



Software Requirements Specification [SRS]

(IEEE Std. 830-1998)

INTEGRANTES DEL GRUPO:

ALBERTO ALEJANDRO RIVAS FERNÁNDEZ
CARLOS GÓMEZ LÓPEZ
CARLOS FORRIOL MOLINA
JAVIER DE HOYOS PINO
LUIS ENRIQUE BARRERO PEÑA
MARIO GONZÁLEZ DE SANTOS
MIGUEL MARCOS ASENJO
PABLO GAMO GONZÁLEZ

1. Introducción	2
1.1 Propósito	2
1.2 Alcance	2
1.3 Definiciones, acrónimos y abreviaturas	3
1.4 Referencias.....	4
1.5 Resumen.....	4
2. Descripción general.....	5
2.1 Perspectiva del producto.....	5
2.2 Funciones del producto	5
2.2.1 Producto.....	5
2.2.2 Sección.....	5
2.2.3 Trabajadores.....	6
2.2.4 Clientes.....	6
2.2.5 Compras.....	6
2.2.6 Proveedor	6
2.2.7 Marca.....	7
2.3 Características del usuario.....	7
2.4 Restricciones	7
2.4.1 Restricción principal	7
2.4.2 Restricciones de tiempo.....	7
2.4.3 Requisitos de lenguajes de alto nivel	7
2.4.4 Requisitos de Bases de datos	8
2.4.5 Restricciones del proyecto.....	8
2.4.5 Funciones de auditoría	8
2.4.5 Requisitos de Fiabilidad.....	8
2.5 Supuestos y dependencias	8
3. Requisitos Específicos	9
3.1 Interfaces Externas	9
3.2 Funciones.....	11
3.2.1 Módulo de Productos	11
3.2.2 Módulo de Secciones	18
3.2.3 Módulo de Trabajadores.....	23
3.2.4 Módulo de Clientes.....	29
3.2.5 Módulo de Compras.....	34
3.2.6 Módulo de Proveedores.....	38
3.2.7 Módulo de Marcas.....	43
3.3 Requisitos de Rendimiento.....	48
3.4 Logical Database Requirements (Modelo del Dominio)	48
3.5 Restricciones de Diseño	48
3.6 Atributos del Sistema Software	48

1. Introducción

El programa a desarrollar será un software de gestión de auto compra de supermercados.

A través de este software, el cliente puede hacer su compra en la tienda física sin necesidad de que le atienda un cajero. A la hora de comprar, el cliente debe iniciar sesión introduciendo su DNI para poder acceder a la interfaz de auto compra.

El software proporciona un control sobre todas las compras que se realicen, ya sean efectuadas por los trabajadores o por los clientes mediante auto compra. También controlara los productos, las marcas de estos y los proveedores que las distribuyen.

1.1 Propósito

Este documento pretende proporcionar información sobre los límites y requerimientos del producto, junto a la información necesaria para el entendimiento del mismo.

Está dirigido a los empleados del supermercado para un óptimo control de su negocio.

El propósito de este documento es hacer la especificación de requisitos (SRS) siguiendo el estándar IEEE Std.830-1998 para que nuestro sistema permita el correcto funcionamiento de un gestor de auto compra de un supermercado

1.2 Alcance

El proyecto “Supermercados” ha sido creado con el propósito de facilitar la gestión de un supermercado, pero es posible que se ejecute en otros posibles negocios de la misma naturaleza. Gracias a este producto y en especial a su interfaz sencilla, cualquier persona, sin estar capacitada en este ámbito, será capaz de utilizar esta herramienta.

El principal objetivo de nuestro sistema es proporcionar al cliente una forma rápida de realizar su compra sin la necesidad de que le atienda un cajero, mediante una BBDD que contenga la información de los clientes, proveedores, marcas, productos, empleados. El programa podrá ser utilizado por el administrador que, entre otras cosas, podrá gestionar los clientes, las secciones, los productos y sus marcas y sus respectivos proveedores.

1.3 Definiciones, acrónimos y abreviaturas

SRS: Software Requirements Specification, es una descripción completa del comportamiento del sistema que se va a desarrollar.

BBDD: Base de datos, es un conjunto de datos pertenecientes a un mismo contexto que son almacenados conjuntamente para su posterior uso.

IEEE (Institute of Electrical and Electronics Engineers): El estándar que se utiliza en este documento es el Std. 830-1998 del IEEE.

S.O: Sistema operativo.

Cliente: Organización que ha contratado nuestros servicios para producir este software.

Usuario: Toda persona que va a interactuar con el software.

UCM: Universidad Complutense de Madrid.

IBM (International Business Machines) RSAD (Rational Software Architect Designer): Herramienta Case utilizada para desarrollar parte de los requisitos de la SRS.

Diagrama de actividades: Es un diagrama que muestra un flujo de actividades, está formado por nodos y arcos. Los utilizamos para describir las acciones de las funciones.

Herramienta CASE: Son herramientas que facilitan el almacenamiento de requisitos, la gestión del cambio y la gestión de la traza.

SCV (Sistema de Control de Versiones): Es un programa que ayuda a crear y mantener las diversas versiones de un conjunto de ficheros.

RDBMS (Sistema de gestión de bases de datos relacionales): es una base de datos que almacena datos de forma estructurada.

MySQL: Sistema de gestión de bases de datos relacional, es una de las más populares, sobre todo para entornos de desarrollo web.

Backup: Consiste en una copia de seguridad de los datos originales que se realiza con el fin de disponer de un medio para recuperarlos en caso de su pérdida.

Revisiones técnicas formales (RTFs): El objetivo básico de las RTFs es detectar errores antes de que se conviertan en defectos.

Elementos de configuración software (ECS): Son los resultados de cada tarea de trabajo.

JPA: Es el encargado de convertir los objetos Java en instrucciones para el Manejador de Base de Datos (MDB).

Marca: Signo, que permite diferenciar y hacer únicos los productos o servicios de una empresa.

Compra: Acción de comprar unos productos por un precio.

Trabajador: Persona que realiza un trabajo a cambio de un salario.

Producto: Cosa producida natural o artificialmente, o resultado de un trabajo u operación.

Proveedor: Que se dedica a proveer o abastecer de productos necesarios a una persona o empresa.

Perecedero: Que tiene duración limitada, está destinado a perecer, perder su utilidad o validez, o estropearse en un determinado plazo de tiempo.

Sección: Parte que junto con otras constituye una empresa, un establecimiento comercial, un organismo, una agrupación, etc.

1.4 Referencias.

Apuntes del profesor ANTONIO NAVARRO MARTIN del Campus Virtual de la UCM:
https://cvmdp.ucm.es/moodle/pluginfile.php/687418/mod_resource/content/1/6.%20Ing%20enier%C3%ADa%20de%20requisitos.pdf

Wikipedia:

-https://es.wikipedia.org/wiki/Base_de_datos

-https://es.wikipedia.org/wiki/Especificaci%C3%B3n_de_requisitos_de_software

1.5 Resumen.

Este documento sigue el estándar 830-1998 y está dividido en tres apartados:

- El primer apartado, introducción de la SRS consiste en una descripción breve del producto que vamos a desarrollar, nuestros propósitos y alcance y una pequeña guía para entender los próximos apartados.
- El segundo apartado es la descripción general. En este apartado se describen los factores generales que afectan al producto y sus requisitos. Es una descripción breve y no define requisitos específicos ya que se realiza de manera más comprensible.
- Por último, se representan los requisitos específicos los cuales contienen los requisitos del sistema de forma extensa, pero sin dejar de lado la legibilidad. Este punto referencia a los anteriores.

2.Descripción general

2.1 Perspectiva del producto

La plataforma constará del software que se desarrolle para poder ser utilizada por el supermercado y fomentar una mayor actividad comercial de auto compra o autoservicio.

Se realizará habitualmente y de forma automática, copias de seguridad o backups en la propia base de datos para evitar riesgos en caso de pérdida de información o caída del sistema y así poder restaurarla posteriormente.

2.2 Funciones del producto

Las funciones del producto son:

2.2.1 Producto

Añadir un nuevo producto: Crea un nuevo producto, con los datos introducidos, en la base de datos.

Vincular producto a proveedor: Vincula un producto a un proveedor específico.

Buscar datos de todos los productos: Muestra los datos de todos los productos.

Buscar datos de un solo producto: Busca y muestra los datos del producto solicitado.

Actualizar/modificar stock e información de un producto: Actualizar o modificar el stock/información de un producto en la base de datos.

Eliminar un producto: Elimina el producto solicitado de la base de datos.

Desvincular producto de proveedor: Desvincula un producto de un proveedor específico.

2.2.2 Sección

Crear nueva sección: Crea una nueva sección, con los datos introducidos, en la base de datos.

Buscar información de las secciones: Muestra los datos de todas las secciones.

Buscar información de una sección: Busca y muestra los datos de la sección solicitada.

Editar datos de sección: Modificar los datos de la sección en la base de datos.

Eliminar una sección: Elimina la sección introducida de la base de datos.

2.2.3 Trabajadores

Dar de alta a un nuevo: Añade un nuevo empleado a la base de datos.

Buscar datos de un trabajador concreto: Busca un empleado y muestra sus datos.

Buscar datos de todos los trabajadores: Muestra los datos de todos los empleados

Actualizar/modificar datos de un trabajador: Modifica los datos de un empleado en la base de datos.

Dar de baja a un trabajador: Introduce un empleado y lo elimina de la base de datos.

Identificar al trabajador: Identifica al trabajador en el sistema para que pueda realizar las respectivas acciones.

2.2.4 Clientes

Dar de alta a un nuevo cliente: Añade un nuevo cliente a la base de datos.

Buscar información general de los clientes: Muestra los datos de todos los clientes.

Buscar datos específicos de un cliente: Busca un cliente y muestra sus datos.

Editar información del cliente: Modifica los datos del cliente solicitado en la base de datos.

Dar de baja al cliente registrado: Introduce un cliente y lo elimina de la base de datos.

2.2.5 Compras

Mostrar mensajes de ayuda en la compra: Muestra un texto indicativo con los pasos necesarios para realizar la compra.

Añadir producto a la lista: lee el ID del producto y se añade al carrito.

Eliminar producto de la lista: lee el ID del producto que se seleccione y se elimina del carrito.

Pagar compra: Se suma el precio de todos los productos del carrito y se procede al pago bancario.

2.2.6 Proveedor

Añadir datos de un nuevo proveedor: Registra y añade un nuevo proveedor, con los datos introducidos en la base de datos.

Búsqueda de proveedores: Busca y muestra los datos del proveedor deseado.

Editar información de un proveedor: Edita la información del proveedor que se desee.

Dar datos de todos los proveedores: Lista los datos de todos los proveedores.

Dar de baja un proveedor antiguo: Elimina de forma lógica a un proveedor de la base de datos.

2.2.7 Marca

Añadir nuevas marcas: Registra una nueva marca, con los datos introducidos, en la base de datos.

Buscar información de una marca: Busca y muestra los datos de la marca solicitada.

Editar datos específicos de una marca: Permite editar los datos de una marca-

Buscar información de las marcas en general: Muestra los datos de todas las marcas.

Eliminar una marca que ya no se comercializa o que no distribuimos: Elimina la marca solicitada de la base de datos.

2.3 Características del usuario

El software está dirigido a cualquier empleado sin necesidad de tener conocimientos u experiencia en programas técnicos o complejos.

Los únicos requisitos para la correcta utilización del software es un nivel básico de español, ya que la interfaz no tendrá capacidad de cambiar de idioma, y conocimientos a nivel usuario de informática.

El uso en su totalidad solo será posible mediante la identificación con contraseña, en concreto, para el control de datos sensibles como los clientes.

2.4 Restricciones

2.4.1 Restricción principal

Se van a utilizar técnicas de Ingeniería del software a nivel de diseño y gestión de este software para lograr un producto de calidad y cumplir con los requisitos.

2.4.2 Restricciones de tiempo

El proyecto constara de dos entregas:

- La primera entrega se realizará el día 22/01/2021 y se entregará el plan de proyecto software y la especificación de requisitos software del producto.
- La segunda entrega se realizará el día dd/mm/aa y constará del producto final con todos su datos y documentos.

2.4.3 Requisitos de lenguajes de alto nivel

El diseño se realizará con el lenguaje UML 2.x el cual desarrollaremos desde la herramienta CASE IBS RSAD.

En cuanto al lenguaje que se usará para la codificación será Java y la aplicación será de escritorio.

Utilizaremos el SCV de la Facultad de Informática (SVN) para gestionar tanto el código como la documentación.

2.4.4 Requisitos de Bases de datos

En cuanto a la base de datos utilizaremos un formato de texto o un formato relacional. En caso de que sea relacional utilizaremos MySQL.

2.4.5 Restricciones del proyecto

- El proyecto va a realizarse en equipo, el cual constará de 8 personas exactamente.
- Se utilizará el sistema de control de versiones (SCV) de la Facultad de Informática para gestionar la documentación y entrega.
- La entrega debe hacerse como tag en el repositorio del SCV de la facultad.
- La entrega debe incluir el archivo sobre la planificación temporal en formato MS Project y los archivos correspondientes a los diagramas de casos de uso y de actividades UML en formato IBM Rational Software Architect 9.6.
- Los alumnos son los únicos responsables del contenido almacenado en el SCV. La presencia de contenido inadecuado en el SCV conllevará su borrado y la invalidación del proyecto.

2.4.5 Funciones de auditoría

En este proyecto vamos a realizar revisiones técnicas formales (RTFs) para asegurarnos de poder encontrar y solucionar los fallos que encontremos en los elementos de configuración software (ECS) con el fin de lograr un producto de calidad.

2.4.5 Requisitos de Fiabilidad


Vamos a garantizar la fiabilidad en la autenticación de los usuarios y la posibilidad de dar marcha atrás en la definición del perfil de cada usuario, así como el seguro almacenamiento de los datos en nuestra base de datos en caso de cualquier fallo del sistema o pérdida de conexión.

2.5 Supuestos y dependencias

- El software está creado para utilizarlo únicamente en ordenadores cuyo sistema operativo sea Windows 7, Windows 8 o Windows 10. Este software no está disponible para macOS.
- Los módulos utilizados por el software utilizarán una de las marcas más populares de software RDBMS (MySQL) ya que implementa un modelo cliente-servidor, útil para nuestro software.


3. Requisitos Específicos

3.1 Interfaces Externas




Introduce tu N.º de Socio

INICIAR SESIÓN

 Ayuda



REGISTRARSE





Nombre

E-Mail

DNI

REGISTRARSE

 Ayuda



TOTAL: 28,08 €



*Si la compra supera los 50€, se le otorgará un descuento del 5% para su próxima cuenta

	N.º Stock	Precio
<input checked="" type="checkbox"/> Patatas Lays	1	1,50€
<input checked="" type="checkbox"/> Ketchup Heinz	2	6,10€
<input checked="" type="checkbox"/> Chocolate Lindt	4	7,88€
<input checked="" type="checkbox"/> Naranjas 250g	1	2,20€
<input checked="" type="checkbox"/> Pack Leche Asturiana	1	4,00€
<input checked="" type="checkbox"/> Galletas María	4	6,40€
<input type="checkbox"/> Pack 4 Yogur de Plátano	1	1,80€



Ayuda



FINALIZAR COMPRA

	N.º Stock	Precio
Patatas Lays	1	1,50€
Ketchup Heinz	2	6,10€
Chocolate Lindt	4	7,88€
Naranjas 250g	1	2,20€
Pack Leche Asturiana	1	4,00€
Galletas María	4	6,40€

