOING BEHAVIORAL RELATIONSHIPS**

- ↳ ControlFlow from Registra la venta to Confirma que la venta fue registrada

### **INCOMING BEHAVIORAL RELATIONSHIPS**

- ⇒ ControlFlow from Finaliza la compra y da click en "Generar Venta" to Registra la venta

## **Sistema**

ActivityPartition in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### **ELEMENTS OWNED BY Sistema**

- Añade productos al carrito, actualiza tabla y calcula el precio : Activity
- Envia la lista de productos : Activity
- Ingresá a la interfaz : Activity
- Muestra la tabla de productos con la opción de búsqueda : Activity

## ELEMENTS OWNED BY Sistema

- Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad : Activity
- ¿Existen productos disponibles? : DecisionNode
- Confirma que la venta fue registrada : ActivityFinal

## Añade productos al carrito, actualiza tabla y calcula el precio

Activity owned by 'Sistema', in package 'Use Case Model'

Añade productos al carrito, actualiza tabla y calcula el precio

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## INCOMING BEHAVIORAL RELATIONSHIPS

- ⇒ ControlFlow from Synchronization to Añade productos al carrito, actualiza tabla y calcula el precio

## Envia la lista de productos

Activity owned by 'Sistema', in package 'Use Case Model'

Envia la lista de productos

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

- ⇐ ControlFlow from Envia la lista de productos to Muestra la tabla de productos con la opcion de busqueda

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Si

- ⇒ ControlFlow from ¿Existen productos disponibles? to Envia la lista de productos

## Ingresa a la interfaz

Activity owned by 'Sistema', in package 'Use Case Model'

Ingresa a la interfaz

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Ingresa a la interfaz to Consulta todos los productos disponibles

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Accede al apartado para registrar venta to Ingresa a la interfaz

⇒ ControlFlow from Mensaje: No existe ningun producto disponible to Ingresa a la interfaz

## Muestra la tabla de productos con la opcion de busqueda

Activity owned by 'Sistema', in package 'Use Case Model'

Muestra la tabla de productos con la opcion de busqueda

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Muestra la tabla de productos con la opcion de busqueda to Synchronization

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Envia la lista de productos to Muestra la tabla de productos con la opcion de busqueda

Name: Si

⇒ ControlFlow from Synchronization to Muestra la tabla de productos con la opcion de busqueda

## Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

Activity owned by 'Sistema', in package 'Use Case Model'

Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Busca un producto en la tabla y selecciona uno (doble click) to Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

## ¿Existen productos disponibles?

DecisionNode owned by 'Sistema', in package 'Use Case Model'

OUTGOING BEHAVIORAL RELATIONSHIPS
Name: Si ↳ ControlFlow from ¿Existen productos disponibles? to Envia la lista de productos
Name: No ↳ ControlFlow from ¿Existen productos disponibles? to Mensaje: No existe ningun producto disponible

INCOMING BEHAVIORAL RELATIONSHIPS
⇒ ControlFlow from Consulta todos los productos disponibles to ¿Existen productos disponibles?

## Confirma que la venta fue registrada

ActivityFinal owned by 'Sistema', in package 'Use Case Model'

INCOMING BEHAVIORAL RELATIONSHIPS
⇒ ControlFlow from Registra la venta to Confirma que la venta fue registrada

## ¿Solicita mas productos?

DecisionNode in package 'Use Case Model'

OUTGOING BEHAVIORAL RELATIONSHIPS
↳ ControlFlow from ¿Solicita mas productos? to Synchronization
Name: No ↳ ControlFlow from ¿Solicita mas productos? to Finaliza la compra y da click en "Generar Venta"

INCOMING BEHAVIORAL RELATIONSHIPS
⇒ ControlFlow from Synchronization to ¿Solicita mas productos?

## Busca un producto en la tabla y selecciona uno (doble click)

Activity owned by 'Administrador', in package 'Use Case Model'

Busca un producto en la tabla y selecciona uno (doble click)

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Busca un producto en la tabla y selecciona uno (doble click) to Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Busca un producto en la tabla y selecciona uno (doble click)

### Finaliza la compra y da click en "Generar Venta"

Activity owned by 'Administrador', in package 'Use Case Model'

Finaliza la compra y da click en "Generar Venta"

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Finaliza la compra y da click en "Generar Venta" to Registra la venta

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: No

⇒ ControlFlow from ¿Solicita mas productos? to Finaliza la compra y da click en "Generar Venta"

### Ingresa la cantidad deseada

Activity owned by 'Administrador', in package 'Use Case Model'

Ingresa la cantidad deseada

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Ingresa la cantidad deseada

### Accede al apartado para registrar venta

ActivityInitial owned by 'Administrador', in package 'Use Case Model'

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Accede al apartado para registrar venta to Ingresa a la interfaz

## Añade productos al carrito, actualiza tabla y calcula el precio

Activity owned by 'Sistema', in package 'Use Case Model'

Añade productos al carrito, actualiza tabla y calcula el precio

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### INCOMING BEHAVIORAL RELATIONSHIPS

- ⇒ ControlFlow from Synchronization to Añade productos al carrito, actualiza tabla y calcula el precio

## Envia la lista de productos

Activity owned by 'Sistema', in package 'Use Case Model'

Envia la lista de productos

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

- ⇐ ControlFlow from Envia la lista de productos to Muestra la tabla de productos con la opcion de busqueda

### INCOMING BEHAVIORAL RELATIONSHIPS

Name: Si

- ⇒ ControlFlow from ¿Existen productos disponibles? to Envia la lista de productos

## Ingresa a la interfaz

Activity owned by 'Sistema', in package 'Use Case Model'

Ingresa a la interfaz

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

- ⇐ ControlFlow from Ingresa a la interfaz to Consulta todos los productos disponibles

### INCOMING BEHAVIORAL RELATIONSHIPS

- ⇒ ControlFlow from Accede al apartado para registrar venta to Ingresa a la interfaz

- ⇒ ControlFlow from Mensaje: No existe ningun producto disponible to Ingresa a la interfaz

## Muestra la tabla de productos con la opcion de busqueda

Activity owned by 'Sistema', in package 'Use Case Model'

Muestra la tabla de productos con la opcion de busqueda

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Muestra la tabla de productos con la opcion de busqueda to Synchronization

### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Envia la lista de productos to Muestra la tabla de productos con la opcion de busqueda

Name: Si

⇒ ControlFlow from Synchronization to Muestra la tabla de productos con la opcion de busqueda

## Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

Activity owned by 'Sistema', in package 'Use Case Model'

Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Busca un producto en la tabla y selecciona uno (doble click) to Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

## ¿Existen productos disponibles?

DecisionNode owned by 'Sistema', in package 'Use Case Model'

### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Si

↳ ControlFlow from ¿Existen productos disponibles? to Envia la lista de productos

Name: No

↳ ControlFlow from ¿Existen productos disponibles? to Mensaje: No existe ningun producto disponible

### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Consulta todos los productos disponibles to ¿Existen productos disponibles?

## INCOMING BEHAVIORAL RELATIONSHIPS

### Confirma que la venta fue registrada

ActivityFinal owned by 'Sistema', in package 'Use Case Model'

## INCOMING BEHAVIORAL RELATIONSHIPS

- ⇒ ControlFlow from Registra la venta to Confirma que la venta fue registrada

### Actualiza el stock en base a la cantidad comprada

Activity owned by 'BD\_Inventario', in package 'Use Case Model'

Actualiza el stock en base a la cantidad comprada

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## INCOMING BEHAVIORAL RELATIONSHIPS

- ⇒ ControlFlow from Synchronization to Actualiza el stock en base a la cantidad comprada

### Consulta todos los productos disponibles

Activity owned by 'BD\_Inventario', in package 'Use Case Model'

Consulta todos los productos disponibles

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## STRUCTURAL PART OF Consulta todos los productos disponibles

- ✿ ObjectNode1 : ActionPin

## OUTGOING BEHAVIORAL RELATIONSHIPS

- ◀ ControlFlow from Consulta todos los productos disponibles to ¿Existen productos disponibles?

## INCOMING BEHAVIORAL RELATIONSHIPS

- ⇒ ControlFlow from Ingresa a la interfaz to Consulta todos los productos disponibles

## ObjectNode1

ActionPin owned by 'Consulta todos los productos disponibles', in package 'Use Case Model'

## **Registra la venta**

Activity owned by 'BD\_Ventas', in package 'Use Case Model'

Registra la venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### **OUTGOING BEHAVIORAL RELATIONSHIPS**

↳ ControlFlow from Registra la venta to Confirma que la venta fue registrada

### **INCOMING BEHAVIORAL RELATIONSHIPS**

↗ ControlFlow from Finaliza la compra y da click en "Generar Venta" to Registra la venta

## **Synchronization**

Synchronization in package 'Diagrama de actividades'

### **OUTGOING BEHAVIORAL RELATIONSHIPS**

↳ ControlFlow from Synchronization to ¿Solicita mas productos?

↳ ControlFlow from Synchronization to Busca un producto en la tabla y selecciona uno (doble click)

↳ ControlFlow from Synchronization to Ingresa la cantidad deseada

### **INCOMING BEHAVIORAL RELATIONSHIPS**

↗ ControlFlow from Muestra la tabla de productos con la opcion de busqueda to Synchronization

## **Synchronization**

Synchronization in package 'Diagrama de actividades'

### **OUTGOING BEHAVIORAL RELATIONSHIPS**

↳ ControlFlow from Synchronization to Añade productos al carrito, actualiza tabla y calcula el precio

↳ ControlFlow from Synchronization to Actualiza el stock en base a la cantidad comprada

## Synchronization

Synchronization in package 'Diagrama de actividades'

### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Si

↳ ControlFlow from Synchronization to Muestra la tabla de productos con la opcion de busqueda

### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from ¿Solicita mas productos? to Synchronization

## Diagrama de componentes

Package in package 'Use Case Model'

Diagrama de componentes  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 6/8/2024. Last modified 6/8/2024

## Diagrama de componentes diagram

Component diagram in package 'Diagrama de componentes'

Diagrama de componentes  
Version 1.0  
Fabian created on 6/8/2024. Last modified 6/8/2024

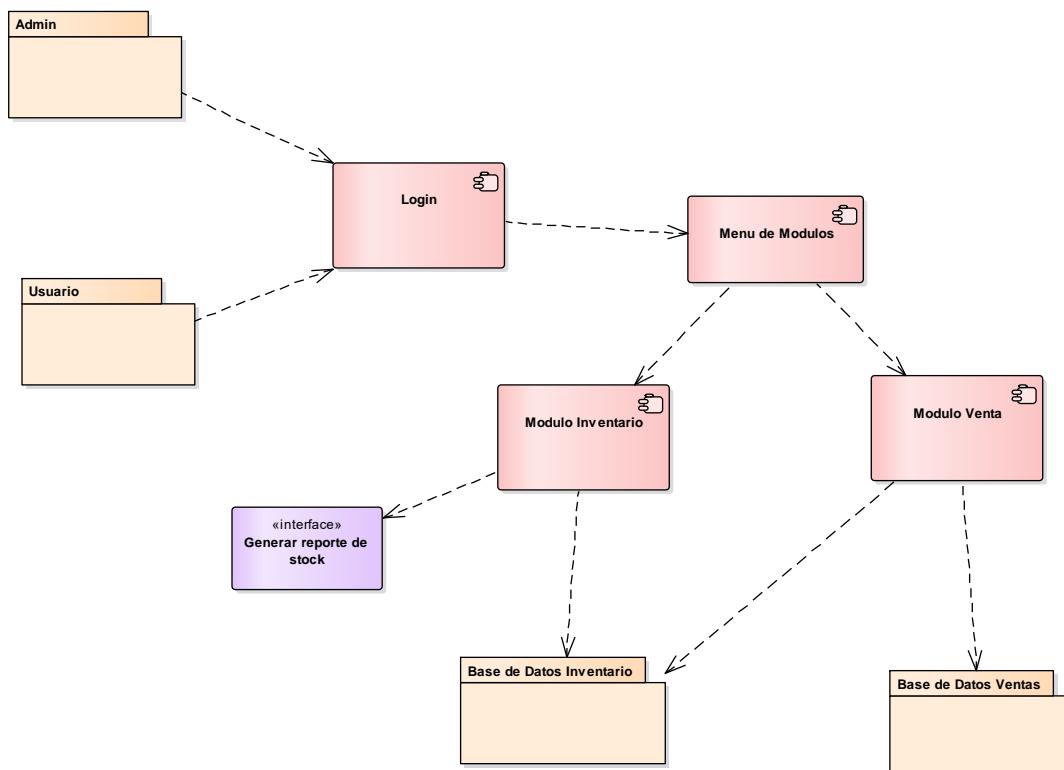


Figure 5: Diagrama de componentes

## Admin

Package in package 'Diagrama de componentes'

Admin  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 6/8/2024. Last modified 6/8/2024

CONNECTORS	
<b>Dependency</b>	Source -> Destination From: Admin : Package, Public To: Login : Component, Public

## **Base de Datos Inventory**

Package in package 'Diagrama de componentes'

Base de Datos Inventory

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### **CONNECTORS**

↗ **Dependency**    Source -> Destination  
From: Modulo Venta : Component, Public  
To: Base de Datos Inventory : Package, Public

↗ **Dependency**    Source -> Destination  
From: Modulo Inventory : Component, Public  
To: Base de Datos Inventory : Package, Public

## **Base de Datos Ventas**

Package in package 'Diagrama de componentes'

Base de Datos Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### **CONNECTORS**

↗ **Dependency**    Source -> Destination  
From: Modulo Venta : Component, Public  
To: Base de Datos Ventas : Package, Public

## **Login**

Component in package 'Diagrama de componentes'

Login

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### **CONNECTORS**

↗ **Dependency**    Source -> Destination  
From: Login : Component, Public  
To: Menu de Modulos : Component, Public

↗ **Dependency**    Source -> Destination  
From: Admin : Package, Public  
To: Login : Component, Public

## CONNECTORS

### ↗ Dependency    Source -> Destination

From:    Usuario : Package, Public  
To:       Login : Component, Public

## Menu de Modulos

Component in package 'Diagrama de componentes'

Menu de Modulos

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## CONNECTORS

### ↗ Dependency    Source -> Destination

From:    Menu de Modulos : Component, Public  
To:       Modulo Venta : Component, Public

### ↗ Dependency    Source -> Destination

From:    Menu de Modulos : Component, Public  
To:       Modulo inventario : Component, Public

### ↗ Dependency    Source -> Destination

From:    Login : Component, Public  
To:       Menu de Modulos : Component, Public

## Modulo inventario

Component in package 'Diagrama de componentes'

Modulo inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## CONNECTORS

### ↗ Dependency    Source -> Destination

From:    Modulo inventario : Component, Public  
To:       Base de Datos inventario : Package, Public

### ↗ Dependency    Source -> Destination

From:    Modulo inventario : Component, Public  
To:       Generar reporte de stock : Interface, Public

### ↗ Dependency    Source -> Destination

From:    Menu de Modulos : Component, Public  
To:       Modulo inventario : Component, Public

## Modulo Venta

Component in package 'Diagrama de componentes'

Modulo Venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### CONNECTORS

 **Dependency**    Source -> Destination  
From: Modulo Venta : Component, Public  
To: Base de Datos Inventario : Package, Public

 **Dependency**    Source -> Destination  
From: Modulo Venta : Component, Public  
To: Base de Datos Ventas : Package, Public

 **Dependency**    Source -> Destination  
From: Menu de Modulos : Component, Public  
To: Modulo Venta : Component, Public

## Usuario

Package in package 'Diagrama de componentes'

Usuario

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### CONNECTORS

 **Dependency**    Source -> Destination  
From: Usuario : Package, Public  
To: Login : Component, Public

## Admin

Package in package 'Diagrama de componentes'

Admin

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### LINKS TO OTHER PACKAGES

 Dependency from 'Admin' Package to 'Login' Component	Direction:    Source -> Destination
--	-------------------------------------

## **Generacion de reportes diagram**

Component diagram in package 'Admin'

Generacion de reportes

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

Figure 6: Generacion de reportes

## Base de Datos Inventory

Package in package 'Diagrama de componentes'

Base de Datos Inventory

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

LINKS TO OTHER PACKAGES	
 Dependency from 'Modulo Venta' Component to 'Base de Datos Inventory' Package	Direction: Source -> Destination
 Dependency from 'Modulo inventario' Component to 'Base de Datos Inventory' Package	Direction: Source -> Destination

## Base de Datos Inventory diagram

Component diagram in package 'Base de Datos Inventory'

Base de Datos Inventory

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

Figure 7: Base de Datos Inventory

## **Base de Datos Ventas**

Package in package 'Diagrama de componentes'

Base de Datos Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

LINKS TO OTHER PACKAGES	
 Dependency from 'Modulo Venta' Component to 'Base de Datos Ventas' Package	Direction: Source -> Destination

## **Base de Datos Ventas diagram**

Component diagram in package 'Base de Datos Ventas'

Base de Datos Ventas

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

Figure 8: Base de Datos Ventas

## **Usuario**

Package in package 'Diagrama de componentes'

Usuario

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### **LINKS TO OTHER PACKAGES**

 Dependency from 'Usuario' Package to 'Login' Component	Direction: Source -> Destination
--	----------------------------------

## **Usuario diagram**

Component diagram in package 'Usuario'

Usuario

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

Figure 9: Usuario

## Login

Package in package 'Diagrama de componentes'

Login

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

LINKS TO OTHER PACKAGES	
Dependency from 'Login' Component to 'Menu de Modulos' Component	Direction: Source -> Destination
Dependency from 'Admin' Package to 'Login' Component	Direction: Source -> Destination
Dependency from 'Usuario' Package to 'Login' Component	Direction: Source -> Destination

## Generacion de reportes diagram

Component diagram in package 'Login'

Generacion de reportes

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

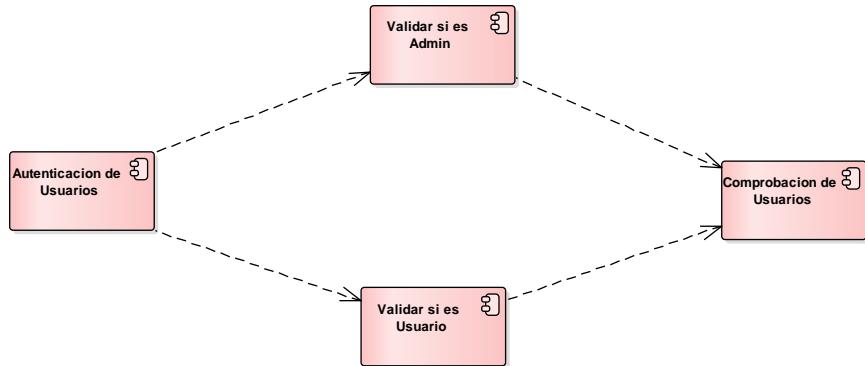


Figure 10: Generacion de reportes

## Autenticacion de Usuarios

Component in package 'Login'

Autenticacion de Usuarios

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## CONNECTORS

### ↗ Dependency    Source -> Destination

From: Autenticacion de Usuarios : Component, Public  
To: Validar si es Admin : Component, Public

### ↗ Dependency    Source -> Destination

From: Autenticacion de Usuarios : Component, Public  
To: Validar si es Usuario : Component, Public

## Comprobacion de Usuarios

Component in package 'Login'

Comprobacion de Usuarios

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## CONNECTORS

### ↗ Dependency    Source -> Destination

From: Validar si es Usuario : Component, Public  
To: Comprobacion de Usuarios : Component, Public

### ↗ Dependency    Source -> Destination

From: Validar si es Admin : Component, Public  
To: Comprobacion de Usuarios : Component, Public

## Validar si es Admin

Component in package 'Login'

Validar si es Admin

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## CONNECTORS

### ↗ Dependency    Source -> Destination

From: Validar si es Admin : Component, Public  
To: Comprobacion de Usuarios : Component, Public

### ↗ Dependency    Source -> Destination

From: Autenticacion de Usuarios : Component, Public  
To: Validar si es Admin : Component, Public

## Validar si es Usuario

## Component in package 'Login'

Validar si es Usuario

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### CONNECTORS

#### ↗ Dependency    Source -> Destination

From:    Validar si es Usuario : Component, Public

To:       Comprobacion de Usuarios : Component, Public

#### ↗ Dependency    Source -> Destination

From:    Autenticacion de Usuarios : Component, Public

To:       Validar si es Usuario : Component, Public

## Menu de Modulos

Package in package 'Diagrama de componentes'

Menu de Modulos

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

LINKS TO OTHER PACKAGES	
 Dependency from 'Menu de Modulos' Component to 'Modulo Venta' Component	Direction: Source -> Destination
 Dependency from 'Menu de Modulos' Component to 'Modulo inventario' Component	Direction: Source -> Destination
 Dependency from 'Login' Component to 'Menu de Modulos' Component	Direction: Source -> Destination

## Menu de Modulos diagram

Component diagram in package 'Menu de Modulos'

Menu de Modulos

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

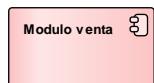


Figure 11: Menu de Modulos

## Modulo inventario

Component in package 'Menu de Modulos'

Modulo inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## Modulo venta

Component in package 'Menu de Modulos'

Modulo venta

Version 1.0 Phase 1.0 Proposed  
Fabian created on 6/8/2024. Last modified 6/8/2024

## Modulo Inventory

Package in package 'Diagrama de componentes'