TOTAL	51,28€
DESCUENTO	5%
TOTAL A PAGAR	48,72€



Ayuda

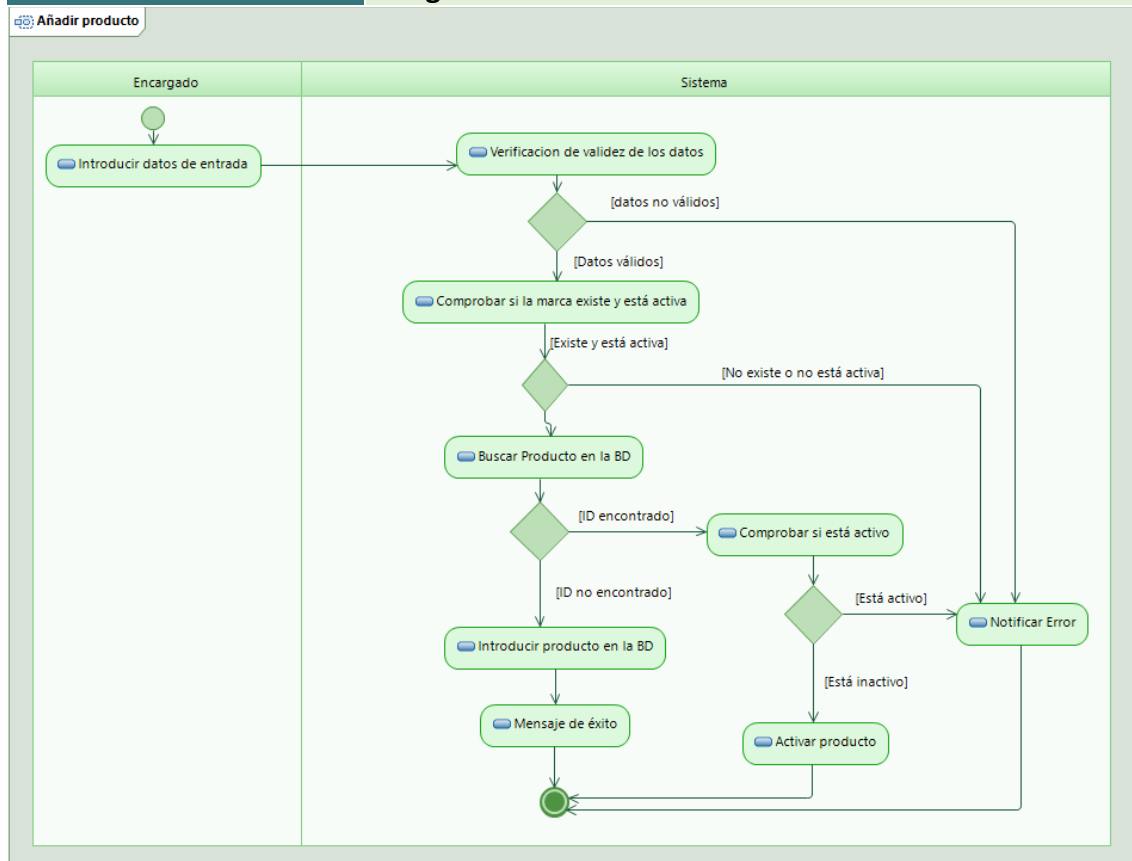


PAGAR

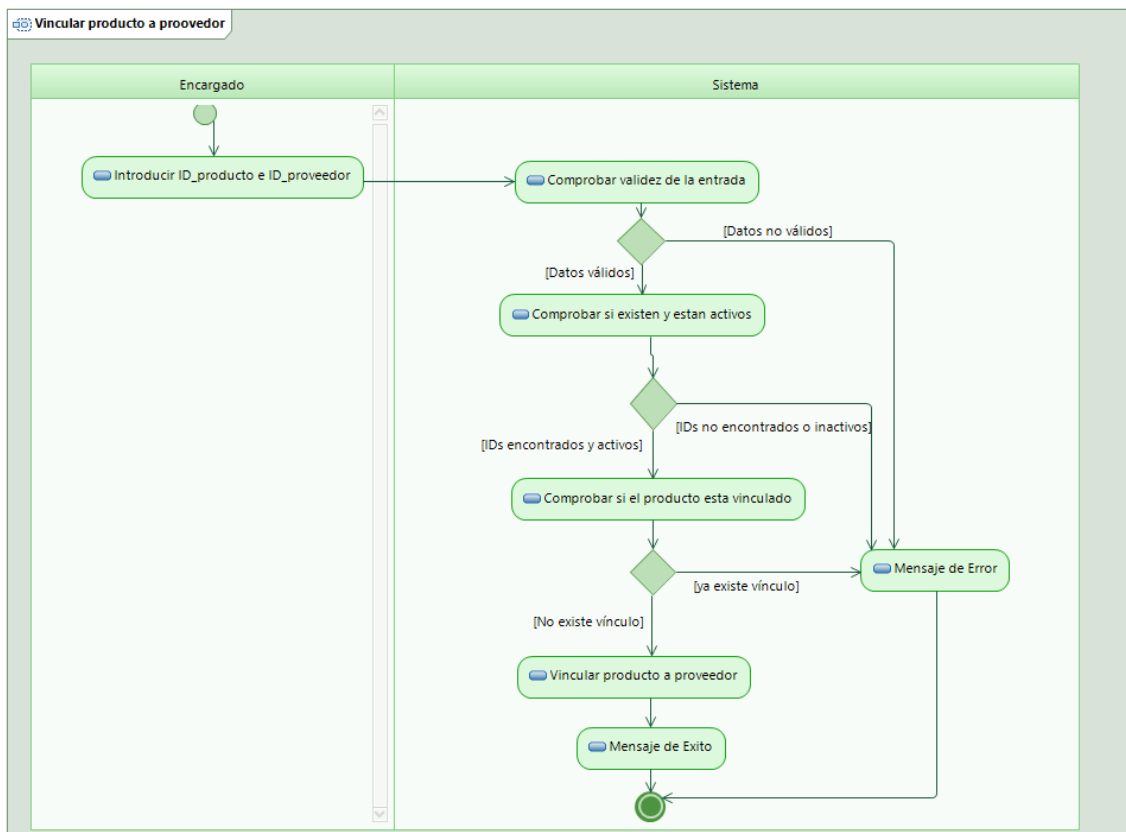
3.2 Funciones

3.2.1 Módulo de Productos

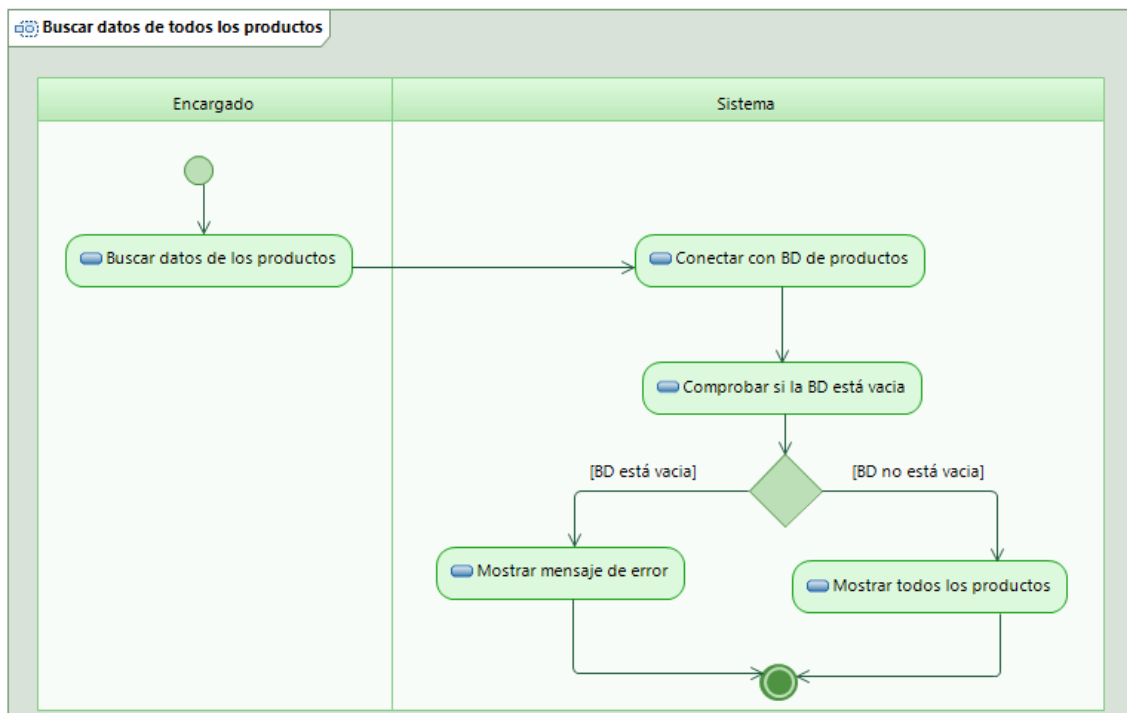
Función 3.2.1.1 Añadir un nuevo producto	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Añade un producto a la BD de productos.
Entrada	Nombre, stock, precio e ID_Marca.
Salida	Muestra ID_Producto y su nombre.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de productos.
Acción	Comprobar que los datos introducidos son sintácticamente correctos y que su marca exista y este activa, que no haya ningún producto con el mismo nombre y marca. En caso afirmativo añadir el producto y devolver ID_Producto.
Precondición	No hay productos con el mismo nombre, ni identificadores repetidos en la BD.
Postcondición	No hay productos con el mismo nombre, ni identificadores repetidos en la BD; si éxito, nuevo producto se añadirá.
Efectos laterales	Ninguno.



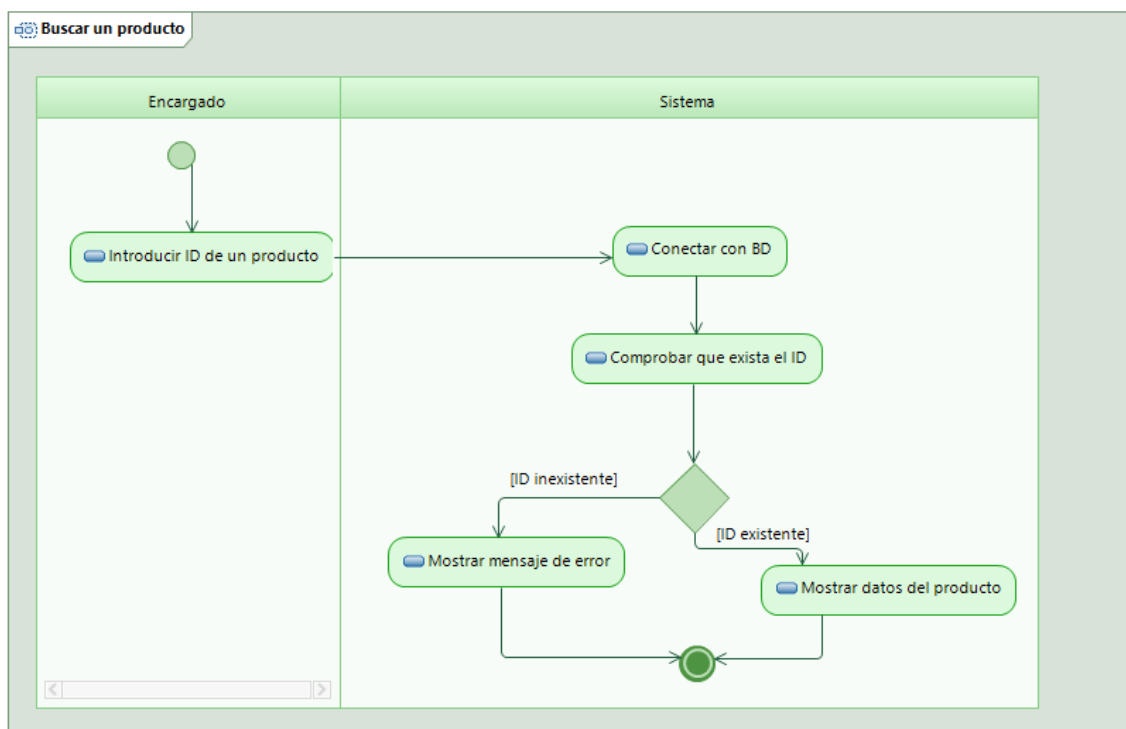
Función 3.2.1.2 Vincular Producto a Proveedor	
Prioridad	Media.
Estabilidad	Media.
Descripción	Vincula un producto a un proveedor específico.
Entrada	ID_Producto e Id_Proveedor.
Salida	Muestra ID_Producto, su nombre y su proveedor asignado con su ID.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de productos.
Acción	Comprobar que los IDs sean correctos, que existan y estén activos, que el producto no esté vinculado a otro proveedor y si se cumple, se asignara el producto a un proveedor en concreto.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si éxito, se vinculará el producto a un proveedor.
Efectos laterales	Ninguno.



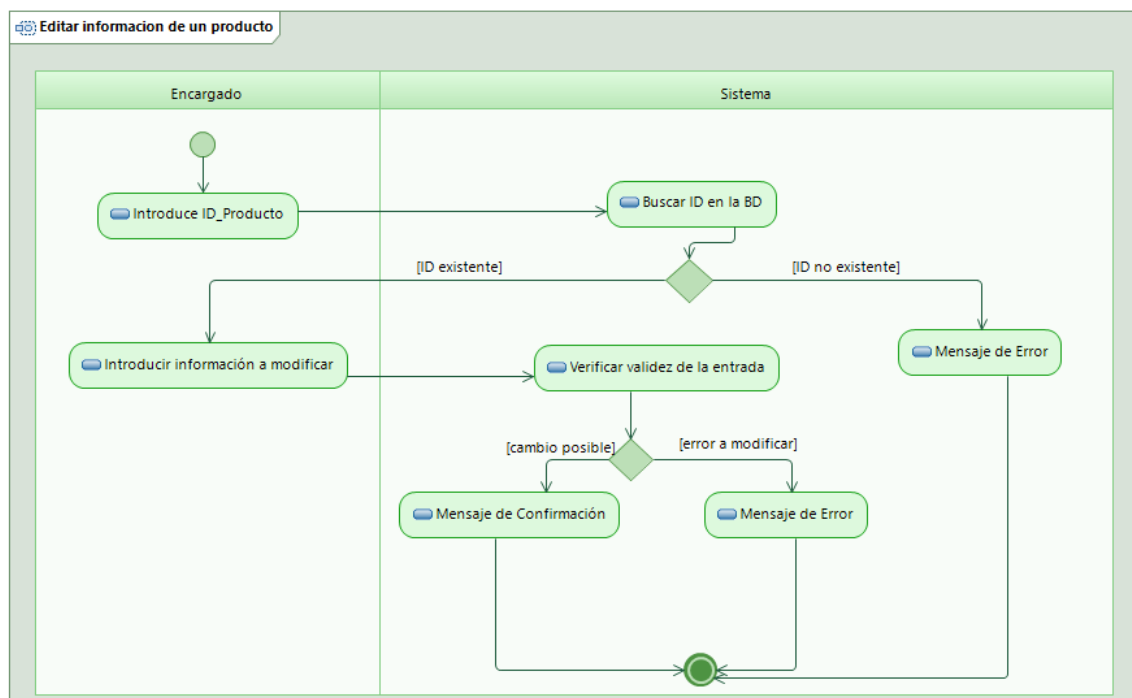
Función 3.2.1.3 Buscar datos de todos los productos	
Prioridad	Media.
Estabilidad	Alta.
Descripción	Busca y muestra los datos de todos los productos.
Entrada	Ninguna.
Salida	Todos los atributos del producto.
Origen	Interfaz Usuario - Empleado
Destino	Sistema.
Necesita	Base de datos de productos.
Acción	Comprueba todos los productos que existen para mostrar los atributos, en caso de que no exista ninguno, muestra mensaje de error.
Precondición	Que la BD no esté vacía.
Postcondición	Que la BD no esté vacía; si tiene éxito, se imprimirán los datos por pantalla.
Efectos laterales	Ninguno.



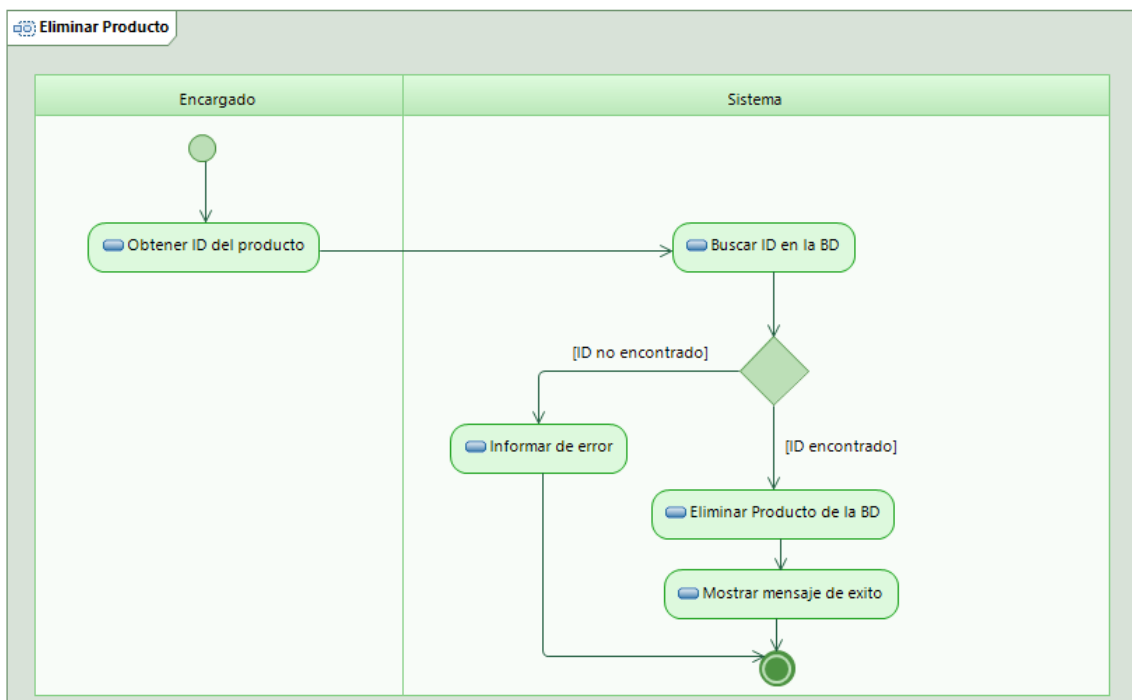
Función 3.2.1.4 Buscar datos de un solo producto	
Prioridad	Media.
Estabilidad	Alta.
Descripción	Busca y muestra los datos del producto indicado.
Entrada	ID_Producto.
Salida	Todos los atributos del producto.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de productos.
Acción	Comprueba si existe el producto que se quiere buscar. En caso de existir muestra todos sus atributos, si no, muestra mensaje de error.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si éxito, imprimirá los datos por pantalla.
Efectos laterales	Ninguno.



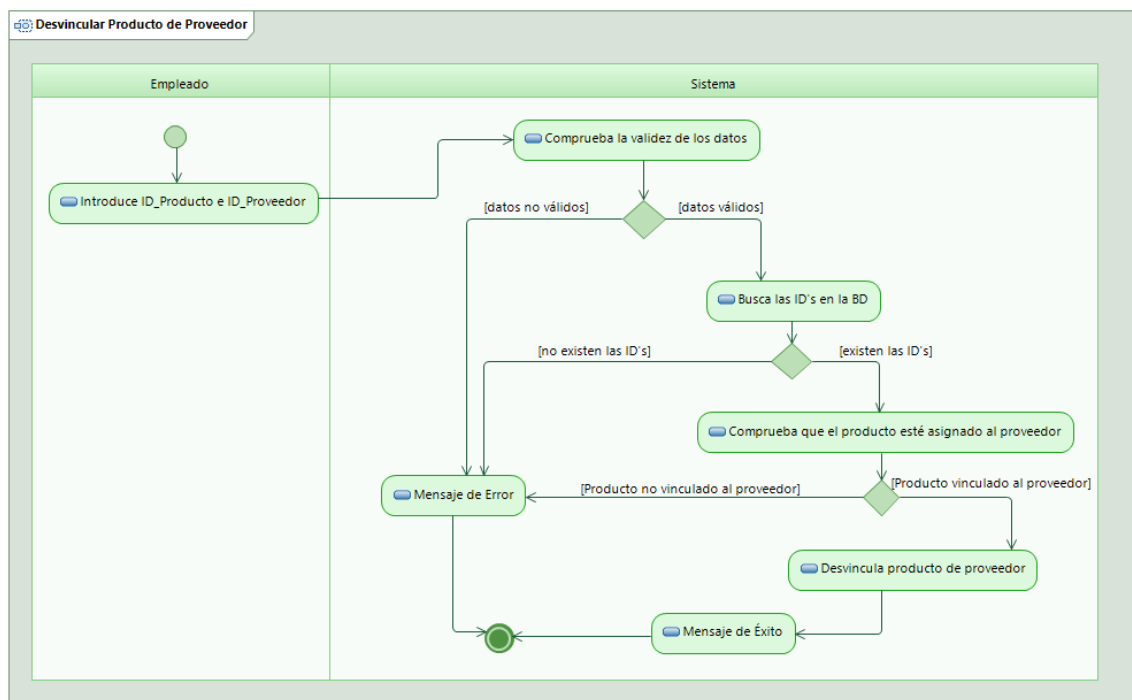
Función 3.2.1.5 Actualizar/Modificar stock e información de un producto	
Prioridad	Media.
Estabilidad	Media.
Descripción	Se modificarán los productos.
Entrada	ID_Producto.
Salida	Mensaje de confirmación.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de productos.
Acción	Busca el producto en la base de datos, en caso de que exista, cambiará los datos.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se modificará el producto.
Efectos laterales	Ninguno.



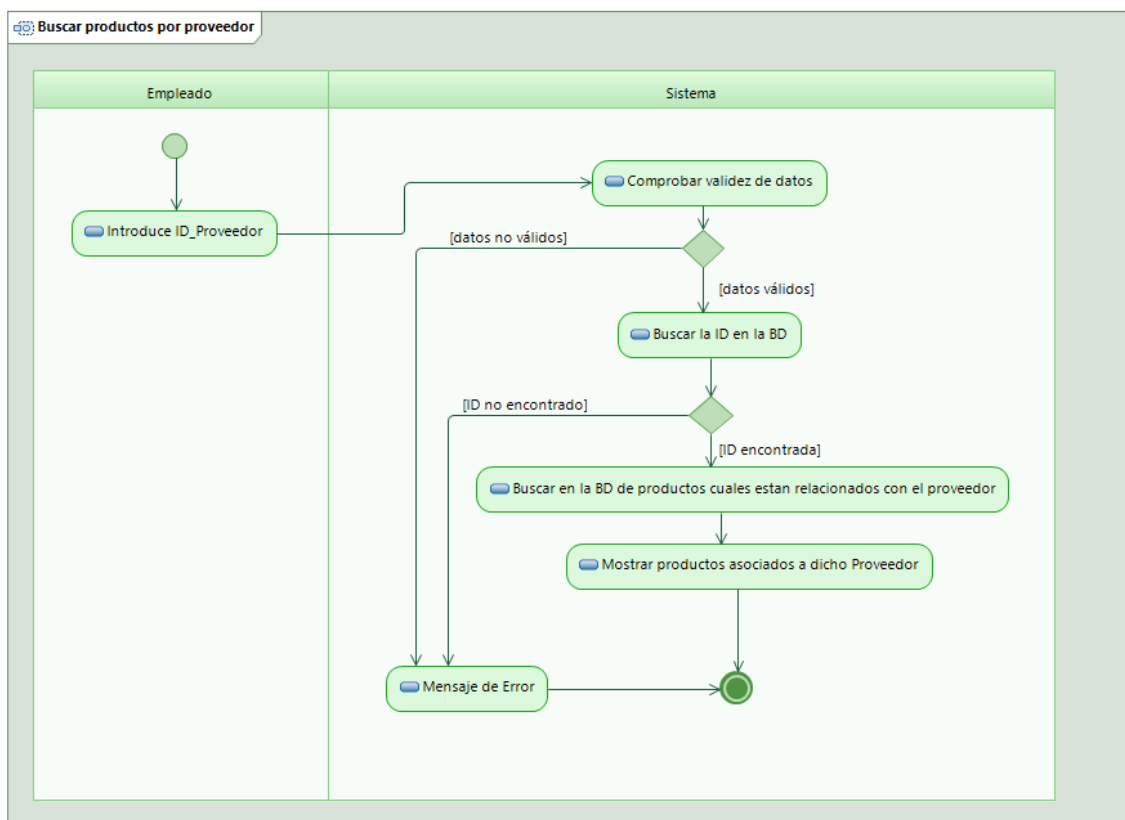
Función 3.2.1.6 Eliminar Producto	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Se elimina un producto del sistema.
Entrada	ID_Producto.
Salida	Mensaje de que se ha eliminado.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de productos.
Acción	Busca el producto en la base de datos, en caso de que exista, lo elimina.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se eliminará el producto.
Efectos laterales	Ninguno.



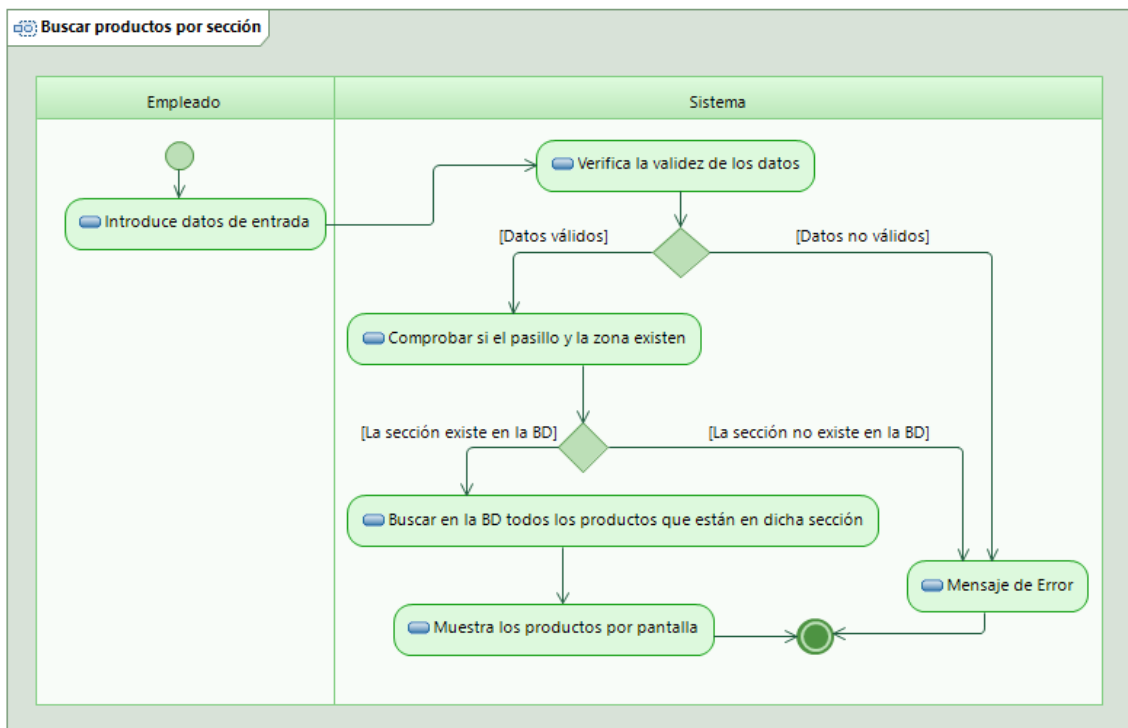
Función 3.2.1.7 Desvincular Producto de Proveedor	
Prioridad	Media.
Estabilidad	Media.
Descripción	Desvincula un producto de un proveedor específico.
Entrada	ID_Producto e Id_Proveedor.
Salida	Muestra mensaje de confirmación.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de productos y de proveedores.
Acción	Comprobar que los IDs sean correctos y desasignar el producto a un proveedor en concreto.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si hay éxito, el producto será desvinculado.
Efectos laterales	Ninguno.



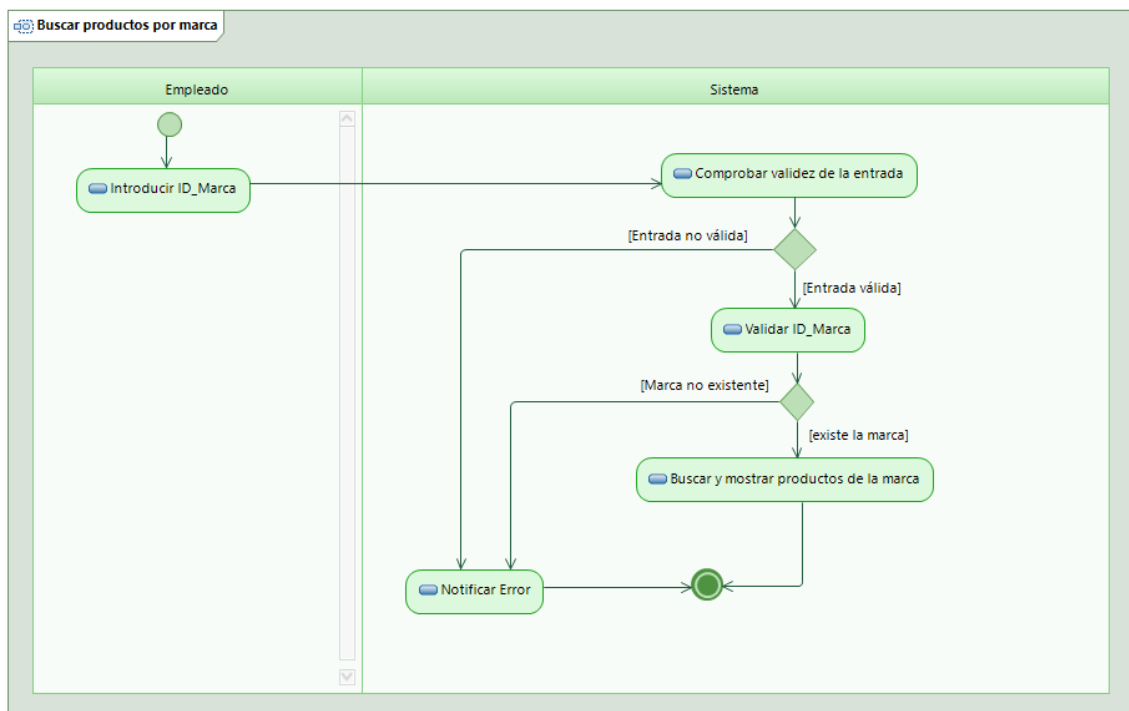
Función 3.2.1.8		Buscar productos por proveedor	
Prioridad		Media.	
Estabilidad		Alta.	
Descripción		Busca y muestra los productos que proporciona el proveedor indicado.	
Entrada		ID_Proveedor.	
Salida		Muestra mensaje de confirmación.	
Origen		Interfaz Usuario – Empleado.	
Destino		Sistema.	
Necesita		Base de datos de productos y de proveedores	
Acción		Comprueba que los datos han sido bien introducidos, seguido comprueba que el proveedor exista, si tiene éxito mostrará los productos adquiridos por dicho proveedor.	
Precondición		No hay identificadores repetidos en la BD.	
Postcondición		No hay identificadores repetidos en la BD; si tiene éxito, mostrará los productos asignados a dicho proveedor.	
Efectos laterales		Ninguno.	



Función 3.2.1.9		Buscar Productos por Sección
Prioridad		Alta.
Estabilidad		Media.
Descripción		Realiza una búsqueda de todos los productos de nuestra base de datos pertenecientes a una sección concreta.
Entrada		Zona y Pasillo.
Salida		Muestra una lista de productos pertenecientes a la sección.
Origen		Interfaz Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos de productos y base de datos de sección.
Acción		Comprueba que los datos introducidos son sintácticamente correctos, posteriormente se asegura de que existe una sección cuya zona y pasillo coinciden con los datos introducidos y finalmente busca todos los productos que pertenecen a dicha sección
Precondición		Ninguna.
Postcondición		Si tiene éxito, muestra los productos por pantalla.
Efectos laterales		Ninguno.

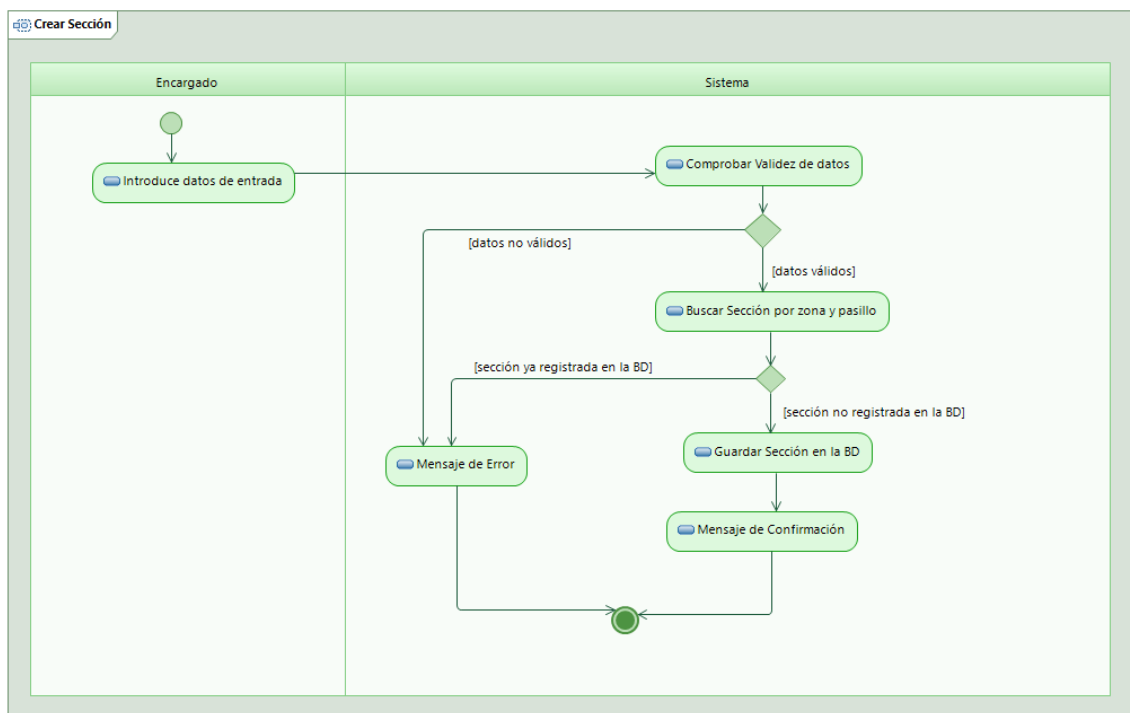


Función 3.2.1.10		Buscar productos por marca
Prioridad		Media.
Estabilidad		Alta.
Descripción		Busca y muestra los productos vinculados a la marca indicada.
Entrada		ID_Marca.
Salida		Muestra mensaje de confirmación.
Origen		Interfaz Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos de productos y de proveedores
Acción		Comprueba que los datos han sido bien introducidos, seguido comprueba que la marca exista, si tiene éxito mostrará los productos de dicha marca.
Precondición		No hay identificadores repetidos en la BD.
Postcondición		No hay identificadores repetidos en la BD; si tiene éxito, mostrará los productos de dicha marca.
Efectos laterales		Ninguno.