Modulo Inventory

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

LINKS TO OTHER PACKAGES	
 Dependency from 'Modulo Inventory' Component to 'Base de Datos Inventory' Package	Direction: Source -> Destination
 Dependency from 'Modulo Inventory' Component to 'Generar reporte de stock' Interface	Direction: Source -> Destination
 Dependency from 'Menu de Modulos' Component to 'Modulo Inventory' Component	Direction: Source -> Destination

## Inventory diagram

Component diagram in package 'Modulo Inventory'

Inventory

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024

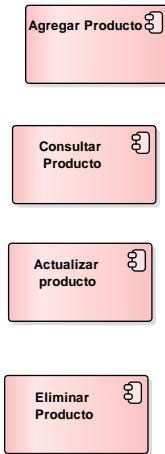


Figure 12: Inventory

## Actualizar producto

Component in package 'Modulo Inventory'

Actualizar producto

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## **Agregar Producto**

Component in package 'Modulo Inventario'

Agregar Producto

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## **Consultar Producto**

Component in package 'Modulo Inventario'

Consultar Producto

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## **Eliminar Producto**

Component in package 'Modulo Inventario'

Eliminar Producto

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## Modulo Venta

Package in package 'Diagrama de componentes'

Modulo Venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

LINKS TO OTHER PACKAGES	
 Dependency from 'Modulo Venta' Component to 'Base de Datos Inventory' Package	Direction: Source -> Destination
 Dependency from 'Modulo Venta' Component to 'Base de Datos Ventas' Package	Direction: Source -> Destination
 Dependency from 'Menu de Modulos' Component to 'Modulo Venta' Component	Direction: Source -> Destination

## Venta diagram

Component diagram in package 'Modulo Venta'

Venta

Version 1.0

Fabian created on 6/8/2024. Last modified 6/8/2024



Figure 13: Venta

## Actualizar Ventas

Component in package 'Modulo Venta'

Actualizar Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## Agregar una Venta

Component in package 'Modulo Venta'

Agregar una Venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## Consultar Ventas

Component in package 'Modulo Venta'

Consultar Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## Eliminar Ventas

Component in package 'Modulo Venta'

Eliminar Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

## Generar reporte de stock

Interface in package 'Diagrama de componentes'

Generar reporte de stock

Version 1.0 Phase 1.0 Proposed

Fabian created on 6/8/2024. Last modified 6/8/2024

### CONNECTORS

 **Dependency**    Source -> Destination  
From: Modulo Inventario : Component, Public  
To: Generar reporte de stock : Interface, Public

diagrama de clases  
Package in package 'Use Case Model'

diagrama de clases  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 4/8/2024. Last modified 4/8/2024

**diagrama de clases diagram**  
Class diagram in package 'diagrama de clases'

diagrama de clases  
Version 1.0  
Fabian created on 4/8/2024. Last modified 4/8/2024

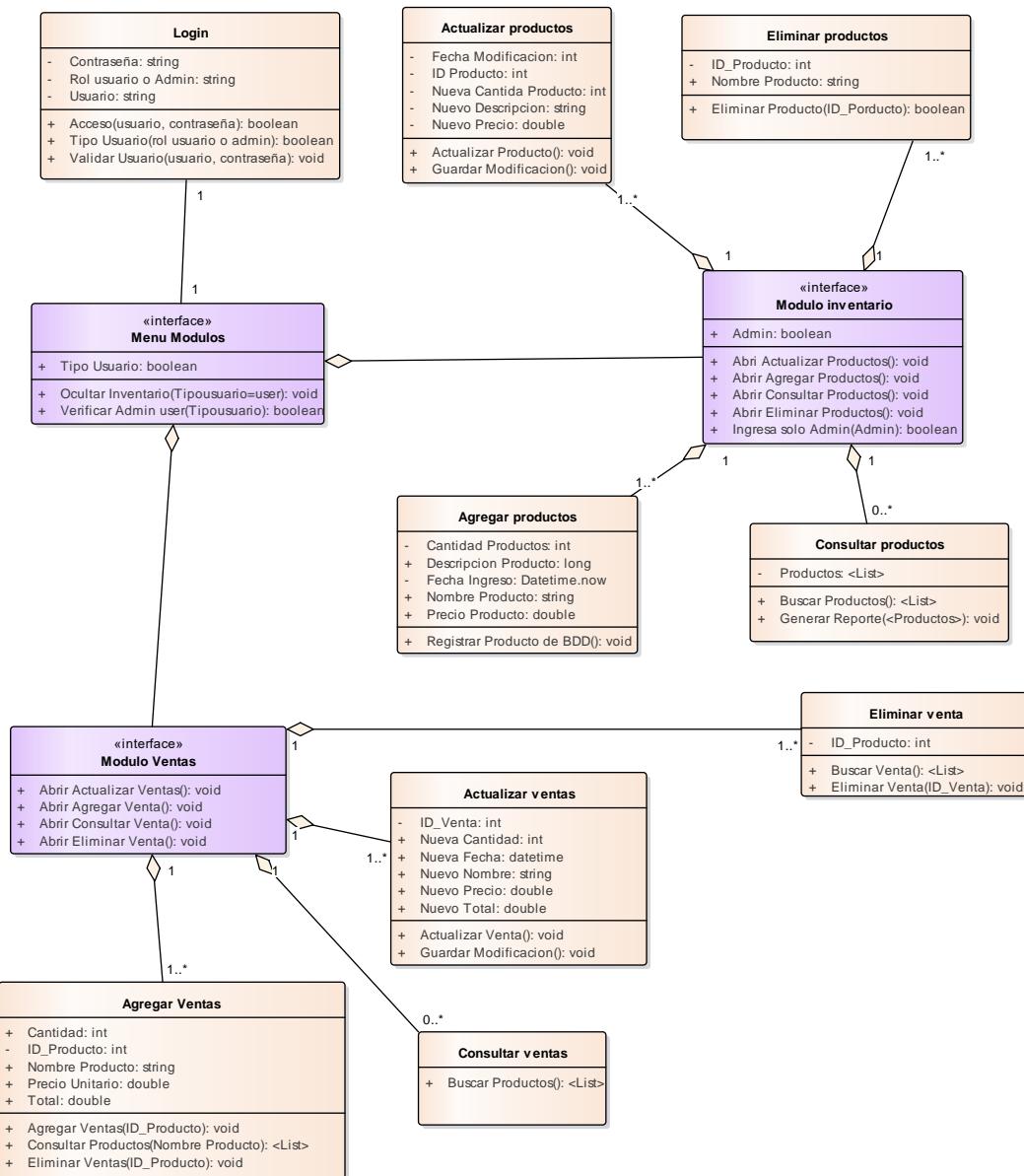


Figure 14: diagrama de clases

## Actualizar productos

Class in package 'diagrama de clases'

Actualizar productos

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Aggregation from Actualizar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

### ATTRIBUTES

- ◆ Fecha Modificacion : int Private  
[ Is static False. Containment is Not Specified. ]
- ◆ ID Producto : int Private  
[ Is static False. Containment is Not Specified. ]
- ◆ Nueva Cantida Producto : int Private  
[ Is static False. Containment is Not Specified. ]
- ◆ Nuevo Descripcion : string Private  
[ Is static False. Containment is Not Specified. ]
- ◆ Nuevo Precio : double Private  
[ Is static False. Containment is Not Specified. ]

### OPERATIONS

- ◆ Actualizar Producto () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]
- ◆ Guardar Modificacion () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Actualizar ventas

Class in package 'diagrama de clases'

Actualizar ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Aggregation from Actualizar ventas to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ ID\_Venta : int Private  
[ Is static False. Containment is Not Specified. ]

◆ Nueva Cantidad : int Public  
[ Is static False. Containment is Not Specified. ]

◆ Nueva Fecha : datetime Public  
[ Is static False. Containment is Not Specified. ]

◆ Nuevo Nombre : string Public  
[ Is static False. Containment is Not Specified. ]

◆ Nuevo Precio : double Public  
[ Is static False. Containment is Not Specified. ]

◆ Nuevo Total : double Public  
[ Is static False. Containment is Not Specified. ]

## OPERATIONS

◆ Actualizar Venta () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Guardar Modificacion () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Agregar Ventas

Class in package 'diagrama de clases'

Agregar Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Aggregation from Agregar Ventas to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ Cantidad : int Public  
[ Is static False. Containment is Not Specified. ]

◆ ID\_Producto : int Private  
[ Is static False. Containment is Not Specified. ]

◆ Nombre Producto : string Public  
[ Is static False. Containment is Not Specified. ]

◆ Precio Unitario : double Public  
[ Is static False. Containment is Not Specified. ]

◆ Total : double Public  
[ Is static False. Containment is Not Specified. ]

## OPERATIONS

◆ Agregar Ventas (int : ID\_Producto ) : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Consultar Productos (string : Nombre Producto ) : <List> Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Eliminar Ventas (int : ID\_Producto ) : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Agregar productos

Class in package 'diagrama de clases'

Agregar productos

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

◀ Aggregation from Agregar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ Cantidad Productos : int Private  
[ Is static False. Containment is Not Specified. ]

◆ Descripcion Producto : long Public

## ATTRIBUTES

[ Is static False. Containment is Not Specified. ]

◆ Fecha Ingreso : Datetime.now Private

[ Is static False. Containment is Not Specified. ]

◆ Nombre Producto : string Public

[ Is static False. Containment is Not Specified. ]

◆ Precio Producto : double Public

[ Is static False. Containment is Not Specified. ]

## OPERATIONS

◆ Registrar Producto de BDD () : void Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Consultar productos

Class in package 'diagrama de clases'

Consultar productos

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

◀ Aggregation from Consultar productos to Modulo inventario

[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ Productos : <List> Private

[ Is static False. Containment is Not Specified. ]

## OPERATIONS

◆ Buscar Productos () : <List> Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Generar Reporte (List : <Productos> ) : void Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Consultar ventas

Class in package 'diagrama de clases'

Consultar ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Aggregation from Consultar ventas to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

## OPERATIONS

- ❖ Buscar Productos () : <List> Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Eliminar productos

Class in package 'diagrama de clases'

Eliminar productos

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Aggregation from Eliminar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

- ◆ ID\_Producto : int Private

[ Is static False. Containment is Not Specified. ]

- ◆ Nombre Producto : string Public

[ Is static False. Containment is Not Specified. ]

## OPERATIONS

- ❖ Eliminar Producto (Producto : ID\_Porducto ) : boolean Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Eliminar venta

Class in package 'diagrama de clases'

Eliminar venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Aggregation from Eliminar venta to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ ID\_Producto : int Private  
[ Is static False. Containment is Not Specified. ]

## OPERATIONS

◆ Buscar Venta () : <List> Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Eliminar Venta (int : ID\_Venta) : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Login

Class in package 'diagrama de clases'

Login

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## ATTRIBUTES

◆ Contraseña : string Private  
[ Is static False. Containment is Not Specified. ]

◆ Rol usuario o Admin : string Private  
[ Is static False. Containment is Not Specified. ]

◆ Usuario : string Private  
[ Is static False. Containment is Not Specified. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Class) Login  
Cardinality: [1]

Target: Public (Interface) Menu Modulos  
Cardinality: [1]

## OPERATIONS

◆ Acceso (char : usuario , char2 : contraseña) : boolean Public

## OPERATIONS

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Tipo Usuario (char : rol usuario o admin ) : boolean Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Validar Usuario (char : usuario , char2 : contraseña ) : void Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Menu Modulos

Interface in package 'diagrama de clases'

Menu Modulos

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Aggregation from Modulo inventario to Menu Modulos

[ Direction is 'Source -> Destination'. ]

⇒ Aggregation from Modulo Ventas to Menu Modulos

[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ Tipo Usuario : boolean Public

[ Is static False. Containment is Not Specified. ]

## ASSOCIATIONS

↙ Association (direction: Unspecified)

Source: Public (Class) Login

Cardinality: [1]

Target: Public (Interface) Menu Modulos

Cardinality: [1]

## OPERATIONS

◆ Ocultar Inventario (r: : Tipousuario=user ) : void Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Verificar Admin user (boolean : Tipousuario ) : boolean Public

[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Modulo Ventas

Interface in package 'diagrama de clases'

Modulo Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

#### OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Aggregation from Modulo Ventas to Menu Modulos  
[ Direction is 'Source -> Destination'. ]

#### INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Aggregation from Agregar Ventas to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

- ⇒ Aggregation from Consultar ventas to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

- ⇒ Aggregation from Eliminar venta to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

- ⇒ Aggregation from Actualizar ventas to Modulo Ventas  
[ Direction is 'Source -> Destination'. ]

#### OPERATIONS

- ◊ Abrir Actualizar Ventas () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

- ◊ Abrir Agregar Venta () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

- ◊ Abrir Consultar Venta () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

- ◊ Abrir Eliminar Venta () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

## Modulo inventario

Interface in package 'diagrama de clases'

Modulo inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 4/8/2024. Last modified 4/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Aggregation from Modulo inventario to Menu Modulos  
[ Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Aggregation from Agregar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

⇒ Aggregation from Actualizar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

⇒ Aggregation from Eliminar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

⇒ Aggregation from Consultar productos to Modulo inventario  
[ Direction is 'Source -> Destination'. ]

## ATTRIBUTES

◆ Admin : boolean Public  
[ Is static False. Containment is Not Specified. ]

## OPERATIONS

◆ Abri Actualizar Productos () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Abrir Agregar Productos () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Abrir Consultar Productos () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Abrir Eliminar Productos () : void Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

◆ Ingresar solo Admin (boolean : Admin ) : boolean Public  
[ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ]

diagrama de colaboracion  
Package in package 'Use Case Model'

diagrama de colaboracion  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 4/8/2024. Last modified 4/8/2024

**Actualizar Productos diagram**  
Analysis diagram in package 'diagrama de colaboracion'

Actualizar Productos  
Version 1.0  
Fabian created on 1/8/2024. Last modified 6/8/2024

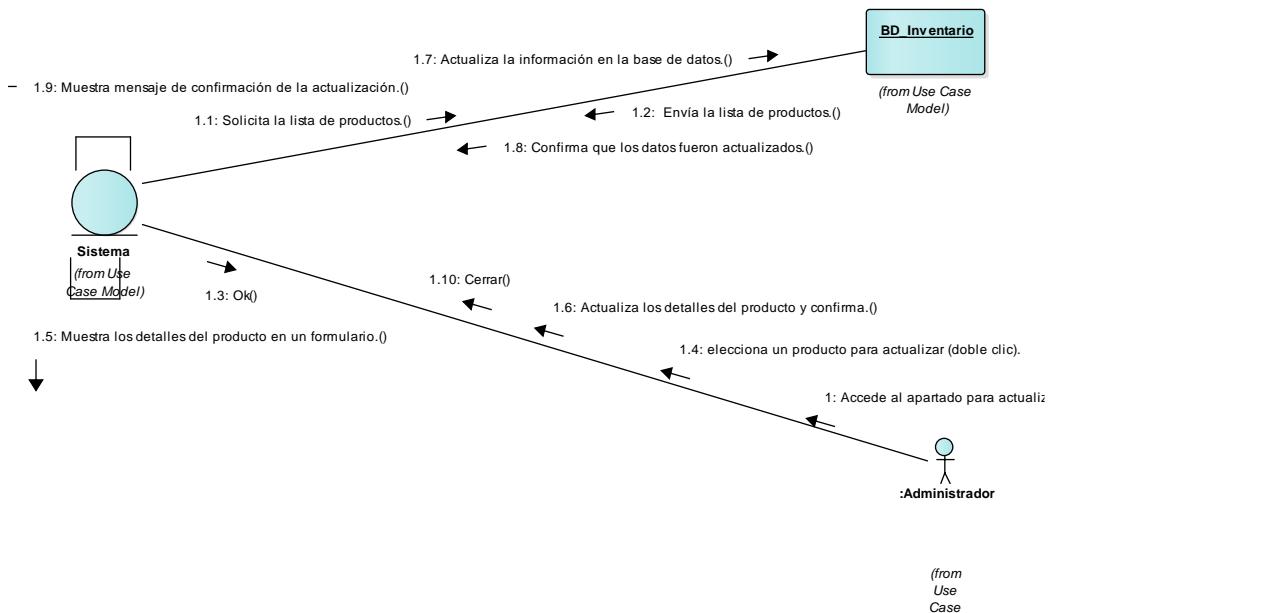


Figure 15: Actualizar Productos

**Object**  
Object in package 'Use Case Model'

Object  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 1/8/2024. Last modified 1/8/2024

OUTGOING STRUCTURAL RELATIONSHIPS
↳ Collaboration from Object to «entity» Sistema [ Name is elección un producto para actualizar (doble clic).. Direction is 'Source -> Destination'. ]
↳ Collaboration from Object to «entity» Sistema [ Name is Cerrar. Direction is 'Source -> Destination'. ]
↳ Collaboration from Object to «entity» Sistema [ Name is Actualiza los detalles del producto y confirma.. Direction is 'Source -> Destination'. ]

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para actualizar productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Solicita la lista de productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los detalles del producto en un formulario.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Solicita la lista de productos.. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los detalles del producto en un formulario.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is elecciona un producto para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

➡ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Actualiza los detalles del producto y confirma.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para actualizar productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Actualizar Ventas diagram

Analysis diagram in package 'diagrama de colaboracion'

Actualizar Ventas

Version 1.0

Fabian created on 31/7/2024. Last modified 6/8/2024

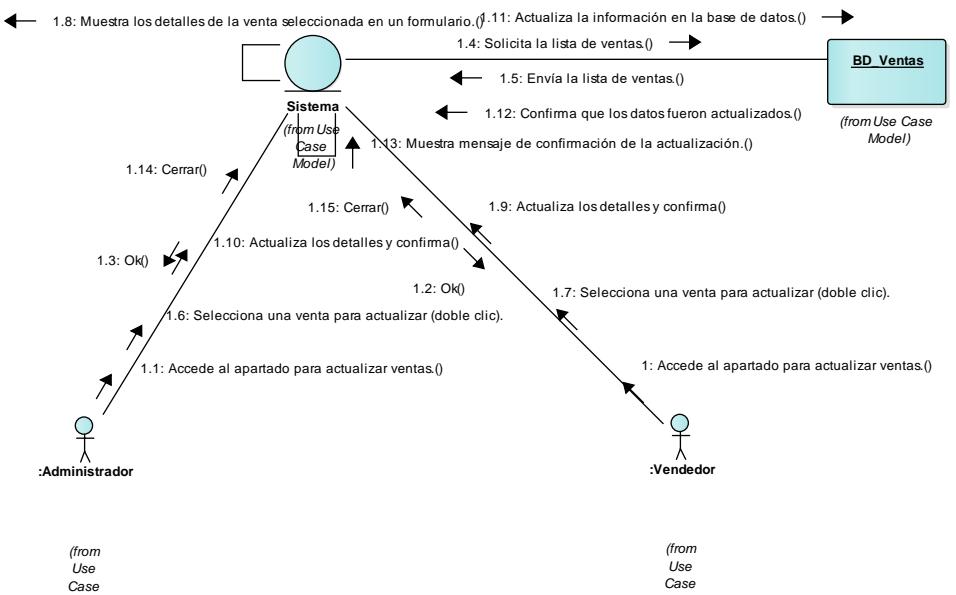


Figure 16: Actualizar Ventas

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema

[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Object

Object in package 'Use Case Model'

### Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

#### ⇒ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

#### ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## BD\_Ventas

Object in package 'Use Case Model'

### BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

#### ↳ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

- ⇒ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Solicita la lista de ventas.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

- ↳ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

- ↳ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

- ↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra los detalles de la venta seleccionada en un formulario.. Direction is 'Source -> Destination'. ]

- ↳ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

- ↳ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Solicita la lista de ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra los detalles de la venta seleccionada en un formulario.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Consultar Productos diagram

Analysis diagram in package 'diagrama de colaboracion'

Consultar Productos

Version 1.0

Fabian created on 1/8/2024. Last modified 6/8/2024

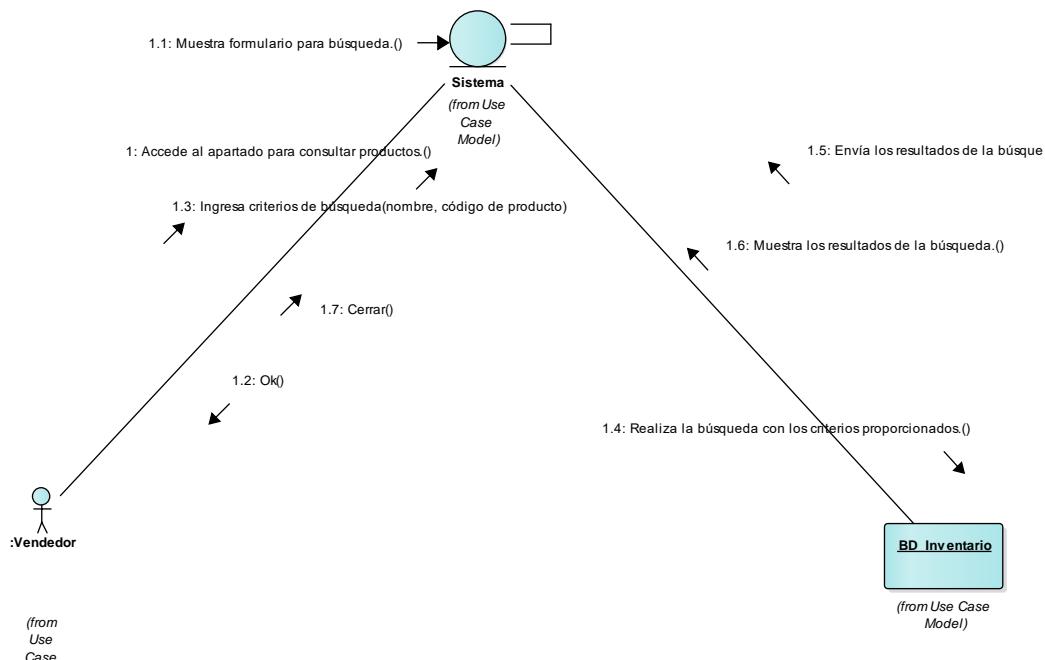


Figure 17: Consultar Productos

### Object

Object in package 'Use Case Model'

#### Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

#### OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para consultar productos.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema

[ Name is Ingresa criterios de búsqueda. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

#### INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

#### ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### BD\_Inventario

Object in package 'Use Case Model'

#### BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

#### OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

### Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para consultar productos.. Direction is 'Source -> Destination'. ]

➡ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Ingresa criterios de búsqueda. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Consultar Ventas diagram

Analysis diagram in package 'diagrama de colaboracion'

Consultar Ventas

Version 1.0

Fabian created on 1/8/2024. Last modified 6/8/2024

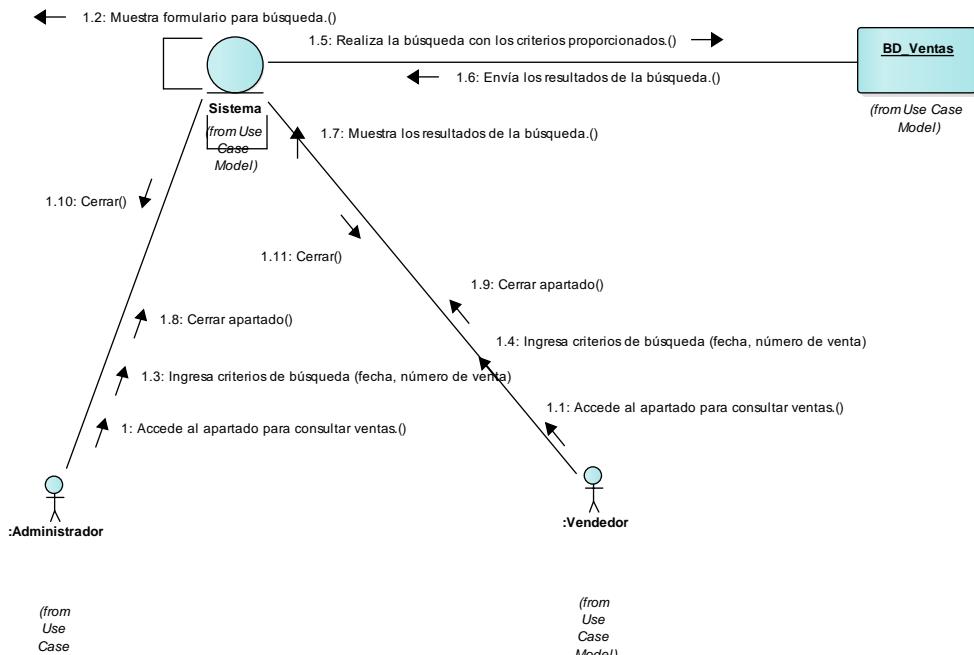


Figure 18: Consultar Ventas

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
 [ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
 [ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
 [ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to Object  
 [ Name is Cerrar. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to Object  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## BD\_Ventas

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

### Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↙ Collaboration from «entity» Sistema to Object

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

↙ Collaboration from «entity» Sistema to Object

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

↙ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

↙ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

↙ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

↗ Collaboration from Object to «entity» Sistema

[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

↗ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

↗ Collaboration from Object to «entity» Sistema

[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

↗ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

↗ Collaboration from Object to «entity» Sistema

[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Eliminar Productos diagram

## Analysis diagram in package 'diagrama de colaboracion'

Eliminar Productos

Version 1.0

Fabian created on 1/8/2024. Last modified 6/8/2024

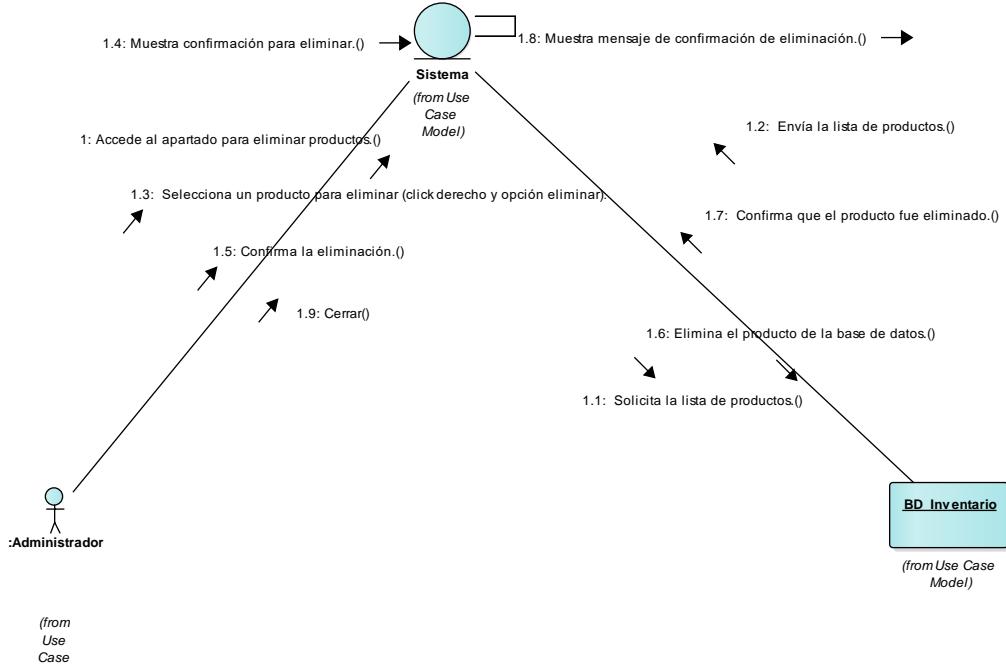


Figure 19: Eliminar Productos

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Selección de producto para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar productos.. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

Association (direction: Unspecified)

## ASSOCIATIONS

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que el producto fue eliminado.. Direction is 'Source -> Destination'. ]

↳ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Elimina el producto de la base de datos.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Solicita la lista de productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Elimina el producto de la base de datos.. Direction is 'Source -> Destination'. ]

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Sigue la lista de productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que el producto fue eliminado.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Selecciona un producto para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Eliminar Ventas diagram

Analysis diagram in package 'diagrama de colaboracion'

Eliminar Ventas

Version 1.0

Fabian created on 1/8/2024. Last modified 6/8/2024

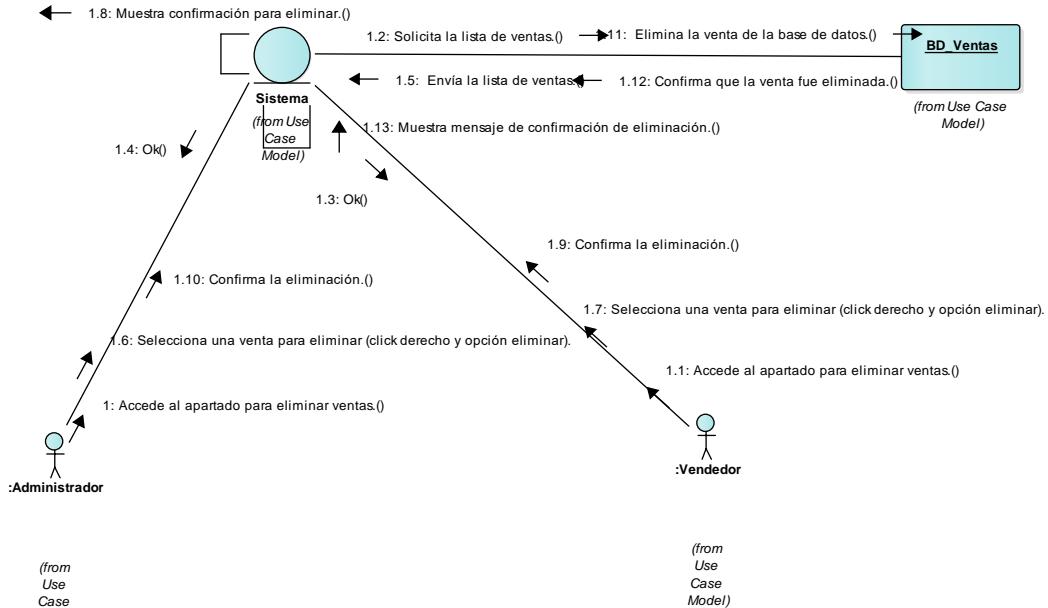


Figure 20: Eliminar Ventas

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Collaboration from Object to «entity» Sistema  
[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ↗ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ✓ Association (direction: Unspecified)  
Source: Public (Object) Object      Target: Public (Object) Sistema «entity»

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from Object to «entity» Sistema  
[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## BD\_Ventas

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ⬅ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Confirma que la venta fue eliminada.. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Solicita la lista de ventas.. Direction is 'Source -> Destination'. ]

- ➡ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Elimina la venta de la base de datos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Sigue la lista de ventas.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Elimina la venta de la base de datos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que la venta fue eliminada.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

### ➡ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

### ➡ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Ingresar nuevos Productos diagram

Analysis diagram in package 'diagrama de colaboracion'

Ingresar nuevos Productos

Version 1.0

Fabian created on 1/8/2024. Last modified 6/8/2024

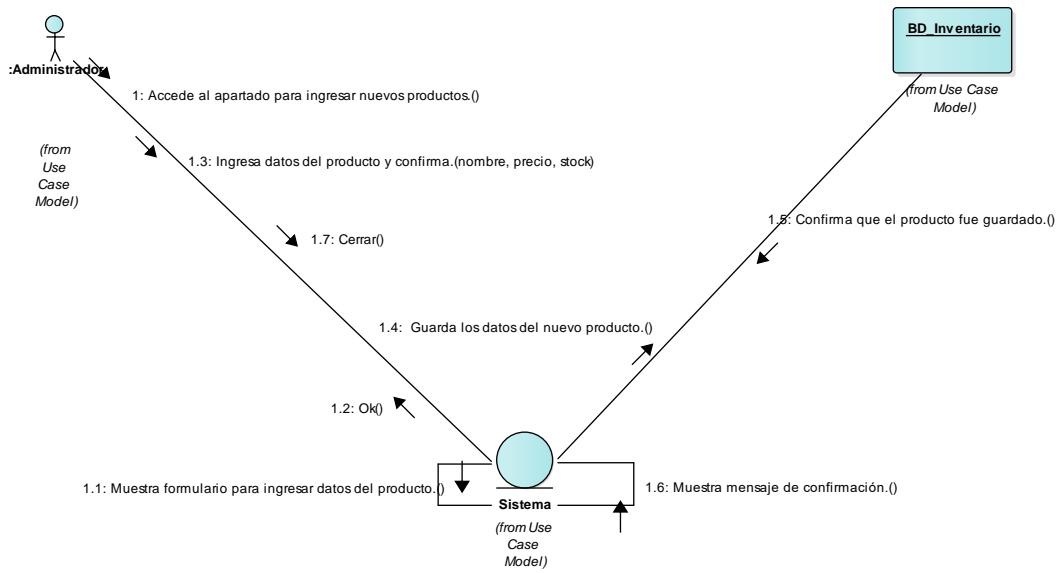


Figure 21: Ingresar nuevos Productos

## Object

Object in package 'Use Case Model'

### Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

#### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
 [ Name is Cerrar. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
 [ Name is Ingresa datos del producto y confirma.. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
 [ Name is Accede al apartado para ingresar nuevos productos.. Direction is 'Source -> Destination'. ]

#### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to Object  
 [ Name is Ok. Direction is 'Source -> Destination'. ]

#### ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## **BD\_Inventario**

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### **OUTGOING STRUCTURAL RELATIONSHIPS**

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que el producto fue guardado.. Direction is 'Source -> Destination'. ]

### **INCOMING STRUCTURAL RELATIONSHIPS**

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Guarda los datos del nuevo producto.. Direction is 'Source -> Destination'. ]

### **ASSOCIATIONS**

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## **Sistema**

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### **OUTGOING STRUCTURAL RELATIONSHIPS**

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Guarda los datos del nuevo producto.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra formulario para ingresar datos del producto.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra mensaje de confirmación.. Direction is 'Source -> Destination'. ]

➡ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Confirma que el producto fue guardado.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa datos del producto y confirma.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra formulario para ingresar datos del producto.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para ingresar nuevos productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Registrar Ventas diagram

Analysis diagram in package 'diagrama de colaboracion'

Registrar Ventas

Version 1.0

Fabian created on 3/8/2024. Last modified 6/8/2024

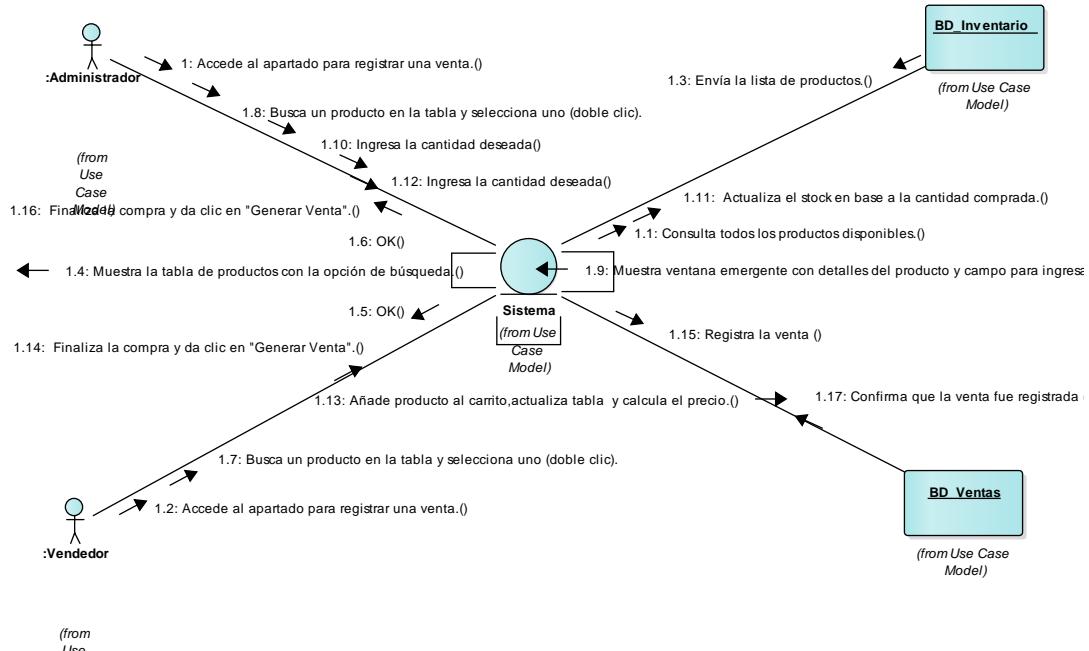


Figure 22: Registrar Ventas

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Collaboration from Object to «entity» Sistema	[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]
↳ Collaboration from Object to «entity» Sistema	[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]
↳ Collaboration from Object to «entity» Sistema	[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]
↳ Collaboration from Object to «entity» Sistema	[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to Object

[ Name is OK. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema

[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema

[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to Object

[ Name is OK. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## **BD\_Inventario**

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### **OUTGOING STRUCTURAL RELATIONSHIPS**

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

### **INCOMING STRUCTURAL RELATIONSHIPS**

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Actualiza el stock en base a la cantidad comprada.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Consulta todos los productos disponibles.. Direction is 'Source -> Destination'. ]

### **ASSOCIATIONS**

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## **BD\_Ventas**

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### **OUTGOING STRUCTURAL RELATIONSHIPS**

⬅ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que la venta fue registrada . Direction is 'Source -> Destination'. ]

### **INCOMING STRUCTURAL RELATIONSHIPS**

➡ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Registra la venta . Direction is 'Source -> Destination'. ]

### **ASSOCIATIONS**

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

## ASSOCIATIONS

### Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

### ↳ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Actualiza el stock en base a la cantidad comprada.. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to Object

[ Name is OK. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Añade producto al carrito,actualiza tabla y calcula el precio.. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra la tabla de productos con la opción de búsqueda.. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to Object

[ Name is OK. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Consulta todos los productos disponibles.. Direction is 'Source -> Destination'. ]

### ↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Registra la venta . Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

### ⇒ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que la venta fue registrada . Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Añade producto al carrito,actualiza tabla y calcula el precio.. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra la tabla de productos con la opción de búsqueda.. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

### ⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

### / Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### / Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

### / Association (direction: Unspecified)

## ASSOCIATIONS

Source: Public (Object) Sistema «entity» Target: Public (Object) BD\_\_Inventario

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

Casos de uso  
Package in package 'Use Case Model'

Casos de uso  
Version Phase 1.0 Proposed  
Carolina created on 30/7/2024. Last modified 30/7/2024

**Actualizar Información de Productos diagram**  
Use Case diagram in package 'Casos de uso'

Actualizar Información de Productos  
Version 1.0  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

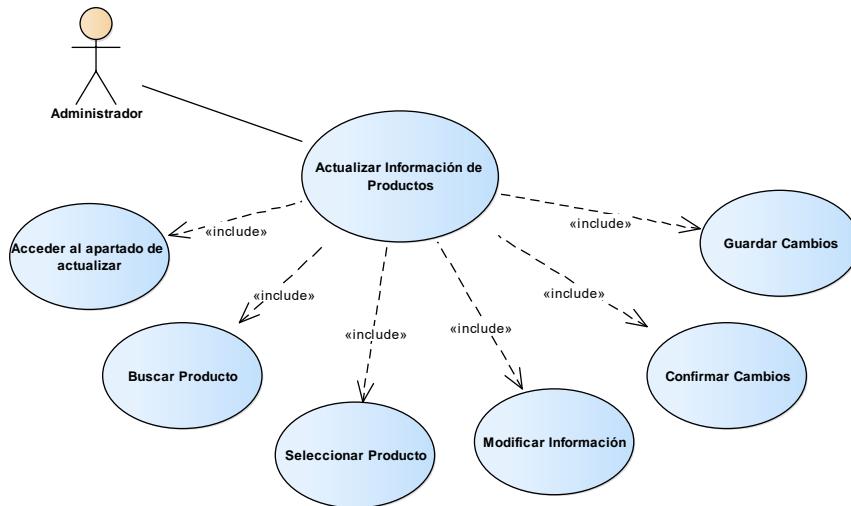


Figure 23: Actualizar Información de Productos

**Actualizar Ventas diagram**  
Use Case diagram in package 'Casos de uso'

Actualizar Ventas  
Version 1.0  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

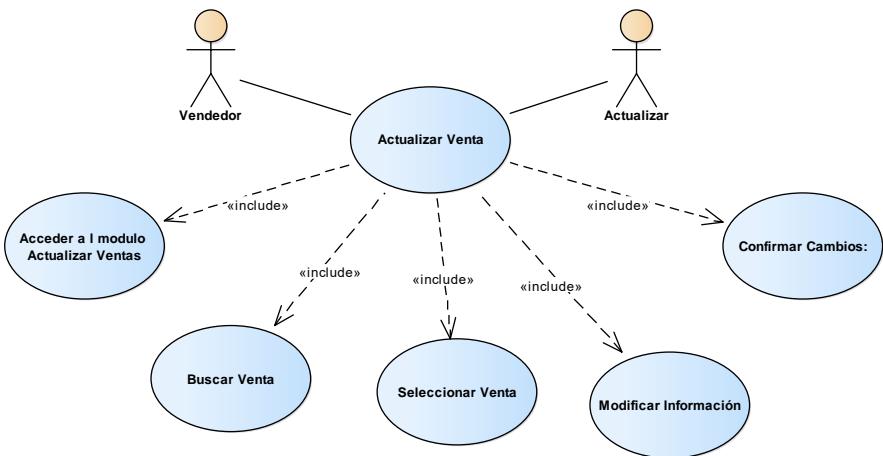


Figure 24: Actualizar Ventas

### Agregar venta diagram

Use Case diagram in package 'Casos de uso'

te

Agregar venta

Version 1.0

Carolina created on 30/7/2024. Last modified 30/7/2024

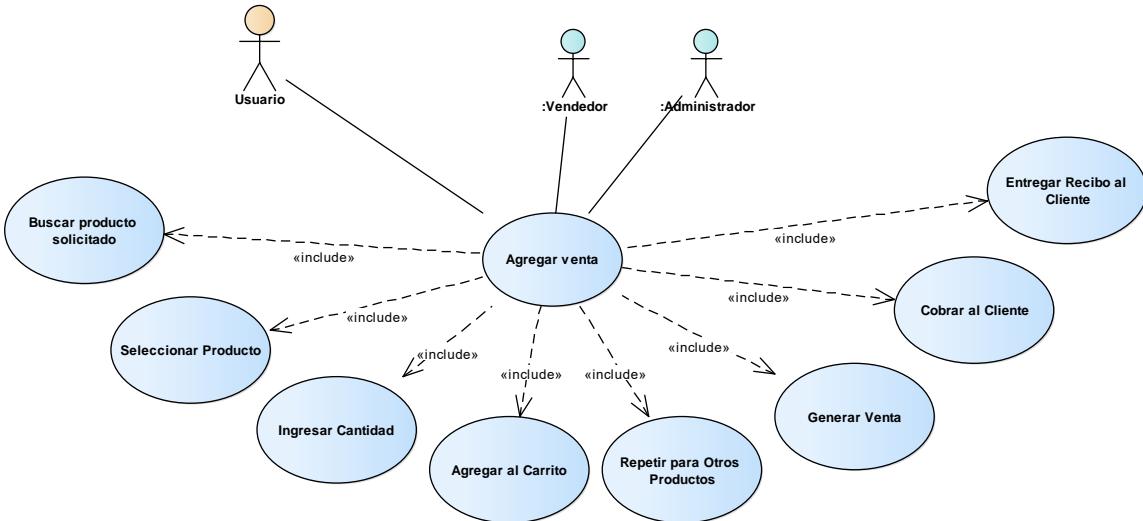


Figure 25: Agregar venta

### Consultar Productos diagram

Use Case diagram in package 'Casos de uso'