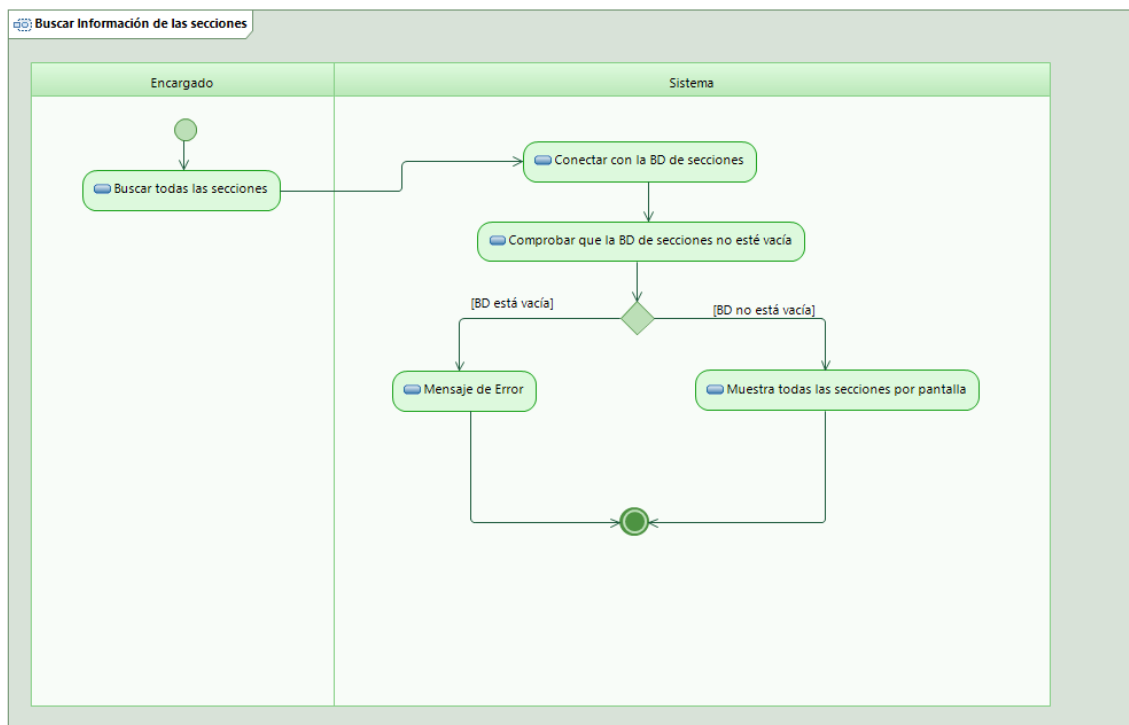


3.2.2 Módulo de Secciones

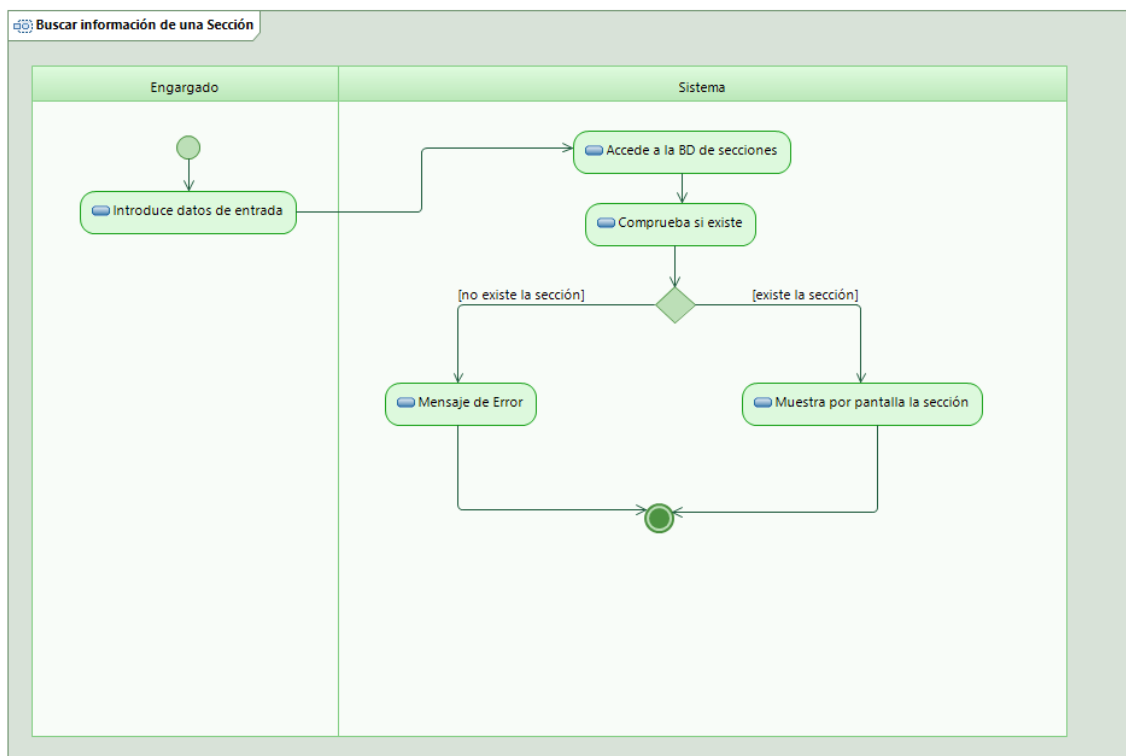
Función 3.2.2.1 Crear una sección	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Crea una sección en la BD de secciones.
Entrada	Zona y pasillo.
Salida	Mensaje de confirmación.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos secciones.
Acción	Comprueba si están mal introducidos los datos. En caso de estar bien introducidos, comprueba que no haya una sección ya con ese nombre, si la hay mandara un mensaje de error, si es nueva, se mandará un mensaje de confirmación.
Precondición	No hay secciones repetidas en la BD.
Postcondición	No hay secciones repetidas en la BD; si tiene éxito, se creará la nueva sección.
Efectos laterales	Ninguno.



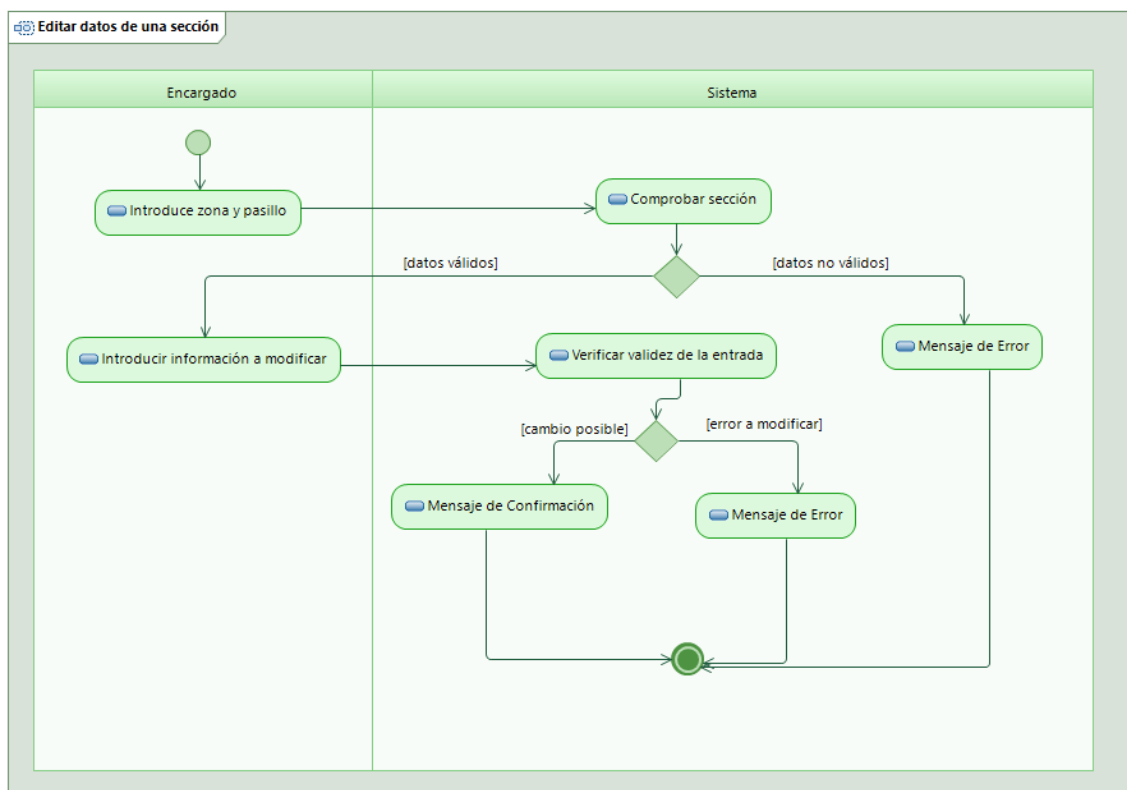
Función 3.2.2.2 Buscar información de las secciones	
Prioridad	Media.
Estabilidad	Alta.
Descripción	Busca y muestra los datos de todas las secciones.
Entrada	Ninguna.
Salida	Todos los atributos de todas las secciones.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de secciones.
Acción	Comprueba todas las secciones que existen para mostrar los atributos, en caso de que no exista ninguna, muestra mensaje de error.
Precondición	Que la BD de secciones este vacía.
Postcondición	Que la BD de secciones este vacía; si éxito, imprimirá los datos por pantalla.
Efectos laterales	Ninguno.



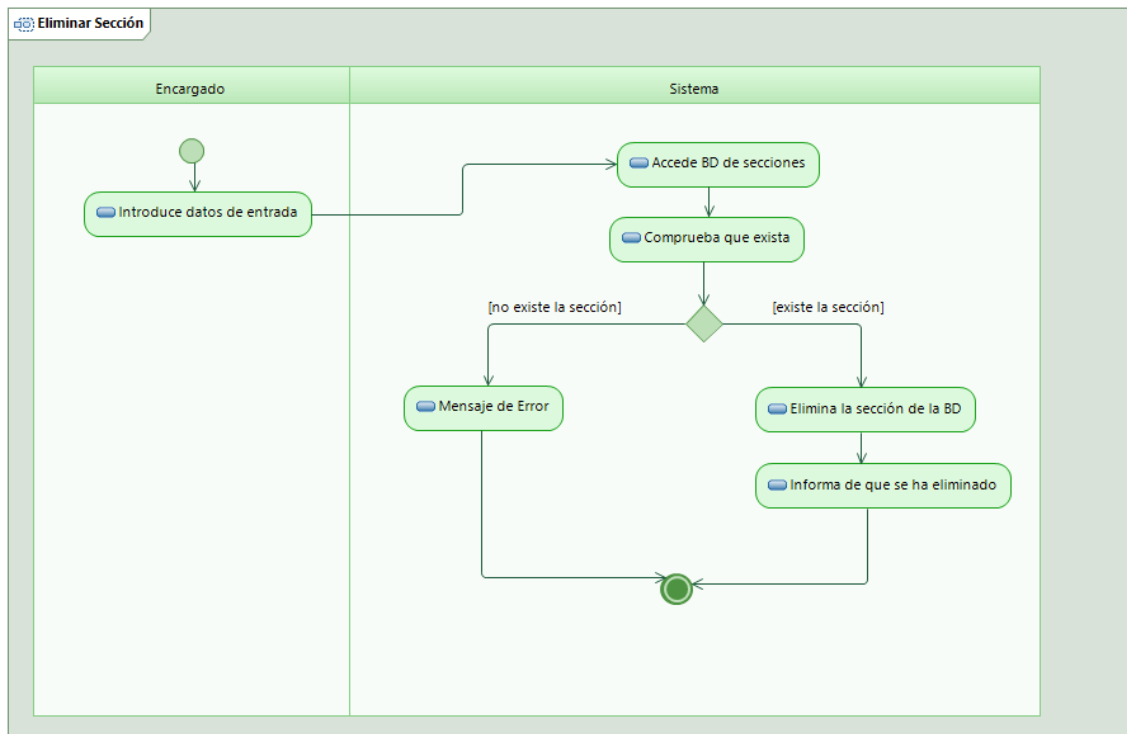
Función 3.2.2.3		Buscar información de una sección
Prioridad		Media.
Estabilidad		Alta.
Descripción		Busca y muestra los datos de la sección indicada.
Entrada		Zona y pasillo.
Salida		Todos los atributos de la sección solicitada.
Origen		Interfaz Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos de secciones.
Acción		Comprueba si existe la sección que se quiere buscar. En caso de existir muestra todos sus atributos, si no, muestra mensaje de error.
Precondición		No hay secciones repetidas en la BD.
Postcondición		No hay secciones repetidas en la BD; si éxito, imprimirá los datos por pantalla.
Efectos laterales		Ninguno.



Función 3.2.2.4 Editar datos de sección	
Prioridad	Media.
Estabilidad	Media.
Descripción	Se modificará la sección.
Entrada	Zona y pasillo.
Salida	Mensaje de confirmación.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de secciones.
Acción	Busca la sección en la base de datos, en caso de que exista, comprobara su validez.
Precondición	No hay secciones repetidas en la BD.
Postcondición	No hay secciones repetidas en la BD; si tiene éxito, se modificará la sección.
Efectos laterales	Ninguno.

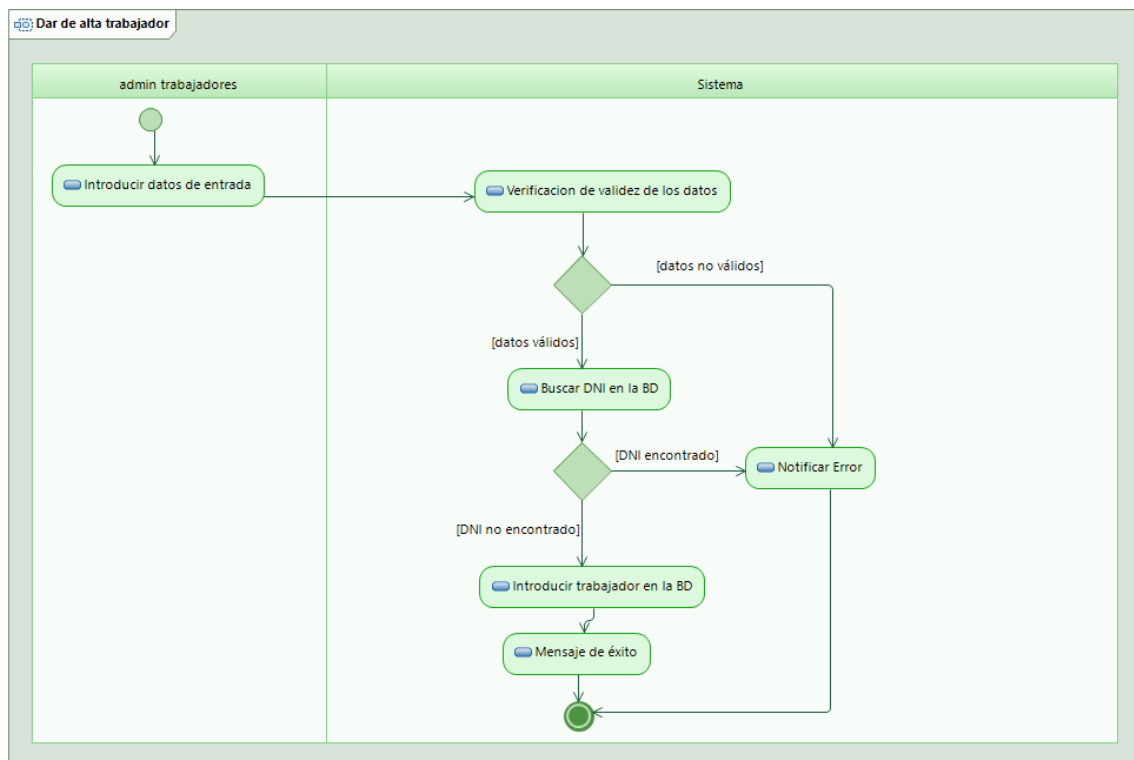


Función 3.2.2.5		Eliminar una sección
Prioridad		Alta.
Estabilidad		Media.
Descripción		Se elimina una sección del sistema.
Entrada		Zona y Pasillo.
Salida		Mensaje de que se ha eliminado.
Origen		Interfaz Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos de secciones.
Acción		Busca la sección en la base de datos, en caso de que exista, la elimina.
Precondición		No hay secciones repetidas en la BD.
Postcondición		No hay secciones repetidas en la BD; si tiene éxito, se eliminará.
Efectos laterales		Eliminar una Sección implicará que todos los productos asociados a esa sección pondrán este campo a null.

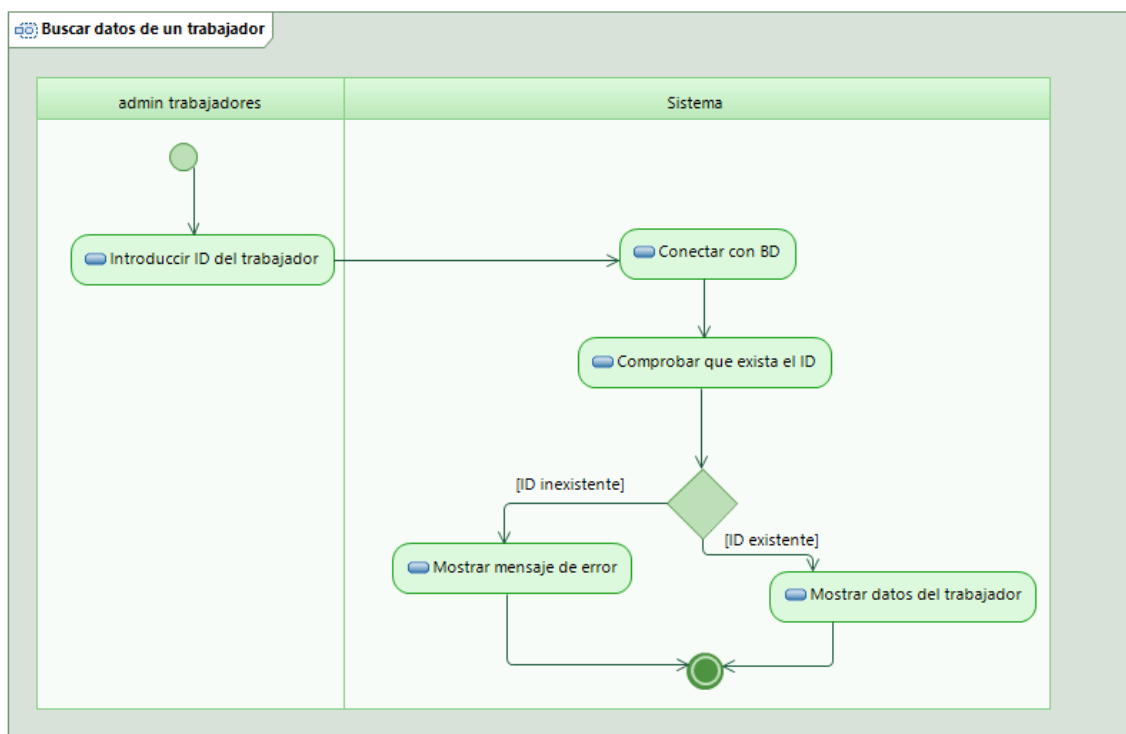


3.2.3 Módulo de Trabajadores

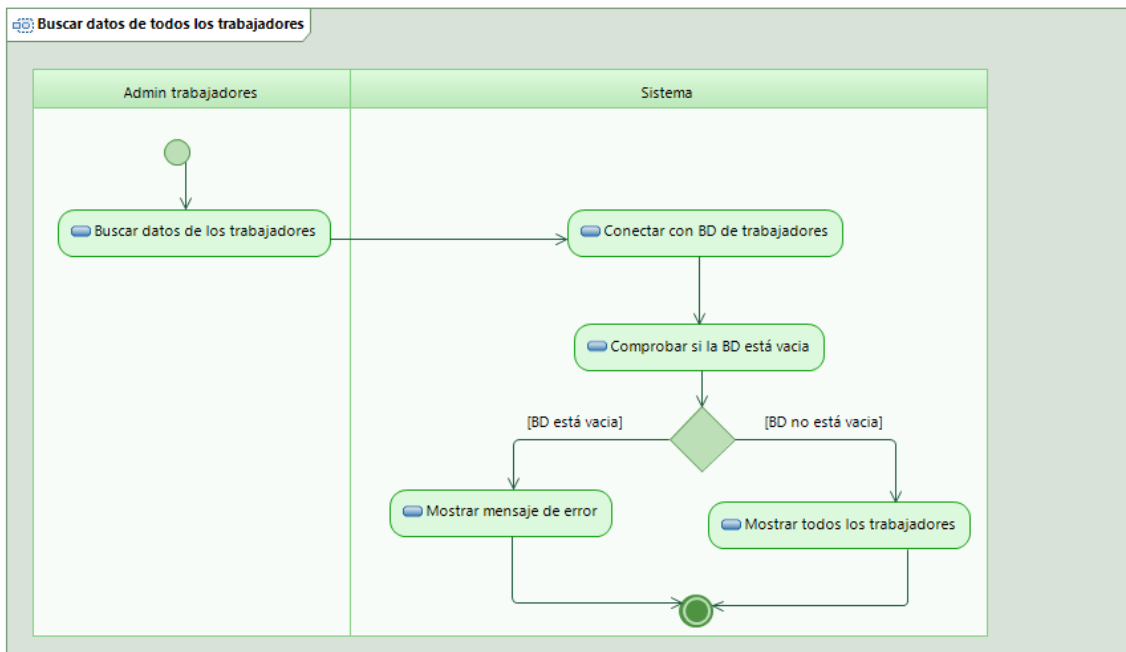
Función 3.2.3.1 Dar de alta a un nuevo trabajador	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Da de alta a un trabajador del supermercado.
Entrada	Nombre, mail, DNI.
Salida	ID del trabajador.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de trabajadores.
Acción	Comprobar que los datos sean correctos y que no hay otros trabajadores con el mismo DNI en caso afirmativo dar de alta al trabajador y devolver la ID.
Precondición	No hay identificadores ni DNIs repetidos en la BD.
Postcondición	No hay identificadores ni DNIs repetidos en la BD; si tiene éxito, se añadirá el nuevo trabajador.
Efectos laterales	Si falla el procedimiento avisar mediante un mensaje para repetir el proceso.