Consultar Productos

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

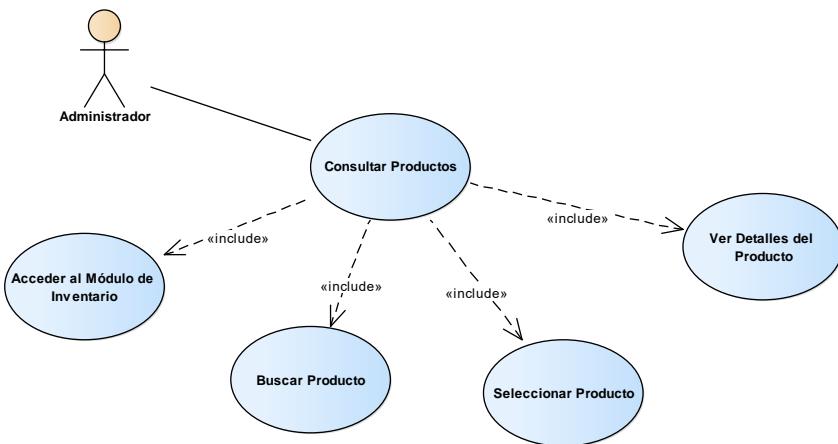


Figure 26: Consultar Productos

### Consultar Ventas diagram

Use Case diagram in package 'Casos de uso'

Consultar Ventas  
 Version 1.0  
 Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

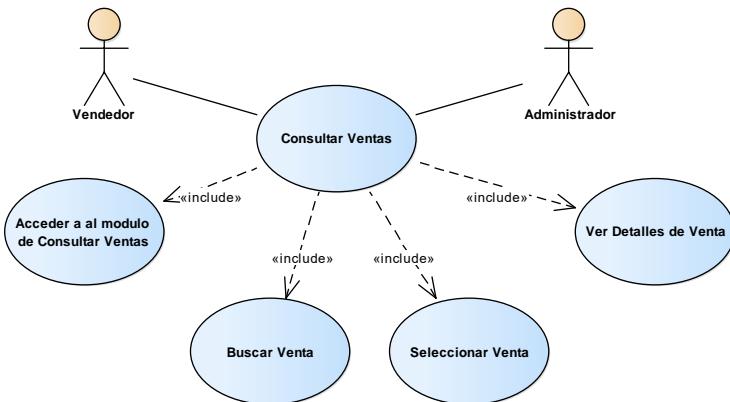


Figure 27: Consultar Ventas

### Eliminar Productos diagram

Use Case diagram in package 'Casos de uso'

Eliminar Productos  
 Version 1.0  
 Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

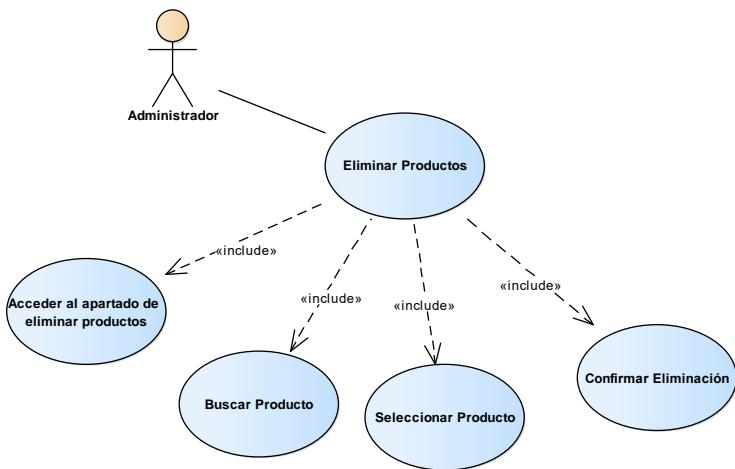


Figure 28: Eliminar Productos

### **Eliminar Venta diagram**

Use Case diagram in package 'Casos de uso'

Eliminar Venta  
 Version 1.0  
 Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

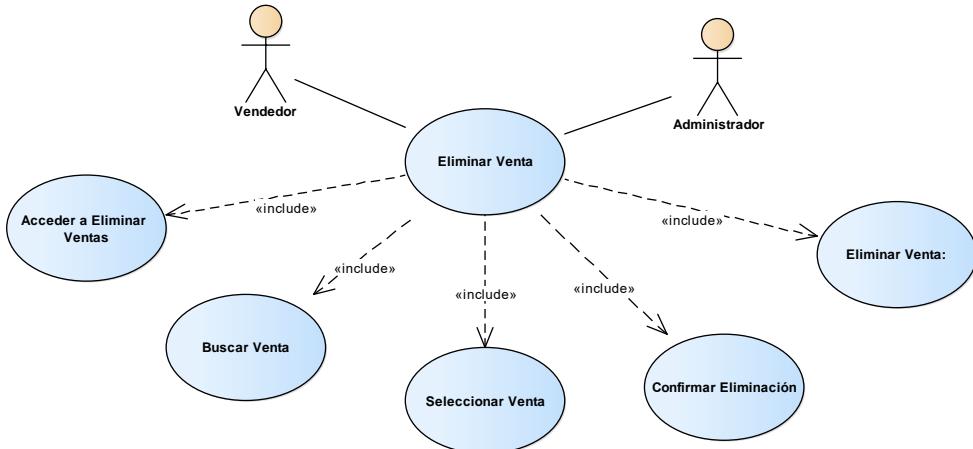


Figure 29: Eliminar Venta

### **Ingresra nuevo producto diagram**

Use Case diagram in package 'Casos de uso'

Ingresra nuevo producto  
 Version 1.0  
 Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

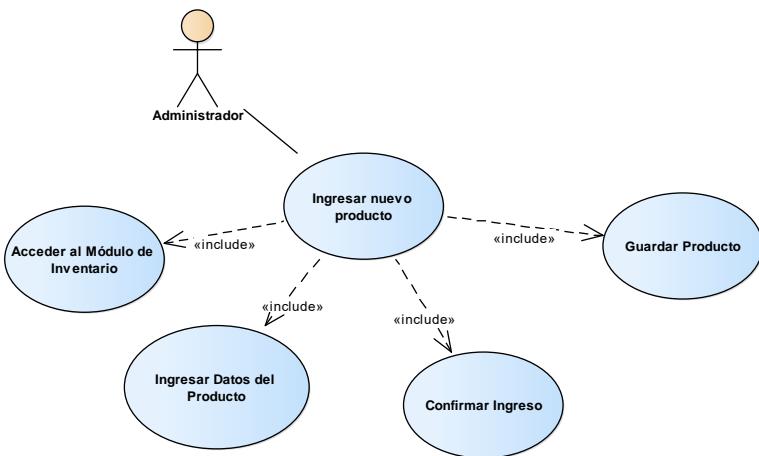


Figure 30: Ingresar nuevo producto

## Actors

Package in package 'Casos de uso'

Actors

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

## Actors diagram

Use Case diagram in package 'Actors'

Actors

Version 1.0

BryaN created on 30/7/2024. Last modified 30/7/2024

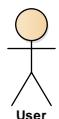


Figure 31: Actors

## User

Actor in package 'Actors'

User

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from User to Object1

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public  
To:       Use Case1 : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public  
To:       Use Case2 : UseCase, Public

## Primary Use Cases

Package in package 'Casos de uso'

Primary Use Cases

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

## Primary Use Cases diagram

Use Case diagram in package 'Primary Use Cases'

Primary Use Cases

Version 1.0

BryaN created on 30/7/2024. Last modified 30/7/2024

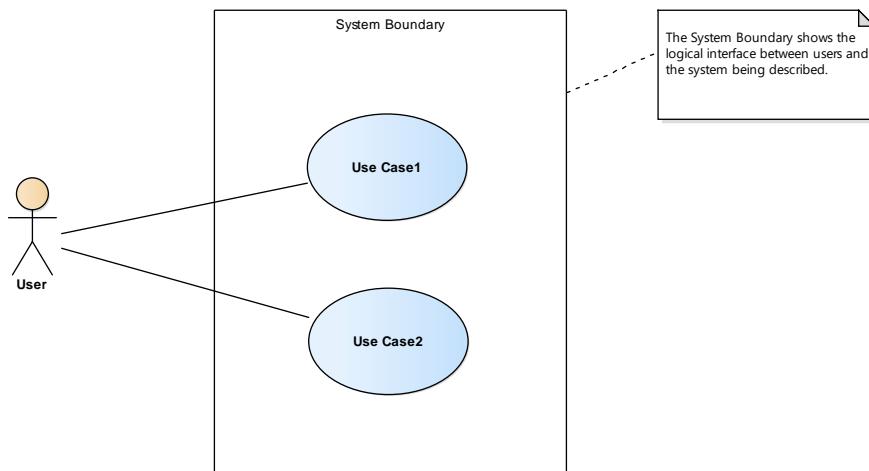


Figure 32: Primary Use Cases

## User

Actor in package 'Actors'

User

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

➡ Sequence from User to Object1

### CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public

To:       Use Case1 : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public

To:       Use Case2 : UseCase, Public

## System Boundary

Boundary in package 'Primary Use Cases'

System Boundary

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

Extends

## Note

Note in package 'Primary Use Cases'

The System Boundary shows the logical interface between users and the system being described.

Note

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

Extends

## Use Case1

UseCase in package 'Primary Use Cases'

Use Case1

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

### ELEMENTS OWNED BY Use Case1

Object1 : Sequence

### CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public

To:        Use Case1 : UseCase, Public

## Use Case1 diagram

Interaction diagram in package 'Primary Use Cases'

Use Case1

Version 1.0

BryaN created on 30/7/2024. Last modified 30/7/2024

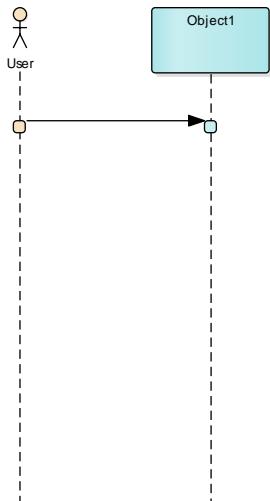


Figure 33: Use Case1

## User

Actor in package 'Actors'

User

Version 1.0 Phase 1.0 Proposed

Carolina created on 30/7/2024. Last modified 30/7/2024

### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from User to Object1

### CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public

To:       Use Case1 : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    User : Actor, Public

To:       Use Case2 : UseCase, Public

### INTERACTION MESSAGES

✉ **1.0** " from 'User' sent to 'Object1'.

Synchronous Call.

[ Return is False. Iteration is False. New group is False. ]

## Object1

Sequence owned by 'Use Case1', in package 'Primary Use Cases'

Object1  
Version 1.0 Phase 1.0 Proposed  
Carolina created on 30/7/2024. Last modified 30/7/2024

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Sequence from User to Object1

## Use Case2

UseCase in package 'Primary Use Cases'

Use Case2  
Version 1.0 Phase 1.0 Proposed  
Carolina created on 30/7/2024. Last modified 30/7/2024

#### CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    User : Actor, Public  
To:       Use Case2 : UseCase, Public

## Actor1

Actor in package 'Casos de uso'

Actor1  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## Actualizar

Actor in package 'Casos de uso'

Actualizar  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

#### CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Actualizar : Actor, Public  
To:       Actualizar Venta : UseCase, Public

## Administrador

Actor in package 'Casos de uso'

Administrador  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

 **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Eliminar Productos : UseCase, Public

**Administrador**  
Actor in package 'Casos de uso'

Administrador  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

 **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Actualizar Información de Productos : UseCase, Public

**Administrador**  
Actor in package 'Casos de uso'

Administrador  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

 **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Consultar Productos : UseCase, Public

**Administrador**  
Actor in package 'Casos de uso'

Administrador  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS



## **UseCaseLink**      Source -> Destination

From: Administrador : Actor, Public  
To: Ingresar nuevo producto : UseCase, Public

## **Administrador**

## Actor in package 'Casos de uso'

## Administrador

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS



## **UseCaseLink**      Source -> Destination

From: Administrador : Actor, Public  
To: Eliminar Venta : UseCase, Pub

## **Administrador**

## Actor in package 'Casos de uso'

## Administrador

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## ASSOCIATIONS



Association (direction: Unspecified)

Source: Public (Actor) Administrador

## Target: Public (UseCase) Consultar Ventas

## Usuario

## Actor in package 'Casos de uso'

## Usuario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## ASSOCIATIONS



Association (direction: Unspecified)

Source: Public (Actor) Usuario

Target: Public (UseCase) Agregar venta

## Vendedor

Actor in package 'Casos de uso'

Vendedor

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### CONNECTORS

#### ↗ UseCaseLink    Source -> Destination

From: Vendedor : Actor, Public

To: Eliminar Venta : UseCase, Public

## Vendedor

Actor in package 'Casos de uso'

Vendedor

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### CONNECTORS

#### ↗ UseCaseLink    Source -> Destination

From: Vendedor : Actor, Public

To: Actualizar Venta : UseCase, Public

## Vendedor

Actor in package 'Casos de uso'

Vendedor

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### ASSOCIATIONS

#### ↙ Association (direction: Unspecified)

Source: Public (Actor) Vendedor

Target: Public (UseCase) Consultar Ventas

## Seleccionar Producto

UseCase in package 'Casos de uso'

Seleccionar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Agregar venta : UseCase, Public  
To:       Seleccionar Producto : UseCase, Public

### Acceder a Eliminar Ventas

UseCase in package 'Casos de uso'

Acceder a Eliminar Ventas  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Eliminar Venta : UseCase, Public  
To:       Acceder a Eliminar Ventas : UseCase, Public

### Acceder a al modulo de Consultar Ventas

UseCase in package 'Casos de uso'

Acceder a al modulo de Consultar Ventas  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Consultar Ventas : UseCase, Public  
To:       Acceder a al modulo de Consultar Ventas : UseCase, Public

### Acceder a l modulo Actualizar Ventas

UseCase in package 'Casos de uso'

Acceder a l modulo Actualizar Ventas  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Actualizar Venta : UseCase, Public  
To:       Acceder a l modulo Actualizar Ventas : UseCase, Public

## **Acceder al Módulo de Inventario**

UseCase in package 'Casos de uso'

Acceder al Módulo de Inventario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

#### **Include** «include»    Source -> Destination

From:    Consultar Productos : UseCase, Public

To:       Acceder al Módulo de Inventario : UseCase, Public

## **Acceder al Módulo de Inventario**

UseCase in package 'Casos de uso'

Acceder al Módulo de Inventario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

#### **Include** «include»    Source -> Destination

From:    Ingresar nuevo producto : UseCase, Public

To:       Acceder al Módulo de Inventario : UseCase, Public

## **Acceder al apartado de actualizar**

UseCase in package 'Casos de uso'

Acceder al apartado de actualizar

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

#### **Include** «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public

To:       Acceder al apartado de actualizar : UseCase, Public

## **Acceder al apartado de eliminar productos**

UseCase in package 'Casos de uso'

Acceder al apartado de eliminar productos

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From:    Eliminar Productos : UseCase, Public  
To:        Acceder al apartado de eliminar productos : UseCase, Public

## Actualizar Información de Productos

UseCase in package 'Casos de uso'

Actualizar Información de Productos

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public  
To:        Modificar Información : UseCase, Public

### ↗ Include «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public  
To:        Acceder al apartado de actualizar : UseCase, Public

### ↗ Include «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public  
To:        Seleccionar Producto : UseCase, Public

### ↗ Include «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public  
To:        Guardar Cambios : UseCase, Public

### ↗ Include «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public  
To:        Confirmar Cambios : UseCase, Public

### ↗ Include «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public  
To:        Buscar Producto : UseCase, Public

### ↗ UseCaseLink    Source -> Destination

From:    Administrador : Actor, Public  
To:        Actualizar Información de Productos : UseCase, Public

## Actualizar Venta

UseCase in package 'Casos de uso'

Actualizar Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

CONNECTORS
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Actualizar Venta : UseCase, Public To:       Confirmar Cambios: : UseCase, Public</p>
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Actualizar Venta : UseCase, Public To:       Seleccionar Venta : UseCase, Public</p>
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Actualizar Venta : UseCase, Public To:       Modificar Información : UseCase, Public</p>
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Actualizar Venta : UseCase, Public To:       Buscar Venta : UseCase, Public</p>
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Actualizar Venta : UseCase, Public To:       Acceder a l modulo Actualizar Ventas : UseCase, Public</p>
<p>↗ <b>UseCaseLink</b>    Source -&gt; Destination From:    Vendedor : Actor, Public To:       Actualizar Venta : UseCase, Public</p>
<p>↗ <b>UseCaseLink</b>    Source -&gt; Destination From:    Actualizar : Actor, Public To:       Actualizar Venta : UseCase, Public</p>

## Agregar al Carrito

UseCase in package 'Casos de uso'

Agregar al Carrito

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

CONNECTORS
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Agregar venta : UseCase, Public To:       Agregar al Carrito : UseCase, Public</p>

## Agregar venta

UseCase in package 'Casos de uso'

Agregar venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ UseCaseLink    Source -> Destination

From: Agregar venta : UseCase, Public

To: Object : Object, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Buscar producto solicitado : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Seleccionar Producto : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Cobrar al Cliente : UseCase, Public

### ↗ UseCaseLink    Source -> Destination

From: Agregar venta : UseCase, Public

To: Object : Object, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Generar Venta : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Agregar al Carrito : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Ingresar Cantidad : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Repetir para Otros Productos : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Entregar Recibo al Cliente : UseCase, Public

## ASSOCIATIONS

### ↗ Association (direction: Unspecified)

Source: Public (Actor) Usuario

Target: Public (UseCase) Agregar venta

## ASSOCIATIONS

### Buscar Producto

UseCase in package 'Casos de uso'

Buscar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### Include «include» Source -> Destination

From: Eliminar Productos : UseCase, Public

To: Buscar Producto : UseCase, Public

### Buscar Producto

UseCase in package 'Casos de uso'

Buscar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### Include «include» Source -> Destination

From: Actualizar Información de Productos : UseCase, Public

To: Buscar Producto : UseCase, Public

### Buscar Producto

UseCase in package 'Casos de uso'

Buscar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### Include «include» Source -> Destination

From: Consultar Productos : UseCase, Public

To: Buscar Producto : UseCase, Public

### Buscar Venta

UseCase in package 'Casos de uso'

Buscar Venta

Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public  
To:       Buscar Venta : UseCase, Public

## Buscar Venta

UseCase in package 'Casos de uso'

Buscar Venta  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From:    Actualizar Venta : UseCase, Public  
To:       Buscar Venta : UseCase, Public

## Buscar Venta

UseCase in package 'Casos de uso'

Buscar Venta  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From:    Consultar Ventas : UseCase, Public  
To:       Buscar Venta : UseCase, Public

## Buscar producto solicitado

UseCase in package 'Casos de uso'

Buscar producto solicitado  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From:    Agregar venta : UseCase, Public  
To:       Buscar producto solicitado : UseCase, Public

## **Cobrar al Cliente**

UseCase in package 'Casos de uso'

Cobrar al Cliente

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

↗ **Include** «include»    Source -> Destination

From:    Agregar venta : UseCase, Public

To:    Cobrar al Cliente : UseCase, Public

## **Confirmar Cambios**

UseCase in package 'Casos de uso'

Confirmar Cambios

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

↗ **Include** «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public

To:    Confirmar Cambios : UseCase, Public

## **Confirmar Cambios:**

UseCase in package 'Casos de uso'

Confirmar Cambios:

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

↗ **Include** «include»    Source -> Destination

From:    Actualizar Venta : UseCase, Public

To:    Confirmar Cambios: : UseCase, Public

## **Confirmar Eliminación**

UseCase in package 'Casos de uso'

Confirmar Eliminación

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Eliminar Productos : UseCase, Public  
To:       Confirmar Eliminación : UseCase, Public

## Confirmar Eliminación

UseCase in package 'Casos de uso'

Confirmar Eliminación  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Eliminar Venta : UseCase, Public  
To:       Confirmar Eliminación : UseCase, Public

## Confirmar Ingreso

UseCase in package 'Casos de uso'

Confirmar Ingreso  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Ingresar nuevo producto : UseCase, Public  
To:       Confirmar Ingreso : UseCase, Public

## Consultar Productos

UseCase in package 'Casos de uso'

Consultar Productos  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Consultar Productos : UseCase, Public  
To:       Ver Detalles del Producto : UseCase, Public

## CONNECTORS

↗ **Include** «include» Source -> Destination

From: Consultar Productos : UseCase, Public  
To: Buscar Producto : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Consultar Productos : UseCase, Public  
To: Seleccionar Producto : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Consultar Productos : UseCase, Public  
To: Acceder al Módulo de Inventario : UseCase, Public