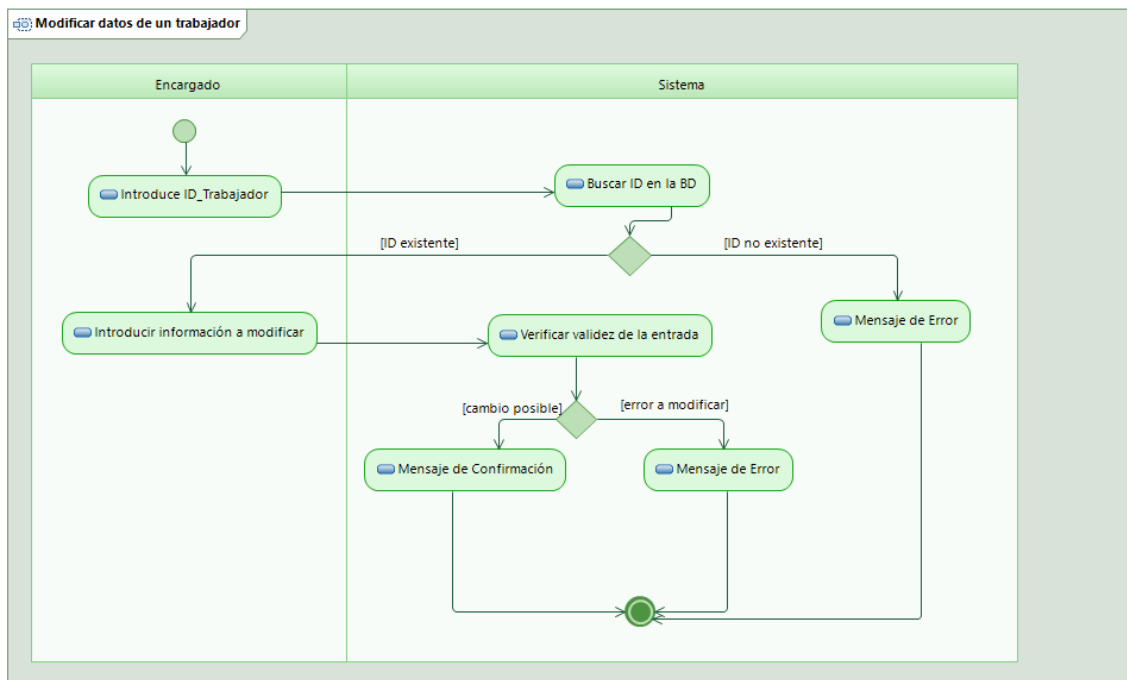
Función 3.2.3.2		Buscar datos de un trabajador
Prioridad		Media.
Estabilidad		Alta.
Descripción		Buscar y mostrar los datos de un trabajador.
Entrada		ID_Trabajador.
Salida		Todos los atributos del trabajador.
Origen		Interfaz Usuario – Administrador Trabajadores.
Destino		Sistema.
Necesita		Base de datos de trabajadores.
Acción		Comprueba si existe el trabajador que se quiere buscar. En caso de existir, muestra todos sus atributos, si no, muestra mensaje de error.
Precondición		No hay identificadores repetidos en la BD.
Postcondición		No hay identificadores repetidos en la BD ; si tiene éxito, se muestra por pantalla la información del trabajador.
Efectos laterales		Ninguno.



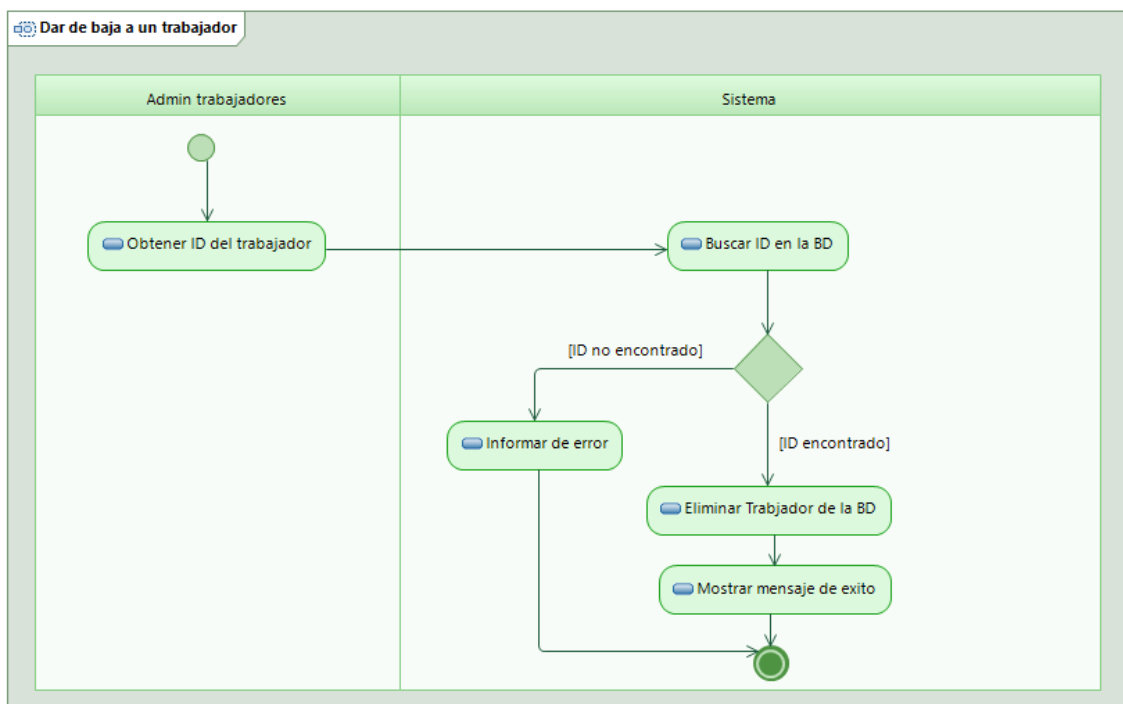
Función 4.2.3.3 Buscar datos de todos los trabajadores	
Prioridad	Media.
Estabilidad	Alta.
Descripción	Buscar y mostrar los datos de todos los trabajadores.
Entrada	Ninguna.
Salida	Todos los atributos de los trabajadores.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de trabajadores.
Acción	Comprueba todos los trabajadores que existen para mostrar los atributos, en caso de que no exista ninguno, muestra mensaje de error.
Precondición	Comprobar que la BD no esté vacía.
Postcondición	Comprobar que la BD no esté vacía; si tiene éxito, se mostrarán los datos de todos los trabajadores.
Efectos laterales	Ninguno.



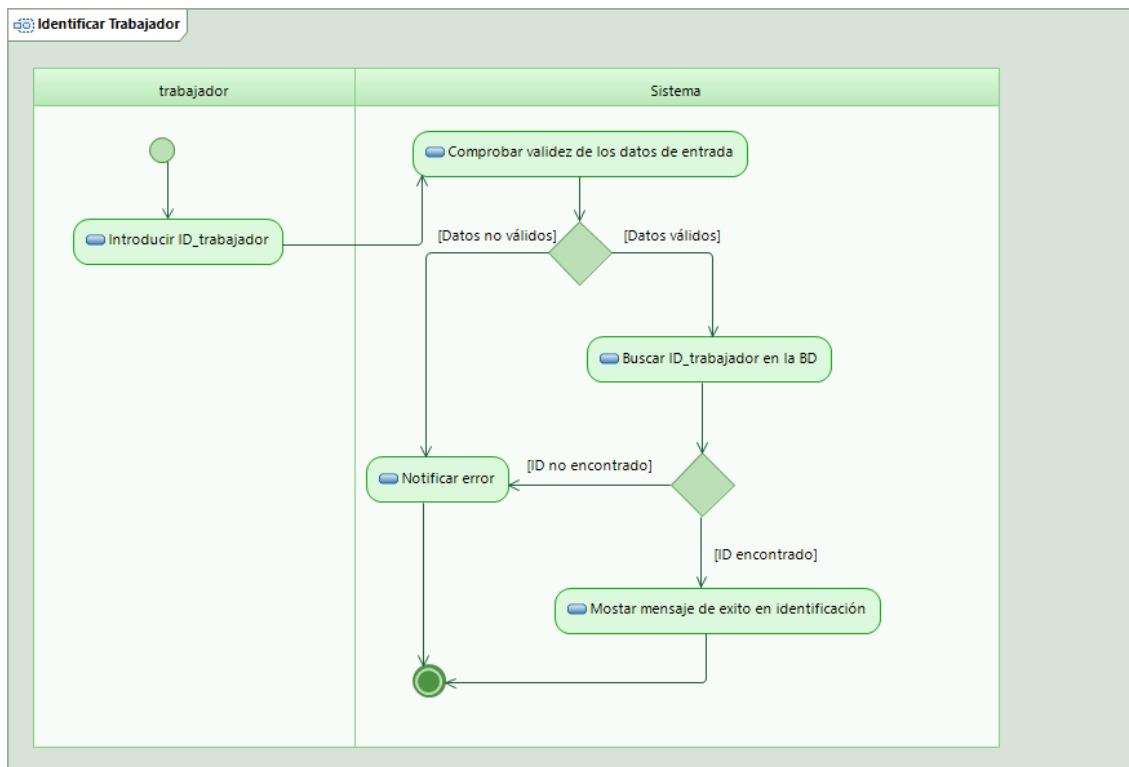
Función 3.2.3.4 Modificar datos de un trabajador	
Prioridad	Media.
Estabilidad	Media.
Descripción	Modificar los datos de un trabajador.
Entrada	ID del trabajador a modificar y los atributos que se desean modificar.
Salida	Mensaje de éxito.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de trabajadores.
Acción	Buscará al trabajador por ID y se modificarán los datos que haya introducido el usuario por los que había antes.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, modificará los datos del trabajador.
Efectos laterales	Ninguno.



Función 3.2.3.5 Dar de baja a un trabajador	
Prioridad	Media.
Estabilidad	Media.
Descripción	Da de baja a un trabajador del supermercado.
Entrada	ID del trabajador.
Salida	Mensaje de éxito.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de trabajadores.
Acción	Buscará al trabajador por ID y se eliminará de la BD.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se dará de baja el trabajador.
Efectos laterales	Si falla el procedimiento avisar mediante un mensaje para repetir el proceso.

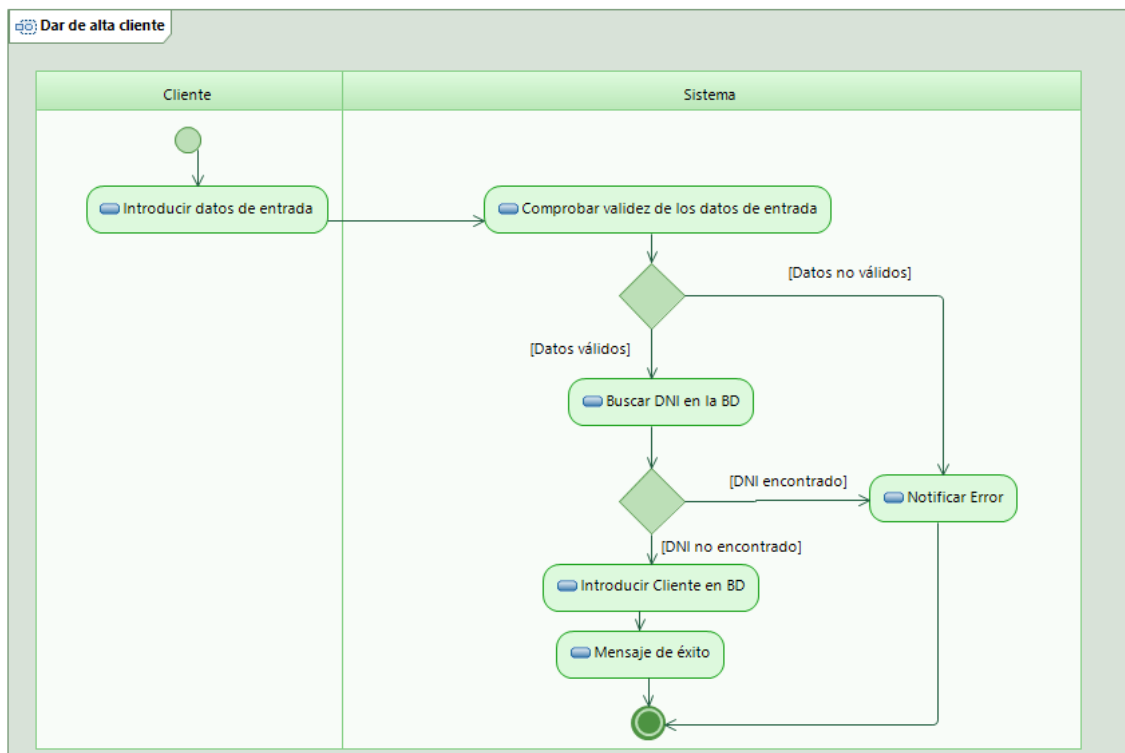


Función 3.2.3.6 Identificar al trabajador	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Identificar a un trabajador.
Entrada	ID_Trabajador.
Salida	Mensaje de éxito en registro del trabajador.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de trabajadores.
Acción	Busca en la base de datos de trabajadores, y se registra como empleado para tener acceso a la máquina.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si existe, se registrará y mostrará un mensaje de registro realizado correctamente.
Efectos laterales	Ninguno.

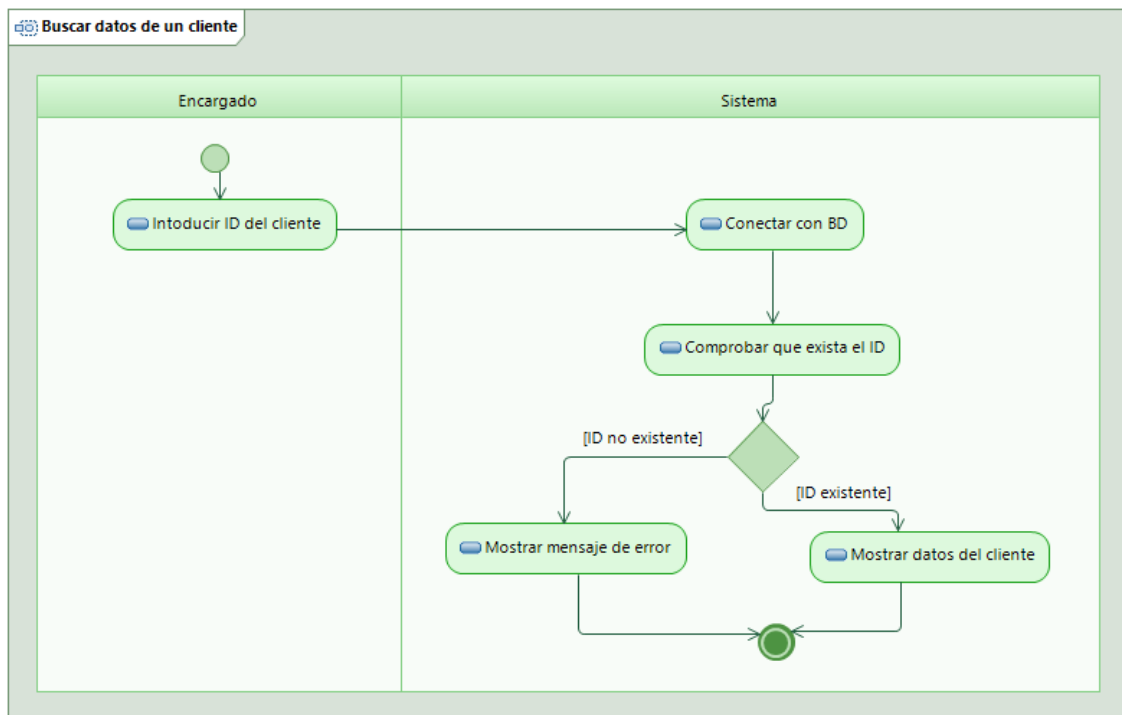


3.2.4 Módulo de Clientes

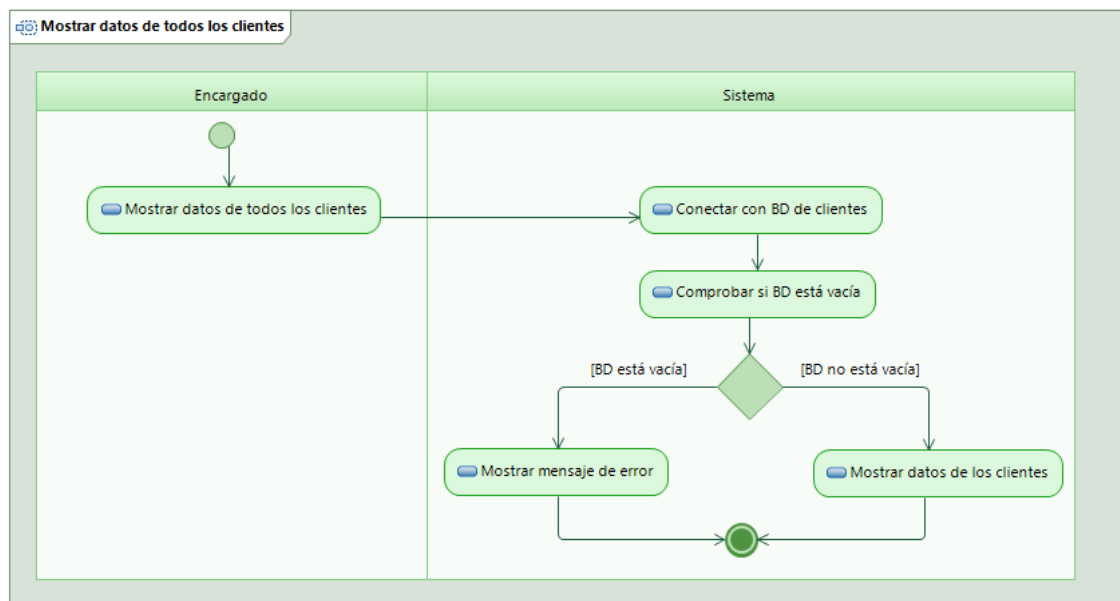
Función 3.2.4.1 Dar de alta a un nuevo cliente	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Da de alta a un nuevo cliente.
Entrada	Nombre, mail, DNI.
Salida	ID del cliente.
Origen	Interfaz de Usuario – Nuevo Cliente.
Destino	Sistema.
Necesita	Base de datos de clientes.
Acción	Comprobar que los datos sean correctos y que no hay otros clientes con el mismo DNI en caso afirmativo dar de alta al cliente y devolver ID.
Precondición	No hay identificadores ni DNIs repetidos en la BD.
Postcondición	No hay identificadores repetidos ni DNIs en la BD; si tiene éxito, nuevo cliente añadido.
Efectos laterales	Si falla el procedimiento avisar mediante un mensaje para repetir el proceso.