↗ **UseCaseLink** Source -> Destination

From: Administrador : Actor, Public  
To: Consultar Productos : UseCase, Public

## Consultar Ventas

UseCase in package 'Casos de uso'

Consultar Ventas

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include» Source -> Destination

From: Consultar Ventas : UseCase, Public  
To: Buscar Venta : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Consultar Ventas : UseCase, Public  
To: Seleccionar Venta : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Consultar Ventas : UseCase, Public  
To: Acceder a al modulo de Consultar Ventas : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Consultar Ventas : UseCase, Public  
To: Ver Detalles de Venta : UseCase, Public

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Actor) Vendedor

Target: Public (UseCase) Consultar Ventas

## ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) Administrador

Target: Public (UseCase) Consultar Ventas

## Eliminar Productos

UseCase in package 'Casos de uso'

Eliminar Productos

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

 **Include** «include» Source -> Destination

From: Eliminar Productos : UseCase, Public

To: Acceder al apartado de eliminar productos : UseCase, Public

 **Include** «include» Source -> Destination

From: Eliminar Productos : UseCase, Public

To: Buscar Producto : UseCase, Public

 **Include** «include» Source -> Destination

From: Eliminar Productos : UseCase, Public

To: Seleccionar Producto : UseCase, Public

 **Include** «include» Source -> Destination

From: Eliminar Productos : UseCase, Public

To: Confirmar Eliminación : UseCase, Public

 **UseCaseLink** Source -> Destination

From: Administrador : Actor, Public

To: Eliminar Productos : UseCase, Public

## Eliminar Venta

UseCase in package 'Casos de uso'

Eliminar Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

 **Include** «include» Source -> Destination

From: Eliminar Venta : UseCase, Public

To: Eliminar Venta: : UseCase, Public

## CONNECTORS

### ↗ **Include** «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public  
To:        Acceder a Eliminar Ventas : UseCase, Public

### ↗ **Include** «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public  
To:        Confirmar Eliminación : UseCase, Public

### ↗ **Include** «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public  
To:        Buscar Venta : UseCase, Public

### ↗ **Include** «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public  
To:        Seleccionar Venta : UseCase, Public

### ↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public  
To:        Eliminar Venta : UseCase, Public

### ↗ **UseCaseLink**    Source -> Destination

From:    Administrador : Actor, Public  
To:        Eliminar Venta : UseCase, Public

## **Eliminar Venta:**

UseCase in package 'Casos de uso'

Eliminar Venta:

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

### ↗ **Include** «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public  
To:        Eliminar Venta: : UseCase, Public

## **Entregar Recibo al Cliente**

UseCase in package 'Casos de uso'

Entregar Recibo al Cliente

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Entregar Recibo al Cliente : UseCase, Public

## Generar Venta

UseCase in package 'Casos de uso'

Generar Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Generar Venta : UseCase, Public

## Guardar Cambios

UseCase in package 'Casos de uso'

Guardar Cambios

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From: Actualizar Información de Productos : UseCase, Public

To: Guardar Cambios : UseCase, Public

## Guardar Producto

UseCase in package 'Casos de uso'

Guardar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From: Ingresar nuevo producto : UseCase, Public

To: Guardar Producto : UseCase, Public

## Ingresar Cantidad

UseCase in package 'Casos de uso'

Ingresar Cantidad

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Agregar venta : UseCase, Public

To:    Ingresar Cantidad : UseCase, Public

## Ingresar Datos del Producto

UseCase in package 'Casos de uso'

Ingresar Datos del Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Ingresar nuevo producto : UseCase, Public

To:    Ingresar Datos del Producto : UseCase, Public

## Ingresar nuevo producto

UseCase in package 'Casos de uso'

Ingresar nuevo producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Ingresar nuevo producto : UseCase, Public

To:    Acceder al Módulo de Inventario : UseCase, Public

↗ **Include** «include»    Source -> Destination

From:    Ingresar nuevo producto : UseCase, Public

To:    Ingresar Datos del Producto : UseCase, Public

↗ **Include** «include»    Source -> Destination

From:    Ingresar nuevo producto : UseCase, Public

To:    Confirmar Ingreso : UseCase, Public

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Ingresar nuevo producto : UseCase, Public  
To:       Guardar Producto : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Ingresar nuevo producto : UseCase, Public

## Modificar Información

UseCase in package 'Casos de uso'

Modificar Información  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Actualizar Información de Productos : UseCase, Public  
To:       Modificar Información : UseCase, Public

## Modificar Información

UseCase in package 'Casos de uso'

Modificar Información  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Actualizar Venta : UseCase, Public  
To:       Modificar Información : UseCase, Public

## Repetir para Otros Productos

UseCase in package 'Casos de uso'

Repetir para Otros Productos  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Agregar venta : UseCase, Public

To:    Repetir para Otros Productos : UseCase, Public

## Seleccionar Producto

UseCase in package 'Casos de uso'

Seleccionar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Eliminar Productos : UseCase, Public

To:    Seleccionar Producto : UseCase, Public

## Seleccionar Producto

UseCase in package 'Casos de uso'

Seleccionar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Actualizar Información de Productos : UseCase, Public

To:    Seleccionar Producto : UseCase, Public

## Seleccionar Producto

UseCase in package 'Casos de uso'

Seleccionar Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Consultar Productos : UseCase, Public

To:    Seleccionar Producto : UseCase, Public

## **Seleccionar Venta**

UseCase in package 'Casos de uso'

Seleccionar Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

↗ **Include** «include»    Source -> Destination

From:    Eliminar Venta : UseCase, Public

To:       Seleccionar Venta : UseCase, Public

## **Seleccionar Venta**

UseCase in package 'Casos de uso'

Seleccionar Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

↗ **Include** «include»    Source -> Destination

From:    Actualizar Venta : UseCase, Public

To:       Seleccionar Venta : UseCase, Public

## **Seleccionar Venta**

UseCase in package 'Casos de uso'

Seleccionar Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **CONNECTORS**

↗ **Include** «include»    Source -> Destination

From:    Consultar Ventas : UseCase, Public

To:       Seleccionar Venta : UseCase, Public

## **Ver Detalles de Venta**

UseCase in package 'Casos de uso'

Ver Detalles de Venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Consultar Ventas : UseCase, Public  
To:       Ver Detalles de Venta : UseCase, Public

## Ver Detalles del Producto

UseCase in package 'Casos de uso'

Ver Detalles del Producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **Include** «include»    Source -> Destination

From:    Consultar Productos : UseCase, Public  
To:       Ver Detalles del Producto : UseCase, Public

## Object

Object in package 'Casos de uso'

Object

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## Object

Object in package 'Casos de uso'

Object

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    Agregar venta : UseCase, Public  
To:       Object : Object, Public

## Object

Object in package 'Casos de uso'

Object

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## CONNECTORS

 **UseCaseLink**    Source -> Destination

From:    Agregar venta : UseCase, Public

To:      Object : Object, Public

Diagrama de secuencias  
Package in package 'Use Case Model'

Diagrama de secuencias  
Version Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Actualizar Venta diagram**  
Interaction diagram in package 'Diagrama de secuencias'

Actualizar Venta  
Version 1.0  
Jennifer2002 created on 30/7/2024. Last modified 31/7/2024

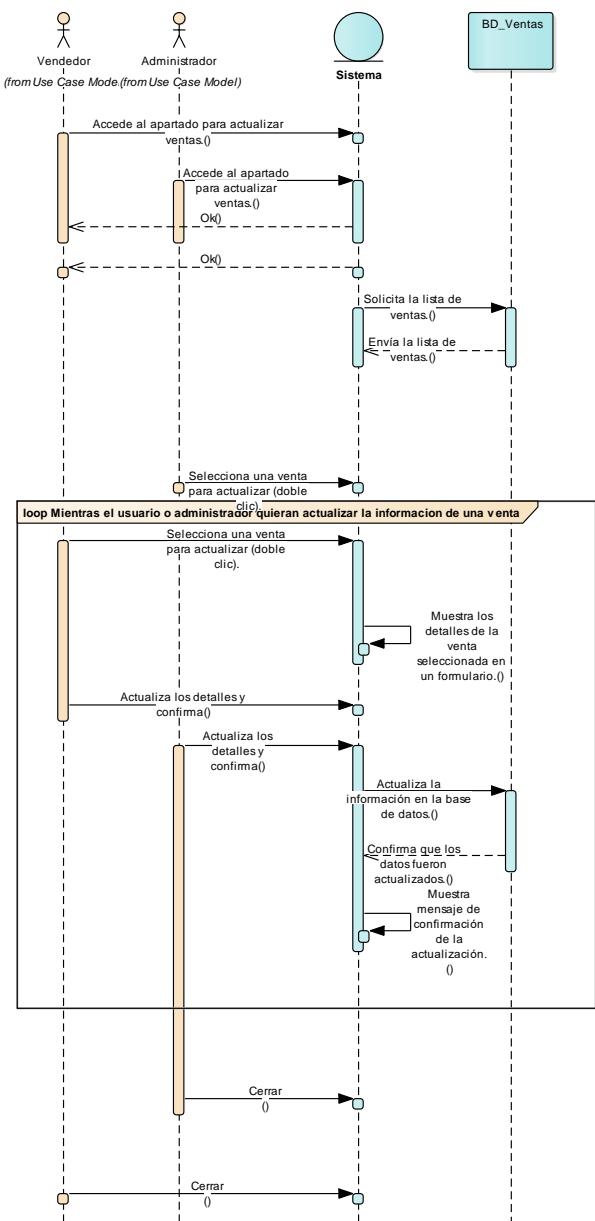


Figure 34: Actualizar Venta

## Vendedor

Actor in package 'Use Case Model'

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

### ELEMENTS OWNED BY VENDEDOR

- Administrator : Actor

### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Vendedor to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Vendedor to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar productos.

↳ Sequence from Vendedor to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar  
↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado  
↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda  
↳ Sequence from Vendedor to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Vendedor

Name: OK  
⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok  
⇒ Sequence from «entity» Sistema to Vendedor

Name: Oñ  
⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok  
⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok  
⇒ Sequence from «entity» Sistema to Vendedor

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestión venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestión venta : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

➡ Sequence from «entity» Sistema to Administrador

Name: Ok

➡ Sequence from «entity» Sistema to Administrador

Name: Ok

➡ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink** Source -> Destination

From: Administrador : Actor, Public

To: Gestionar venta : UseCase, Public

↗ **UseCaseLink** Source -> Destination

From: Administrador : Actor, Public

To: Gestionar inventario : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

➡ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

➡ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

➡ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

➡ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

➡ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

➡ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Actualiza los detalles y confirma  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: elige un producto para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada  
↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".  
↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.  
↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

- ✉ 1.0 'Accede al apartado para actualizar ventas.'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.1 'Accede al apartado para actualizar ventas.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.2 'Ok'** from 'Sistema' sent to 'Vendedor'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.3 'Ok'** from 'Sistema' sent to 'Vendedor'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.4 'Solicita la lista de ventas.'** from 'Sistema' sent to 'BD\_Ventas'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.5 'Envía la lista de ventas.'** from 'BD\_Ventas' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.6 'Selecciona una venta para actualizar (doble clic).'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.7 'Selecciona una venta para actualizar (doble clic).'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.8 'Muestra los detalles de la venta seleccionada en un formulario.'** from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.9 'Actualiza los detalles y confirma'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.10 'Actualiza los detalles y confirma' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.11 'Actualiza la información en la base de datos.' from 'Sistema' sent to 'BD\_Ventas'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.12 'Confirma que los datos fueron actualizados.' from 'BD\_Ventas' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.13 'Muestra mensaje de confirmación de la actualización.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.14 'Cerrar' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.15 'Cerrar' from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

## Actualizar productos diagram

Interaction diagram in package 'Diagrama de secuencias'

Actualizar productos

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

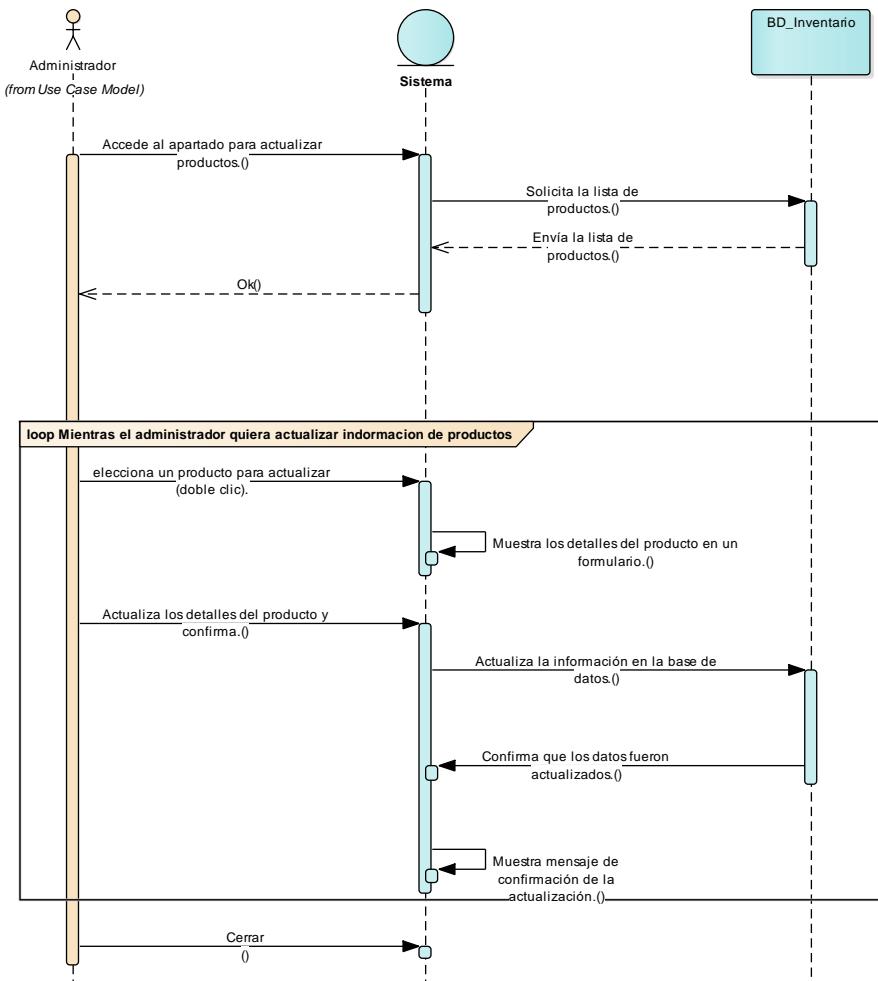


Figure 35: Actualizar productos

### Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

#### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada  
↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: elige un producto para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada  
↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".  
↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Actualiza los detalles del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

## CONNECTORS

 **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

 **1.0 'Accede al apartado para actualizar productos.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

 **1.1 'Solicita la lista de productos.'** from 'Sistema' sent to 'BD\_Inventario'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

 **1.2 'Envía la lista de productos.'** from 'BD\_Inventario' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is True. Iteration is False. New group is False. ]

 **1.3 'Ok'** from 'Sistema' sent to 'Administrador'.

Synchronous Call. Returns void.  
[ Return is True. Iteration is False. New group is False. ]

 **1.4 'elecciona un producto para actualizar (doble clic).'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

 **1.5 'Muestra los detalles del producto en un formulario.'** from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

 **1.6 'Actualiza los detalles del producto y confirma.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

 **1.7 'Actualiza la información en la base de datos.'** from 'Sistema' sent to 'BD\_Inventario'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

✉ 1.8 'Confirma que los datos fueron actualizados.' from 'BD\_Inventario' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.9 'Muestra mensaje de confirmación de la actualización.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.10 'Cerrar' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

## Consultar Productos diagram

Interaction diagram in package 'Diagrama de secuencias'

Consultar Productos

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

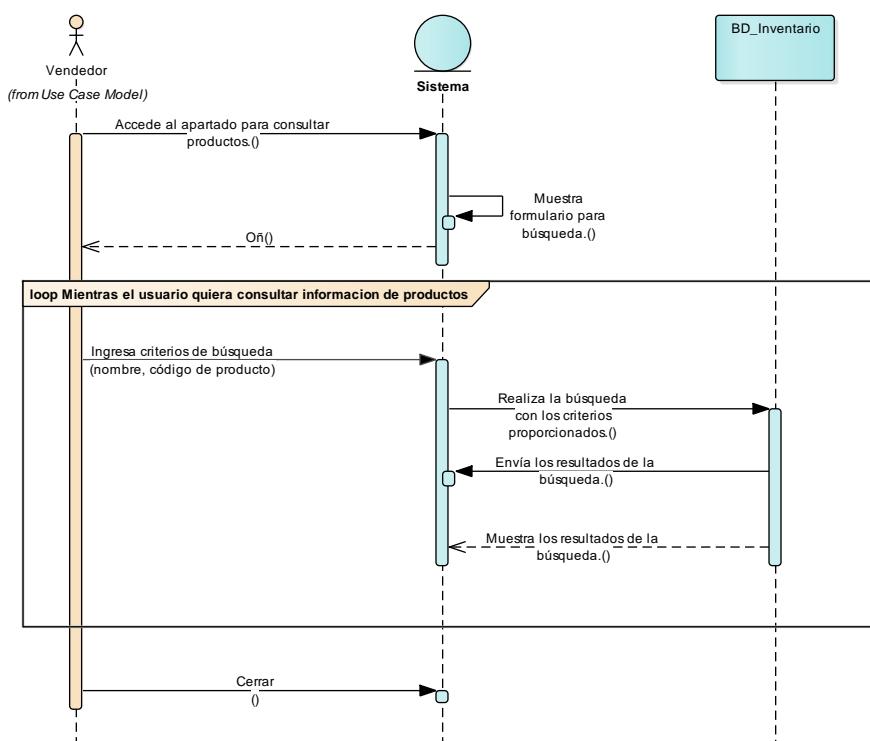


Figure 36: Consultar Productos

## Vendedor

Actor in package 'Use Case Model'

Vendedor

ELEMENTS OWNED BY VENDEDOR
<p>Administrator : Actor</p>
OUTGOING BEHAVIORAL RELATIONSHIPS
<p>Name: Busca un producto en la tabla y selecciona uno (doble clic). ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Cerrar ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Accede al apartado para registrar una venta. ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Confirma la eliminación. ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Ingresa criterios de búsqueda ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Selecciona una venta para eliminar (click derecho y opción eliminar). ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Actualiza los detalles y confirma ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Selecciona una venta para actualizar (doble clic). ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Accede al apartado para consultar ventas. ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Accede al apartado para eliminar ventas. ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Finaliza la compra y da clic en "Generar Venta". ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Accede al apartado para consultar productos. ↳ Sequence from Vendedor to «entity» Sistema</p>
<p>Name: Accede al apartado para actualizar ventas. ↳ Sequence from Vendedor to «entity» Sistema</p>

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

Name: OK

⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

Name: Oñ

⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public

To:       Gestion venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public

To:       Gestionar venta : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

OUTGOING BEHAVIORAL RELATIONSHIPS
Name: Accede al apartado para eliminar productos. ↳ Sequence from Administrador to «entity» Sistema
Name: Confirma la eliminación. ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para ingresar nuevos productos. ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para registrar una venta. ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para eliminar ventas. ↳ Sequence from Administrador to «entity» Sistema
Name: Ingresa la cantidad deseada ↳ Sequence from Administrador to «entity» Sistema
Name: Actualiza los detalles y confirma ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para consultar ventas. ↳ Sequence from Administrador to «entity» Sistema
Name: Ingresa datos del producto y confirma. ↳ Sequence from Administrador to «entity» Sistema
Name: Confirma la eliminación. ↳ Sequence from Administrador to «entity» Sistema
Name: Ingresa criterios de búsqueda ↳ Sequence from Administrador to «entity» Sistema
Name: Cerrar ↳ Sequence from Administrador to «entity» Sistema
Name: elige un producto para actualizar (doble clic). ↳ Sequence from Administrador to «entity» Sistema
Name: Cerrar ↳ Sequence from Administrador to «entity» Sistema
Name: Cerrar ↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    Administrador : Actor, Public

To:       Gestionar venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    Administrador : Actor, Public

To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

✉ **1.0 'Accede al apartado para consultar productos.'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ **1.1 'Muestra formulario para búsqueda.'** from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ **1.2 'On'** from 'Sistema' sent to 'Vendedor'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

✉ **1.3 'Ingresa criterios de búsqueda'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ **1.4 'Realiza la búsqueda con los criterios proporcionados.'** from 'Sistema' sent to 'BD\_Inventario'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.5 'Envía los resultados de la búsqueda.'** from 'BD\_Inventario' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.6 'Muestra los resultados de la búsqueda.'** from 'BD\_Inventario' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ **1.7 'Cerrar'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

## Consultar ventas diagram

Interaction diagram in package 'Diagrama de secuencias'

Consultar ventas

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

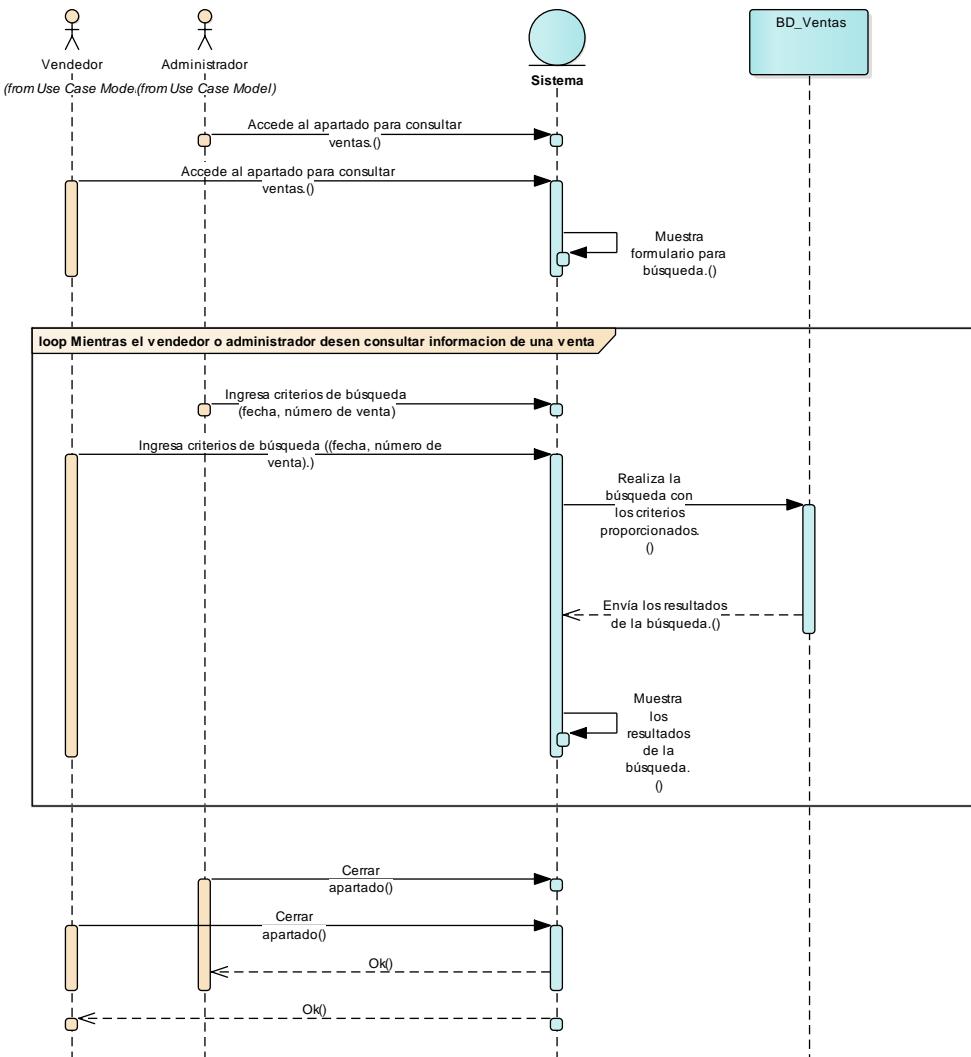


Figure 37: Consultar ventas

### Vendedor

Actor in package 'Use Case Model'

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

#### ELEMENTS OWNED BY VENDEDOR

- Administrator : Actor

#### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para registrar una venta.

↳ Sequence from Vendedor to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Vendedor to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar productos.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

## INCOMING BEHAVIORAL RELATIONSHIPS

➡ Sequence from «entity» Sistema to Vendedor

Name: OK

➡ Sequence from «entity» Sistema to Vendedor

Name: Ok

➡ Sequence from «entity» Sistema to Vendedor

Name: Oñ

➡ Sequence from «entity» Sistema to Vendedor

Name: Ok

➡ Sequence from «entity» Sistema to Vendedor

Name: Ok

➡ Sequence from «entity» Sistema to Vendedor

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public

To:       Gestion venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public

To:       Gestionar venta : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    Administrador : Actor, Public

To:        Gestionar venta : UseCase, Public

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

✉ **1.0 'Accede al apartado para consultar ventas.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

✉ **1.1 'Accede al apartado para consultar ventas.'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

✉ **1.2 'Muestra formulario para búsqueda.'** from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.3 'Ingresa criterios de búsqueda '** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.4 'Ingresa criterios de búsqueda '** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.5 'Realiza la búsqueda con los criterios proporcionados.'** from 'Sistema' sent to 'BD\_Ventas'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.6 'Envía los resultados de la búsqueda.'** from 'BD\_Ventas' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.7 'Muestra los resultados de la búsqueda.'** from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.8 'Cerrar apartado'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.9 'Cerrar apartado'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.10 'Ok'** from 'Sistema' sent to 'Administrador'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.11 'Ok'** from 'Sistema' sent to 'Vendedor'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

## **Eliminar productos diagram**

Interaction diagram in package 'Diagrama de secuencias'

Eliminar productos

Version 1.0

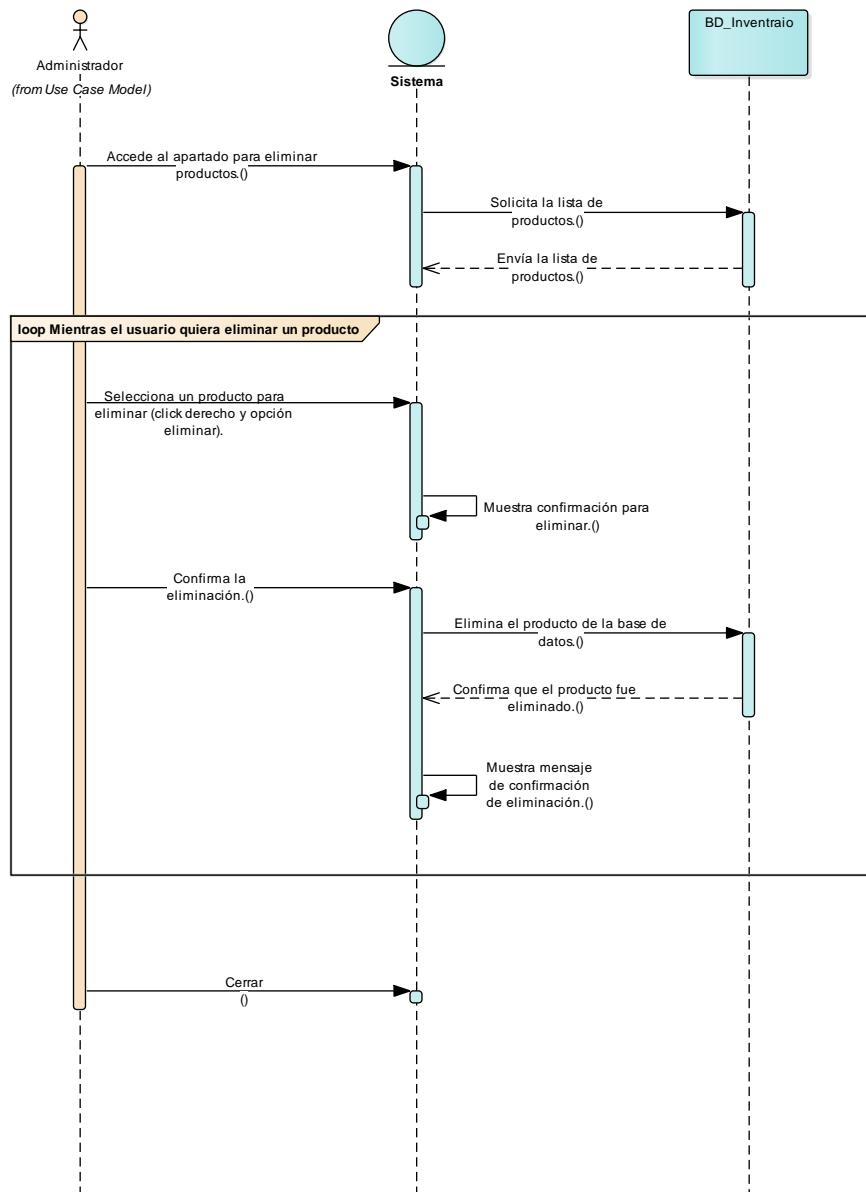


Figure 38: Eliminar productos

### Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

#### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para ingresar nuevos productos.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada  
↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

## CONNECTORS

### UseCaseLink    Source -> Destination

From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

### UseCaseLink    Source -> Destination

From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

### 1.0 'Accede al apartado para eliminar productos.' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

### 1.1 'Solicita la lista de productos.' from 'Sistema' sent to 'BD\_Inventraio'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

### 1.2 'Envía la lista de productos.' from 'BD\_Inventraio' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

### 1.3 'Selecciona un producto para eliminar (click derecho y opción eliminar).' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

### 1.4 'Muestra confirmación para eliminar.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

### 1.5 'Confirma la eliminación.' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

### 1.6 'Elimina el producto de la base de datos.' from 'Sistema' sent to 'BD\_Inventraio'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.7 'Confirma que el producto fue eliminado.' from 'BD\_Inventraio' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

✉ 1.8 'Muestra mensaje de confirmación de eliminación.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.9 'Cerrar' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

## Eliminar ventas diagram

Interaction diagram in package 'Diagrama de secuencias'

Eliminar ventas

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

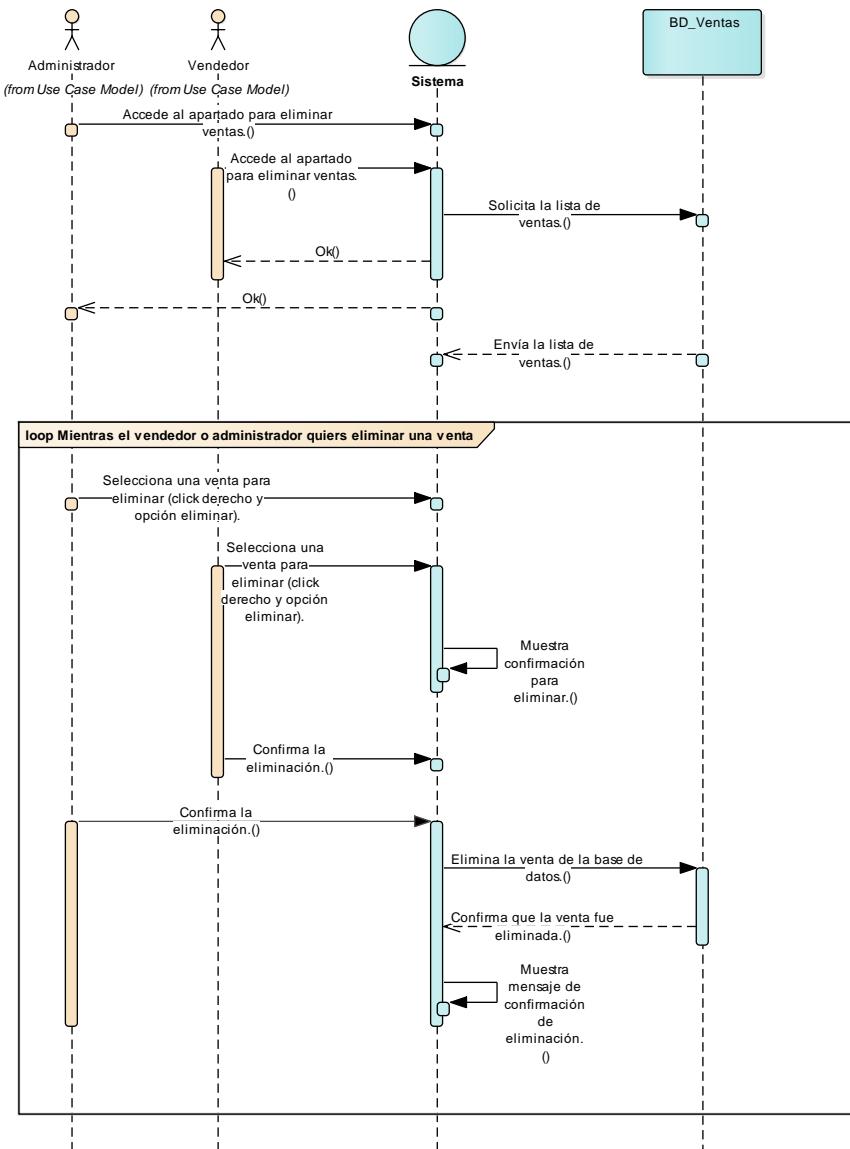


Figure 39: Eliminar ventas

## Vendedor

Actor in package 'Use Case Model'

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

### ELEMENTS OWNED BY Vendedor

- Administrador : Actor

### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Busca un producto en la tabla y selecciona uno (doble clic).

Sequence from Vendedor to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Vendedor to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Vendedor to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar productos.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

Name: OK  
➡ Sequence from «entity» Sistema to Vendedor

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

Name: Oñ  
➡ Sequence from «entity» Sistema to Vendedor

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestion venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestionar venta : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.  
➡ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
➡ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.  
➡ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

Name: Ok

↗ Sequence from «entity» Sistema to Administrador

## CONNECTORS

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public  
To: Gestionar venta : UseCase, Public

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public  
To: Gestionar inventario : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: selecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

✉ **1.0 'Accede al apartado para eliminar ventas.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

✉ **1.1 'Accede al apartado para eliminar ventas.'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.2 'Solicita la lista de ventas.'** from 'Sistema' sent to 'BD\_Ventas'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.3 'Ok'** from 'Sistema' sent to 'Vendedor'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ **1.4 'Ok'** from 'Sistema' sent to 'Administrador'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ **1.5 'Envía la lista de ventas.'** from 'BD\_Ventas' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ **1.6 'Selecciona una venta para eliminar (click derecho y opción eliminar).'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.7 'Selecciona una venta para eliminar (click derecho y opción eliminar).'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.8 'Muestra confirmación para eliminar.'** from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.9 'Confirma la eliminación.'** from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.10 'Confirma la eliminación.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ **1.11 'Elimina la venta de la base de datos.'** from 'Sistema' sent to 'BD\_Ventas'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.12 'Confirma que la venta fue eliminada.' from 'BD\_Ventas' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.13 'Muestra mensaje de confirmación de eliminación.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

## Ingresar Nuevos Productos diagram

Interaction diagram in package 'Diagrama de secuencias'

Ingresar Nuevos Productos

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

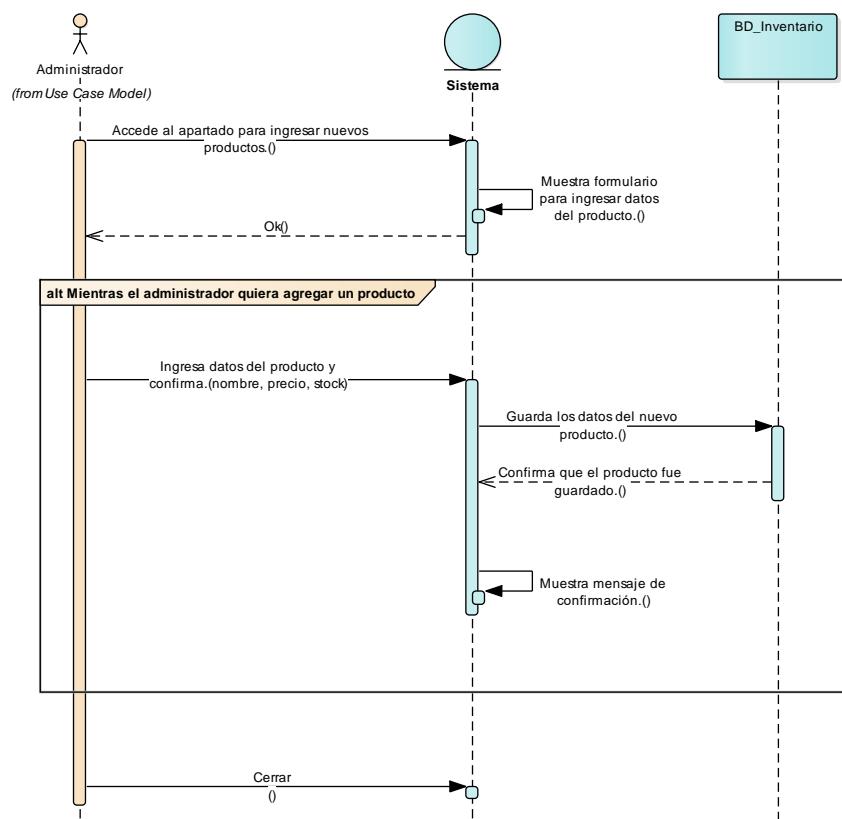


Figure 40: Ingresar Nuevos Productos

### Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.  
↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada  
↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public  
To: Gestionar venta : UseCase, Public

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public  
To: Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

✉ 1.0 'Accede al apartado para ingresar nuevos productos.' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.1 'Muestra formulario para ingresar datos del producto.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.2 'Ok' from 'Sistema' sent to 'Administrador'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

✉ 1.3 'Ingresa datos del producto y confirma.' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.4 'Guarda los datos del nuevo producto.' from 'Sistema' sent to 'BD\_Inventario'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.5 'Confirma que el producto fue guardado.' from 'BD\_Inventario' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.6 'Muestra mensaje de confirmación.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.7 'Cerrar' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

## Registrar venta diagram

Interaction diagram in package 'Diagrama de secuencias'

Registrar venta

Version 1.0

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

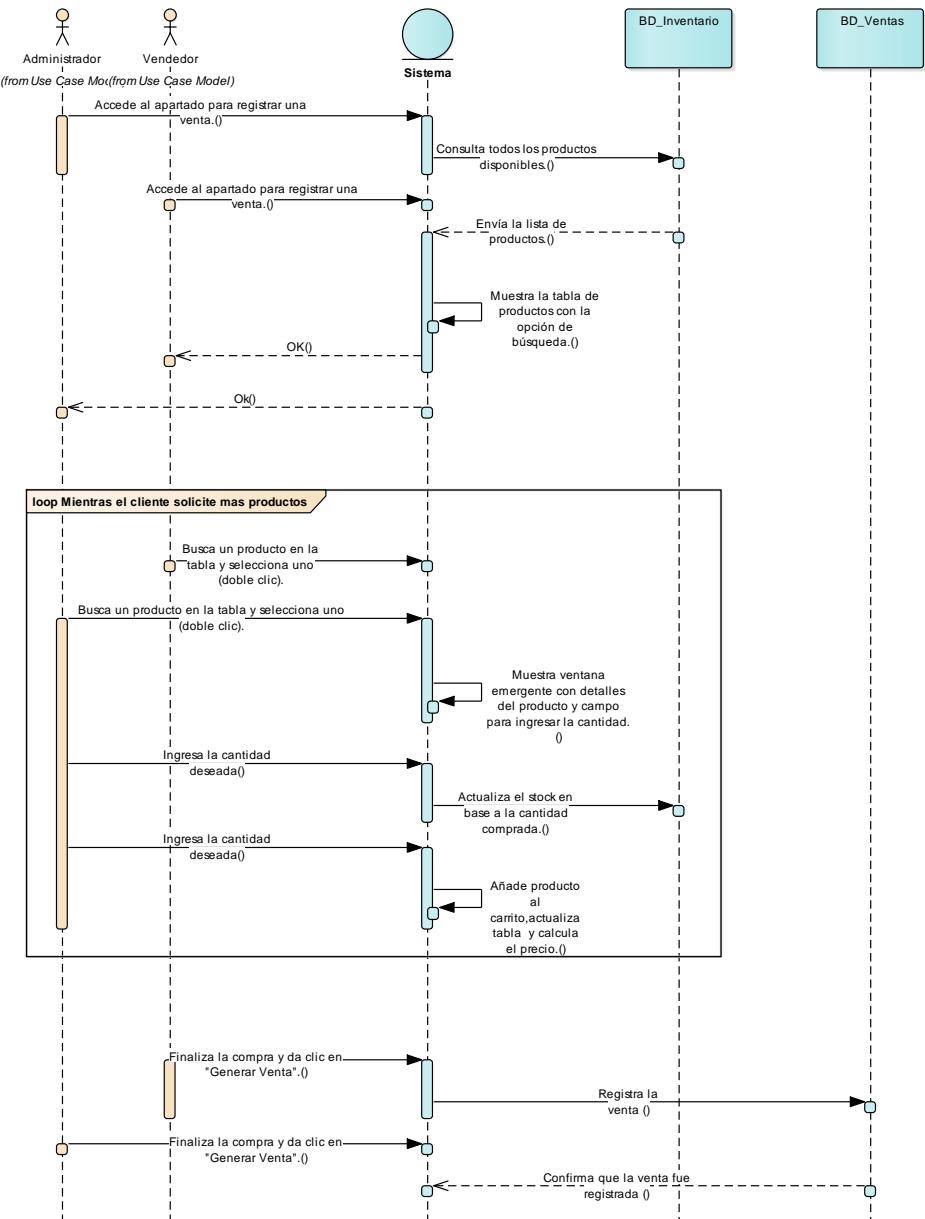


Figure 41: Registrar venta

## Vendedor

Actor in package 'Use Case Model'

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

### ELEMENTS OWNED BY Vendedor

- Administrador : Actor

### OUTGOING BEHAVIORAL RELATIONSHIPS

- Name: Busca un producto en la tabla y selecciona uno (doble clic).
  - Sequence from Vendedor to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Vendedor to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Vendedor to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar productos.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

Name: OK  
➡ Sequence from «entity» Sistema to Vendedor

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

Name: Oñ  
➡ Sequence from «entity» Sistema to Vendedor

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

Name: Ok  
➡ Sequence from «entity» Sistema to Vendedor

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestion venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestionar venta : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador  
Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.  
⬅ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.  
⬅ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: elecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Finaliza la compra y da clic en "Generar Venta".  
↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).  
↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).  
↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado  
↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public  
To: Gestionar venta : UseCase, Public

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public  
To: Gestionar inventario : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa datos del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: selecciona un producto para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para actualizar ventas.  
↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar inventario : UseCase, Public

## INTERACTION MESSAGES

✉ **1.0 'Accede al apartado para registrar una venta.'** from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

✉ **1.1 'Consulta todos los productos disponibles.'** from 'Sistema' sent to 'BD\_Inventario'.