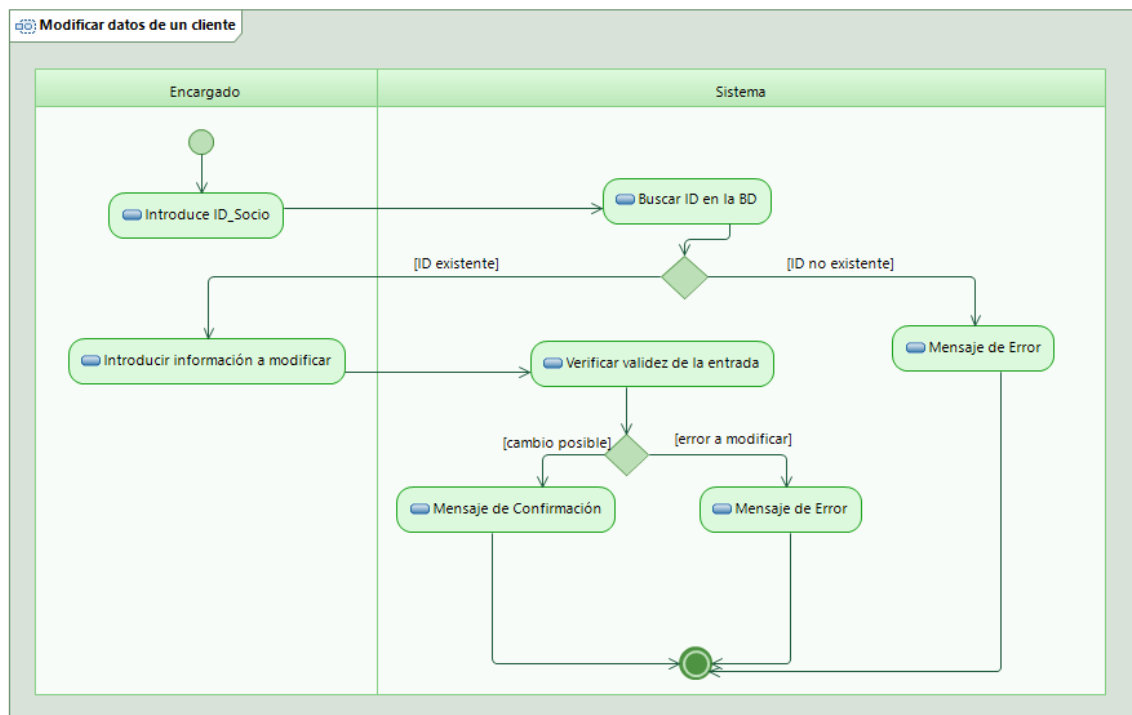
Función 3.2.4.2		Buscar datos de un cliente
Prioridad		Media.
Estabilidad		Alta.
Descripción		Buscar y mostrar los datos de un cliente.
Entrada		ID_Socio.
Salida		Todos los atributos del cliente.
Origen		Interfaz de Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos de clientes.
Acción		Comprueba si existe el cliente que se quiere buscar. En caso de existir muestra todos sus atributos, si no, muestra mensaje de error.
Precondición		No hay identificadores repetidos en la BD.
Postcondición		No hay identificadores repetidos en la BD; si tiene éxito, se muestra por pantalla la información del cliente.
Efectos laterales		Ninguno.



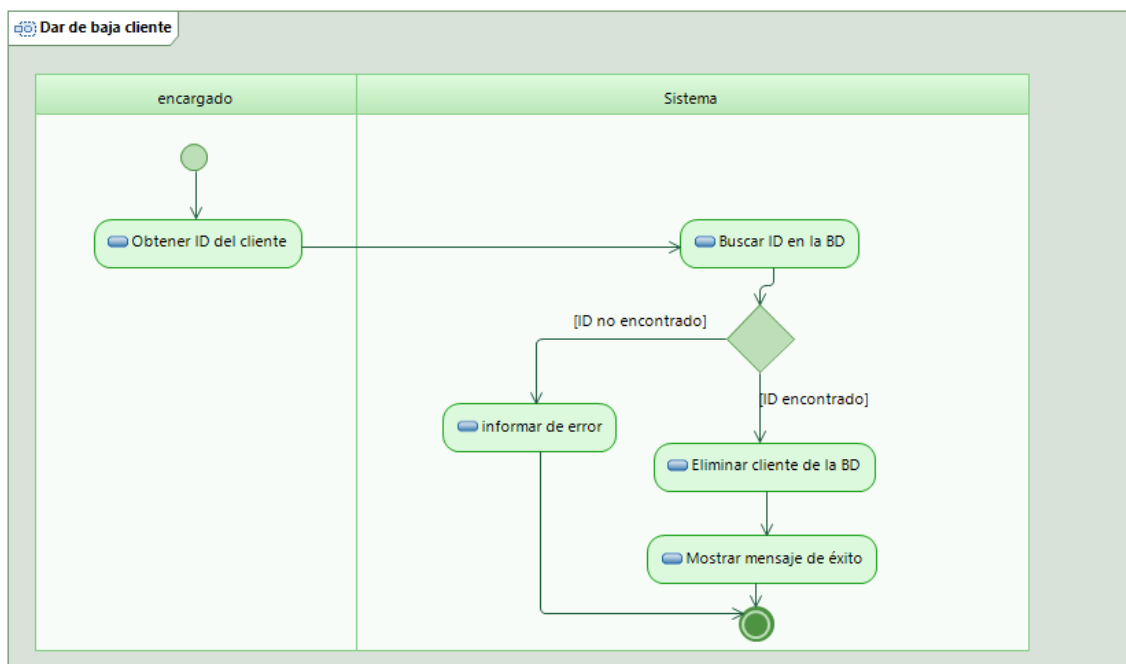
Función 3.2.4.3 Buscar datos de todos los clientes	
Prioridad	Media.
Estabilidad	Alta.
Descripción	Buscar y mostrar los datos de todos los clientes.
Entrada	Ninguna.
Salida	Todos los atributos de los clientes.
Origen	Interfaz de Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de clientes.
Acción	Comprueba todos los clientes que existen para mostrar los atributos, en caso de que no exista ninguno, muestra mensaje de error.
Precondición	Que la BD no esté vacía.
Postcondición	Si tiene éxito, se muestra por pantalla la información de los clientes.
Efectos laterales	Ninguno.



Función 3.2.4.4 Modificar datos de un cliente	
Prioridad	Media.
Estabilidad	Media.
Descripción	Modificar los datos de un cliente.
Entrada	ID del cliente a modificar y los atributos que se desean modificar.
Salida	Mensaje de confirmación.
Origen	Interfaz de Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de clientes.
Acción	Buscará al cliente por ID y se modificarán los datos que haya introducido el usuario por los que había antes.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, modificará los datos del cliente.
Efectos laterales	Ninguno.

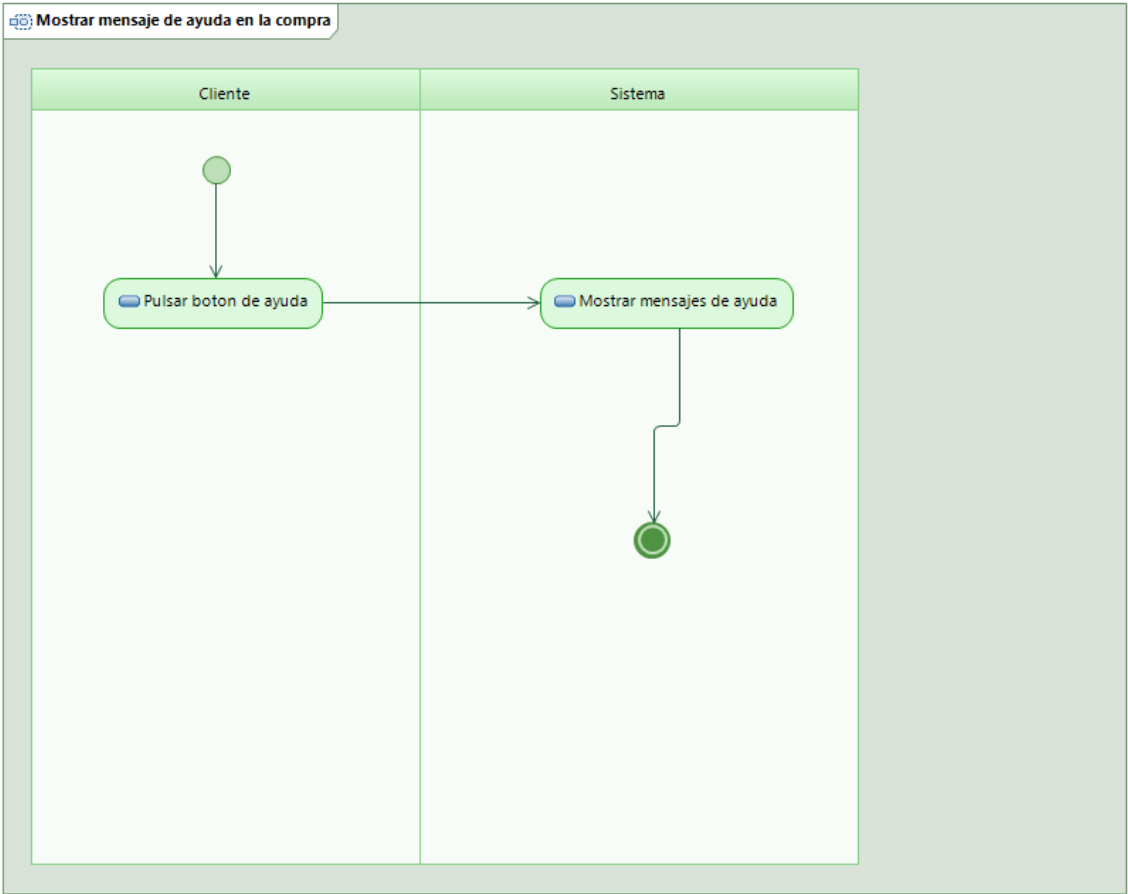


Función 3.2.4.5 Dar de baja a un cliente	
Prioridad	Media
Estabilidad	Media.
Descripción	Da de baja a un cliente.
Entrada	ID del cliente.
Salida	Mensaje de éxito.
Origen	Interfaz de Usuario – Cliente.
Destino	Sistema.
Necesita	Base de datos de clientes.
Acción	Buscar al cliente por su ID en la BD y eliminarlo de ella.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si hay éxito, se dará de baja el cliente.
Efectos laterales	Si falla el procedimiento avisar mediante un mensaje para repetir el proceso.

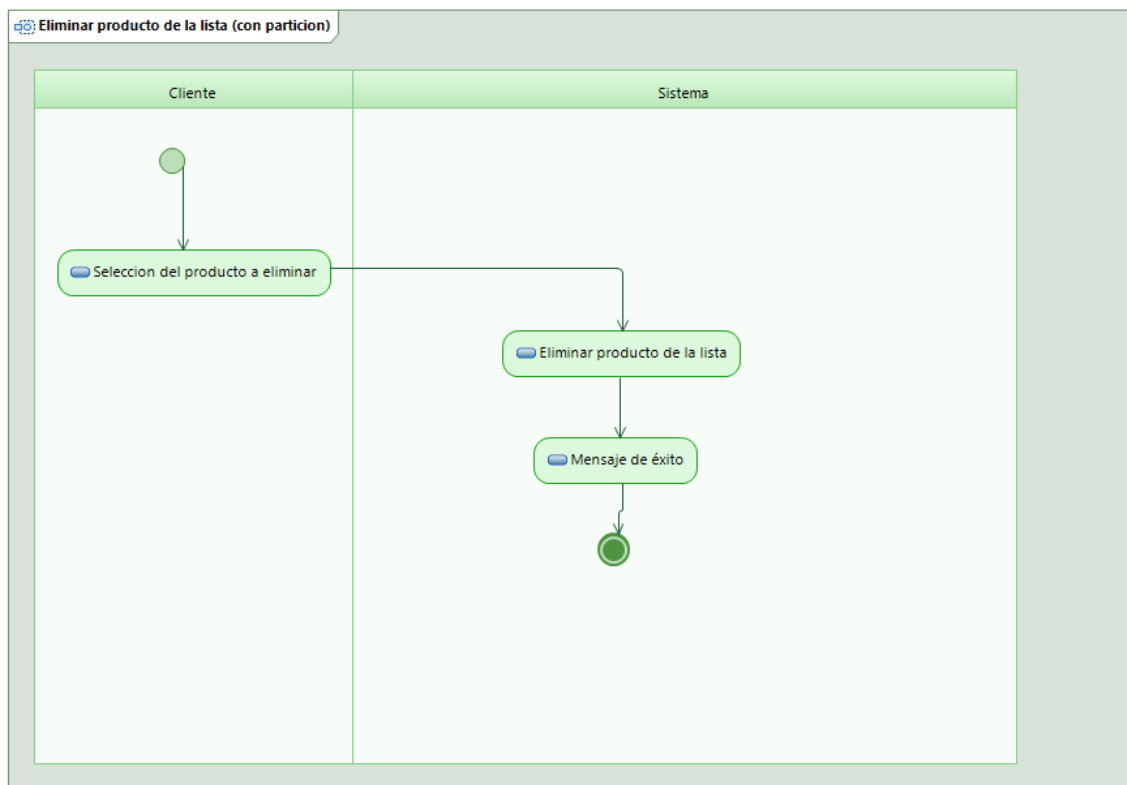


3.2.5 Módulo de Compras

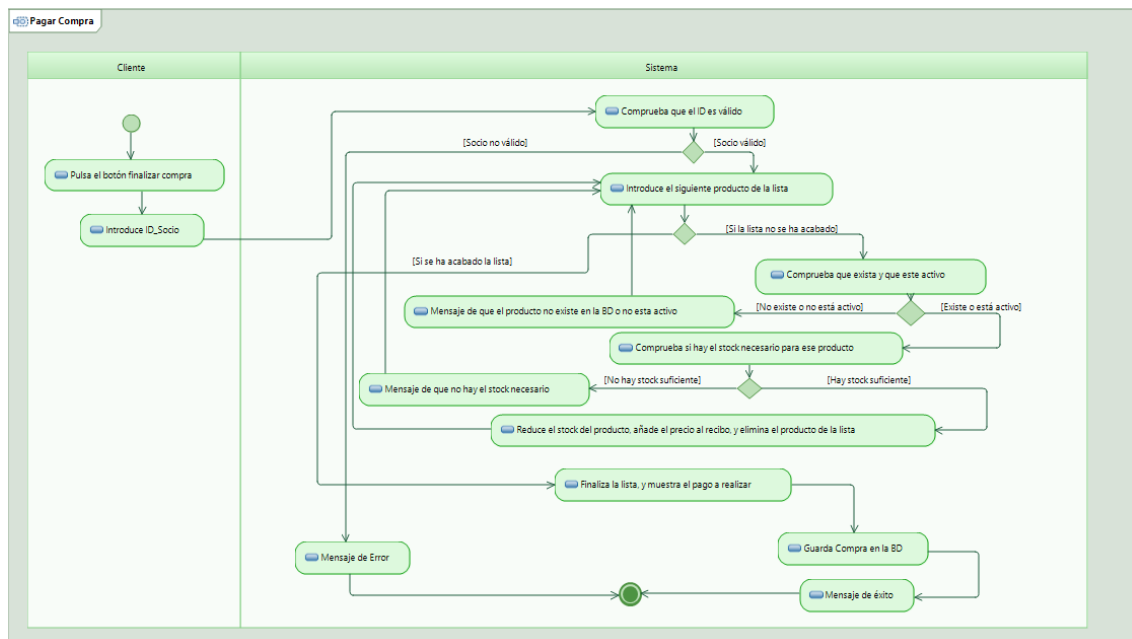
Función 3.2.5.1 Mostrar mensajes de ayuda en la compra	
Prioridad	Media.
Estabilidad	Media.
Descripción	Muestra mensajes de ayuda durante el proceso de compra.
Entrada	Ninguna.
Salida	Mensaje de ayuda.
Origen	Interfaz de usuario – Cliente.
Destino	Sistema.
Necesita	Nada.
Acción	Mostrará mensajes de ayuda por pantalla mientras el cliente realiza la compra.
Precondición	Ninguna.
Postcondición	Ninguna.
Efectos laterales	Ninguno.