Synchronous Call. Returns void.  
[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.2 'Accede al apartado para registrar una venta.' from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.3 'Envía la lista de productos.' from 'BD\_Inventario' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.4 'Muestra la tabla de productos con la opción de búsqueda.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.5 'OK' from 'Sistema' sent to 'Vendedor'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.6 'Ok' from 'Sistema' sent to 'Administrador'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

- ✉ 1.7 'Busca un producto en la tabla y selecciona uno (doble clic).' from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.8 'Busca un producto en la tabla y selecciona uno (doble clic).' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.9 'Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.10 'Ingresa la cantidad deseada' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

- ✉ 1.11 'Actualiza el stock en base a la cantidad comprada.' from 'Sistema' sent to 'BD\_Inventario'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.12 'Ingresa la cantidad deseada' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.13 'Añade producto al carrito,actualiza tabla y calcula el precio.' from 'Sistema' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.14 ' Finaliza la compra y da clic en "Generar Venta".' from 'Vendedor' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.15 'Registra la venta ' from 'Sistema' sent to 'BD\_Ventas'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.16 ' Finaliza la compra y da clic en "Generar Venta".' from 'Administrador' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is False. Iteration is False. New group is False. ]

✉ 1.17 'Confirma que la venta fue registrada ' from 'BD\_Ventas' sent to 'Sistema'.

Synchronous Call. Returns void.

[ Return is True. Iteration is False. New group is False. ]

## Domain Objects

Package in package 'Diagrama de secuencias'

Domain Objects

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## Domain Objects diagram

Class diagram in package 'Domain Objects'

Domain Objects

Version 1.0

BryaN created on 30/7/2024. Last modified 30/7/2024

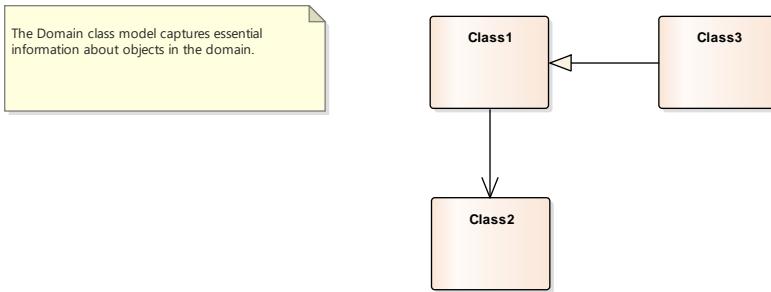


Figure 42: Domain Objects

## Note

Note in package 'Domain Objects'

The Domain class model captures essential information about objects in the domain.

Note

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

Extends

## Class1

Class in package 'Domain Objects'

Class1

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Generalization from Class3 to Class1  
[ Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

- Association (direction: Source -> Destination)

Source: Public (Class) Class1

Target: Public (Class) Class2

## Class2

Class in package 'Domain Objects'

Class2

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## ASSOCIATIONS

↗ Association (direction: Source -> Destination)

Source: Public (Class) Class1

Target: Public (Class) Class2

## Class3

Class in package 'Domain Objects'

Class3

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

Extends Class1

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Generalization from Class3 to Class1

[ Direction is 'Source -> Destination'. ]

## Actor1

Actor in package 'Diagrama de secuencias'

Actor1

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## Vendedor

Actor in package 'Diagrama de secuencias'

Vendedor

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## BD\_Inventario

Sequence in package 'Diagrama de secuencias'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Envía la lista de productos.

➡ Sequence from BD\_Inventario to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Confirma que los datos fueron actualizados.

↳ Sequence from BD\_Inventario to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Actualiza la información en la base de datos.

↗ Sequence from «entity» Sistema to BD\_Inventario

Name: Sigue la lista de productos.

↗ Sequence from «entity» Sistema to BD\_Inventario

## BD\_Inventario

Sequence in package 'Diagrama de secuencias'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra los resultados de la búsqueda.

↳ Sequence from BD\_Inventario to «entity» Sistema

Name: Envía los resultados de la búsqueda.

↳ Sequence from BD\_Inventario to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Realiza la búsqueda con los criterios proporcionados.

↗ Sequence from «entity» Sistema to BD\_Inventario

## BD\_Inventario

Sequence in package 'Diagrama de secuencias'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Confirma que el producto fue guardado.

↳ Sequence from BD\_Inventario to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Guarda los datos del nuevo producto.  
⇒ Sequence from «entity» Sistema to BD\_Inventario

## BD\_Inventario

Sequence in package 'Diagrama de secuencias'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Envía la lista de productos.  
⇐ Sequence from BD\_Inventario to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Consulta todos los productos disponibles.  
⇒ Sequence from «entity» Sistema to BD\_Inventario

Name: Actualiza el stock en base a la cantidad comprada.

⇒ Sequence from «entity» Sistema to BD\_Inventario

## BD\_Inventraio

Sequence in package 'Diagrama de secuencias'

BD\_Inventraio

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Confirma que el producto fue eliminado.  
⇐ Sequence from BD\_Inventraio to «entity» Sistema

Name: Envía la lista de productos.

⇐ Sequence from BD\_Inventraio to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Sigue la lista de productos.  
⇒ Sequence from «entity» Sistema to BD\_Inventraio

Name: Elimina el producto de la base de datos.

⇒ Sequence from «entity» Sistema to BD\_Inventraio

## INCOMING BEHAVIORAL RELATIONSHIPS

### BD\_Ventas

Sequence in package 'Diagrama de secuencias'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Envía la lista de ventas.

↳ Sequence from BD\_Ventas to «entity» Sistema

Name: Confirma que la venta fue eliminada.

↳ Sequence from BD\_Ventas to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Solicita la lista de ventas.

↗ Sequence from «entity» Sistema to BD\_Ventas

Name: Elimina la venta de la base de datos.

↗ Sequence from «entity» Sistema to BD\_Ventas

### BD\_Ventas

Sequence in package 'Diagrama de secuencias'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 31/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Confirma que los datos fueron actualizados.

↳ Sequence from BD\_Ventas to «entity» Sistema

Name: Envía la lista de ventas.

↳ Sequence from BD\_Ventas to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Actualiza la información en la base de datos.

↗ Sequence from «entity» Sistema to BD\_Ventas

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Sigue la lista de ventas.

⇒ Sequence from «entity» Sistema to BD\_Ventas

## BD\_Ventas

Sequence in package 'Diagrama de secuencias'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Envía los resultados de la búsqueda.

⇐ Sequence from BD\_Ventas to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Realiza la búsqueda con los criterios proporcionados.

⇒ Sequence from «entity» Sistema to BD\_Ventas

## BD\_Ventas

Sequence in package 'Diagrama de secuencias'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Confirma que la venta fue registrada

⇐ Sequence from BD\_Ventas to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Registra la venta

⇒ Sequence from «entity» Sistema to BD\_Ventas

## Mientras el administrador quiera actualizar indormacion de productos

InteractionFragment in package 'Diagrama de secuencias'

Mientras el administrador quiera actualizar indormacion de productos

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el administrador quiera agregar un producto**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el administrador quiera agregar un producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el cliente solicite mas productos**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el cliente solicite mas productos

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el usuario o administrador quieran actualizar la informacion de una venta**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el usuario o administrador quieran actualizar la informacion de una venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el usuario quiera consultar informacion de productos**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el usuario quiera consultar informacion de productos

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el usuario quiera eliminar un producto**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el usuario quiera eliminar un producto

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el vendedor o administrador desen consultar informacion de una venta**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el vendedor o administrador desen consultar informacion de una venta

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

**Mientras el vendedor o administrador quiers eliminar una venta**

InteractionFragment in package 'Diagrama de secuencias'

Mientras el vendedor o administrador quiera eliminar una venta  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **Object1**

Sequence in package 'Diagrama de secuencias'

Object1  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

### **Object2**

Sequence in package 'Diagrama de secuencias'

Object2  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 31/7/2024. Last modified 31/7/2024

### **Object3**

Sequence in package 'Diagrama de secuencias'

Object3  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 31/7/2024. Last modified 31/7/2024

### **Sistema**

Sequence «entity» in package 'Diagrama de secuencias'

Sistema  
Version 1.0 Phase 1.0 Proposed  
Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

#### **OUTGOING BEHAVIORAL RELATIONSHIPS**

Name: Solicita la lista de productos.  
↳ Sequence from «entity» Sistema to BD\_Inventraio

Name: Muestra confirmación para eliminar.  
↳ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra mensaje de confirmación de eliminación.  
↳ Sequence from «entity» Sistema to «entity» Sistema

Name: Elimina el producto de la base de datos.  
↳ Sequence from «entity» Sistema to BD\_Inventraio

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Accede al apartado para eliminar productos.  
⇒ Sequence from Administrador to «entity» Sistema

Name: Confirma que el producto fue eliminado.  
⇒ Sequence from BD\_Inventraio to «entity» Sistema

Name: Confirma la eliminación.  
⇒ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
⇒ Sequence from Administrador to «entity» Sistema

Name: Envía la lista de productos.  
⇒ Sequence from BD\_Inventraio to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).  
⇒ Sequence from Administrador to «entity» Sistema

Name: Muestra confirmación para eliminar.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra mensaje de confirmación de eliminación.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra los detalles del producto en un formulario.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra mensaje de confirmación de la actualización.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Actualiza la información en la base de datos.  
⇐ Sequence from «entity» Sistema to BD\_Inventario

Name: Ok  
⇐ Sequence from «entity» Sistema to Administrador

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Solicita la lista de productos.

↳ Sequence from «entity» Sistema to BD\_Inventario

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Muestra los detalles del producto en un formulario.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Envía la lista de productos.

⇒ Sequence from BD\_Inventario to «entity» Sistema

Name: Muestra mensaje de confirmación de la actualización.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: elige un producto para actualizar (doble clic).

⇒ Sequence from Administrador to «entity» Sistema

Name: Cerrar

⇒ Sequence from Administrador to «entity» Sistema

Name: Confirma que los datos fueron actualizados.

⇒ Sequence from BD\_Inventario to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

⇒ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

⇒ Sequence from Administrador to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra formulario para búsqueda.

↳ Sequence from «entity» Sistema to «entity» Sistema

Name: Of

↳ Sequence from «entity» Sistema to Vendedor

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Realiza la búsqueda con los criterios proporcionados.

↳ Sequence from «entity» Sistema to BD\_Inventario

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Muestra formulario para búsqueda.

↗ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra los resultados de la búsqueda.

↗ Sequence from BD\_Inventario to «entity» Sistema

Name: Ingresa criterios de búsqueda

↗ Sequence from Vendedor to «entity» Sistema

Name: Envía los resultados de la búsqueda.

↗ Sequence from BD\_Inventario to «entity» Sistema

Name: Accede al apartado para consultar productos.

↗ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↗ Sequence from Vendedor to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra mensaje de confirmación.

↳ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra formulario para ingresar datos del producto.

↳ Sequence from «entity» Sistema to «entity» Sistema

Name: Guarda los datos del nuevo producto.

↳ Sequence from «entity» Sistema to BD\_Inventario

Name: Ok

↳ Sequence from «entity» Sistema to Administrador

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Muestra mensaje de confirmación.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Accede al apartado para ingresar nuevos productos.  
⇒ Sequence from Administrador to «entity» Sistema

Name: Muestra formulario para ingresar datos del producto.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Ingresa datos del producto y confirma.  
⇒ Sequence from Administrador to «entity» Sistema

Name: Cerrar  
⇒ Sequence from Administrador to «entity» Sistema

Name: Confirma que el producto fue guardado.  
⇒ Sequence from BD\_Inventario to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra mensaje de confirmación de eliminación.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Ok  
⇐ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇐ Sequence from «entity» Sistema to Vendedor

Name: Sigue la lista de ventas.  
⇐ Sequence from «entity» Sistema to BD\_Ventas

Name: Elimina la venta de la base de datos.  
⇐ Sequence from «entity» Sistema to BD\_Ventas

Name: Muestra confirmación para eliminar.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Confirma la eliminación.

⇒ Sequence from Administrador to «entity» Sistema

Name: Confirma la eliminación.

⇒ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

⇒ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

⇒ Sequence from Vendedor to «entity» Sistema

Name: Envía la lista de ventas.

⇒ Sequence from BD\_Ventas to «entity» Sistema

Name: Muestra mensaje de confirmación de eliminación.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

⇒ Sequence from Vendedor to «entity» Sistema

Name: Confirma que la venta fue eliminada.

⇒ Sequence from BD\_Ventas to «entity» Sistema

Name: Muestra confirmación para eliminar.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

⇒ Sequence from Administrador to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇐ Sequence from «entity» Sistema to Vendedor

Name: Muestra mensaje de confirmación de la actualización.

⇐ Sequence from «entity» Sistema to «entity» Sistema

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Actualiza la información en la base de datos.  
↳ Sequence from «entity» Sistema to BD\_Ventas

Name: Ok  
↳ Sequence from «entity» Sistema to Vendedor

Name: Muestra los detalles de la venta seleccionada en un formulario.  
↳ Sequence from «entity» Sistema to «entity» Sistema

Name: Sigue la lista de ventas.  
↳ Sequence from «entity» Sistema to BD\_Ventas

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Cerrar  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Actualiza los detalles y confirma  
⇒ Sequence from Administrador to «entity» Sistema

Name: Muestra mensaje de confirmación de la actualización.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Actualiza los detalles y confirma  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Confirma que los datos fueron actualizados.  
⇒ Sequence from BD\_Ventas to «entity» Sistema

Name: Cerrar  
⇒ Sequence from Administrador to «entity» Sistema

Name: Envía la lista de ventas.  
⇒ Sequence from BD\_Ventas to «entity» Sistema

Name: Accede al apartado para actualizar ventas.  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Muestra los detalles de la venta seleccionada en un formulario.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Selecciona una venta para actualizar (doble clic).  
⇒ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.  
⇒ Sequence from Administrador to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra formulario para búsqueda.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra los resultados de la búsqueda.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Ok  
⇐ Sequence from «entity» Sistema to Administrador

Name: Realiza la búsqueda con los criterios proporcionados.  
⇐ Sequence from «entity» Sistema to BD\_Ventas

Name: Ok  
⇐ Sequence from «entity» Sistema to Vendedor

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Muestra formulario para búsqueda.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Accede al apartado para consultar ventas.  
⇒ Sequence from Administrador to «entity» Sistema

Name: Ingresa criterios de búsqueda  
⇒ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para consultar ventas.  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Envía los resultados de la búsqueda.  
⇒ Sequence from BD\_Ventas to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Muestra los resultados de la búsqueda.  
⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Cerrar apartado  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda  
⇒ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado  
⇒ Sequence from Administrador to «entity» Sistema

## Sistema

Sequence «entity» in package 'Diagrama de secuencias'

Sistema

Version 1.0 Phase 1.0 Proposed

Jennifer2002 created on 30/7/2024. Last modified 30/7/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Muestra la tabla de productos con la opción de búsqueda.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Consulta todos los productos disponibles.  
⇐ Sequence from «entity» Sistema to BD\_Inventario

Name: OK  
⇐ Sequence from «entity» Sistema to Vendedor

Name: Actualiza el stock en base a la cantidad comprada.  
⇐ Sequence from «entity» Sistema to BD\_Inventario

Name: Añade producto al carrito,actualiza tabla y calcula el precio.  
⇐ Sequence from «entity» Sistema to «entity» Sistema

Name: Registra la venta  
⇐ Sequence from «entity» Sistema to BD\_Ventas

Name: Ok  
⇐ Sequence from «entity» Sistema to Administrador

## OUTGOING BEHAVIORAL RELATIONSHIPS

### INCOMING BEHAVIORAL RELATIONSHIPS

Name: Busca un producto en la tabla y selecciona uno (doble clic).

⇒ Sequence from Vendedor to «entity» Sistema

Name: Confirma que la venta fue registrada

⇒ Sequence from BD\_Ventas to «entity» Sistema

Name: Accede al apartado para registrar una venta.

⇒ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para registrar una venta.

⇒ Sequence from Vendedor to «entity» Sistema

Name: Ingresa la cantidad deseada

⇒ Sequence from Administrador to «entity» Sistema

Name: Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Envía la lista de productos.

⇒ Sequence from BD\_Inventario to «entity» Sistema

Name: Muestra la tabla de productos con la opción de búsqueda.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

⇒ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

⇒ Sequence from Vendedor to «entity» Sistema

Name: Ingresa la cantidad deseada

⇒ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

⇒ Sequence from Administrador to «entity» Sistema

Name: Añade producto al carrito,actualiza tabla y calcula el precio.

⇒ Sequence from «entity» Sistema to «entity» Sistema

Vendedor

Actor in package 'Use Case Model'

Vendedor

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

#### ELEMENTS OWNED BY VENDEDOR

- Administrator : Actor

#### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para registrar una venta.

↳ Sequence from Vendedor to «entity» Sistema

Name: Confirma la eliminación.

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Vendedor to «entity» Sistema

Name: Actualiza los detalles y confirma

↳ Sequence from Vendedor to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para eliminar ventas.

↳ Sequence from Vendedor to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para consultar productos.

↳ Sequence from Vendedor to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar

↳ Sequence from Vendedor to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Vendedor to «entity» Sistema

Name: Ingresa criterios de búsqueda

↳ Sequence from Vendedor to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

Name: OK

⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

Name: Oñ

⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

Name: Ok

⇒ Sequence from «entity» Sistema to Vendedor

## CONNECTORS

↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public

To:       Gestion venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination

From:    Vendedor : Actor, Public

To:       Gestionar venta : UseCase, Public

## Administrador

Actor owned by 'Vendedor', in package 'Use Case Model'

Administrador

OUTGOING BEHAVIORAL RELATIONSHIPS
Name: Accede al apartado para eliminar productos. ↳ Sequence from Administrador to «entity» Sistema
Name: Confirma la eliminación. ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para ingresar nuevos productos. ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para registrar una venta. ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para eliminar ventas. ↳ Sequence from Administrador to «entity» Sistema
Name: Ingresa la cantidad deseada ↳ Sequence from Administrador to «entity» Sistema
Name: Actualiza los detalles y confirma ↳ Sequence from Administrador to «entity» Sistema
Name: Accede al apartado para consultar ventas. ↳ Sequence from Administrador to «entity» Sistema
Name: Ingresa datos del producto y confirma. ↳ Sequence from Administrador to «entity» Sistema
Name: Confirma la eliminación. ↳ Sequence from Administrador to «entity» Sistema
Name: Ingresa criterios de búsqueda ↳ Sequence from Administrador to «entity» Sistema
Name: Cerrar ↳ Sequence from Administrador to «entity» Sistema
Name: elecciona un producto para actualizar (doble clic). ↳ Sequence from Administrador to «entity» Sistema
Name: Cerrar ↳ Sequence from Administrador to «entity» Sistema
Name: Cerrar

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar

↳ Sequence from Administrador to «entity» Sistema

Name: Busca un producto en la tabla y selecciona uno (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Ingresa la cantidad deseada

↳ Sequence from Administrador to «entity» Sistema

Name: Finaliza la compra y da clic en "Generar Venta".

↳ Sequence from Administrador to «entity» Sistema

Name: Actualiza los detalles del producto y confirma.

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar productos.

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona un producto para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para actualizar (doble clic).

↳ Sequence from Administrador to «entity» Sistema

Name: Selecciona una venta para eliminar (click derecho y opción eliminar).

↳ Sequence from Administrador to «entity» Sistema

Name: Cerrar apartado

↳ Sequence from Administrador to «entity» Sistema

Name: Accede al apartado para actualizar ventas.