Función 3.2.5.3 Eliminar producto a la lista	
Prioridad	Alta.
Estabilidad	Media.
Descripción	Se elimina un producto de la lista de compra.
Entrada	ID_Producto.
Salida	Mensaje de que se ha eliminado.
Origen	Interfaz Usuario – Cliente.
Destino	Sistema.
Necesita	Nada.
Acción	Busca el producto en la lista de la compra y lo elimina.
Precondición	Ninguna.
Postcondición	Ninguna.
Efectos laterales	Ninguno.

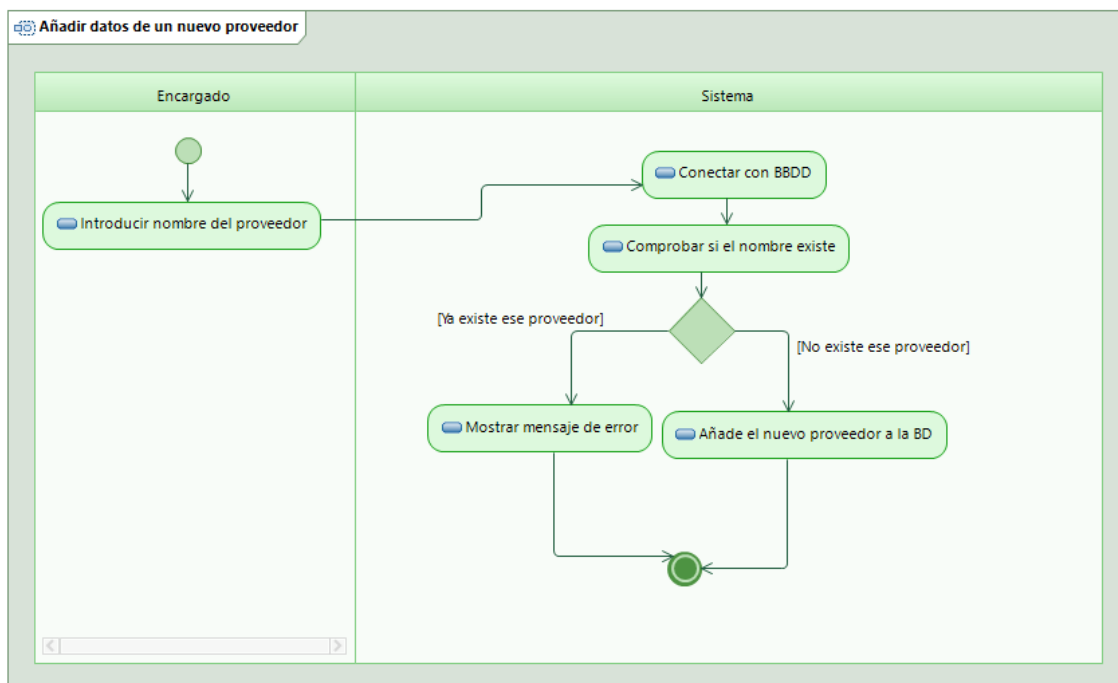


Función 3.2.5.4		Pagar compra
Prioridad		Alta.
Estabilidad		Media.
Descripción		Se confirma el pago de los productos que hay en la lista.
Entrada		Datos Bancarios.
Salida		Mensaje de compra realizada con éxito.
Origen		Interfaz de Usuario – Cliente.
Destino		Sistema.
Necesita		Base de datos de Compra.
Acción		Valida el socio, comprueba cada producto, comprobando si existe, estás activo y tiene el stock necesario, si no se cumple algunas de éstas se informará en la lista, y seguirá comprobando los productos hasta que acabe la lista, mostrando el dinero a pagar y guardando dicha compra en la BD.
Precondición		No hay identificadores repetidos en la BD.
Postcondición		No hay identificadores repetidos en la BD; si tiene éxito, se mostrará el pago a realizar y se guardará en la BD la información de dicha compra.
Efectos laterales		Ninguno.

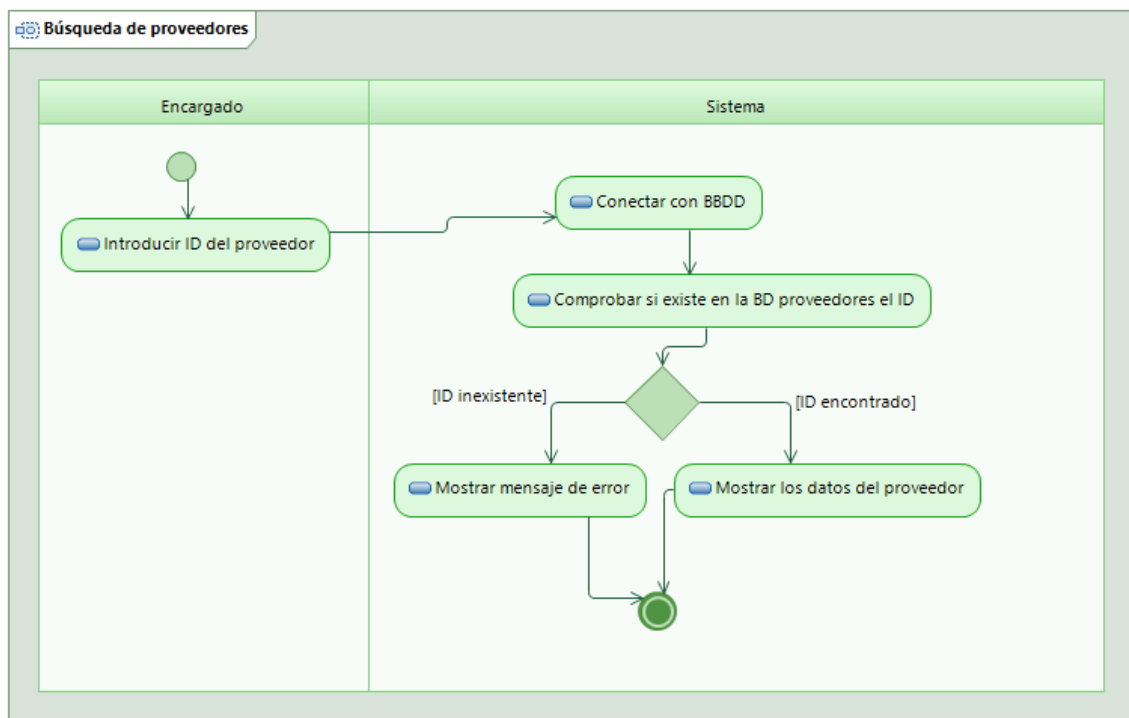


3.2.6 Módulo de Proveedores

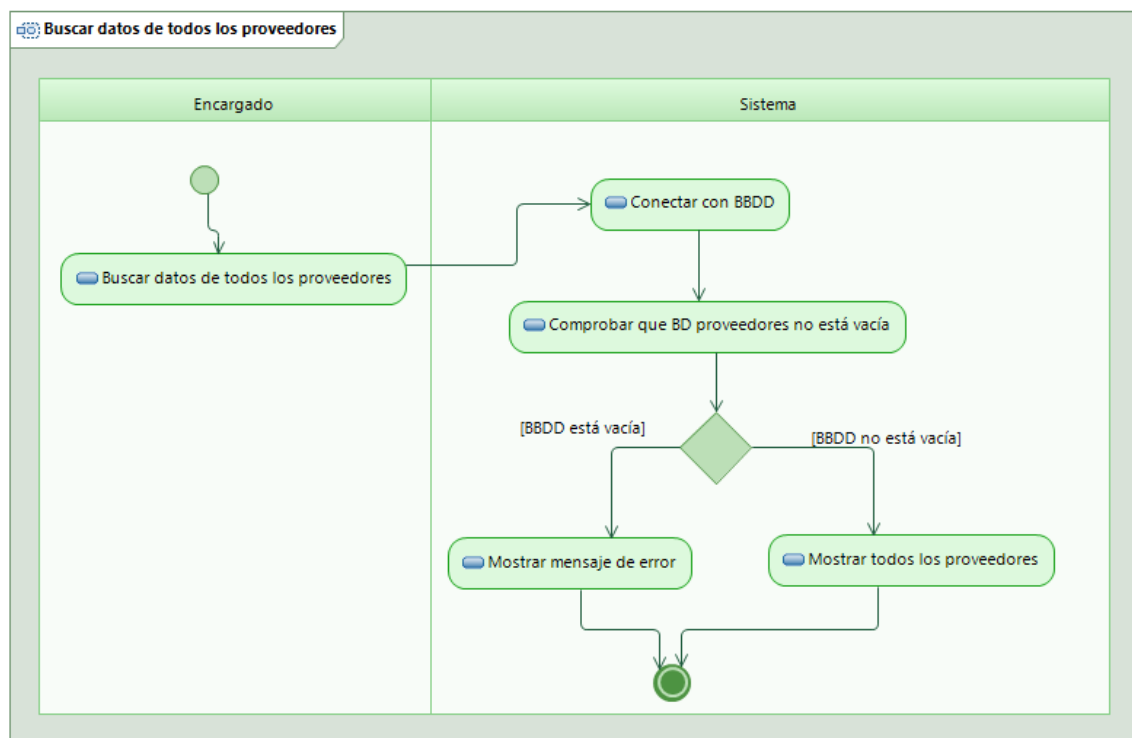
Función 3.2.6.1 Añadir datos de un nuevo proveedor	
Prioridad	Alta.
Estabilidad	Alta.
Descripción	Dar de alta a un nuevo proveedor.
Entrada	Nombre del proveedor.
Salida	ID del proveedor.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de proveedores.
Acción	Añade un nuevo proveedor con los datos introducidos y se incorpora a la BD.
Precondición	No hay identificadores ni nombres repetidos en la BD.
Postcondición	No hay identificadores ni nombres repetidos en la BD; si tiene éxito, se añadirá a la BD.
Efectos laterales	Ninguno.



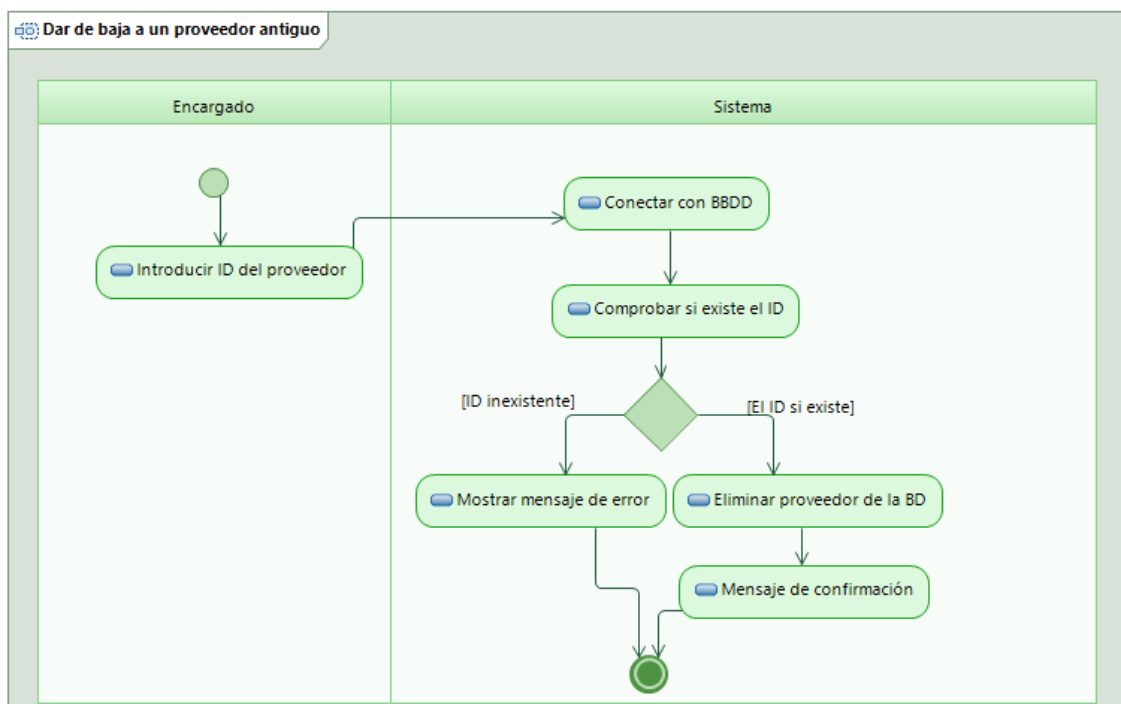
Función 3.2.6.2		Búsqueda de proveedor
Prioridad		Media.
Estabilidad		Media.
Descripción		Se buscarán y mostrarán los datos del proveedor.
Entrada		ID de proveedor.
Salida		Datos del proveedor.
Origen		Interfaz Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos de proveedores.
Acción		Buscará el proveedor y mostrará por pantalla su información.
Precondición		No hay identificadores repetidos en la BD.
Postcondición		No hay identificadores repetidos en la BD; si tiene éxito, se mostrará los datos por pantalla.
Efectos laterales		Ninguno.



Función 3.2.6.3		Buscar datos de un proveedor
Prioridad		Media.
Estabilidad		Media.
Descripción		Busca y muestra los datos de todos los proveedores
Entrada		Ninguna.
Salida		Datos de todos los proveedores.
Origen		Interfaz Usuario – Empleado.
Destino		Sistema.
Necesita		Base de datos proveedores.
Acción		Buscará los proveedores y mostrará por pantalla su información.
Precondición		Que la BD no esté vacía.
Postcondición		Que la BD no esté vacía; si tiene éxito, se mostrará los datos por pantalla.
Efectos laterales		Ninguno.

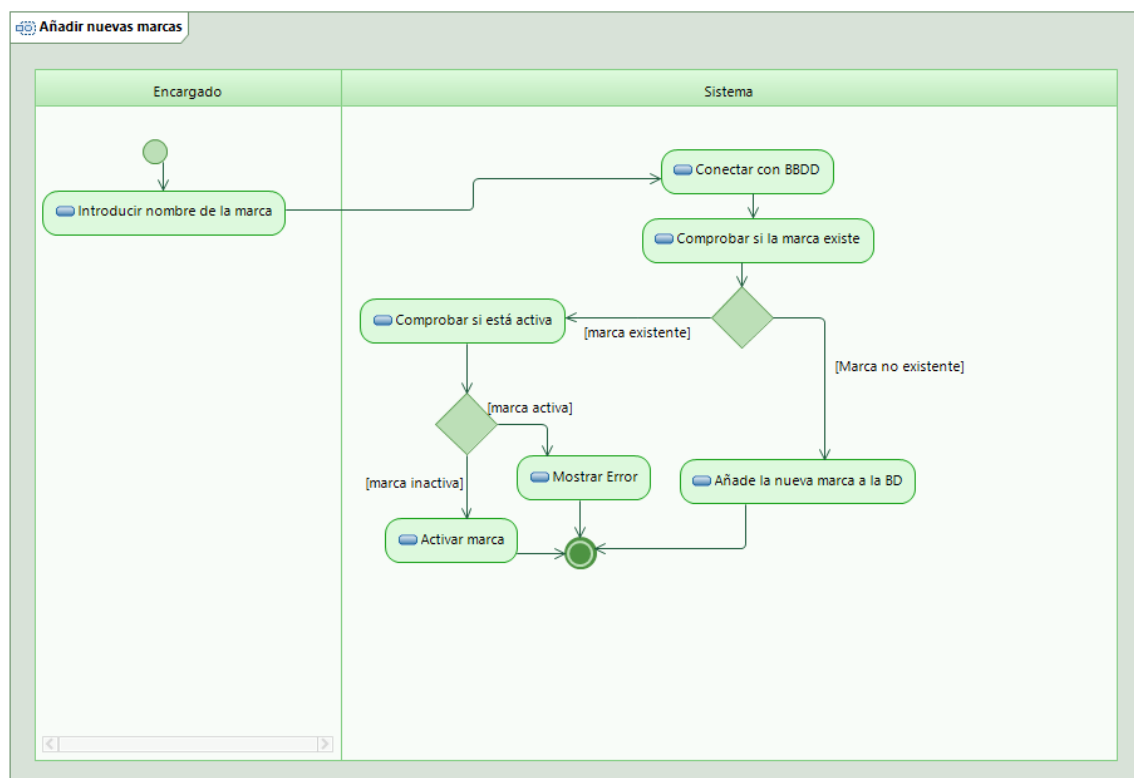


Función 3.2.6.5 Dar de baja a un proveedor antiguo	
Prioridad	Media.
Estabilidad	Media.
Descripción	Se dará de baja a un proveedor.
Entrada	ID del proveedor.
Salida	Mensaje de confirmación.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de proveedor.
Acción	Se dará de baja lógica al proveedor de la BD.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se dará de baja.
Efectos laterales	Ninguno.

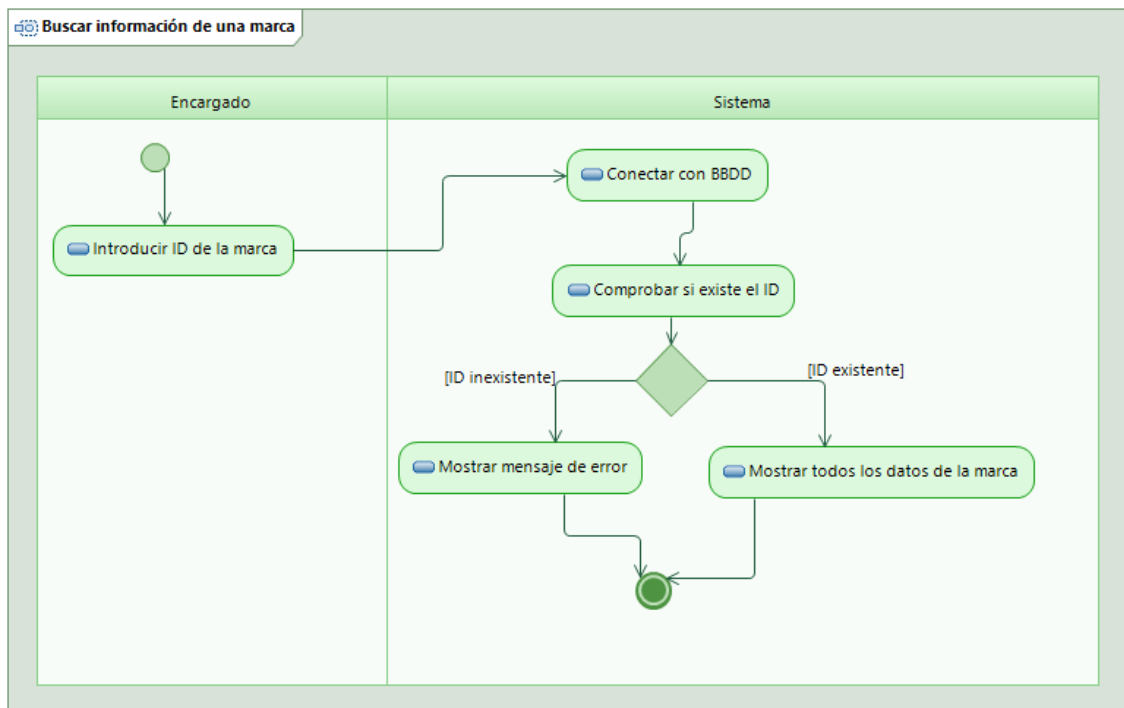


3.2.7 Módulo de Marcas

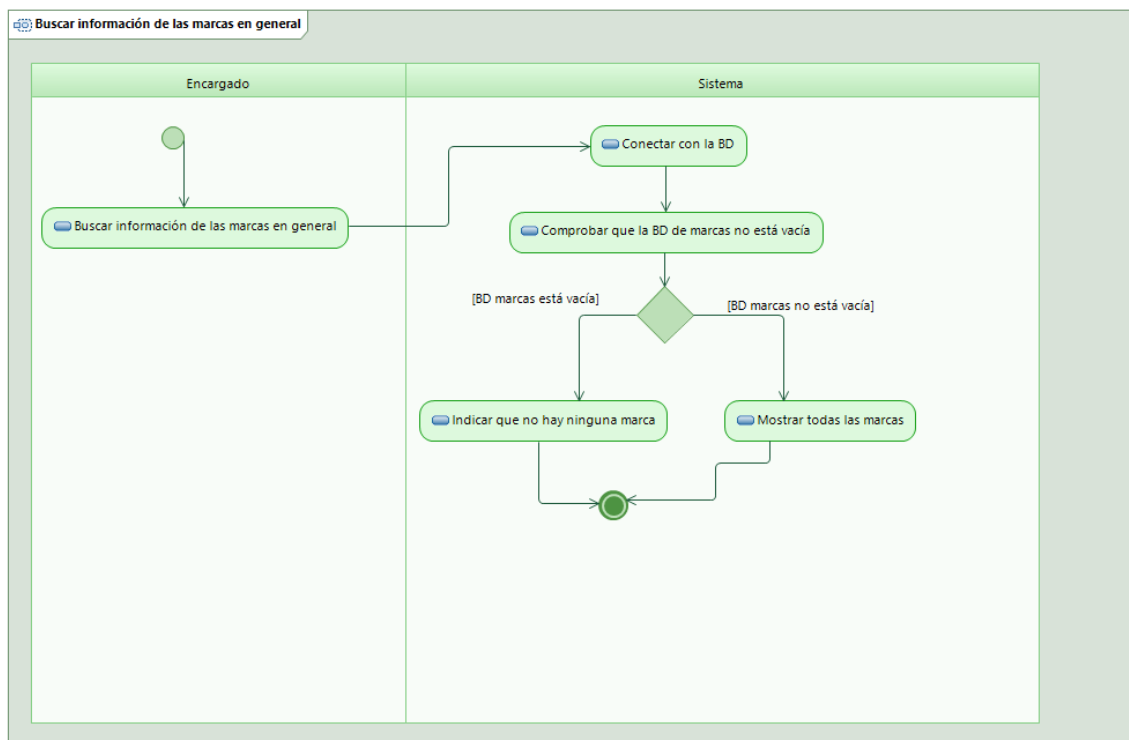
Función 3.2.7.1 Añadir nuevas marcas	
Prioridad	Alta.
Estabilidad	Alta.
Descripción	Añadir marca a la BD.
Entrada	Nombre de la marca.
Salida	ID_Marca.
Origen	Interfaz Usuario – Administrador Trabajadores.
Destino	Sistema.
Necesita	Base de datos de marcas.
Acción	Introduce el nombre de la marca, comprueba si existe y esta activa, si no lo está, se activa, y si no existe, la añade.
Precondición	No hay identificadores ni nombre repetidos en la BD.
Postcondición	No hay identificadores ni nombre repetidos en la BD; si tiene éxito, añade la nueva marca o la activa.
Efectos laterales	Ninguno.



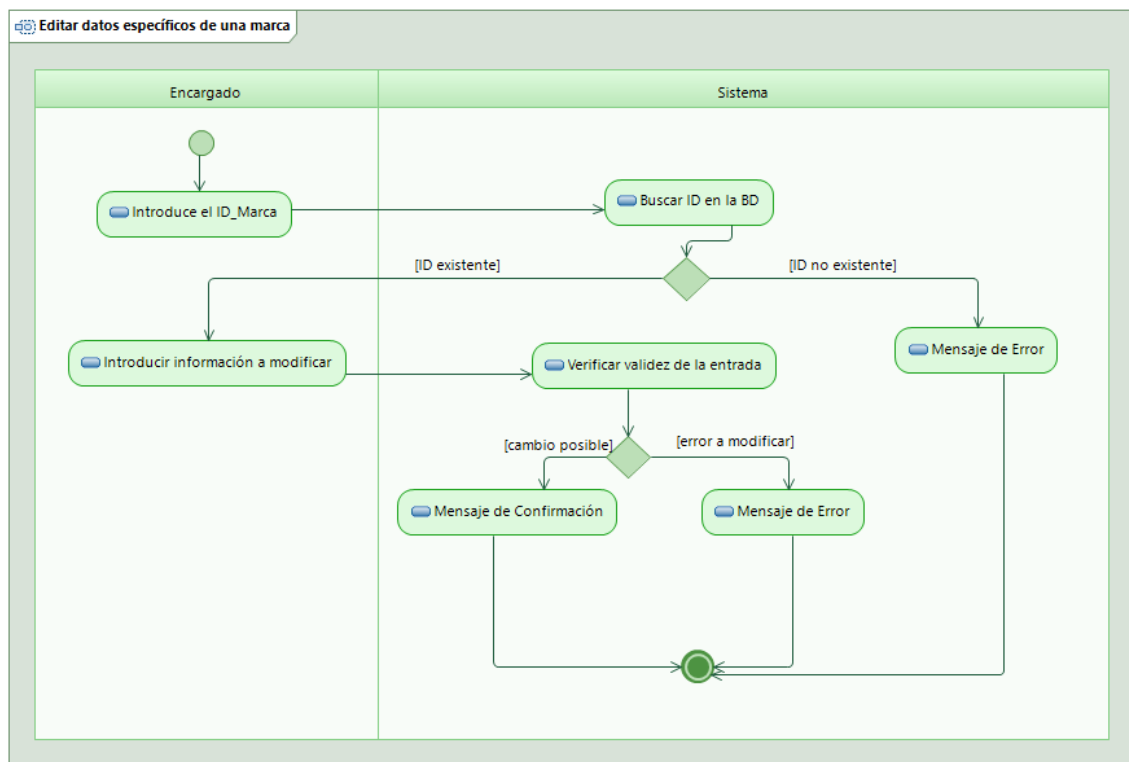
Función 3.2.7.2 Busca información de una marca	
Prioridad	Media.
Estabilidad	Media.
Descripción	Busca y muestra el nombre de la marca.
Entrada	ID_Marca.
Salida	Datos de la marca.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de marcas.
Acción	Busca en BD de marcas los datos introducidos de la maraca deseada y mostrará por pantalla la información.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se mostrará la información de la marca por pantalla.
Efectos laterales	Ninguno.



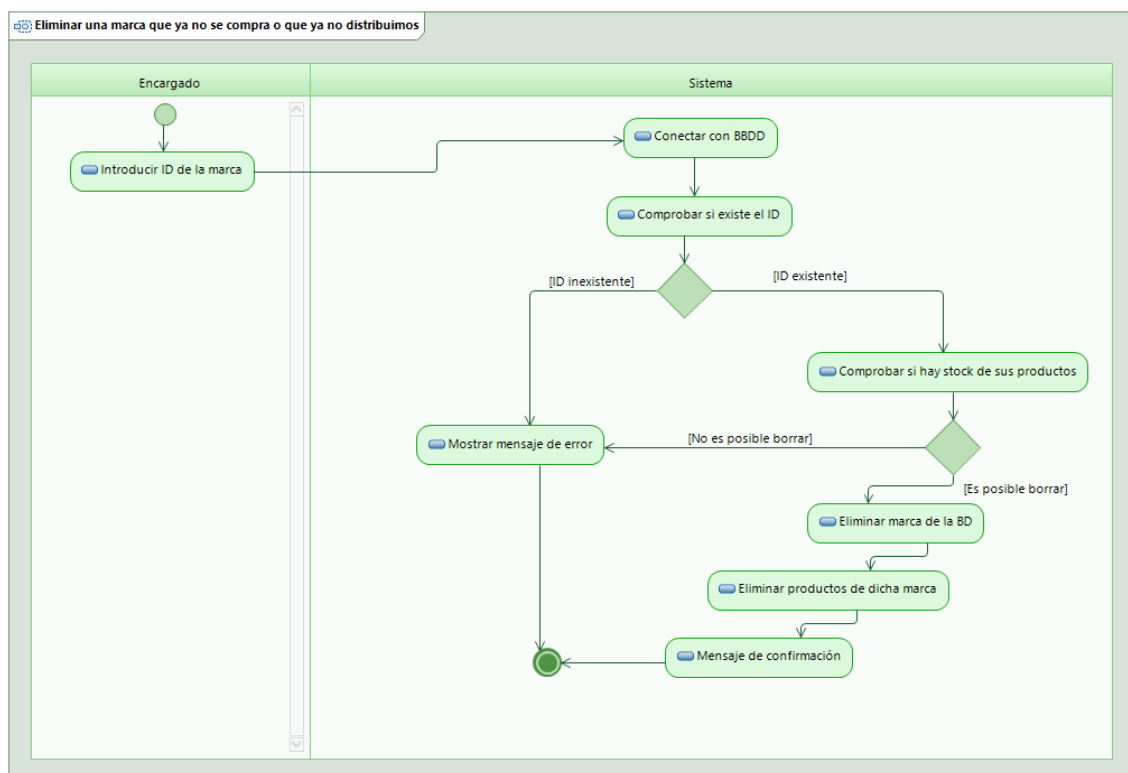
Función 3.2.7.3 Buscar Información de las marcas	
Prioridad	Media.
Estabilidad	Media.
Descripción	Busca y muestra los nombres de las marcas.
Entrada	Ninguna.
Salida	Datos de las marcas.
Origen	Interfaz Usuario – Empleado.
Destino	Sistema.
Necesita	Base de datos de marcas.
Acción	Buscará las marcas y mostrará por pantalla su información.
Precondición	Que la BD no esté vacía.
Postcondición	Que la BD no esté vacía; si tiene éxito, se mostrarán los datos por pantalla.
Efectos laterales	Ninguno.



Función 3.2.7.4 Editar datos específicos de una marca	
Prioridad	Media.
Estabilidad	Media.
Descripción	Modificar los datos de una marca.
Entrada	ID de la marca y los datos a modificar.
Salida	Mensaje de confirmación.
Origen	Interfaz Usuario – Administrar Trabajadores.
Destino	Sistema.
Necesita	Base de datos de marca.
Acción	Busca la marca y se modifican los datos deseados.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se modificará la marca de la BD.
Efectos laterales	Ninguno.



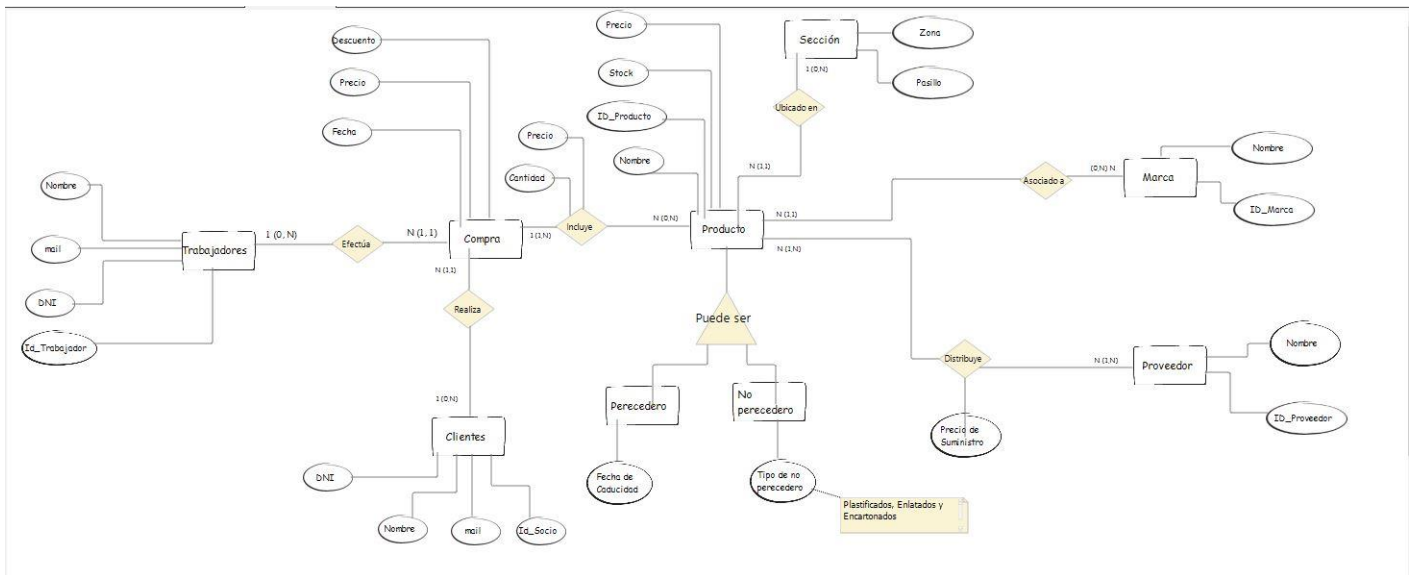
Función 3.2.7.5	Eliminar una marca que ya no se compra o que ya no distribuimos
Prioridad	Alta.
Estabilidad	Alta.
Descripción	Baja lógica de una marca de la BD.
Entrada	ID_Marca.
Salida	Mensaje de confirmación.
Origen	Interfaz Usuario – Administrar Trabajadores.
Destino	Sistema.
Necesita	Base de datos de marcas.
Acción	Introducir datos de una marca, comprobando si existe, si su stock está vacío, y entonces la eliminará.
Precondición	No hay identificadores repetidos en la BD.
Postcondición	No hay identificadores repetidos en la BD; si tiene éxito, se eliminará la marca de la BD y sus productos.
Efectos laterales	Ninguno.



3.3 Requisitos de Rendimiento

El software será implementado en supermercados, para que los clientes puedan realizar el auto compra. Estará adecuado exclusivamente para S.O. Windows (versiones a partir de la 7) con un mínimo de 1 GB de RAM, y 3 GB de almacenamiento.

3.4 Logical Database Requirements (Modelo del Dominio)



3.5 Restricciones de Diseño

El diseño se realizará con el lenguaje UML 2.x el cual desarrollaremos desde la herramienta CASE IBS RSAD.

En cuanto al lenguaje que se usará para la codificación será Java y la aplicación será de escritorio.

Utilizaremos el SCV de la Facultad de Informática (SVN) para gestionar tanto el código como la documentación.

La pertinencia de datos será en formato de texto o en formato relacional. En caso de que sea relacional utilizaremos MySQL.

3.6 Atributos del Sistema Software

- **Mantenibilidad:** Se adapta a los posibles cambios durante su desarrollo y posteriores a su uso.
- **Seguridad:** El sistema debe garantizar la información personal. Solo se debe acceder al sistema de forma autorizada y en un ámbito controlado.
- **Eficiencia:** No desperdiciar recursos ni consumir más de los necesarios.

- **Disponibilidad:** La disponibilidad de nuestro sistema está ligada y condicionada por la que nos ofrezcan nuestros proveedores. Debe estar disponible el máximo tiempo posible, garantizando mantener el sistema en condiciones óptimas en caso de haber errores.
- **Fiabilidad:** Debe garantizar su correcto funcionamiento, evitar los posibles fallos que puedan producirse en el SGBD y la Base de Datos. Para el correcto uso y tratamiento de los datos han de evitarse fallos. de JPA.
- **Portabilidad:** Dado que el sistema será un programa Java, este podrá funcionar en todas las plataformas que dispongan de una versión reciente de la JVM. Además, el gestor de bases de datos MySQL también está disponible para las plataformas que usaremos, y no habrá problemas para la portabilidad del programa. JPA es el encargado de convertir los objetos Java en instrucciones para el Manejador de Base de Datos (MDB).