↳ Sequence from Administrador to «entity» Sistema

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

Name: Ok

⇒ Sequence from «entity» Sistema to Administrador

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

Name: Ok  
⇒ Sequence from «entity» Sistema to Administrador

## CONNECTORS

↗ **UseCaseLink** Source -> Destination  
From: Administrador : Actor, Public  
To: Gestionar venta : UseCase, Public

↗ **UseCaseLink** Source -> Destination  
From: Administrador : Actor, Public  
To: Gestionar inventario : UseCase, Public

### UseCase

UseCase in package 'Use Case Model'

UseCase  
Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 11/7/2024

### Gestionar inio de cuenta

UseCase in package 'Use Case Model'

Gestionar inio de cuenta  
Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Include** «include» Source -> Destination  
From: Gestionar venta : UseCase, Public  
To: Gestionar inio de cuenta : UseCase, Public

↗ **Include** «include» Source -> Destination  
From: Gestionar inventario : UseCase, Public  
To: Gestionar inio de cuenta : UseCase, Public

### Gestionar inventario

UseCase in package 'Use Case Model'

Gestionar inventario

Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 11/7/2024

CONNECTORS
<p>↗ <b>Include</b> «include»    Source -&gt; Destination From:    Gestionar inventario : UseCase, Public To:       Gestionar inicio de cuenta : UseCase, Public</p>
<p>↗ <b>UseCaseLink</b>    Source -&gt; Destination From:    Administrador : Actor, Public To:       Gestionar inventario : UseCase, Public</p>

**Gestionar inventario diagram**  
Use Case diagram in package 'Use Case Model'

Gestionar inventario  
Version 1.0  
ASUS created on 11/7/2024. Last modified 11/7/2024

Figure 43: Gestionar inventario

**Gestionar venta**  
UseCase in package 'Use Case Model'

Gestionar venta  
Version 1.0 Phase 1.0 Proposed  
ASUS created on 11/7/2024. Last modified 11/7/2024

ELEMENTS OWNED BY Gestionar venta
█ Administrador : Actor
█ Actualizar venta : UseCase
█ Agregar venta : UseCase
█ Bucar precio del producto por ID : UseCase
█ Consultar venta : UseCase

## ELEMENTS OWNED BY Gestionar venta

- █ Eliminar venta : UseCase
- █ Generar recibo : UseCase
- █ Gestion venta : UseCase
- █ Verficar que el id de laventa exista : UseCase

## CONNECTORS

↗ **Include** «include»    Source -> Destination  
From:    Gestionar venta : UseCase, Public  
To:       Gestionar inio de cuenta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Administrador : Actor, Public  
To:       Gestionar venta : UseCase, Public

↗ **UseCaseLink**    Source -> Destination  
From:    Vendedor : Actor, Public  
To:       Gestionar venta : UseCase, Public

## Gestionar venta diagram

Use Case diagram in package 'Use Case Model'

Gestionar venta

Version 1.0

ASUS created on 11/7/2024. Last modified 11/7/2024

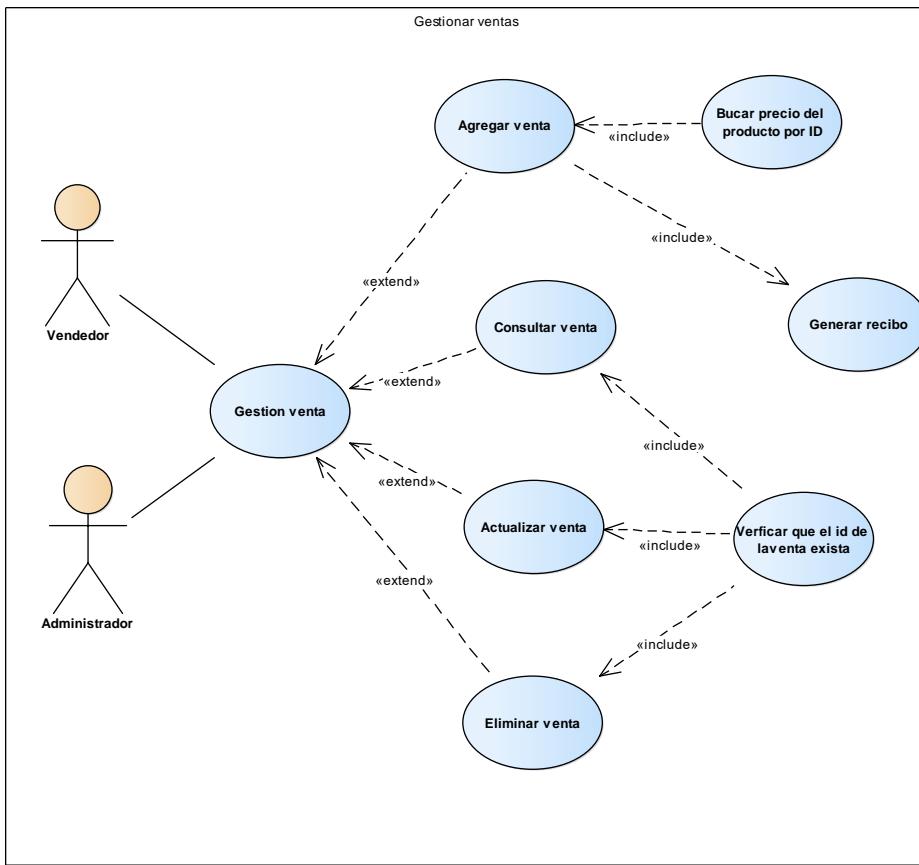


Figure 44: Gestionar venta

## Administrador

Actor owned by 'Gestionar venta', in package 'Use Case Model'

Administrador

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

### CONNECTORS

**UseCaseLink**    Source -> Destination  
 From:    Administrador : Actor, Public  
 To:        Gestion venta : UseCase, Public

## Actualizar venta

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Actualizar venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Extend** «extend» Source -> Destination

From: Actualizar venta : UseCase, Public  
To: Gestión venta : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Verificar que el id de laventa exista : UseCase, Public  
To: Actualizar venta : UseCase, Public

## Agregar venta

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Agregar venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Extend** «extend» Source -> Destination

From: Agregar venta : UseCase, Public  
To: Gestión venta : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Agregar venta : UseCase, Public  
To: Generar recibo : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Bucar precio del producto por ID : UseCase, Public  
To: Agregar venta : UseCase, Public

## Bucar precio del producto por ID

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Bucar precio del producto por ID

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Include** «include» Source -> Destination

From: Bucar precio del producto por ID : UseCase, Public  
To: Agregar venta : UseCase, Public

## Consultar venta

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Consultar venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Extend** «extend» Source -> Destination

From: Consultar venta : UseCase, Public  
To: Gestion venta : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Verificar que el id de laventa exista : UseCase, Public  
To: Consultar venta : UseCase, Public

## Eliminar venta

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Eliminar venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Extend** «extend» Source -> Destination

From: Eliminar venta : UseCase, Public  
To: Gestion venta : UseCase, Public

↗ **Include** «include» Source -> Destination

From: Verificar que el id de laventa exista : UseCase, Public  
To: Eliminar venta : UseCase, Public

## Generar recibo

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Generar recibo

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

↗ **Include** «include» Source -> Destination

From: Agregar venta : UseCase, Public  
To: Generar recibo : UseCase, Public

## Gestion venta

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Gestion venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

### ↗ UseCaseLink    Source -> Destination

From: Vendedor : Actor, Public

To: Gestion venta : UseCase, Public

### ↗ Extend «extend»    Source -> Destination

From: Consultar venta : UseCase, Public

To: Gestion venta : UseCase, Public

### ↗ Extend «extend»    Source -> Destination

From: Agregar venta : UseCase, Public

To: Gestion venta : UseCase, Public

### ↗ UseCaseLink    Source -> Destination

From: Administrador : Actor, Public

To: Gestion venta : UseCase, Public

### ↗ Extend «extend»    Source -> Destination

From: Actualizar venta : UseCase, Public

To: Gestion venta : UseCase, Public

### ↗ Extend «extend»    Source -> Destination

From: Eliminar venta : UseCase, Public

To: Gestion venta : UseCase, Public

## Verificar que el id de laventa exista

UseCase owned by 'Gestionar venta', in package 'Use Case Model'

Verificar que el id de laventa exista

Version 1.0 Phase 1.0 Proposed

ASUS created on 11/7/2024. Last modified 11/7/2024

## CONNECTORS

### ↗ Include «include»    Source -> Destination

From: Verificar que el id de laventa exista : UseCase, Public

To: Actualizar venta : UseCase, Public

### ↗ Include «include»    Source -> Destination

From: Verificar que el id de laventa exista : UseCase, Public

To: Eliminar venta : UseCase, Public

## CONNECTORS

### ↗ Include «include» Source -> Destination

From: Verificar que el id de laventa exista : UseCase, Public  
To: Consultar venta : UseCase, Public

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

### ← Collaboration from Object to «entity» Sistema

[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]

### ← Collaboration from Object to «entity» Sistema

[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]

### ← Collaboration from Object to «entity» Sistema

[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]

### ← Collaboration from Object to «entity» Sistema

[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

### ← Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

### ⇒ Collaboration from «entity» Sistema to Object

[ Name is OK. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

### ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

#### OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa datos del producto y confirma.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para ingresar nuevos productos.. Direction is 'Source -> Destination'. ]

#### INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

#### ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

#### OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from Object to «entity» Sistema  
[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

⬅ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona un producto para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para eliminar productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to Object  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
[ Name is elecciona un producto para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Actualiza los detalles del producto y confirma.. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para actualizar productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

## Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

#### ⇒ Collaboration from «entity» Sistema to Object

[ Name is OK. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

#### ✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Object

Object in package 'Use Case Model'

## Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

#### ↳ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

- ⬅ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ➡ Collaboration from «entity» Sistema to Object  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

- ↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar productos.. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### Object

Object in package 'Use Case Model'

#### Object

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

↳ Collaboration from Object to «entity» Sistema  
[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to Object  
[ Name is Ok. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Actualiza el stock en base a la cantidad comprada.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Consulta todos los productos disponibles.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que el producto fue guardado.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Guarda los datos del nuevo producto.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que el producto fue eliminado.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Elimina el producto de la base de datos.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Sigue la lista de productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

↳ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

### ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## BD\_Inventario

Object in package 'Use Case Model'

BD\_Inventario  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 1/8/2024. Last modified 1/8/2024

### OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

↳ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

### INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Solicita la lista de productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

## BD\_Ventas

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Confirma que la venta fue registrada . Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Registra la venta . Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

## BD\_Ventas

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Confirma que la venta fue eliminada.. Direction is 'Source -> Destination'. ]

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Sigue la lista de ventas.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Elimina la venta de la base de datos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

## BD\_Ventas

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

## BD\_Ventas

Object in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Solicita la lista de ventas.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

Object1

Object «boundary» in package 'Use Case Model'

Object1

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

Object2

Object «control» in package 'Use Case Model'

Object2

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Actualiza el stock en base a la cantidad comprada.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to Object  
[ Name is OK. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Añade producto al carrito,actualiza tabla y calcula el precio.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra la tabla de productos con la opción de búsqueda.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to Object  
[ Name is OK. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Consulta todos los productos disponibles.. Direction is 'Source -> Destination'. ]
- ↳ Collaboration from «entity» Sistema to BD\_Ventas  
[ Name is Registra la venta . Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]
- ⇒ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Confirma que la venta fue registrada . Direction is 'Source -> Destination'. ]
- ⇒ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]
- ⇒ Collaboration from Object to «entity» Sistema  
[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]
- ⇒ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]
- ⇒ Collaboration from Object to «entity» Sistema  
[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from Object to «entity» Sistema

[ Name is Ingresa la cantidad deseada. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Busca un producto en la tabla y selecciona uno (doble clic).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Añade producto al carrito,actualiza tabla y calcula el precio.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra la tabla de productos con la opción de búsqueda.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Finaliza la compra y da clic en "Generar Venta".. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para registrar una venta.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_\_Inventario

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↖ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↖ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

← Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Guarda los datos del nuevo producto.. Direction is 'Source -> Destination'. ]

← Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación.. Direction is 'Source -> Destination'. ]

← Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra formulario para ingresar datos del producto.. Direction is 'Source -> Destination'. ]

← Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

→ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

→ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Confirma que el producto fue guardado.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa datos del producto y confirma.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra formulario para ingresar datos del producto.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para ingresar nuevos productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Solicita la lista de ventas.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Elimina la venta de la base de datos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que la venta fue eliminada.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

### ➡ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

### ➡ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para eliminar ventas.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

### ✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Elimina el producto de la base de datos.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Inventario  
[ Name is Solicita la lista de productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra mensaje de confirmación de eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra confirmación para eliminar.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Confirma que el producto fue eliminado.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Inventario to «entity» Sistema  
[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Selecciona un producto para eliminar (click derecho y opción eliminar).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Confirma la eliminación.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para eliminar productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

✓ Association (direction: Unspecified)

## ASSOCIATIONS

Source: Public (Object) Sistema «entity»      Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»      Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Object      Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»      Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»      Target: Public (Object) Sistema «entity»

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to Object

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

➡ Collaboration from BD\_Ventas to «entity» Sistema  
[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Cerrar apartado. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Accede al apartado para consultar ventas.. Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema  
[ Name is Ingresa criterios de búsqueda . Direction is 'Source -> Destination'. ]

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

## ASSOCIATIONS

Source: Public (Object) Sistema «entity»      Target: Public (Object) Sistema «entity»

✍ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

✍ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Realiza la búsqueda con los criterios proporcionados.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para consultar productos.. Direction is 'Source -> Destination'. ]

➡ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Muestra los resultados de la búsqueda.. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Ingresa criterios de búsqueda. Direction is 'Source -> Destination'. ]

➡ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

➡ Collaboration from «entity» Sistema to «entity» Sistema  
[ Name is Muestra formulario para búsqueda.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 1/8/2024. Last modified 1/8/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los detalles del producto en un formulario.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

⬅ Collaboration from «entity» Sistema to BD\_Inventario

[ Name is Solicita la lista de productos.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Envía la lista de productos.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los detalles del producto en un formulario.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is elecciona un producto para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Inventario to «entity» Sistema

[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Actualiza los detalles del producto y confirma.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para actualizar productos.. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Inventario

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## Sistema

Object «entity» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed

Fabian created on 31/7/2024. Last modified 31/7/2024

## OUTGOING STRUCTURAL RELATIONSHIPS

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Actualiza la información en la base de datos.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los detalles de la venta seleccionada en un formulario.. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to Object

[ Name is Ok. Direction is 'Source -> Destination'. ]

↳ Collaboration from «entity» Sistema to BD\_Ventas

[ Name is Solicita la lista de ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra mensaje de confirmación de la actualización.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Confirma que los datos fueron actualizados.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

## INCOMING STRUCTURAL RELATIONSHIPS

⇒ Collaboration from BD\_Ventas to «entity» Sistema

[ Name is Envía la lista de ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Accede al apartado para actualizar ventas.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Selecciona una venta para actualizar (doble clic).. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from «entity» Sistema to «entity» Sistema

[ Name is Muestra los detalles de la venta seleccionada en un formulario.. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Cerrar. Direction is 'Source -> Destination'. ]

⇒ Collaboration from Object to «entity» Sistema

[ Name is Actualiza los detalles y confirma. Direction is 'Source -> Destination'. ]

## ASSOCIATIONS

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) BD\_Ventas

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

✓ Association (direction: Unspecified)

Source: Public (Object) Sistema «entity»

Target: Public (Object) Sistema «entity»

## ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

↙ Association (direction: Unspecified)

Source: Public (Object) Object

Target: Public (Object) Sistema «entity»

Mensaje: No existe ningun producto disponible  
Activity in package 'Use Case Model'

Mensaje: No existe ningun producto disponible  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Mensaje: No existe ningun producto disponible to Ingresa a la interfaz

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: No

↗ ControlFlow from ¿Existen productos disponibles? to Mensaje: No existe ningun producto disponible

Administrador  
ActivityPartition in package 'Use Case Model'

Administrador  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 3/8/2024. Last modified 3/8/2024

## ELEMENTS OWNED BY Administrador

▣ Busca un producto en la tabla y selecciona uno (doble click) : Activity

▣ Finaliza la compra y da click en "Generar Venta" : Activity

▣ Ingresa la cantidad deseada : Activity

▣ Accede al apartado para registrar venta : ActivityInitial

**Busca un producto en la tabla y selecciona uno (doble click)**

Activity owned by 'Administrador', in package 'Use Case Model'

Busca un producto en la tabla y selecciona uno (doble click)

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

#### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Busca un producto en la tabla y selecciona uno (doble click) to Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Busca un producto en la tabla y selecciona uno (doble click)

### Finaliza la compra y da click en "Generar Venta"

Activity owned by 'Administrador', in package 'Use Case Model'

Finaliza la compra y da click en "Generar Venta"

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

#### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Finaliza la compra y da click en "Generar Venta" to Registra la venta

#### INCOMING BEHAVIORAL RELATIONSHIPS

Name: No

⇒ ControlFlow from ¿Solicita mas productos? to Finaliza la compra y da click en "Generar Venta"

### Ingresa la cantidad deseada

Activity owned by 'Administrador', in package 'Use Case Model'

Ingresa la cantidad deseada

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Ingresa la cantidad deseada

### Accede al apartado para registrar venta

ActivityInitial owned by 'Administrador', in package 'Use Case Model'

## OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Accede al apartado para registrar venta to Ingresa a la interfaz

### Agregar Venta

InteractionFragment in package 'Use Case Model'

Agregar Venta

Version 1.0 Phase 1.0 Proposed

ASUS created on 19/7/2024. Last modified 19/7/2024

### BD\_Inventario

ActivityPartition in package 'Use Case Model'

BD\_Inventario

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## ELEMENTS OWNED BY BD\_Inventario

▣ Actualiza el stock en base a la cantidad comprada : Activity

▣ Consulta todos los productos disponibles : Activity

### Actualiza el stock en base a la cantidad comprada

Activity owned by 'BD\_Inventario', in package 'Use Case Model'

Actualiza el stock en base a la cantidad comprada

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Actualiza el stock en base a la cantidad comprada

### Consulta todos los productos disponibles

Activity owned by 'BD\_Inventario', in package 'Use Case Model'

Consulta todos los productos disponibles

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## STRUCTURAL PART OF Consulta todos los productos disponibles

ObjectNode1 : ActionPin

## OUTGOING BEHAVIORAL RELATIONSHIPS

ControlFlow from Consulta todos los productos disponibles to ¿Existen productos disponibles?

## INCOMING BEHAVIORAL RELATIONSHIPS

ControlFlow from Ingresa a la interfaz to Consulta todos los productos disponibles

## ObjectNode1

ActionPin owned by 'Consulta todos los productos disponibles', in package 'Use Case Model'

### BD\_Ventas

ActivityPartition in package 'Use Case Model'

BD\_Ventas

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## ELEMENTS OWNED BY BD\_Ventas

Registra la venta : Activity

## Registra la venta

Activity owned by 'BD\_Ventas', in package 'Use Case Model'

Registra la venta

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

ControlFlow from Registra la venta to Confirma que la venta fue registrada

## INCOMING BEHAVIORAL RELATIONSHIPS

ControlFlow from Finaliza la compra y da click en "Generar Venta" to Registra la venta

Sistema  
ActivityPartition in package 'Use Case Model'

Sistema  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 3/8/2024. Last modified 3/8/2024

ELEMENTS OWNED BY Sistema	
■	Añade productos al carrito, actualiza tabla y calcula el precio : Activity
■	Envia la lista de productos : Activity
■	Ingresar a la interfaz : Activity
■	Muestra la tabla de productos con la opción de búsqueda : Activity
■	Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad : Activity
■	¿Existen productos disponibles? : DecisionNode
■	Confirma que la venta fue registrada : ActivityFinal

**Añade productos al carrito, actualiza tabla y calcula el precio**  
Activity owned by 'Sistema', in package 'Use Case Model'

Añade productos al carrito, actualiza tabla y calcula el precio  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 3/8/2024. Last modified 3/8/2024

INCOMING BEHAVIORAL RELATIONSHIPS	
⇒	ControlFlow from Synchronization to Añade productos al carrito, actualiza tabla y calcula el precio

**Envia la lista de productos**  
Activity owned by 'Sistema', in package 'Use Case Model'

Envia la lista de productos  
Version 1.0 Phase 1.0 Proposed  
Fabian created on 3/8/2024. Last modified 3/8/2024

OUTGOING BEHAVIORAL RELATIONSHIPS	
⇐	ControlFlow from Envía la lista de productos to Muestra la tabla de productos con la opción de búsqueda

## INCOMING BEHAVIORAL RELATIONSHIPS

Name: Si

⇒ ControlFlow from ¿Existen productos disponibles? to Envia la lista de productos

### Ingresa a la interfaz

Activity owned by 'Sistema', in package 'Use Case Model'

Ingresa a la interfaz

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Ingresa a la interfaz to Consulta todos los productos disponibles

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Accede al apartado para registrar venta to Ingresa a la interfaz

⇒ ControlFlow from Mensaje: No existe ningun producto disponible to Ingresa a la interfaz

### Muestra la tabla de productos con la opcion de busqueda

Activity owned by 'Sistema', in package 'Use Case Model'

Muestra la tabla de productos con la opcion de busqueda

Version 1.0 Phase 1.0 Proposed

Fabian created on 3/8/2024. Last modified 3/8/2024

## OUTGOING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Muestra la tabla de productos con la opcion de busqueda to Synchronization

## INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Envia la lista de productos to Muestra la tabla de productos con la opcion de busqueda

Name: Si

⇒ ControlFlow from Synchronization to Muestra la tabla de productos con la opcion de busqueda

### Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

Activity owned by 'Sistema', in package 'Use Case Model'

Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

Version 1.0 Phase 1.0 Proposed  
Fabian created on 3/8/2024. Last modified 3/8/2024

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Busca un producto en la tabla y selecciona uno (doble click) to Muestra ventana emergente con detalles del producto y campo para ingresar la cantidad

#### ¿Existen productos disponibles?

DecisionNode owned by 'Sistema', in package 'Use Case Model'

#### OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Si

↳ ControlFlow from ¿Existen productos disponibles? to Envia la lista de productos

Name: No

↳ ControlFlow from ¿Existen productos disponibles? to Mensaje: No existe ningun producto disponible

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Consulta todos los productos disponibles to ¿Existen productos disponibles?

#### Confirma que la venta fue registrada

ActivityFinal owned by 'Sistema', in package 'Use Case Model'

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Registra la venta to Confirma que la venta fue registrada

Sistema

Sequence «boundary» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed  
ASUS created on 19/7/2024. Last modified 19/7/2024

Sistema

Sequence «control» in package 'Use Case Model'

Sistema

Version 1.0 Phase 1.0 Proposed  
ASUS created on 19/7/2024. Last modified 19/7/2024

¿Solicita mas productos?

DecisionNode in package 'Use Case Model'

#### OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from ¿Solicita mas productos? to Synchronization

Name: No

↳ ControlFlow from ¿Solicita mas productos? to Finaliza la compra y da click en "Generar Venta"

#### INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to ¿Solicita mas productos?